А.Е.Коваленко, В.В.Гула

ОТКАЗОУСТОЙЧИВЫЕ МИКРО-ПРОЦЕССОРНЫЕ СИСТЕМЫ

Киев «Техніка» 1986

32.973
К56
УДК 621.325.5 : 621.382.049.77

Коваленко А. Е., Гула В. В.

К56 Отказоустойчивые микропроцессорные системы.— К.: Техніка, 1986.—150 с., ил.— Библиогр.: с. 147—149.

В пер.: 65 к. 20000 экз.

Рассмотрены вопросы построения отказоустойчивых самоконтролирующихся цифровых систем на микропроцессорных и больших интегральных схемах. Приведены модели и алгоритмы контроля и диагностирования многомашинных микропроцессорных систем при возможности отказов большого количества элементов. Изложены вопросы проектирования и оценки эффективности отказоустойчивых систем. Описаны способы контроля и диагностирования БИС микропроцессоров и ЗУ.

Рассчитана на инженерно-технических работников, занимающихся разработкой, проектированием и эксплуатацией отказоустойчивых микромашинных систем и сетей.

$К\frac{2405000000\text{-}115}{М202(04)\text{-}86}$ 38.86 32.973

Рецензенты: канд. техн. наук О. В. Викторов, А. В. Кобылинский, канд техн. наук Н. Г. Сабадаш

Редакция литературы по энергетике, электронике, кибернетике и связи
Зав. редакцией З. В. Божко

ПРЕДИСЛОВИЕ

В Основных направлениях экономического и социального развития СССР на 1986—1990 годы и на период до 2000 года подчеркивается необходимость значительного расширения в приборах и средствах автоматизации применения элементной базы повышенной надежности и быстродействия, сверхбольших интегральных схем. Дальнейшее повышение производительности систем на основе широкого использования больших (БИС) и сверхбольших (СБИС) позволяет решить комплекс проблем, связанных с управлением производством, наиболее рациональным использованием энергетических, сырьевых и других материальных и информационных ресурсов, а также кардинально решать целый ряд экологических проблем. Системы на основе БИС и СБИС позволяют создавать распределенные информационно-вычислительные системы и сети, обладающие способностью наращивания и распределения вычислительной мощности, быстрой адаптацией к классу решаемых задач. Мощные многомашинные микропроцессорные системы, содержащие десятки и сотни тысяч микропроцессоров, однокристальных микро-ЭВМ, других БИС и СБИС, обеспечивают высокопроизводительную параллельную обработку информации, что характерно для ЭВМ пятого поколения.

Совершенствование технологии и повышение качества производства БИС и СБИС, других микроэлектронных устройств, использование новых схемно-технических решений не снимает проблем, связанных с возникновением в них отказов. Это обусловлено в первую очередь ростом аппаратной сложности БИС и СБИС, которая достигает уровня 10^6 логических вентилей, а в ближайшее время ожидается увеличение ее до 10^7 вентилей в одном корпусе. Поэтому системы на основе БИС и СБИС должны обладать свойством отказоустойчивости, т. е. нечувствительностью к возникающим в системе отказам. Одним из наиболее рациональных подходов обеспечения отказоустойчивости является автоматическое выявление места отказа с последующим восстановлением работоспособности.

Это позволяет существенно сократить количество обслуживающего персонала, снизить расходы, связанные с эксплуатацией и простоем систем. Наиболее сложным при этом является контроль и диагностирование неисправностей. Характерной особенностью контроля и диагностирования в отказоустойчивых системах, содержащих большое количество БИС и СБИС, является возможность возникновения большого количества неисправностей как в проверяемых частях системы, так и в средствах диагностики.

Основной целью, которую ставили авторы при работе над данной книгой, является изложение основных результатов теории и практики построения отказоустойчивых систем. При этом основное внимание уделено вопросам контроля и диагностирования на различных этапах создания и эксплуатации системы.

Авторы выражают благодарность рецензентам за замечания, способствовавшие улучшению содержания книги.

Главы 1—5 и параграфы 1, 4 гл. 6 написаны А. Е. Коваленко, остальные параграфы — совместно А. Е. Коваленко и В. В. Гула.

Отзывы и пожелания просим направлять по адресу: *252601, Киев 1, Крещатик, 5, издательство «Техніка».*

АРХИТЕКТУРЫ ОТКАЗОУСТОЙЧИВЫХ СИСТЕМ

1. ОБЕСПЕЧЕНИЕ ОТКАЗОУСТОЙЧИВОСТИ СИСТЕМ

Основным требованием к многомашинным системам является высокая производительность, которая зависит от их быстродействия, надежности и продолжительности обмена информацией с внешними устройствами. Создание гибких, многофункциональных систем, способных автономно функционировать при ограниченном участии человека в течение длительного времени, позволяет расширить сферы использования многомашинных систем. Важной проблемой при их эксплуатации является обеспечение отказоустойчивости при наличии неисправностей. Система считается отказоустойчивой или нечувствительной к неисправностям, если ее организация предусматривает устранение последствий неисправностей или отказов элементов и программного обеспечения системы за счет использования аппаратной, информационной и алгоритмической избыточности.

В отказоустойчивых системах необходимо различать отказ элементов системы и отказ системы в целом. Если система обладает свойствами отказоустойчивости, то при отказах элементов она обычно сохраняет работоспособность в отличие от систем, не обладающих свойствами отказоустойчивости.

Отказ элементов отказоустойчивой системы может быть необратимым, т. е. быть, например, следствием возникающих в системе дефектов или носить самоустраняющийся характер. Самоустраняющиеся отказы до момента их идентификации называют перемежающимися отказами. Поскольку в системе возможно, в частности при повреждениях, возникновение состояний, не приводящих ее в неработоспособное состояние, то можно говорить об устранении последствий неисправностей. Переход системы в неисправное состояние в результате, например, повреждения, может через определенное время привести к раннему прекращению ее использования по назначению ввиду возникновения отказа системы или перехода системы в предельное состояние. Предельное состояние отказоустойчивой системы может возникнуть при определенных условиях, например, в случае неработоспособности системы,

возникновения предаварийных режимов работы или перехода в состояние, при котором дальнейшее использование системы нецелесообразно или недопустимо ввиду вредности или опасности при дальнейшей эксплуатации.

Поскольку существует много разных способов обеспечения избыточности в системе, а функции системы ограничены определенным кругом решаемых задач, то часто используется более узкое понятие отказоустойчивых систем. Например, для отказоустойчивых вычислительных систем основной задачей является способность их правильно выполнять заданные алгоритмы при наличии отказов в аппаратуре, ошибках в программе и т. д. Введение в этом случае аппаратурной и информационной избыточности может, например, обеспечить прекращение работы системы при аварийных ситуациях, вычисление по эквивалентным или по альтернативным алгоритмам при обнаружении отказа в системе. Мера, которой оценивают нечувствительность системы к возникающим в ней неисправностям (отказам), называется свойством отказоустойчивости или просто отказоустойчивостью. В качестве оценки меры часто принимают наибольшее количество неисправных элементов, при котором система способна устранить последствия неисправностей.

Избыточность в отказоустойчивой системе используется для организации процесса идентификации отказов и устранения их влияния на правильность выполняемых системой функций. Избыточность в системе обеспечивается, в частности, информационным, структурным, функциональным, временным и нагрузочным резервированием. Прекращение пользования системой при резервировании происходит после исчерпывания имеющихся резервов системы или при переходе ее в предельное состояние.

Попытки обеспечить отказоустойчивость в системах относятся к первому поколению ЭВМ, в которых применялись, в основном, различные средства и методы резервирования с небольшой кратностью резерва: нагруженное постоянное и общее резервирование на уровне процессоров, раздельное резервирование на уровне функциональных плат и узлов, информационное резервирование данных и программ, которые продолжают широко использоваться в современных системах.

Отказоустойчивость, например, в системах SAPO, SAGE, обеспечивалась тройным постоянным резервированием модулей процессора и информационным резервированием данных, хранимых в запоминающих устройствах (ЗУ) [1].

Созданная позднее электронная система коммутации ESS, структура которой показана на рис. 1,*а*, включает быстродей-

ствующий дублируемый центральный процессор и периферийные устройства, непосредственно управляющие коммутацией телефонных каналов [32].

Резервный процессор обеспечивает управление системой при обнаружении отказов в основном процессоре. При этом работать может только один из них. Каждый процессор ESS первого поколения содержит центральное устройство управления *ЦУУ* и два устройства памяти: запоминающее устройство программ *ЗУП* и вызовов *ЗУВ*. В *ЗУП*, выполненном в виде постоянного запоминающего устройства (ПЗУ), нахо-

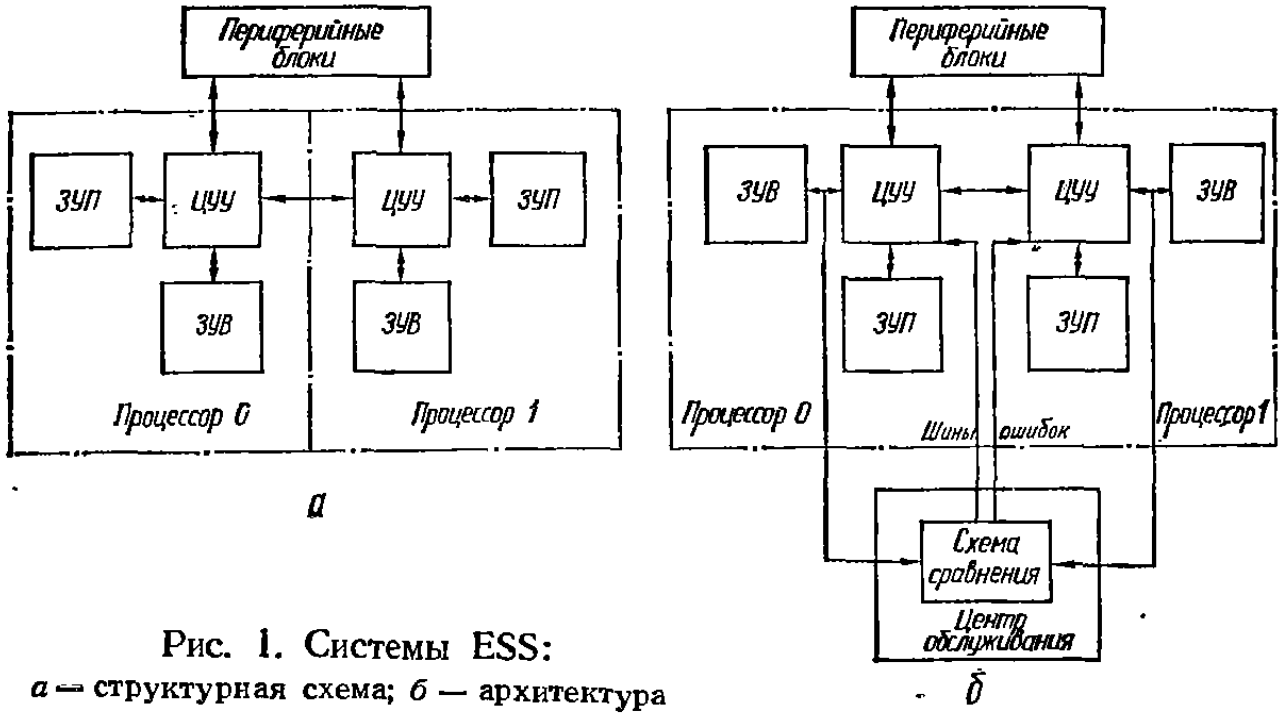


Рис. 1. Системы ESS:
а — структурная схема; *б* — архитектура

дятся программы обработки вызовов, обслуживания системы и программы диспетчеризации, а также параметры системы и ряд рабочих программ. Резервирование на уровне процессоров не обеспечивает высокой надежности при малых аппаратурных затратах. Поэтому в системе ESS второго поколения, структура которой показана на рис 1,*б*, резервирование ведется уже на уровне устройств: *ЗУП*, *ЗУВ*, *ЦУУ*, а также шин вызовов, программ и периферийных блоков.

Центр обслуживания, включающий схему сравнения, производит сравнение данных обмена между *ЗУП* и *ЦУУ*, выдачу по шинам ошибок команд останова резервного *ЦУУ* или прогона программ диагностирования рабочего *ЦУУ*. Локализация неисправностей осуществляется с помощью диагностической программы и встроенной аппаратуры контроля (схем самопроверки), которая контролирует также микропрограммное управление системы. Защита от неисправностей в *ЗУВ* производится дублированием записей в рабочем и резервном ЗУ. При возникновении неисправностей в рабочем

ЗУВ в работу включается резервное. Отказоустойчивость в системах ESS обеспечивается следующими мерами:

обнаружением неисправностей путем сравнения выходов идентичных синхронно работающих блоков;

введением избыточного кодирования (в частности, использованием кодов m из n) цифровой информации в системе;

применением встроенных схем контроля, которые совместно с диагностическими программами позволяют устранять влияние значительной части неисправностей;

использованием таймера аварий, который разрешает перебор возможных конфигураций системы с целью создания работоспособной структуры. Работоспособность выбранной конфигурации процессора определяется с помощью специальной программы;

введением специального канала обслуживания, по которому производится диагностирование одного процессора другим;

автоматическим и ручным восстановлением процессора при обнаружении неисправностей.

Восстановление работоспособности системы при неисправностях в аппаратной части осуществляется отключением рабочего процессора и запуском резервного. Переход на продолжение программы происходит в три этапа: установка состояния системы (задание начального адреса, состояний регистров ошибки, сброс триггеров остановки и др.); защита данных, хранимых в резервном ЗУ; запуск программ. Ручное восстановление используется в тех случаях, когда автоматическое не обеспечивает приведение системы в работоспособное состояние, например, в случае безуспешной многократной инициализации. Способы обеспечения отказоустойчивости описанных систем характерны для систем малой производительности. Повышение производительности в 10 раз и пятикратное увеличение числа обслуживаемых устройств ввода-вывода за счет сокращения времени обслуживания, поиска неисправностей и восстановления работоспособности достигается в сети мини-ЭВМ ARPANET на основе отказоустойчивых мультипроцессоров PLURIBUS [28].

Мультипроцессор состоит из модулей, называемых шинами ввода-вывода, процессорными шинами и шинами памяти. В каждую из шин входит арбитр, соединители и функциональная часть. Функциональной частью процессорной шины являются один или два процессора и местная память, шины памяти — ряд блоков памяти, шины ввода-вывода — интерфейсы, устройства псевдопрерываний (УПП), удлинители шин и интерфейсы связи (рис. 2). Связь между компонентами шины

выполняется по общей шине, а связи между шинами — через соединители. Наиболее часто используемые программы хранятся в местной памяти каждой процессорной шины, а редко используемые — в шинах памяти.

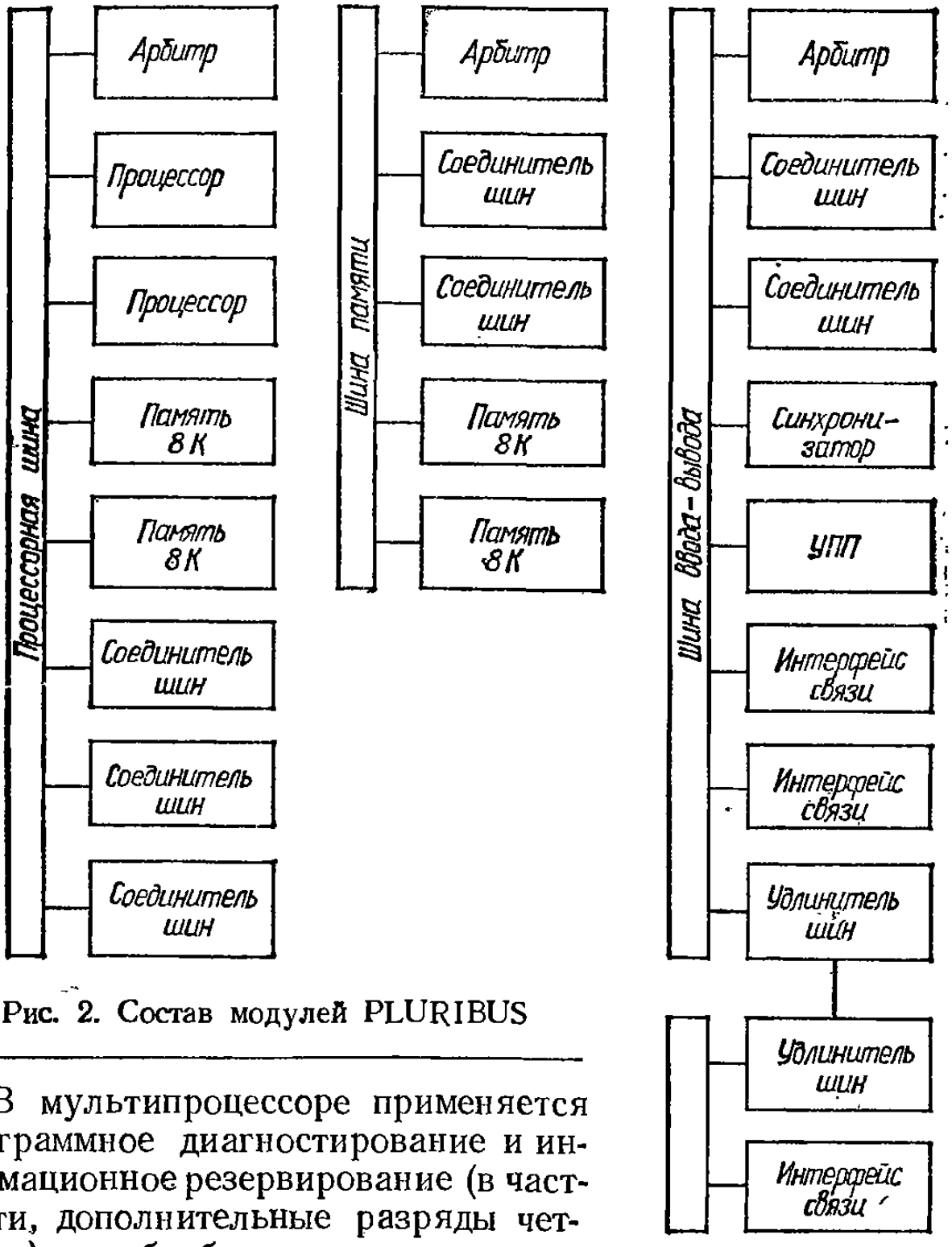


Рис. 2. Состав модулей PLURIBUS

В мультипроцессоре применяется программное диагностирование и информационное резервирование (в частности, дополнительные разряды четности) в обрабатываемых и передаваемых данных. Проверяется четность данных и адресов по всем путям, образованным соединителями, причем дополнительно обнаруживаются также отказы типа «все нули», «все единицы» на общих шинах. Системное диагностирование выполняется с помощью специальных программных тестов. При обнаружении неисправностей программное обеспечение использует избыточность аппаратных средств для образования новой логической конфигурации системы, в которой

отсутствуют отказавшие блоки. Предусмотрены также специальные переключатели, позволяющие отключать отдельные шины. При этом подключение и блокировка трактов доступа производится любым процессором, который может принимать решение о неисправности контролируемого процессора.

Мультипроцессор имеет систему защиты от тяжелых остановов и восстановления работоспособности при отказах в аппаратуре, ошибках в программном обеспечении, построенную на основе операционной системы STAGE. Эта система поддерживает достоверную карту текущего состояния имеющихся программных и аппаратных ресурсов (как своих, так и прикладной операционной системы).

Карта ресурсов составляется на основе многоэтапного испытания и проверки правильности работы системы с помощью ресурсов, проверенных на предшествующих шагах (примитивов операционной системы). Для координации работы различных процессоров при определении конфигурации системы используются три массива соглашений: «следующий», «сглаженный» и «фиксация», в которых каждому из процессоров соответствует свой разряд. Перед началом соглашения запросы посылаются процессорами в массив «следующий», который переносится в «сглаженный» через время, достаточное для определения состояния участвующих процессоров. Состав процессоров, участвующих в работе системы, определяется картой конфигурации. Процессор, модифицирующий информацию о конфигурации, записывает единицу в свой разряд массива «фиксация». Процессоры, согласные с картой конфигурации, сбрасывают свои разряды. Сбрасываются также разряды, соответствующие процессорам, не вошедшим в массив «сглаженный». Перед проведением фиксации все процессоры ждут, чтобы массив «фиксация» совпал с массивом «сглаженный», обеспечивая тем самым возможность принятия согласованной карты конфигурации. Решение о модификации карты конфигурации может быть принято также по большинству процессоров.

При обнаружении отказа процессор устанавливает в единицу свой разряд в массиве «фиксация», блокируя выполнение следующих этапов. При достаточно большом количестве процессоров, обнаруживших отказ, происходит реконфигурация в описанном выше порядке. При несоглашении неисправного процессора с предложенной конфигурацией он зависает в точке обнаружения несоответствия. Однако в дальнейшем он вновь может включиться в работу.

В мультипроцессоре имеется возможность восстановления структур данных и прикладных программ, причем для ради-

кального изменения структур данных (например, полной повторной инициализации прикладной системы) используется описанный механизм соглашения по запросам проверяющих программ. Для проверки данных в процессе функционирования наряду с контролем по избыточной информации используются контрольные таймеры, проверяющие использование этих данных. Так, если буфер процессора в процессе вычисления не запрашивается более двух минут, то выдается принудительный сигнал на переход его в разряд свободных для использования. В результате повышается эффективность загрузки пространства памяти.

Доступ оператора в процессе ремонта и восстановления мультипроцессора предусматривается лишь в том случае, когда его собственные диагностические возможности оказываются недостаточными. С этой целью в системе используется «фотографирование» большей части процессора в момент прерывания. Эта часть включает содержимое регистров общего назначения, время системы, признаки установки регистров отображений и др.

Опыт эксплуатации системы ARPANET подтвердил ее высокую готовность (до 99,7—99,9 %). При этом обеспечивалось более 92 % номинальной производительности в течение 99,76 % обычного времени и более 50 % производительности в течение 99,83 % обычного времени. Время восстановления системы при таких отказах, как потеря физической общей памяти, искажение данных и программ составляет 10—15 с. Таким образом, даже частичное обеспечение отказоустойчивости системы дает значительное улучшение эксплуатационных параметров радиоэлектронных систем.

Известно, что одним из недостатков ЭВМ третьего поколения является большой разрыв между потенциальным и реальным быстродействием. Опыт эксплуатации таких машин показывает, что при потенциальном быстродействии 10^5—10^7 операций/с реальное быстродействие с учетом простоев составляет иногда 10^3—10^6 операций/с, что существенно снижает эффективность их использования.

Применение микромашинных систем с малым количеством микропроцессоров (МП) и микро-ЭВМ позволяет обеспечить реальную скорость выполнения команд до 10^4—10^6 операций/с и выше, а при большом их количестве повысить реальное быстродействие до 10^8—10^{10} операций/с.

Например, минимашинная система управления (СУММА) позволяет потенциально обеспечить быстродействие 10^8—10^9 операций/с [8, 10]. Минимальная программно-коммутируемая система (МИНИМАКС) позволяет достаточно просто наращивать вычислительную мощность путем увеличения количества

функционирующих вычислительных машин без существенного изменения структуры системы.

Работы по созданию отказоустойчивых систем ведутся и за рубежом. Некоторые из них успешно эксплуатируются, что описано в соответствующей литературе [1, 6, 14, 16, 27, 29, 32—36, 40, 47, 54].

Применение МП и микро-ЭВМ в многомашинных системах не снимает проблему обеспечения их надежного функционирования. Ввиду ограниченного доступа к МП и микро-ЭВМ (обычно число внешних контактов в них не превышает 40—200), их большого количества и высокой аппаратурной сложности усложняется задача поиска возникающих неисправностей. Кроме того, при большом количестве МП и микро-ЭВМ, несмотря на их высокую надежность, в системе возможен отказ нескольких элементов. А поскольку при диагностировании системы могут принимать участие и неработоспособные элементы, то для обеспечения отказоустойчивости необходимо, чтобы система была способна устранять их влияние на результаты контроля и диагностирования. Способность системы выполнять эти функции определяется в первую очередь ее структурной организацией.

2. АРХИТЕКТУРЫ МНОГОМАШИННЫХ СИСТЕМ

Выбор архитектуры многомашинной системы связан с решением следующих основных задач: распределение вычислительной мощности и информации в системе таким образом, чтобы в наибольшей степени обеспечить выполнение целевых функций системы с допустимой задержкой;

организация логических и физических связей в системе с целью обмена информацией между компонентами системы с необходимой пропускной способностью;

разработка программного обеспечения (системного и прикладного), осуществляющего эффективное накопление и обработку данных, взаимодействие компонентов системы (процессоров, запоминающих устройств, периферийных устройств и др.) между собой, контроль и диагностирование, восстановление работоспособности в процессе функционирования.

Иногда задают требования расширения системы по составу оборудования в процессе эксплуатации и совершенствования технических показателей системы: производительности, надежности, пропускной способности и др. Удовлетворение этих требований также основывается на решении перечисленных задач. На решение основных задач по созданию многомашинных систем существенно влияет и отражается на отказоустойчивости системы количество и порядок соединения основных

компонентов в системе, т. е. структура связей в системе. Распределение информации и вычислительной мощности в системе зависит от скорости поступления данных от источников информации (датчиков, первичных преобразователей, органов управления), быстродействия внешних устройств, обслуживаемого объекта, использующего результаты обработки данных системой, протоколов обмена данными в системе, возможностей используемых технических средств и т. д.

Современный подход к анализу архитектур многомашинных систем состоит в том, что система рассматривается как совокупность элементарных машин (ЭМ), выступающих в качестве элементов системы, которые могут вступать во взаимодействие между собой в процессе функционирования. В качестве ЭМ могут использоваться МП, микро-ЭВМ, универсальные и управляющие ЭВМ, другие подсистемы, специализированные устройства (контроллеры, ЗУ со схемами управления и др.). Взаимодействие состоит в обмене управляющей и обрабатываемой информациями с целью выполнения задач, поставленных перед системой. В процессе выполнения задач возможно объединение нескольких ЭМ на заданном временном интервале, выделение среди работающих ЭМ ведущих и ведомых, исключение из состава некоторых ЭМ и включение новых, не участвовавших до этого в работе. Например, в одной из первых отечественных многомашинных вычислительных систем МИНИМАКС в качестве вычислительных модулей используются управляющие вычислительные машины М-6000, М-7000, а связи между процессорами осуществляются с помощью модуля межмашинной связи [10, 16]. Элементарной машиной служит пара, состоящая из вычислительного модуля (ВМ) и модуля межмашинной связи (ММС). Структура связей в системе определяется конкретными требованиями производительности, реконфигурации и наращиваемости. Структура такой ЭМ показана на рис. 3. Двумерные связи *1* обеспечивают взаимодействие между *ЭМ*, одна из которых ведущая, остальные — ведомые. По связям *1* могут пересылаться, например, тестовые данные, адреса, массивы информации. Одномерные связи *2* служат для обмена данными *ЭМ* с соседними (например, смежными слева и справа). Алгоритм взаимодействия по связям *1* и *2* зависит от конкретного технического

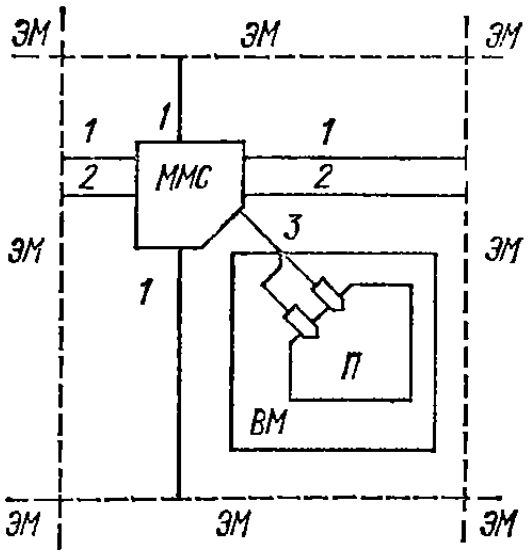


Рис. 3. Структура элементарной машины системы МИНИМАКС

решения *ММС*. Связи *3* между *ММС* и *ВМ*, в который входит процессор *П*, позволяет производить обмен в рамках интерфейса 2к.

Основное преимущество системы МИНИМАКС состоит в гибкой перестройке двумерных структур с общей одномерной связью между ЭМ. Исключение из системы или введение в систему ЭМ осуществляется подстройкой (программированием) соответствующих ММС без изменения всей структуры связей в целом. Вместе с тем подключение большого количества ЭМ данного типа по общей шине экономически неоправдано и затруднительно, поскольку требует дополнительной доработки аппаратуры сопряжения и интерфейса. В этом отноше-

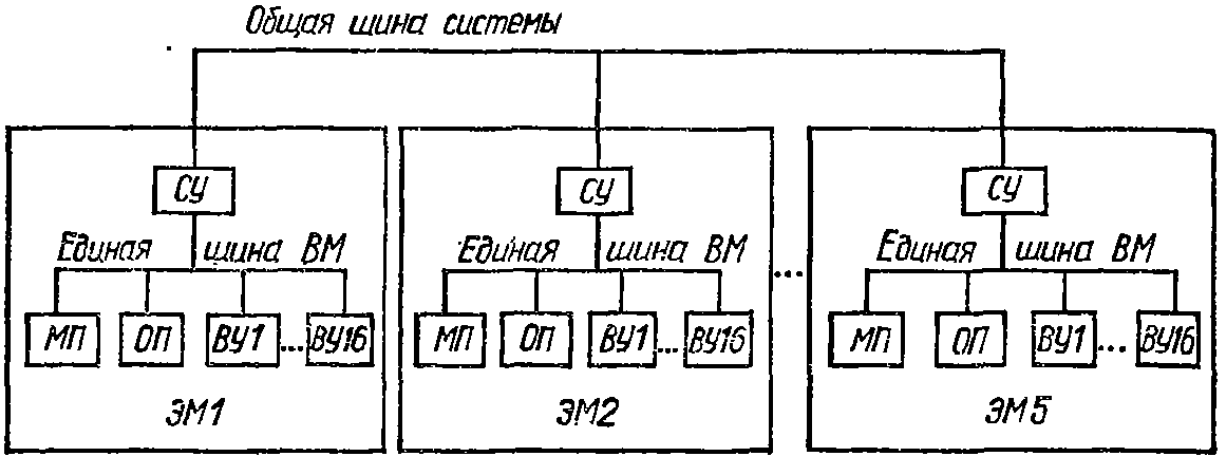


Рис. 4. Общая структура системы «Статистик-10»

нии существенный выигрыш достигается за счет применения МП и микро-ЭВМ.

На основе отечественных высокопроизводительных микро-ЭВМ (в частности, «Электроника К1-20», «Электроника-80») можно создавать достаточно простые многомашинные системы обработки информации с большим количеством ЭМ. Примером может служить система «Статистик-10» (рис. 4), предназначенная для автоматизации решения задач учета, планирования и управления.

Элементами нижнего уровня системы являются элементарные машины *ЭМ1*, ..., *ЭМ5*, объединяемые через общую шину в подсистему более высокого уровня с помощью системных устройств *СУ*. Элементарная машина включает микро-ЭВМ «Электроника-60», в которую входят микропроцессор *МП*, оперативная память *ОП* и внешние устройства *ВУ1*, ..., *ВУ16*, объединенные общей шиной. В систему можно подключать до 15 ЭМ, причем исключается ЭМ переводом соответствующих выходов в третье состояние (высокое сопротивление). Это позволяет достаточно просто наращивать вычислительную мощность и исключать неисправные ЭМ из системы.

Более широкими возможностями структурной организации обладает созданная в СССР микропроцессорная однородная система МИКРОС [16]. Основное отличие этой системы от предшествующих состоит в возможности свободной реконфигурации с помощью модуля системного устройства (МСУ), обеспечивающего связь каждой ЭМ с другими ЭМ, количество которых может быть от 1 до 12. Модуль системного устройства, структура которого показана на рис. 5, служит для выполнения операций приема и передачи данных, приема слова состояния по одному из каналов. Для этих операций исполь-

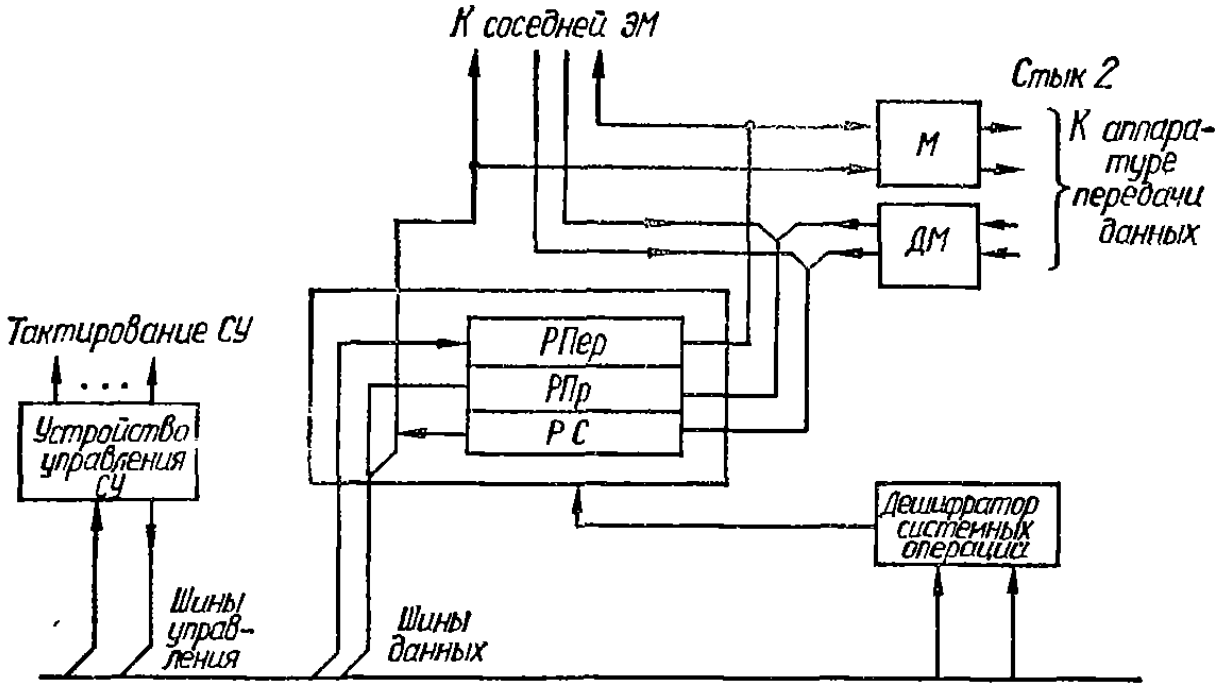


Рис. 5. Модуль системного устройства

зуется регистр передачи *РПер*, регистр приема *РПр*, регистр слова состояния *РС*, модулятор *М* и демодулятор *ДМ* под управлением *Устройства управления СУ*.

Модулятор и демодулятор служат для преобразования данных из параллельного в последовательный код и наоборот, а также для синхронизации сигналов прием-передача данных и прерываний микро-ЭВМ. Взаимодействуют ЭМ между собой под управлением операционной системы, основой которой является программное ядро ЭМ, обеспечивающее загрузку программ и данных, мультипрограммную работу и реализацию алгоритмов системных взаимодействий. Такая организация обеспечивает полное использование ЭМ, и как следствие, высокую производительность системы.

В системе МИКРОС, таким образом, достигается гибкое распределение вычислительной мощности и занятости ЭМ при одновременном расширении возможностей реконфигурации. При использовании систем МИНИМАКС и МИКРОС возможны различные структуры связей ЭМ, однако стремятся вы-

бирать наиболее простые. Так, например, если можно разделить процесс вычисления на последовательные этапы, каждый из которых выполняет одна ЭМ, то структура системы в простейшем случае представляет собой последовательно связанные между собой ЭМ (система конвейерного типа). Однако при отказе одной из ЭМ происходит останов системы. Поэтому с точки зрения обеспечения отказоустойчивости целесообразно использовать максимальное количество связей между машинами. В результате ЭМ, свободные от вычислений в данный момент, могут проверять другие свободные машины или участвовать в поиске дефектов в ЭМ.

Таким образом, выбор структурной организации связей между компонентами многомашинной системы в значительной степени предопределяет не только организацию физических и логических связей в системе, распределение вычислительной мощности и памяти на этапе проектирования, но и отказоустойчивость системы в процессе эксплуатации.

С точки зрения унификации программного обеспечения и упрощения обслуживания целесообразно, чтобы структура многомашинной системы была симметричной. Вместе с тем особенности обслуживаемого объекта (неравная удаленность и необходимая пропускная способность между отдельными ЭМ, различие выполняемых ЭМ функций, обусловленное, например, структурой обработки данных), разные типы используемых ЭМ и другие факторы требуют в ряде случаев использования систем с несимметричной, нерегулярной структурой связей.

Опыт разработки многомашинных систем позволил выделить типы систем для практического использования. На рис. 6 и в табл. 1 на основе данных [5, 8], показаны основные типы многомашинных систем с различной структурной организацией, а также их наиболее важные качественные и эксплуатационные характеристики.

В дальнейшем обозначения структурной организации системы, приведенные в табл. 1, будут использованы в качестве типа системы (например, система С7 — система, имеющая архитектурную организацию «глобальная шина»). Потенциальная производительность большинства из перечисленных типов систем составляет 1—3 Мбит/с, причем часть систем эксплуатируется.

Показанные на рис. 6 системы неравноценны с точки зрения обеспечения их отказоустойчивости. Так, например, в системе со звездообразной организацией связей между ЭМ (система С8) отказ переключателя приводит к полному отказу системы. Отказ одной ЭМ в большинстве приведенных систем не приводит к отказу системы в целом. Однако ввиду различ-

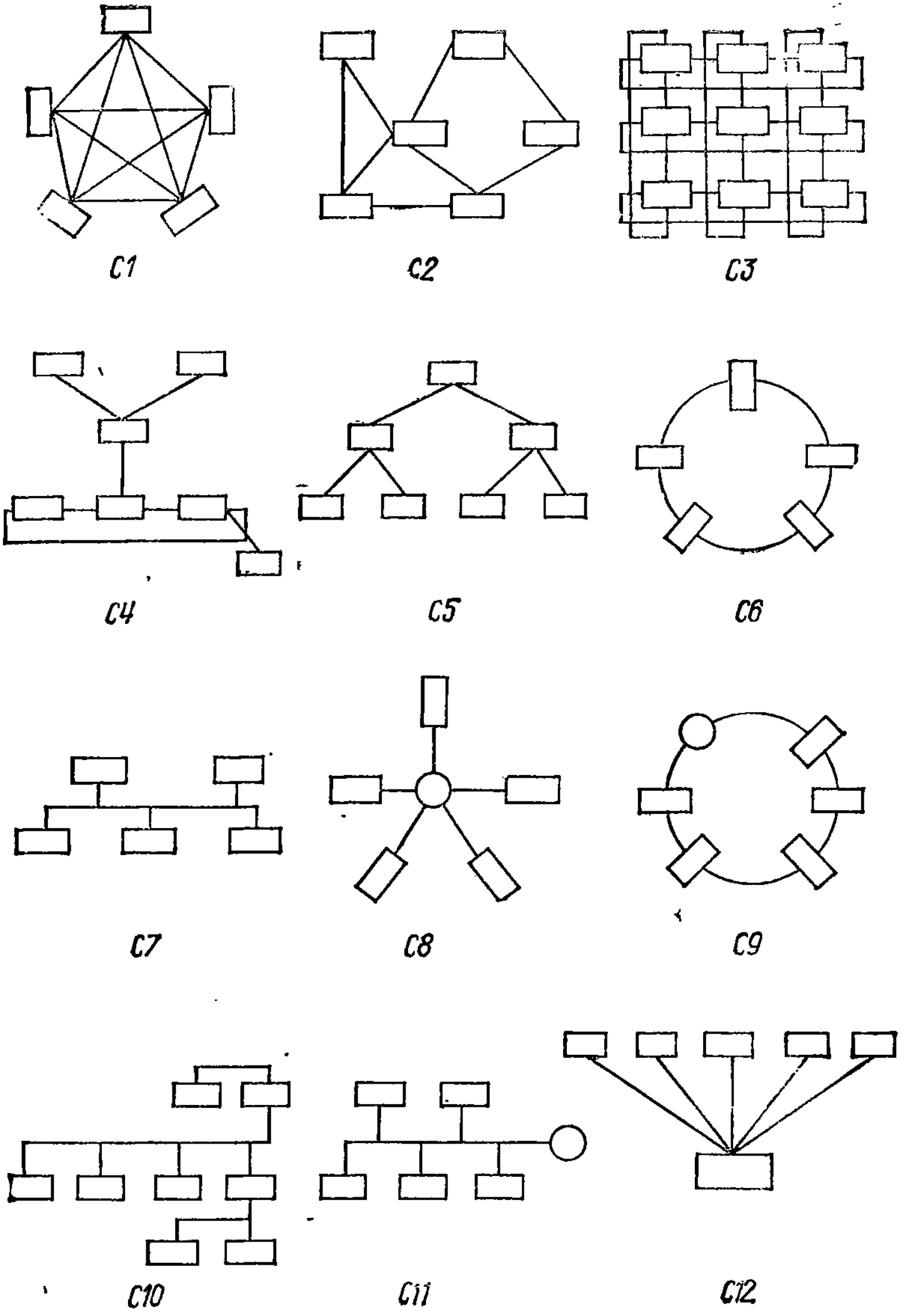


Рис. 6. Типовые структуры многомашинных систем:
○ — переключатель; ▭ — ЭМ

Таблица 1

Архитектурная организация системы			Надежность	Стоимость	Возможность развития	Распределенность
Обозначение	Наименование	Технология соединения				
C1	Полносвязная	Каждая ЭМ соединена со всеми остальными по выделенному каналу	Система продолжает работать при отказе ЭМ. При отказе основных линий связи могут использоваться резервные пути передачи данных	Высокая, быстро растет с увеличением количества ЭМ и расстояния между ними	Очень хорошая	Неограниченная
C2	Сеть с коммутацией пакетов	Сообщения, разбитые на пакеты, передаются через доступные узлы. Между двумя ЭМ существует минимум два пути	Система продолжает работать при отказе одной ЭМ	Высокая, требуется маршрутизация пути	Хорошая	То же
C3	Регулярная сеть	Каждая ЭМ соединена с соседними	Выход одной ЭМ не влияет на функционирование сети в целом. При отказах линий связи могут использоваться резервные пути передачи данных	Высокая	Плохая	Может быть неограниченной
C4	Нерегулярная сеть	Структура ЭМ зависит от требуемой топологии связи	Частичное резервирование при отказе линий связи	Средняя, зависит от расстояния между ЭМ	Очень хорошая	Неограниченная
C5	Иерархическая	Иерархическая структура обмена данными и распределения функций по уровням	Неравноценность отказов ЭМ верхних и нижних уровней. Отказ на более высоких уровнях более существенный	То же	Хорошая	То же
C6	Петлевая или кольцевая	Каждая ЭМ подсоединена к двум соседним. Потоки могут проходить как в одном (однонаправленные), так и в двух	Низкая, устойчива к отказу одной линии при двойной петле и неработоспособна в случае однонаправленной магистрали	Средняя	»	Ограниченная

		направлениях (двунаправленные)				
С7	Глобальная шина	Используется общая (глобальная) шина, по которой осуществляется взаимодействие между ЭМ	Отказ при отказе общей шины. Нечувствительна к отказам отдельных ЭМ при наличии соответствующих схем изоляции и защиты	Средняя	Хорошая	Ограниченная
С8	Звездообразная	Взаимодействие между элементами проходит через общий переключатель	При отказе переключателя система становится неработоспособной. Отказ ЭМ при наличии защитных средств не препятствует дальнейшей работе системы			
С9	Петля с переключателем	Аналогична петлевой. Переключатель удаляет сообщения из петли, преобразует и заменяет адреса, направляя к нужному приемнику	Система чувствительна к отказам переключателя или петли	Средняя, наибольшая стоимость переключателя	Хорошая, очень хорошая пока ие перегружен переключатель	»
С10	Окно шины	Используется несколько переключателей, сообщения передаются в любом направлеиии по свободным шинам	Частичный отказ системы при отказе отдельных переключателей или шин	Низкая, наибольшая стоимость переключателя	Плохая	Очень ограниченная
С11	Шина с переключателем	Каждая ЭМ соединена с переключателем. Поток данных идет от ЭМ-источника к ЭМ-приемнику. ЭМ разделяют шину для доступа к переключателю	Чувствительность к отказу шины или переключателя	То же	Хорошая, очень хорошая пока не перегружен переключатель	»
С12	С разделяемой памятью	Взаимодействие между ЭМ происходит через память, доступную всем ЭМ	Чувствительиость к отказу общей памяти	»	Плохая, ограничена числом портов	Очень ограниченная

ного характера связей между элементами, как правило, требуется обеспечение специальных средств изоляции неисправной ЭМ. Последнее зависит от функций, выполняемых ЭМ как элемента системы (например, вычислительный элемент, вычислительный элемент с коммутацией входных и выходных каналов, элемент реконфигурации системы). Линии связи между ЭМ имеют неодинаковую нагрузку при работе для различных структур. Так, например, в системах, имеющих структуру С7, С11 («глобальная шина», «шина с переключателем») связь по общей шине обеспечивает взаимодействие между всеми ЭМ. В результате выполнение функций контроля, диагностирования и восстановления в таких системах должно быть разделено во времени.

Некоторые особенности обеспечения отказоустойчивости многомашинных систем с различной структурной организацией приведены в табл. 2, из которой следует, что наиболее отказоустойчивыми являются системы С1, С2, однако они требуют большего количества связей. Естественно также ожидать, что программное обеспечение и протоколы обмена, лежащие в основе взаимодействия ЭМ, в таких системах будут сложнее по сравнению, например, с системой С7 (например, система МИКРОС) или С11.

Выявление неисправности без последующего исключения из системы дефектных узлов приводит к накоплению дефектов и переходу системы в непредсказуемые технические состояния. Для поддержания работоспособности в этом случае необходима большая аппаратурная избыточность (например, дублирование или троирование).

Применение в отказоустойчивых системах специальных внешних средств контроля и диагностирования имеет недостатки, к которым относятся:

сложность проверки внешними средствами системы в динамическом режиме;

необходимость введения в систему специальных контрольных выводов и изменения, в ряде случаев, алгоритмов работы и протоколов системы;

влияние необнаруженных отказов в средствах контроля и диагностирования на результаты проверки;

необходимость в организации связей с ЭМ при проверке. Поскольку необходим автоматический режим проверки, то число этих связей, которые также необходимо контролировать, соизмеримо с количеством ЭМ.

При поиске дефектов в системе с помощью встроенных средств диагностирования обычно проверяют наличие ошибок входных данных ЭМ (в противном случае возможно ложное выявление дефектов), а затем правильность сигналов синхро-

Таблица 2

Тип архитектуры	Аппаратные и структурные средства обеспечения отказоустойчивости системы	Программные и информационные меры по обеспечению отказоустойчивости системы
С1	Реконфигурация системы при отказах линий связи и ЭМ (изоляция, обход отказавших элементов, частичное дублирование линий связи, обеспечивающее связность ЭМ)	Наличие программ и реконфигураций в каждой ЭМ, изменение в программном обеспечении адресации и состава ЭМ
С2	Средства отключения и изоляции неисправных линий связи и ЭМ	Контроль и изменение пути следования передаваемых пакетов при отказах (проверка правильности назначений, маршрутизация с учетом отказов в узлах и линиях связи), проверка контрольных сумм
С5	Применение переключающего процессора по уровням иерархии и альтериативных ЭМ для замены отказавших Введение перекрестных связей между ЭМ смежных уровней. Дублирование, защита линий связи особенно для верхних уровней Применение на верхних уровнях ЭМ с развитой системой самоконтроля и самодиагностирования	Обеспечение временного хранения данных отказавших ЭМ нижнего уровня для передачи в ЭМ верхних уровней Наличие основных функциональных программ в программном обеспечении каждой ЭМ
С6	Устранение влияния, с помощью реконфигурации, отказов линий связи кольца и петлевых (кольцевых) интерфейсов, через которые осуществляется связь с ЭМ. Применение обхода (закорачивания) петли в месте подключения петлевого интерфейса, изоляция петлевого интерфейса от шины. При наличии запасной двойной петли для однонаправленных петель применяют обход петлевых интерфейсов через резервную часть петли; для двунаправленных — исключение отказавших петлевых интерфейсов. Создание иерархических многопетлевых структур, соединенных через интерфейсы	Программное обеспечение реконфигурации системы, контроля и изоляции ЭМ, проверки линий связи. Применение контрольных разрядов в передаваемых сообщениях
С7	Изоляция отказавшей ЭМ от шины, применение запасных (резервных) шин, дублирование шин	Резервирование временных тактов для контроля и диагностирования, частичного уплотнения для связи ЭМ-передатчика с

Продолжение табл. 2

Тип архитектуры	Аппаратные и структурные средства обеспечения отказоустойчивости системы	Программные и информационные меры по обеспечению отказоустойчивости системы
		ЭМ-приемником (настроенных на одну частоту)
С8	Резервирование переключателя, использование в качестве переключателя мощной ЭВМ с расширенными возможностями самодиагностирования и восстановления, изоляция от отказавших ЭМ и линий связи (путем переадресации, реконфигурации и т п.)	Дублирование обслуживающих программ переключателя, наличие программ обмена по избыточным входным каналам связи переключателя, программные средства самодиагностики и самоконтроля переключателя, контроля отклика связи с ЭМ
С12	Для шин с разделением времени — резервирование шин, общих для всех ЭМ	Программы резервного подключения (например, путем переадресации с переходом на исправные участки памяти, многоступенчатого переключения с резервированием). Применение разделения общей памяти между ЭМ

низации, питающих напряжений, функциональных частей ЭМ. Для этого в систему вводятся специальные схемы контроля, по выходным сигналам которых отдельные функциональные узлы или система в целом могут переходить в заранее определенное техническое состояние (например, останов с индикацией отказавшего узла). Ввиду возможности отказа встроенных и внешних средств контроля и диагностирования использование их для выявления дефектов в многомашинных системах недостаточны. Вместе с тем использование взаимодействия ЭМ не обеспечивает часто требуемую глубину диагностирования и полноту контроля. Поэтому комплексное решение проблемы обеспечения отказоустойчивости многомашинных систем состоит в применении известных методов и средств внешнего и встроенного контроля, диагностирования и использования возможностей взаимодействия ЭМ для выявления дефектов и устранения последствий неисправностей.

К наиболее распространенным известным средствам и методам относятся: контроль по четности; помехоустойчивое избыточное кодирование передаваемых и обрабатываемых данных; применение параллельно работающих функциональных блоков с последующим выбором результата по большинству с помощью схем голосования; повторная многократная передача сообщений; использование парафазных сигналов

при обмене и обработке данных; самоконтролирующиеся функциональные узлы (регистры, арифметико-логическое устройство, запоминающее устройство и т. п.).

3. ОТКАЗОУСТОЙЧИВЫЕ СИСТЕМЫ НА ОСНОВЕ МИКРОПРОЦЕССОРОВ И БИС

Большинство современных отказоустойчивых систем являются микропроцессорными, т. е. содержат МП и микросхемы микропроцессорных комплектов. Характерной особенностью таких систем является использование, наряду с МП, других больших интегральных схем, которые позволяют расширить функциональные возможности и повысить производительность многомашинных систем.

Одними из первых отказоустойчивых многомашинных систем типа С12 с разделением памяти являются системы С. mmp и С. m [12]. Отказоустойчивость в них обеспечивается за счет использования различных встроенных аппаратных и программных средств контроля и диагностирования.

В *мультипроцессорной системе* С. mmp (рис. 7) процессоры *П1...П16*, построенные на основе мини-ЭВМ РДР-11, через блоки *БОА* и матричный коммутатор *КМС* имеют доступ к разделяемой памяти на основе портов общей памяти *ПП1...ПП16*.

Обмен сигналами между процессорами системы осуществляется по шине *МПШ*, через которую производится общее тактирование, прерывание и управление на межпроцессорном уровне. В системе предусмотрены аппаратные и программные средства контроля. Аппаратный побайтный контроль по четности процессора позволяет, в частности, выявить ошибки в системе синхронизации. Некоторые проверки производятся *КМС*. Так, признак четности адреса проверяется в интерфейсе коммутатора, а образование и проверка признаков четности данных — в интерфейсе между коммутатором и шиной, входящим в блок *БОА*.

Перемежающиеся отказы (обращение к несуществующей области памяти, ошибки в стеке, приводящие к неправильному выполнению команд вызова (возврата) подпрограмм, выходу в прерывание и др.) обнаруживаются путем комбинированного использования аппаратных и программных средств, а также программного восстановления. Различают восстановление системы для двух групп отказов: первая для частых отказов (например, перемежающихся) и вторая для редких, но существенных отказов, которые могут привести к полному останову системы.

К основным отказам первой группы относятся отказы межпроцессорного прерывания, выход за границы прямого доступа

к памяти, нарушение признака четности на страницах, предоставленных пользователю. Ошибки при потерянном прерывании исправляются при перезаписи из маски в регистр запроса на прерывание. При выходе за границу памяти в режиме прямого доступа к памяти производится пятикратное повторение обращения внешних контроллеров. Затем во все процессоры, кроме обращающихся, посылаются сигналы прерывания на выполнение программ, связанных с местной памятью.

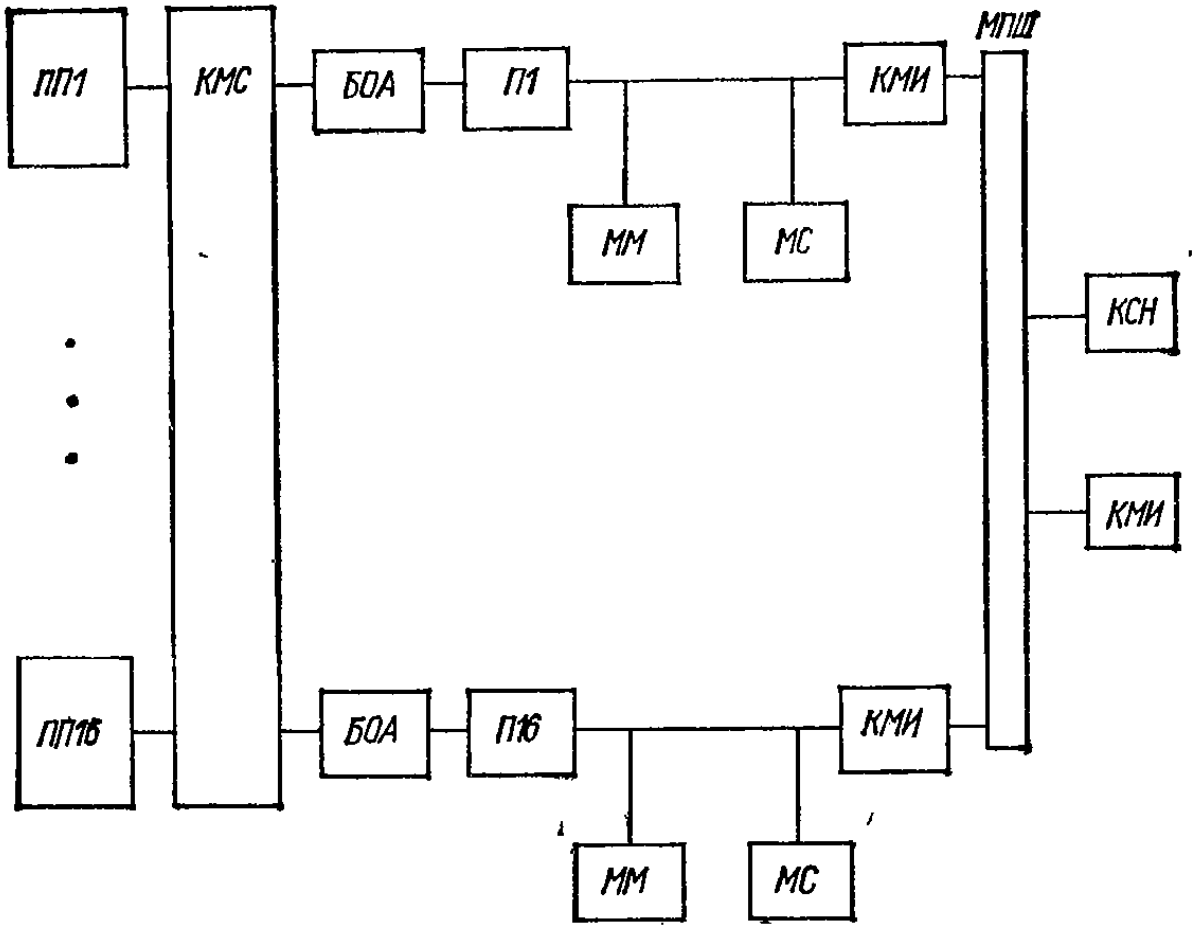


Рис. 7. Структура системы С. mmp:

ПП1—ПП16 — порты памяти; *П1—П16* — процессоры; КМС — матричный коммутатор системы; *БОА* — блок отображения памяти; *КМИ* — контроллер межшинного интерфейса; *КСН* — контроллер синхронизации; *ММ* — локальная местная память; *МС* — страничная память; *МПШ* — межпроцессорная шина

Нарушение признака четности страниц выявляет программа поиска ошибок, которая выдает номера отказавших ячеек и адрес выполняемой команды. Это обеспечивает, с помощью следящих регистров, восстановление логической страницы в другой физической странице. Отказы второй группы, вызванные общесистемными ошибками, восстанавливаются с помощью механизма «подозрение/наблюдение», при котором делается пауза для прихода системы в известное состояние. Вызывается механизм двумя способами.

Первый способ — некоторый процессор проводит поиск ошибок в своем функционировании. При обнаружении оши-

бок процессор становится подозреваемым на отказ. Среди остальных процессоров выбирается наблюдающий, который осуществляет дальнейшее диагностирование и, в случае необходимости, замену подозреваемого процессора.

Второй способ — один наблюдающий процессор (сторож) программно проверяет подозреваемый на отказ процессор. Первоначально производится синхронизация этих двух процессоров. При отсутствии синхронизации наблюдающий процессор инициирует подозреваемый к выполнению программы восстановления, в частности, последовательности операций микропроцессорной шины. Сообщение о завершении операции передается наблюдающему процессору посредством общей переменной состояния. Если синхронизация не достигается, то наблюдающий процессор организует повторную загрузку.

При достижении синхронизации наблюдающий процессор контролирует выполнение последовательности операций подозреваемым процессором. Если не соблюдаются временные интервалы выполнения этих операций, то производится повторная загрузка. По результатам анализа выполнения последовательности операций наблюдающий процессор принимает решение о продолжении работы, исключении из работы, повторной загрузке и «успокоении» подозреваемого процессора. «Успокоение» разрешает обработку прерывания только от устройства ввода-вывода. Повторную загрузку системы может выполнять также механизм повторного запуска. Механизм включает корректировку масок конфигурации с учетом исключенных и «успокоенных» процессоров, составление списка свободной памяти (удаление страниц, содержащих ошибки) и загрузку копии ядра операционной системы.

Система восстанавливается также с использованием диагностического монитора, который запускает диагностическую программу в исключенном из конфигурации процессоре, с распечаткой получаемых результатов. При отсутствии ошибок в течение определенного времени монитор автоматически возвращает исключенный процессор в систему.

Экспериментальная проверка системы C. mmp показала низкое значение среднего времени между отказами (2,9—16,5 ч), значительную часть которых (до 37 % от общего количества) составили ошибки неизвестного происхождения.

Мультимикропроцессорная система C. m состоит из вычислительных модулей *ВМ*, соединенных между собой по группам общими шинами, а группы между собой — контроллером отображения *К* (рис. 8, *а*). Память каждого *ВМ* общедоступна другим *ВМ*. На рис. 8,*а* показана структура вычислительного модуля на основе ЭВМ LSI-11.

Использование структуры из групп *ВМ* приводит к сокращению количества контроллеров *К*, позволяет обеспечить взаимоисключающий доступ к структурам данных памяти.

В контроллер *К* введены специальные аппаратные средства — «ловушки», используемые при аппаратном и микропрограммном диагностировании. В коммутаторе *КМ* имеются регистры для хранения диагностической информации от контроллера *К*. При наличии ошибки дальнейшее выполнение следующей команды блокируется до ликвидации ошибки. При наличии ошибки предусмотрен прием сообщения по другому физическому тракту через контроллер *К* (рис. 8,*б*).

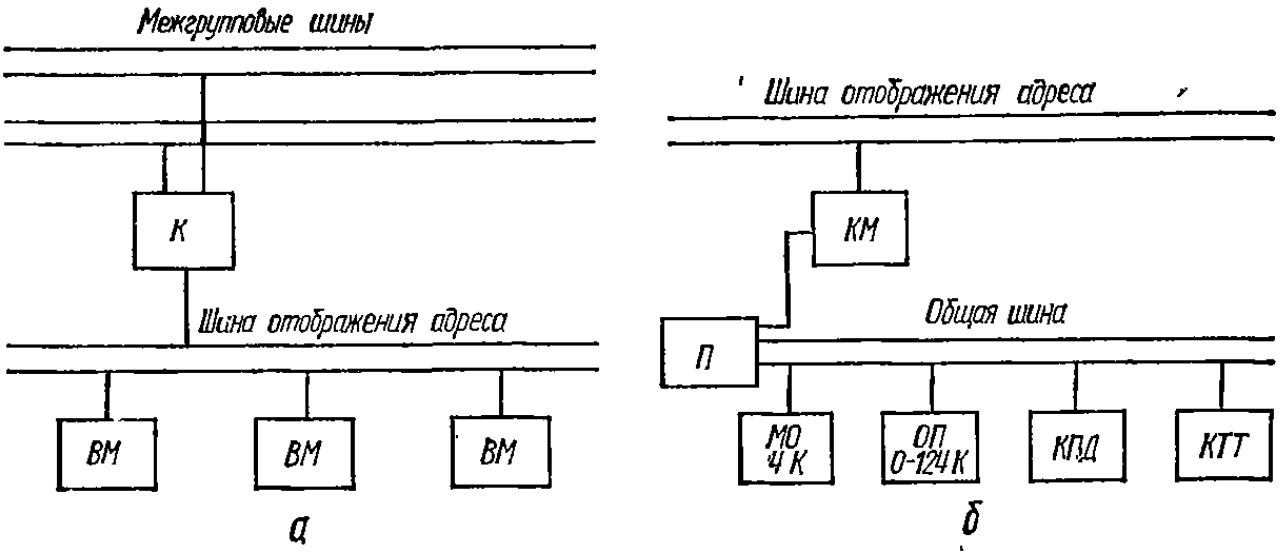


Рис. 8. Структура системы С.m* с простыми межмодульными связями (*а*) и с местиыми коммутаторами (*б*)

В системе предусмотрено выполнение программ самодиагностирования любым *ВМ*, связанным последовательным каналом с машиной-диспетчером. Получив информацию о занятости *ВМ*, машина-диспетчер загружает в свободные модули диагностические программы и выдает команду на их выполнение. Диагностические программы применяются, в основном, для поиска постоянных отказов и лишь частично — перемежающихся. Программы загружаются последовательно, контролируя отдельные части модуля. Условием окончания диагностирования является обнаружение заданного количества ошибок или успешное завершение многократных прогонов программ. Диагностическими программами выявляют, в частности, зацикливание работы модуля, превышение данного времени ожидания символа от диспетчера, а также ряд неисправностей контроллера *К*, основной памяти *ОП* и местного коммутатора *КМ*, процессора *П* (рис. 8,*б*).

Данные эксплуатации системы в течение одного года показали, что диагностирование центрального процессора занимает лишь 10 % общего времени проверки системы. Среднее вре-

мя наработки на отказ за год увеличилось со 128 до 563 ч (из расчета на один ВМ) при периоде до 30 мин между загрузкой диагностических программ. Таким образом, применение микропроцессоров и более совершенная организация взаимодействия модулей в системе С. m привели к значительному повышению отказоустойчивости по сравнению с системой С. mmp.

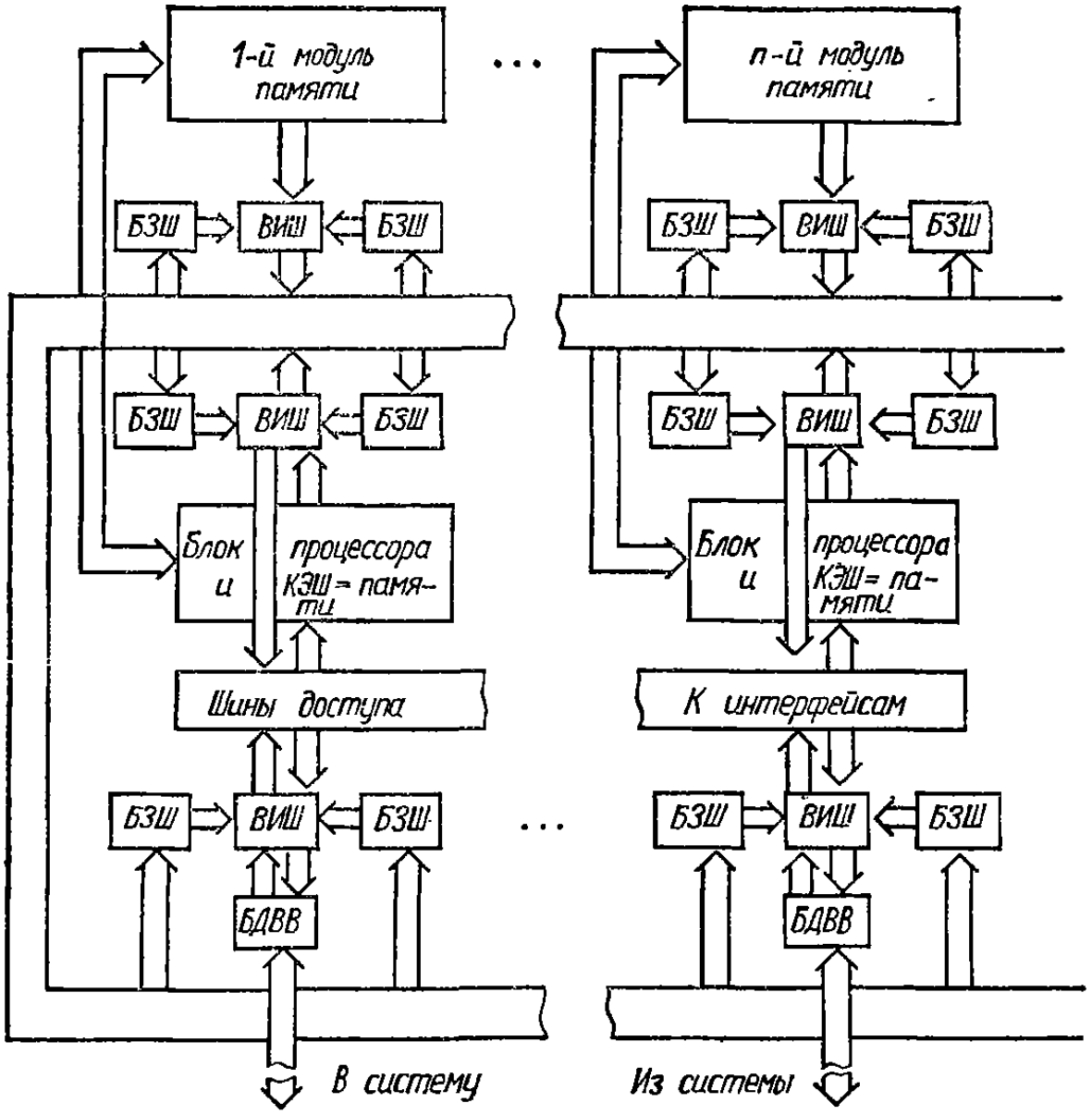


Рис. 9. Структура системы FTMR

Несколько иной подход использован для обеспечения отказоустойчивости в системах FTMR и SIFT, имеющих интенсивность отказов 10^9—10^{10} отказов /ч [6,58].

В обеих системах используется тройное модульное смешанное резервирование процессоров и блоков памяти, которое обеспечивает высокую достоверность вычислений. Смешанное резервирование заключается в перестройке триад элементов (процессоров, блоков памяти) в новые триады, в которых неисправности маскируются (использован принцип голосования 2 из 3, 3 из 3).

Отказоустойчивость системы FTMR обеспечивается в три этапа:

обнаружение, локализация и удаление неисправного элемента из триады;

замена (восстановление) его исправным;

инициализация перестроенной триады.

Система FTMR (рис. 9) состоит из модулей (блоков) процессоров с блоками КЭШ-памяти (индивидуальной промежуточной памяти), соединенных с резервированными шинами доступа к памяти и к интерфейсу через блоки защиты шины *БЗШ*, вентили изоляции шины *ВИШ*. В систему входят модули памяти, соединенные с шиной через *БЗШ*, *ВИШ* и блоки доступа к вводу-выводу *БДВВ*.

В число функций управления блоков *БЗШ* входит выбор режима питания, триад шин модулей памяти и тракта передачи некоторых схем проверки. Адресация блоков *БЗШ* производится к общей памяти, причем в каждый момент выполняется лишь одна предшествующая команда. При возникновении отказа блоки *БЗШ* стремятся перевести систему в состояние, неопасное для продолжения работы. Отказы в самих блоках *БЗШ* приводят к отключению всего модуля или к перестройке триад шин (например, использование резервных).

В процессе работы системы участвуют триады модулей процессоров и памяти и триады шин. Выбор триад модулей определяется только готовностью выбираемых модулей к работе. При работе системы независимо полученные данные передаются по трем независимым между собой шинам, что обеспечивает повышение достоверности вычислений. В системе могут одновременно функционировать несколько триад модулей памяти и процессоров, однако совместно лишь одна триада памяти и одна триада шин интерфейса. Поиск неисправностей в элементах триад производится сравнением выходов элементов триады с использованием блоков *БЗШ* и *ВИШ* и зондированием. В последнем случае один из элементов триады выдает ошибки, а два других — проверяют, что позволяет устранить накопление неисправностей и повысить глубину поиска дефекта.

Исключение неисправного процессора из триады процессоров происходит через его *БЗШ* после завершения шага задания, что занимает время меньше одной секунды. После исключения неисправного процессора резервный модуль процессора, если он имеется, через свой *БЗШ* связывается с соответствующей шиной по команде процессорной триады, назначенной для выполнения реконфигурации. При отсутствии резервных модулей процессоров триада с неисправными модулями отключается, причем исправные процессоры можно использовать как резервные.

Восстановление триад модулей памяти происходит аналогичным образом за исключением лишь того, что при переключениях необходимо учитывать адресное пространство этих модулей и заложенные функции (особенно для ПЗУ). При отказе ПЗУ допускается использование ЗУПВ, в которые производится предварительная загрузка содержимого исправных модулей. Инициализация с перестроенной триады производится под управлением триад, выполняющих системные программы с привлечением *БЗШ*. В системе дополнительно предусмотрено ограничение области распространения неисправностей, защиты цепей синхронизации модулей и наличие аккумуляторов питания для защиты содержимого ЗУ при отключении питания. Результаты проверки системы FTMR показали значительное влияние длительности перемежающихся отказов на вероятность безотказной работы системы, недостаточную эффективность *БЗШ* и *ВИШ*.

В системе SIFT программный контроль осуществляется с помощью общих и локальных исполнительных программ [8].

Локальные исполнительные программы входят в состав каждого модуля обработки данных и включают подпрограммы голосования, обработки ошибок, сопряжения буферов, а также подпрограммы-планировщики. Локальные программы обеспечивают данные для общей исполнительной программы, которая по этапам осуществляет реконфигурацию системы.

Подпрограмма голосования производит обработку голосованием не менее чем троекратно результатов вычислений, полученных от других процессоров. При наличии ошибок фиксируется ошибка данных от соответствующих процессоров. При наличии ошибок подпрограмма обработки ошибок формирует таблицу ошибок процессор/шина для каждого процессора, в которую заносится количество ошибок при обращении данного процессора к другим процессорам через различные шины. Таблица ошибок передается программе составления таблицы ошибок и общей исполнительной программе, которая обеспечивает реконфигурацию системы через программу локальной реконфигурации.

Микро-ЭВМ позволили создавать многомашинные мультимикропроцессорные системы типов, близких С2, С4, С6. Примером такой частично сетевой кольцевой системы служит базовая отказоустойчивая система BFS, описанная в работе [54]. Система содержит несколько микро-ЭВМ, каждая из которых связана с соседними микро-ЭВМ, кроме одной.

Принцип построения системы показан на рис. 10. Коммутаторы S обеспечивают связь вычислительной части с периферийными устройствами: дисплеями *DS* и устройством печати *ПТ*. Два элемента *М1* и *М2* имеют связь с накопителями на диске

ДС1 и *ДС2*. Каждый из элементов M_i системы содержит центральный процессорный элемент на микропроцессоре 8086; локальное ОЗУ объемом 32 Кбайт и ПЗУ объемом 16 Кбайт со средствами обнаружения и корректировки ошибок; двунаправленный порт ОЗУ на 16 К и процессор связи, обеспечивающий связь с 4 портами ввода-вывода и последовательную передачу данных. В системе отсутствует общий супервизор, поэтому каждый из элементов может автономно использовать свои ресурсы и интерфейс. Кроме того, для повышения эффективности обмена центральный процессорный элемент и процессор связи разделены. Это обеспечивает связь от элемента к элементу и возможность создания локальных сетей из микро-ЭВМ.

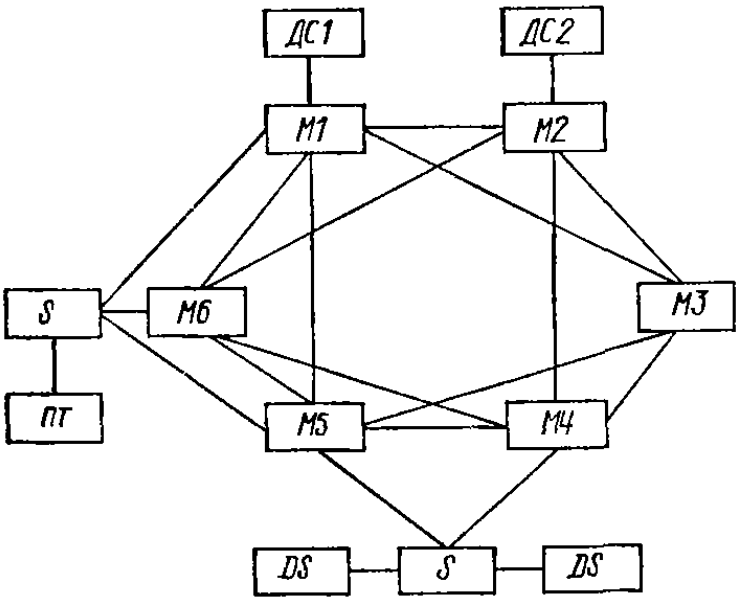


Рис. 10. Структура системы BFS

Система связи представляет собой 12 независимых дуплексных каналов. При этом обмен данными между процессорами связи и процессорным элементом зависит от прерываний по состоянию буфера канала. Структурно программное обеспечение системы BFS подразделяется на функционально независимые части, которые выполняются в соответствии с иерархической подчиненностью и резервируются. В частности, они обеспечивают постоянно работоспособное ядро в каждом модуле M_i, имеют стандартный формат обмена, позволяют производить обработку избыточных таблиц и реализацию избыточных функций. Выполняемые программным обеспечением функции зависят от прерываний и стратегии реконфигурации с плавной деградацией.

Общее управление системы обеспечивает самопроверку и передачу элементов своего состояния соседним элементам с целью реконфигурации и автоматического восстановления системы. Таким образом, в системе предполагается эффективное средство отказоустойчивости.

В ряде современных бортовых систем часто используют распределенные сетевые или иерархические структуры систем, в которых ЭМ обладают повышенной надежностью.

Реализация таких машин на сверхбольших интегральных схемах в виде самопроверяемых вычислительных модулей

(СВМ) системы STAR описана в работе [27]. Структура СВМ показана на рис. 11. Обмен каждого СВМ с другими модулями идет через адаптеры шины *АШ* и контроллер шины *КШ* программируемого стандартного блока сопряжения с шиной *СШ—СБ*. Самопроверяемый вычислительный модуль имеет внутреннюю избыточную память, соединенную с внутренней шиной

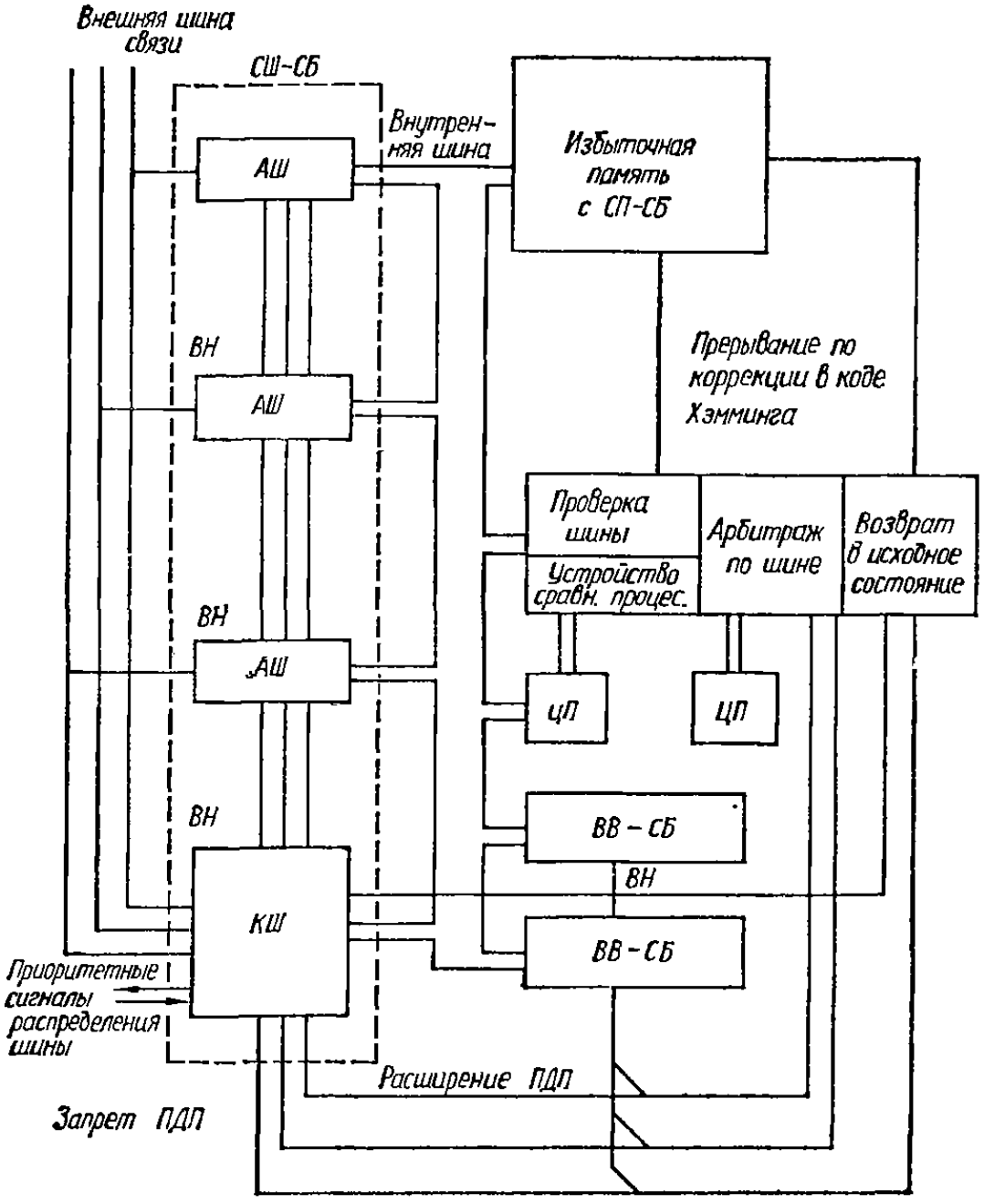


Рис. 11. Упрощенная структура схемы СВМ

через стандартный блок сопряжения с памятью *СП—СБ*, обнаруживающий и корректирующий ошибки, а также центральный стандартный блок (ЦСБ), включающий два центральных процессора *ЦП*, стандартные блоки ввода-вывода *ВВ—СБ*, соединенные между собой внутренней шиной.

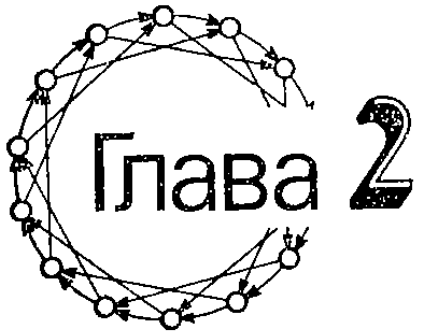
Блоки *СП—СБ* и *СШ—СБ* обеспечивают контроль входящих в них схем путем сравнения результатов выполнения одинаковых функций на двух процессорах *ЦП*.

В ЦСБ принимается решение о дальнейшем функционировании СВМ, повторении выполнения программ и запуске процессора. Арбитражный элемент, входящий в ЦСБ, на основе сигналов запроса от контроллеров *ПДП* других СВМ обеспечивает освобождение шины процессорам и разрешает доступ СВМ к внутренней шине. ЦСБ управляет также работой других СВМ по внутренней шине и осуществляет сбор сигналов о неисправностях СВМ. В системе по запросу из внешней шины ЗУ выдает запрашивающему СВМ необходимую информацию. В системе предусмотрена возможность управления данным СВМ по внешней шине внешними СВМ. Так, например, внешний СВМ может получать информацию о неисправных состояниях адресуемого СВМ, управлять внутренней реконфигурацией, загружать и считывать данные и управлять системой ввода-вывода адресуемого СВМ.

По зарубежным источникам [1, 27, 35, 40] спецификой вычислительных систем космических летательных аппаратов является рассредоточенность обслуживаемых ими подсистем космического корабля. В то же время малые габариты, масса, энергопотребление СВМ на базе микро-ЭВМ и микропроцессоров, обладающих большими функциональными возможностями, позволяют располагать СВМ в непосредственной близости от подсистем корабля, объединяя их для решения общих задач. Поэтому в таких вычислительных системах используют иерархический принцип построения (системы типа С5), при котором СВМ низшего уровня связаны непосредственно с получением и обработкой первичной информации, а СВМ высшего уровня осуществляют координацию работы СВМ низшего уровня и обеспечивают получение и обработку информации в целом по системе.

Связь между модулями высшего и низшего уровней происходит по внешним шинам связи, однако круг выполняемых модулями функций может быть принудительно ограничен (например, введен запрет на считывание данных из СВМ верхнего уровня). С учетом этого в объединенной системе обработки данных различают СВМ высокого уровня (МВУ) и низкого уровня — терминальные модули (ТМ). Управление шинами связи при этом возлагается на МВУ, а в случае их отказа передается другим МВУ. В случае отказа шины два контроллера шин МВУ могут работать на одну (исправную) шину связи. Обмен данными между МВУ и ТМ происходит по сигналам прерывания длительностью 2,5 мс от МВУ, что обеспечивает синхронизацию МВУ и ТМ. После обнаружения неисправности средствами самоконтроля СВМ заменяется на исправный. После завершения вычислений один из исправных модулей может диагностировать этот неисправный модуль.

В системе STAR предусмотрен иерархический принцип обнаружения и устранения влияния неисправностей, восстановления и реконфигурации системы. В частности, модули МВУ осуществляют проверку и изоляцию шин связи и модулей нижнего уровня, управляют перегрузкой памяти и реконфигурацией при замене неисправных модулей, производят инициализацию при перемежающихся отказах. Таким образом создаются условия функционирования этой системы с планируемым сроком автономной работы в течение пяти лет.



Глава 2

МОДЕЛИ ДИАГНОСТИРОВАНИЯ НЕИСПРАВНОСТЕЙ

1. МОДЕЛИ ВЫЯВЛЕНИЯ ОТКАЗОВ

Дальнейшее повышение производительности и расширение функциональных возможностей многомашинных отказоустойчивых систем связано с увеличением количества ЭМ в системе, следствием чего является возможность отказа большого количества ЭМ в процессе работы системы.

Использование внешних средств контроля и диагностирования в таких системах является, как правило,неоправданным, поскольку вмешательство таких средств в работу системы приводит к ее усложнению и снижению производительности. При этом дополнительно возникает проблема обеспечения надежной работы этих средств. Поэтому при определении работоспособности системы и диагностировании неисправностей в ней целесообразно использовать взаимодействия ЭМ между собой, возникающие в процессе их функционирования. В процессе проверки системы в качестве элементов системы с определенными диагностическими характеристиками взаимодействия используют ЭМ.

Существует большое разнообразие моделей взаимодействия, описывающих диагностирование и контроль в многомашинных системах. Наиболее распространенными являются модели, которые условно назовем P-, B-, R- и K-моделями.

P-модель [51] основана на представлении процесса контроля и диагностирования таким взаимодействием пар элементов системы (пара состоит из контролирующего и контролируемого элемента), следствием которого является получение множества результатов контроля. В каждой паре при исправности конт-

ролирующего элемента результат контроля правильно отражает состояние работоспособности контролируемого им элемента, т. е. работоспособный элемент признается работоспособным, а неработоспособный — неработоспособным. При отказе контролирующего элемента результат контроля может быть произвольным независимо от состояния контролируемого им элемента, т. е. контролируемый элемент признается работоспособным или неработоспособным независимо от наличия отказа в нем.

Взаимодействия в рассматриваемой системе S можно представить графом $G = G(V, E)$. Множество V есть множество элементов системы v_i, $i = \overline{1, n}$, $v_i \in V$, а E — множество связей (v_i, v_j), $v_i v_j \in V$, образуемых при контроле элементом v_i элемента v_j. Тогда для P-моделей при наличии связи (v_i, v_j) получаемый результат контроля описывается следующим образом:

$$a_{ij} = \begin{cases} 0, & \text{если элемент } v_i \text{ признает элемент } v_j \\ & \text{работоспособным;} \\ 1, & \text{если элемент } v_i \text{ признает элемент } v_j \\ & \text{неработоспособным.} \end{cases}$$

При работоспособности элемента v_i ($v_i = 0$) результат a_{ij} совпадает с состоянием элемента v_j, а при отказе в элементе v_i ($v_i = 1$) значение a_{ij} будет произвольным — 0 или 1.

Множество значений $\{a_{ij}\}$ системы называется синдромом.

Приведенная в работе [37] B-модель отличается от P-модели лишь тем, что недействительным считается получение результата контроля $a_{ij} = 0$ при наличии отказов одновременно в элементах v_i и v_j. Применение B-модели приводит к меньшему количеству допустимых состояний работоспособности системы.

Введенная в работе [52] R-модель предусматривает описание проверки одного элемента несколькими другими аппаратно или с помощью программных процедур. В системах, описываемых такими моделями, n элементам ставится в соответствие множество F_0 допустимых одиночных неисправностей f_i: $F_0 = \{f_1, \ldots, f_n\}$, для которых задается множество J тестов h_j, $j = 1, \ldots, p$, $J = \{h_1, \ldots, h_p\}$. При этом предполагается, что каждый тест является полным тестом для одной неисправности из множества F_0.

Тест h_j считается полным для неисправности f_i, если он не проходит при наличии неисправности f_i и проходит при отсутствии неисправностей из множества F_0. Например, для комбинационного элемента тест не проходит, если существует по крайней мере один такой входной код, при котором код на выходе элемента отличен от кода работоспособного элемента.

В общем случае в таких системах допускается наличие множества неисправностей F при отказе более одного элемента системы:

$$F = \{F_1, \ldots, F_k, \ldots, F_{2^n}\}.$$

Каждая неисправность F_k (ее часто называют k-м набором неисправных элементов), $k = 1, \ldots, 2^n$ представляет собой подмножество множества F_0 одиночных неисправностей.

Для неисправностей множества F условие полноты теста может нарушаться. Поэтому для оценки полноты тестов для каждой неисправности F_k вводится множество $T(F_k)$ тестов, искажаемых при наличии F_k. Тест h_j искажается при наличии неисправности F_k, если результат выполнения теста является непредсказуемым или недостоверным в том смысле, что тест h_j может успешно пройти при $h_j \in h(F_k)$ или может не пройти при $h_j \notin h(F_k)$, где $h(F_k)$ — множество полных тестов одиночных неисправностей, образующих неисправность F_k.

Искажение тестов одиночных неисправностей позволяет учитывать влияние отказов элементов, участвующих в проверке данного элемента. Так, например, при контроле запоминающего устройства используются шина связи, приемные регистры адреса и данных, которые могут, при наличии в них отказов, исказить прохождение теста для запоминающего устройства.

К-модель [48] выделяет следующие виды неисправностей:

$A = \{a_1, \ldots, a_{|A|}\}$ — множество атомарных неисправностей;

$M = \{m_1, \ldots, m_{|M|}\}$ — множество макронеисправностей;

$P = \{p_1, \ldots, p_{|P|}\}$ — множество групповых дефектов, где $|X|$ обозначает мощность множества X.

Множество атомарных неисправностей A описывает отказы на наиболее низком логическом уровне системы. Они могут быть описаны перечнем функциональных элементов или постоянными неисправностями. Групповые дефекты множества M представляют собой физически или логически заменяемые части системы, которые определяют глубину поиска дефекта в системе. Макронеисправности являются следствием возникновения подмножества атомарных неисправностей из множества неисправностей A.

Между множествами неисправностей существуют определенные отношения. В частности, введена функция f_{ap} из A в P и отображение связи r_{am} между A и M, а также r_{mp} между M и P.

Функция f_{ap} из A в P, обозначаемая $f_{ap} : A \to P$, назначает каждой атомарной неисправности один и только один

групповой дефект. Считается, что групповой дефект возник, если имеет место атомарная неисправность, т. е.

$$\forall a_i \in A, \quad \exists p_j \in P : f_{ap}(a_i) = p_j.$$

Отображение r_{am} (записывается $a_i r_{am} m_j$) описывает такую связь между элементами множеств A и M, при которой макронеисправность m_j возникает всякий раз, как только появляется атомарная неисправность a_i. Во многих случаях отображение r_{am} удается представить функцией $f_{am} : A \to M$.

Связь между элементами множеств M и P состоит в том, что если m_j — дефект из группового дефекта, то по крайней мере один из дефектов является следствием макронеисправности.

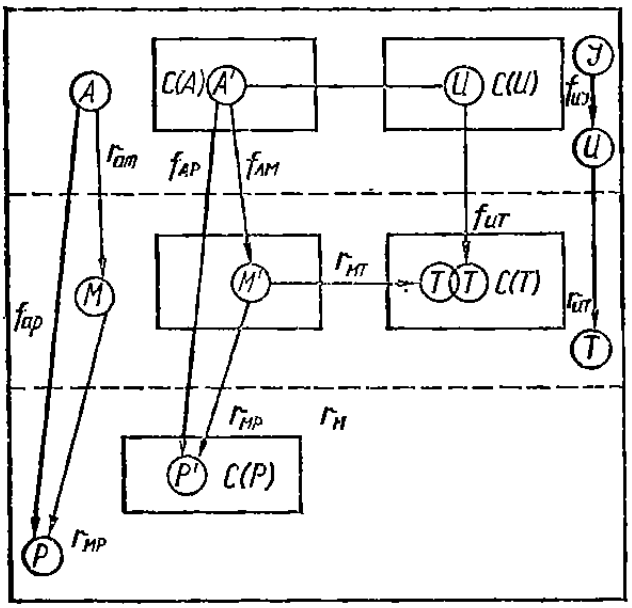


Рис. 12. Взаимосвязи в К-модели

Элементы множеств A, M, P являются одиночными неисправностями системы. Для задания наборов неисправностей для каждого из этих множеств определяют множества A', M', P' подмножеств всех элементов соответственно из A, M, P, т. е.

$$A' \subseteq C(A); \; M' \subseteq C(M); \; P' \subseteq C(P).$$

Для множеств A', M', P' определяют функции f_{AP} и f_{AM} следующим образом:

$$f_{AP} : A' \to C(P), \quad \forall A^i \in A', \quad f_{AP}(A^i) = P^i = \{p_k \mid p_k = f_{ap}(a_i), \; a_i \in A^i\};$$

$$f_{AM} : A' \to C(M), \quad \forall A^i \in A' \quad f_{AM}(A^i) = M^i = \{m_k \mid a_i r_{am} m_k, \; a_i \in A^i\}.$$

Обобщенная структура связей между множествами показана на рис. 12.

Диагностирование в системе производится с помощью атомарных тестов u_i множества $U = \{u_1, \ldots, u_i, \ldots, u_{|U|}\}$ и макротестов h_j множества $T = \{h_1, \ldots, h_j, \ldots, h_{|T|}\}$, а также входных последовательностей множества $J = \{J_1, \ldots, J_{|J|}\}$. Атомарный тест состоит в наблюдении одного выхода в момент времени при подаче на вход схемы входной последовательности. Макротест представляет собой множество атомарных тестов.

Сравнивая модели между собой, можно отметить, что рассмотренные P- и B-модели ориентированы, в основном, на диагностирование с использованием результатов взаимодействия элементов между собой. В R- и K-моделях при поиске дефектов в большей степени учитывается функциональная структура объекта диагностирования, диагностические признаки и возможности программных средств.

В многомашинных отказоустойчивых системах существует два способа диагностирования: поиск по крайней мере одного неработоспособного элемента с последующей заменой его или исключением из системы и продолжение поиска; поиск всех неработоспособных элементов системы.

Первый способ принято называть последовательным (многошаговым) диагностированием с восстановлением, а второй способ — одношаговым без восстановления.

Введем ряд определений, которые будут использованы при изучении процессов диагностирования в отказоустойчивых системах. Обозначим через t наибольшее количество неисправных элементов в диагностируемой системе S, а через t_j — количество неисправных элементов некоторого подмножества неисправных элементов F_j, $F_j \subseteq V$.

Определение 1. Система S называется t-диагностируемой (t- ДС) без восстановления, если для любого подмножества F_j, $t_j \leqslant t$ по заданному множеству результатов $\{a_{ij}\}$ все неисправные элементы подмножества F_j могут быть однозначно и одновременно определены.

Следует отметить, что при диагностировании можно использовать несколько синдромов, соответствующих данной неисправности F_j.

Определение 2. Система S называется t-диагностируемой с восстановлением, если по результатам контроля $\{a_{ij}\}$ может быть определен по крайней мере один неисправный элемент v_k, $v_k \in V$, $k = 1, \ldots, n$.

Определение 3. Диагностическим графом (ДГ) $G(V, E)$ называется множество элементов V, $V = \{v_1, \ldots, v_n\}$ с определенным на нем множеством E направленных связей (v_i, v_j), $v_i, v_j \in V$.

Определение 4. Допустимым набором состояний элементов (ДНС) называется множество состояний элементов системы S, не противоречащих заданному набору синдромов $\{a_{ij}\}$.

Определение 5. Система, описанная ДГ $G(V, E)$ и моделью взаимодействия элементов в процессе контроля называется структурой взаимоконтроля (СВК).

Выбор модели взаимодействия элементов в процессе проверки системы предопределяет в значительной степени сложность реализации и основные характеристики алгоритмов диагностирования (их быстродействие, занимаемый объем памяти, возможности структуризации и т. п.). Поэтому при разработке отказоустойчивых систем естественным является стремление к использованию наиболее простых моделей, в которых функциональные особенности элементов системы не играют существенной роли при анализе технического состояния системы в целом.

Такой подход позволяет создавать достаточно универсальные алгоритмы и программы диагностирования и контроля, применимые для широкого класса реальных систем. Это послужило одной из основных причин широкого распространения Р-моделей взаимодействия, которые достаточно просто и полно описывают диагностирование в многомашинных распределенных системах, построенных на основе микро-ЭВМ и микропроцессоров.

Действительно, при достаточно надежных связях между элементами системы каждый из них может устанавливать связи с другими элементами и подавать на них тесты. Получаемый при проверке результат выполнения теста анализируется контролирующим элементом и принимается решение об исправности. Примером одного из простейших способов такого тестирования является проверка отклика контролируемого элемента по запросу контролирующего элемента с использованием таймера, который входит во многие современные ЭВМ.

Поэтому основное внимание при рассмотрении диагностирования без восстановления уделяется СВК, в которых взаимодействия элементов описываются Р-моделями. Будем называть такие структуры Р-структурами.

Обозначим через $\Gamma^{-1}(v_i) = \{v_j \mid \exists (v_j, v_i) \in V, v_j \neq v_i\}$, т. е. множество вершин v_j, соответствующих элементам, контролирующим элемент v_i.

Теорема 1 [44]. Пусть диагностируемая система S описывается Р-структурой для множества V элементов ($|V| = n$) таких, что для любого $(v_i, v_j) \in E$, $v_i, v_j \in V$, $\exists (v_j, v_i) \in E$. Система S является t-ДС без восстановления тогда и только тогда, когда $\forall v_i \in V$, $|\Gamma^{-1}(v_i)| \geqslant t$.

Данная теорема позволяет определять t-ДС без восстановления по структурным свойствам графа $G = G(V, E)$.

Используемое в теореме 1 предположение об отсутствии взаимонаправленных связей, т. е. одновременном наличии

в Р-структуре связей $(v_i, v_i) \in E$, $(v_j, v_i) \in E$, $v_i, v_j \in V$, ограничивает класс рассматриваемых реальных систем.

Во многих существующих микропроцессорных системах, построенных на основе микро-ЭВМ, обычно в процессе функционирования предусматривается возможность выбора в качестве ведущих произвольных микро-ЭВМ. Выбранные микро-ЭВМ могут применяться в качестве контролирующих элементов, причем выбор ведомых микро-ЭВМ, выполняющих функции контролируемых элементов, производится ведущей с использованием схем арбитража и данных о необходимой структурной организации t-ДС.

Следовательно, для многих реальных микропроцессорных систем естественным является предположение о наличии двунаправленных связей, т. е. связей

$$(v_i, v_j) \in E, \quad (v_j, v_i) \in E, \quad v_i, v_j \in V, \quad v_i \neq v_j.$$

Основные требования к структурной организации таких систем, обеспечивающей их t-диагностируемость, вытекают из следующей теоремы [44].

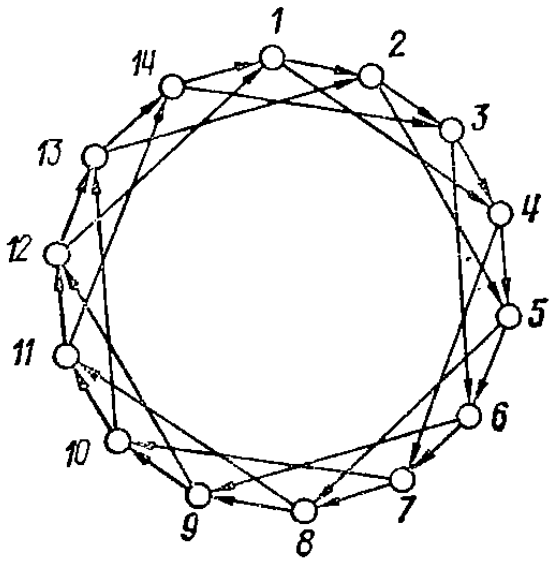


Рис. 13. Диагностируемая система

Теорема 2. Пусть система S описывается Р-структурой. Система S является t-диагностируемой тогда и только тогда, когда выполняются следующие условия:

1) $n \geqslant 2t + 1$;

2) $\forall v_i \in V$, $\Gamma^{-1}(v_i) \geqslant t$;

3) для каждого $0 \leqslant p < t$ и каждого $X \subset V$ $|X| = n - 2t + p$, $|\Gamma X| > p$, где $\Gamma X = \{v_j \mid \exists (v_i, v_j) \in E, v_i \in X, v_j \in V \setminus X\}$.

Теорема 2 позволяет оценить при заданном количестве элементов n наибольшее количество неисправных элементов t, при котором возможно однозначное определение неисправных элементов в системе. Действительно, непосредственно из условия 1 теоремы 2 следует, что $t \leqslant \lfloor (n-1)/2 \rfloor$, где через $\lfloor x \rfloor$ обозначено наибольшее натуральное число, не большее x.

Так, например, система, представленная ДГ на рис. 13, является 1-диагностируемой и 2-диагностируемой без восстановления.

Анализ Р-структур на основе теорем 1 и 2 позволяет оценить t-ДС для заданных значений t и разбиений системы на инцидентные подмножества элементов. Однако в ряде случаев, используя данные об архитектурных особенностях построения

системы, удается определить подмножества направленных путей в ДГ системы или оценить связность элементов системы, обеспечиваемую через подмножества других элементов. Наличие таких данных позволяет оценить потенциальные возможности диагностирования в рассматриваемых системах.

Предположим, что ДГ G является сильно связным графом, т. е. для любой пары элементов $v_i, v_j \in V$ существует направленный путь связей из одной вершины к другой. Связностью $k(G)$ направленного графа G называется минимальное число вершин ДГ, исключение любой из которых приводит к несвязности графа G. Тогда для Р-структур справедливы следующие два следствия к теореме 2 [44, 51].

Следствие 1. Пусть $G = G(V, E)$ есть ДГ Р-структуры системы S, состоящей из n элементов, $n = |V|$. Тогда, если $k(G) \geqslant t$ и $n \geqslant 2t + 1$, то S является t-ДС.

Следствие 2. Система S, представленная ДГ $G = G(V, E)$, $n = |V|$, является t-диагностируемой, если связь между вершинами с номерами i и j удовлетворяет соотношению $j - i = \delta m \pmod n$, $m = 1, 2, \ldots, t$ и n и δ представляют собой взаимно простые числа.

Результаты следствия 2 удобно использовать для построения симметричных t-диагностируемых Р-структур.

Оценку диагностических возможностей ДГ можно производить на основе отображения множеств элементов на область натуральных чисел.

Обозначим P множество всех разбиений множества элементов V системы S на три непересекающихся подмножества (X, Y, Z), удовлетворяющих следующим условиям:

1) $|Z| \geqslant 1$;
2) $\Gamma X \subseteq Y$; $\Gamma X = \{v \mid v \in V, \exists X \in X : (x, v) \in E\} - X$;
3) $\Gamma^{-1} Z = \{v \mid v \in V, \exists z \in Z : (v, z) \in E\} - Z$;
4) $V = X \cup Y \cup Z$.

Пусть K есть функция из множества P разбиений p на множество натуральных чисел такое, что $\forall p \in P$:

$$K(p) = |Y| + [|Z|/2].$$

Тогда условие t-диагностируемости без восстановления имеет следующий вид.

Лемма 1 [36]. Система S является t-диагностируемой тогда и только тогда, когда $\forall p \in P : K(p) > t$.

В тех случаях, когда известно минимальное значение $K(p)$, представляется возможным непосредственное определение t-ДС. Это значение может быть задано в процессе проектирования отказоустойчивой системы или получено в процессе перестройки структуры системы.

Теорема 3. Система, представленная ДГ $G(V, E)$, t-диагностируема тогда и только тогда, когда $t \leqslant \tau(G)$, где $\tau(G) = K_{\min}(G) - 1$ — диагностическое число; $K_{\min}(G) = \min\limits_{p \in P} \{K(p)\}$.

Теорема 4. Для системы S, представленной ДГ $G(V, E)$, имеет место следующее соотношение: $\tau(G) < [(n-1)/2]$, где $[x]$ наибольшее натуральное число, не большее x.

Таким образом, по известному диагностическому числу $\tau(G)$ системы можно просто оценить фактические и потенциальные диагностические возможности отказоустойчивой системы.

Непосредственное определение значений $\tau(G)$ для структур $G = G(V, E)$ с большим количеством элементов связано с громоздким перебором непересекающихся подмножеств X, Y, Z. Ввиду этого в процессе перестройки структуры отказоустойчивой системы во время эксплуатации определение $\tau(G)$ может оказаться неоправданным ввиду отсутствия необходимых вычислительных мощностей и высокой инерционности при реконфигурации системы.

Поэтому в практике проектирования и обслуживания таких систем диагностическое число $\tau(G)$ оценивают, не прибегая к сложным вычислениям. [36, 44, 51].

Теорема 5. Для Р-структур справедливо соотношение $\tau(G) \leqslant d_{\min}$, где $d_{\min}$ — минимальная полустепень захода вершин графа.

Теорема 6. Для СВК, представленных Р-структурой, описываемой ДГ $G = G(V, E)$, в которой существуют взаимонаправленные связи взаимоконтроля между элементами, имеет место соотношение $\tau(G) = d_{\min}$.

3. ДИАГНОСТИРОВАНИЕ С ВОССТАНОВЛЕНИЕМ

С увеличением количества неисправных элементов в системе диагностирование без восстановления становится сложной задачей. Это связано с тем, что обычно в отказоустойчивых системах стремятся к использованию наименьшего количества связей, достаточного для обнаружения и локализации наиболее характерных неисправностей. Однако при эксплуатации системы могут возникать отказы произвольных элементов, количество которых превышает максимально допустимое для характерных неисправностей. При малом количестве связей это приводит к неоднозначности диагностирования работоспособности элементов системы, в результате чего возникает необходимость замены большого количества элементов, подозреваемых на отказ, или усложняются процедуры контроля работоспособности выявленных элементов. В первом случае увеличивается количество необходимой аппаратуры, во вто-

ром — усложняется обслуживание системы. Более целесообразно многоэтапно выявлять небольшое количество неработоспособных элементов с последующей их заменой. Это позволяет сократить количество резервируемых элементов и упростить алгоритмы обслуживания системы.

Поэтому при отказах произвольного количества элементов системы наиболее рационально по аппаратурным затратам и простоте обслуживания использовать диагностирование с восстановлением, которое можно рассматривать как многошаговый процесс выполнения процедур идентификации отказов и восстановления. Необходимым условием такого процесса является обнаружение на каждом шаге, по крайней мере, одного из неисправных элементов.

Пусть F_i — подмножество элементов, идентифицированных на шаге i диагностирования как неисправные. Тогда необходимое условие диагностирования с восстановлением имеет вид

$$\forall_i,\ i>0, |F_i| \geqslant 1,\ F_i \subseteq V.$$

Если восстановление проводится для всех подозреваемых на отказ элементов, то справедливо условие

$$\bigcap_{j=1 \ldots t} F_i = \varnothing, \quad \bigcup_{j=1 \ldots (i-1)} F_j = V_в,\ V_в \subseteq V,$$

где $V_в$ — множество восстановленных элементов; i — номер шага диагностирования.

Для структур взаимоконтроля, в которых связи взаимодействия описываются R-моделями, допустимое количество неисправных элементов определяется следующим образом [52].

Лемма 2. Пусть $\{F_i, F_j, \ldots, F_m\} \subseteq F(t)$ — два или более наборов неисправных элементов, совместимых для заданного синдрома, таких, что $F_i \cap F_j \cap \cdots \cap F_m = \varnothing$. Тогда

$$|F_i \cup F_j \cup \cdots \cup F_m| \leqslant \lfloor((t+2)/2)^2\rfloor.$$

Данный результат сохраняется справедливым и для структур взаимоконтроля с P-моделями связи. Поскольку на каждом шаге диагностирования с восстановлением определяется по меньшей мере один неисправный элемент, а общее их число в системе равно n, то из решения неравенства для t, удовлетворяющего условию: $\lfloor(t+2)/2\rfloor^2$ — натуральное число, получаем

$$t > 2\sqrt{n} - 2.$$

Минимальное значение $t_{\min}$ при произвольных значениях t вытекает из следствия к лемме 2.

Следствие. $t_{\min} = \lceil 2\sqrt{n} \rceil - 3$.

Таким образом, получаем соотношение, позволяющее оценить минимальные диагностические возможности для структур с восстановлением. Например, при наличии произвольных неисправностей для системы, состоящей из 100 элементов, структуры взаимоконтроля с восстановлением позволяют выявить неисправности при отказе не менее 17 % элементов. Это значительно превышает процент выявляемых одиночных неисправностей одного элемента (1 %) и обеспечивает полноту контроля.

Теорема 7 [38]. Пусть система S описывается сильносвязным ДГ $G = G(V, E)$ с Р-моделью взаимодействия и пусть $M_1 = \max\limits_{v_j \in V} |\Gamma^{-1}(v_j)|$. Тогда система S является наименьшее t-диагностируемой с восстановлением, если

$$M_1 > 2t - 2 - \lceil \sqrt{n - 2t} \rceil - \left\lceil \frac{n - 2t}{\lceil \sqrt{n - 2t} \rceil} \right\rceil,$$

где $\lceil x \rceil$ — наименьшее натуральное число, не меньшее x.

Следовательно, верхняя граница t-диагностируемости с восстановлением, как и для диагностирования без восстановления, определяется наибольшим количеством элементов, контролирующих каждый из элементов структуры.

Более точную оценку удается получить при использовании данных о множестве элементов v_j, v_l, инцидентных v_i.

Обозначим через $\Delta\{v_j, v_l\}$ множество

$$\Delta\{v_j, v_l\} = \{v_t \mid v_t \neq v_j, v_l, \exists (v_t, v_j) \in E, (v_t, v_l) \in E\}.$$

Теорема 8 [38]. Пусть S система описывается сильносвязным диагностическим графом $G = G(V, E)$ с Р-моделью взаимодействия и пусть $M_2 = \max\limits_{v_j \neq v_l v_j, v_l \in V} |\Delta\{v_j, v_l\}|$. Тогда система S является по крайней мере t-диагностируемой с восстановлением, если t удовлетворяет соотношению

$$M_2 > 2t - \lceil (t + 2 - \sqrt{(t+2)^2 - 4n})/2 \rceil.$$

Приведенные оценки для t позволяют уточнить границы диагностируемости с восстановлением.

4. ОБЕСПЕЧЕНИЕ ОТКАЗОУСТОЙЧИВОСТИ ПРИ НЕОДНОЗНАЧНОМ ОПРЕДЕЛЕНИИ НЕИСПРАВНОСТЕЙ

Использование рассмотренных моделей при определении работоспособности в реальных системах сопряжено с некоторой недостоверностью диагностирования. Это связано с тем, что в реальной «физической» системе не выполняются вводимые предположения о характере неисправностей, видах вза-

имодействия элементов в процессе диагностирования, однозначности и полноте контроля одного элемента другим и т. д.

Поэтому возникает неоднозначность идентификации отказов элементов отказоустойчивой системы, что приводит к снижению эффективности диагностирования.

К основным причинам такой неоднозначности относятся:

1. Нарушение предположения о минимальности отказа в системе, при котором характерным считается отказ наименьшего количества элементов, что может быть обусловлено неоднородностью элементов системы (например, использованием в качестве элементов микро-ЭВМ различных типов, различием схем сопряжения элементов системы между собой и т. д.), неравноценностью влияния отказов элементов на работоспособность системы и процесс диагностирования (в частности, возникновение аварийных или тупиковых ситуаций, препятствующих дальнейшему правильному диагностированию) и др.

2. Отказ такого числа элементов системы, которое превышает допустимое наибольшее. Следствием может быть невозможность продолжения диагностирования по заданным алгоритмам или ухудшение числовых значений показателей контролепригодности (например, уменьшение коэффициента глубины поиска дефекта).

3. Искажение тестов в каналах связи между элементами. При использовании небольшого количества физических связей между элементами (шин связи, магистралей) допустимо предположение об их абсолютной надежности, а при большом количестве связей существенное влияние на показатели диагностирования и контроля (например, на априорные и апостериорные вероятности ошибок и вероятность правильного диагностирования) начинают оказывать неисправности физических элементов связи (шин связи, мультиплексеров, коммутаторов, шинных приемников и передатчиков, контроллеров и др.).

4. Неадекватность модели диагностирования элемента, что обусловлено ограниченной полнотой тестовых и других проверок (обычно необходимых для проверки лишь весьма ограниченного числа основных функций и параметров элемента). В значительной степени это обусловлено тем, что полная тестовая проверка, например микропроцессоров, даже в условиях массового выпуска (на стендах контроля и диагностирования) приводит к многократному (в десять и более раз) возрастанию стоимости микропроцессоров при одновременном увеличении времени контроля.

5. Возникновение критических ситуаций в процессе диагностирования, обусловленных «нестандартным» поведением контролируемой системы или нехарактерными типами неис-

правностей. К числу нехарактерных отказов относятся скрытые неисправности, взаимное влияние неисправностей отдельных элементов и др.

6. Наличие в системе перемежающихся отказов, которые приводят к различным синдромам и подмножествам исправных и неисправных элементов.

Неопределенность в определении технического состояния системы обусловливает ее недостоверное функционирование. Вследствие восстановления работоспособности системы по неправильным результатам возможно накопление отказов элементов, отказ системы или возникновение аварийных ситуаций.

Неопределенность идентификации отказов устраняется путем доопределения модели, вследствие чего усложняются алгоритмы диагностирования и контроля, увеличиваются требуемые объемы оперативной памяти, снижается быстродействие, возникают нежелательные последствия. В результате вместо улучшения качества диагностирования происходит его ухудшение. Поэтому в практике создания и эксплуатации отказоустойчивых систем необходимо стремиться к использованию достаточно простых способов доопределения структуры взаимоконтроля системы, обеспечивающих улучшение показателей диагностирования в сочетании с использованием естественных преимуществ мультимикропроцессорных систем.

Можно выделить следующие наиболее простые из этих способов.

1. Увеличение количества связей в отдельных частях структуры системы, обеспечивающее повышение числа обнаруживаемых отказов элементов и улучшение качества диагностирования.

2. Применение внешних средств контроля и диагностирования некоторых элементов системы.

3. Использование специального диагностического ядра, которое имеет высокую достоверность диагностирования и обеспечивает проверку (поэтапную или одновременную) других частей системы с требуемой полнотой.

4. Расширение функциональных и диагностических структур системы путем включения новых элементов с известными состояниями.

Диагностирование с использованием перечисленных или других способов должно сопровождаться анализом каждого конкретного случая неоднозначности идентификации. При анализе необходимо учитывать, что часть элементов в данный момент времени проверяется, часть — используется в качестве резервных, а определенная часть решает основные задачи.

Рассмотрим пример такого анализа с учетом функциональных особенностей системы .

Обозначим через $V_{\Phi}(\tau) \subseteq V$ подмножество элементов системы, функционирующих в момент времени τ, а через $V_{к}(\tau) \subseteq V$ — подмножество контролирующих и контролируемых элементов.

Если предполагается использовать только тестовое диагностирование элементов подмножества $V_{к}(\tau)$, то $V_{\Phi}(\tau) \cap V_{к}(\tau) = \varnothing$ и $V_{\Phi}(\tau) \cup V_{к}(\tau) \subseteq V$. Оставшаяся часть свободных элементов $V_{с}(\tau) = V \setminus (V_{\Phi}(\tau) \cup V_{к}(\tau))$ может использоваться для проверки элементов или для реализации функций системы.

Можно предположить, что перед началом диагностирования в $V_{\Phi}(\tau)$ входит множество исправных элементов, выполняющих заданные функции системы S, а подмножество $V_{с}(\tau)$ включает резервные исправные $V_{ср}(\tau)$ и неисправные $V_{сн}(\tau)$ элементы, выявленные в процессе предыдущих проверок.

Необходимо учитывать, что множество V также зависит от τ, в частности, за счет пополнения новыми элементами или исключения ряда элементов.

Отказоустойчивость реализуется после диагностирования структур взаимоконтроля, т. е. определения неисправных и исправных элементов. Однако определение всех исправных элементов является неоднозначным. Действительно, выделим в множесте V_{Φ} подмножество элементов $V' \subseteq V_{\Phi}$, для которых невозможно однозначно определить, какие из них являются исправными, и какие — неисправными. Разделим некоторое множество V' на два подмножества: V_1' и V_2', связанных между собой связью (v_i, v_j), $v_i \in V_1'$, $v_j \in V_2'$, имеющей значение «1» для модели взаимодействия P. Элементы подмножеств V_1' и V_2' могут иметь значения $v_i = 0$, $v_j = 1$ или $v_i = 1$, $v_j = 1$, т. е. все элементы подмножества V_2' неисправны. Неопределенность идентификации отказа в системе состоит в том, что существуют наборы состояний элементов, в которых $v_i = 0$ или $v_i = 1$.

Основной причиной неоднозначности определения исправности элементов подмножества V' является то, что при анализе системы для некоторых элементов при одном и том же синдроме в ДНС возможно присваивание этим элементам как работоспособного, так и неработоспособного состояния. Практически за счет повышения надежности этих элементов, использования методов резервирования, внешнего контроля работоспособности удается однозначно определить их исправность. Точно зная состояние элементов, можно доопределить состояния работоспособности оставшихся элементов за счет связей СВК.

Будем называть часть элементов, удовлетворяющих требованию однозначного определения состояния внешними сред-

ствами, внешнедиагностируемым ядром системы S (ВДЯ). Диагностическое ядро может быть задано как для определенных элементов системы, так и распределено в пределах заданного количества элементов с учетом их вероятностных показателей. В последнем случае распределенное ВДЯ может быть задано на основе выбора элементов, с наибольшим влиянием на вероятность безотказной работы системы.

Использование ВДЯ приводит к упрощению диагностирования элементов, связанных с ядром. Например, если X — ядро системы, состоящее из исправных элементов, то по заданному синдрому $\{a_{ij}\}$ можно однозначно определить подмножество исправных V'_p:

$$V'_p = \Gamma(X)_0 = \{v_j \mid (v_i,\ v_j) = 0,\ v_i \in X,\ v_j \in V \setminus X\}$$

и подмножество $V'_н$ неисправных элементов

$$V'_н = \Gamma^{-1}(X)_1 = \{v_i \mid (v_i,\ v_j) = 1,\ v_i \in V \setminus X,\ v_j \in X\}.$$

Использование ВДЯ снижает общую эффективность использования СВК при эксплуатации систем, поскольку требует вмешательства в работу системы и тем самым снижает ее производительность и независимость от внешних устройств контроля и диагностирования. Более целесообразным подходом является создание внутридиагностируемого ядра, элементы которого контролируются более тщательно, чем остальные. Это может быть сделано, например, путем построения для диагностического ядра специальных структур взаимоконтроля, удовлетворяющих более жестким требованиям диагностирования (например, с малым числом одновременно неисправных элементов, более полными возможностями по диагностированию неисправностей и т. д.), разделением структур взаимоконтроля с различным составом элементов, организацией централизованных сеансов диагностирования, анализа статистики проявления отказов элементов ВДЯ и т. д.

Таким образом, обеспечивается повышение качества контроля и диагностирования, что в конечном счете улучшает характеристики отказоустойчивости системы. Более полными возможностями при этом будут обладать структуры взаимоконтроля, полученные с учетом надежностных характеристик элементов, образующих систему.

5. ВЕРОЯТНОСТНЫЕ МОДЕЛИ

При диагностировании отказоустойчивых систем возникает потребность использовать модели, в которых учитывалась бы неравноценность влияния отказов элементов на работоспособность системы в целом.

К таким моделям относятся сложные модели, построенные по иерархическому принципу (например, К-модели) или предполагающие определенную подчиненность моделей высших уровней моделям низших. Однако учесть неравноценность отказов для таких моделей достаточно сложно.

Более простыми являются модели, в которых учитываются характеристики надежности отдельных элементов. В качестве наиболее распространенного показателя функционирования элементов системы и системы в целом является вероятность безотказной работы. При введении естественных допущений о независимом характере отказов, происходящих в элементах, удается построить достаточно простые модели диагностирования. Модели, использующие вероятностные характеристики надежности функционирования элементов при диагностировании, называют вероятностными моделями диагностирования.

Пусть $p(v_i)$ — вероятность отказа элемента v_i, $v_i \in V$. Тогда вероятность безотказной работы элемента $v_i \in V$ равна $1 - p(v_i)$. Для подмножества элементов $F_к \subseteq V$ априорная вероятность отказа системы S за счет неисправности всех элементов подмножества $F_к$

$$P(F_к) = \prod_{v_j \in V \setminus F_к} (1 - p(v_j)) \prod_{v_i \in F_к} p(v_i).$$

Обозначим P_{max} наибольшее допустимое значение вероятности отказа $P(F_к)$, тогда диагностирование системы проводится до выполнения условия

$$P(F_к) \leqslant P_{max}.$$

Вероятностные модели можно использовать при определенных ограничениях на характер взаимодействия между элементами. В частности, используя модели детерминированных взаимодействий элементов Р и В, получаем возможность проводить поиск и выявление неисправностей не с точки зрения обнаружения наибольшего количества неисправных элементов, а по заданным предельным значениям вероятности отказа системы в целом. Вероятностные модели, в которых поиск неисправностей ведется с использованием Р-модели, называют FP-моделями, а использующие В-модели — FB-моделями. Рассмотрим некоторые свойства диагностических структур, описываемых FP- и FB-моделями с восстановлением и без восстановления [41, 42].

Систему S, представленную ДГ $G(V, E)$, назовем вероятностно-диагностируемой (p-диагностируемой) без восстановления, если для любого взвешенного ДГ G_S и заданного синд-

рома $\{a_{ij}\}$ существует наибольшее одно совместимое множество неисправных элементов $F \subseteq V : P(F) > p$.

Множество неисправных элементов F является совместимым, если множество $V \setminus F$ исправных элементов не противоречит заданному синдрому $\{a_{ij}\}$.

Вводим обозначения $K(p) = -\log p + \sum\limits_{v_i \in V} \log(1 - p(v_i))$ и $W(v_i) = \log \frac{1 - p(v_i)}{p(v_i)}$; $W(F) = \sum\limits_{v_i \in F} W(v_i)$.

Тогда условие p-диагностируемости отказоустойчивой системы определяется следующей теоремой.

Теорема 9. Система S для моделей FP является p-диагностируемой без восстановления тогда и только тогда, когда не существует разбиения $\{V_1, V_2\}$ V:

$$W(V_1) < K(p) \text{ и } W(V_2) < K(p).$$

Назовем систему S, представленную ДГ $G(V, E)$, вероятностно-диагностируемой с восстановлением, если для любого взвешенного графа G и заданного синдрома $\{a_{ij}\}$ не существует совместимых множеств $F_1, F_2, \ldots, F_l$:

$$\bigcap_{i=1}^{l} F_i = \varnothing \text{ и } P(F_2) > p \text{ для } i = 1, 2, \ldots, l.$$

Таким образом, в p-ДС с восстановлением пересечение совместимых множеств неисправностей $F_1, F_2, \ldots, F_l$, имеющих $P(F_2) > p$, является не пустым.

Теорема 10. Система S для моделей FP является p-диагностируемой с восстановлением тогда и только тогда, когда $\forall F \subseteq V$, $\Gamma^{-1}(F) = \varnothing$ для каждого набора подмножеств $F_1, F_2, \ldots, F_l$ $\bigcup\limits_{i=1 \ldots l} F_i = F$ $\bigcap\limits_{i=1 \ldots l} F = \varnothing$ и $W(F_i) < K(t) \forall F_i \in \{F_1, F_2, \ldots, F_l\}$ имеет место $\Gamma^{-1}(\bar{F}_\alpha \cap \bar{F}_\beta) \cap \bar{F}_\alpha \cap \bar{F}_\beta \neq 0$ для некоторых $F_\alpha, F_\beta \in \{F_1, F_2, \ldots, F_l\}$.

Назовем вероятностно-диагностическим ядром системы S элемент v_i, $v_i \in V$, для которого $p(v_i) \leqslant p$.

Лемма 3. Если система S является p-диагностируемой с восстановлением для модели FB и среди V отсутствуют вероятностно-диагностические ядра, то $\Gamma^{-1}(v_i) \neq \varnothing$, $\forall v_i \in V$ и $\forall F \subseteq V$, $|F| = 2$, $\Gamma^{-1}(F) \neq \varnothing$.

Для систем с восстановлением и без восстановления, описываемых FP- и FB-моделями (FB- и FP-системы) имеется связь по p-диагностируемости. Эта связь проявляется в том, что p-диагностируемость отказоустойчивой системы по одной из моделей предопределяет p-диагностируемость для другой.

Теорема 11. Для любого значения $0 < p < 1$:

1) из p-диагностируемости FP-системы без восстановления следует p-диагностируемость: FP-системы с восстановлением и FB-системы без восстановления;

2) из p-диагностируемости FP-системы с восстановлением следует p-диагностируемость FB-системы с восстановлением;

3) из p-диагностируемости FB-системы без восстановления следует p-диагностируемость FB-системы с восстановлением.

6. ВЫБОР МОДЕЛЕЙ ВЗАИМОКОНТРОЛЯ

Выбор модели диагностирования реальных систем определяется комплексом факторов: допустимыми отказами и полнотой модели, целями диагностирования и возможностями обработки и использования диагностической информации, наибольшим количеством неисправных элементов, ограничением на длительность диагностирования и восстановления и т. д. В ряде случаев удается ввести численные характеристики, помогающие оценивать преимущества той или иной модели. Оценим возможности моделей на основе требований, которые они предъявляют к объекту диагностирования.

Выбор P-моделей связан с учетом следующих основных ограничений:

1. Каждый исправный контролирующий элемент структуры должен обеспечивать полный контроль каждого из контролируемых им элементов.

2. Получение результатов проверки контролирующим элементом контролируемого им элемента должно быть независимым.

3. Наиболее вероятным при отказе нескольких элементов является неисправность наименьшего количества элементов.

Основной особенностью B-модели, в отличие от P-модели, является необходимость обеспечения защиты в системе от одновременной неисправности контролируемого и контролирующего элементов при наличии результата контроля об исправности. Для использования R-моделей при диагностировании систем должны выполняться следующие требования:

1. Диагноз системы должен допускать возможность рассмотрения процесса диагностирования как прохождение или непрохождение тестов, обнаруживающих неисправности.

2. Неисправности и тесты должны быть четко определены. В частности, каждый тест из допустимого набора тестов должен быть полным лишь для единственной неисправности. Разбиение системы на элементы должно проводиться с учетом требуемой степени детализации атомарной неисправности и неисправных наборов, поскольку от этого существенно мо-

гут зависеть получаемые результаты (оценка полноты диагностирования, допустимое число неисправностей, состав связей в СВК и т. д.). В ряде случаев целесообразно моделирование нескольких возможных вариантов разбиения системы на неисправности и наборы неисправностей.

3. Должно обеспечиваться полное диагностирование компонентов системы для получения множеств полных тестов для каждой неисправности f_i, а для получения множеств $T(F_к)$, искажаемых при наличии наборов неисправностей $F_к$, необходима информация о включении элементов из $F_к$ в представление тестов.

4. Не должно существовать детерминированной связи (корреляции) между всеми искаженными результатами прохождения тестов для различных наборов неисправностей, а наличие неисправности не должно искажать свой собственный тест.

В результате невыполнения перечисленных требований, обусловленных спецификой модели, резко снижается эффективность использования диагностических структур при анализе и синтезе отказоустойчивых систем.

В К-моделях диагностирования, несмотря на большую общность, также имеются некоторые ограничения.

1. При построении функций и отображений возникают трудности выбора формы и полноты представления таких отображений, что влияет на основные цели их применения: априорную оценку глубины поиска дефектов, анализ результатов прохождения тестов или оценку данной модели по другим показателям диагностирования.

2. Построение К-моделей для реальных систем связано с большим объемом запоминаемой информации, что приводит к росту необходимых объемов запоминающих устройств и требует значительных временных затрат. Применение моделей диагностирования с использованием СВК требует развитой системы взаимосвязи между элементами с целью выполнения процедур контроля, передачи данных от одних элементов структуры к другим и анализа результатов прохождения тестов с целью получения результатов контроля.

Одна из принципиальных особенностей микропроцессорных систем по сравнению с существовавшими ранее системами — большой удельный вес при реализации функций системы программного обеспечения. В настоящее время до 80—90 % операций в системах выполняется программно.

Известно, что большинство логических функций, в том числе ряд функций контроля и диагностирования, достаточно просто реализуется программно. Например, функция сравнения содержимого двух байтов памяти между собой выполня-

ется с помощью 4—6 команд микропроцессора К580 и занимает 4—14 байт. Поэтому программная реализация процессов контроля и диагностирования в микропроцессорных системах является основной для обеспечения отказоустойчивости системы. Реализовать тесты в микропроцессорных системах можно путем применения внешних программ тестирования (непосредственно или с помощью интерпретаторов тестов) или за счет внутренних возможностей микропроцессоров, микропрограммного управления, генерации внутренних микроопераций, использования внутренней памяти, микрокоманд и т. д. Таким образом, при организации структур взаимоконтроля на основе микропроцессоров возможна обычная программная и микропрограммная генерация тестов и обработка их прохождения, т. е. микродиагностика неисправностей.

Под микродиагностикой обычно понимается множество диагностических процедур, организованных в виде микропрограмм. К основным достоинствам микродиагностики по сравнению с использованием систем команд машинных языков относятся:

1) более высокая резрешающая способность и полнота диагностирования. При использовании микроопераций и микроинструкций, входящих в микропрограммы, удается проводить диагностирование на уровне сложных модулей (например, микропроцессоров и запоминающих устройств) в пределах одной микрооперации;

2) возможность диагностирования в неавтономном режиме работы системы, т. е. одновременно с выполнением основных функций. Это осуществляется при включении в работу последовательности микроопераций и микропрограмм диагностических микроинструкций;

3) реконфигурация системы при обнаружении ошибки в выполняемой микрокоманде путем перехода на эквивалентную по функциям микрокоманду, выполняемую на исправных функциональных узлах, чем обеспечивается отказоустойчивость систем в реальном масштабе времени.

Обычно в существующих системах различают резидентную и нерезидентную части микродиагностики. Резидентная часть входит в состав самой системы, а нерезидентная находится на внешних устройствах и используется в процессе диагностирования. Так, резидентная часть диагностических микропрограмм может проверять ПЗУ, ОЗУ, ЦП, а нерезидентная часть — схемы контроля, управления приоритетом, мультиплексные и селекторные каналы, схемы защиты памяти.

Получение и передача тестов в отказоустойчивых микропроцессорных системах возможна централизованно, с ис-

пользованием отдельных ЭМ или внешних средств (тестеров, генераторов тестов), и децентрализованно, с помощью подмножеств ЭМ СВК. При децентрализованном генерировании тестов можно обеспечить высокую достоверность диагностирования системы за счет взаимно независимого получения тестов от нескольких источников (ЭМ диагностической структуры).

При использовании микродиагностики для генерации тестов используют микроинструкции, выполняющие машинные инструкции, а также специальные микрооперации и микропрограммы. В первом случае организация микродиагностики проще, а достоверность работы микропрограммы выше, однако недостатком является малая полнота диагностирования.

Программы диагностирования и контроля должны обеспечивать не только получение результатов контроля одного элемента другим, но и обработку этих результатов.

Решение задачи получения результатов контроля сводится к реализации тестов и не зависит существенно от вида модели. Обработка результатов контроля должна быть реализована на заведомо исправных элементах системы или путем использования резервирования (например, информационного или структурного). В этом случае целесообразно, в частности, применение нерезидентной части микродиагностики для проверки основных узлов, участвующих в диагностировании (для повышения достоверности диагностирования), и для изменения алгоритмов контроля текущего технического состояния системы в зависимости от работоспособности отдельных частей системы.

Таким образом, при диагностировании отказоустойчивых систем необходимо иметь данные о работоспособности отдельных ЭМ. Такие данные можно получить путем перечисления всех допустимых сочетаний состояний работоспособности элементов. В этом случае заданные синдромы и модели контроля позволяют ограничить априорное множество всех технических состояний диагностируемой системы. На данном множестве выделяются характерные неисправности, проводится в необходимых случаях контроль работоспособности отдельных элементов традиционными методами, что позволяет устранить некоторые недостатки, свойственные отдельным моделям.

Глава 3 АНАЛИЗ РАБОТОСПОСОБНОСТИ СИСТЕМ

1. ОЦЕНКА ТИПОВЫХ СТРУКТУР ВЗАИМОКОНТРОЛЯ

Контроль и диагностирование неисправностей в СВК при произвольных результатах контроля предполагает наличие некоторых априорных данных о характере и оценке количества неисправностей в системе. Одним из наиболее важных предположений для СВК, описываемых P-, B-, R-, K-моделями взаимодействия, является ограничение на максимальное количество одновременно неисправных элементов в системе, а для моделей FB, FP — наибольшая допустимая вероятность отказа.

Получение таких данных требует больших объемов статистических данных по эксплуатации системы, собрать которые сложно, поскольку отказы в отказоустойчивых системах происходят достаточно редко. Однако даже при наличии таких данных не гарантируется выявление критических отказов в системе (например, отказ определенных элементов системы, приводящий к непредсказуемому поведению системы, или отказ количества элементов, превышающего предельно допустимое). Наконец, при эксплуатации системы возможны ситуации, приводящие к неоднозначному определению технического состояния системы, что обусловливает необходимость анализа допустимых сочетаний состояний работоспособности элементов.

Решение перечисленных вопросов значительно упрощается при известных допустимых неисправностях системы для заданных синдромов, полученных в процессе контроля и диагностирования. Поэтому возникает задача перебора всех допустимых сочетаний отказов элементов при некоторых ограничениях, обусловленных известными априорными данными о состоянии элементов. В дальнейшем под допустимым набором неисправностей (ДНС) будем понимать совокупность наборов состояний исправности элементов системы, допустимых для заданного синдрома. Количество ДНС зависит от вида взаимодействия элементов в процессе диагностирования, направленности связей и значений синдромов, представляющих собой результаты контроля. Оценка количества ДНС для n элементов путем перечисления и проверки всех 2^n состояний элементов трудоемка даже при небольшом количестве

n. Поскольку в процессе такого перебора должна проверяться допустимость заданному синдрому, т. е. каждому из $|E|$ результатов контроля, то общее время вычисления количества ДНС будет пропорционально $2^n \cdot |E|$.

Одним из путей значительного сокращения трудоемкости этого процесса является использование разбиений графа $G = G(V, E)$ на типовые СВК, для которых можно, в частности, получить аналитические оценки для ДНС. К числу таких типовых структур СВК относятся направленные цепи, замкнутые контуры, звездообразные структуры.

Рассмотрим оценку ДНС в случае использования СВК, описываемых Р-моделями взаимодействия, для подмножества цепей L_j, для каждой из которых определены количество элементов $n_j = |V_j|$, структура связей элементов E_j, $E_j \subseteq E$, $|E| = n_j - 1$ и заданное множество A_j синдромов A_j^t:

$$A_j = \{A_j^t\}, \quad |A_j^t| \leqslant |E_j|.$$

Разбиение системы на множество подграфов $G_j = (E_j, V_j)$, удовлетворяющих указанным условиям, упрощает получение верхних и нижних границ количества ДНС.

Предположим, что цепочка L_j, если не оговорено противное, задана числом элементов n_j, множеством элементов V_j, множеством направленных связей контроля E_j и множеством A_j синдромов A_j^i, причем $|E_j| = |A_j^t|$, т. е.

$$L_j = L_j(n_j, V_j, E_j, A_j^i);$$
$$E_j = \{(v_i, v_{i+1}) | \forall v_i, v_{i+1} \in V_j \leftrightarrow \exists (v_t, v_{t+1}) \in E_j,$$
$$t = 1, \ldots, n_j - 1 \text{ и } \overline{\exists} v_l : (v_i, v_l) \in E_j, (v_l, v_i) \in E_j,$$
$$l \neq i + 1, i - 1\}.$$

Обозначим общее количество ДНС для цепочки L_j при заданном синдроме A_j^t, $A_j^t \in A_j$, через $N(L_j)_t$. Структуру называют максимальной, если в пределах всех возможных синдромов A_j^i и фиксированных E_j значение $N(L_j)_t$ будет максимальным, т. е.

$$N(L_j)_{\max} = \max_t N(L_j)\, t;$$
$$\forall A_j^t : |A_j^t| = |E_j|;$$
$$n_j = n_0 = \text{const}.$$

Для максимальных структур характерно следующее свойство постоянства числа ДНС, которое вытекает из следующей теоремы.

Теорема 12. Мощность множества ДНС для фиксированного значения n_0, $n_0 \geqslant 2$ для произвольных максимальных струк-

тур L_j, $n_j = n_0$ постоянна и определяется рекуррентным соотношением

$$\left.\begin{array}{c} \max\limits_{t} N(L_j) = f_{nj}; \\ f_n = X_{n-1} + Y_{n-1},\ n = 2, \ldots n_j; \\ X_{n-1} = X_{n-2} + Y_{n-2},\ Y_n = X_{n-2}; \\ X_0 = Y_0 = 1. \end{array}\right\} \qquad (3.1)$$

Доказательство проводим методом индукции.

Обозначим через X_n количество ДНС, оканчивающихся на одно состояние (исправен или неисправен), а через Y_n — оканчивающихся на второе состояние (неисправен или исправен) для первых n последовательно связанных элементов структуры L_j, причем $X_n > Y_n$, $n \geqslant 2$. Это следует из того, что $Y_n > 0$ при $n \geqslant 2$.

Для случая двух элементов, т. е. при $n_j = 2$, соотношение (3.1) является справедливым, поскольку в Р-моделях взаимодействия при фиксированном результате контроля v_i, v_{i+1} допустимым является только три набора состояний работоспособности элементов v_i и v_{i+1}.

Предположим, что соотношение (3.1) справедливо при некотором значении $n > 2$. Докажем его справедливость для $n = n + 1$.

Действительно, значение f_{n+1} может быть определено одним из двух соотношений:

$$f'_{n+1} = 2X_{n-1} + Y_{n-1} \qquad (3.2)$$

и

$$f''_{n+1} = 2Y_{n-1} + X_{n-1}. \qquad (3.3)$$

Однако $f_n = X_{n-1} + Y_{n-1}$, поэтому из соотношений (3.2) и (3.1) следует, что $f'_{n+1} = f_n + X_{n-1}$, а из (3.3) и (3.1), что $f''_{n+1} = f_n + Y_{n-1}$. Тогда, подставляя значения f'_{n+1} и f''_{n+1}, находим, что

$$f'_{n+1} - f''_{n+1} = X_{n-1} - Y_{n-1}.$$

Однако, поскольку $X_{n-1} > Y_{n-1}$ и $n > 1$, то $f'_{n+1} - f''_{n+1} \geqslant 0$. Следовательно, $f'_{n+1} > f''_{n+1}$ и максимальное количество ДНС определяется как $f_{n+1} = f'_{n+1} = 2X_n + Y_n$, что доказывает справедливость соотношения (3.1) для f_{n+1} при значении $n + 1$. Поскольку значение f_n определяется лишь числовым значением n_j, то f_n, при заданном $n_j = n_0$, является постоянным. Существование максимального L_j, удовлетворяющего соотношению (3.1), вытекает из допустимости произвольных синдромов A_j^i при фиксированной структуре связей E_j. Таким образом, теорема доказана.

Применяя доказанную теорему, можно оценить количество ДНС при произвольном значении числа элементов n_j последовательно связанных элементов v_t, $v_t \in V_j$ на основании следующих двух следствий.

Следствие 1. Наибольшее количество f_n ДНС для n элементов равно числу Фибониччи n-й степени:

$$f_n = f_{n-1} + f_{n-2}, \quad n \geqslant 1, \tag{3.4}$$

где $f_0 = 1$; $f_{-1} = 1$.

Действительно, производя замену $f_{n-1} = X_{n-1}$, $f_{n-2} = Y_{n-1} = X_{n-2}$ и подставляя полученные значения в соотношение (3.1), получаем $f_n = f_{n-1} + f_{n-2}$. При этом начальные условия определяются следующим образом: при $n = 1$ $f_1 = X_0 + Y_0 = 2$, т. е. $f_0 = X_0 = 1$ и $f_{-1} = Y_0 = 1$, что и доказывает следствие.

Следствие 2. Цепочка из элементов, имеющих синдром из единичных результатов контроля, является максимальной.

Данный результат можно получить непосредственной проверкой выполнения соотношения (3.1) при наличии единичных результатов контроля между элементами цепи. Назовем такие цепочки элементов I-структурами.

Теорема 12 и следствия 1 и 2 дают возможность оценивать свойства и верхнюю границу количества ДНС для произвольных структур взаимоконтроля.

Оценка количества ДНС вытекает из следующей леммы.

Лемма 4. Пусть $L_1, \ldots, L_k$ — совокупность цепочных структур, полученных из структуры $G = G(V, E)$, таких, что $\forall i, m \in \{1, \ldots, k\}$ $l \neq m$:

$$V_l \cap V_m = \varnothing, \; V_l, \; V_m \subseteq V.$$

Тогда количество допустимых наборов неисправностей N удовлетворяет неравенству

$$N \leqslant 2^{2r} \prod_{j=1}^{k} f_{n_j}, \tag{3.5}$$

где $n_j = |V_j|$; $r = |V \setminus (V_1 \cup V_2 \cup \cdots \cup V_k)|$.

Доказательство следует из свойств цепочных структур и введенных для них ограничений. Действительно, максимальное количество ДНС для каждой из цепочек L_j определяется согласно теореме 12 как f_{n_j}, при условии, что отсутствуют связи между конечными элементами цепи. Следовательно, наибольшее число для цепочек $L_1, \ldots, L_k$ не превышает $\prod_{j=1}^{k} f_{n_j}$.

Рассматривая оставшиеся m элементов, не входящих ни в одну из цепей, как изолированные, т. е. такие, исправность которых не зависит от исправности смежных с ними элементов,

получаем искомое соотношение (3.5), позволяющее оценить верхнюю границу количества ДНС при произвольном значении синдрома и тем самым определяющее диагностические возможности самой структуры при анализе системы.

Одной из основных задач диагностирования является определение участков диагностической структуры с неисправными элементами. Наиболее просто это сделать в структурах с заданными для них синдромами, которые имеют небольшое количество ДНС. При диагностировании в таких структурах достаточно просто перечислить все ДНС с последующим выделением из них наиболее характерных неисправностей.

По аналогии с ранее рассмотренными максимальными структурами L_j, введем определение минимальных структур.

Незамкнутую цепочку элементов L_j будем называть минимальной структурой, если при заданных E_j и множестве синдромов A_j^i количество ДНС является минимальным, т. е.

$$N(L_j)_{\min} = \min_i N(L_j), \quad i: \forall A_j^i | A_j^i | = | E_j |,$$

$$A_j^i \in A_j, \quad n_j = \text{const}.$$

Рассмотрим свойства минимальных структур.

Теорема 13. Мощность множества ДНС минимальных структур L_j при фиксированном значении количества элементов $n_j = n_0$ постоянна и определяется рекуррентным соотношением

$$f_n = X_{n-1} + Y_{n-1}, \tag{3.6}$$

где $X_{n-1} = X_{n-2} + Y_{n-2}$; $Y_{n-1} = Y_{n-2}$, $n \geqslant 2$; $X_0 = Y_0 = 1$.

Доказательство проводим методом индукции.

При значении $n = 2$ функция f_n из соотношения (3.4) является минимальной и равна 3 (количество ДНС для двух элементов).

Пусть функция f_n, определяемая соотношением (3.6), является минимальной при $n > 2$. Рассмотрим теперь значение f_n при $n + 1$. При этом возможны два случая.

В первом случае последний элемент $n+1$ имеет первое состояние исправности, совместимое с множеством из X_{n-1} ДНС, а второе — совместимое с X_{n-1} и Y_{n-1} ДНС. Тогда значение f'_{n+1} функции f_{n+1} определяется из соотношения:

$$f'_{n+1} = X_n + Y_n; \quad X'_n = X'_{n-1} + Y'_{n-1}; \quad Y'_n = Y'_{n-1}.$$

Во втором случае последний элемент $n + 1$ имеет первое состояние неисправности с множеством состояний Y_{n-1}, а второе — с множеством X_{n-1}, Y_{n-1} ДНС. Тогда значение f''_{n+1} функции f_{n+1} мощности ДНС определяется из соотношения

$$f''_{n+1} = X_n + Y_n; \quad X_n = X_{n-1} + Y_{n-1}; \quad Y_n = X_{n-1}.$$

Определим разность значений f''_{n+1} и f'_{n+1}:

$$f''_{n+1} - f'_{n+1} = 2X_{n-1} + Y_{n-1} - 2Y_{n-1} - X_{n-1} = X_{n-1} - Y_{n-1}. \quad (3.7)$$

Однако по индуктивной предпосылке

$$f_n = X_{n-1} + Y_{n-1}; \quad X_{n-1} = X_{n-2} + Y_{n-2}; \quad Y_{n-1} = Y_{n-2}.$$

Подставляя эти значения в выражение (3.7), получаем

$$X_{n-1} - Y_{n-1} = X_{n-2} > 0.$$

Из выражения (3.7) получаем, что $f''_{n+1} - f'_{n+1} > 0$, т. е. значение функции f'_{n+1} минимально.

Существование минимальных структур следует из того, что при любых множествах X_{n-1} и Y_{n-1} для последнего n-го элемента найдется соответствующее ему состояние последующего $n+1$ элемента, удовлетворяющего соотношению (3.6).

Данная теорема обеспечивает возможность вычисления минимальной оценки количества ДНС для структур взаимоконтроля.

Следствие 1. Минимальное количество ДНС для цепочки из n элементов равно $n+1$.

Из определения оценки мощности минимальной структуры значения X_{n-1}, X_{n-2}, ... образуют арифметическую прогрессию, поскольку $Y_{n-1} = Y_{n-2} = \cdots = Y_0 = 1$. Начальным членом арифметической прогрессии чисел X_{n-1}, X_{n-2}, ... является число $X_0 = 1$, а разностью прогрессии — $Y_0 = 1$. Следовательно, $X_{n-1} = 1 + (n-1) = n$, а $f_n = X_{n-1} + Y_{n-1} = n+1$, что и требовалось доказать.

Таким образом, наличие минимальных структур в СВК обеспечивает быстрый перебор ДНС и позволяет значительно сократить средства на поиск неисправностей.

Определим конкретный вид минимальных структур, удовлетворяющих условиям теоремы.

Следствие 2. Минимальной структурой является цепь элементов L_j из n_j элементов, имеющая синдром

$$A_j^0 = \{(a_{l.\,l+1} = 0)\}, \quad l = 1, 2, \ldots, n_j - 1. \quad (3.8)$$

Выделение минимальных структур в СВК обеспечивает возможность оценки нижних границ количества ДНС.

Лемма 5. Пусть L_j, $j = 1, \ldots, m$ — такие минимальные структуры, что $\forall sp : V_s \cap V_p \neq \varnothing$, $s, p = \{1, \ldots, m\}$,

$$\forall v_l \in V_s, \; v_r \in V_p \exists (v_l, v_r) \in E.$$

Тогда нижней границей числа допустимых ДНС структуры взаимоконтроля является $N \geqslant \prod_{j=1}^{m} n_j$.

Доказательство следует непосредственно из теоремы 13 и следствий к ней.

2. АНАЛИЗ ДОПУСТИМЫХ НЕИСПРАВНОСТЕЙ В СИСТЕМЕ

Целью диагностирования в отказоустойчивых системах является определение неисправных элементов. Применение СВК обеспечивает возможность выделения неисправных элементов при заданных предположениях и характерных отказах.

Определение характерных неисправностей в системе в общем случае возможно лишь после перечисления всех допустимых наборов неисправностей при заданных синдромах. Поскольку появление синдромов является случайным, то для оценки диагностических возможностей в системе целесообразно производить оценку верхних и нижних границ количества ДНС при фиксированной структуре связи элементов между собой в процессе контроля.

Количество ДНС зависит от СВК и синдромов, полученных в процессе проверки системы. Во многих практических случаях СВК удается представить совокупностью типовых СВК.

Различные структуры СВК можно реализовать для одних и тех же типовых структур многомашинных систем. Однако в процессе проверки целесообразно использовать дополнительные условия для контроля и диагностирования неисправностей в системе. Так, например, при отказах в шинах и кольцевых интерфейсах в однокольцевых однонаправленных системах наиболее рационально использовать рассмотренные ранее типовые цепочные СВК. Наиболее удобными при диагностировании неисправностей для распространенных в практике многомашинных систем типов С5, С6, С7, С8, С12 являются звездные и кольцевые СВК.

Рассмотрим вопрос о количестве ДНС для звездных типовых структур.

Лемма 6. Для звездных структур СВК количество ДНС не превышает $\prod_{j=1}^{m} f_{n_j-1} + \prod_{j=1}^{m} f_{n_j}$ и не меньше $\prod_{j=1}^{m} (n_j + 1) + 1$, где m — количество ветвей звезды; n_j — количество элементов в радиальной цепи j.

Доказательство проводим с использованием результатов, полученных для цепочных структур. Звездную структуру можно представить как совокупность цепочек $L_1, \ldots, L_m$, имеющих один общий для всех элемент $v_0 = V_1 \cap V_2 \cap \ldots \cap V_m$.

Количество ДНС для каждой из максимальных структур L_j равно f_{n_j-1}, исключая общий для всех цепей элемент v_0, который может принимать одно из двух состояний — исправен или неисправен. В случае $v_0 = 0$ произвольное состояние может иметь лишь второй элемент цепи, т. е. $v_2(j)$, $v_2(j) \in V_j$, $j = 1, \ldots, m$. Общее число допустимых ДНС в этом случае составит

$$N_1 = \prod_{j=1}^{m} f_{n_j-1}.$$

При $v_0 = 1$ элементы $v_1(j)$, $j = 1, \ldots, m$ могут принимать произвольные состояния, т. е. $N_2 = \prod_{j=1}^{m} f_{n_j}$. Следовательно, верхняя оценка количества ДНС определяется как

$$N_{\max} = N_1 + N_2 = \prod_{j=1}^{m} f_{n_j-1} + \prod_{j=1}^{m} f_{n_j}.$$

Минимальное количество ДНС оценим следующим образом. Предполагая однонаправленность цепей L_j, идущих от центрального элемента (рис. 14, *а*), получаем, что при результатах связей 0 центральный элемент для первого случая $v_0 = 0 \Rightarrow$ $\Rightarrow v_1 = 0$, $v_2 = 0$, $\ldots$, $v_{n_j} = 0$, т. е. $N_1 = 1$.

При $v_0 = 1$ из теоремы 13 следует, что количество ДНС

$$N_2 = \prod_{j=1}^{m} (n_j + 1).$$

Общее количество ДНС

$$N = N_1 + N_2 = \prod_{j=1}^{m} (n_j + 1) + 1.$$

Аналогично определяют количество ДНС во втором случае — для расходящихся от общего элемента v_0 минимальных структур L_j, $j = 1, \ldots, m$ (рис. 14, *б*). При $v_0 = 0$ $v_1 = 1 = v_{n_j} = 1$, т. е. $N_1 = 1$, а при $v_0 = 1$ количество ДНС $N_2 = \prod_{j=1}^{m} (n_j + 1)$.

Общее количество ДНС $N = \prod_{j=1}^{m} (n_j + 1) + 1$.

В третьем случае (рис. 14, *в*, *г*) при $v_0 = 0$ количество ДНС $N_1 = 1$, а при $v_0 = 1 \Rightarrow N_2 = \prod_{j=1}^{m} (n_j + 1)$. Следовательно,

$$N = \prod_{j=1}^{m} (n_j + 1) + 1.$$

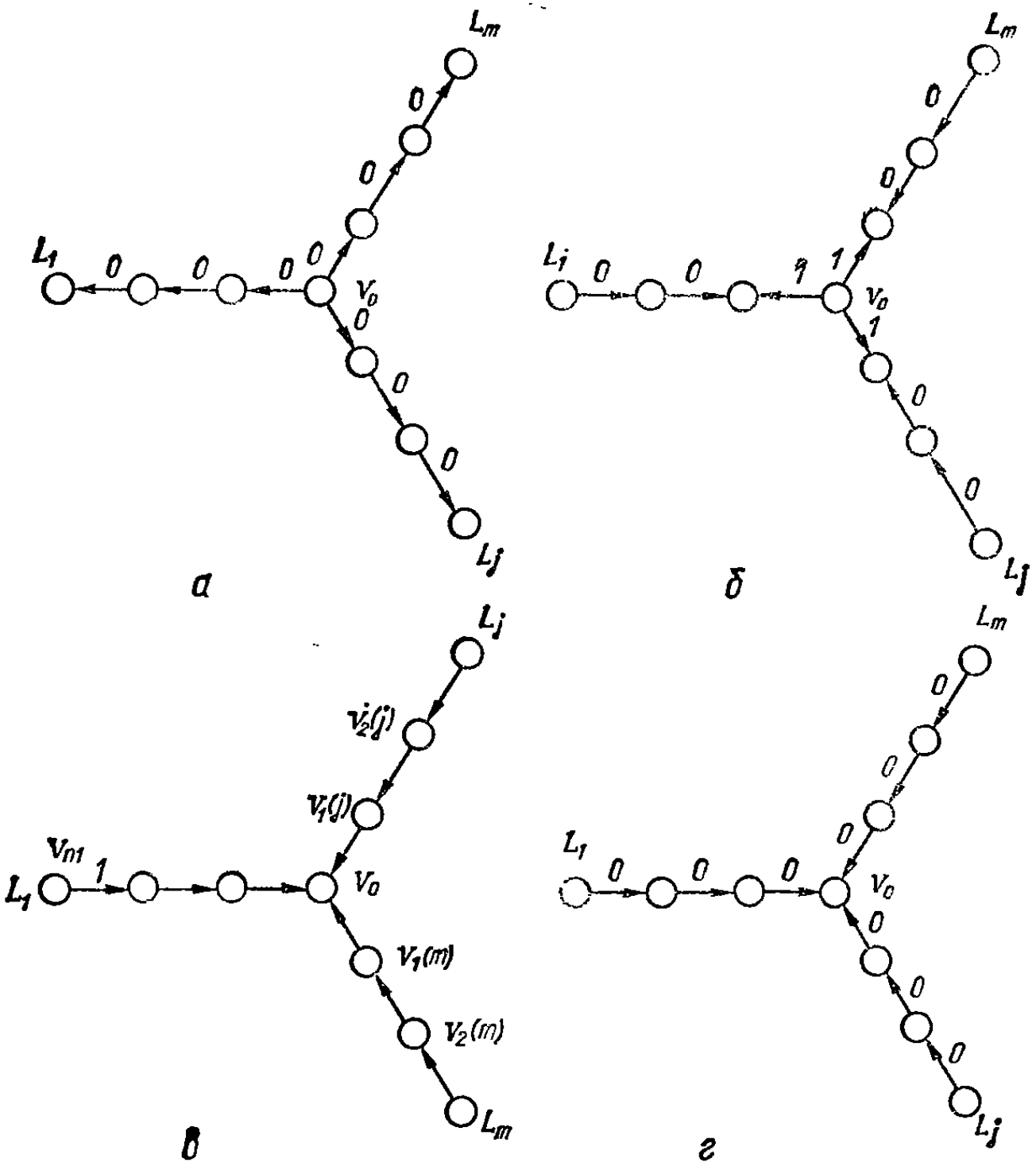


Рис. 14. Синдромы минимальных звездных структур

Таким образом, при объединении минимальных структур в звезду

$$N_{\min} = \prod_{j=1}^{m} (n_j + 1) + 1. \qquad (3.9)$$

Интерес представляет здесь сравнение полученных оценок со случаем последовательно соединенных элементов. Пусть $n_j = (n-1)/m$, $j = 1, \ldots, m$ является целым числом. Тогда максимальное количество ДНС

$$N_{\max} = \prod_{j=1}^{m} f_{n_j-1} + \prod_{j=1}^{m} f_{n_j} = (f_{(n-1)/m-1})^m [1 + (1 + f_{(n-1)/m-2}/f_{(n-1)/m-1})^m],$$

а минимальное определится как

$$N_{\min} = \prod_{j=1}^{m} (n_j + 1) + 1 = \left(\frac{n-1}{m} + 1\right)^m + 1.$$

Таким образом, при объединении в звезду, по сравнению с цепочными структурами, ухудшаются верхняя оценка $N_{\max}$ и нижняя $N_{\min}$ количества ДНС. Так, при $n = 13$, $m = 4$ для цепочной структуры $N'_{\max} = f_{13} = 610$, $N'_{\min} = 14$, а для звездной структуры

$$N_{\max} = (f_{(n-1)/m-1})^m + (f_{(n-1)/m})^m = (f_2)^4 + (f_3)^4 = (3)^4 + (5)^4 = 81 + 625 = 706; \quad N_{\min} = (3+1)^4 + 1 = 257.$$

Следовательно, для звездной структуры верхняя оценка $N_{\max}$ увеличилась в 1,2 раза, а нижняя $N_{\min}$ возросла в 14 раз.

Увеличение количества ДНС для произвольных СВК усложняет выделение характерных неисправностей и снижает эффективность использования результатов контроля при диагностировании системы. Кроме того, при большом количестве ДНС требуются большие объемы памяти и время обработки результатов контроля, что существенно влияет на стоимость и время поиска неисправностей в системе.

При отсутствии данных о взаимном контроле для $n = 13$ элементов каждый из элементов может произвольно принимать состояние исправен или неисправен. Поэтому общее количество ДНС в данном примере составит $N = 2^{13} = 8192$. Следовательно, звездные структуры обеспечивают сокращение количества ДНС в $W_n = N/N_{\max} = 8192/706 = 11{,}6$ раз.

В результате при анализе неисправностей в системе даже для самого неблагоприятного случая максимальных структур существенно сокращается объем вычислений или требуемый объем памяти по сравнению с цепочными структурами.

Рассмотрим кольцевые структуры.

Лемма 7. Для кольцевых структур минимальное количество ДНС равно 2, а максимальное $(f_{n-1} + f_{n-3})$, где n — общее количество элементов, $n \geqslant 3$.

Из рассмотренного ранее следует, что любая цепочка элементов, отличная от минимальной структуры, будет иметь большее количество ДНС. Поэтому рассмотрим минимальные структуры. Установим связь между первым и последним элементами цепочки и проанализируем, при каких значениях связи количество ДНС будет минимальным, а затем рассмотрим СВК с минимальными ДНС.

Для первого случая (рис. 15, *а*) при присвоении произвольному элементу состояния 1 все остальные элементы получают состояние 1, а при присвоении 0 — состояние 0. В этом случае элементы могут быть либо все исправные, либо все неис-

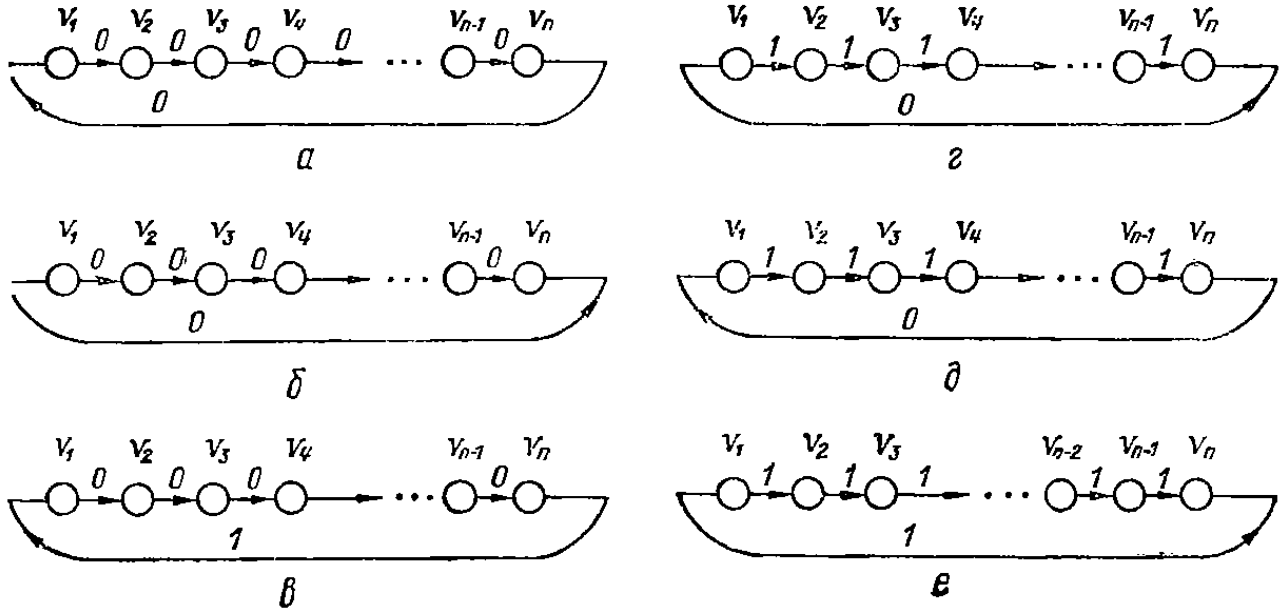


Рис. 15. Синдромы кольцевых структур

правные, т. е. $N = 2$. Для определения состояний всех элементов достаточно установить исправность хотя бы одного.

Во втором случае (рис. 15, *б*) при исправности одного элемента все остальные элементы исправны, а в случае его неисправности допустимыми являются все n наборов неисправностей, т. е. общее количество ДНС составляет $n + 1$ как и для минимальных структур. Следовательно, введение такой связи не оказывает практически влияния на количество и состав ДНС. Таким образом, минимальное количество ДНС для кольцевых структур равно 2, что можно использовать при диагностировании системы. Учитывая, что отказ всех элементов кольцевой структуры является маловероятным, достаточно просто устанавливается, что элементы этой структуры работоспособны. С этой целью между элементами кольцевой структуры устанавливаются однонаправленные связи. При получении результатов контроля «исправен» по всем связям можно с достоверностью говорить об исправности этих элементов.

Определение максимального количества ДНС для кольцевых структур производится с использованием свойств максимальных цепочных структур. Сокращение количества ДНС для кольцевых структур по сравнению с цепочными достигается за счет того, что часть наборов становится недопустимой (за счет недопустимого сочетания состояний элементов, замыкающих кольцо). Максимальную кольцевую структуру можно получить из максимальной цепочной структуры.

В третьем случае (рис. 15, *в*) элемент v_1 всегда будет неисправным, поэтому общее количество ДНС равно n, причем среди них отсутствует ДНС, имеющий все исправные элементы. Наименьшее сокращение количества ДНС достигается лишь при таком соединении крайних элементов цепочной структуры, при котором исключаемое множество ДНС является минимальным. Нетрудно проверить, что минимальное количество исключаемых ДНС будет в этом случае равно f_{n-4} (рис. 15, *г*, *д*, *е*). В результате общее количество ДНС определяется как

$$N_{\max} = f_n - f_{n-4} = f_{n-1} + f_{n-3}.$$

Таким образом, наибольшее сокращение количества ДНС как для нижней границы, так и для верхней достигается при наличии нулевых результатов контроля.

Сравнивая кольцевые структуры с минимальными и максимальными цепочными, можно сделать вывод об улучшении характеристик диагностируемости в первом случае. При этом верхняя граница количества ДНС снижается по меньшей мере на величину f_{n-4}, а минимальная граница может быть снижена до двух.

Применение кольцевых минимальных структур обеспечивает возможность однозначной идентификации неисправных элементов. Множество таких элементов называют ядром неисправностей данной структуры взаимоконтроля. Ядро неисправностей V_c можно определить таким образом:

$$V_c = V_0 \cup \{v_j \mid \exists v_l \in V_c \cup V_0 : (v_l, v_j) = 0\}, \qquad (3.10)$$

где $V_0 = \{v_j \mid \exists v_j, v_{j+1}, \ldots, v_{j+l} : v_j, v_{j+1}, \ldots, v_{j+l} \in V$ такие, что $(v_j, v_{j+1}) = 0$, $(v_{j+1}, v_{j+2} = 0, \ldots, (v_{j+m-1}, v_{j+m}) = 0$, $(v_{j+m}, v_j = 1)$ либо $(v_j, v_{j+m}) = 1\}$.

Использование ядра неисправностей является одним из эффективных средств определения технического состояния системы в целом.

Оценивая эффективность использования кольцевых структур для анализа технического состояния отказоустойчивой системы, можно отметить значительное уменьшение количества ДНС. Так, для рассмотренного ранее примера $n = 13$

сокращение числа ДНС по сравнению с полным перебором состояний работоспособности элементов достигается в $8192/(f_{12}+f_{10}) = 15{,}7$ раза.

Таким образом, применение СВК обеспечивает существенное сокращение множества ДНС и тем самым повышает эффективность контроля и диагностирования в отказоустойчивых микропроцессорных системах.

3. ДИАГНОСТИРОВАНИЕ ХАРАКТЕРНЫХ НЕИСПРАВНОСТЕЙ

Поиск неисправных элементов в СВК путем полного перечисления неисправностей приводит к большим затратам машинного времени. Поэтому вводятся некоторые предположения, упрощающие поиск. Распространенным является предположение о том, что наиболее вероятен отказ наименьшего числа элементов системы.

Диагностирование системы может выполняться однократно, что характерно при отсутствии возможностей замены элементов на заведомо исправные (диагностирование без восстановления) и многократно с использованием частичного или полного восстановления подозреваемых на неисправность элементов после каждого этапа диагностирования.

В случае диагностирования без восстановления алгоритмы поиска получаются сложнее, однако количество выявленных неработоспособных элементов оказывается как правило бо́льшим, чем при диагностировании с восстановлением. Поэтому имеется определенная взаимная дополняемость алгоритмов диагностирования с восстановлением и без восстановления.

Рассмотрим один из наиболее распространенных алгоритмов системы диагностирования без восстановления [46]. При поиске неисправностей предполагается, что количество t одновременно неисправных элементов не превышает числа $\tau(G)$, удовлетворяющего следующим условиям:

$$\tau(G) \leqslant \lfloor (n-1)/2 \rfloor;$$
$$\tau(G) \leqslant d_{\min}(G), \quad d_{\min} = \min_{v_i \in V} \Gamma^{-1}(v_i).$$

В процессе диагностирования по данному алгоритму различают множество элементов, технические состояния которых задаются произвольным образом (эксплицитные назначения), и элементы, состояния которых являются следствием эксплицитных присваиваний (имплицитные назначения). Возможны два случая имплицитных назначений: при эксплицитном назначении элементу v_i значения «исправен» ($v_i := 0$)

и «неисправен» ($v_i := 1$). В первом случае эксплицитное назначение порождает подмножества

$$N(i) = \{v_i\} \cup \{v_j \mid v_i := 0 \Rightarrow v_j := 0\};$$
$$F(i) = \{v_j \mid v_j := 0 \Rightarrow v_j := 1\}.$$

Во втором случае

$$N(i) = \varnothing;$$
$$F(i) = \{v_i\} \cup \{v_j \mid v_i := 1 \Rightarrow v_j := 1\}.$$

В результате выполнения алгоритма получается подмножество $V_р$ исправных элементов структуры $G = G(V, E)$ и подмножество $V_н$ неисправных элементов. В процессе выполнения используется граф $G'(V', E')$, получаемый при исключении вершин, входящих в подмножества $V_р$, $V_н$, и соответствующих им связей графа $G(V, E)$.

Состояния элементов перечисляются путем эксплицитного назначения состояний элементам с получением имплицитных подмножеств элементов и записью тех и других в стек. Запись элемента (множества элементов) в стек производится последовательно, в порядке поступления запроса на запись, а после считывания соответствующий считанный элемент (множество элементов) исключается из массива вершин.

Алгоритм 1.

1. Присвоить $V_р := \varnothing$, $V_н := \varnothing$; $G'(V', E') := G(V, E)$.

2. Выбрать $(v_i, v_j) \in E'$, $(v_i, v_j) = 1$. Если такой дуги нет, то $V_н$ — множество неисправных элементов и вычисление окончено.

3. Вычислить $N(j)$, $F(j)$, при эксплицитном назначении $v_j := 0$.

4. Если $N(j) \cap F(j) \neq \varnothing$, то принять $v_j := 1$ и вновь вычислить $N(j)$, $F(j)$.

5. а) если $|V_н \cup F(j)| \leqslant \tau(G)$ и $v_j = 0$, то получить $V_р : V_р \cup N(j)$, $V_н := V_н \cup F(j)$, поместить $(v_j, N(j) \cup F(j))$ в стек и перейти к шагу 2;

б) если $|V_н \cup F(j)| \leqslant \tau(G)$ и $v_j = 1$, то получить $V_н := V_н \cup F(j)$. Если значение $v_j = 1$ было присвоено на шаге 4, то перейти к шагу 2. В противном случае поместить множество $F(j)$ в стек и перейти к шагу 2;

в) если $|V_н \cup F(j)| > \tau(G)$, то исключить (читать) записи из стека вплоть до первой пары $(v_j, N(j) \cup F(j))$. В случае отсутствия такой пары в стеке, решения не существует и вычисление окончено. В противном случае все вершины, исключенные из стека, исключить из подмножества $V_н$ и $V_р$. В качестве элемента v_j принять элемент v_i связи (v_i, v_j). Произвести назначение $v_j = 1$, вычислить $N(j)$, $F(j)$. Повторить шаг 5.

Описанный алгоритм показан на рис. 16. Время вычисления допустимого набора неисправностей по данному алгоритму пропорционально произведению наибольшего количества неисправных элементов t и количества дуг $m = |E|$.

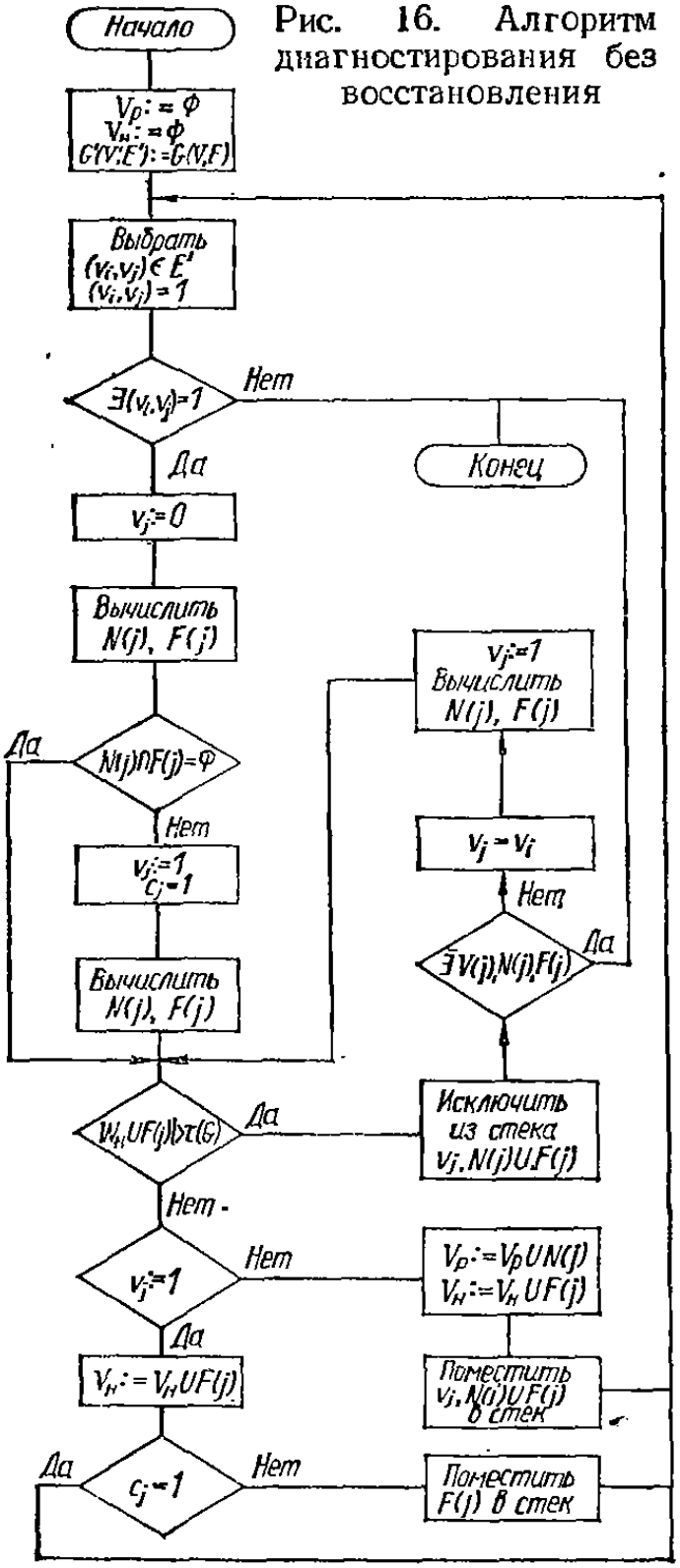


Рис. 16. Алгоритм диагностирования без восстановления

Оценим быстродействие алгоритма в случае отсутствия взаимонаправленных связей от одного элемента к другому. Максимально допустимое количество неисправностей t удовлетворяет условию $t \leqslant \lfloor (n-1)/2 \rfloor$, а m —условию $m \leqslant (n-1)n/2$. Тогда при $n > 1$ максимальное время вычисления допустимого набора неисправностей можно оценить как

$$W_1 \approx c_1 tm \approx c_1 n^3/4.$$

В качестве c_1 выбирают константу, определяемую максимальным временем вычисления одного набора неисправностей.

Если алгоритм не используется, то для вычисления всех ДНС необходим полный перебор состояний элементов V структуры $G = G(V, E)$, с проверкой допустимости каждого из получаемых наборов состояний при заданном синдроме. В этом случае максимальное время поиска ДНС оценивается как

$$W_2 \approx c_2 2^n.$$

Здесь в качестве c_2 также выбирают константу, соответствующую времени получения одного набора неисправностей.

Таким образом, при испльзовании алгоритма 1 удается сцизить максимальное время поиска приблизительно в $p =$

$= W_2/W_1$ раз, т. е.

$$p \approx (c_2/c_1)(4 \cdot 2^n/n^3) = c_3 2^n/n^3,$$

где $c_3 = 4c_2/c_1$.

При условии $c_1 = c_2$ (время цикла вычисления одинаково) и значениях $n = 10{,}20$ выигрыш по быстродействию p составляет приблизительно 4 и 500.

Таким образом, при $V > 10$ применение алгоритма позволяет существенно сократить максимальную оценку времени вычисления допустимого набора неисправностей.

В практике построения отказоустойчивых мультимикропроцессорных систем при наличии развитой системы связи отдельных ЭМ между собой представляется возможным введение новых, заведомо исправных микропроцессоров или микроЭВМ. Путем достаточно простых операций переадресации вновь вводимые элементы могут занять место неисправных элементов СВК. Неисправности в таких восстанавливаемых системах определяются путем последовательного обнаружения одного или более неисправных элементов на каждом этапе диагностирования с последующей заменой их на исправные. В результате значительно упрощаются не только алгоритмы контроля и диагностирования, но и сокращается время определения ДНС . При этом возможно диагностирование гораздо большего количества неисправных элементов, чем в случае структур без восстановления.

Рассмотрим алгоритм диагностирования с восстановлением для СВК, описываемых моделями P, B, R. Обобщенная модель тестирования для этих трех моделей структур является R-модель, для которой имеются наборы полных тестов h_1, h_2, ..., h_m для элементов v_1, v_2, ..., $v_n \in V$, причем предполагается, что для произвольного набора элементов v_i, v_j, ..., v_k полные тесты $h(\{v_i, v_j, \ldots, v_k\})$ являются объединениями полных тестов $h(v_i)$, $h(v_k)$ для отдельных элементов, т. е.

$$h(\{v_i, v_j, \ldots, v_k\}) = h(v_i) \cup h(v_i) \cup h(v_j) \cup \cdots \cup h(v_k).$$

Назовем агрегатированным синдромом упорядоченное множество $\sigma = \langle \sigma_1, \sigma_2, \ldots, \sigma_n \rangle$, где σ_i — общее число тестов для v_i, которые не проходят (дают результат неисправен) при данном множестве тестов $h_1, \ldots, h_m$.

Рассмотрим простейшие алгоритмы диагностирования по агрегатированным синдромам с использованием полной или частичной замены элементов структуры [55].

Алгоритм 2.

1. Определить агрегатированный синдром σ для множества тестов $h_1, \ldots, h_m$.

2. Если $F = \{v_j \mid \sigma_j \neq 0$ и v_j не заменен$\} = \varnothing$, то система считается исправной и процесс диагностирования окончен. В противном случае заменить все элементы в F и перейти к шагу 1.

Алгоритм 3.

1. Определить агрегатированный синдром σ для множества тестов $h_1, \ldots, h_m$.

2. Определить $F = \{v_j \mid \sigma_j \neq 0$ и v_j не заменен$\}$. Если $F = \varnothing$, то система считается исправной и диагностирование окончено. В противном случае выбрать

$$F' = \{v_i \mid v_i \in F : \forall v_j \in F,\ \sigma_i \geqslant \sigma_j\}.$$

3. Заменить все элементы в F' и перейти к шагу 1.

Алгоритм 2 состоит в замене всех элементов структуры при наличии, по крайней мере, одного результата непрохождения теста.

В алгоритме 3 замене подлежат последовательно все элементы, для которых число непрошедших тестов является наибольшим. Алгоритм можно естественно обобщить на случай замены нескольких элементов. Множество F' в этом случае следует выбирать исходя из фиксированного наибольшего количества элементов s

$$F' = \{v_k \mid v_k \in F \text{ и } \forall v_j \neq v_k,\ v_j \in F : \sigma_k \neq 0,$$
$$\sigma_k > \sigma_j,\ k \in (i,\ i-1,\ \ldots,\ i-s)\}.$$

В реальных системах реализация тестов $(h_1, \ldots, h_m)$ сопряжена с использованием элементов $(v_1, \ldots, v_n)$, неисправность которых может привести к искажению отдельных тестов.

Поэтому для всех множеств тестов $T(v_i)$, в формировании которых принимает участие элемент v_i, возможно искажение получаемых синдромов, т. е. множеств результатов контроля. При симметричном искажении теста $h_j \in T(v_i)$ тест $h_j \in h(v_j)$ может проходить (выдается результат «исправен») даже при неисправности v_j. При несимметричном искажении тест h_j не проходит (например, выдается результат «неисправен») при исправности элемента v_j.

Таким образом, для случаев симметричного и несимметричного искажений теста алгоритмы 2 и 3 приводят к полному восстановлению систем при любых состояниях работоспособности элементов системы перед началом диагностирования. При этом время вычисления по данным алгоритмам пропорционально количеству тестов m и количеству практически неисправных элементов t. Некоторое повышение быстродействия алгоритмов возможно за счет исключения из проверки тестов, проверяющих функционирование восстановленных (замененных) элементов структуры.

В практике диагностирования систем для сокращения времени диагностирования и нормирования количества восстанавливаемых элементов (например, в случае периодических регламентных проверок) целесообразно задаваться фиксированным числом k проверок и числом s заменяемых (восстанавливаемых) элементов.

В реальной системе количество неисправных элементов $t_c \leqslant t$. Поэтому для конкретных неисправностей в системе можно говорить о (t_c, s)-диагностируемости, т. е. диагностируемости при наличии отказов в t_c элементах при условии замены s элементов на исправные.

Использование алгоритма 2 обеспечивает, независимо от характера искажения тестов (симметричный или несимметричный), наименьшую $t_{c,\ T'+1}$-диагностируемость, но не более чем $t_{c,n}$ -диагностируемость. Здесь T' определяется как [55]

$$T' = \max_{v_i \in V} |T(v_i)|.$$

Например, для кольцевых структур взаимоконтроля, представляющих собой замкнутую цепочку элементов, обеспечивается наименьшая $t_{c,2}$-диагностируемость.

Использование алгоритма 3 при диагностировании неисправностей позволяет уменьшить, по сравнению с алгоритмом 2, количество заменяемых элементов при одном и том же количестве неисправностей t_c в системе. В частности, для системы с симметричным искажением теста использование алгоритма 3 обеспечивает $t_{c,\ T'-t'+2}$ -диагностируемость при условии, что $t' \leqslant T'$, а для систем с несимметричным искажением тестов — обеспечивает $t_{c,\ (T'/t')+1}$-диагностируемость, $t' = \min_{v_i \in V} |h(v_i)|$. Таким образом, применение алгоритма 3 диагностирования системы обеспечивает, по сравнению с алгоритмом 2, существенное сокращение количества заменяемых (восстанавливаемых) элементов системы.

В рассмотренных в данной главе алгоритмах определения допустимых наборов состояний неисправностей структур взаимоконтроля важное значение имеют структурные особенности построения системы, в частности, модели диагностирования неисправностей, направленность и количество связей. При этом в алгоритмах полного перебора ДНС и нахождения единственного ДНС основное значение имеют направленность и значение связей контроля между элементами. В алгоритмах диагностирования путем восстановления основное значение имеют сами результаты контроля, причем эффективность этих алгоритмов существенно возрастает при снижении максимального количества тестов, искажаемых при неисправности

отдельных элементов, и увеличении минимального количества тестов, контролирующих каждый отдельный элемент структуры.

Таким образом, важным аспектом повышения эффективности диагностирования СВК является построение такой организации связей между элементами, которые, с одной стороны, обеспечивают удобство ее анализа с точки зрения диагностируемости, синтеза систем с требуемыми характеристиками диагностируемости, и с другой, — обеспечивают быстрый поиск неисправностей и восстановление функционирования системы.

4. ОПРЕДЕЛЕНИЕ СОСТОЯНИЙ ЭЛЕМЕНТОВ СИСТЕМЫ

Полученные ранее результаты показывают возможность значительного сокращения количества ДНС при наличии большого количества связей элементов. При таком сокращении целесообразен перебор всех допустимых ДНС с последующим отбором тех из них, которые удовлетворяют заданным ограничениям. В качестве ограничений могут выступать: количество одновременно неисправных элементов системы, характерные сочетания неисправностей или наличие данных об исправности отдельных элементов и др.

Использование результатов контроля структур взаимоконтроля в процессе перебора ДНС позволяет существенно сократить время полного перебора состояний элементов. Поэтому при получении допустимых наборов необходимо учитывать исключение тех ветвей перебора, которые связаны с получением заведомо недопустимых сочетаний исправности элементов. В предложенном алгоритме запрет на продолжение недопустимых ветвей производится путем введения специальных подмножеств.

Для рассмотрения алгоритма определения всех допустимых состояний исправности элементов введем обозначения.

Обозначим через $V_{\text{р}}$ множество исправных элементов, а через $V_{\text{н}}$ — множество неисправных элементов:

$$V_{\text{р}},\ V_{\text{н}} \subseteq V,\quad V_{\text{р}} \cap V_{\text{н}} = \varnothing.$$

Каждому элементу $v_i \in V$ сопоставим два набора $M_0(i)$, $M_1(i)$, $M_2(i)$ элементов $\{v_j\}$, $v_j \in V$, $v_j \neq v_i$:

$$M_0(i) = \{v_j \mid (v_i,\ v_j) = 1 \text{ или } (v_j,\ v_i) = 1\};$$
$$M_1(i) = \{v_j \mid (v_j,\ v_i) = 0\};$$
$$M_2(i) = \{v_j \mid (v_i,\ v_j) = 0\}.$$

Присвоение некоторому элементу v_i значения 0 или 1 обозначим $v_i := 0$, $v_j := 1$ соответственно. Все элементы $v_i \in V$ $i = 1, \ldots, n$ упорядочены некоторым образом.

Алгоритм 4.

1. Задать начальные значения $V_р := \varnothing$, $V_н := \varnothing$, $i := 0$.

2. Увеличить i на 1 . Если полученное значение больше n, то перейти к шагу 6. В противном случае перейти к шагу 3.

3. Присвоить $v_i := 0$. Если $V_р \cap M_0(i) \neq \varnothing$ или $V_н \cap \cap M_2(i) \neq 0$, то перейти к шагу 4. В противном случае включить v_i в состав $V_р : V_р = V_р \cup v_i$ и перейти к шагу 2.

4. Уменьшить i на 1. Если $i < 1$, то процесс включения допустимых наборов неисправностей завершен и выполнение алгоритма окончено. В противном случае проверить принадлежность v_i множеству $V_р$. Если $v_i \in V_р$, то исключить v_i из $V_р$, т. е. $V_р := V_р \setminus v_i$ и повторить выполнение шага 4. Если $v_i \in V_н$, то включить v_i в $V_н : V_н := V_н \cup v_i$.

5. Проверить, является ли множество $V_р \cap M_1(1)$ пустым.

Если $V_р \cap M_1(i) = \varnothing$, то перейти к шагу 2.

Если $V_р \cap M_1(i) \neq \varnothing$, то перейти к шагу 4.

6. Полученное множество $V_р \cup V_н$ является допустимым набором неисправностей. Для получения следующего ДНС перейти к шагу 4.

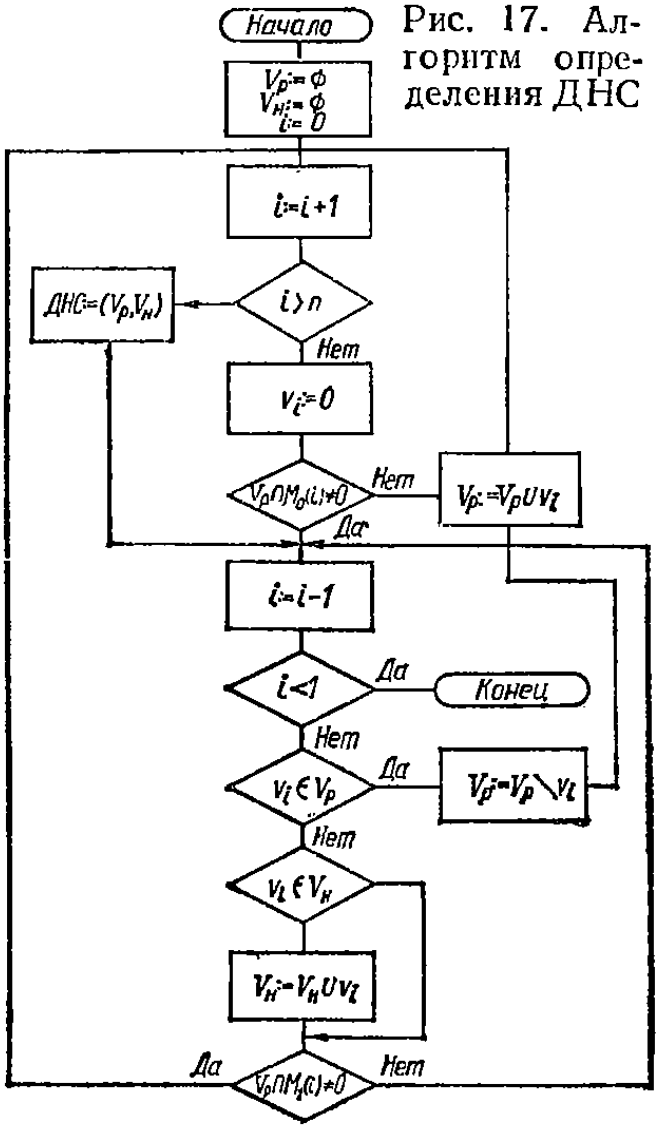


Рис. 17. Алгоритм определения ДНС

Общий вид алгоритма показан на рис. 17.

Выполнение данного алгоритма существенно ускоряется при введении упорядочения вершин множества V следующим образом. Вершины v_i, $v_i \in V$, $i = 1, \ldots, n$, имеющие меньшее количество связей, получают наименьшие индексы i, а имеющие большее количество связей — большие значения индексов i. В результате такого упорядочения сокращается количество продолжений дерева состояний элементов структуры взаимоконтроля.

Рассмотрим теперь алгоритм определения ДНС при использовании имеющихся данных об исправности отдельных элементов. Если известными являются подмножество $V'_р$ ис-

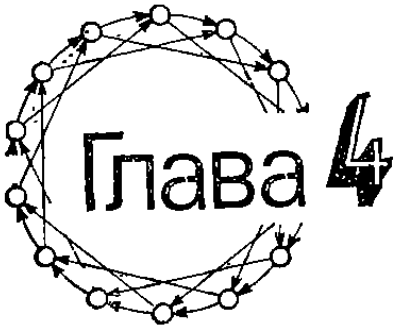
правных элементов и подмножество $V'_н$ неисправных элементов, то множество V, используемое в алгоритме, получается путем исключения данных элементов из рассмотрения, т. е.

$$V := V \setminus (V'_р \cup V'_н).$$

В дальнейшем к полученным наборам $V_н$ и $V_р$ эти элементы добавляются, т. е.

$$V_н := V_н \cup V'_н; \quad V_р := V_р \cup V'_р.$$

Более сложным является получение ДНС, имеющих определенное количество допустимых неисправных элементов t. В этом случае в шаге 4 алгоритма добавляется проверка на превышение $V_н$ значения t. Если такое превышение имеется, то происходит повторение шага 4.



Глава 4

ПРОЕКТИРОВАНИЕ ОТКАЗОУСТОЙЧИВЫХ СИСТЕМ

1. ОСНОВНЫЕ ЭТАПЫ ПРОЕКТИРОВАНИЯ

Реализация программ создания отказоустойчивых систем занимает в среднем 5—15 лет и зависит от уровня промышленного освоения новых элементов и устройств радиоэлектроники и вычислительной техники.

При целевом планировании разработок сложных систем можно выделить следующие стадии:

1. Анализ, теоретическая и экспериментальная проверка систем.

2. Разработка перспективных систем с учетом упреждения морального и технического старения.

3. Модификация существующих и разработка новых вариантов систем с целью удовлетворения новых требований к технико-экономическим параметрам системы.

Первая стадия сопровождается часто разработкой экспериментальных моделей и систем, что составляет в среднем 3—7 лет, включая техническое и рабочее проектирование, отработку документации и изготовление опытного образца.

Вторая стадия завершается отработанной базовой системой, которая служит обычно основой для последующих ее модификаций. Сроки создания систем колеблются в широких пределах: 3—10 лет в зависимости от качества начальных прора-

боток, правильности избранных принципов построения, стабильности и производительности основного контингента разработчиков, возможностей выхода на промышленное внедрение системы и т. д.

Третья стадия разработки системы обычно проводится параллельно с эксплуатацией базовой системы и позволяет улучшить ее технико-экономические показатели.

Вторая стадия во многих случаях обеспечивает основную эффективность и отдачу от применения системы, поскольку в конечном счете при правильной организации работ обеспечивает получение значительно лучших параметров и характеристик, охраноспособность и новизну системы с учетом сроков морального старения.

На стадии модификации разрабатываются новые варианты системы, совершенствуется обслуживание и корректируются технические решения по данным эксплуатации и в соответствии с новыми требованиями. Изменение технологии изготовления, создание более совершенного программного обеспечения, определение режимов наиболее эффективного использования системы могут служить важными факторами существенного повышения ее производительности и надежности.

На второй стадии разработки отказоустойчивой микропроцессорной системы можно выделить следующие этапы:

1. Определение целей введения отказоустойчивости и значений технических показателей систем (например, повышение коэффициентов технического использования и готовности системы на заданном интервале времени, сокращение затрат на обслуживание увеличения вероятности выполнения системой ответственных задач).

2. Формирование требований к структуре связей и их организации в системе с учетом особенностей использования технических средств.

3. Определение состава аппаратных и программных средств системы, корректировка решений, полученных на втором этапе.

4. Техническое проектирование с проработкой конкретных вариантов системы и моделированием ее работы.

5. Рабочее проектирование и создание опытного образца с последующей передачей в промышленную эксплуатацию.

Выполнение работ на каждом из перечисленных этапов так или иначе связано со структурой и особенностями организации физических и логических связей в системе. Например, выбор элементов системы с заданным интерфейсом взаимодействия элементов накладывает ограничения на скорость обмена данными. Последнее служит основой для разделения времени

функционирования элемента по выполнению диагностических и рабочих функций.

На втором этапе определяются основные требования к микропроцессорным отказоустойчивым системам, среди которых можно выделить следующие:

1. Защищенность системы от влияния аварийных неисправностей элементов.

2. Симметричность организаций СВК.

3. Возможность поиска неисправных элементов при малом количестве связей.

4. Наличие средств обеспечения перестройки структуры системы для выявления различных неисправностей.

5. Унифицированность программ диагностирования неисправностей.

Требование к реализации СВК при аварийных ситуациях («зависание» системы, непредсказуемое поведение, отказ основных линий связи и др.) состоит в обеспечении процесса поиска неисправных элементов в системе при отказе элементов, участвующих в диагностировании. Обычно для защиты системы от аварийных неисправностей используются различные способы информационного и структурного резервирования.

Симметричность СВК вытекает из удобства использования таких систем. В частности, симметричность СВК позволяет унифицировать алгоритмы диагностирования и программное обеспечение диагностирования элементов и системы в целом.

Введение большого количества связей в СВК может привести к увеличению времени диагностирования, повышению аппаратурной сложности реализации, усложнению синхронизации и обмена информацией между отдельными элементами. Поэтому поиск дефектов в СВК с наименьшим количеством связей является одним из важных факторов сокращения стоимости эксплуатации системы, повышения ее надежности. Вместе с тем, сокращение количества связей приводит к снижению допустимого количества диагностируемых неисправных элементов или к увеличению длительности проверки и восстановления системы. Разумный компромисс между количеством связей и заменяемых элементов системы обеспечивает эффективное диагностирование.

Перестройка структуры системы является одним из важных резервов обеспечения живучести системы и восстановления ее работоспособности. Используя перестройку структуры, можно обеспечить одновременное диагностирование одних элементов и работу по основному назначению других элементов системы.

Требование к унификации и простоте алгоритмов диагностирования обусловлено тем, что соответствующие им программы должны распределяться по элементам системы и работать в неблагоприятных условиях, при наличии большого количества неисправных элементов.

Отказоустойчивость системы позволяет обеспечить постепенное ухудшение (деградацию) функциональных характеристик системы при отказе большого количества элементов, что отличает их от обычных систем, в которых потеря работоспособности происходит при отказе одного из элементов.

В системе достаточно сложно бывает одновременно выполнять функциональное диагностирование и реализацию основных функций. При этом обычно усложняется программа обеспечения системы, в которой учитывается возможность «перекрещивания» процессов диагностирования и основного вычисления. Усложняется также система программных прерываний, использование оперативной памяти и каналов связи между элементами, что может привести к перегрузке каналов связи и другим последствиям. Наконец, организация параллельных процессов диагностирования и вычисления повышает требования к надежности отдельных элементов системы и приводит к дополнительным аппаратурным осложнениям. В результате, вместо ожидаемого улучшения свойств отказоустойчивости, может наблюдаться ее ухудшение.

Выходом из этого положения является перестройка структуры системы.

Перестройка может быть обеспечена за счет изменения связей между элементами (системы с перестраиваемой структурой связей) или путем замены или восстановления неисправных элементов системы (системы с перестраиваемым составом элементов). Выполнение системой основных функций производится на множестве исправных элементов (точнее, проверенных на предшествующих этапах контроля). Параллелизм диагностирования и функционирования достигается здесь при некотором снижении производительности системы.

Построение эффективности перестраиваемых СВК должно основываться на анализе возможных вариантов и включает решение следующих задач:

1. Выбор структуры связей элементов системы, обеспечивающей диагностирование и выполнение системой основных функций. Вид структуры зависит от требований к качеству проверки и производительности системы при решении основных задач.

2. Определение перечня элементов, входящих в состав СВК, из числа свободных от функционирования. В процессе работы выбираются те из них, которые обеспечивают вы-

полнение поставленных задач контроля и диагностирования.

3. Оценка характеристик эффективности функционирования системы. Нахождение таких характеристик дает возможность определить потенциальные возможности и недостатки системы и своевременно перестроить процесс вычисления и диагностирования системы таким образом, чтобы обеспечить необходимую ее отказоустойчивость.

2. ВЫБОР СТРУКТУР ВЗАИМОКОНТРОЛЯ

При решении задачи построения структур взаимоконтроля необходимо учитывать различные параметры процедур диагностирования неисправностей в системе. В зависимости от выбранного способа диагностирования СВК различаются количеством и структурой связей элементов, временем и стоимостью восстановления системы и т.п. В конечном счете это сказывается на общей стоимости обслуживания системы и ее производительности. Так, например, при диагностировании с восстановлением требуется небольшое количество связей взаимодействия между элементами, однако необходимо большее количество шагов диагностирования, замены и восстановления неработоспособных элементов в системе по сравнению с диагностированием без восстановления. При диагностировании без восстановления используется лишь одна процедура поиска неисправностей и восстановления работоспособности системы, но время выполнения этой процедуры может быть достаточно большим.

Поскольку увеличение времени диагностирования и восстановления приводит к возрастанию стоимости обслуживания, снижению производительности системы, то при выборе СВК необходимо проводить оценку системы по параметрам.

Пусть каждая из структур взаимоконтроля характеризуется набором параметров $\{b_{i,j}\}$. Параметры $b_{i,j}$, $i = 1...n$, $j = 1...m$ задают, например, кратность обнаруживаемых неисправностей при диагностировании с восстановлением и без него, количество резервных элементов, минимальное количество элементов, обеспечивающих отказоустойчивость системы и т. д.

Каждому варианту l диагностирования и восстановления можно сопоставить вектор-столбец $\{c_{l,i}\}$, $l = 1...m$, элементу $c_{j,l}$ которого соответствует «стоимость» или «вес» соответствующего параметра j из подмножества $\{b_{i,j}\}$. Так, например, кратности обнаружения неисправности в стратегии l может соответствовать определенное время поиска неисправных элементов системы.

Рассмотрим матрицу D, каждый элемент которой определяется как

$$d_{k,t} = \sum_{j=1}^{m} b_{k,j} c_{j,t},$$

т. е. D есть произведение матриц

$$B = \{b_{t,j}\} \text{ и } C = \{c_{j,t}\}, \text{ т. е. } D = BC.$$

Каждая строка матрицы D соответствует «взвешенным» оценкам соответствующей СВК при различных стратегиях диагностирования и восстановления.

Введем некоторую общую оценку $d_{j,0}$ строки j матрицы D. В качестве такой оценки могут выступать, например, максимальные, минимальные приведенные относительно некоторого значения элементов строки. Значение $d_{j,0}$ соответствует количественной оценке качества структуры j для различных стратегий диагностирования и восстановления из множества заданных.

Обозначим через $d_{0,i}$ некоторую количественную меру для столбца j матрицы D. Она будет характеризовать качество стратегии i для множества заданных структур.

В зависимости от задаваемых требований, используя $d_{j,0}$ и $d_{0,i}$, можно определить соответственно целесообразность применения структуры j или стратегии диагностирования и восстановления i.

Вводя упорядочение на множестве $d_{i,j}$ по строкам и столбцам, можно рассчитать показатели функционирования системы для подмножеств структур и стратегий. Вводя граничные значения $d^{\text{г}}_{0,i}$, $d^{\text{г}}_{0,j}$, можно отбросить заведомо бесперспективные варианты структур и стратегий. Это позволяет выбирать наилучшие варианты одновременно на множестве структур и стратегий, в наибольшей степени удовлетворяющих всем требованиям.

Рассмотрим пример оценки СВК с использованием изложенного способа.

Предположим, что каждая структура S_j характеризуется кратностью обнаруживаемых неисправностей $b_{j,1}$, числом $b_{j,2}$ элементов, при которых система считается работоспособной, количеством допустимых синдромов $b_{j,3}$. Пусть имеется четыре стратегии i ($i = 1, 2, 3, 4$) с оценками $c_{j,t}$, каждая из которых характеризует значение соответствующего параметра $j = 1...3$ структур S_j. Так, например, если кратности $b_{j,1}$ соответствует определенное время непрерывной работы, значению $b_{j,2}$ соответствует коэффициент снижения призводительности системы (выраженного к единицах времени), а $b_{j,3}$— время обработки одного синдрома с учетом вероятности

восстановления, то элементы $d_{j,\,i}$ матрицы D характеризуют качество отказоустойчивой системы, выраженное в единицах времени.

Пусть для определенности

$$B = \begin{vmatrix} 2 & 2 & 30 \\ 3 & 1 & 20 \\ 3 & 2 & 10 \end{vmatrix},$$

$$C = \begin{vmatrix} 1 & 1 & 2 & 2 \\ 0{,}2 & 0{,}3 & 0{,}1 & 0{,}2 \\ 0{,}01 & 0{,}02 & 0{,}02 & 0{,}01 \end{vmatrix},$$

тогда

$$D = BC = \begin{vmatrix} 2{,}7 & 3{,}2 & 4{,}8 & 4{,}7 \\ 3{,}4 & 3{,}7 & 6{,}5 & 6{,}4 \\ 3{,}5 & 3{,}8 & 6{,}4 & 6{,}5 \end{vmatrix}.$$

При выборе в качестве оценки минимальных простоев системы

$$d_{1,0} = \min_{j=1,\,2,\,3,\,4} d_{1,\,j} = 2{,}7;$$

$$d_{2,0} = \min_{j=1,\,2,\,3,\,4} d_{2,\,j} = 3{,}4;$$

$$d_{3,0} = \min_{j=1,\,2,\,3,\,4} d_{3,\,j} = 3{,}5$$

получаем, что наилучшей является первая структура, для каждой $d_{1,\,0} = 2{,}7$.

Определяя аналогично $d_{0,\,j}$, получаем, что наилучшей является первая стратегия диагностирования и восстановления. Если ввести новое значение

$$d'_{i,\,j} = \begin{cases} 0 & \text{при } d_{i,\,j} \geqslant 3{,}5; \\ 1 & \text{при } d_{i,\,j} < 3{,}5, \end{cases}$$

то получаем модифицированную матрицу

$$D' = \begin{vmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{vmatrix}.$$

Нулям соответствуют неудовлетворительные стратегии и структуры отказоустойчивой системы. Поэтому «усеченная» матрица

$$D'' = \begin{vmatrix} 1 & 1 \\ 1 & 0 \end{vmatrix}$$

описывает подмножество структур и стратегий обслуживания отказоустойчивой системы, удовлетворяющих заданным временным характеристикам. В частности, удовлетворительны-

ми считаются первые две структуры и стратегии диагностирования и восстановления. Необходимо отметить, что с помощью матриц D', D'' можно решать задачи расчета и оптимизации вариантов технических решений отказоустойчивых систем.

Для проведения более полного анализа необходимо строить недетерминированные модели процесса контроля и диагностирования, которые позволяют проводить более полную оценку на основе использования методов моделирования и статистических характеристик работы системы.

С учетом изложенного выше можно предложить следующий обобщенный алгоритм построения структур отказоустойчивых систем:

1. Выбрать и оценить различные виды структур взаимоконтроля по количеству связей, числу обнаруживаемых неисправностей.

2. Выбрать и оценить временные и аппаратурные затраты заданного множества допустимых для данных видов структур алгоритмов поиска неисправностей.

3. Оценить время восстановления элементов системы для выбранных алгоритмов поиска неисправностей.

4. Выбрать способ построения, алгоритмы поиска и восстановления, приводящие к минимальным затратам по обеспечению отказоустойчивости системы. Если получение такого минимума невозможно, то при заданных ограничениях на числовые значения одних характеристик добиться наименьших затрат по другим характеристикам.

5. Если не все варианты структур рассмотрены, то перейти к шагу 1, изменив количество неисправных элементов или аварийных неисправностей системы.

6. Провести корректировку структуры взаимоконтроля или модификацию полученной структуры для снижения затрат на восстановление системы.

Исходя из требований диагностируемости, удается построить большое разнообразие структур взаимоконтроля, различающихся по сложности реализации связей, возможностям использования алгоримов диагностирования. Получаемые в процессе диагностирования результаты контроля могут существенно упростить структуру взаимоконтроля, если удается локализовать значительную часть неисправных элементов, либо, наоборот, могут привести к необходимости расширения количества связей с целью нахождения неисправностей (в частности, аварийных).

Таким образом, возникает задача быстрой перестройки структуры систем. Зная, в частности, наибольшую кратность отказов, удается построить СВК, обеспечивающие локализацию неисправностей для группы или всех элементов системы.

При оценке систем удобно использовать аналитические способы построения СВК, обеспечивающие получение вариантов СВК с малыми затратами памяти и времени с помощью простых алгоритмов. Рассмотрим один из таких способов.

Система S описывается $D_{\delta t}$-структурой, если связь контроля от элемента v_i к элементу v_j существует только в том

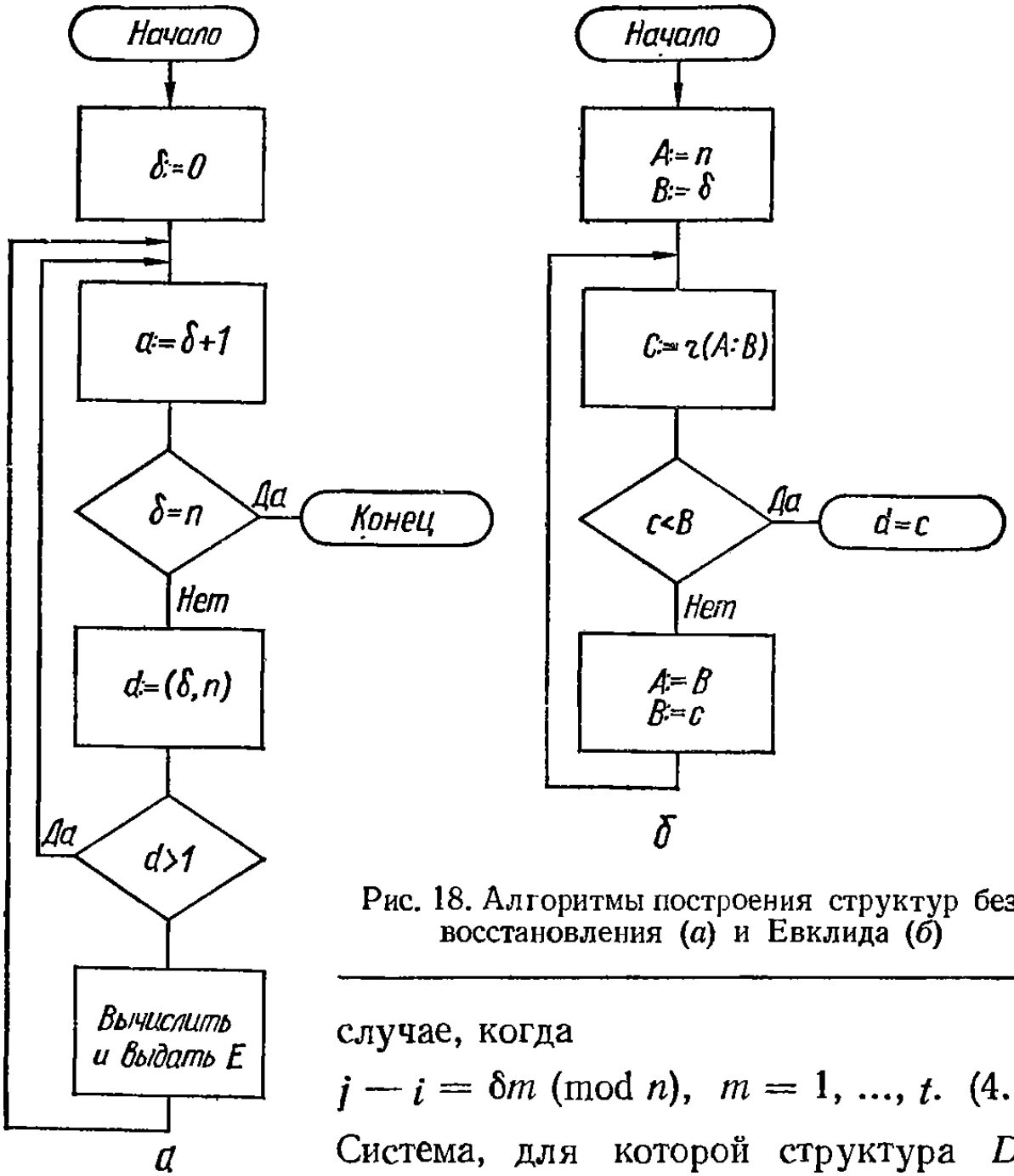


Рис. 18. Алгоритмы построения структур без восстановления (*а*) и Евклида (*б*)

случае, когда

$$j - i = \delta m \pmod{n}, \quad m = 1, \ldots, t. \quad (4.1)$$

Система, для которой структура $D_{\delta t}$ удовлетворяет условию взаимной простоты чисел δ и n, является t-диагностируемой без восстановления [46]. Следовательно, при заданном значении количества элементов $n = |V|$ существует множество СВК, удовлетворяющих требованию t-ДС.

Исходя из соотношения (4.1) получение допустимых структур взаимоконтроля можно проводить перечислением всех взаимно простых чисел δ и n. Поскольку все графы $G = G(V, E)$ при фиксированных взаимно простых числах δ и n являются изоморфными между собой, т. е. существует

взаимно однозначное соответствие между вершинами графов, при котором сохраняются соотношения инцидентности вершин множества V, то их множества допустимых структур выбирают СВК с наименьшими показателями стоимости и наибольшими показателями эффективности. В соответствии с выражением (4.1) алгоритм получения t-диагностируемых без восстановления СВК имеет следующий вид:

1. Принять δ равным 0.
2. Вычислить $\delta = \delta + 1$.
3. Если $\delta = n$, то конец. Все структуры перечислены. Если $\delta < n$, то перейти к следующему шагу.
4. Определить наибольший общий делитель чисел δ и n.
5. Если наибольший общий делитель больше единицы, то перейти к шагу 2. В противном случае перейти к следующему шагу.
6. Вычислить и выдать структуру взаимоконтроля, удовлетворяющую заданным значениям δ, t и n.
7. Перейти к шагу 2.

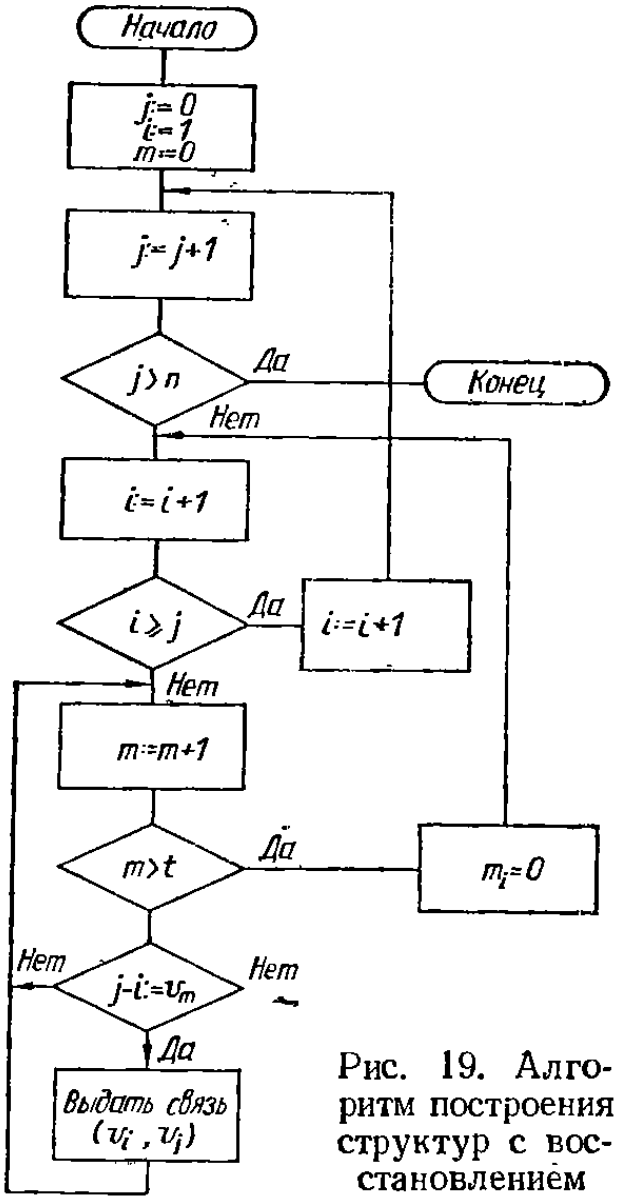


Рис. 19. Алгоритм построения структур с восстановлением

Шаг 4 может быть выполнен, например, с использованием алгоритма Евклида, а шаг 6 алгоритма — перебором допустимых связей при заданных δ, t, n.

На рис. 18,*а* показан общий вид предложенного алгоритма, а на рис. 18,*б* — реализация, соответственно, алгоритма Евклида для шага 4 и для шага 6 построения структур. Здесь через d обозначен наибольший общий делитель для δ и n, а $c = r$ (A/B) — остаток от деления A на B.

Простота алгоритма обеспечивает быстрое построение СВК при проектировании системы и при перестройке по результатам диагностирования и восстановления в процессе эксплуатации. В ряде случаев оказывается целесообразно строить СВК с ограниченным числом связей.

Обозначим наибольшее количество связей, которые могут иметь элементы v_1, v_2, ..., $v_n \in V$, входящие в v, через d_j^+,

а выходящие — через d_j^-, тогда СВК могут быть построены первоначально для

$$d_{\max}^+ = \max_{j=1,\ldots,n} d_j^+ \text{ и } d_{\min}^- = \min_{j=1,\ldots,n} d_j^- \qquad (4.2)$$

с последующим дополнением оставшихся связей между остальными элементами. При построении t-диагностируемых СВК с восстановлением наиболее просто использовать кольцевые структуры, для которых условие t-диагностируемости имеет следующий вид [41]:

$$n \geqslant 1 + (m+1)^2 + \lambda(m+1), \qquad (4.3)$$

где m — целое число; $t = 2m + \lambda$ и λ принимает значение 0 или 1.

Для заданного числа n наибольшее t может быть определено следующим образом. Перебирая значения $m = 1, 2, \ldots$ находят первое из них, удовлетворяющее соотношению (4.3) при $\lambda = 0,1$. Полученное значение t сравнивается с наибольшим заданным для системы $t_{\max}$. Если требования t-диагностируемости с восстановлением не удовлетворяются ($t < t_{\max}$), то дополняем кольцо новыми связями в следующем порядке. Выбираем элементы, которые могут иметь не меньше, чем $2t-1$ связей, и образуем связи со смежными с ними $2t - 2$ элементами. Алгоритм, поясняющий принцип построения таких СВК, показан на рис. 19.

3. СИНТЕЗ ПЕРЕСТРАИВАЕМЫХ СТРУКТУР

Преимущества отказоустойчивых систем в наибольшей степени проявляются в возможности непрерывного достоверного функционирования. Можно выделить два способа контроля и диагностирования: путем проверки элементов, выполняющих основные функции (функциональный контроль и диагностирование), и путем перевода подмножеств проверяемых элементов в режим диагностирования. Проверенные исправные элементы во втором случае «возвращаются» в состав элементов, выполняющих основные функции.

При втором способе не требуется разделение времени работы одних и тех же элементов для выполнения основных и контрольных функций. В результате упрощается программное обеспечение и алгоритмы контроля и диагностирования. Последовательная проверка элементов системы в этом случае производится путем перестройки структуры связей между элементами. В процессе такой перестройки образуются два основных подмножества элементов: контролируемых и выпол-

няющих основные функции. Часть элементов может служить в качестве резервных.

Построение структур и рациональный выбор элементов в процессе перестройки определяется конкретными требованиями к диагностированию системы. Поэтому ниже изучаются принципы построения перестраиваемых структур с учетом стоимостных параметров эксплуатации системы. Для определенности в основу перестраиваемых структур положены теоремы, полученные [38, 39] при t-диагностируемости для D_{1L}-структур [51].

В структурах D_{1L} связь между элементами v_i и v_j существует лишь в том случае, когда выполняется равенство $j - i = \text{mod } n$, $n = |V|$ СВК. Рассмотрим первоначально случай диагностирования без восстановления.

Теорема 14. В D_{1L}-структурах при исключении произвольных B элементов ($1 \leqslant B \leqslant L - 1$) обеспечивается t-диагностируемость без восстановления, где

$$t = \min \{L - B,\ [(n - B - 2)/2]\}.$$

Следовательно, с увеличением количества B элементов системы, используемых для решения основных задач, снижается число обнаруживаемых неисправных элементов. Вместе с тем повышается эффективность работы системы за счет использования этих элементов по своему прямому назначению.

Обозначим через $t_к$ время диагностирования с помощью СВК. Предположим, что период между очередными профилактическими проверками равен t_0. Полагая, что стоимость работы элемента системы в единицу времени составляет c_1, стоимость эффекта от работы элемента системы c_2, получаем ориентировочную оценку для показателя эффективности вычисления системы в течение периода t_0:

$$W_0 = nc_2 (t_0 - t_к) - (n - B) c_1 t_к.$$

При полном исключении элементов системы на время контроля от выполнения основных функций возможно некоторое снижение значения времени $t_к$ до значения $t'_к$. Показатель эффективности системы в этом случае будет иметь вид

$$W_1 = nc_2 (t_0 - t'_к) - nc_1 t'_к.$$

Можно определить также нормированный показатель эффективности K системы, который равен отношению показателя W_0 системы к показателю W_1, т. е.

$$K = \frac{W_0}{W_1} 100 = \frac{nc_2 (t_0 - t_к) - (n + B) c_1 t_к}{nc_2 (t_0 - t'_к) - nc_1 t'_к} 100,$$

или при $t_к = t'_к$

$$K = (1 + Bc_1 t_к / W_1)\, 100.$$

Таким образом, значение показателя K растет линейно с ростом количества элементов B, освобожденных от контроля.

Вместе с тем при использовании связей элементов на время $t_к$ возрастает коэффициент полезного использования связей системы. Численное значение этого коэффициента можно определить как отношение времени использования связей для элементов B к общему времени использования связей в структуре $D1_L$, т. е.

$$K_L = \frac{t_к BL + L(n-1)(t_0 - t_к)}{(t_0 - t_к) L (n-1)} = 1 + \frac{t_к B}{(t_0 - t_к) L (n-1)}.$$

Таким образом, общая эффективность системы (по времени и стоимости выполнения заданных основных функций) возрастает при увеличении количества элементов, исключенных на время диагностирования из СВК. Поэтому при построении систем важно определить наибольшее количество элементов, выполняющих основные функции системы, при котором не ухудшается качество диагностирования.

Частные решения этой задачи вытекают из следующих теорем.

Теорема 15. Структура D_{1L} является t-диагностируемой без восстановления, если при $n \geqslant 2L + 1$, положительном значении k, таком, что $kL \leqslant n \leqslant (k+1)L$, и любом $1 \leqslant t \leqslant L$ возможен выбор элементов $B = k(L - t) + \alpha$, где $\alpha = 0$, если $n - kL - t \leqslant 0$, и $\alpha = n - kL - t$, если $n - kL - t > 0$.

Одна из возможных процедур построения t-диагностируемых перестраиваемых систем состоит в следующем:

1. Обозначить первые из элементов $v_1, \ldots, v_t$ множества V, входящими в число элементов B, а элементы $v_{t+1}, \ldots, v_L$ — входящими в СВК, повторяя такое присвоение k раз.

2. Среди оставшихся $n - kL$ элементов присвоение производим в следующем порядке. Если $n - kL \leqslant t$, то вводим оставшиеся $n - kL$ элементов в состав СВК. При $n - kL > t$ элементы $v_{kL+1}, \ldots, v_{kL+t}$ вводим в состав СВК, а оставшиеся элементы $v_{kL+t+1}, \ldots, v_n$ вводим в число элементов B, участвующих в выполнении системой основных функций.

В более общем случае построения структур имеет место следующий результат.

Теорема 16. В t-диагностируемых без восстановления структурах D_{1L} при $L + 1 \leqslant n \leqslant 2L$ возможно выделение B

элементов, $B = n - 2t - 1$, не участвующих в диагностировании.

Процедура построения таких структур проводится аналогично предыдущей и состоит в следующем.

Если $L - t \geqslant B$, то обозначаемые элементы $v_1, \ldots, v_B$ относим к «вычислительному» ядру, а оставшиеся $v_{B+1}, \ldots, v_{n-B}$ — к СВК.

В случае $L - t < B$ элементы $v_1, \ldots, v_{L-t}, v_{L+1}, \ldots, v_{L-t+1}$ относим к элементам, выполняющим основные функции системы, а элементы $v_{L-t+1}, \ldots, v_L - v_{n-t}, \ldots, v_n$ — к элементам, выполняющим функции контроля.

Для СВК с восстановлением диагностируемость определяется возможностью обнаружения по крайней мере одного неисправного элемента. Поэтому определение наибольшего количества элементов B производится на основе построения СВК, способной обнаруживать неисправность хотя бы одного элемента.

При наибольшем количестве t неисправных элементов системы в любой цепочке из $t + 1$ элементов обнаруживается неисправность хотя бы одного элемента. В более общем случае разбиения структуры взаимоконтроля на $t + 1$ цепочек $L_1, L_2, \ldots, L_{t+1}$ длины $n_i \geqslant i$, не имеющих между собой связей, также идентифицируется неисправность, по крайней мере, одного элемента. Это означает, что при исправных элементах возможно появление всех нулевых результатов в синдромах СВК, а само определение конкретного неисправного элемента сводится к задаче перечисления всех допустимых наборов неисправностей, которая была подробно рассмотрена в гл. 3.

Таким образом, на основе данных свойств структур с восстановлением становится ясным, что число B будет больше, чем для структур без восстановления. Это находит свое подтверждение в следующей теореме.

Теорема 17. В системе D_{1L} при заданных n и t, $n = a(L + t)\, n \bmod (L + t)$ и $n \geqslant (t + 1)(L + t)$ глубина диагностирования по крайней мере до одного элемента возможна, если количество B элементов, не входящих в СВК, удовлетворяет условию

$$B \leqslant aL + c - 1 \text{ при } c < L$$
$$\text{или } B \leqslant aL + L - 1 \text{ при } L \leqslant c \leqslant L + t - 1,$$

где $c = n \bmod (L + t)$.

Поскольку $a \leqslant n/(L + t)$, то $B \leqslant aL + L - 1 \leqslant nL/(L+t) + L - 1$.

Сравнивая это значение со значением B для структур без восстановления, получаем, что для структур с восстановлением верхняя оценка значения B увеличивается в K_B такое число раз:

$$K_B \approx [L(nL + L^2 - L + Lt - t)]/(L^2 - t^2)n > L + t + 1.$$

Таким образом, при использовании структур с восстановлением удается значительно увеличить количество элементов, используемых для выполнения системой основных функций.

До сих пор рассматривались СВК, обеспечивающие возможность определения неисправных элементов. Однако для диагностирования и работы системы в целом не менее важное значение имеет вопрос определения исправных элементов, которые можно включить в состав элементов, выполняющих основные функции. Постановка задачи поиска таких элементов является корректной лишь в том случае, когда наибольшее количество неисправных элементов не превосходит t, а исправные элементы находятся из условия минимальности количества неисправных элементов.

Минимальная длина n_c цепочки при условии определения хотя бы одного исправного элемента

$$n_c \geqslant \left(\frac{t+3}{2}\right)^2 - 1 \quad t \text{ — нечетное;}$$

$$n_c \geqslant \frac{t}{2}\left(\frac{t}{2} + 3\right) + 1 \quad t \text{ — четное.}$$

В частном случае кольцевой структуры D_{1L} количество элементов, выполняющих основные функции,

$$B = n - \left\lfloor \left(\frac{t+3}{2}\right)^2 \right\rfloor + 1.$$

В структурах с восстановлением для системы D_{1L} при обнаружении по крайней мере одного исправного элемента при $L = \lfloor t/2 \rfloor + 1$ наибольшее число элементов, выполняющих основные функции,

$$B = n - (2t + \lfloor t/2 \rfloor + 1),$$

а при $L = t$ это число $B = n - (2t + 1)$, что соответствует включению в СВК цепочек длиной $2t + \lfloor t/2 \rfloor + 1$ и $2t + 1$ соответственно.

При построении перестраиваемых систем возникает вопрос об оптимизации состава связей в системе. Определим оптимальность как выбор для реализации основных функций системы B элементов, $0 \leqslant B \leqslant n - 3$, при условии сохранения системой свойства t-диагностируемости без восстановления для оставшихся $n - B$ элементов.

Оптимальным в этом смысле являются системы D_{1L} при $L + 1 \leqslant n \leqslant 2L + 1$, а также системы, синтезируемые в следующем порядке.

Устанавливаем связи между элементами v_i, v_j в соответствии со следующим правилом:

$$\exists (v_i, v_j) \leftrightarrow j = (i + \alpha + 1) \bmod n$$

или

$$j = (i + \lfloor (n+1)/2 \rfloor - \alpha - 1) \bmod n, \quad 1 \leqslant i \leqslant n,$$
$$\alpha = 0, 1, \ldots, t-1.$$

Множество элементов с индексами $v_1, \ldots, v_{t+1}$ и $v_{\lfloor (n+1)/2 \rfloor + 1}, \ldots, v_{\lfloor n/2 \rfloor + t + 2}$ относим к элементам СВК, а оставшиеся элементы относим к вычислительным элементам. Полученная таким образом структура является оптимальной в указанном смысле и обладает наиболее высокими диагностическими возможностями.

Рассматривая вопрос оптимальности в более широком смысле, следует отметить следующее.

При построении структур отказоустойчивых систем перестройка определяется конкретными аппаратурными и программными ресурсами системы, обеспечивающими гибкое перераспределение функций системы. Такое перераспределение приводит к некоторой избыточности связей элементов отдельных подсистем или требует хранения большого количества информации о состоянии системы, полученной на предыдущих стадиях решения задач. Это, в свою очередь, ухудшает основные функциональные характеристики. В частности, снижается производительность (быстродействие) вычислений в системе.

Наконец, перестройка усложняет, как правило, реализацию интерфейсов, коммутационных и согласующих устройств. Это обусловлено необходимостью выбирать в широких пределах подмножества элементов и управлять системой одновременно при диагностировании и выполнении требуемых функций. Полный учет этих противоречивых требований с целью построения наиболее эффективных структур перестройки, замены и перераспределения функций возможен лишь путем получения и анализа обобщенных характеристик функционирования системы. Получение таких характеристик представляет собой самостоятельную задачу, решаемую в тесной связи с возможными аппаратурными реализациями отдельных частей системы (интерфейсов, вычислительных элементов и контрольно-диагностической аппаратуры и др.), а также с учетом основных целей обеспечения отказоустойчивости системы.

Практически в ряде случаев удается значительно упростить СВК за счет обработки конкретных множеств синдромов на протяжении достаточно длительного времени эксплуатации. Получение таких упрощенных структур можно решить на основе рассмотренных ранее алгоритмов поиска неисправных элементов. Вследствие этого повышается разрешающая способность и достоверность диагностирования при одновременном упрощении структур перестраиваемых систем.

4. АВТОМАТИЗАЦИЯ ПОСТРОЕНИЯ СТРУКТУР

Необходимость построения СВК с заданными свойствами возникает как в процессе создания отказоустойчивой системы, так и во время ее эксплуатации. Построение СВК при большом количестве элементов системы и допустимом числе неисправных элементов становится трудоемкой задачей.

При создании системы решаются задачи определения минимального состава аппаратурных связей между элементами, обеспечивающего длительное функционирование. При этом должна быть предусмотрена определенная избыточность этих связей, допускающая сохранение системой работоспособности при наличии критических ситуаций.

Построение СВК в процессе эксплуатации является одним из основных средств обеспечения процессов контроля и диагностирования с приемлемыми аппаратурными и временными затратами на их реализацию. Обычно организация взаимодействия между элементами производится по ограниченному количеству физических связей (шин связей) с учетом определенной приоритетности участвующих во взаимодействии элементов (выделение контролируемых и контролирующих элементов). Расширение требуемого количества информационных связей взаимоконтроля производится в этом случае за счет увеличения количества шагов (тактов, циклов) взаимодействия элементов. При этом следует иметь в виду, что в одном таком цикле проверки каждому контролирующему элементу могут соответствовать несколько контролируемых.

В некоторых из существующих систем количество информационных связей между отдельными элементами фиксировано и соответствует числу физических связей. В этом случае наибольшее количество неисправных элементов, которые можно обнаружить, имеет, при исправности аппаратурных линий связи, фиксированное значение.

Автоматизация построения структур взаимоконтроля с учетом отмеченных и других ограничений позволяет существенно сократить время при создании и перестройке структуры отказоустойчивой системы, а также уменьшить затраты, свя-

занные с простоем (неиспользованием) ресурсов системы в процессе функционирования.

В процессе построения структур отказоустойчивых систем возможно большое количество вариантов структур, удовлетворяющих заданным требованиям диагностируемости. Кроме того, вследствие неоднозначности получаемых данных исправности элементов системы возможно большое количество допустимых наборов состояний системы, удовлетворяющих фиксированным синдромам. Поэтому необходимо производить перебор большого количества вариантов структур связей и оценивать их технические характеристики с целью выбора наилучшего. Решение этой задачи возможно за счет автоматизации получения и анализа отдельных технических решений структур систем.

К основным этапам построения структур взаимоконтроля при автоматизации относятся:

1) формирование комплекса требований к структуре отказоустойчивой системы (наибольшее количество неисправных элементов, количество элементов структуры, ограничения и требования по организации связей взаимодействия между элементами);

2) определение базового состава связей взаимодействия элементов;

3) выбор типа интерфейса между элементами в процессе взаимодействия, определение системы прерываний;

4) проверка условий соответствия базового набора задаваемым требованиям;

5) получение технического варианта структуры взаимоконтроля;

6) моделирование структур связей по критическим состояниям в системе (отказ физических линий связи, возникновение сбоев, наличие гонок и информационных перегрузок при обмене данными между элементами и др.);

7) разработка мер по устранению выявленных несоответствий поставленным (заданным) требованиям;

8) моделирование характеристик обнаружения и локализации неисправных элементов. Моделирование проводится по множеству характерных, случайных или произвольных, из числа заданных, синдромов с оценкой временных показателей обнаружения и локализации неисправностей;

9) задание требований на программное обеспечение. Моделирование систем команд отказоустойчивой системы и отработка алгоритмов (условных и безусловных) диагностирования и восстановления системы.

Каждый из этих этапов разбивается на определенное число подэтапов и может выполняться как одновременно, так и по-

следовательно при получении определенных решений на предшествующих этапах. Помимо функций взаимоконтроля в процессе построения структур могут учитываться имеющиеся средства контроля и диагностирования: защита по недопустимым комбинациям, повышение надежности путем повторных просчетов по нескольким независимым алгоритмам. В процессе моделирования можно учитывать функциональные особенности выбираемых элементов (микро-ЭВМ, микропроцессоров). Необходимо отметить также, что выполнение каждого этапа сопровождается числовым просчетом количественных показателей (критериев), позволяющих оценить качество структуры и системы в целом.

Основой для автоматизации проектирования структур отказоустойчивых систем является формализация описания структур. При такой формализации должны учитываться параметры системы, необходимые для выполнения всех или отдельных из перечисленных этапов. Кроме того, должна быть обеспечена возможность машинной обработки данных о структуре системы на основе использования унифицированных алгоритмов.

Определим с помощью матрицы инцидентности S-ю структуру физических связей в системе следующим образом. Элемент равен единице в том случае, если существует физическая связь, и нулю — в противоположном случае, т. е.

$$a^S_{i,\,j} = \begin{cases} 1 & \exists\,(v_i,\ v_j),\ v_i,\ v_j \in V; \\ 0 & \overline{\exists}\,(v_i,\ v_j),\ v_i,\ v_j \in V. \end{cases}$$

Назовем такую матрицу FL-матрицей и обозначим ее через A.

Так, например, в случае наличия в системе l независимых групп двунаправленных шин для интерфейса «общая шина» получаем матрицу инциденций вида (положим $S = 1$, $l = 2$)

$$A^1 = \begin{vmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{vmatrix}.$$

В данном случае две группы, каждая из которых содержит по три элемента, не связаны между собой. Каждый квадрат (с нулевой диагональю) соответствует группе элементов, соединенных между собой общей линией связи.

Если между l группами элементов имеется связь через один из элементов группы, то матрица имеет вид

$$A^1 = \begin{vmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{vmatrix}.$$

В этом случае третий элемент первой группы связан с первым элементом второй группы.

Для задания информационно-логических связей СВК для каждого варианта z определим матрицу инциденций B

$$b^z_{i,j} = \begin{cases} 1 & \exists I(v_i, v_j),\ v_i,\ v_j \in V; \\ 0 & \overline{\exists} I(v_i, v_j),\ v_i,\ v_j \in V. \end{cases}$$

Здесь $I(v_i, v_j)$ обозначает информационно-логическую связь контроля элемента v_j с элементом v_i. Так, например, в случае интерфейса «общая шина», если в структуре предполагаются двунаправленные связи взаимоконтроля, то матрица B^1 (положим $z = 1$) совпадает с матрицей A^1, т. е. $B^1 = A^1$.

Таким образом, условиям диагностируемости отвечает матрица B^z, а требованию устойчивых физических связей — матрица A^S.

В достаточно общем случае можно предположить, что в каждый момент времени (такт контроля) каждый контролирующий элемент проверяет один контролируемый. Выбор пар элементов, удовлетворяющих этим условиям, определяется конкретной стратегией (алгоритмом) контроля и диагностирования. Обозначим матрицу инцидентности, соответствующую шагу контроля k для алгоритма проверки r, через $C_k(r)$, элемент $C^{k,r}_{i,j}$ которой удовлетворяет следующему условию:

$$C^{k,r}_{i,j} = \begin{cases} 1, & \text{если элемент } v_i \text{ в } k\text{-м шаге при выполнении} \\ & \text{алгоритма } r \text{ контролирует элемент } v_j,\ v_i,\ v_j \in V; \\ 0 & \text{— в противном случае.} \end{cases}$$

Условиями совместимости проверки элементов являются:

$$\forall_j : \sum_i C^{k,r}_{i,j} = 1;$$

$$\forall_i : \sum_j C^{k,r}_{i,j} = 1.$$

Организация логической связи взаимоконтроля обеспечивается за счет наличия физической связи между элементами. Поэтому для произвольного алгоритма (программы) контроля должно выполняться соотношение:

$$\bigvee_k C_k(r) \leqslant A^S, \ \forall S; \ S = 1, 2, \ldots$$

Неравенство здесь соответствует условию

$$\bigvee_k C_{i,j}(r) \leqslant a^S_{i,j}, \ \forall_{i,j} = 1, \ldots, n, \ i \neq j.$$

В более общем случае, если в процессе функционирования системы предполагается перестройка структуры, данное соотношение для некоторых структур A^S может не выполняться. Однако, объединяя структуры, участвующие в перестройке: $A^1 = \bigvee_S A^S$, также придем к искомому выражению.

Неисправности, имеющие место в системе, при безусловных программах диагностирования обнаруживаются при получении синдрома для всех информационных связей СВК. Это означает, что при выполнении всех шагов контроля алгоритма r для фиксированной структуры

$$\bigvee_k C_k(r) \geqslant b^z,$$

т. е.

$$\bigvee_k C_{i,j}(r) \geqslant b^z_{i,j}.$$

Учитывая, что обычно отдельные шаги программы безызбыточны, т. е. служат лишь для получения новых составляющих синдрома, получаем

$$\bigwedge_k C_k(r) = 0.$$

Необходимо отметить, что для определенной физической структуры S и структуры взаимоконтроля z должно выполняться условие

$$A^S \geqslant B^z.$$

В противном случае выбирается другая структура B^z или A^S. В частности, можно выбрать z_1 такое, что

$$B^{z_1} \leqslant B^z.$$

Чтобы выполнялось условие $A^S \geqslant B^{z_1}$, можно также расширить состав связей структуры A^S до выполнения соотношения

$$A^{S_1} \geqslant B^z.$$

Таким образом, представленные соотношения позволяют достаточно полно оценить совместимость различных вариантов

отказоустойчивых структур с учетом допустимых вариантов структур связей.

В процессе функционирования отказоустойчивой системы целесообразно включение минимального количества элементов в число проверяемых, что позволяет обеспечить высокую производительность системы. Вместе с тем диагностические свойства структур существенно зависят от количества элементов, контролирующих каждый из элементов.

Например, в случае Р-моделей при диагностировании без восстановления минимальное количество связей при t-диагностируемости равно tn. Поэтому при одном и том же количестве связей N СВК, при различных значениях t и n, справедливо соотношение

$$t_i n_i \approx N \approx t_j n_j, \text{ или приближенно, } t_i/t_j = n_j/n_i.$$

Следовательно, t-диагностируемость обратно пропорциональна количеству проверяемых элементов. При этом $t_k \leqslant \lfloor (n_k - 1)/2 \rfloor$, $k = i, j$.

Таким образом, при автоматическом проектировании структур (при создании или перестройке структур) необходимо оценить кратность отказов t, в соответствии с которой подбирается состав элементов. Наименьшее число определяется исходя из предельно допустимой для данной модели кратности отказов. В том случае, когда ограничения по количеству связей не являются жесткими, можно сокращать количество проверяемых элементов за счет увеличения насыщенности связей в структуре.

Рассмотрим упрощенный пример обобщенного алгоритма автоматического построения структур для Р-моделей контроля элементов в отказоустойчивой системе.

1. Определить структуры физических связей с помощью матриц $A^1, \ldots, A^S$.

2. С учетом оценки значения кратности отказов t и заданного количества N информационных связей между элементами определить минимальное количество элементов структуры n.

3. Получить структуры информационных связей, используя, например, способ построения D_{nt}-структур. Преобразовать полученные структуры в матрицы инциденций $B^1, \ldots, B^z$.

4. Провести оценку совместимости структур $B^1, \ldots, B^z$ для заданных вариантов решений структур физических связей $A^1, \ldots, A^S$.

5. Выделить набор шагов проверки системы путем задания матриц $C_k(r)$.

6. Провести моделирование исключения физических связей в системе, выполняя для каждого случая проверку с учетом п. 4.

7. Провести моделирование обнаружения и локализации неисправностей в системе, используя результаты выполнения п. 5.

8. Провести корректировку структур A, B, C с целью удовлетворения задаваемых требований.

Необходимо отметить, что при моделировании по п. 7 могут быть использованы результаты гл. 3 по перечислению допустимых наборов неисправностей и их количественных оценок. Реализация обобщенного алгоритма построения структур, особенно при эксплуатации, должна сопровождаться анализом способов его выполнения на элементах системы.

Глава 5

ЭФФЕКТИВНОСТЬ ОТКАЗОУСТОЙЧИВЫХ СИСТЕМ

1. ПОКАЗАТЕЛИ КАЧЕСТВА СИСТЕМ

В процессе создания и эксплуатации радиоэлектронных систем возникает необходимость в оценке их функциональных и эксплуатационных характеристик. На ранних стадиях разработки системы такие оценки позволяют определить научно-технический уровень и новизну системы, ее конкурентоспособность и перспективность. При использовании системы на основе анализа опытных образцов получают оценки, которые могут уточняться и дополняться и служат для решения вопроса о количестве выпускаемых систем, разработки их модификаций с учетом условий эксплуатации, совершенствования функциональных характеристик и др.

Основой для оценки качественных характеристик радиоэлектронных систем служат количественные показатели или критерии качества системы. Показатель количественно оценивает характеристику одного или нескольких свойств системы. К основным качественным характеристикам относятся надежность, эффективность и достоверность информации.

Под надежностью понимается свойство объекта выполнять заданные функции, сохраняя во времени значения установленных эксплуатационных показателей в заданных пределах, соответствующих заданным режимам и условиям использования, технического обслуживания, ремонтов, хранения и транспортирования. Надежность включает безотказность, долговечность, ремонтопригодность и другие свойства. В число экс-

плуатационных показателей входят показатели производительности, наработки на отказ, вероятности безотказной работы, средний срок службы, коэффициент готовности и т. п.

Под эффективностью системы часто понимают степень ее приспособленности к выполнению стоящей перед ней задачи. Например, эффективность функционирования системы, позволяющей оценить ухудшение (улучшение) качества функционирования. В этом случае показатель эффективности определяется значениями показателей технических, надежностных, эксплуатационных характеристик и условиями, в которых система работает [21].

Различие между надежностью и эффективностью состоит в том, что надежность оценивает способность системы к поддержанию заданных параметров в допустимых границах, а эффективность позволяет оценить качество выполнения поставленной задачи. В частности, показатели эффективности могут быть как стоимостными (например, стоимость экономического эффекта, ущерба), так и комплексными, включающими вероятность выполнения требуемых функций и возможные потери (эффект) от использования.

Достоверность информации позволяет оценить соответствие принятого сообщения переданному. В вычислительных системах достоверность оценивает соответствие полученных результатов вычислений (с учетом ошибок исходных данных и процесса вычисления) истинным, полученным при отсутствии ошибок. К числу таких ошибок относятся ошибки в исходных данных, отказы и сбои в аппаратуре, ограниченная точность вычислений, ошибки в программах, использование системы в непредусмотренных режимах.

Особенностью оценки достоверности с помощью показателей достоверности является то, что в первую очередь учитывается характер влияния возникающих ошибок, а не причины их возникновения. Высокая достоверность предполагает, как правило, надежность работы системы. Однако большое влияние на показатели достоверности оказывает качество исходных данных и алгоритмов функционирования, что не столь существенно в показателях надежности и эффективности.

Выбор показателей надежности, эффективности и достоверности, а также точность их вычисления существенно зависят от стадии разработки и эксплуатации системы. Так, например, перед началом проектирования системы имеются очень неточные данные о возможных отказах системы и распределении вероятностей появления их, достоверности исходных данных, характере ошибок в программном обеспечении. Поэтому в практике на этом этапе обычно используют данные эксплуатации аналогичных систем, ориентировочные оценки на-

дежности системы исходя из ее структурно-функциональных параметров, результатов моделирования поведения системы при возникновении отказов. На последующих стадиях уточняются основные функциональные характеристики и показатели качества системы. В частности, по результатам рабочего проектирования и опытной эксплуатации системы достаточно точно можно установить производительность системы, время наработки на отказ, время восстановления, характеристики достоверности исходных данных и т. п.

В табл. 3 представлены основные этапы развития первого образца системы.

Таблица 3

Стадия	Разработка					Изготовление опытной партии или установочной серии		Эксплуатация		
Этап	ТЗ	ТП	РП	ОЭ	КД	ПП	Выпуск первой системы	Освоение	Эффективное применение	Техническое старение
Срок, годы	3—5					1—2	0,5—1,5	0,5—1	5—8	2—10

Упреждающие технические решения

Технические усовершенствования и модернизация

Перспективные решения

Упреждающие технические решения новой системы

Примечание. ТЗ — разработка технических предложений и технического задания; ТП — техиическое проектирование; РП — рабочее проектироваине и изготовление опытиого образца; ОЭ — опытная эксплуатация; КД — корректировка документации, подготовка документации для изготовления опытной партии или установочной серии; ПП — подготовка производства.

На этапах изготовления удается улучшить показатели качества системы за счет более совершенной технологии, качества используемых материалов по сравнению с опытным образцом.

На этапе эксплуатации получают наиболее достоверные эмпирические данные функционирования системы в реальных условиях, что обеспечивает уточнение показателей надежности, эффективности и достоверности. Доминирующими здесь являются эксплуатационные показатели, в частности, показатели обслуживания системы. Полученные по ним оценки по-

вволяют принять решение о последующих модификациях системы или нецелесообразности ее дальнейшего развития ввиду бесперспективности.

На этапах технического задания и технического проектирования оценка качества системы производится на основе априорных оценок составных компонентов системы или аналогичных существующих систем. Числовые значения этих показателей позволяют обосновать целесообразность создания данной системы и эффективность принятия технических решений. Последнее уточняется также после разработки принципиальной схемы, сопровождающей документацию (техническое описание, инструкции по эксплуатации, технические условия).

Отработка системы на этапе опытной эксплуатации позволяет достаточно полно оценить основные показатели надежности, эффективности и достоверности системы.

Таким образом, наряду с разработкой показателей на разных этапах должна вестись работа по уточнению значений основных технических параметров функционирования системы. Это позволяет совершенствовать систему в правильном направлении.

Сбор и обработка данных для уточнения показателей системы оговариваются уже на начальных подэтапах создания системы (в техническом задании, в частности). Так, в качестве дополнительной информации можно использовать результаты анализа надежности при проектировании; предыдущих испытаний системы и ее составных частей; испытаний систем-аналогов и систем, имеющих аналогичные составные части; предыдущих эксплуатационных наблюдений за системами или их составными частями.

При получении исходных параметров обычно используются априорные оценки функционирования системы или ее составных частей. Уточняемые параметры получают на основе данных испытаний или опытной эксплуатации системы или ее элементов. При этом широко используются байесовские методы оценки. В частности, при отсутствии априорных данных можно использовать байесовский подход к эмпирическим данным работы системы, полученным в процессе ее эксплуатации.

Необходимо отметить, что помимо недостоверности, обусловленной искажениями и ошибками в системе, имеются недостоверности оценки самих показателей качества работы системы, влияние которых наиболее существенно на первых этапах создания системы (этапы технического задания и технического проектирования) и снижается на последующих (период освоения, период эффективного применения). На этапе технического старения точность оценок вновь ухудшается.

Поэтому данные по значениям показателей, полученным на этапе эффективного применения системы, служат для определения технического уровня развития данного класса систем.

Рассмотренный процесс создания системы позволяет говорить о необходимости многостадийного выбора и уточнения показателей качества системы. Большое значение при этом имеет выделение доминирующих показателей и меры по их уточнению на каждом этапе.

В табл. 4 приведены некоторые характеристики оценки качества отказоустойчивых систем на различных этапах ее развития. Обеспечение отказоустойчивости систем сказывается на особенностях оценки качества. К таким особенностям относятся необходимость учета постепенного ухудшения функциональных характеристик (деградации) системы, влияние количества выявленных неисправностей на производительность системы и др.

Таблица 4

Стадия	Исходные параметры	Уточненные параметры	Показатели качества
Разработка	Интенсивность отказов элементов системы, время диагностирования и восстановления, характеристика структуры системы, режимов работы, стоимость разработки, наибольшее количество неисправных элементов, число ЭМ системы	Структурная надежность отказов системы, аварийных ситуаций, наиболее важные требования к системе	Потенциальная производительность и показатели надежности системы, ремонтопригодности, коэффициенты унификации, стандартизации, взаимозаменяемости и др.
Изготовление	Надежность связей между элементами конструкции, допустимых условий эксплуатации и обслуживания	Структура системы, режимы работы, стоимость разработки	Ремонтопригодность, энергопотребление, достоверность отработки алгоритмов
Эксплуатация	Интенсивность восстановления, время наработки на отказ, законы распределения отказов в системе	Искажения информации в системе, время аварийных отказов, стоимость разработки модификаций	Коэффициенты готовности, вероятность и среднее время восстановления, экономическая эффективность, производительность. Оценка показателей надежности и эффективности перспективных систем

Показатели используемые при оценке надежности радиоэлектронных систем и их компонентов, принято разделять на следующие группы:

1) показатели безотказности, включающие вероятность безотказной работы, интенсивность отказов для невосстанавливаемых систем и параметр потока отказов и наработки на отказ для восстановления систем;

2) показатели долговечности, оценивающие ресурс наработки на отказ в системе;

3) показатели ремонтопригодности, включающие вероятность восстановления в заданное время и среднее время восстановления;

4) показатели сохраняемости;

5) комплексные показатели надежности, в частности коэффициенты готовности, технического использования и оперативной готовности, а также показатели трудоемкости и стоимости ремонтов и технического обслуживания в процессе эксплуатации системы.

При проведении расчетов по показателям надежности с целью упрощения обычно используют допущение о пуассоновском законе распределения вероятности отказов компонентов системы. Это предположение справедливо при следующих условиях:

1) вероятность перехода системы, имеющей n отказов, в состояние с $n + 1$ отказами за достаточно малый интервал времени равна $\lambda \Delta t$;

2) отказы в системе происходят независимо;

3) вероятность возникновения более одного отказа элемента за достаточно малый промежуток времени Δt пренебрежимо мала.

При использовании таких предположений общая вероятность безотказной работы элемента за время $t > 0$ при условии, что в момент $t = 0$ система исправна, определяется как $P = e^{-\lambda t}$, а общая вероятность функционирования системы — как

$$P = \sum_{S_i \subseteq V} \prod_{i \in S_i} P_i (1 - P_i),$$

где P_i — вероятность безотказной работы элементов подмножества S_i системы; S_i — подмножество элементов, при исправности которых система является работоспособной.

В общем случае допустимыми являются все 2^n состояний элементов системы, а вероятности отсутствия отказов отдельных элементов $P_э = e^{-\lambda t}$ равны между собой.

Если предположить, что исправность системы обеспечивается при исправности, по крайней мере, k элементов, система является исправной для всех тех S_i, для которых $|S_i| \geqslant k$. В этом случае общая вероятность безотказной работы системы

$$P = \sum_{i=0}^{n-k} C_n^i P_э^{n-i} (1 - P_э)^i.$$

Как показывает опыт, значение интенсивности отказов компонентов системы, особенно в микроэлектронном исполнении, существенно зависит от внешних влияющих факторов (температура, атмосферное давление, степень облучения и т. п.), а также технологии изготовления. Это влияние учитывается введением специальных коэффициентов. Например, интенсивность отказов полупроводниковых кристаллов [27] оценивается с помощью следующей формулы:

$$\lambda = \pi_L \pi_Q (C_1 \pi_T + C_2 \pi_E),$$

где π_L — показатель, учитывающий особенности технологического процесса, значение его колеблется от 1 до 10; π_Q — коэффициент качества, определяемый входной отбраковкой и меняющийся от 1 до 150; π_T — параметр, зависящий от внешних температурных условий работы и от типа полупроводника, значение его меняется от 0,1 до 1000; π_E — коэффициент, характеризующий остальные внешние условия работы, значение его колеблется в пределах от 0,2 до 10; C_1, C_2 — коэффициенты сложности, определяемые числом вентилей в кристалле и числом разрядов (памятью) одного компонента.

Интенсивность отказов интегральных схем [31]

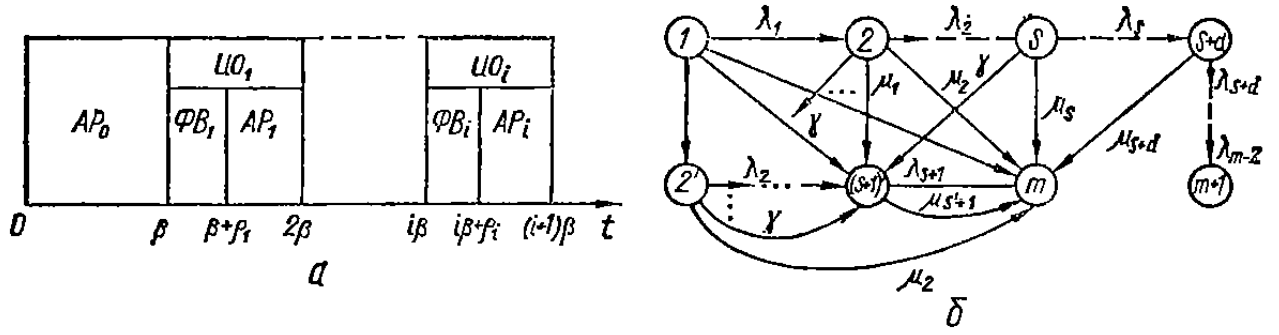
$$\lambda = (\lambda_A A + \lambda_F \Pi_E) \Pi_T \Pi_Q + (\lambda_B B + \lambda_C A) \Pi_T \Pi_Q \Pi_E + \\ + (\lambda_D C + \lambda_E \Pi_E D + \lambda_G E + \lambda_н E) \Pi_Q,$$

где λ_A, λ_B, λ_C, λ_D, λ_E, λ_F, λ_G, λ_H — интенсивности отказов, обусловленных дефектами соответственно оксидного слоя, сварных соединений с кристаллом, металлизации, диффузии, инородных включений, некачественного крепления кристалла, поверхности кристалла, структуры кристалла; A, B, C, D, E — соответственно площадь металлизации, количество сварных соединений и ступеней диффузии, отношение площади активных элементов и кристалла к 0,645; Π_E, Π_Q, Π_T — коэффициенты, учитывающие внешние факторы (температура, отбраковка и др.).

Применение таких оценок интенсивности отказов элементов позволяет в значительной степени приблизить получаемые априорные значения показателей надежности к реальным.

Многомашинную отказоустойчивую систему можно рассматривать как систему со скользящим резервированием и постепенной деградацией функциональных возможностей при отказе элементов. Функционирование ее состоит в последовательной смене фаз работы по основному назначению и обслуживанию с целью полного восстановления работоспособности. Рассмотрим пример анализа надежности таких систем [49, 50].

Процесс функционирования системы разделяется на последовательно выполняемые фазы восстановления (ФВ) и фазы автономной работы (АР) (рис. 20, *а*). В системе возможно на-



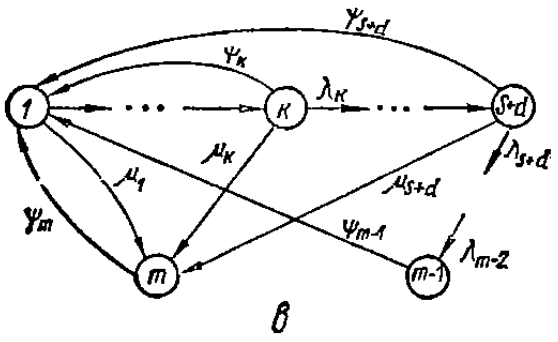


Рис. 20. Фазы работы (*а*) и графы фаз автономной работы (*б*) и восстановления (*в*) системы

личие d уровней деградации и S запасных элементов, подключаемых взамен вышедших из строя. В фазе автономной работы в начальный момент $t = 0$ система работает с полным ресурсом. Последующие интервалы обслуживания (ИО) длительностью β включают случайные интервалы восстановления длительностью ρ_i и интервалы автономной работы длительностью $\beta - \rho_i$, $i = 1, 2 \ldots$ В процессе автономной работы в системе возможны отказы элементов. Обнаруженные неработоспособные элементы заменяются работоспособными из числа резервных. При использовании всех элементов начинается деградация системы до уровня d.

Граф автономной работы (рис. 20,*б*) показывает, что система деградирует от состояния S до состояния $S + d$ (верхние вершины графа). Состояния системы, соответствующие нижним вершинам, описывают поведение системы при возникновении необнаруживаемых отказов (вершины $2'$, ..., $(S + 1)'$).

Фаза восстановления (рис. 20, *в*) характеризуется возможностью перехода системы в один из уровней деградации 1, ... d, (верхние вершины графа) или перехода в состояния первичного и вторичного безопасного отключения m и $m - 1$. Для полу-

чения числовых оценок интенсивностей переходов из состояния в состояние обозначим $Y(i)$ количество работоспособных элементов на уровне i деградации системы, $z(i)$ количество элементов системы, обеспечивающих решение системой поставленных задач. Очевидно, что $z(i) \leqslant Y(i)$. Например, в системе из трех элементов, выполняющих одни и те же функции системы, $Y(0) = 3$, но $z(0) = 1$. При отказе двух элементов $Y(1) = 1$, $z(1) = 1$, а при безопасном отключении $Y(2) = 0$ и $z(2) = 0$.

Символом $C(d)$ обозначим часть переходов в состояние безопасного отключения $S + d$. Ниже приведены используемые параметры системы.

λ — интенсивность отказа работающего элемента
μ — интенсивность отказа резервного элемента
γ — интенсивность невосстанавливаемого отказа элемента
α — часть восстанавливаемых элементов из числа резервных
β — интервал обслуживания системы
ρ_j — длительность j-й фазы восстановления
ψ_i — интенсивность восстановления i-го элемента
δ — среднее время запуска системы после аварийного отказа
φ — время проверки системы
λ_k — интенсивность отказа k-го работающего элемента
μ_k — интенсивность отказа k-го резервного элемента
m — число состояний безопасного отключения и катастрофического отказа системы
σ_j — j-я компонента вектора собственных значений матрицы переходов для фазы автономной работы
θ_j — j-я компонента вектора собственных значений матрицы интенсивностей переходов в системе для фазы восстановления

В результате при $k = \overline{1, m-1}$ для фазы автономной работы системы получаем следующие оценки [49]:

$$\lambda_k = W(k) CW(k) \lambda + X(k) \alpha\mu;$$
$$\mu_k = W(k)(1 - CW(k))\lambda;$$
$$\gamma = (1 - \alpha)\mu,$$

а для фазы восстановления

$$\lambda_k = W(k) CW(k) \lambda + X(k) \mu;$$
$$\mu_k = W(k)(1 - CW(k))\lambda;$$
$$\psi_k = \varphi/[N + S - (W(k) + X(k))],$$

где $$\psi_m = 1/\delta,$$

$$W(k) = \begin{cases} Y(j) & \text{при } k \leqslant S + d; \\ Y(d) + S + d + 1 - k & \text{при } k > S + d; \end{cases}$$

$$CW(k) = \begin{cases} C(0) & \text{при } k \leqslant S,\ S > 0; \\ C(j+1) & \text{при } S < k \leqslant S + d,\ j = \max(0, k - S, -1); \\ 1 & \text{при } k > S + d. \end{cases}$$

Таблица 5

Показатель производительности	Обозначение	Характеристика показателя	Оценка показателя
Интервальная надежность	$R_i(t)$	Вероятность функционирования системы в момент $i\beta+\rho_i \leqslant t \leqslant (i+1)\beta+\rho_{i+1}$ при условии, что она работоспособна в момент $t=i\beta+\rho_i$	$R_i(t)=\begin{cases} R^c(t)=\sum_j d_j l^{-\sigma_j t} & \text{при } 0 \leqslant t \leqslant \beta-\rho_i, \\ R^r(t)=\sum_j b_j l^{-\theta_j t} & \text{при } 0 \leqslant t \leqslant \rho_{i+1} \end{cases}$
Коэффициент готовности системы с определенной производительностью	CA_j	Произведение $z(j)$ на сумму всех вероятностей состояний, которые обеспечивают эту производительность, деленное на $z(0)$	$CA_j=\sum \pi_k z(j)$, π_k — усредненная вероятность нахождения системы в состоянии k в момент t при условии, что в момент $t=0$ она находилась в состоянии 1. Суммирование ведется по всем k, удовлетворяющим $j=\max(0,\ k-s,\ -1)$
Среднее время потерь времени вычисления из-за аварийных отказов	TF	Ожидаемое время неработоспособности системы на интервале обслуживания в результате возникновения аварийных отказов	$TF=\int_0^{\beta-\rho} P_m(t)\,dt+\int_0^{\rho} Q_m(t)\,dt$, $P_m(t)$, $Q_m(t)$ — вероятность нахождения в состоянии аварийного отказа при автономной работе и восстановлении в момент t
Ожидаемое количество отказов на интервале обслуживания	F_i	Число элементов, в которых произошли аварийные отказы на интервале i, при условии, что в момент $t=0$ работает $N+S$ элементов	$F_i=\sum_{k=0}^{N+S} kP_k(\beta)$, $P_k(\beta)$ — вероятность отказа k элементов в конце интервала β

На основе значений λ_k, μ_k, γ, ψ_k можно определить показатели качества функционирования системы. Описание некоторых из таких показателей приведено в табл. 5.

3. ЭФФЕКТИВНОСТЬ МНОГОМАШИННЫХ СИСТЕМ

Существует много способов оценки эффективности радиоэлектронных систем. Это связано прежде всего с тем, что само понятие эффективности является комплексным, включающим большое количество свойств и зависит от конкретного применения системы. Можно выделить два основных типа показателей эффективности: экономические и функциональные. Такое разделение довольно условно и больше связано с доминированием в показателях экономических и функциональных параметров системы.

В экономических показателях доминирующим является экономический эффект, получаемый при использовании рассматриваемой системы. Эти показатели позволяют вычислить реальную прибыль от использования системы в народном хозяйстве.

Функциональные показатели ориентированы больше на оценку потенциальных функциональных возможностей, заложенных в системе. Такие показатели наиболее ценны в тех случаях, когда прекращение работы системы недопустимо или невыполнение системой заданных функций приводит к безвозвратной потере вложенных средств.

При оценке экономической эффективности наиболее широко применяется многокритериальный подход, при котором эффективность оценивается набором критериев, отражающих определенные свойства исследуемой системы. Сам экономический эффект от применения системы рассматривается как уменьшение стоимости или увеличение объема производства продукции по сравнению с базовой системой (реально существующей или потенциально осуществимой) либо по фактическому экономическому выигрышу в результате использования данной системы в конкретных условиях.

Показатели эффективности оценивают повышение эффективности использования труда, основных и оборотных фондов, капитальных вложений материальных ресурсов, экономическую эффективность общественного производства. В показателях эффективности целесообразно учитывать уровень техники в области, где используются данные системы. Так, например, при оценке сравнительной эффективности данной системы по сравнению с базовой показатель эффективности $W_б$ определяется как разность:

$$W_б = Z_н - Z_б,$$

где $Z_н$, $Z_б$ — приведенные затраты соответственно на данную и базовую системы. Приведенные затраты Z определяют по формуле

$$Z = C + E_н K,$$

где С — себестоимость годового объема продукции, выпускаемой с применением системы; $E_н$ — нормативный коэффициент эффективности капитальных вложений; К — капитальные вложения в производственные фонды.

Основное преимущество применения отказоустойчивых систем состоит в увеличении объема выпускаемой продукции, повышении производительности труда и снижении затрат на обслуживание, что приводит к снижению себестоимости продукции.

При оценке общего экономического эффекта от применения данной системы определяется разность между вложенными средствами и полученным экономическим эффектом. Например, общий экономический эффект [19] определяется по формуле

$$W_о = D_г - E_о B,$$

где $D_г$ — годовой доход при применении многомашинной системы; $E_о$ — нормативный коэффициент окупаемости капитальных вложений; В — затраты на создание системы.

Годовой доход $D_г$, полученный в результате эксплуатации многомашинной отказоустойчивой системы, определяется, в основном, сокращением стоимости обслуживания системы и увеличением ее производительности. В общем виде его можно представить так:

$$D_г = D_г(S,\ n_р,\ n_п,\ m,\ n_з,\ T_о,\ T_п,\ V_в,\ C_о,\ T_р,\ Q_п,\ ПО),$$

где S — структура связей в системе; $n_р$ — количество элементов системы, выполняющих основные функции; $n_п$ — количество контролируемых элементов СВК; m — количество связей; $n_з$ — количество запасных элементов системы; $T_о$ — цикл технического обслуживания; $T_п$ — время простоя системы: $V_в$ — достоверность входной информации; $C_о$ — трудоемкость и стоимость обслуживания системы; $T_р$ — ремонтный цикл; $Q_п$ — объем обрабатываемой информации, выпускаемой продукции и т. п.; ПО — программное обеспечение системы (структура, быстродействие, помехозащищенность, наличие ошибок и т.д.).

Затраты В на создание системы включают расходы на разработку системы, отнесенные к числу выпущенных систем. Числовые значения могут на этапе эксплуатации определяться реальными доходами от применения системы, а на этапах разработки и изготовления — на основе анализа аналогичных систем, моделирования и др. Таким образом, для внедрения

эффективных отказоустойчивых систем должен проводиться анализ как по сравнительным, так и общим показателям эффективности. Анализ системы по функциональным показателям эффективности позволяет оценить такие рабочие характеристики системы, как производительность, необходимые параметры обслуживания (периодичность обслуживания, количество запасных элементов системы, ее допустимые простои), выполнение необходимых функций по времени и т. д.

Одно из основных преимуществ применения отказоустойчивых систем состоит в повышении производительности обработки информации. Повышение производительности является следствием сокращения простоев, обусловленных неисправностью аппаратуры, и уменьшения интервалов обработки информации по недостоверным данным. Производительность существенно зависит от алгоритмов обработки данных, режима обработки (однопрограммный, мультипрограммный), архитектуры системы и организации процедур обмена данными, контроля, диагностирования и восстановления и др. Поэтому в зависимости от выбранной модели числовые оценки производительности даже для одних и тех же систем могут существенно различаться между собой, а сама оценка представляет сложную проблему [6, 19].

Наиболее простыми и достаточно общими подходами при оценке производительности является использование моделей функционирования системы с применением конечных цепей Маркова. В последнее время широко применяются модели, в которых отказоустойчивые системы рассматриваются как системы с постепенной деградацией функциональных возможностей. В этом случае производительность системы определяется как способность ее выполнять некоторое множество заданий на резервируемой аппаратурной среде. Рассмотрим одну из таких моделей [50].

Пусть отказоустойчивая система состоит из n подсистем и характеризуется текущим вектором состояний $s = (a_1, a_2, \ldots, a_n)$ и начальным вектором состояния $s_0 = (N_1, N_2, \ldots, N_n)$. Здесь a_i, N_i, $1 \leqslant i \leqslant n$ обозначает соответственно количество элементов в начальный и текущий моменты работы системы: $0 \leqslant a_i \leqslant N_i$.

Система выполняет множество заданий $J_1, J_2, \ldots, J_m$, причем для выполнения задания J_i, $1 \leqslant i \leqslant m$ не обязательно использование всех подсистем. Предполагается также, что известны распределения вероятностей $H_i(t)$ выполнения заданий J_i. Вектор заданий для системы определяется как вектор $u = (b_1, b_2, \ldots, b_m)$, где $b_i = 1$, если выполняется задание J_i, и $b_i = 0$, если не выполняется.

Обозначив состояние системы в момент t в пространстве состояний S через s, $t \geqslant 0$, а через $P_s(t)$ — вероятность нахождения системы в момент t в состоянии s при условии, что начальным состоянием было состояние s_0 в момент $t = 0$, получим следующую систему дифференциальных уравнений:

$$\frac{dP_i(t)}{dt} = -\sum_{s' \neq s} P_{ss'} P_s(t) + \sum_{s'=s} P_{s's} P_{s'}(t),$$

где $P_{ss'}\,dt$ — вероятность перехода системы из состояния s в состояние s' в бесконечно малый интервал dt.

Если предположить, что вероятность восстановления подсистемы после отказа равна c_i, интенсивность отказа системы (переход в состояние $s = 0$) при отказе единственного элемента i-й подсистемы равна λ_i, а вероятность восстановления системы после реконфигурации равна C, то выражение вероятности [50]

$$P_s(t) = \sum_{s \leqslant s' \leqslant s_0} C_{ss'} \exp\left(-\sum_i a_i \lambda_i t\right),$$

где $s' = (a'_1, a'_2, a'_3, \ldots, a'_n)$.

Значения $C_{ss'}$ определяются следующим образом:

$$C_{s_0 s_0} = 1;$$

$$C_{ss'} = \left\{\prod_i d_i^{(a'_i - a_i)}(a_i)\right\} P_{ss'} \text{ при } s \neq 0;$$

$$C_{ss} = -\sum_{s \leqslant s' \leqslant s_0} C_{ss'} \text{ при } s \neq s_0;$$

$$C_{0s'} = \sum_{0 \leqslant s' \leqslant s} C_{s's} \text{ при } s \neq 0,$$

где $d_i^1(a) = -(ac_i + 1)C$; $d_i^j(a) = -\{(ac_1 + 1)Cd_i^{(j-1)}(a+1) + (1 - c_i)C\sum_{k=2}^{j} d_i^{(j-k)}(a+k)\}/j$.

Проводя аналогичные рассуждения, определяем вероятность $Q_u(t)$ выполнения заданий $J_1, \ldots, J_n$, определяемых вектором u в момент t при условии, что в начальный момент $t = 0$ $u(t=0) = U_0 = (1, 1, \ldots, 1)$;

$$Q_u(t) = \sum_{u \leqslant u' \leqslant u_0} C_{uu'} \exp\left\{-\sum_i a(u)_i \lambda_i t\right\},$$

где $C_{uu'} = \begin{cases} 1, & \text{если } S'(u') > S(u) \text{ и } |u' - u|_1 \text{ четно;} \\ -1, & \text{если } S(u) > S(u) \text{ и } |u' - u|_1 \text{ нечетно;} \\ 0, & \text{если } S'(u) = S(u) \text{ и } u' \neq u; \end{cases}$

$$C_{u_0 u_0} = 1 \text{ и } C_{uu} = \sum_{u < u' \leqslant u_0} C_{uu'} \text{ при } u = u_0.$$

При определении вероятности $Q_u(t)$ предполагается, что для каждого множества состояний

$$S(u) = \bigcup_{b_i=1} S(J_i),\ u = (b_1, \ldots, b_i, \ldots, b_m),$$

где $S(J_i) = \{s|$ система в состоянии s может выполнить задание $J_i\}$.

Используя конкретные значения $P_s(t)$ и $Q_u(t)$, можно определить вероятность успешного выполнения задания системой

$$R(t) = \sum_i \Pi_i(t) R_i(t),$$

где $R_i(t)$ — определяет вероятность успешного выполнения задания J_i, начиная с момента t; $\Pi_i(t)$ — вероятность появления задания J_i для выполнения в момент t.

Вероятность $R_i(t)$ можно определить как для отдельного задания J_i, так и для группы заданий вектора U.

В первом случае вероятность

$$R_i(t) = \int_0^\infty \sum_{s \in S(J_i)} P_s(t+x)\, d(H_i(x)),$$

во втором —

$$R_i(t) = \int_0^\infty \sum_{ub_i=1} Q_u(t+x)\, d(H_i(x)).$$

Таким образом, используя достаточно простые соотношения, можно оценить эффективность системы с учетом производительности на заданном временном интервале.

Глава 6

ЭКСПЛУАТАЦИЯ ОТКАЗОУСТОЙЧИВЫХ СИСТЕМ

1. КОНТРОЛЬ МИКРОПРОЦЕССОРОВ

Обеспечение отказоустойчивости в системе в реальных условиях эксплуатации достигается за счет сочетания различных способов контроля, диагностирования и восстановления элементов системы. К традиционным способам повышения надежности радиоэлектронных систем относятся:

1) отбор элементов, прошедших достаточно полную проверку на работоспособность;

2) приработка элементов, обеспечивающая выход из строя наименее надежных элементов;

3) выявление и устранение неисправных элементов в процессе использования и обслуживания системы.

Во всех этих способах главным является определение работоспособности отдельных элементов. Использование БИС и СБИС в качестве элементов системы приводит к значительному увеличению количества тестов, необходимых для проверки, реализация которых повышает многократно стоимость проверенных элементов по сравнению с непроверенными. Вместе с тем неполная проверка таких элементов усложняет последующее выявление неисправностей в системе.

Наиболее простым способом автономной проверки БИС и СБИС является задание одинаковых наборов входных сигналов на заведомо исправный и проверяемый элемент с последующим сравнением выходных сигналов.

Наиболее распространенными в практике модификациями этого способа являются потактный контроль состояния выходов микропроцессора и сигнатурный анализ состояния микропроцессора. При потактном контроле выходной информации микропроцессора используются специальные анализаторы логических состояний, которые фиксируют сигналы на выходных шинах микропроцессора в заданные такты времени. Контроль производится путем сравнения выходов исправного микропроцессора с контролируемым или путем анализа результатов выполнения микропроцессором определенных команд.

Визуализацию данных контроля можно производить в виде отображения таблиц состояний, квазивременных диаграмм и диаграмм (карт) состояний (рис. 21).

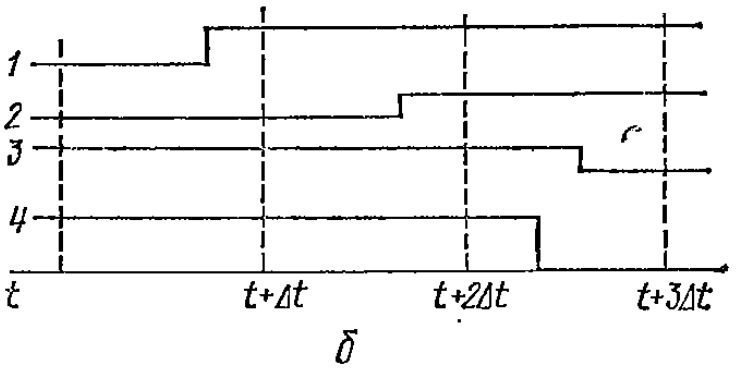
При отображении работы микропроцессора с помощью таблиц (рис. 21,*а*) или квазивременных диаграмм (рис. 21,*б*) в выбранные моменты времени t, $t + \Delta t$, ... индицируется состояние соответствующей шины *1*, *2*, *3*, *4*. При индикации с помощью карт состояний (рис. 21,*в*) для каждого допустимого состояния шины магистрали ставится точка (*1* — *16*) на экране устройства отображения. В результате достаточно просто визуально определяется характер отказа.

Контроль микропроцессоров на основе сигнатур основывается на «свертывании» последовательных выходных кодовых слов (например, суммированием в счетчике с обратной связью) за несколько тактов работы. Ошибки выявляются при сравнении значения сигнатуры контролируемого микропроцессора с сигнатурой исправного микропроцессора. Метод позволяет выявить бо́льшую часть ошибок.

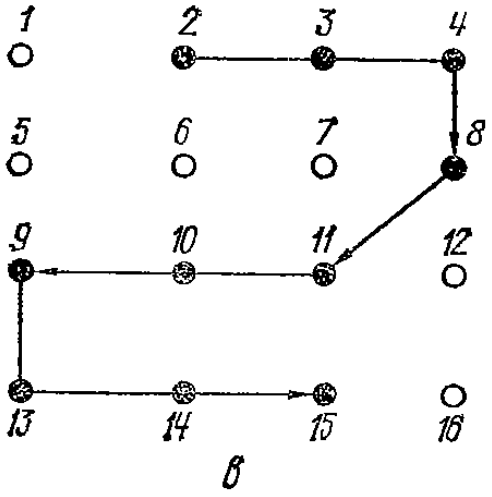
Полная проверка широко применяемых БИС требует порядка 100 тыс. и более тестов. Реализация этих тестов приводит при соблюдении режима проверки к повышению стоимости микропроцессора в десятки и более раз. Количество тестов и полнота проверки в значительной степени определяется выбором модели элемента системы. Так, например, удачно выбранная модель микропроцессора Intel 8080 позволяет сократить количество тестов с 10^6 до 12 500 [56]. Однако единственная выбранная модель не гарантирует качества проверки оконча-

	t	$t+\Delta t$	$t+2\Delta t$	$t+3\Delta t$
1	0	1	1	1
2	0	0	1	1
3	1	0	1	0
4	1	1	1	0

а



б



в

Рис. 21. Визуализация данных контроля

тельно. Поэтому при эксплуатации отказоустойчивых систем целесообразно применение различных моделей.

Модели БИС и СБИС для контроля и диагностирования можно условно разбить на следующие группы.

1. Модели, рассматривающие БИС и СБИС на уровне вентильных схем типа И, ИЛИ, НЕ, триггеров и т. п.

2. Модели, в которых элемент рассматривается как совокупность укрупненных функциональных узлов типа регистра, арифметико-логического устройства, мультиплексора и т. п.

3. Модели, в которых проверка элемента проводится по правильности выполнения им конкретных наборов операций или команд.

Модели первой группы наиболее громоздки и требуют сложных процедур проверки. Поэтому рассмотрим особенности моделей последних двух групп применительно к микропроцессорам.

Функционально структуру микропроцессора можно рассматривать как совокупность модулей (регистров, АЛУ, мультиплексоров и т. п.), между которыми устанавливаются связи в зависимости от выполняемой команды. Поэтому контроль работоспособности микропроцессора можно производить

путем задания внешних тестовых наборов при фиксированных внешних управляющих входах с учетом небольшого количества обратных связей. Обратная связь в микропроцессорах обычно сводится к передаче сигналов с выхода АЛУ на входные мультиплексоры (например, в микропроцессорах К580ИК80). Внешние тестовые наборы получают на основе тестовых наборов для отдельных модулей. Затем производится «продвижение назад» этих тестовых наборов до внешних входов и «продвижение вперед» до внешних выходов микропроцессора.

В общем случае все модули можно разбить на подмножества комбинационных схем (например, мультиплексоров, схем сдвига, АЛУ) и последовательных схем (например, регистров, аккумуляторов). Каждое подмножество модулей имеет общие управляющие входы. Обозначим через $M_{i,j}$ j-й модуль i-го подмножества модулей, через $I_{i,t,j}$ — t-й набор управляющих входных сигналов для модуля, $I_{i,t}$ — t-й набор управляющих входных сигналов для модуля i-го подмножества.

При фиксированном t-м наборе входных управляющих сигналов можно задавать отдельные тесты $T_{i,t,j,r}$, $r = 1, 2, \ldots, n_j$, позволяющие проверить отдельные функции модуля $M_{i,j}$. Наборы тестов для проверки подмножества i модулей $M_{i,j}$ при фиксированном t и управляющих входах $I_{i,t}$ обозначим

$$T_{i,t} = \underset{r,j}{U} T_{i,t,j,r}.$$

Наборы тестов $T_{i,t}$ представляют объединение тестов $T_{i,t,j,r}$ по всем возможным сочетаниям r и j.

Тесты $T_{i,t,j,r}$ получают путем задания для модуля $M_{i,j}$ информационных входных сигналов при фиксированном наборе входных управляющих сигналов. Входы модуля $M_{i,j}$ связаны с внешними информационными входами и выходами через последовательность модулей других подмножеств $M_{i,j}$. Поэтому для получения тестов $T^{B}_{i,t,j,r}$, приведенных к внешним контактам микропроцессора, производится «продвижение назад» теста $T_{i,t,j,r}$ от входа модуля $M_{i,j}$ к внешним входам микропроцессора и «продвижение вперед» от выходов этого модуля к выходам микропроцессора.

При наличии неисправностей в некоторых модулях могут возникнуть ситуации маскирования неисправностей, обнаружение которых составляет самостоятельную проблему.

Рассмотрим пример функционального диагностирования одноразрядного микропроцессора [56].

На рис. 22, *а* показана упрощенная схема одноразрядного секционированного микропроцессора. Управляющие входы I_0, I_1, ..., I_9 задают режимы работы узлов. На рис. 22,*б* — *д* пока-

заны функции, выполняемые этими узлами. Предполагается, что отказ может быть лишь в одном из них. Узлы M_L, M_R, M_S, M_F являются комбинационными, а M_A и M_T — последовательностными схемами.

В зависимости от управляющих сигналов для каждого из узлов может быть определено подмножество тестов. Так, например, при подаче нулевых сигналов на входы I_8, I_9 на выходе узла M_L получается сигнал F. Тестами для этого модуля является подмножество $T_{1,1}$ комбинаций состояний сигналов на входы RI, F, LI:

$$T_{1,1} = \{(I_8,\ I_9,\ RI,\ F,\ LI)\} = \{(0,\ 0,\ RI,\ F,\ LI)\}.$$

Для продвижения этих тестов вперед необходимо задать сигналы управления таким образом, чтобы выходные сигналы модуля M_L попали на выход F, по которому можно проводить контроль.

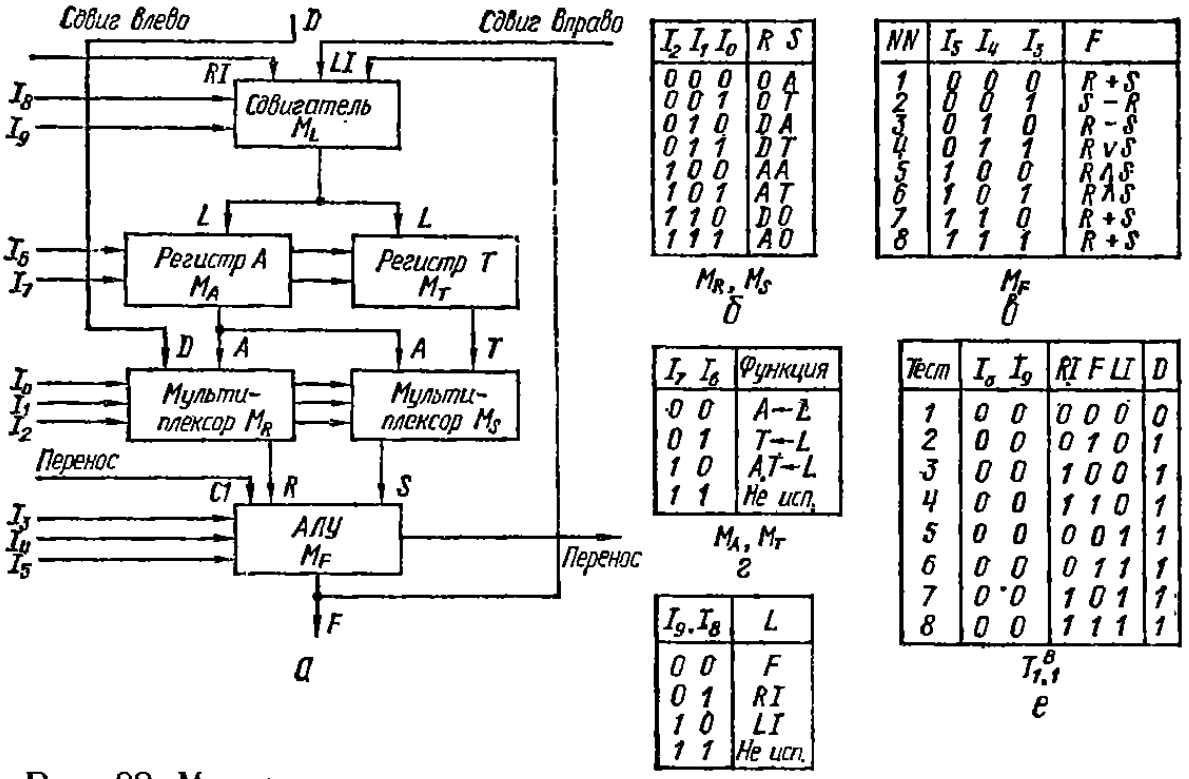


Рис. 22. Модель одноразрядного микропроцессора

В частности, задавая $I_{1,1} = \{I_t\}$, $I_2 = I_3 = I_0 = I_6 = I_7 = 0$, $I_4 = I_5 = I_1 = 1$, получаем подмножество $T^B_{1,1}$ тестов, показанных на рис. 22, *е*, при фиксированном наборе $I_8 = I_9 = 0$.

Аналогично получаем подмножество тестов при двух других комбинациях сигналов управляющих входов: I_8 и I_9. При проверке регистров M_A, M_T подмножества тестов определяют, рассматривая их как конечные автоматы.

Проводя объединение получаемых подмножеств тестов, находят полное множество тестов. В процессе такого объединения

сокращается количество тестов с учетом требуемой глубины диагностирования микропроцессора.

Рассмотренные модели контроля работоспособности микропроцессоров позволяют решать задачи контроля при наличии достаточно полных данных об архитектуре микропроцессора. Основная цель при использовании таких моделей состоит обычно в проверке всех функциональных компонент. При этом, вследствие вводимых упрощений, значительная часть неисправностей остается необнаруженной.

Однако в практике часто приходится проверять микропроцессоры при отсутствии полных данных о внутренней структуре, связях между отдельными его функциональными частями. Кроме того, в процессе функционирования во многих случаях оказывается невозможным специально переключать микропроцессор на контроль и диагностирование. Наиболее полно указанным ограничениям удовлетворяют модели третьей группы. Рассмотрим упрощенный пример, поясняющий диагностирование микропроцессоров при использовании моделей третьей группы.

```
FSM1    MACRO    ADC1
        LXI      H, ADC1
        MOV      A, M
        INX      H
        ADD      M
        MOV      M,A
        ENDM

FSM2    MACRO    ADC1,ADC2,R1,R2
        LXI      R1, ADC1
        LDAX     R1
        MOV      R2, A
        LXI      R1, ADC2
        LDAX     R1
        ADD      R2
        STAX     R1
        ENDM

FSM3    MACRO    ADC1, ADC2
        LHLD     ADC1
        XCHG
        LHLD     ADC2
        DAD      D
        SHLD     ADC2
        ENDM
```

Рис. 23. Макрокоманды сложения

Предположим, что необходимо проверить правильность выполнения операции суммирования над однобайтными числами. Операция может быть выполнена на функциональных узлах микропроцессора с использованием различных команд. Предположим, что в микропроцессоре и в памяти микрокоманд и управляющих схемах может быть лишь единичная неисправность. Выполнение операции при наличии неисправности приводит к неверному результату. Тогда многократное выполнение операций, различающихся между собой по используемым функциональным узлам и командам и приводящих к одинаковому результату, позволяют выявить неисправность. Использование принципов голосования при наличии нескольких одинаковых результатов обеспечивает получение правильного результата.

На рис. 23 показаны макрокоманды различных вариантов выполнения операции сложения для микропроцессора К580ИК80.

Макрокоманды FSM2, FSM3 допускают использование различных регистров. Введем обозначения команд, используемых в макрокомандах, в соответствии с табл. 6.

Таблица 6

Команда	LXI	MOV	INX	ADD	LDAX	DAD	STAX	LHLD	XCHG	SHLD
Обозначение	*P1*	*P2*	*P3*	*P4*	*P5*	*P6*	*P7*	*P8*	*P9*	*P10*

Тогда для макрорасширений перечисленных макрокоманд можно записать подмножество используемых команд и регистров:

$$\text{FSM1} = (P1,\ P2,\ P3,\ P4,\ A,\ H,\ \text{L})$$
$$\text{FSM2 (1)} = (P1,\ P2,\ P4,\ P5,\ P7,\ A,\ B,\ C,\ D)$$
$$\text{FSM2 (2)} = (P1,\ P2,\ P4,\ P5,\ P7,\ A,\ B,\ C,\ E)$$
$$\text{FSM2 (3)} = (P1,\ P2,\ P4,\ P5,\ P_7,\ A,\ B,\ D,\ E)$$
$$\text{FSM2 (4)} = (P1,\ P2,\ P4,\ P5,\ P7,\ A,\ C,\ D,\ E)$$
$$\text{FSM2 (5)} = (P1,\ P2,\ P5,\ P7,\ A,\ B,\ C,\ H)$$
$$\text{FSM2(6)} = (P1,\ P2,\ P4,\ P5,\ P7,\ A,\ B,\ C,\ \text{L})$$
$$\text{FSM2(7)} = (P1,\ P2,\ P4,\ P5,\ P7,\ A,\ B,\ E,\ H)$$
$$\text{FSM2(8)} = (P1,\ P2,\ P2,\ P5,\ P7,\ A,\ D,\ E,\ \text{L})$$
$$\text{FSM3} = (P6,\ P8,\ P9,\ P10,\ H,\ \text{L},\ D,\ E)$$

Для макрокоманды FSM2 в скобках указан номер возможной реализации.

Непосредственно из анализа различных реализаций операции сложения с применением макрокоманд FSM1, FSM2, FSM3 можно сделать вывод, что выполнение всех этих операций позволяет получить 10 результатов сложения двух байтов с занесением их в память. Если имеется неисправность лишь в одном из элементов множества $\{P_i\} \cup \{A, P, B, C, D, E, H, \text{L})$, $i = \overline{1,10}$, приводящая к неправильному результату вычисления, то эту неисправность можно выявить, а ее влияние в ряде случаев устранить путем сравнения получаемых результатов вычисления.

Действительно, пересечение перечисленных подмножеств равно пустому множеству,т. е.

$$\text{FSM1} \bigcap_{i=1}^{8} \text{FSM2}(i) \cap \text{FSM3} = \varnothing.$$

Следовательно, неисправный элемент не участвует во всех операциях сложения. Кроме того, неисправность ряда узлов можно достаточно просто выявить. Так, неисправность регистра H можно выявить путем изменения тестовых наборов, над которыми выполняется сложение. Если для операций FSM1, FSM2 (i), $i = 1, 2,3\ 4, 6, 8$ получаем один результат и для FSM2 (5), FSM2 (7) — другой, то при принятых предположениях неисправным является регистр H.

Учитывая различный характер использования регистров (например, применение регистра для адресации или хранения данных) в макрокомандах FSM1 и FSM2, можно получить достаточное количество результатов сложения. Анализ их позволяет локализовать неисправность ряда регистров (например, регистра H). При выборе тестов можно использовать традиционные подходы к оптимизации их состава [17, 18, 21, 22]. Для повышения отказоустойчивости целесообразно применять несколько микропроцессоров. Это позволяет в значительной степени устранить взаимокомпенсацию неисправностей в пределах отдельных микропроцессоров и выявлять неисправности нескольких элементов.

Необходимо отметить, что приведенный пример показывает лишь принцип контроля выполняемых микропроцессором операций. Алгоритмы контроля и диагностирования для конкретных условий должны учитывать особенности выполнения операций и размещение исходных программ и данных. Кроме того, зная внутреннюю структуру узлов управления и дешифрации, выполняющих требуемые команды, можно расширить подмножество $\{P_i\}$, учитывая разделение отдельных команд на подмножества микрокоманд и участвующих в их выполнении функциональных узлов микропроцессора. Таким образом можно обеспечить улучшение глубины диагностирования в пределах отдельных элементов системы.

Описанный подход можно использовать как для автономной проверки микропроцессоров, так и для функционального контроля. Во втором случае может быть организована СВК, в которой проверяющий микропроцессор производит задание варианта реализации одной или нескольких макрокоманд контролируемым микропроцессором. При этом контролирующий микропроцессор осуществляет вычисление по макрорасширению, отличному от заданного для контролируемого, с последующим сравнением получаемых результатов вычислений. Таким образом обеспечивается сокращение количества одиночных неисправностей и ошибок вычислений и повышается достоверность самих вычислений. При этом контролируется не только работоспособность самих микропроцессоров, но и практически всех других элементов системы, участвующих

в процессе контроля (ЗУ, линий связи, мультиплексоров и т. д.).

Последующая обработка синдромов, получаемых для различных связей взаимоконтроля, позволяет выделить неисправные и исправные элементы системы, как это описано в гл. 2 и 3.

При наличии заведомо работоспособного микропроцессора контроль проверяемого микропроцессора можно произвести, задавая одинаковые входные тестовые наборы на их входы с последующим сравнением выходных сигналов. Одним из важных преимуществ такого подхода является то, что проверка практически исключает ошибки и неточности, связанные с моделями диагностирования микропроцессора. Задавая случайным образом произвольные тестовые наборы, получают возможность выявлять неисправности, не предусмотренные в моделях.

Аппаратурное решение устройств контроля микропроцессоров при этом получается достаточно простым, а генерирование тестов не представляет больших затруднений, так как обычно используется случайное генерирование входных тестов.

При конкретных применениях возникает потребность оценки качества диагностирования по случайным входным тестовым последовательностям. После задания входной последовательности тестов представляет интерес решение двух основных вопросов: какая вероятность обнаружения заданной неисправности по выходам микропроцессора; какова вероятность обнаружения дефекта в наихудшем случае.

Рассмотрим модель диагностирования по случайным последовательностям [57].

Микропроцессор можно рассматривать как конечный автомат Мура $M = (X, Q, Z, \delta, \omega)$, где $X = (x_1, x_2, \ldots)$ — множество входных векторов; $Q = \{q_1, \ldots, q_n\}$ — множество внутренних состояний; $Z = \{z_1, z_2, \ldots\}$ — множество выходных векторов, а δ определяет функцию перехода.

Обозначим $J = X_1, X_2, \ldots, X_v$ — входную последовательность, а $\delta(q_t, X_l)$ и $\delta(q_t, J)$ — состояния, в которые приходит автомат из начального состояния q_t при задании соответственно входного набора X_l и последовательности J. Выходы микропроцессора задаются с помощью выходной функции $\omega(q_t) \in Z$. При наличии неисправности f автомат M превращается в автомат $M^f = (X, Q^f, Z, \delta^f, \omega^f)$, где $Q^f = (q_1^f, \ldots, q_m^f)$ такова, что $m = n$ или $m \neq n$.

Предполагается, что конечный автомат в исправном состоянии можно перевести в заранее известное состояние $q_0 \in Q$. Диагностический эксперимент состоит в задании входных последовательностей и наблюдении выходов автомата. Говорят,

что состояние q_j^f из M^f I-совместимо с состоянием q_i, если при диагностическом эксперименте M^f находится в состоянии q_j^f при условии, что M находится в состоянии q_i. Множество состояний в Q^f I-совместимых с $q_i \in Q$ обозначим C_i^f.

Последовательность переходов T из состояния q_i есть последовательность последовательных переходов в исправном автомате M при задании входной последовательности J:

$$T = q_i J.$$

Последовательность T называется последовательностью проверки переходов (ППП) для неисправности f, если задание последовательности J для автомата, находящегося в состоянии q_i, позволяет выявить неисправность для любых состояний из C_i^f. Минимальной ППП (МППП) для неисправности f является последовательность, представляющая собой также ППП. Множество D^f всех МППП для заданной неисправности f образует множество проверки неисправности f.

Предположим, что длина входной последовательности достаточно велика, следовательно, вероятность состояния q является стационарной. Неопределенность проверки Q_D^f при наличии неисправности f есть вероятность не обнаружения ее.

Средняя вероятность p_f обнаружения неисправности f для любого входного вектора при диагностическом эксперименте

$$Q_D^f = (1 - p_f)^{L_f}.$$

Отсюда длина тестовой последовательности

$$L_f = \log Q_D^f / \log (1 - p_f).$$

Для заданного множества неисправностей F величина

$$Q_D = \max Q_D^f, \quad f \in F.$$

Задача контроля микропроцессора состоит в получении нижней границы p для наиболее сложно выявляемых неисправностей:

$$p \leqslant \min p_f, \quad f \in F.$$

Следовательно, средняя длина случайной последовательности

$$L = \log Q_D / \log (1 - p).$$

Обозначим $p\,[T]$ вероятность применения T, а $p\,[D^f]$ — вероятность использования любой МППП из D^f.

Для тестовых последовательностей справедливы соотношения:

1. $p[qJ] = p[q]\,p[J] = p[q]\,p[x_1]\ \ldots\ p[x_r]$, $J = X_1, X_2, \ldots, X_r$.

2. $p[D^f] = \sum_t p[T_t]$, $T_t \in D^f$.

3. Если для любого $T_t \in D^f$ существует наименьшая одна ППП T_j, $T_j \in D^q$, T_j есть часть последовательности T_t (D^f охватывает D^q), то $p[D^f] \leqslant p[D^q]$.

4. $p[D^f] \leqslant p_f$.

Таким образом, используя данные соотношения, можно оценить вероятность выявления неисправностей для произвольных входных последовательностей тестовых сигналов.

Для оценки качества диагностирования необходимо найти множества диагностических последовательностей.

Микропроцессор можно представить графом, вершинами которого являются регистры и операторы R_t, O_j. Входная дуга вершины O_j оператора соответствует операндам, а выходная — результату. Одной и той же команде микропроцессора может соответствовать несколько вершин операторов, в зависимости от используемых операндов и места занесения результата вычисления. Вершины связаны дугами, соответствующими выполнению операторов. Будем говорить, что оператор O_j следует (предшествует) за регистром R_t, если существует путь из R_t к O_j, $R_t \to O_j$ ($O_j \to R_t$). Направленный путь от R_t к R_n представляет собой такую цепочку $R_1O_2R_2 \ldots \ldots R_{i-1}O_tR_i \ldots O_nR_n$, что $R_{i-1} \to O_t$, $O_t \to R_i$, $R_t \neq R_j$, $O_i \neq O_j$.

В простейшем случае оператор O_j микропроцессора реализуется одной командой I_j, которая называется изменяющей, если существует $R_t \neq R_t$: $R_t \to O_j$ и $O_j \to R_t$. Множество изменяющих команд для регистра R_i обозначим C_i'.

Команда I_j, выполняющая оператор O_j, называется командой передачи для R_t, если существует один направленный путь от R_i к $R_{\text{вых}}$, начиная с R_iO_j.

Множество команд передачи обозначим P_t'.

Командой установки для регистра R_t называется команда, которая не является изменяющей командой или командой передачи. Множество таких команд определяется как

$$K_t = \overline{C}_t' \cap \overline{P}_t',$$

т. е. пересечение множеств команд, не входящих в C_t' и P_t'.

Обозначим A_k^t множество команд, не содержащих направленного пути $k \in \{1, \ldots, m\}$ и содержащие наименьшее одну команду каждого из $m-1$ других путей. Произвольно выбранную из A_k^t команду обозначим A_k. Условия обнаружения неисправностей в регистре R_t микропроцессора вытекают из следующих соотношений: при $D^f \supset Q_fS_t$, где $Q_f \subset Q$ — мно-

жество состояний в R_i, на которые влияет f; S_i — множество последовательностей команд, определяемых, если существует один направленный путь,

$$R_i O_{i+1} R_{i+1} \ldots R_{n-1} O_n R_{\text{вых}}$$

от заданного регистра к выходному $R_{\text{вых}}$: $S_i = I_{i+1} K^*_{i+1} I_{i+2} \times \times K^*_{i+2} \ldots I^*_{n-1} K^*_{n-1} I_n$; если существует m направленных путей,

$$R_i O^k_{i+1} K^k_{i+1} \ldots R^k_{n-1} O^k_n R_{\text{вых}} \quad (k = 1, \ldots, m)$$

от R_i к $R_{\text{вых}}$;

$$S_i = \sum_{k=1}^{m} I^k_{i+1} (K^k_{i+1} \cap \bar{A}_k)^* I^k_{i+2} (K^k_{i+2} \cap \bar{A}_k)^* \ldots I^k_n.$$

При наличии одного пути команда I_{i+1} пропускает ошибку R_{i+1}, затем любая цепочка команд из K^*_{i+1} задерживает ошибку в R_{i+1} и далее I_{i+2} пропускает ошибку к R_{i+2} и т. д.

При наличии неисправностей операторов, также соответствующих вершинам графа, описывающего микропроцессор, установлены следующие условия выявления этих отказов [57].

Пусть O_j — оператор, реализуемый командой I_j такой, что O_j предшествует только одному регистру R_i. Тогда для неисправности f в O_j

$$D^f \supset Q_f I_j (K_i \cap \bar{I}_j)^* S_i,$$

где Q_f — множество состояний, определенных относительно регистров, предшествующих оператору O_j, таких, что O_j выдает неправильный результат при выполнении I_j.

Таким образом, рассмотренная модель позволяет оценить качество диагностирования микропроцессора в процессе эксплуатации и при отбраковке микропроцессоров, предназначенных для обслуживания отказоустойчивых микропроцессорных систем.

2. ОРГАНИЗАЦИЯ СВЯЗЕЙ В СИСТЕМЕ

Организация контроля и восстановления элементов в микропроцессорных отказоустойчивых системах производится по линиям связи между элементами. Связь обеспечивает передачу тестовой информации, получение результатов контроля, выдачу команд на отключение или подключение элементов.

Для качественного диагностирования необходимо большое количество связей, реализация которых приводит к значительному снижению показателей надежности микропроцессорной системы, а также существенно ухудшает характеристи-

ки стоимости, массы, габаритов, энергопотребление, функциональную гибкость системы. Поэтому обычно ограничиваются небольшим количеством общих (магистральных) линий связи, имеющих резервированные. В этом случае организация связей между элементами обеспечивается аппаратно с использованием схем арбитража. При соединении различных элементов между собой используются специальные интерфейсные устройства, обеспечивающие выбор и подключение к шинам. Различные узлы или программные реализации арбитров шин позволяют определить приоритет подключения элементов к шинам.

Связь с внешними устройствами осуществляется с помощью интерфейса, под которым понимают совокупность цепей (бу-

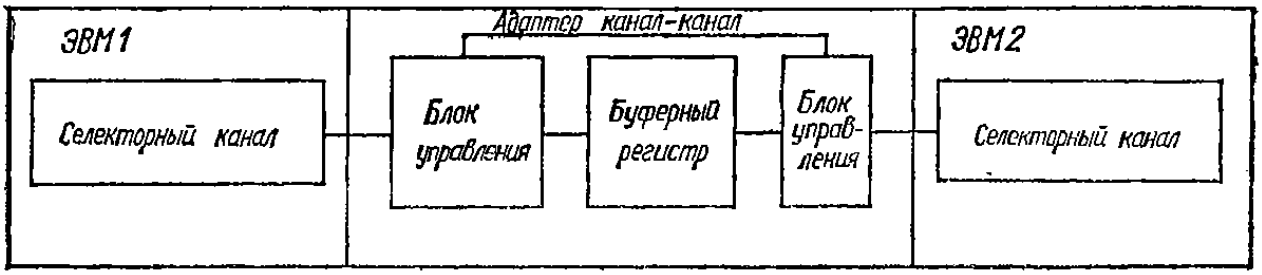


Рис. 24. Сопряжение ЭВМ через адаптер

ферных регистров, устройств управления, шин), сигналов и алгоритмов, обеспечивающих сопряжение между микропроцессором и периферийными устройствами (внешний интерфейс), а также между блоками микропроцессоров (внутренний интерфейс).

Организация связей осуществляется с применением специальных устройств сопряжения: адаптеров, контроллеров и др. Однако связь может осуществляться и непосредственно через выходные и входные буферные регистры.

Адаптеры согласуют работу различных частей системы без логической обработки. В отказоустойчивых системах адаптеры типа канал—канал могут использоваться для связей каналов отдельных ЭВМ или процессоров между собой в пределах одной системы, или для каналов различных систем. Пример подключения селекторных каналов двух ЭВМ с помощью адаптера канал—канал показан на рис. 24. Селекторный канал, управляющий на время связи обменом данными только между двумя ЭВМ, позволяет производить быструю передачу информации, например тестов, от одной ЭВМ. Адаптер периферийного интерфейса согласует взаимодействие периферийных устройств с микропроцессором через порты ввода-вывода.

Контроллеры представляют собой устройства управления вводом и выводом. Они согласуют интерфейсы различных устройств системы и работают, как правило, по жесткому алго-

ритму обмена. Контроллер обычно обслуживает определенные внешние устройства, что позволяет аппаратурно выполнять основные логические операции (упаковка и распаковка данных и др.) и резко повышать быстродействие системы.

Наиболее распространенными в микропроцессорных системах являются контроллеры периферийных устройств электрической пишущей машины «Консул», дисплеев («Видеотон-340», «ДМ-2000», «ВТА-2000», «РИН-609»), фотосчитывателя FS-1501, перфоратора ПЛ-150. В связи с расширением номенклатуры и количества периферийных устройств, повышением требований по надежности все большее распространение получают программируемые контроллеры, выполненные на нескольких БИС. Кроме того, схемы контроллеров часто входят в состав схем микропроцессорных комплектов.

В программируемых контроллерах возможно использование системы команд и микропрограмм, реализующих достаточно сложные протоколы обмена. Смена алгоритмов может производиться с помощью замены встроенных ПЗУ микропрограмм и использования ОЗУ для сравнения параметров, промежуточных результатов, необходимых данных при реализации конкретных алгоритмов обмена. Такие контроллеры в сочетании с универсальностью обеспечивают высокое быстродействие и разгружают основной процессор от выполнения процедур обмена. Применение программируемых интерфейсов, допускающих активное использование в обмене программ процессора, увеличивает универсальность системы. Программируемый интерфейс на базе микропроцессорного комплекта К587 показан на рис. 25. В качестве функциональных узлов обмена информацией БИС ОИ выступают микросхемы К587 ИК1 (микропроцессор), а в качестве управляющей памяти БИС УП — микросхемы К587 РП1 (управляющая память). Портами в данном интерфейсе являются двунаправленные магистральные регистры БИС ОИ2 — БИС ОИ5, управление которыми осуществляется с помощью БИС УП2.

Связь микро-ЭВМ с магистралью производится с помощью узла выборки, БИС УП1, БИС ОИ1 и узла прерываний. По линиям ввода инициативных сигналов поступают сигналы прерывания (с учетом приоритета запроса на прерывание). В состав программного обеспечения интерфейса можно вводить специальные программы — драйверы, предназначенные для работы с определенными портами системы и учитывающие специфику режимов их работы (особенно при асинхронном обмене информацией). В частности, можно учесть возможность последовательного приема или выдачи информации.

Необходимость передачи данных в системе заставляет решать вопросы организации мультиплексных и селекторных

каналов в мультимикропроцессорной системе. При организации селекторных каналов блокируется связь с другими портами в соответствии с приоритетами портов и производится быстрая пересылка пакетов данных от внешних устройств под управлением специальной микропрограммы селекторного канала.

Организация мультиплексного канала, например, при одновременном побайтном приеме данных от различных пор-

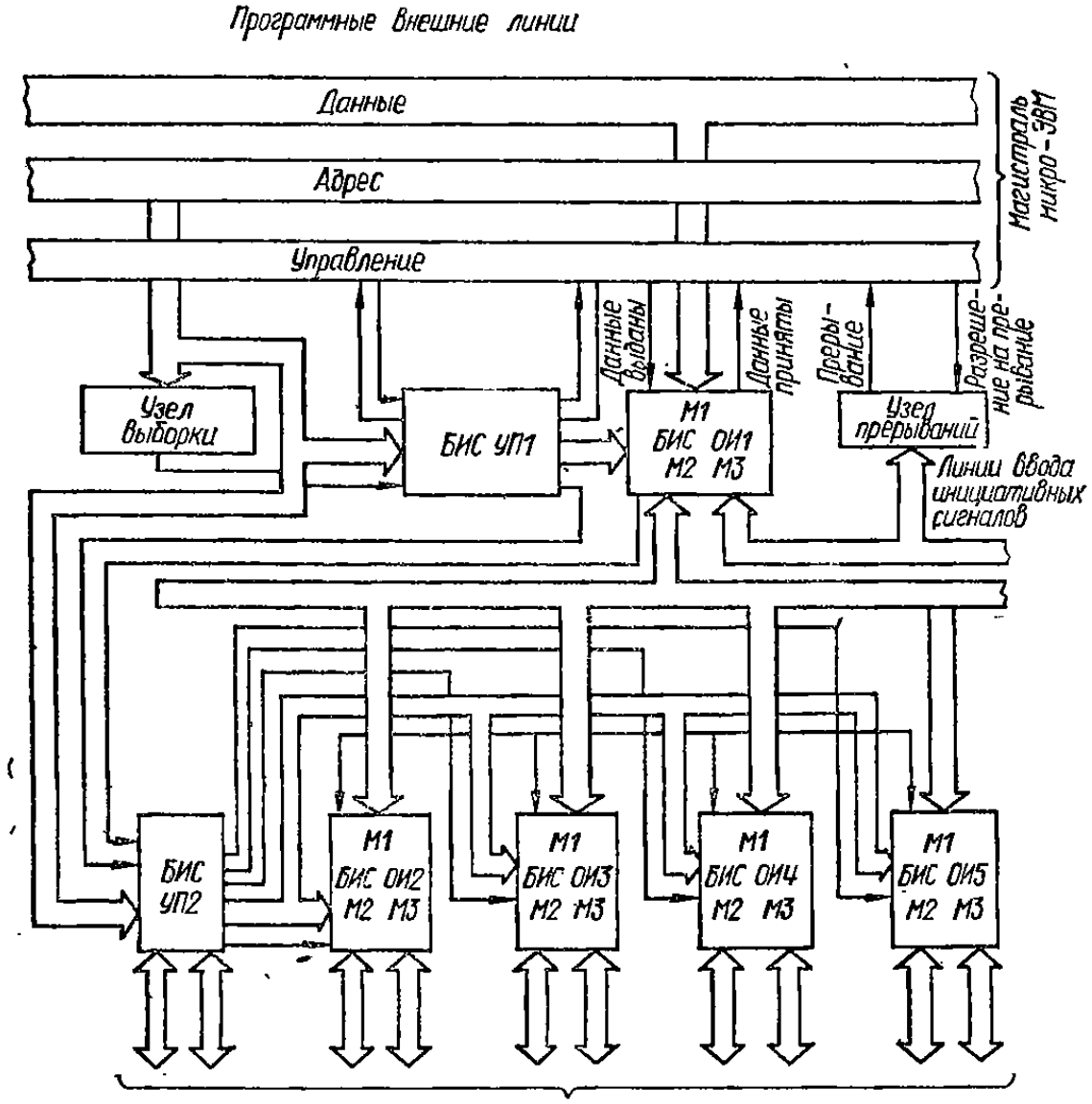


Рис. 25. Программируемый интерфейс

тов, может производиться последовательным опросом согласно с приоритетом портов соответствующих двунаправленных регистров. При управлении может быть обеспечена реализация требуемого протокола обмена (в частности, выдача и сброс сигналов готовности информации). Такая организация значительно расширяет функциональную гибкость обменом информацией в микропроцессорной системе. В процессе диагностирования отказоустойчивой системы это позволяет принудительно инициировать работу отдельных микро-ЭВМ для проведения, в частности, замен.

Таким образом, применение адаптеров и контроллеров, а также программируемых интерфейсов, обладающих боль-

шими возможностями адаптации к выбору интерфейса обмена, обеспечивает эффективность использования одних и тех же линий связи для взаимодействия процессоров, запоминающих устройств и внешнего периферийного оборудования. В связи с применением микропроцессоров удается создавать малогабаритные контроллеры, встраиваемые в состав периферийного оборудования, что расширяет возможности взаимодействия в процессе диагностирования.

Связь между отдельными элементами отказоустойчивой системы для передачи данных, адресов и управляющей информации осуществляется обычно по небольшому числу физических связей. Одновременный обмен между отдельными элементами, как правило, невозможен, а образование большого количества физических связей приводит к увеличению стоимости и ухудшению технических характеристик (надежности, объемов, массы, энергопотребления и др.).

В многомашинных системах наиболее просто осуществлять попарное взаимодействие элементов системы (микро-ЭВМ, микропроцессоров, ЗУ). Поэтому для улучшения характеристик диагностируемости, требующей большого количества связей, используются централизованные устройства арбитража по шинам или специальные схемы идентификации, захвата и разрешения на обмен данными.

Выбор логической структуры арбитража определяется спецификой отказоустойчивой системы, ее функциональным назначением. Так, например, в микропроцессорах и микро-ЭВМ с совмещенными линиями связи для адресов и данных (входных и выходных) возникает необходимость передачи информации в различных направлениях. Это делается с использованием двунаправленных коммутаторов сигналов.

Практический интерес представляет рассмотрение аппаратурной реализации подключения нескольких выходов на одни и те же линии связи. Наиболее широко применяются следующие типы объединения одноименных выходов устройств на однопроводные линии связи: объединение выходов с открытым коллектором; логическое объединение; объединение выходов с использованием схем с тремя состояниями.

Первые два вида объединения достаточно широко использовались для микросхем малой и средней степени интеграции, изготовленных, например, по ТТЛ- и ТТЛДШ- технологии (микросхемы серии 133, 155, 131 и др.). Подключив несколько выходов микросхем 133 ЛА8 на общее коллекторное сопротивление, получим единичный сигнал, выставляемый при единичных выходах всех микросхем (отсутствие захвата). Наличие же, по крайней мере, одного нулевого сигнала на выходе одной из схем приводит к нулевому сигналу на общем

выходе. Логическое объединение производится с использованием схем U или дополнительной аппаратуры. В обоих случаях наблюдается взаимозависимость режимов работы выходных микросхем.

Третий вид объединения широко используется для логических элементов, имеющих в отключенном состоянии практически бесконечное выходное сопротивление.

Схемы объединения выходов с открытым коллектором показаны на рис. 26,*а*. Здесь $R_к$ — общая для выходов транзисторов нагрузка, а транзисторы обозначают выходные каскады микросхем.

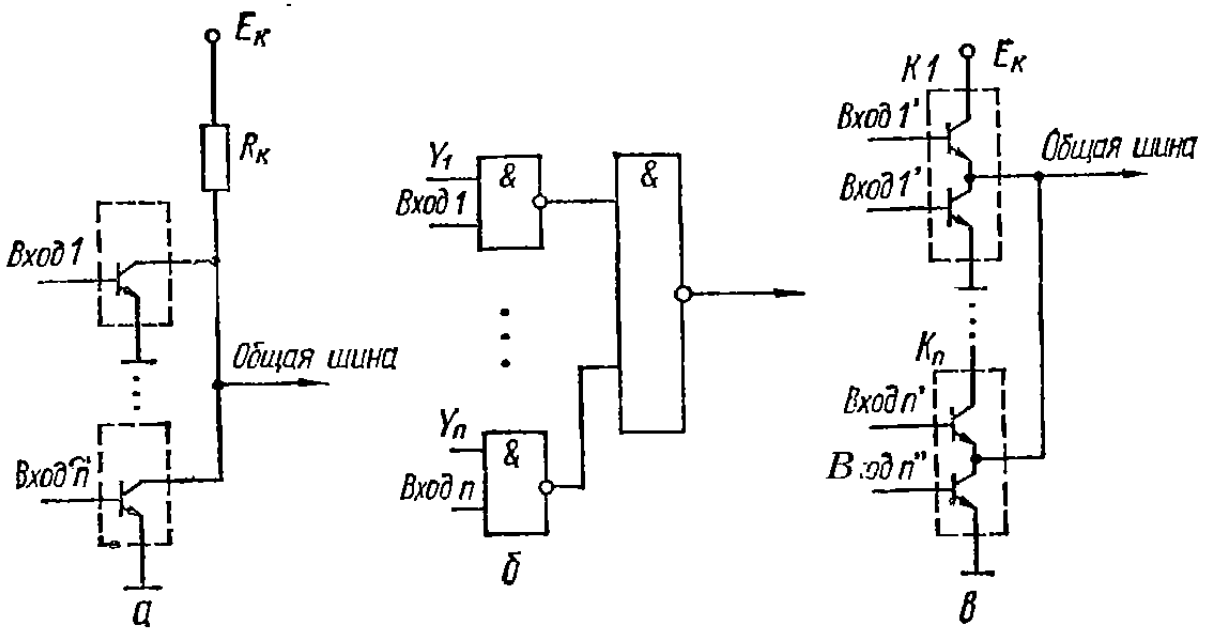


Рис. 26. Схемы объединения выходных шин

На рис. 26,*б* показана схема объединения выходов. В этом случае для передачи входного сигнала *Вход i* на выход *Общая шина* на все управляющие входы $Y1$, ..., Y_n, кроме передаваемого Y_i, необходимо подать сигналы «логический 0», а на Y_i — «логическая 1».

Существуют различные схемные решения объединения выходов с использованием схем с тремя состояниями. Пример реализации объединения для случая ТТЛ-технологии показан на рис. 26,*в*. Применение схем с тремя состояниями позволяет создавать двунаправленные шинные формирователи, необходимые для целого ряда интерфейсов (например, для интерфейса типа *Общая шина*). На рис. 27,*а,б* показаны соответственно функциональная схема и условное обозначение двунаправленных шинных формирователей К589 АП16 и К589 АП26.

При наличии сигнала «логический 0» на управляющих входах *ВК* и *УВ* формирователи *С*, связанные с выходами C_1, C_2, C_3, C_4, переходят в третье состояние (высокое выходное сопротивление) и сигнал передается от A_1, A_2, A_3, A_4 к B_1, B_2, B_3, B_4.

При сигнале «логический 0» на входе $ВК$ и «логическая 1» на входе $УВ$ в третье состояние переходят формирователи, связанные с входами A_1, A_2, A_3, A_4, и сигналы передаются от B_1, B_2, B_3, B_4 к C_1, C_2, C_3, C_4. В случае «логическая 1» на входах $ВК$ и $УВ$ передача не происходит вообще и выходы A_1, ..., A_4, B_1, ..., B_4 не влияют на связанные с ними шины B_1, B_2, B_3, B_4, C_1, C_2, C_3, C_4.

Подключая B_1, ..., B_4 к общей шине, получаем возможность двунаправленной передачи информации по «общей шине» (прием и передача) или вообще отключение микропроцессорных элементов от шины (например, микропроцессоров К584).

Используя выходы B_1, ..., B_4 в качестве совмещенных входов/ выходов микропроцессора (например, с совмещением шин адресов и данных), получаем возможность работать с однонаправленными интерфейсными шинами (например, с интерфейсами 2К и ЕС ЭВМ).

Таким образом, существующие схемные решения обеспечивают эффективную связь между элементами отказоустойчивой системы при достаточно ограниченном количестве связей в системе.

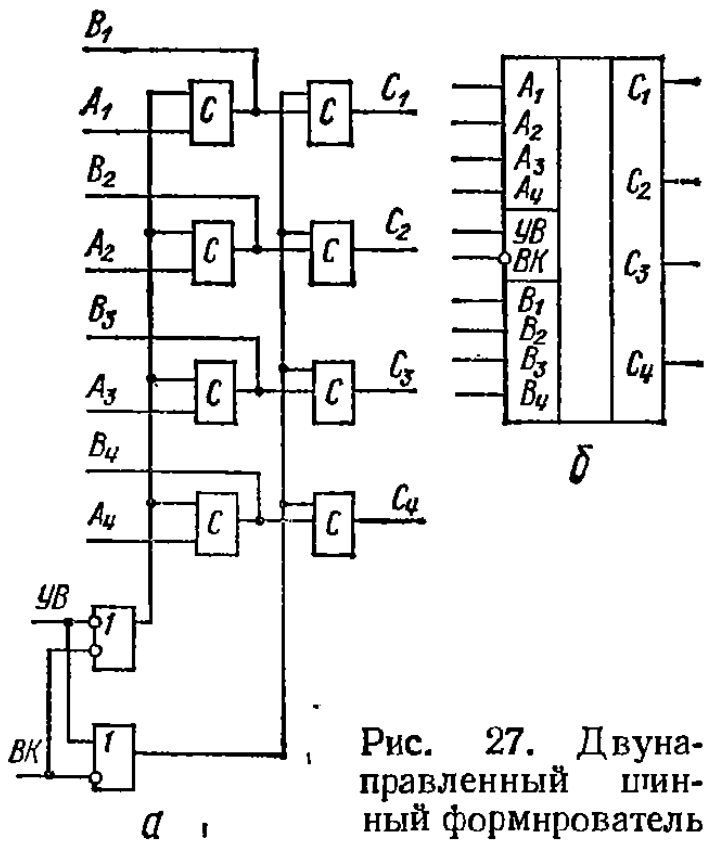


Рис. 27. Двунаправленный шинный формнрователь

3. ОБРАБОТКА РЕЗУЛЬТАТОВ КОНТРОЛЯ

Выявление и устранение неисправостей в процессе эксплуатации отказоустойчивой системы обеспечивается соответствующими средствами. Взаимный контроль одного элемента системы другим должен по существу учитывать функциональные и структурные особенности этих элементов. Обработка же полученных результатов контроля носит более универсальный характер. Поэтому программные и аппаратные средства обработки данных контроля можно использовать для большого количества систем. При этом необходимо учитывать, что в процессе эксплуатации для повышения эффективности обнаружения и устранения дефектов в системе можно применять различные алгоритмы диагностирования и контроля.

Рассмотрим некоторые примеры определения неисправных элементов системы по известным результатам контроля.

Обозначим $M_t(i)$ — подмножество элементов, выделенных на данном этапе проверки как исправные, M_b — подмножество элементов v_i, которые проверены на наличие связей $a_{ij} = 0$, а S_i — подмножество элементов v_j, связанных с исходным элементом $v_t \in V$ связью $a_{ij} = 0$, т. е.

$$S_t = \{v_j \mid \exists a_{t,j} = 0,\ v_t,\ v_j \in V\};$$

$$M_t(i) = \{v_{j_n} \mid \exists i,\ j_1,\ \ldots,\ j_n : a_{i,j_1} = 0,$$

$$a_{j_1,j_2} = 0,\ \ldots\ a_{j_{n-1},j_n} = 0\}.$$

Все элементы системы, не вошедшие в подмножество $M_t(i)$, потенциально являются неработоспособными.

Для каждого работоспособного элемента v_t, $v_j \in V$ подмножество $M_t(i)$ состоит из работоспособных элементов. Все остальные элементы $M \setminus M_t(i)$ можно первоначально считать неработоспособными. В результате для каждого заданного элемента v_t может быть получен набор состояний работоспособности элементов.

Сравнивая эти наборы для различных начальных элементов v_t, можно сделать вывод о работоспособности элементов системы в целом.

Определение наборов для отдельных элементов можно выполнять как аппаратно, так и программно.

Для определения подмножества $M_t(i)$ предложена схемная реализация устройств контроля для определения наборов $M_t(i)$ и $M \setminus M_t(i)$ [45]. Устройство состоит из унифицированных функциональных узлов, связанных между собой в соответствии со структурой связей в системе. Количество таких устройств, в общем случае, равно числу элементов системы.

На рис. 28,*а* показана упрощенная схема функционального узла. По выходу W выдается сигнал работоспособности по соответствующему элементу, а выходы Z_1, Z_0 предназначены для формирования транзитных сигналов передачи от одних элементов к другим по однонаправленным связям. Устройство для элемента v_0 системы, структура связи которой показана на рис. 28,*б*, показано на рис. 28,*в*. В квадратах, соответствующих унифицированным функциональным узлам указаны состояния связей.

При определении характерных неисправностей в системе необходимо сравнить между собой наборы состояний элементов v_1, ..., v_n от соответствующих устройств. В качестве характерных можно выбрать наиболее часто встречающиеся наборы или, в пределах одного элемента, набор состояния элемента по большинству. Во всех случаях требуется дополнительная

аппаратура. В результате снижается общая надежность системы, требуется дополнительная проверка работоспособности устройств контроля.

Таким образом, схемная реализация обработки контрольно-диагностической информации требует большого количества дополнительной аппаратуры и снижает надежность системы. Поэтому большие преимущества имеет программная реализа-

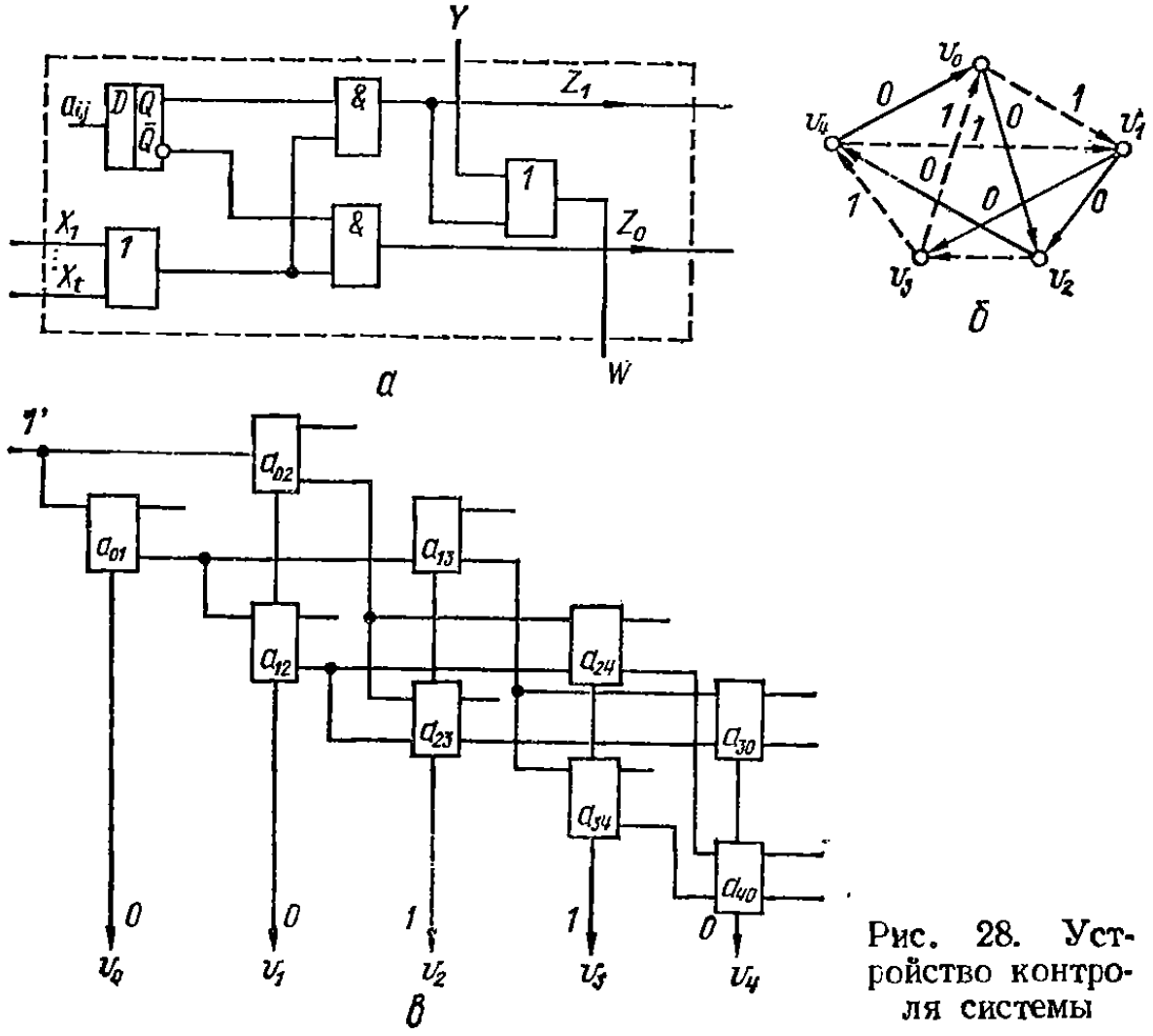


Рис. 28. Устройство контроля системы

ция. Алгоритм выделения множества $M_t^{'}(i)$ для элемента i имеет вид

1. $M_b = \varnothing$, $M_t = \varnothing$, $S_t = \varnothing$.
2. $M_b = M_b \cup i$.
3. $M_t = M_t \cup S_t \cup i$.
4. $M_t^{'} = M_t \setminus M_b$.
5. Если $M_t^{'} = \varnothing$, то конец.
6. Выбрать $v_i \in M_t^{'}$. Перейти к шагу 2.

Программная реализация алгоритма на языке ассемблер для микропроцессора К580ИК80 для восьми элементов системы занимает около 30 байт.

Высокую достоверность при использовании программных модулей можно обеспечить за счет дублирования данных, применения различных вариантов реализации модулей.

4. КОНТРОЛЬ УЗЛОВ УПРАВЛЕНИЯ И ЗАПОМИНАЮЩИХ УСТРОЙСТВ

В настоящее время в микропрограммных устройствах управления и контроллерах микропроцессорных систем используются программируемые логические матрицы (ПЛМ). Регулярность структуры ПЛМ позволяет создавать на их основе БИС и СБИС с возможностями реализации логических функций от большого числа двоичных переменных. Поскольку наличие отказов в устройствах управления значительно сказывается на правильном выполнении системой заданных функций, то возникает необходимость постоянной их проверки. Проверка работоспособности ПЛМ требует задание входных тестовых наборов и анализ их на выходных контактах ПЛМ.

Тестовые наборы можно задавать, используя специальные ЗУ. Однако при качественной проверке ПЛМ на БИС и СБИС это требует больших объемов памяти, что снижает достоверность контроля и усложняет обслуживание ПЛМ. Поэтому более распространенным в настоящее время является введение в состав ПЛМ специальных схем, упрощающих получение тестов и выявление отказов. Рассмотрим пример такой проверки [23, 24, 58].

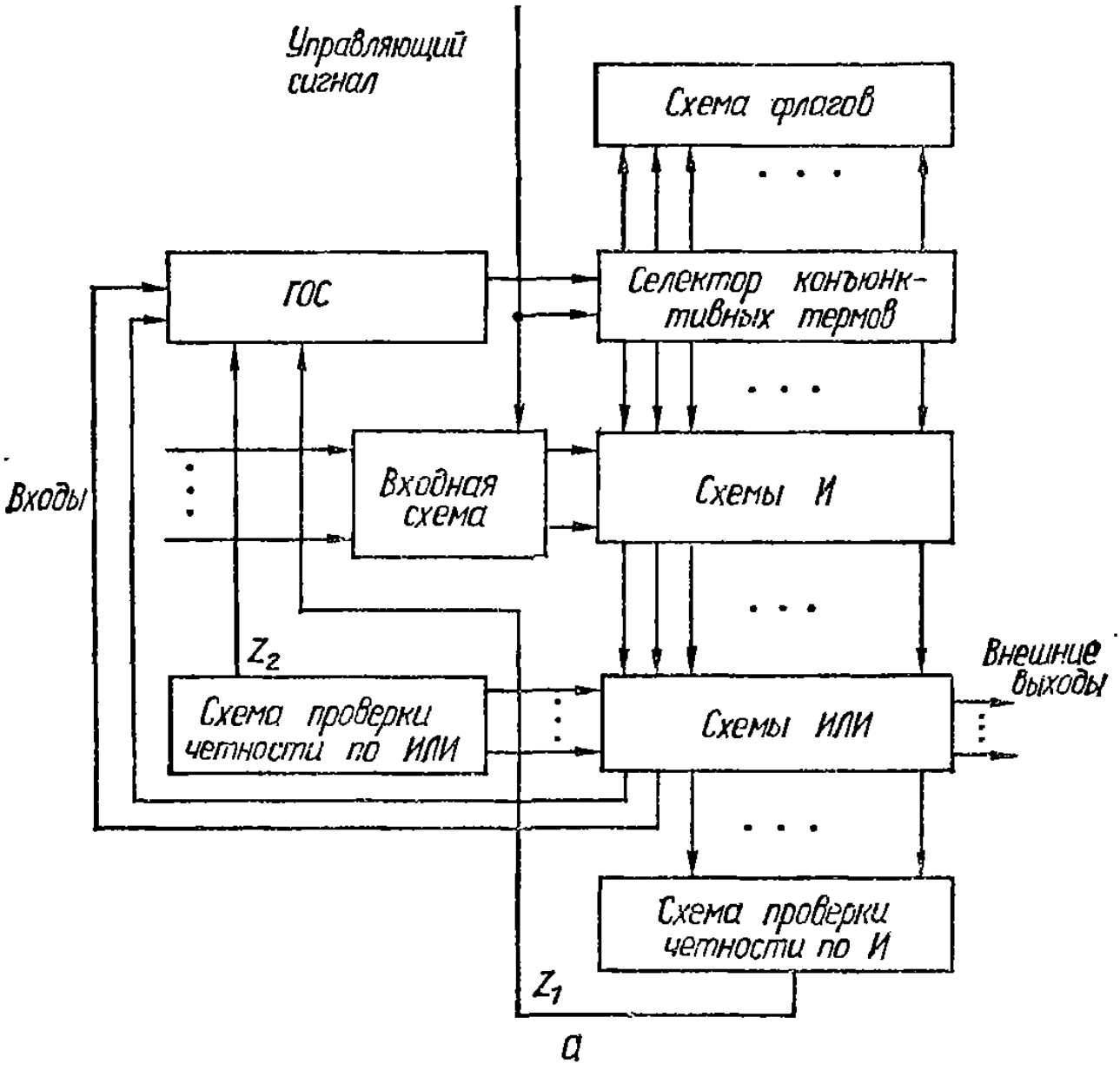
Программируемая матрица, показанная на рис. 29, *а*, имеет дополнительные схему сигнализации, генератор обратной связи *ГОС* и схемы проверки четности по И и ИЛИ. Кроме того, в схемы И и ИЛИ ПЛМ введены дополнительные линии $W_{m+1}, \ldots, W_{m+4}$ и два выхода O_{t+1}, O_{t+2} (рис. 29, *б*).

Соединение линии W_{m+1} таково, что оно позволяет получать нечетное число контактов в ПЛМ по каждой строке и соединено только с выходом O_{t+1}. Линии W_{m+2}, W_{m+3} не имеют соединений в схемах И и соединены со всеми строчками схемы ИЛИ, за исключением строки, связанной с выходом O_{t+2}. Выход O_{t+1} служит для проверки по четности.

Схема проверки по четности выходных значений по И вычисляет четность выходных значений $W_1, \ldots, W_{m+2}$. Схемы проверки четности по ИЛИ определяют четность по выходам $O_1, \ldots, O_{t+2}$. Выходы Z_1, Z_2 соответствуют выходам схем проверки четности по И и ИЛИ. Входная схема и селектор конъюнктивных термов содержит соответственно регистры сдвига *RI*, *RK*, которые используются для проверки ПЛМ.

В режиме обычной работы ПЛМ на выходы $S_1, \ldots, S_m$ регистра *RK* подаются сигналы 1, а на выходы $S_{m+1}, \ldots, S_{m+4}$ — сигналы 0.

Генератор обратной связи по значениям Z_1, Z_2, B_{m+3}, B_{m+4}, (B_{m+3}, B_{m+4} соответствуют выходным переменным W_{m+3},



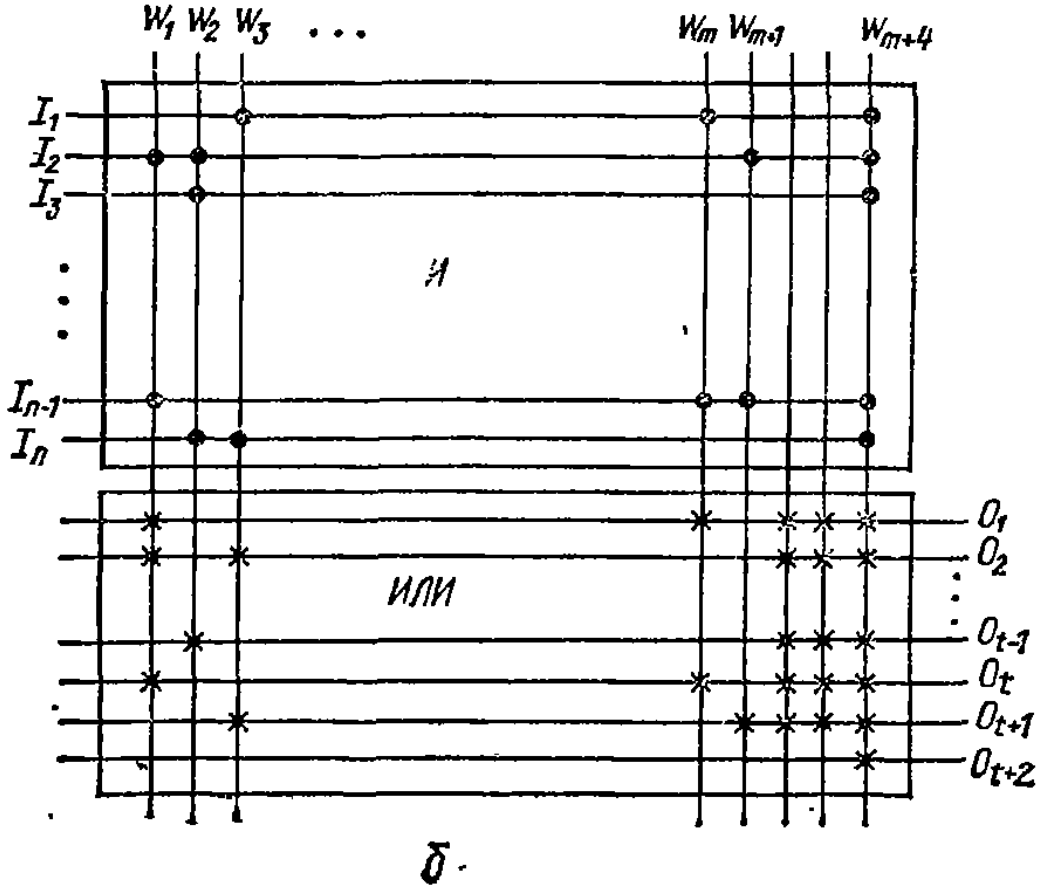


Рис. 29. Самоконтролируемая ПЛМ

W_{m+4}) формирует значение Y, подаваемое на вход регистра RK, следующим образом:

$$Y = \bar{Z}_2\bar{B}_{m+4} + Z_1\bar{B}_{m+3}B_{m+4} + Z_1B_{m+3}\bar{B}_{m+4} + \bar{Z}_1Z_2B_{m+3}.$$

Задание режима контроля ПЛМ производится по управляющему сигналу, поступающему на входы селектора конъюнктивных термов и входной схемы.

Перед началом проверки в регистры RI и RK ПЛМ заносится начальный тестовый набор T_{11} (табл. 7).

Процесс проверки протекает циклически. В каждом цикле сигналы, получаемые по текущему тестовому набору, поступают на входы *ГОС*. При этом сигналы B_{m+3}, B_{m+4} поступают непосредственно с выхода ПЛМ, а Z_1 и Z_2— путем преобразования выходных сигналов ПЛМ по И и по ИЛИ.

В *ГОС* формируется значение Y.

При отсутствии неисправностей по сигналу Y в регистрах RK, RI формируется следующий тест. Наборы генерируемых тестов на различных тактах работы представлены в табл. 7. Общее количество тестовых наборов равно $n + 2m + 8$.

Прохождение последнего теста показывает отсутствие отказов в ПЛМ и в дополнительных схемах. В таких ПЛМ выявляются все одиночные перекрестные дефекты в схемах И и ИЛИ, включая дефекты дополнительных линий, все одиночные константные неисправности в схемах проверки четности по И и ИЛИ и все неисправности входной схемы и селектора конъюнктивных термов [58].

Перекрестные дефекты состоят в прохождении входных сигналов по не заданному пути. Например, проверка перекрестных дефектов по линиям $W_1, \ldots, W_{m+3}$ производится путем задания тестов T_{4j} таким образом, что единица появляется лишь по одной из этих линий. При отсутствии дефектов $Z_2 = 1$. При наличии дефекта на W_j в схемах ИЛИ сумма подключенных точек в j-м столбце становится четной и $Z_2 = 0$. Следствием дефекта является установка на выходе Y единичного сигнала.

Контроль схем проверки четности по И производится с помощью тестов $T_{4m+3}, T_{41}, \ldots, T_{4m+2}$. С помощью тестов $T_{41}, \ldots, T_{4m+4}$ выявляются дефекты в схемах проверки четности по ИЛИ. Выявление отказов в схемах производится с помощью тестов $T_{11}, \ldots, T_{1n}$.

Таким образом, применение достаточно простого аппаратного контроля в ПЛМ обеспечивает обнаружение большинства одиночных неисправностей.

При длительной эксплуатации ПЛМ на основе СБИС возможно возникновение нескольких неисправностей. Контроль и диагностирование в этом случае требуют большого количест-

Таблица 7

№ теста	Обозначение	Состояние линий связи ПЛМ						Место неисправности
		$I_1 \ldots I_j \ldots I_n$	$S_1 S_2 \ldots S_j \ldots S_{m+2} S_{m+3} S_{m+4}$	Z_1	Z_2	B_{m+3}	B_{m+4}	Y	
1	T_{11}	0..1..1	1 1..1..1 1 1	1	1	1	0	1	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
j	T_{1j}	1..0..1	1 1..1..1 1 1	1	1	1	0	1	Схемы И
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
n	T_{1n}	1..1..0	1 1..1..1 1 1	1	1	1	0	1	
$n+1$	T_{21}	1..1..1	1 1..1..1 1 1	0	0	1	1	0	
$n+2$	T_{31}	1..1..1	0 1..1..1 1 1	1	0	1	1	0	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
$n+j+1$	T_{3j}	1..1..1	0 0..0.. 1 1 1	e_j	0	1	1	0	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
$n+m+2$	T_{3m+1}	1..1..1	0 0..0.. 1 1 1	1	0	1	1	0	
$n+m+3$	T_{3m+2}	1..1..1	0 0..0.. 0 1 1	0	0	1	1	0	
$n+m+4$	T_{4m+4}	1..1..1	0 0..0.. 0 0 1	0	0	0	1	0	W_{m+4}
$n+m+5$	T_{51}	1..1..1	0 0..0.. 0 0 0	0	0	0	0	1	
$n+m+6$	T_{41}	1..1..1	1 0..0.. 0 0 0	1	1	0	0	0	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	$W_1 \ldots W_{m+3}$
$n+m+j+5$	T_{4j}	1..1..1	0 0..1.. 0 0 0	1	1	0	0	0	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
$n+2m+7$	T_{4m+2}	1..1..1	0 0..0.. 1 0 0	1	1	0	0	0	
$n+2m+8$	T_{4m+3}	1..1..1	0 0..0.. 0 1 0	0	1	1	0	0	
		Конец	0 0..0..0 0 1						

Примечание. $e_j = \begin{cases} 0, \text{ если } j \text{ четное;} \\ 1, \text{ если } j \text{ нечетное.} \end{cases}$

ва тестов. Применение локальной и глобальной памяти для хранения тестов позволяет эффективно решать задачи контроля работоспособности ПЛМ при наличии большого числа дефектов.

Характерными для ПЛМ являются дефекты, связанные с неправильной коммутацией входных сигналов на выходе ПЛМ. Это может быть следствием как ошибок при изготовлении (например, ошибки трассировки), так и неисправностей, возникающих в процессе функционирования. В ряде работ показывается, что тесты одиночных неисправностей позволяют обнаружить значительную часть кратных неисправностей ПЛМ.

Основой для получения наборов тестов являются модели неисправностей. Рассмотрим одну из наиболее известных моделей [35]. Предположим, что в исправной ПЛМ между строкой и столбцом существует 1-связь (0-связь), если между ними существует (отсутствует) связь. Неисправность рассматривается, как изменение 0-связи (1-связи) на 1-связь (0-связь).

Физически 0-связь и 1-связь реализуется с помощью диодов и плавких вставок. 1-связь соответствует, например, наличию плавкой вставки между соответствующей строкой и столбцом, а 0-связь — ее пережиганию. Неисправность сводится к тому, что 1-связь пережигается, т. е. превращается в 0-связь, либо возникает паразитная связь, например, при коротком замыкании (переход от 0-связи к 1-связи). Входные переменные ПЛМ представлены своими прямыми и инверсными входными столбцами. Поэтому общее количество одиночных неисправностей в ПЛМ

$$N_1 = m(2n + p),$$

где m — количество строк ПЛМ; n — количество входных переменных; p — количество выходных переменных (столбцов).

Число N_r r-кратных неисправностей, $r > 1$, состоит из r одиночных неисправностей и определяется как число сочетаний

$$N_r = C^r_{m(2n+p)}.$$

Следовательно, общее количество всех кратных неисправностей, представленных от 1 до $m(2n + p)$ одиночными неисправностями,

$$N_0 = \sum_{r=1}^{m(2n+p)} C^r_{m(2n+p)}.$$

Получение полного множества T_m для всех кратных неисправностей ($r > 1$) представляет достаточно сложную задачу. Поэтому значительный практический интерес представляет оценка диагностических возможностей полного множества T_c

тестов одиночных неисправностей N_1 при контроле кратных неисправностей.

Обозначим через переменные a и b соответственно наличие связи входных столбцов x и y со строкой W. Тогда выражение для логического значения в строке имеет вид

$$W = (a + x)(b + y).$$

Одиночная неисправность, например, 0-связи (1-связи) соответствует наличию отказа типа константа 1 (константа 0) на линии a. Функция, реализуемая при исправной ПЛМ, на выходном столбце f определяется как

$$f = c_1 W_1 + c_2 W_2,$$

где W_1, W_2— входные переменные, формирующие выходной сигнал f; c_1, c_2— переменные, соответствующие наличию связи переменных W_1 и W_2 с выходом f.

Отказ 0-связи (1-связи) на линии W_1 соответствует отказу типа константа 1 (константа 0) на линии c_1.

Используя модели логических значений переменных для отдельных связей линии W_j и выходов f_k, можно определить общие выражения для выходов ПЛМ.

Обозначив $A_{i,j}$ связь между переменной x_j и строкой W_i, получим результирующее значение W_i исправной ПЛМ:

$$W_i = A_{i,1}A_{i,2} \ldots A_{i,n} = \bigwedge_{j=1}^{n} (a_{i,j} \vee \bar{x}_j)(b_{i,j} \vee x_j),$$

где i — номер строки ПЛМ; $\bar{x}_j$, x_j — инверсные и прямые значения входной переменной x_j; a_{ij}, b_{ij} — связи соответственно входных столбцов $\bar{x}_j$, x_j со строкой W_i ПЛМ.

Значение выходного столбца f_k, $1 \leqslant k \leqslant p$, при наличии связи $C_{k,i}$ со строками W_i

$$f_k = \bigvee_{i=1}^{m} C_{k,i} W_i = \bigvee_{i=1}^{m} C_{k,i} \bigwedge_{j=1}^{n} A_{ij}.$$

Отказ в ПЛМ выявляется при отличии получаемого значения f_k от значения f_k, определяемого данным выражением. Будем называть неисправность $0 \to 1$-диагностируемой ($1 \to 0$-диагностируемой) для заданного входного теста x_t, если значение исправной ПЛМ равно 0 (1), а при неисправной ПЛМ на выходе получаем значение 1 (0). При кратных неисправностях может происходить маскирование одних неисправностей другими. В частности, 0-связные неисправности в строке ПЛМ маскируются только 1-связными неисправностями в той же строке, а 1-связные неисправности маскируются только 0-связными неисправностями в той же строке.

Можно показать, что для описанной модели тесты T_c одиночных неисправностей позволяют выявлять двух-и трехкратные неисправности ПЛМ. Неисправности большей кратности не все обнаруживаются с помощью этих тестов. Как показал В. К. Агарвел [35], кратные неисправности обнаруживаются с помощью множества T_c тестов одиночных неисправностей, если кратная неисправность не содержит пар 0-связных и 1-связных неисправностей, расположенных в более чем одной строке. В частности, из общего числа $C^r_{m(2n+p)}$ различных связных неисправностей кратности 4 в безызбыточной ПЛМ наибольшее $C^2_m (n + p/2)^4$ неисправностей не выявляется с помощью множества T_c тестов одиночных неисправностей.

В более общем случае при $r > 4$ этим же автором показано, что наибольшее $C^2_m\ (n + p/2)^4\ C^{r-4}_{m(2n+p)-4}$ различных неисправностей не выявляется с помощью множества тестов одиночных неисправностей ПЛМ. Очевидно, что эта оценка верхней границы имеет смысл лишь для величин, меньших $C^r_{m(2n+p)}$.

Используя данные оценки, легко определить процент выявляемых с помощью множества тестов T_c кратных неисправностей в ПЛМ. В частности, при $m = 48$, $n = 16$, $p = 8$ выявляется более 98 % всех неисправностей кратности меньше или равной 8, что является достаточным при практическом применении большинства БИС.

Таким образом, используя достаточно простые наборы тестов, удается выявить бо́льшую часть неисправностей в ПЛМ.

Дальнейшее увеличение сложности ПЛМ, построенных на основе СБИС, существенно сказывается на глубине диагностирования. Невыявление небольшого процента кратных неисправностей в СБИС существенно сказывается на качестве диагностирования. Поэтому для повышения качества диагностирования применяют специальные схемные решения ПЛМ, улучшающие проверку их работоспособности. Выявление всех кратных неисправностей является необходимым условием эффективной эксплуатации таких ПЛМ.

Достаточно полная проверка может производиться с помощью сдвигающих регистров, подключенных к входным столбцам ПЛМ.

Перемещение единственной единицы в таком регистре позволяет выбирать необходимые столбцы ПЛМ.

Формирование тестов в этом случае сводится к заданию входных наборов тестов. Примером может служить множество тестовых наборов $A_{n,\,m,\,j}$, каждый из которых включает $2n + 1$ тестов. Полное множество $A_{n,\,m}$ тестов ПЛМ состоит из

подмножеств $A_{n,\,m,\,j}: A_{n,\,m} = \{A_{n,\,m,\,j} \mid j = 1,\ 2,\ \ldots,\ m\}$. Как показано в работе [39], полное множество тестов, включая $A_{n,\,m}$, состоит из $m(2n + 1) + 4n + 5$ тестов.

Важнейшим фактором, влияющим на применение структур взаимоконтроля в отказоустойчивых системах, является организация связей между отдельными парами элементов. Обычно, с целью сокращения аппаратурных затрат на такие связи, используют общие шины с разделением их по времени.

Захват общих шин теми или иными ЭМ производится с использованием специальных схем арбитража.

Поэтому возникает задача обеспечения правильного функционирования этих схем, поскольку наличие в них отказов может привести к невозможности дальнейшего контроля и диагностирования в системе. При этом необходимо учитывать, что отключение арбитров для проверки и диагностирования достаточно сложно реализовать.

Одним из наиболее эффективных подходов к диагностированию, с учетом этих особенностей, является применение самоконтролирующихся арбитров. Рассмотрим реализацию самоконтролируемого арбитра, описанную в работе [39].

Арбитр делится на задающее устройство и решающий блок. Показанный на рис. 30, *а* арбитр проверяет сигналы запросов с помощью соотношения

$$R = \sum_{i=1}^{n} R_i.$$

Введение обратной связи с выхода входного регистра позволяет также контролировать запрос согласно выражению

$$E = \sum_{i=1}^{n} \bar{R}_i R'_i,$$

где R_i, R'_i — соответственно поступившие и принятые запросы от ЭМ системы.

Задающее устройство реализует диаграмму переходов, показанную на рис. 30,*б*. По запросу R загружается входной регистр. В состоянии *3* вырабатывается сигнал DEC и в состоянии *4* выдается ответ на один запрос (сигнал LOR). Возврат в состояние *1* приводит к отключению выходного регистра (выдается сигнал COR очистки выходного регистра).

Функции, реализуемые решающим блоком арбитра, показаны на рис. 30, *в*. Функции самоконтроля в арбитре решаются за счет самоконтролируемости отдельных блоков, а также в целом по всем этим блокам.

Основными выявляемыми неисправностями являются константные неисправности входов и выходов функциональных

узлов (триггеров, вентилей, регистров и т. п.). Предполагается, что входы $R_1, \ldots, R_n$ и генераторы синхросигналов не имеют отказов.

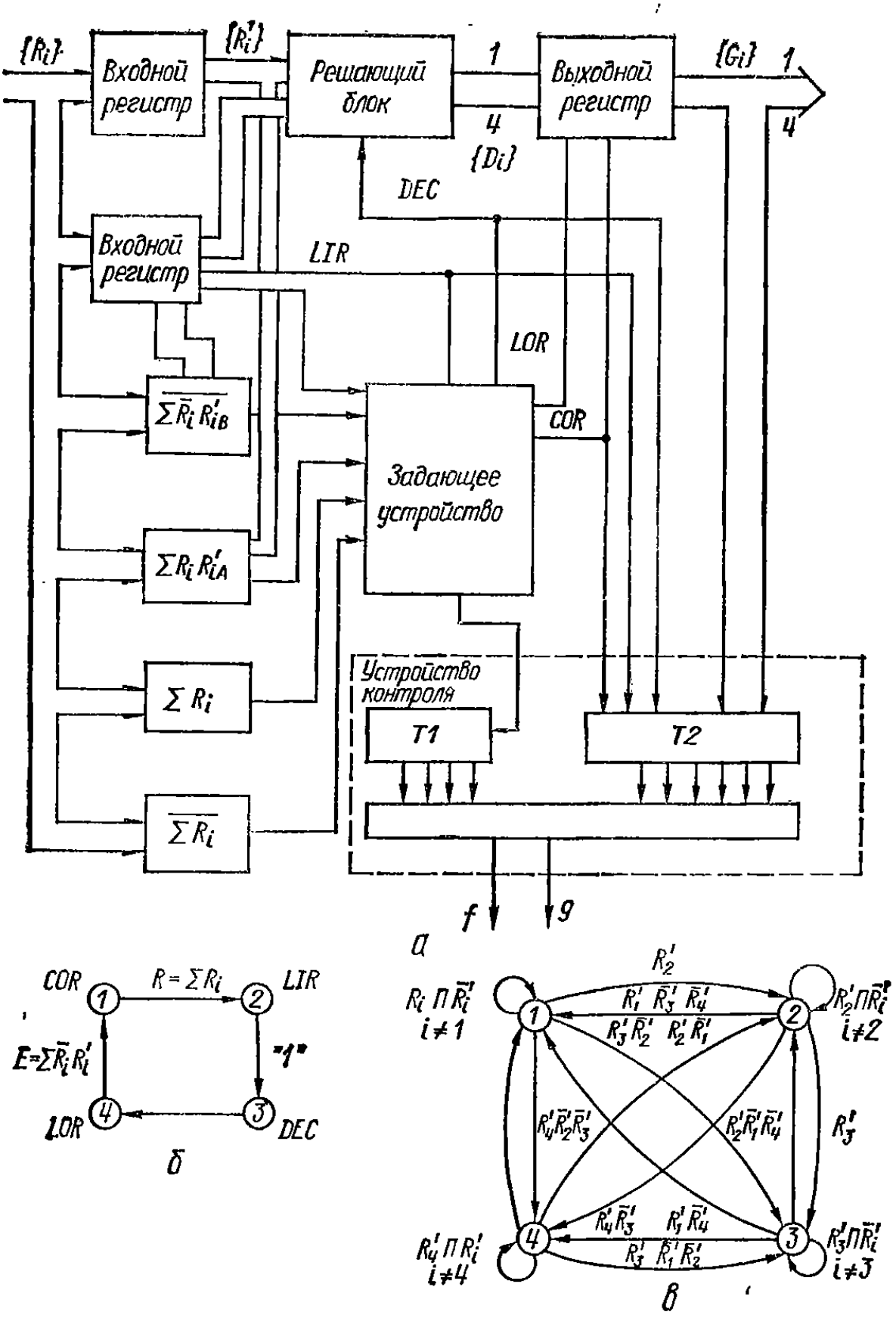


Рис. 30. Самоконтролируемый арбитр

При проверке задающего устройства используется контроль кодами 1 из 4. Входной регистр в целях проверки дублируется. Переключательные функции триггеров, входящих в задающее

устройство, имеют следующий вид:

$$y_1 = E_A Y_4 + R_B Y_1; \quad y_3 = Y_2;$$
$$y_2 = R_A Y_1; \quad y_4 = Y_3 + E_B Y_4.$$

В решающем блоке используются коды k из n, в частности, 1 из n. Это обеспечивает непосредственное получение выходных сигналов $\{D_i\}$ на основе внутренних переменных $\{Y_i\}$.

Согласно графу переходов (рис. 30, *в*) сигналы триггеров решающего блока определяются следующим образом:

$$y_1 = R_1' R_3' R_4' Y_2 + R_1' R_4' Y_3 + R_1' Y_4 + Y_1 \prod_{i \neq 1} \bar{R}_i;$$
$$y_2 = R_2' \bar{R}_1' \bar{R}_4' Y_3 + R_2' \bar{R}_1' Y_4 + R_2' Y_1 + Y_2 \prod_{i \neq 1} \bar{R}_i;$$
$$y_3 = R_3' \bar{R}_1' \bar{R}_2' Y_4 + R_3' \bar{R}_2' Y_1 + R_3' Y_2 + Y_3 \prod_{i \neq 3} \bar{R}_i;$$
$$y_4 = R_4' \bar{R}_2' \bar{R}_3' Y_1 + R_4' \bar{R}_3' Y_4 + R_4' Y_4' + Y_4 \prod_{i \neq 4} \bar{R}_i.$$

В арбитре используется самопроверяемое устройство контроля. Контроль осуществляется в кодах 1 из n. Анализ работы решающего блока и выходного регистра в устройстве контроля производится на основе анализа выходов $\{G_i\}$ выходного регистра, а также управляющих сигналов COR, LIR, DEC.

Практически их анализ сводится к проверке кода 2 из $n + 3$.

К числу неисправностей, которые не контролируются в арбитре, относится общее отсутствие ответа арбитра по запросу при блокировании синхронизации. Выявление этих неисправностей производится, например, использованием локальных контролируемых устройств для каждой ЭМ либо введением в состав арбитра централизованных схем проверки.

Повышение показателей надежности работы генератора синхросигналов достигается обычно за счет нагруженного резервирования этих цепей.

Таким образом, введение самоконтролирующих устройств позволяет производить функциональный контроль и диагностирование узлов отказоустойчивых систем, не охваченных СВК.

Запоминающие устройства составляют значительную часть (до 40—80 %) аппаратуры микропроцессорных систем. При переходе к ЗУ на СБИС объемом более 10^6 бит количество отказов в них возрастает. В результате велика удельная доля отказов, возникающих в ЗУ, по сравнению с другими устройствами системы. Поскольку именно ЗУ позволяют реализовать основные управляющие функции системы и обеспечивают обработку данных в системе, то влияние отказов может быть

более существенным, чем отказы других устройств. Поэтому задача своевременного обнаружения и устранения влияния дефектов в ЗУ является важным условием обеспечения отказоустойчивости микропроцессорной системы. В практике применения ЗУ больших объемов для повышения достоверности данных, хранимых и считываемых из ЗУ, наибольшее распространение получили различные способы введения информационной и аппаратурной избыточности и контроль ЗУ как в процессе функционирования, так и в автономном режиме проверки.

Введение избыточности осуществляется обычно применением избыточного кодирования (например, с использованием кодов Хемминга) и введением дополнительных корректирующих ЗУ меньшего объема по сравнению с исходным ЗУ. Введение корректирующих ЗУ наиболее целесообразно использовать для контроля и корректировки постоянных ЗУ, поскольку в процессе функционирования отказы происходят лишь в наибольшем по удельному весу количестве элементов ЗУ.

Применение постоянных и оперативных ЗУ требует решения вопросов их контроля и диагностирования. С увеличением степени интеграции ЗУ, представляющих БИС и СБИС, возрастает сложность их проверки. Выявление произвольных комбинаций отказов элементов в ЗУ сложно и нецелесообразно исходя из практических соображений. Обычно считается, что в ЗУ происходят отказы нескольких смежных элементов — групповые отказы. Это соответствует дефектам технологии изготовления этих устройств (например, ошибки в шаблонах или некачественное травление, напыление), а также отказам в процессе эксплуатации (например, пережигание отдельных участков кристалла и, как следствие, разрушение части ячеек ЗУ в локальной пространственной области кристалла).

В ряде работ [20, 21] используются модели для двухразмерных ЗУ. Рассмотрим одну из таких моделей для оперативных ЗУ и основанные на ней процедуры контроля и диагностирования. Предположим, что при чтении из одного элемента памяти содержимое других элементов не меняется. Окрестностью каждого элемента будем называть сам элемент и смежные с ним слева, справа, сверху и снизу запоминающие элементы.

В каждый цикл обращения к элементу памяти возможно запоминание в нем информации: 0 или 1, или динамическое переключение из нулевого состояния в единичное (обозначим это 0*) или из единичного в нулевое (обозначим это 1*). Назовем активным смежным набором ячейки, смежные некоторому элементу памяти и влияющие на ее состояние. Если ячейки C_i изменяют свое значение вследствие определенного сочета-

ния состояний смежных с ним элеметов, то ЗУ называют чувствительным к неисправностям активных смежных наборов элементов, а сами смежные элементы образуют активные смежные неисправности.

Обозначим через C_A, C_B, C_C, C_D соответственно элементы сверху, снизу, слева и справа от элемента C_E. Тогда, если, например, состояние набора $(C_A, C_B, C_C, C_D) = (0\ 0^*\ 0\ 1)$ влияет на состояние элемента C_E, то активные смежные наборы состояний будут $(C_A, C_B, C_C, C_D) = (0\ 0\ 0\ 1)$ и $(C_A, C_B, C_C, C_D) = (0\ 1\ 0\ 1)$. При диагностировании элементов памяти проверяется установка его в состояние 0 или 1.

В достаточно общем случае можно рассматривать изменение состояния контролируемого элемента памяти при изменении состояния одного из смежных с ним элементов (переход из 0 в 1 или из 1 в 0) и фиксированном значении других смежных элементов. Тогда в пределах каждого цикла общее количество состояний элементов памяти, смежных с рассматриваемым, равно 64. Общее число тестов для проверки элемента с учетом того, что он может быть в одном из двух фиксированных состояниях —0 или 1, равно 128.

Для контроля неисправностей активных смежных элементов необходимо 32 операции записи и 65 операций чтения, а минимальное количество операций на проверке каждого элемента памяти равно 97 [53]. Таким образом, контроль и диагностирование оперативных ЗУ требует достаточно больших тестовых наборов.

Хранение больших массивов тестовых наборов может существенно снизить эффект от диагностирования и контроля, поскольку в памяти, где хранятся эти массивы, возможны дефекты. Поэтому более предпочтительным является программная реализация тестовых наборов.

Предложенные в работе [53] алгоритмы диагностирования для ЗУ, состоящих из m столбцов и n строк, основаны на присвоении смежным элементам индексов, определенных их положением на пересечении столбцов и строчек ЗУ.

Обозначим $S_ч(S_н)$ множество элементов ЗУ, сумма индексов строк и столбцов которых четна (нечетна). Очевидно, что элементы из $S_н$ и $S_ч$ являются смежными.

Алгоритм для выявления всех активных смежных неисправностей:

1. Записать нули в элементы $S_ч$, затем нули в элементы из $S_н$. Прочитать нули из каждого элемента $S_ч$.

2.1. Установить $i = 1$.

2.2. Выполнить операцию i в $S_ч$ и прочитать все элементы из $S_н$.

2.3. $i = i + 1$.

2.4. Если $i = 65$, то перейти к 2,5, в противном случае — к 2.2.

2.5. $i = 1$.

2.6. Выполнить операцию i в $S_н$ и прочитать все элементы из $S_ч$.

2.7. $i = i + 1$.

2.8. Если $i = 65$, то перейти к шагу 3. В противном случае — к шагу 2.6.

3. Записать единицы в каждый элемент из $S_ч$.

4.1. $i = 1$.

4.2. Выполнить операцию i в $S_н$.

4.3. $i = i + 1$.

4.4 Если $i = 13$, то перейти к 4.5, в противном случае — к 4.2.

4.5. Прочитать единицу из каждого элемента в $S_ч$.

4.6. Выполнить операцию i в $S_ч$.

4.7. $i = i + 1$.

4.8. Если $i = 65$, то $i = 1$ и перейти к 4.6.

Если $i = 13$, то перейти к 4.9, в противном случае — к 4.6.

4.9. Прочитать единицу из каждого элемента $S_н$.

4.10. Выполнить операцию i над каждым элементом из $S_н$.

4.12. Если $i = 65$, то перейти к 4.13, в противном случае — к 4.10.

4.13. Прочитать единицу из каждого элемента в $S_ч$.

Таким образом, для реализации диагностирования неисправностей в ЗУ необходимо лишь хранение массивов $S_ч$ и $S_н$, а также самой программы диагностирования.

5. ПЕРСПЕКТИВЫ РАЗВИТИЯ ОТКАЗОУСТОЙЧИВЫХ СИСТЕМ

Дальнейшее развитие теории и практики создания отказоустойчивых систем тесно связано с совершенствованием элементной базы. Единая целевая комплексная программа использования микропроцессоров предусматривает создание, освоение в производстве и ввод в эксплуатацию систем и комплексов на базе микропроцессорных средств, в частности:

автоматизированных технологических комплексов;

систем и устройств автоматического управления и регулирования;

информационных систем;

предприятий, гибких переналаживаемых производств и систем управления для них;

систем для научных исследований, проектно-конструкторских работ, обучения и отладки микропроцессорной техники

измерительных систем, комплексов и приборов.

Обеспечение отказоустойчивости таких систем тесно связано с развитием системных принципов проектирования, совершенствованием технологии изготовления компонентов системы и расширением круга решаемых ими задач. Большое разнообразие требований, предъявляемых к функциональным возможностям, приводит к различным архитектурам отказоустойчивых систем.

Можно выделить следующие основные факторы, предопределяющие дальнейшее развитие архитектур отказоустойчивых систем:

1. Дальнейшее возрастающее использование большого количества (до 10^3—19^5) БИС и СБИС в различных компонентах систем.

2. Развитие принципов организации физических и программных связей между элементами системы.

3. Совершенствование структурной организации системы, обеспечивающей высокую степень параллельности обработки данных.

4. Новые подходы к построению математического обеспечения отказоустойчивых систем, направленных, в частности, на повышение их помехозащищенности, отказоустойчивости, устранение влияния различных видов ошибок.

5. Формирование нового взгляда на обслуживание систем, оценку их эффективности, надежности и достоверности функционирования.

Применение БИС и СБИС диктует определенные требования к архитектуре системы. В настоящее время большое внимание при построении БИС и СБИС уделяется вопросам их диагностируемости и самоконтролируемости. Высокая аппаратурная избыточность СБИС позволяет значительно повысить время наработки на отказ отдельных компонент системы.

В ближайшие годы ожидается увеличение емкости оперативной памяти до 8—16 Мбит, а микропроцессоры будут содержать до 10^5 и более вентилей на 1 мм^2 [2—7, 25, 26, 30].

Наиболее перспективным является использование ЗУ на полупроводниковых СБИС, магнитных доменах, ЗУ с зарядовой связью и электронно-лучевой памяти. В частности, использование электронно-лучевой памяти позволяет получить ЗУ объемом сотни мегабит при времени доступа 10—20 мкс.

Широкое распространение получают 16- и 32-разрядные микро-ЭВМ, что приводит к дальнейшему росту функциональных возможностей микропроцессорных систем, росту их производительности.

Использование способов повышения глубины диагностирования СБИС, их контролепригодности (например, распро-

страненный подход, состоящий в перестройке структуры БИС ЗУ в длинный сдвигающий регистр) обеспечивает своевременное выявление неисправностей в отдельных компонентах системы при отбраковочном контроле элементов и в процессе диагностирования при эксплуатации.

Можно ожидать, что по мере дальнейшего развития многомашинных систем сохранится подразделение на распределенные системы с малой удаленностью (десятки метров—километров), средней удаленностью (сотни — тысячи километров), удаленные системы (десятки тысяч километров) и глобальные системы на основе комплексного использования спутниковых систем связи и наземных подсистем. В связи с этим будут совершенствоваться существующие линии связи, пропускная способность которых составляет десятки килобит в секунду.

При создании высокопроизводительных локально-распределенных отказоустойчивых систем с малой удаленностью компонентов системы получают волоконно-оптические линии связи, обеспечивающие пропускную способность до сотен мегабит в секунду. Важными преимуществами таких систем является высокая помехозащищенность от внешних магнитных и электрических полей, надежность, стойкость к изменению температуры, давления, влажности окружающей среды и др.

Для сильносвязных отказоустойчивых систем необходима высокая разветвленность физических линий связи, что также достаточно просто достигается при использовании волоконно-оптических линий связи.

При создании подсистем связи распределенных систем для повышения пропускной способности каналов связи будет широко применяться разделение по частоте и по времени передаваемых сообщений, что обеспечит высокую пропускную способность этих каналов. Это потребует проведения дальнейших работ по освоению методов пакетной передачи данных, совершенствования протоколов обмена, организации взаимодействия новой системы с уже существующими [5].

Структурная организация системы имеет важное значение для обеспечения эффективности микропроцессорных многомашинных систем [2, 6, 8, 9—13, 20].

Многие из существующих систем имеют одну из перечисленных в гл. 1 архитектур. Так, например, распределенная мультимикропроцессорная система MuTEAM [43] состоит из групп элементов, причем взаимодействие достаточно просто осуществляется между элементами различных групп. Система имеет иерархическую структуру, диагностирование в ней осуществляется путем взаимодействия элементов между собой с использованием общей памяти, распределенной по элементам системы.

Шире будут применяться многомашинные системы с кольцевой архитектурой и архитектурой с разделением памяти, используемые в промышленных отказоустойчивых системах.

В перспективных глобальных системах архитектуры систем будут комбинированными, т. е. включать локальные подсистемы с различной архитектурой. Это позволяет в наибольшей степени удовлетворить конкретные требования обслуживаемых объектов и обмена данными на больших расстояниях. Например, при создании системы контроля за состоянием окружающей среды отдельные наземные подсистемы по сбору и обработке данных могут быть построены по кольцевой структуре (система типа С6), а обмен данными между этими подсистемами — по звездной структуре (система типа С8) через спутниковые системы.

Программное обеспечение сейчас занимает по удельному весу 50—90 % общей стоимости системы. Поэтому одной из основных задач можно считать разработку подходов к автоматизации программирования, средств выявления ошибок в ПО. Унификация и упрощение ПО за счет использования структурного программирования в сочетании с повышением отказоустойчивости ПО позволит создавать эффективные системы с высокой преемственностью математического обеспечения.

Большое значение при эксплуатации отказоустойчивых систем имеют протоколы обмена данными между элементами системы. Многие из существующих протоколов обмена ориентированы на поблочную передачу данных. Блоки обычно включают достаточно большое число разрядов контроля, что затрудняет их обработку при приеме. При некоторых реализациях систем, ориентированных, в частности, на непрерывную передачу данных, более оправданным является применение межблочных, сквозных избыточных кодов. Поэтому со временем большое распространение получат протоколы обмена, в которых будет применяться избыточное кодирование передаваемых потоков данных.

Быстрый рост объемов выпуска микропроцессоров, БИС и СБИС, создание многомашинных систем, построенных на их основе, ставит задачу эффективного обслуживания таких систем. Круг пользователей многомашинных систем, различающихся уровнем подготовки, расширяется. Вместе с тем превышение темпов роста количества микропроцессорных устройств и систем по сравнению с темпами роста обслуживающего их квалифицированного персонала требует разработки единых принципов обеспечения их контролепригодности с учетом рационального использования не только самих технических средств, но и затрат, связанных с их обслуживанием. Фактор

обслуживания становится одним из доминирующих факторов обеспечения эффективной работы систем.

Таким образом, комплекс перечисленных факторов, а также дальнейшее развитие единых принципов обеспечения устойчивости устройств и систем к возникающим отказам, позволит создавать высокопроизводительные отказоустойчивые системы с длительным временем автономного функционирования.

СПИСОК ЛИТЕРАТУРЫ

1. **Авиженис А.** Отказоустойчивость — свойство, обеспечивающее постоянную работоспособность цифровых систем.—Тр. ин-та инженеров по электротехнике и радиоэлектронике, 1978, т. 66, № 10, с. 5—15.

2. **Балашов Е. П., Пузанков Д. В.** Микропроцессоры и микропроцессорные системы — М.: Радио и связь, 1982.— 326 с.

3. **Баумс А. К., Гуртовцев А. Л., Зазнова Н. Е.** Микропроцессорные средства.— Рига: Зинатне, 1977.— 235 с.

4. **Брусенцов Н. П.** Миникомпьютеры.— М.: Наука, 1979.— 271 с.

5. **Вейцман К.** Распределенные системы мини- и микро-ЭВМ.— М.: Финансы и статистика, 1983.— 381 с.

6. **Высоконадежный** отказоустойчивый мультипроцессор для управления самолетом /А. Л. Хопкинс., С. Б. Смит., Д. Х. Лала и др. Тр. ин-та инженеров по электротехнике и радиоэлектронике, 1978, т. 66, № 10, с. 142—165.

7. **Гибсои Г., Лю-Ч.** Аппаратные и программные средства микроЭВМ.— М.: Финансы и статистика, 1983.— 255 с.

8. **Головкии Б. А.** Параллельные вычислительные системы.—М.: Наука, 1980.— 519 с.

9. **Денисенко О. С.** Анализ функциональных структур, описываемых иормальными формами.— Автоматика и вычислител. техника, 1985 г., № 1, с. 49—55.

10. **Димитриев Ю. К., Хорошевский В. Г.** Вычислительные системы из мини-ЭВМ.— М.: Радио и связь, 1982.— 304 с.

11. **Евреинов Э. В., Хорошевский В. Г.** Однородные вычислительные системы.— Новосибирск: Наука. Сиб. отд-ние, 1978.— 320 с.

12. **Исследование систем/Д.** П. Северек, В. Кини, Х. Мешбери и др.— Тр. ин-та инженеров по электротехнике и радиоэлектронике, 1978, т. 66, № 10, с. 89—141.

13. **Каляев А. В.** Однородные коммутирующие регистровые структуры.— М.: Сов. радио, 1978.— 334 с.

14. **Карпов Ю. Г., Толстяков А. В.** Аналитический метод верификации протоколов.— Автоматика и вычислител. техника, 1985, № 1, с. 35—41.

15. **Коваленко А. Е., Масленников В. П.** Об одной интерпретации алгоритмов диагноза с помощью временных логических уравнений.— М.: Труды/МВТУ, 1976, № 210, с. 36—38.

16. **Кривоногов Ю. А., Поздняк Г. Е., Саркисян В. И.** Микромашинные распределенные вычислительные системы.— К.: Вища шк. Головное изд-во, 1982.— 179 с.

17. **Микропроцессориые** БИС и микро-ЭВМ/А. А. Васеиков, Н. М. Воробьев, В. Л. Дшхунян и др.— М.: Сов. радио, 1980.— 279 с.

18. **Микро-ЭВМ** «Электроника С5» и их применение/М. П. Гальперин, В. Я. Кузнецов, Ю. А. Масленников и др. — М.: Сов. радио, 1980.— 157 с.

19. **Нечипоренко В. И.** Структурный анализ систем.— М.: Сов. радио, 1977.— 213 с.

20. **Отказоустойчивая** вычислительная система с тремя вычислительными машинами/Х. Ихара, К. Фукуока, Ю. Кубо и др. — Тр. ин-та инженеров по электротехнике и радиоэлектронике, 1978, с. 66, № 10, с. 68—89.

21. **Пархоменко П. П.** О технической диагностике.— М.: Знание, 1969 .— 64 с.

22. **Пархоменко П. П., Согомонян Е. С.** Основы технической диагностики.— М.: Энергоиздат, 1981.— 319 с.

23. **Петренко П. А., Теслер Г. С.** Обработка данных в вычислительных системах и сетях.— К.: Техніка, 1980.— 180 с.

24. **Пирс У.** Построение надежных вычислительных машин.— М.: Мир, 1968.— 270 с.

25. **Прангишвили И. В.** Микропроцессоры и микро-ЭВМ.— М.: Энергия, 1979.— 231 с.

26. **Пролейко В. М.** Микропроцессоры, микро-ЭВМ и их развитие.— Электрон. пром-сть, 1979, № 11—12, с. 3—6.

27. **Реннельс Д. А.** Архитектура космических бортовых вычислительных систем, устойчивых к отказам.— Тр. ин-та инженеров по электротехнике и радиоэлектронике, 1978, т. 66, № 10, с. 186—205.

28. **PLURIBUS**—отказоустойчивый операционный мультипроцессор/Д. Катсуки, Э. С. Эслам, У. Ф. Мани и др.— Тр. ин-та инженеров по электронике и радиоэлектронике, 1978, т. 66, № 10, с. 49—67.

29. **Самофалов К. Г., Луцкий Г. М.** Структура и организация функционирования ЭВМ и систем.— К.: Вища шк. Головное изд-во, 1978.— 391 с.

30. **Соучек Б.** Микропроцессоры и микро-ЭВМ.—М.: Сов. радио, 1979.— 520 с.

31. **Справочник** по цифровой вычислительной технике/Под ред. Б. Н. Малиновского.— К.: Техніка, 1980.— 320 с.

32. **Той В. Н.** Проектирование отказоустойчивых местных процессоров для систем электронной коммутации.— Тр. ин-та инженеров по электротехнике и радиоэлектронике, 1978. т. 66 , № 10, с. 26—68.

33. **Турута Е. Н.** Организация распределения задач в вычислительных системах, обеспечивающая их отказоустойчивость.—Автоматика и вычислител. техника, 1985, № 1, с. 5—14.

34. **Щураков В. В.** Надежность программного обеспечения систем обработки данных.— М.: Финансы и статистика, 1981.— 216 с.

35. **Agarvel V. K.** Multiple fault Detection in programmable logic arrays.— IEEE Trans., 1980, vol. C—29, N 6, p. 518—522.

36. **Allan T. J., Kameda J., Tolda S.** An Approach to the Diagnosability Analysis of a System.— IEEE Trans., 1975, vol. C—24, N 10, p. 1040—1042.

37. **BARSI T., Crandoni T., Maestrini P.** A theory of diagnosability of digital systems.— IEEE Trans., 1976, vol. C—25, N 6, p. 585—593.

38. **Ciompi P., Simoncini L.** Analysis and optimal design of self-diagnosable systems with repair.— IEEE Trans., 1979., vol. C—28, N 5, p. 362—365.

39. **Courvoisier M., Gettroy J. C., Seck J.** A self-testony arbiter circuit for multimicrocomputer systems. Proceedings of the FTCS.— 10, 1980, p. 281—283.

40. **Jornson J., Kinnie G., Maer M.** Tripple-bus architecture: spase—board microcomputer's CPU operate at full speed while other system components shave the main memory.— Electronic Design, 1978, vol. 26, N 15, p. 231—236.

41. **Fujiwara H., Kinoshita K.** Connections assignments for probabilistically diagnosable systems.— IEEE Trans, 1978. vol. C—27, N3, p. 280—283.

42. **Fujiwara H., Kinoshita K.** Some Existence theorems for probabilistically diagnosable systems.— IEEE Trans., 1978. v. C—27, N 4, p. 379—384.

43. **Grandoni F.,** u, a. The MuTEAM system: general guidelines.— Proceedings of the FTCS —11, 1981, p. 15—16.

44. **Hakimi C. L., Amin A. T.** Characterization of connection assignment of diagnosable systems.— IEEE Trans., 1974, vol. C—23, N 1, p. 86—88.

45. **Hlavicka J.** Design of a self-diagnosable module system.— Proceeding of the FTCS—11, 1981, p. 106—108.

46. **Kameda T., Toida J., Allan J.** A diagnosing algorithm for network.— Information and Control, 1975, vol. C—29, N 1, p. 141—148.

47. **Kawakubo K., Nakamura I., Okumura I.** The architecture of a fail—safe and fault-tolerant computer for railway signalling device. Proceeding of the FTCS—10, 1980, 372—373.

48. **Kime C. K.** An abstract model for digital system fault diagnosis.— IEEE Trans, 1980, vol. C—28, N 8, p. 754—767.

49. **Makam S. V., Avizienis A.** Modeling and analysis of periodically renewed closed fault—tolerant systems.— Proceedings of the FTCS—11, 1981, p. 134—141.

50. **Mine H., Hatayama K.** Performance evaluation of faulttolerant computing system. Proceedings of the FTCS—9, 1979, п. 59—62.

51. **Preparato F. P., Metze G., Chien R. J.** On the connection assignment problem at diagnosable systems.— IEEE Trans., 1967, vol. EC 16, N 6, p. 848—854.

52. **Russel J., Kime C.** System fault diagnosability without repair.— IEEE Trans., 1975, vol. C—24, N 11, p. 1078—1089.

53. **Sak D. S., Reddy S. M.** An algorithm to detect a class of pattern sensitive fanlts in semicanductor random access memories.— Proceedings of the FTCS. — 9, 1979, p. 219—226.

54. **Sauer A. M., Schmitter E.J.** The fault-tolerant microcomputer system BFS.— Proceedings of the FTCS—11, 1981, p. 252 c.

55. **Smith J. E.** Universal systems diagnosis algorithms.— IEEE Trans., 1979, vol. C—28, N 5, p. 374—378.

56. **Sridhar T., Hayes J. P.** A functional approach to testing bit — sliced microprocessors.— IEEE. Trans., 1981, vol. C —30, № 8, p. 563—571.

57. **Thevenod — Fosse P., David K.** Random testing of the data processing section of a microprocessor.— Proceedings of the FTCS—11, 1981, p. 275—280.

58. **Yajima S., Aramaki T.** Autonomously testable programmable logic arrays. — Proceedings of the FTCS — 11, 1981, p. 41—43.

ОГЛАВЛЕНИЕ

Анатолий Епифанович *Коваленко*, канд. техн. наук
Валентин Васильевич *Гула*

Отказоустойчивые микропроцессорные системы

Редактор Л. О. Полянская
Оформление художника С. И. Райхлина
Художественный редактор В. С. Шапошников
Технический редактор *С. М. Ткаченко*
Корректоры *Л. А. Москаленко, Н. В. Тарабан*

Информ. бланк № 1338

Сдано в набор 16.12.85. Подписано в печать 15.05.86. БФ 05280. Формат 84×108$^{1}/_{32}$. Бумага типогр. № 2. Гарн. лит. Печ. выс. Усл. печ. л. 7,98. Усл. кр.-отт. 8,3. Уч.-изд. л. 8,97. Тираж 20000 экз. Зак. 5-541 Цена 65 к.

Издательство «Техніка», 252601, Киев, 1, Крещатик, 5

Отпечатано с матриц Харьковской книжной фабрики им. М. В. Фрунзе на Харьковской книжной фабрике «Коммунист», 310012, Харьков-12, Энгельса, 11.