

Министерство образования и науки Российской Федерации

Владивостокский государственный университет
экономики и сервиса

Ю.В. МАЛЫШЕНКО
Л.Ф. СТЫЦЮРА
Ю.Л. САЯПИН

ТЕХНИЧЕСКАЯ ДИАГНОСТИКА

Учебное пособие

*Рекомендовано Дальневосточным
региональным учебно-методическим
центром (ДВ РУМЦ) в качестве учебного
пособия для студентов специальности
230101.65 «Вычислительные машины,
комплексы, системы и сети» вузов региона*

Владивосток
Издательство ВГУЭС
2010

УДК 681.31
ББК 32.97
М 20

Рецензенты: В.А. Бобков, д-р. техн. наук, профессор (Институт автоматике и процессов управления ДВО РАН);
А.А. Дыда, д-р. техн. наук, профессор (Морской государственной университет им. Невельского)

Мальшенко, Ю.В., Стыцюра, Л.Ф., Саяпин, Ю.Л.
М 20 ТЕХНИЧЕСКАЯ ДИАГНОСТИКА [Текст] : учебное пособие / под общ. ред. Ю.В. Мальшенко. – Владивосток : Изд-во ВГУЭС, 2010. – 302 с.

ISBN 978-5-9736-0163-8

Учебное пособие разработано в соответствии с программой курса, а также требованиями образовательного стандарта России к учебной дисциплине «Техническая диагностика». Рассматриваются вопросы диагностирования электронных, в том числе программно-управляемых устройств.

Для студентов, изучающих дисциплины, связанные с проектированием и эксплуатацией электронных изделий.

ББК 32.97

ISBN 978-5-9736-0163-8

© Издательство Владивостокский государственный университет экономики и сервиса, 2010

ВВЕДЕНИЕ

Теоретические основы технической диагностики начали формироваться с середины 50-х годов прошлого столетия. Как и любому научному направлению, технической диагностике присущи особенные задачи, которые и отличают ее от других дисциплин, – это построение тестов, формализация процессов определения места неисправности, синтез схем контроля, проектирование контролепригодных и отказоустойчивых устройств и др.

Сегодня можно утверждать, что техническая диагностика – одно из научных направлений, в котором результаты большинства теоретических разработок находят применение в практике проектирования, производства и эксплуатации электронных изделий.

Проблемы, возникающие в диагностике, можно проследить на примере развития средств диагностирования вычислительных машин.

За прошедшие годы несколько раз менялась идеология построения тестов и вообще организации диагностирования. Многие из первых машин обслуживались опытными техниками и инженерами, которые досконально знали структуру и устройство ЭВМ. В своей работе они использовали встроенные схемы контроля, вольтметры и осциллографы. Для обеспечения поиска неисправностей использовались порой двухпроцессорные машины, в которых два одинаковых процессора работали параллельно, а результат сравнивался схемами сравнения. В случае появления рассогласования оба процессора моментально останавливались, и обслуживающий персонал, используя ручные операции, осуществлял поиск фактической неисправности и ремонт ЭВМ. В дополнение к аппаратуре при этом в ограниченном объеме применялись программы-тесты. Это были простейшие программы, которые зацикливали определенные машинные команды. В результате инженер мог наблюдать сигналы в заданных контрольных точках, но их интерпретация осуществлялась вручную на основе знаний схемотехники и конструкции ЭВМ.

В связи с ростом сложности ЭВМ неудовлетворенность в использовании такого подхода становилась все более ощутимой. Поиск неисправности занимал порой дни, и появилась сильная зависимость работы ЭВМ от опыта обслуживающего персонала. По этим причинам возрасла роль диагностических программ и встроенных средств контроля, которые превратились в основной инструмент отыскания неисправности. Самые первые программы были написаны для проверки функций машины. При функциональном подходе проверяемый блок рассматривался как «черный ящик», о структуре которого ничего не известно. Известна только функция, им выполняемая. В этой связи возникали вопросы,

насколько можно доверять результатам функционального контроля, какова полнота такой проверки. Очевидно, что для полной проверки цифровой системы необходимо проверить все ее внутренние состояния и выходные сигналы при всех возможных комбинациях входных переменных. Вообще, если объект имеет n входов и m простейших элементов памяти, то для его полной функциональной проверки требуется, как минимум, 2^{n+m} входных воздействий. Так, например, для БИС, имеющей 25 входов и 50 элементов памяти, необходимо 2^{75} входных наборов. Если предположить, что частота подачи входных воздействий равна 1 МГц, то на проверку потребуется $3,8 \cdot 10^{20}$ сек $\approx 1 \cdot 10^9$ лет, то есть 1 млрд лет. Таким образом, для проверки даже одного из первых микропроцессоров Intel 8080 требуется фантастическое время 10^{20} лет!!! Конечно, современная техника позволяет проводить тестирование в несколько тысяч раз быстрее, но и современный микропроцессор содержит десятки-сотни тысяч элементов памяти, а сама микросхема с процессором содержит сотни входов-выходов.

Данные примеры наглядно показывают, что полной функциональной проверке на практике могут подвергаться лишь достаточно простые схемы.

Ключ к дальнейшему усовершенствованию методов технической диагностики лежал в использовании тестов для проверки аппаратуры машины, а не ее функций. Этот подход, называемый сегодня структурным или элементным, был предложен советскими учеными-математиками С.В. Яблонским и И.А. Чегис (1955) и получил широкое развитие в 60–70-е гг. прошлого века. При таком подходе заранее определяются наиболее вероятные неисправности и задача контроля формулируется не как задача проверки функционирования объекта, а как задача проверки отсутствия (наличия) неисправностей из заданного множества. Объект рассматривался на уровне достаточно простых первичных логических элементов (вентилей) и проверка строилась таким образом, чтобы неисправности любого из элементов отразились на изменении выходных сигналов объекта диагностирования. Главное достоинство элементного подхода заключается в том, что для достижения контроля хорошего качества требуется гораздо меньше входных воздействий по сравнению с функциональным подходом и достаточно просто численно (и достоверно) оценить качество тестов. Например, для полной функциональной проверки 32-х разрядного сумматора требуется 2^{64} входных наборов. Однако при рассмотрении сумматора на структурном уровне и ограничении возможных неисправностей классом константных требуется не более 2^8 тестовых наборов.

В принципе, любое устройство можно представить на вентиляльном уровне, однако использование такой структурной модели сегодня не всегда эффективно из-за сложности реальных устройств (например, БИС содержит десятки-сотни тысяч вентиляей на одном кристалле). По этой причине внимание диагностиков в 80-е годы вновь обратилось в сторону функциональных методов, но на несколько иной модели объекта диагностирования. Объект представляется как соединение более крупных элементов (механизмов, функций). При этом нет четкой привязки элементов модели к структуре объекта, и при построении теста ставится задача проверить каждый механизм (функцию).

Каждый раз, когда появлялись новые модели, это приводило к новому всплеску в развитии теории технической диагностики, так как требовалось создавать методы применительно к новым моделям.

Усложнение схем ЭВМ, возложение на них все более ответственных функций повысило требование к такому свойству электронных изделий, как надежность. Оказалось, что необходимое повышение показателей надежности вычислительной техники невозможно обеспечить только за счет технологических решений. Поэтому в рамках технической диагностики стали развиваться направления, связанные с конструированием контролепригодных и отказоустойчивых устройств, а также встроенных самопроверяемых схем контроля.

Учитывая обширность рассматриваемых тем и учебное назначение издания, мы при изложении некоторых методов ставили задачу показать только общую методологию решения без исчерпывающего обсуждения деталей и нюансов, многие из которых до сих пор являются предметом научных исследований. Пособие предназначено для студентов, изучающих дисциплины, связанные с проектированием и эксплуатацией электронных изделий.

Первый раздел, по сути, является введением в дисциплину, в котором даются основные понятия и общая характеристика основных задач диагностирования.

Обзор основных формальных моделей объектов диагностирования и их неисправностей дается во втором разделе. Рассматриваются диагностические модели как цифровых, так и аналоговых объектов диагностирования. Особо выделены модели описания функционирования объекта (исправного и его неисправных модификаций) в виде таблиц функций неисправностей и таблиц неисправностей. Здесь же обсуждаются понятия проверяемости и различимости неисправностей.

Вопросы оценки качества тестов и результатов диагностирования обсуждаются в третьем разделе. Приводятся оценки, рекомендуемые стандартами. Для практики особо важны оценки полноты и достоверности контроля, глубины поиска неисправностей. Формальные вычисления последних рассмотрены особенно подробно, даются рекомендации по их применению в зависимости от используемой технологии определения места неисправности и восстановления объекта диагностирования. В этом же разделе рассмотрены вопросы анализа тестов цифровых схем на состязания, который является составной частью всех систем автоматизации построения тестов. Состязания являются следствием задержек, возникающих при распространении электрических сигналов в схеме. Эти задержки заранее неизвестны, могут меняться при изменении внешних факторов. В результате возможна неоднозначность в поведении объекта диагностирования, что существенно усложняет процессы диагностирования.

Четвертый раздел посвящен важнейшей и одной из наиболее сложных задач технической диагностики – построению тестов. Здесь даются понятия тестов: контролирующего, диагностического, тупикового, а также общие понятия о существующих подходах к синтезу тестов. Фактически цель задач этого типа – определить последовательность входных воздействий, которые надо подать на внешние входы объекта диагностирования, чтобы обеспечить нужное качество проверки или точность указания места неисправностей. В разделе рассматриваются конкретные методы построения тестов, получивших наибольшее применение в практических системах автоматизации их построения.

Особенности внутрисхемного диагностирования обсуждаются в пятом разделе. В системах поэлементного диагностирования поочередно проверяются все элементы и линии связи. Причем при тестировании компонента необходимо исключить

влияние параллельных цепей. Например, при проверке резистора надо определить именно его сопротивление, а не сопротивление цепи, к которой он подключен.

Внутрисхемное тестирование выполняется после установки компонентов на плату и давно применяется в ходе серийного производства электронных изделий в качестве надежного метода проверки правильности установки элементов на печатной плате и оценки параметров установленных элементов. С помощью методов и средств внутрисхемного диагностирования можно локализовать неисправности при ремонте электронных модулей, не имея их принципиальных схем (например, часто отсутствуют схемы на импортное оборудование). В разделе рассматриваются не только теоретические аспекты методов внутрисхемного диагностирования и современные высокоавтоматизированные системы их реализации, но и особенности применения простейших приборов (вольтметров, амперметров) для измерения параметров резисторов, конденсаторов, диодов, транзисторов и др.

Любые измерения имеют погрешность, поэтому авторы сочли необходимым в одном из подразделов кратко обсудить понятие «погрешность» и дать базовые формулы для ее вычисления.

Сегодня, кажется, уже нет такой области применения, где бы не использовались электронные устройства с микропроцессором. Электрическая схема даже простейшего микропроцессора состоит из тысяч, десятков тысяч логических элементов с большим числом элементов памяти. Уже поэтому задачи диагностирования объектов, содержащих микропроцессор, относятся к наиболее сложным в технической диагностике. Вопросам организации диагностирования и технического обслуживания микропроцессорных устройств, в том числе ПЭВМ, посвящен шестой раздел пособия.

Особенности программно-управляемых устройств, к которым относятся микропроцессорные устройства, инициировали создание специальных средств диагностирования (логические и сигнатурные анализаторы, эмуляторы, диагностические платы и др.), принципы функционирования и особенности конструкций которых также рассмотрены в шестом разделе.

Высокие надежностные характеристики современных электронных устройств невозможно обеспечить, если при проектировании его схемы и конструкции не задумываться о их приспособленности к решению задач диагностирования. Вопросам контролепригодного проектирования посвящен седьмой раздел учебного пособия. Здесь обсуждаются основные критерии ее оценки, прикладные методы обеспечения высокой контролепригодности при проектировании электронных изделий как с использованием аппаратных решений, так и различных методов кодирования обрабатываемой информации.

Восьмой раздел учебного пособия посвящен диагностированию программ. Многие современные электронные изделия относятся к программно-управляемым и их правильная работа предполагает отсутствие дефектов в управляющих и обрабатывающих программах. К тому же при отказе в работе программно-управляемого устройства зачастую трудно определить в чем причина: дефект в аппаратуре или программе. Несмотря на существенные достижения в области проектирования программных продуктов, решение задач их тестирования до сих пор во многом неформальный процесс. Наиболее трудоемким и обязательным является тестирование уже созданной программы. Поэтому в разделе основное внимание уделено вопросам тестирования уже разработанных программ.

Тема 1. ОСНОВНЫЕ ПОНЯТИЯ И ЗАДАЧИ ТЕХНИЧЕСКОЙ ДИАГНОСТИКИ

1.1. Технические состояния, надежность и задачи диагностирования

Виды технических состояний и задачи диагностирования

Диагностика в переводе с греческого «диагнозис» означает распознавание, определение.

Согласно ГОСТ 20911-89 **техническая диагностика** определяется как «область знаний, охватывающая теорию, методы и средства определения технического состояния объектов».

Объект, состояние которого определяется, называют **объектом диагностирования** (ОД). Диагностирование представляет собой процесс исследования ОД. Характерными примерами результатов диагностирования состояния технического объекта являются заключения вида: ОД исправен, неисправен, в объекте имеется такая-то неисправность.

Целью технического диагностирования изделий является поддержание установленного уровня надежности, обеспечение требований безопасности и эффективности использования изделий (по ГОСТ 27518-93).

В стандартах исправное, неисправное, работоспособное и неработоспособное технические состояния определяются следующим образом (ГОСТ 20911-89).

Исправное состояние – состояние объекта, при котором он соответствует всем требованиям нормативно-технической и конструкторской документации.

Неисправное состояние – состояние объекта, при котором он не соответствует хотя бы одному из требований нормативно-технической и (или) конструкторской документации.

Работоспособное состояние – состояние объекта, при котором значения всех параметров, характеризующих способность выполнять заданные функции, соответствуют требованиям нормативно-технической и (или) конструкторской документации.

Неработоспособное состояние – состояние объекта, при котором значение хотя бы одного параметра, характеризующего способность выполнять заданные функции, не соответствует требованиям нормативно-технической и (или) конструкторской документации.

Техническое диагностирование обычно выполняется при производстве, эксплуатации и ремонте изделий. В ходе его решаются следующие задачи (по ГОСТ 27518-93):

- определение вида технического состояния;
- поиск места отказа или неисправностей;
- прогнозирование технического состояния.

В процессе производства, эксплуатации и хранения объектов в них могут появляться и накапливаться неисправности. Некоторые из них приводят к тому, что объект перестает отвечать предъявляемым к нему техническим требованиям. Перед использованием объекта по назначению необходимо знать, есть ли в нем неисправности, которые могут явиться причиной нарушения нормальной работы. С ответом на этот вопрос связан процесс обнаружения неисправности, т.е. определения вида технического состояния. Он детализируется в зависимости от режима и особенностей использования объекта. В соответствии с этим выделяются следующие базовые задачи обнаружения неисправности:

1) **проверка исправности**, целью которой является разбраковка, позволяющая отделить исправные изделия от неисправных. ОД исправен, если он удовлетворяет всем техническим требованиям. Проверка исправности объекта – наиболее полная задача, при которой выносится решение о том, что в объекте не содержится никакой неисправности. Такая проверка обычно выполняется на этапе производства объекта и из всех задач контроля – она наиболее сложная;

2) **проверка работоспособности**, целью которой является выяснение, будет ли объект выполнять те функции, для реализации которых он создан. Такая проверка зачастую применяется, например, при профилактическом обслуживании объекта;

3) **проверка правильности функционирования**, целью которой является обнаружение неисправностей, которые нарушают правильную работу объекта, применяемого по назначению, в данный момент времени. Такая проверка ОД не является полной, т.к. в объекте может существовать неисправность, которая в данный момент времени не влияет на его функционирование, т.е. правильно функционирующий объект может быть неработоспособным.

Исправный ОД всегда является работоспособным и правильно функционирующим. Работоспособный ОД всегда правильно функционирует. Однако обратные утверждения не всегда могут выполняться. Так, радиоприемник может не работать при приеме на длинных волнах, однако вполне правильно функционировать на коротких. Иными словами, правильно функционирующий объект может быть неработоспособным и неисправным, а работоспособный – исправным.

Исправное и все неисправные технические состояния образуют **множество технических состояний** ОД. Рисунок 1.1 иллюстрирует характер разбиения множества технических состояний при решении различных задач технического диагностирования (0 – исправное и х – неисправное техническое состояние).

Если объект неисправен, то для замены или ремонта неисправных компонентов необходимо установить место неисправности.

Рисунок 1.1 иллюстрирует разбиение технических состояний в зависимости от решаемой задачи (проверка исправности, работоспособности, правильности функционирования или поиск неисправностей).

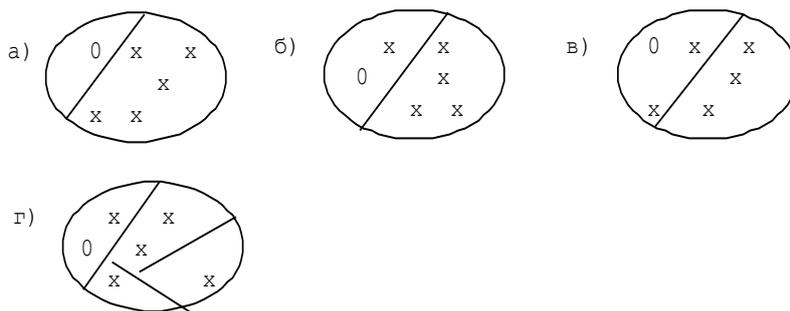


Рис. 1.1. Разбиение технических состояний при решении задач диагностирования

Так, при проверке исправности (рис. 1.1а) надо уметь отделять объекты с исправным техническим состоянием от неисправных; при проверке работоспособности надо отделять состояния, в которых объект исправен или работоспособен (рис. 1.1б), от других; при поиске неисправности – выделять конкретную неисправность или группу неразличимых неисправностей (рис. 1.1г).

При организации диагностирования, как правило, сначала решается задача проверки (контроля). Если результат отрицательный, то переходят к поиску места конкретной неисправности, имеющей место в ОД (рис. 1.2).

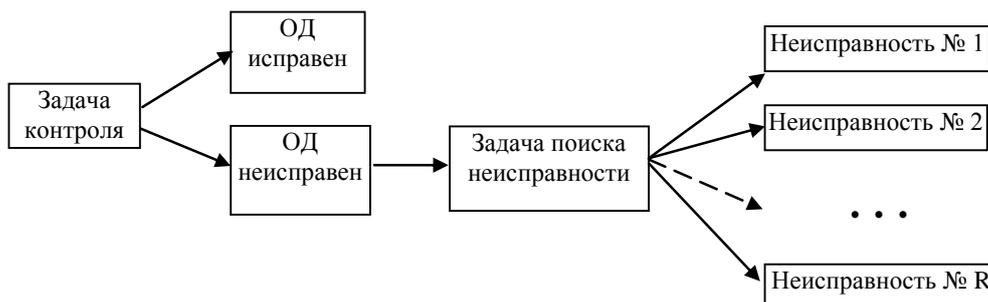


Рис. 1.2. Общая стратегия диагностирования

Процесс проверки и поиска неисправности в общем случае состоит из отдельных частей, каждая из которых связана с подачей на объект входного воздействия (тестового или рабочего) и измерением выходной реакции объекта. Такие части называют **элементарными проверками**.

В ходе выполнения элементарной проверки ответы могут сниматься с основных выходов ОД, т.е. с выходов, необходимых для применения ОД по назначению, и с **дополнительных выходов**, создаваемых специально для организа-

ции диагностирования. Основные и дополнительные выходы называют **контрольными точками** (КТ) или **контролируемыми выходами**. Измеряемые на них параметры называют **контролируемыми** или **диагностическими параметрами**. В одной КТ может измеряться несколько параметров. Например, при контроле сигнала синусоидальной формы часто измеряют одновременно частоту и амплитуду сигнала.

Воздействия формируются специальными средствами либо определяются непосредственно алгоритмом функционирования ОД. **Последовательность воздействий**, поступающих на ОД для целей диагностирования, называют **тестом**.

Поиск неисправности обычно предполагает подачу некоторых воздействий на ОД и дешифрирование его результатов.

Дешифрирование результатов диагностического эксперимента направлено на определение неисправностей, наличие каждой из которых в объекте не противоречит его реальному поведению в процессе выполнения диагностического эксперимента. Такие неисправности включаются в **список подозреваемых неисправностей (СПН)**. Кроме фактической в СПН могут попасть и неисправности, которые неразличимы с фактической при исходных условиях диагностирования. Поэтому далее стараются, используя различные способы, выделить в СПН фактическую неисправность.

Для целей контроля (проверки) и поиска неисправностей обычно используют разные тесты.

Кроме задач проверки и поиска места неисправности возможна задача **прогнозирования неисправностей**. Так, со временем обычно меняются параметры многих элементов электронных схем, что в итоге приводит объект в неработоспособное состояние. Если периодически измерять и анализировать изменения параметров элементов, можно примерно предсказать момент перехода из работоспособного в неработоспособное состояние. Возможен и применяется, например в авиации, способ прогнозирования, когда выделяется объект «лидер». Он имеет повышенную наработку по отношению к другим подобным объектам. Если в нем возникают неисправности, то можно предположить, что аналогичные неисправности могут со временем появиться и в остальных подобных объектах.

Для осуществления технического диагностирования необходимо:

- установить показатели и характеристики диагностирования;
- обеспечить приспособленность изделия к техническому диагностированию;
- разработать диагностическое обеспечение изделия.

Диагностическое обеспечение – комплекс взаимосвязанных диагностических параметров, методов, правил, средств диагностирования, указанных в технической документации, необходимых для осуществления диагностирования изделия.

Диагностическое обеспечение должно включать (по ГОСТ 27518-93):

- номенклатуру диагностических параметров и их характеристик (номинальные, допускаемые значения, точки ввода, точки контроля и т.д.);
- методы диагностирования;
- средства технического диагностирования;
- правила диагностирования.

С увеличением сложности ОД резко растут расходы на диагностирование, если при проектировании, производстве и организации технического обслуживания не учитывать требований диагностирования.

Надежность и ее свойства

Среди показателей качества продукции очень важное место отводится свойству «**надежность**».

Надежность – свойство объекта сохранять во времени в установленных пределах значения всех параметров, характеризующих способность выполнять требуемые функции в заданных режимах и условиях применения, технического обслуживания, хранения и транспортирования.

Надежность является комплексным свойством, которое в зависимости от назначения объекта и условий его применения может включать **безотказность, долговечность, ремонтпригодность и сохраняемость** или определенные сочетания этих свойств.

По ГОСТ 27002-89 они определяются следующим образом.

Безотказность – свойство объекта непрерывно сохранять работоспособное состояние в течение некоторого времени или наработки.

Долговечность – свойство объекта сохранять работоспособное состояние до наступления предельного состояния при установленной системе технического обслуживания и ремонта.

Ремонтпригодность – свойство объекта, заключающееся в приспособленности к поддержанию и восстановлению работоспособного состояния путем технического обслуживания и ремонта.

Сохраняемость – свойство объекта сохранять в заданных пределах значения параметров, характеризующих способности объекта выполнять требуемые функции, в течение и после хранения и (или) транспортирования.

Неполнота обнаружения и неточность в определении места неисправности ухудшает фактические показатели всех вышеуказанных свойств надежности.

Каждое свойство надежности характеризуется некоторыми количественными показателями.

Так, одним из показателей безотказности является **средняя наработка до отказа**, которая численно равна математическому ожиданию времени работы объекта до отказа (**отказ** – событие, заключающееся в нарушении работоспособного состояния объекта). Производной от этой величины является **интенсивность отказов** $\lambda = 1/T$, где T – среднее время наработки на отказ. Еще одним показателем безотказности является **вероятность безотказной работы**, т.е. вероятность того, что в пределах заданной наработки отказ не возникнет.

Современные персональные ЭВМ могут иметь время наработки до отказа несколько лет.

Ремонтпригодность обычно оценивают **средним значением времени восстановления работоспособности** после отказа и/или вероятностью того, что время восстановления работоспособного состояния объекта не превысит заданное значение (**вероятность восстановления**).

При оценке надежности применяют также комплексные показатели надежности, которые количественно характеризуют не менее двух свойств, составляющих надежность. Так, сочетание свойств безотказности и ремонтпригодности с учетом системы технического обслуживания и ремонта называют **готовностью объекта** (availability) и характеризуют **коэффициентом готовности**:

$$K_r = T / (T + T_b),$$

где T – средняя наработка на отказ, T_b – среднее время восстановления.

Статистическую оценку средней наработки на отказ вычисляют по формуле

$$T = t / r(t),$$

где t – время наработки (т.е. суммарное время работы объекта); $r(t)$ – число отказов, фактически происшедших за суммарную наработку t . Коэффициент готовности современных серверов не менее 0,99 (при этом T_b менее 4 суток в год).

Применение методов и средств технической диагностики влияет на показатели многих показателей надежности.

В частности, ряд показателей зависит от времени восстановления работоспособности ОД, которое, в свою очередь, зависит от времени обнаружения и точности указания места неисправности. Так, в коэффициенте готовности K_r среднее время восстановления $T_b = T_n + T_p$, где T_n – время поиска неисправности, T_p – время ремонта. Применение методов и средств диагностирования позволяет резко сократить это время и тем самым увеличить значение K_r .

Некоторые неисправности сразу не приводят к отказу в работе электронного устройства, но резко снижают его показатели надежности (например неисправности, влияющие на температурный режим). Если их вовремя не обнаружить и не устранить, начинают изменяться параметры элементов ОД, в объекте могут прийти в неисправное состояние сразу несколько элементов, что увеличит время и иные затраты на его восстановление. В ряде случаев, обнаружение и устранение неисправности можно выполнять в периоды, когда объект не используется по назначению. Учитывая это обстоятельство, можно организовать техническое обслуживание таким образом, чтобы повысить некоторые показатели надежности, например вероятность безотказной работы.

Достижение высоких показателей надежности современных электронных устройств невозможно без учета требований технической диагностики. В частности, в связи с быстро растущей сложностью ОД активно развиваются методы проектирования схем, обеспечивающие хорошую контролепригодность.

Контролепригодность (приспособленность объекта к диагностированию) – свойство объекта, характеризующее его пригодность к проведению диагностирования (контроля) заданными средствами диагностирования (контроля).

Если ОД хорошо приспособлен для диагностирования, то существенно упрощается построение качественных тестов и поиска места неисправности, что в итоге приводит к снижению времени диагностирования.

Еще одно из направлений технической диагностики – обеспечение отказоустойчивости. **Отказоустойчивость** – возможность компенсации (блокирования) неисправностей. Если объект обладает отказоустойчивостью, то при появлении неисправности он остается работоспособным. Очевидно, отказоустойчи-

вый объект обладает повышенной наработкой на отказ, т.е. повышенной надежностью.

На интуитивном уровне отказоустойчивость компьютерной системы можно интерпретировать следующим образом: допускается, что в системе могут появляться неисправности (сбои из-за помех или катастрофические отказы ее компонентов), но она продолжает безостановочно выполнять свою работу и при этом вырабатывать абсолютно достоверную информацию, а для пользователя и его задачи эти неисправности прозрачны, т.е. ни в чем не ощутимы.

На языке технических требований это означает, что, во-первых, отказ и, следовательно, ошибка в обработке данных должны обнаруживаться в машинном такте появления неисправности, источник ее должен быть локализован в том же такте, а ошибка подавлена (замаскирована). При этом условия гарантируется исключение потерь и искажений информации (сохранение целостности данных), а также предотвращение распространения ошибки в системе при появлении любой неисправности. Во-вторых, при неисправности любого компонента системы выполнение приложения на функциональном модуле с неисправным компонентом не должно прерываться. Этим гарантируется непрерывное выполнение любого приложения без обычных потерь на восстановление системы и процессов. В-третьих, при отказе время выполнения приложения не должно увеличиваться, т.е. необходимо, чтобы потери времени для приложения были равны нулю и, следовательно, эффективная производительность системы не должна уменьшаться. И наконец, замена неисправного компонента системы, ее модификация с целью устранения влияния неисправности не должны прерывать выполнение приложений, т.е. эти операции должны осуществляться в режиме непрерывной обработки данных (on-line обработка).

1.2. Средства и системы диагностирования

Виды средств диагностирования

Диагностирование осуществляется с помощью тех или иных **средств диагностирования (СД)**.

Выделяют встроенные и внешние СД.

Встроенное средство диагностирования (контроля) – средство диагностирования (контроля), являющееся составной частью объекта.

Внешнее средство диагностирования – средство диагностирования (контроля), выполненное конструктивно отдельно от объекта.

Внешние приборы и системы, предназначенные для выполнения диагностирования, часто называют **тестерами**.

Основные функции средств диагностирования – организация подачи воздействий (тестов), измерения контролируемых параметров, индикация результатов измерения и/или диагностирования. Поэтому, в общем случае, в их состав входят источники воздействий, измерительные устройства (или узлы), устройства коммутации источников воздействий и измерительных устройств с объектом, устройства индикации (светодиоды, сегментные жидкокристаллические дисплеи, мониторы и т.п.). Для управления средствами диагностирования и

анализа измеренной реакции ОД часто применяют вычислительные устройства. В современных системах это могут быть специализированные микропроцессоры или универсальные ПЭВМ.

Системы тестового и функционального диагностирования

Взаимодействующие между собой ОД и СД образуют **систему диагностирования**.

Различают системы **тестового** и **функционального диагностирования**. Особенность первых состоит в возможности подачи на ОД специально организованных (тестовых) воздействий от средств ОД. В системах второго типа диагностирование ведется на рабочих воздействиях, предусмотренных рабочим алгоритмом функционирования ОД.

На рисунке 1.3 приведены обобщенные функциональные схемы систем тестового (рис. 1.3а) и функционального диагностирования (рис. 1.3б) соответственно.

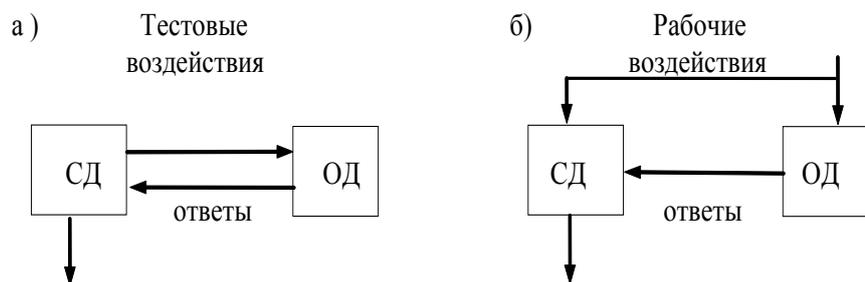


Рис. 1.3. Обобщенные схемы систем диагностирования

Системы тестового диагностирования могут применяться на всех этапах жизненного цикла ОД (в частности, при производстве и ремонте, хранении, перед каждым применением по назначению); функциональные обычно обеспечивают диагностирование в процессе применения ОД по назначению. Основная функция последних – контроль правильности функционирования ОД в процессе выполнения им его функций и, если это возможно, указание вида ошибки либо неправильно работающего узла.

Системы функционального диагностирования являются, как правило, встроенными и создаются одновременно с ОД. Они могут осуществлять диагностирование как непрерывно, так и периодически или эпизодически.

В технической диагностике электронных устройств различают **аналоговые** (непрерывные), **цифровые**¹ и **аналого-цифровые** (гибридные) ОД. В аналоговых ОД сигналы характеризуются континуальным множеством значений, в цифровых – логическими уровнями (их обозначают «лог. 1» и «лог. 0»), в аналого-цифровых – имеются сигналы обоих типов. Очевидно, что используемые в сис-

¹ Цифровые устройства являются частным случаем дискретных, в которых сигналы принимают значения из некоторого конечного ряда значений. На практике дискретные устройства встречаются редко.

темах диагностирования СД и средства подготовки информационного обеспечения зависят от вида ОД.

При большом объеме контрольно-диагностических операций (например, в условиях серийного производства или на специализированных предприятиях по ремонту) используемые системы тестового диагностирования обычно управляются от ПЭВМ. Основные составляющие систем тестового диагностирования показаны на рис. 1.4.



Рис. 1.4. Структурная схема современной системы тестового диагностирования

Прикладное программное обеспечение таких систем условно можно разделить на две части: программы подготовки информационного обеспечения и программы управления процессами диагностирования (включая обработку результатов диагностирования).

Для работы систем диагностирования необходимо заранее подготовить некоторые данные (информационное обеспечение), причем в удобной для последующего использования форме. Обычно они готовятся заранее, до проведения процесса диагностирования. Это могут быть, например, результаты моделирования поведения исправного (эталон) и неисправного ОД, параметры тестовых воздействий, значения допусков на аналоговые контролируемые параметры, электрическая схема ОД, тестовые задачи и др. Их качественное и быстрое получение невозможно без использования вычислительной техники и специальных программных средств моделирования. Современные системы автоматизации проектирования, так называемые САПР, включают подсистемы подготовки информации для диагностирования.

Вторую часть программных средств составляют программы, обеспечивающие управление собственно процессом диагностирования. Они реализуются на языке управляющей ЭВМ. Основные функции – организация подачи тестовых воздействий и измерение контролируемых параметров с помощью соответст-

вующих технических средств, обеспечение диалога «оператор – ЭВМ» в ходе диагностирования, определение технического состояния ОД по результатам выполнения проверок и т.д.

Классификация по принципам организации диагностирования

Современные системы диагностирования очень разнообразны по принципам работы, контролируемым параметрам, измерительным средствам, программному обеспечению и др.

До сих пор инженеры при ручном диагностировании применяют такие традиционные средства измерения, как тестеры сопротивлений и емкостей, осциллографы. Последние разработки таких приборов содержат микропроцессоры, что позволило существенно расширить функциональные возможности приборов.

Существуют автоматизированные системы, в которых при диагностировании доступны только внешние входы и выходы ОД. Их иногда называют системами **функционального** контроля¹, так как с их помощью фактически проверяют функцию объекта диагностирования.

Многие системы позволяют организовать **внутрисхемное диагностирование**, когда возможно измерять параметры каждого подозреваемого на неисправность компонента (резистора, микросхемы, трансформатора и др.).

Есть системы **визуального** и **рентгеновского** контроля.

Некоторые СД позволяют измерять **тепловое поле** и определять место неисправности путем сравнения измерений с эталонными термограммами.

Для контроля высокочастотных устройств могут применяться **рефлектометрические** методы тестирования. Дорожку в такой плате уже нельзя рассматривать только как проводник. Наряду с обычным сопротивлением, в ней необходимо контролировать и волновое (импеданс), которое связано с временными характеристиками распространения сигнала по проводнику.

Существуют специальные системы для **контроля** (прозвонки) **кабелей**.

Многие ОД содержат микропроцессоры и микросхемы (модули) памяти большого объема. Их диагностирование особенно сложная задача. Для таких объектов созданы специальные средства диагностирования: сигнатурные и логические анализаторы, эмуляторы, диагностические и отладочные платы и др. При проектировании подобных объектов широко применяются встроенные схемы самодиагностирования.

Следует сказать, что поиск неисправности в современном электронном объекте – это задача, которую специалисты относят к требующим использования при автоматизации методов искусственного интеллекта. Поэтому для анализа результатов измерений в некоторых случаях используют экспертные системы.

¹ В общей теории технической диагностики системы функционального диагностирования предполагают подачу только рабочих воздействий. Однако производители электронной аппаратуры под системами функционального контроля (диагностирования) обычно понимают любые системы, которые производят проверку уже собранных электронных изделий или узлов во включенном состоянии через внешние разъемы и/или органы управления объектом контроля.

1.3. Использование различных видов систем диагностирования в процессах производства электронных изделий

Ни одна из систем автоматизации диагностирования не позволяет обнаруживать 100% дефектов выпускаемого электронного изделия. Поэтому необходимы стратегии диагностирования и разумное комплексирование систем диагностирования различных типов, обеспечивающие быстрое выявление и устранение максимального числа дефектов при производстве.

Обычно при производстве электронных изделий используется несколько различных систем в зависимости от этапа производства. Это могут быть системы автоматического оптического или рентгеновского контроля, внутрисхемного контроля с подвижными пробниками и др. На рисунке 1.5 показаны этапы (слева направо) производственного процесса по изготовлению печатной платы и виды систем диагностирования, задействованных на том или ином этапе.



Рис. 1.5. Производственная линия изготовления и диагностирования печатной платы

Системы оптического контроля позволяют оценить качество нанесения паяльной пасты и выполнения пайки, правильность установки компонент на плату. Современные системы данного типа, кроме обычных увеличительных линз, могут иметь высокоскоростную цифровую цветную камеру с большим разрешением, высокоскоростной интерфейс связи с компьютером. Обычно имеют подсветку, состоящую из нескольких источников. Для проверки качества пайки или правильности установки элемента в современных системах оптического контроля сканированный образ (возможно при разных источниках освеще-

щения) просматривается оператором или автоматически сравнивается компьютером с его аналогом из поставляемой библиотеки.

Рентгеновский контроль, при котором объект просвечивается рентгеновскими лучами, возможно, в нескольких плоскостях, позволяет выявить скрытые дефекты. Это, например, случаи, когда пайка находится под некоторым элементом и недоступна для визуального просмотра. Обычно системы такого контроля применяются после сборки и пайки.

Системы внутрисхемного (поэлементного) контроля ориентированы на проверку обрывов и замыканий проводников, правильности установки и исправности отдельных элементов, установленных на проверяемой печатной плате. Применяются уже после монтажа элементов на печатной плате.

Сначала проводят измерения при отключенном питании объекта. Напряжения и токи, необходимые для проверки, подаются от средств диагностирования. На следующей стадии проверки может включаться питание объекта. Средства диагностирования подключаются непосредственно к выводам проверяемых элементов с помощью специальных контактных приспособлений.

Системы анализа производственных дефектов – это, по существу, те же системы внутрисхемного контроля, но повышенной производительности, с базой данных большой емкости и с развитыми средствами статистического анализа.

Под **системами комбинированного тестирования** понимают системы контроля, которые применяются как для проверки отдельных элементов объекта диагностирования, так и в целом.

Функциональное тестирование. Являясь завершающим шагом производственного процесса, служит для приемки или отбраковки готовых электронных блоков перед их отправкой заказчику. Тестеры для функционального тестирования обычно подключаются к тестируемым изделиям через краевой разъем или специально предназначенные для диагностирования контрольные точки. Далее симулируются электрические условия, в которых изделие будет находиться при конечном использовании.

Тестовое оборудование, применяемое при функциональном тестировании, должно выполнять следующие задачи:

- подача питающего напряжения с возможностью изменения его в автоматическом режиме (от минимально до максимально допустимого для данного изделия);
- подача цифровых и аналоговых входных сигналов (тестов) в широком диапазоне частот и напряжений;
- измерение параметров выходных сигналов;
- эмуляция нагрузок и внешних воздействий;
- эмуляция помех;
- обработка результатов измерений и вывод их на дисплей или принтер в удобном для пользователя виде;
- накопление и обработка статистической информации.

Обычно при функциональном тестировании проверяется только общая функциональная пригодность платы. При повышенных требованиях к надежности функциональное диагностирование может проводиться многократно в спе-

циальных камерах, создающих соответствующие механические, температурные и иные воздействия на объект диагностирования.

Функциональные тестеры, как правило, имеют модульную архитектуру. Их конфигурация, функциональные возможности и программные средства могут меняться в зависимости от типа объекта и требований заказчика.

Вышерассмотренные виды систем диагностирования могут конструироваться специально под отдельные типы объектов диагностирования.

Например, для измерений в ходе проверки высокочастотных устройств применяются **рефлектометрические методы**. Дорожку в такой печатной плате нельзя рассматривать только как проводник. Наряду с обычным сопротивлением, в таком проводнике необходимо контролировать и волновое (импеданс). Волновое сопротивление измеряется рефлектометрическим методом, то есть измерением временных характеристик отраженного импульса. Поэтому для контроля высокочастотных устройств используется специальное оборудование, реализующее рефлектометрический метод измерений.

В таблице 1.1 приведена одна из оценок эффективности применения различных видов систем автоматизации диагностирования.

Таблица 1.1

Что проверяет	HVI	API	AOI	AXI	FPT	MDA	ICT	FT
Короткое замыкание	2	1	2	1	2	2	2	2
Обрывы	3	3	2	1	2	2	2	2
Непропой	3	2	3	1	3	3	3	3
Не тот компонент	3	3	1	3	1	1	1	2
Негодный компонент	3	3	3	3	2	2	1	1
Пропущенный компонент	1	3	1	1	1	1	1	2
Неверная ориентация	2	3	1	2	1	1	1	2
Функциональные ошибки	3	3	3	3	3	3	3	1

В таблице приняты следующие обозначения:

HVI – неавтоматический визуальный контроль,

API – автоматический контроль паяльной пасты,

AOI – автоматический оптический контроль,

AXI – автоматический рентгеноскопический контроль,

FPT – диагностирование с помощью подвижных пробников,

MDA – анализ производственных дефектов,

ICT – внутрисхемное тестирование,

FT – диагностирование функций.

В клетках табл. 1.1 проставлены числа, определяющие эффективность применения систем диагностирования соответствующего типа: **1** – очень хорошо, **2** – хорошо, **3** – плохо.

Контрольные вопросы

1. Дайте определение понятия «техническая диагностика».
2. Какие возможны виды технических состояний?
3. В чем отличие исправного технического состояния от работоспособного?
4. Что понимают под элементарной проверкой?
5. Перечислите четыре основных свойства надежности.
6. Дайте определения безотказности и ремонтпригодности.
7. По какой формуле вычисляется коэффициент готовности? Что он характеризует?
8. Дайте определения контролепригодности и отказоустойчивости.
9. В чем различие тестовых и функциональных систем диагностирования?
10. Виды систем диагностирования, применяемые при производстве электронных изделий.

Рекомендуемая литература

ГОСТ 20911-89. Техническая диагностика. Термины и определения. – М.: Изд-во стандартов, 1990.

ГОСТ 27002-89. Надежность в технике. Основные понятия. Термины и определения. – М.: Изд-во стандартов, 1990.

Карибский, В.В. Основы технической диагностики / В.В. Карибский, П.П. Пархоменко, Е.С. Согомонян, В.Ф. Халчев. Кн. 1. – М.: Энергия, 1976. – 464 с.

Малышенко, Ю.В. Автоматизация диагностирования электронных устройств / Ю.В. Малышенко, В.П. Чипулис, С.Г. Шаршунов. – М.: Энергоатомиздат, 1986. – 216 с.

Марков, И. Автоматическое тестовое оборудование с подвижными пробниками в производстве электронных изделий / И. Марков, И. Рыков // Компоненты и технологии. – 2005. – № 1.

www.sovtest.ru

Тема 2. МОДЕЛИ ОБЪЕКТОВ И НЕИСПРАВНОСТЕЙ

2.1. Модели исправных объектов

Общая классификация моделей

Формальное исследование задач диагностирования предполагает наличие формального описания – модели ОД. При этом многие постановки и решения важнейших задач технической диагностики предполагают также задание множества допустимых неисправностей и их модели.

Если построена формальная модель ОД, то можно разработать формальные (в том числе математические) методы решения различных задач диагностирования.

Модель ОД может быть задана в явном или неявном виде.

Явная модель ОД представляет собой совокупность формальных описаний исправного объекта и всех (точнее, каждой из рассматриваемых) его неисправных модификаций. Для удобства обработки все указанные описания желательно иметь в одной и той же форме.

Неявная модель ОД содержит какое-либо одно формальное описание объекта и правила получения по этим данным всех других интересующих нас описаний. Чаще всего заданной является математическая модель исправного объекта, по которой можно построить модели его неисправных модификаций.

Общие требования к моделям исправного объекта, а также к моделям неисправностей состоят в том, что они должны с требуемой точностью описывать представляемые ими объекты и их неисправности. В неявных моделях объектов диагностирования модели неисправностей, кроме того, должны удовлетворять требованию удобства их «сопряжения» с имеющимся описанием объекта и тем самым обеспечить достаточно простые правила получения других описаний объекта.

Современная концепция проектирования предполагает использование, в случае создания больших и сложных схем, моделей нескольких уровней. При этом на каждом уровне должны рассматриваться и задачи диагностирования. Однако в наиболее полном и проработанном варианте задачи технической диагностики решаются, когда уже выбрана элементная база электронной части создаваемого изделия и имеются ее схемы.

При решении задач технической диагностики могут использоваться самые разнообразные модели ОД. В инженерной практике исходное описание ОД обычно представлено в виде электрической принципиальной или функциональной схем. Правила их графического изображения оговорены в соответствующих стандартах. Далее это описание может трансформироваться с учетом требований средств автоматизации, используемых для решения задач технической диагностики.

Любое электронное устройство состоит из связанных между собой компонент (блоков, элементов). Состав компонент, связи между ними и с внешними входами составляют **структуру** устройства.

С точки зрения «глубины и точности» представления внутренней структуры ОД **выделяют функциональные, структурные и структурно-функциональные классы модели.**

К первым относятся такие, которые описывают общую функцию ОД на уровне внешних входов/выходов и не содержат сведений о его структуре. К моделям такого типа, в частности, относится модель типа «черный ящик» (рис. 2.1). Такое название обусловлено тем, что ОД представляется некоторым блоком (закрытым «ящиком»), доступ к которому возможен только через внешние входы и выходы, причем известна функциональная зависимость, позволяющая для поступающих на его входы воздействий определить сигналы на внешних выходах. К функциональным следует отнести и модели, элементами которых являются «функции». При этом нет четкой привязки «функции» к структурным элементам ОД.

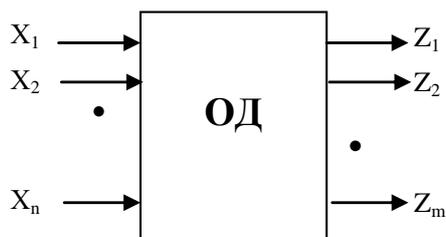


Рис. 2.1. Модель «черный ящик»

Для структурных моделей характерно описание, содержащее подробную информацию о конструктивных элементах, из которых построен ОД, и связях между ними. Так, модель, содержащая информацию об электрической схеме соединений микросхем или логических элементов в объекте и их функциях, относится, с точки зрения задач диагностики, к структурной (рис. 2.2). Часто такую модель называют логической схемой или логической сетью. Основное требование к структурной модели – она должна отражать структуру ОД, по крайней мере, на уровне заменяемых при ремонте элементов (узлов).

Одной из форм представления моделей является граф. Вершине графа сопоставляется элемент (узел, блок, функция) ОД, его дугам – связи между входами и выходами элементов с внешними входами/выходами схемы.

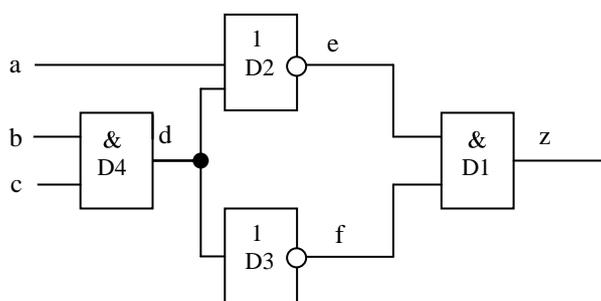


Рис. 2.2. Пример структурной модели

Существует классификация логических схем по виду графа. Если он не имеет разветвлений, такая схема называется древовидной. Для них наиболее просто решаются задачи диагностирования, но, к сожалению, они редко встречаются на практике. Наиболее сложными с точки зрения решения задач диагностирования являются схемы, граф которых содержит контуры обратной связи. Такой граф присущ схемам с памятью.

Структурно-функциональная модель предполагает представление ОД в виде функционально законченных узлов (сумматор, преобразователь, контроллер, регистр и др.) и связей между ними. При этом сам узел физически может представлять собой соединение некоторой группы конкретных компонент (например несколько микросхем). Иными словами, функционально законченный узел – это схема из нескольких заменяемых элементов, реализующая конкретную функцию (например аналого-цифровой преобразователь). В структурно-функциональной модели каждый узел – это «черный ящик», для которого неизвестна его внутренняя структура, но задана реализуемая функция. Иногда такого типа модели называют «серый ящик».

Конкретные модели, используемые в технической диагностике, очень разнообразны с точки зрения применяемого формального аппарата. Они могут представлять из себя: систему логических или дифференциальных уравнений, граф, таблицу, систему матриц, электрическую или функциональную схему, спецификацию, граф регистровых передач и др. При этом, к какому классу относится выбранная модель, во многом зависит от целей диагностирования. Так, создателей микросхемы обычно интересует, какой транзистор, резистор и т.п. явился причиной отказа. В этом случае, очевидно, структурной можно считать модель в виде электрической схемы на уровне транзисторов, резисторов, конденсаторов и т.п. Разработчику печатной платы в процессе диагностирования нужно определять неисправную микросхему, установленную на плате, и его не интересует, что произошло внутри микросхемы (т.е. какой конкретно транзистор ее логической схемы имеет дефект). Поэтому для его целей электрическая схема, отражающая типы установленных микросхем и их связи, является структурной моделью.

Объекты диагностирования подразделяются на **комбинационные и последовательностные** (т.е. с элементами памяти). В первых выходные установившиеся сигналы определяются исключительно значениями сигналов на внешних входах. Примером комбинационного ОД является схема на рис. 2.2. Во вторых

выходные сигналы зависят не только от входных, но и предшествующих состояний элементов памяти. Так, простейшим элементом памяти является триггер. Для некоторых реальных элементов памяти под состоянием можно понимать значения на их выходах. На рисунке 2.5 приведен триггер и его таблица истинности. Нетрудно заметить, что при подаче на его входы RS комбинации 00, выходные сигналы будут иметь значения 01 или 10 в зависимости от предыдущих выходных сигналов.

Типичными последовательностными устройствами являются счетчики, регистры. Они представляют собой несколько последовательно соединенных триггеров. Можно утверждать, что основные проблемы технической диагностики электронных устройств связаны с наличием в ОД элементов памяти.

Рассмотрим наиболее часто применяемые виды формальных моделей ОД.

Конечные автоматы

В теоретических исследованиях некоторых задач технической диагностики используют модель цифрового объекта диагностирования в виде конечного автомата. Слово «конечный» отражает то, что объект имеет конечное число вариантов выходных сигналов, в том числе выходных сигналов (состояний) элементов памяти. Автоматная модель полезна при изучении ряда теоретических проблем. Первоначально задачи технической диагностики электронных устройств изучались на этой модели, однако считается, что в инженерной практике ее использование не эффективно.

В абстрактном смысле **автомат** – «черный ящик», перерабатывающий множество входных цифровых сигналов в множество выходных сигналов. В общем виде конечный автомат представляется (рис. 2.3) моделью, основными компонентами которой является комбинационная часть и элементы памяти (ЭП).

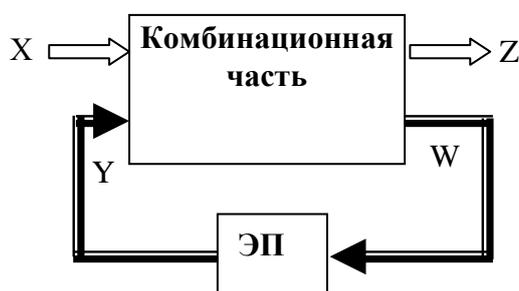


Рис. 2.3. Обобщенная схема конечного автомата

В представленной на рис. 2.3 схеме через X обозначено множество входных сигналов ($X=x_1, x_2, \dots, x_n$), Z – множество выходных сигналов ($Z=z_1, z_2, \dots, z_m$), W – входные сигналы элементов памяти, Y – выходные сигналы элементов памяти. Элементы памяти отличаются от комбинационных тем, что их выходные сигналы зависят не только от входных сигналов устройства, но и от предыдущих состояний элементов памяти.

Обычно считается, что элементы, из которых построена комбинационная часть и сами элементы памяти, являются безынерционными, а вносимые запаздывания в изменения сигналов представляются задержками, находящимися в линиях обратной связи между этими двумя частями модели. Иными словами, сигналы Y появляются с некоторой задержкой по отношению к сигналам W . Поведение элементов памяти определяется их предшествующим состоянием (на рис. 2.3 это набор значений Y).

В современной теории автоматов рассматривается целый ряд различных вариантов автоматов. Если говорить о схемах реальных объектов, то, как правило, они подпадают под определение автомата Мили. При этом считается, что ОД функционирует в дискретном времени, т.е. значения входных/выходных сигналов и состояния элементов памяти изменяются периодически, в некоторые моменты времени – такты.

Например, в одной из первых отечественных монографий давалось следующее определение автомата¹: «Конечная динамическая система называется **конечным автоматом**, если состояние системы в каждый такт однозначно определяется:

- а) состоянием системы в предыдущий такт,
- б) входом в предыдущий или в рассматриваемый такт».

В математической форме функционирование конечного автомата Мили можно описать в виде системы формул:

$$\begin{aligned} Z_t &= f_z(X_t, Y_t), \\ Y_{t+1} &= f_Y(W_t, Y_t), \\ W_t &= f_W(X_t, Y_t). \end{aligned}$$

Здесь X_t , Z_t и Y_t (Y_{t+1}) – значения входных, выходных сигналов и состояния элементов памяти в момент (такт) времени t ($t+1$). Индекс ($t+1$) введен для того, чтобы показать, что новое состояние элементов памяти зависит от предыдущего, которое было в момент поступления входного воздействия X_t .

Существуют два удобных способа описания автоматов:

- **таблица переходов**, дающая табличное представление функции Y_{t+1} ;
- **диаграмма переходов**, которая представляет собой граф.

Строки **таблицы переходов** сопоставлены возможным комбинациям входных сигналов X , столбцы состояниям Y автомата. В клетке ij таблицы проставляются через символ «/» следующее состояние и комбинация значений выходных сигналов, при подаче входной комбинации X_i и предположении, что память объекта перед этим находилась в состоянии Y_j .

Пример таблицы переходов приведен в табл. 2.1. Автомат, представленный таблицей, может иметь: два разных входных воздействия (X_1 и X_2); пять разных состояний, условно обозначенных цифрами 1, 2, ..., 5; два разных набора выходных сигналов (Z_1 и Z_2).

¹ Айзерман М.А. и др. Логика. Автоматы. Алгоритмы. М.: Физматгиз, 1963. – 556 с.

Пример таблицы переходов

Входные сигналы	Состояние				
	1	2	3	4	5
X_1	$1/Z_1$	$1/Z_1$	$5/Z_1$	$3/Z_2$	$2/Z_2$
X_2	$4/Z_2$	$5/Z_2$	$1/Z_2$	$4/Z_2$	$5/Z_2$

Диаграмма (граф) переходов представляет собой ориентированный граф (орграф), в котором вершины – это состояния автомата, а дуги (ориентированные рёбра) – это переходы между состояниями. Над дугой, направленной от состояния (вершины) i к состоянию j через символ «/», указываются комбинации значений входные сигналы/выходные сигналы (X/Z), соответствующие рассматриваемому переходу. На рисунке 2.4 каждое состояние изображается кружком с номером состояния.

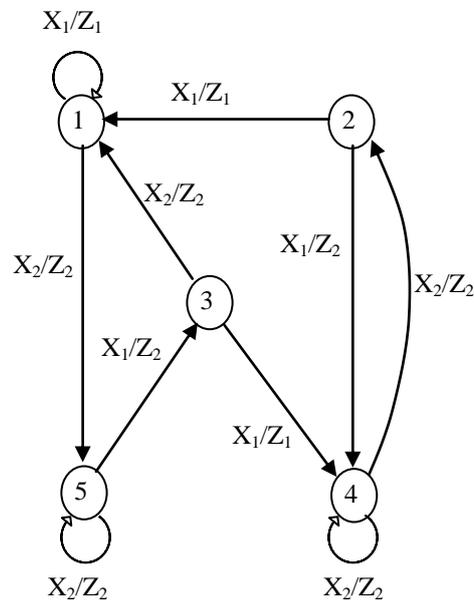


Рис. 2.4. Граф переходов к примеру табл. 2.1

Если, например, за исходное взять состояние 3, то направления дуг из вершины 3 и надписи над ними показывают, что при воздействии X_1 на выходе будет комбинация Z_1 и ОД перейдет в состояние 4, а при воздействии X_2 на выходе будет комбинация Z_2 и ОД перейдет в состояние 1. В то же время переход в состояние 3 возможен только из состояния 5, если на входы подать комбинацию X_1 . Некоторые дуги возвращаются в ту же вершину, из которой вышли. Это оз-

начает, что при соответствующем входном воздействии перехода в новое состояние не происходит.

Рассмотрим, как будет выглядеть таблица и граф переходов для реального устройства, например для RS-триггера, собранного на двух элементах ИЛИ-НЕ (рис. 2.5).

В данной схеме два входа R и S, два выхода – α и β . Значения последних одновременно определяют возможные состояния устройства. Таким образом, в рассматриваемом примере возможно 4 разных комбинации входных сигналов, выходных и состояний 00, 01, 10, 11 (рис. 2.5).

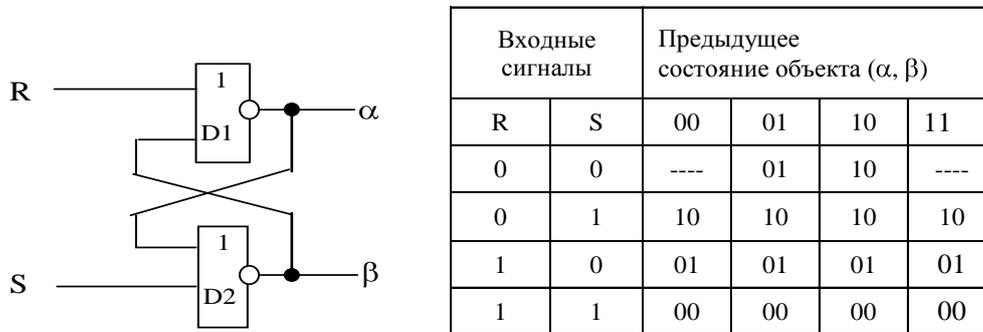


Рис. 2.5. Схема RS-триггера и его таблица переходов

В нашем примере выходные сигналы совпадают с сигналами, определяющими состояние объекта. Поэтому в клетках таблицы переходов проставляется только одна пара значений. В двух клетках таблицы проставлены прочерки. Обычно они ставятся, когда некоторое состояние является недостижимым или входное воздействие недопустимым.

Состояние Z_i называется **недостижимым**, если не существует входного воздействия X_j и состояния $Z_k \neq Z_i$, из которого автомат при подаче X_j переходит в Z_i . Рассматривая таблицу, мы видим, что недостижимым является состояние $\alpha=1, \beta=1$.

Входное воздействие называется **недопустимым**, если при его подаче возможно неоднозначное поведение автомата. Допустим, в нашем примере, первоначально на входах была комбинация «11», и триггер находился в состоянии "00". Если теперь подать на входы комбинацию сигналов «00», то ОД может перейти как в состояние «01», так и в «10» в зависимости от того, насколько одновременно изменяются входные сигналы и в каком элементе больше величина задержки. Так, если сигнал на входе S изменится раньше, чем на R, то триггер перейдет в состояние $\alpha' = 0, \beta' = 1$. Если же ранее изменится сигнал на входе R, то триггер перейдет в состояние $\alpha' = 1, \beta' = 0$. Для такого состояния и входного воздействия в таблице также ставится прочерк.

Граф переходов, соответствующий таблице переходов на рис. 2.5, представлен на рис. 2.6.

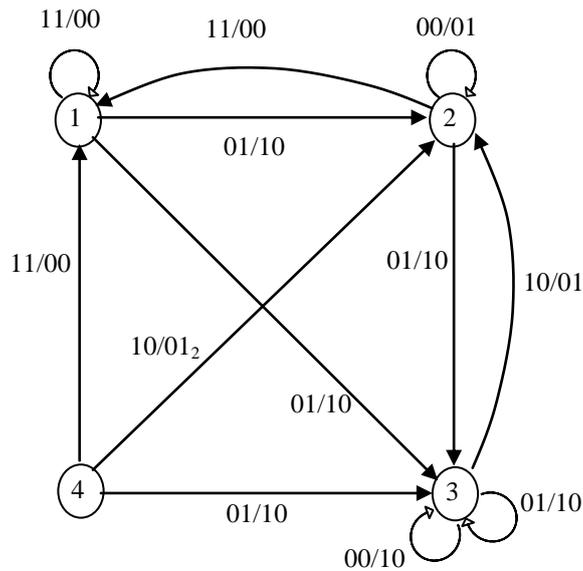


Рис. 2.6. Граф переходов для RS-триггера

Описание объекта системой уравнений непосредственных связей

В системах автоматизации, используемых для решения задач диагностирования, довольно широко используются модели, представляющие собой описание ОД в виде системы уравнений. Так, в случае цифровых ОД могут использоваться модели в виде системы уравнений непосредственных связей или эквивалентной нормальной формы. Перед этим ОД надо представить в виде логической схемы, например как на рис. 2.2.

Чтобы представить ОД в виде системы уравнений непосредственных связей, надо каждой линии на входах и выходах присвоить идентификатор. Затем, рассматривая идентификационные символы как переменные, для каждого элемента составляется его функция. Полученные функции в совокупности и есть модель в виде уравнений непосредственных связей (ниже представлена такая модель для схемы рис. 2.2):

$$d=bc \quad f= \bar{d} \quad e= \bar{a} \bar{d} \quad Z = ef.$$

По модели в виде уравнений непосредственных связей легко вычислять значения выходных сигналов схемы по заданным входным сигналам. Поэтому она часто применяется, когда надо моделировать функционирование (поведение) ОД, в том числе с неисправностями. Описание ОД в моделирующих программах, как правило, задается в виде логической схемы. Однако при этом в неявной форме задается и описание в виде уравнений непосредственных связей. Моделирующая программа содержит библиотеку с программной моделью для каждого типа микросхемы (элемента), которая используется в ОД. В модели запрограммирована функция элемента. Моделирующая программа последовательно

выбирает элементы, присутствующие в описании, вызывает соответствующую программную модель, с помощью которой и производятся вычисления выходных сигналов и состояний очередного элемента.

Эквивалентная нормальная форма

Модель в виде эквивалентной нормальной формы (ЭНФ) нашла применение при решении задач построения тестов (более подробно см. в разделе 4).

Эта модель предполагает, что для каждого выхода логической схемы строится функция, переменные которой ассоциируются с путями от входов схемы до рассматриваемого выхода. Эти переменные называют буквами, причем каждая буква имеет индекс, которому сопоставляются номера элементов схемы, через который проходит соответствующий путь.

Построим ЭНФ для выхода Z схемы рис. 2.2. Как и при формировании системы уравнений непосредственных связей, надо присвоить идентификаторы всем линиям схемы. Функция строится от выхода схемы с использованием правила обратной подстановки:

$$Z=(ef)_1,$$

$$Z=((\bar{a} \bar{d})_2 \bar{d}_3)_1,$$

$$Z=((\bar{a}(\bar{b} \vee \bar{c})_4)_2((\bar{b} \vee \bar{c})_4)_3)_1,$$

$$Z=\bar{a}_{12}(\bar{b}_{124} \vee \bar{c}_{124})(\bar{b}_{134} \vee \bar{c}_{134}),$$

$$Z=\bar{a}_{12} \bar{b}_{124} \bar{b}_{134} \vee \bar{a}_{12} \bar{b}_{124} \bar{b}_{134} \vee \bar{a}_{12} \bar{c}_{124} \bar{b}_{134} \vee \bar{a}_{12} \bar{c}_{124} \bar{c}_{134}.$$

Каждая буква полученного итогового выражения ассоциируется с некоторым путем от одного из внешних входов до выхода схемы. Так, буква \bar{a}_{12} ассоциируется с путем от входа a через элементы $D1$ и $D2$ до выхода Z , буква \bar{b}_{124} с путем от входа b до выхода Z через элементы $D1$, $D2$ и $D4$.

Описание с помощью булевой производной

В задачах построения тестов применяется также аналитическая модель на основе функции, получившей название «булева производная» или «булева разность». Эта модель применяется для цифровых схем.

Пусть выходной сигнал схемы можно представить в виде функции $f(X) = f(x_1, x_2, \dots, x_n)$, где x_i ($i \in \{1, \dots, n\}$) – переменные, сопоставленные входам схемы. Булевой производной функции $f(X) = f(x_1, x_2, \dots, x_n)$ по x_i называется функция $df(X) / dx_i = f(x_1, x_2, \dots, x_i, \dots, x_n) \oplus f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n)$ где \oplus – сумма по модулю 2. Так как переменные могут принимать значения логической «0» или «1», то булева производная может быть также вычислена и по следующей формуле:

$$df(X) / dx_i = f(x_1, x_2, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \oplus f(x_1, x_2, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n).$$

Функция $a \oplus b = 1$, если a и b имеют разные значения. Допустим, $df(X) / dx_i = 1$. Тогда, очевидно, булева производная определяет значения логических переменных x_1, \dots, x_n (кроме x_i), при которых изменение состояния x_i приводит к изменению значения функции $f(X)$.

Пусть, например, $f(X) = \overline{x_1}x_2 \vee x_2 \overline{x_3}$. Тогда булева производная $df(X)/dx_2 = (\overline{x_1} \vee \overline{x_3}) \oplus 0 = \overline{x_1} \vee \overline{x_3}$.

Описание объекта с помощью кубов

Еще одна модель, нашедшая широкое применение в автоматизированных системах построения тестов для цифровых схем, была предложена Ротом (Roth J.P., 1967). В ее основе лежат понятия логических кубов различного вида и правила действия над кубами, что делает удобным реализацию алгоритма построения теста на ЭВМ.

Модель предполагает описание ОД в виде логической схемы, причем каждому логическому элементу схемы задаются три набора различного вида кубов элемента: вырожденные (сингулярные) кубы, d-кубы элемента и d-кубы неисправностей.

Под логическим кубом понимается вектор размерностью n, каждая координата которого имеет одно из пяти значений: 0, 1, x, d, d*. Здесь 0 и 1 представляют обычные булевы значения, x = {0, 1} – символ неопределенного или безразличного значения (иными словами вместо x допустимо любое значение – 0 или 1), символы d и d* (через d* обозначена инверсия d) служат для показа связи сигналов на входах элемента с его выходным сигналом.

Вырожденные (сингулярные) кубы позволяют представить таблицу истинности элемента в сокращенном виде. В этих кубах входам, не влияющим на значение выхода, присваивается значение «x». На рисунке 2.7 для элементов И, ИЛИ-НЕ, M2 показаны их сингулярные кубы. Для элемента M2 нет несущественных входов, поэтому его покрытие полностью повторяет полную таблицу истинности.

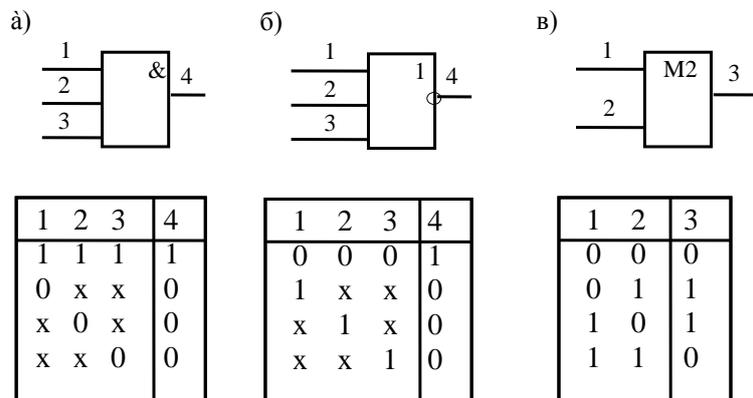


Рис. 2.7. Сингулярные кубы элементов И, ИЛИ-НЕ, M2

d-кубы элементов позволяют указать ситуации, когда сигнал на некотором входе или группе входов определяет значение выхода элемента. Для их построения можно использовать пары вырожденных кубов А и В, при которых значения

на выходной координате в А и В различны. Такой паре соответствует d-куб $G=(g_1, \dots, g_n)$, координаты которого определяются следующим образом:

$$g_j = \begin{cases} d_i, & \text{если } a_i = 1 \text{ и } b_i = 0, \\ d^* & \text{если } a_i = 0 \text{ и } b_i = 1, \\ a_i \cap b_i & \text{в остальных случаях} \end{cases} .$$

Сравнивая подобным образом каждую пару вырожденных кубов, получим множество d-кубов элемента. В каждом из них выходная координата и по крайней мере одна из входных равны d или d*. Например, пересечение первых двух сингулярных кубов элемента И (рис. 2.7а) даст d-куб:

$$1111 \cap 0xx0 = d11d.$$

Полученный куб показывает, что при фиксации в «единицах» второго и третьего входов элемента сигнал на его выходе определяется значением на первом входе. Это делает удобным использование d-кубов для построения пути продвижения сигнала неисправности к контролируемому выходу схемы (один из этапов ряда современных методик построения теста).

В общем случае, необходимо иметь все возможные d-кубы элементов, которые можно получить, если при пересечениях символы «х» заменять на 0 и 1.

Для элементов на рис. 2.7 это d-кубы:

- а) (1,1,d,d), (1,d,1,d), (d,1,1,d), (1,d,d,d),
d,1,d,d), (d,d,1,d), (d,d,d,d), (1,1,d*,d*), (1,d*,1,d*),
(d*,1,1,d*), (1,d*,d*,d*), (d*,1,d*,d*), (d*,d*,1,d*), (d*,d*,d*,d*);
- б) (0,0,d,d*), (0,d,0,d*), (d,0,0,d*), (0,d,d,d*),
(d,0,d,d*), (d,d,0,d*), (d,d,d,d*), (0,0,d*,d), (0,d*,0,d),
(d*,0,0,d), (0,d*,d*,d), (d*,0,d*,d), (d*,d*,0,d), (d*,d*,d*,d);
- в) (0,d,d), (1,d,d*), (d,0,d), (d,1,d*), (0,d*,d*),
(1,d*,d), (d*,0,d*), (d*,1,d).

В d-кубе неисправности координаты, сопоставленные входам элемента, имеют значения, при которых данная неисправность наблюдается (иногда говорят проявляется) по выходу элемента. Координата, сопоставленная выходу, равна d (d*), если при исправном состоянии на выходе элемента 1(0).

Для одной неисправности может существовать несколько d-кубов неисправности. Например, неисправности «0 на линии 1» элемента М2 (рис. 4.4 в) соответствуют два d-куба неисправности: (1 0 d) и (1 1d*).

Ниже, в табл. 2.2, показаны d-кубы неисправностей константного типа элемента И, представленного на рис. 2.7а.

Из таблицы видно, что неисправности типа «0» для всех линий имеют один и тот же d-куб неисправности, а для неисправности «1 на линии 4» существуют три разных d-куба (в этих кубах входам, значения которых безразличны, присвоено значение х).

d-кубы неисправностей элемента И

Неисправность	d-кубы неисправностей
0 на линии 1 0 на линии 2 0 на линии 3 0 на линии 4	1 1 1 d
1 на линии 1	0 1 1 d*
1 на линии 2	1 0 1 d*
1 на линии 3	1 1 0 d*
1 на линии 4	0 x x d* x 0 x d* x x 0 d*

Функциональная модель микропроцессорных устройств

Многие современные электронные изделия относятся к категории программно-управляемых, содержат микропроцессоры, большое число различных регистров и счетчиков, имеют схемы из очень большого числа элементов. Для таких ОД довольно широко применяются модели, которые предполагают представление программно-управляемого ОД из отдельных узлов (называемых функциями или механизмами), которые существенны для выполнения его команд. При этом нет строгой привязки механизмов к конкретным компонентам структуры объекта диагностирования. Поэтому такую модель следует отнести к классу функциональных.

Один из вариантов модели программно-управляемого ОД в виде графа регистровых передач приведен на рис. 2.8. Здесь вершинам графа ставятся в соответствие: регистры из множества $R=\{R_1, R_2, \dots\}$; множество узлов $W=\{W_1, W_2, \dots\}$, специальные вершины IN и OUT. Вершина IN соответствует источнику входных данных микропроцессора (ОЗУ, винчестер и т.д.), а вершина OUT – приемнику выходных данных.

Если существует хотя бы одна команда, во время выполнения которой информация передается из регистра R_i в регистр R_j , то в графе строится ребро, направленное от вершины R_i к вершине R_j . Если существует хотя бы одна команда, при которой информация передается с входов микропроцессора в регистр R_i , то проводится ребро из вершины IN к вершине R_i . Аналогично, если существует хотя бы одна команда, при которой информация передается из R_i на выходы микропроцессора, то строится ребро из вершины R_i к вершине OUT. Над дугой

проставляется список команд, при которых активизируется данная дуга. После команды в скобках могут указываться условия, при которых выполняется данная команда.

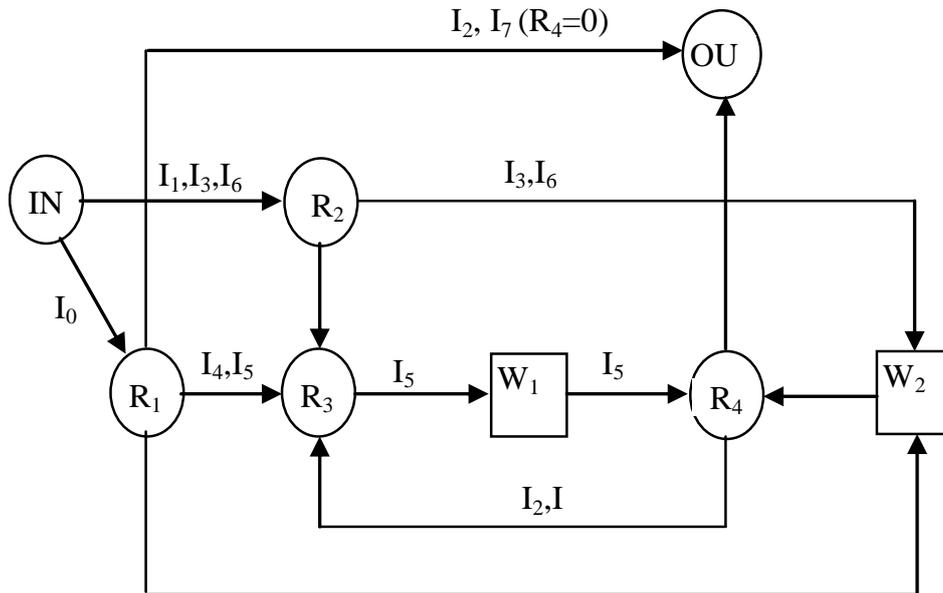


Рис. 2.8. Граф регистровых передач

Вершины типа R (регистры) характеризуются свойством памяти (сохранения информации). Вершины типа W информацию передают далее, производя некоторую обработку, возможно, при выполнении некоторых условий (например, при поступлении только определенных команд). На рисунке 2.8 вершины типа W выделены в виде квадрата. Каждой вершине может сопоставляться некоторая функция, с помощью которой можно определить конкретные значения сигналов на линиях и состояния регистров. Так, в одной из научных публикаций значение z_i для i -го разряда результата при выполнении команд по выполнению арифметической или логической обработки данных предлагается определять по формуле:

$$z_i = U_1 \varphi_i^1 \vee U_2 \varphi_i^2 \vee \dots \vee U_m \varphi_i^m \vee \bar{U}_1 \bar{U}_2 \dots \bar{U}_m z_i^*$$

где U_j – переменная равная «1», если выполняется операция I_j и «0» – в противном случае; φ_i^j – функция вычисления i -го разряда результата при выполнении команды I_j , z_i^* – предыдущее значение i -го разряда, m – число возможных команд. Используя правила булевой алгебры, это выражение можно представить в более компактном виде:

$$z_i = \bigvee_{j=1}^m (U_j \varphi_i^j \vee z_i \cdot \bigwedge_{j=1}^m \bar{U}_j)$$

Модель в виде электрической цепи

Для элементов и устройств с аналоговыми входными и выходными сигналами используются специальные модели.

Для относительно небольших электронных устройств часто используется модель в виде электрической цепи. Ее базовыми элементами являются резисторы, конденсаторы, индуктивности, источники напряжения и тока. Пример такой цепи приведен на рис. 2.9.

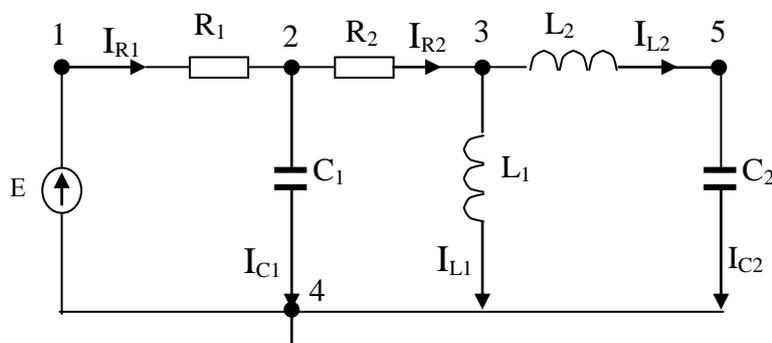


Рис. 2.9. Пример электрической цепи

Если реальная схема содержит транзисторы, диоды, трансформаторы, микросхемы и т.п., то они заменяются эквивалентными схемами (иногда их называют схемами замещения) из базовых элементов. Известно большое число разнообразных эквивалентных схем в зависимости от режимов использования соответствующих компонент. На рисунках 2.10 и 2.11 приведено несколько примеров схем замещения.

На рисунке 2.10а приведена схема замещения транзистора, известная как модель Эберса-Молла. В ней протекание токов через полупроводниковые переходы эмиттер-база и коллектор-база имитируются нелинейными источниками тока I_E и I_K , конденсаторами – емкости переходов, резисторами – утечки и сопротивления переходов. В модели диода (рис. 2.10б) R_d – входное сопротивление и C – емкость полупроводникового перехода диода, с помощью резистора R_y имитируется ток утечки, а с помощью нелинейного источника I_d – ток через переход в зависимости от приложенного напряжения.

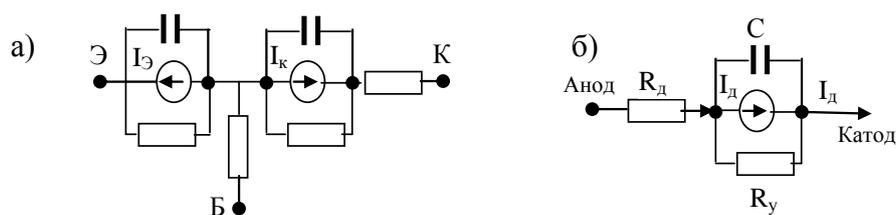


Рис. 2.10. Эквивалентные схемы полупроводниковых элементов: биполярного транзистора (а) и диода (б)

На рисунке 2.11а исходная электрическая схема усилителя включает два операционных усилителя – микросхемы. Их надо заменить эквивалентными схемами. Результирующая электрическая цепь приведена на рис. 2.11б. В ней резисторы $R_{ВХ1}$ и $R_{ВХ2}$, $R_{ВЫХ1}$ и $R_{ВЫХ2}$, K_1 и K_2 имитируют соответственно входные и выходные сопротивления, коэффициенты усиления первого и второго усилителей. Вообще говоря, операционный усилитель содержит десятки транзисторов. Приведенную на рис. 2.11б модель называют макромоделью, так как она не детализирует электрические процессы в усилителе до отдельного транзистора.

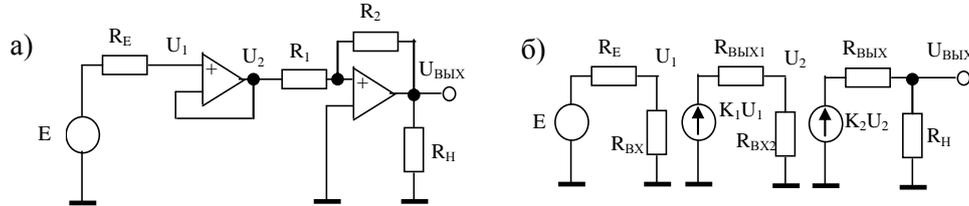


Рис. 2.11. Представление двухкаскадного усилителя в виде электрической цепи

Представление объекта в виде электрической цепи позволяет использовать уравнения на основе законов Кирхгофа для математического описания функционирования цепи.

По первому закону Кирхгофа алгебраическая сумма токов в каждой узловой точке схемы в любой момент времени равна нулю. На рисунке 2.9 узловые точки обозначены цифрами 1, ..., 5. Уравнения токов для этих узлов:

$$-i_E + i_{R1} = 0, \quad i_{C1} - i_{R1} + i_{R2} = 0, \quad -i_{R2} + i_{L1} + i_{L2} = 0, \quad i_{E1} - i_{C1} - i_{C2} - i_{L1} = 0, \quad i_{C2} - i_{L2} = 0.$$

По второму закону Кирхгофа алгебраическая сумма напряжений на элементах схемы, входящих в замкнутый контур, в любой момент времени равна нулю. Чтобы создаваемые уравнения были независимы, каждый новый контур должен иметь хотя бы один элемент, который не входит в другой контур. В нашем примере (цепь на рис. 2.9) имеется три таких контура, для которых можно построить следующие уравнения:

$$U_{R1} + U_{C1} - E = 0, \quad U_{R2} + U_{L1} - U_{C1} = 0, \quad U_{R2} + U_{L2} + U_{C2} - U_{C1} = 0,$$

где U_i – падение напряжения на элементе i .

В зависимости от вида входных воздействий (постоянные или периодические сигналы, импульсы и др.), функциональных свойств элементов, целей использования уравнений они могут быть отнесены к алгебраическим, дифференциальным, линейным, нелинейным и др.

Заметим, что электрическую цепь можно представить в форме графа, уравнения – в матричной форме.

В теоретических исследованиях задач диагностирования зачастую используются методы, получившие название «параметрической идентификации». При этом, обычно, для каждой точки измерения контролируемых сигналов формируется функция, описывающая значение сигнала в точке измерения от внутренних

параметров схемы. После того, как сделаны измерения, их результаты подставляются в построенную систему уравнений, которая решается относительно внутренних параметров. Вычисленные значения параметров сравниваются с установленными на них допусками. Если оказывается, что некоторый параметр вне допуска, то имеет место неисправность среди компонент ОД, определяющих значение соответствующего параметра. Пусть, например, ОД можно представить электрической цепью (рис. 2.12) и в ней можно сделать измерения проводимости между любой парой узлов 1–4 (узлы 5, 6 недоступны для измерений).

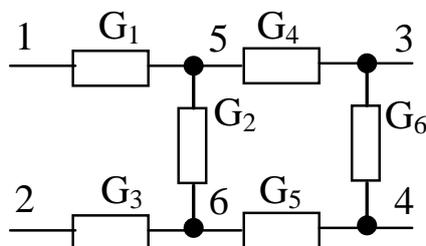


Рис. 2.12. Электрическая цепь с проводимостями

Обозначим через y_{ij} результат измерения проводимости между узлами i и j . Для пар узлов, между которыми измеряется проводимость, можно записать следующие выражения:

$$\begin{aligned}
 y_{12} &= G_1 G_2 G_3 / \Delta, \\
 y_{13} &= G_1 G_4 (G_2 + G_3 + G_5) / \Delta, \\
 y_{14} &= G_1 G_2 G_5 / \Delta, \\
 y_{23} &= G_2 G_3 G_4 / \Delta, \\
 y_{24} &= G_3 G_5 (G_1 + G_2 + G_4) / \Delta, \\
 y_{34} &= G_6 + G_2 G_4 G_5 / \Delta,
 \end{aligned}$$

где $\Delta = (G_1 + G_4) (G_2 + G_3 + G_5) + G_2 (G_2 + G_5)$.

Если измерить значения проводимостей между парами узлов 1–2, 1–3, ..., 3–4 и подставить их в вышеприведенные уравнения вместо соответствующих y_{ij} , то, решая эти уравнения, можно определить действительные значения проводимостей G_1, \dots, G_6 .

Структурная модель аналогового устройства

Многие современные печатные платы, относящиеся к категории аналоговых или аналого-цифровых схем, сегодня конструируются из микросхем, выполняющих законченную функцию. При этом используются структурные модели, в которых отдельные элементы на схеме отображаются отдельным графическим символом, а их функции описываются в виде вход/выходных зависимостей. По

сути, речь идет об описании аналогового или аналого-цифрового объекта в виде уравнений непосредственных связей, широко используемых для цифровых ОД. В таблице 2.2 приведена система таких уравнений для схемы аналого-цифрового ОД, изображенной на рис. 2.13.

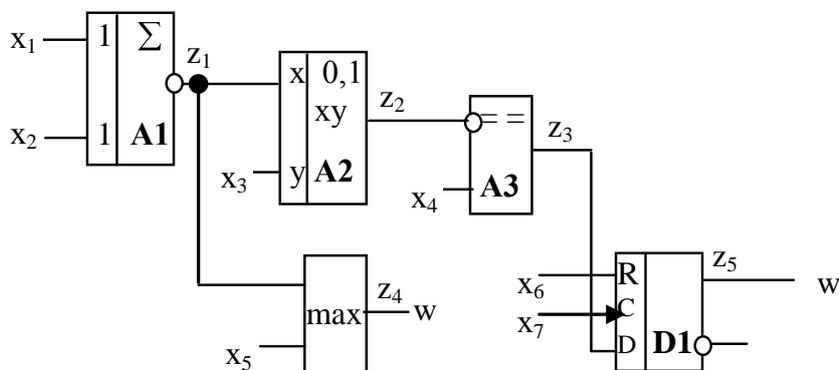


Рис. 2.13. Схема аналого-цифрового устройства

Таблица 2.2

Функции элементов схемы рис. 2.13

Элемент	Уравнение функции элемента
A1 (инвертирующий суммирующий усилитель, коэф. усиления по входам «1»)	$Z_1 = -k(x_1 + x_2)$
A2 (умножитель с коэф. усиления «0,1»)	$Z_2 = 0,1 z_1 x_3$
A3 (компаратор)	$Z_3 = \begin{cases} 0, & \text{если } (-z_2) \leq x_4 \\ 1, & \text{если } (-z_2) > x_4 \end{cases}$
A4 (амплитудный детектор)	$Z_4 = \max(z_1, x_5)$
D1 (D-триггер)	$Z_5 = \bar{R}(C D) \vee y^* \bar{R} \bar{C}$ $y^* = z_5$

Логическая модель аналогового устройства

Для поиска места неисправности в сложных аналоговых системах (в том числе содержащих гидравлические, пневматические или электромеханические узлы) могут применяться методы, основанные на использовании логической модели ОД.

Структурно-функциональная схема объекта диагностирования, по которой будет строиться логическая модель, должна удовлетворять следующим требованиям.

1. Для функционального элемента схемы известны допустимые значения входных и выходных сигналов, их функциональная зависимость.
2. Линии связи между функциональными элементами должны соответствовать линиям связи принципиальной электрической схемы.
3. Выход любого элемента можно соединять с любым числом входов, вход любого элемента может быть соединен только с одним выходом.
4. Внешние входные сигналы всегда принимают только допустимые значения.
5. Если входной сигнал функционального элемента является входным и для другого функционального элемента, то допустимые значения этих сигналов совпадают.
6. Выходной сигнал элемента является допустимым, если допустимыми являются все его входные сигналы, от которых зависит значение данного выхода.

Электрический сигнал, поступающий по некоторому проводнику, может характеризоваться несколькими параметрами, а функциональный элемент иметь несколько выходов. Тогда рекомендуется «расщеплять» сигналы, и соответствующий физический блок заменять на несколько модельных (по числу выходов и параметров) с увеличенным числом входов (хотя такое преобразование и не является обязательным). Так, если физический блок имеет два входа x_1 , x_2 и один выход z , который характеризуется двумя параметрами (рис. 2.14а), то в модели он представляется в виде блока, имеющего четыре входа и два выхода (рис. 2.14б) либо двух блоков отдельно для выхода z_1 и z_2 .

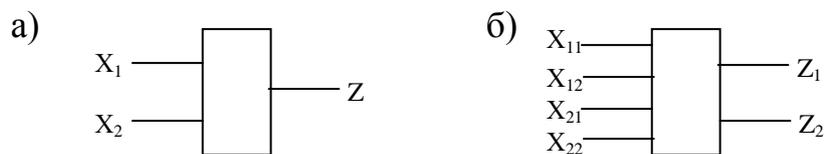


Рис. 2.14. Преобразование блока при «расщеплении» сигналов

Сигналы в логической модели могут иметь одно из двух «состояний» – допустимое и недопустимое. Закодируем допустимое значение символом «1», а недопустимое – символом «0». Тогда для моделирования и анализа процессов на логической модели можно использовать аппарат булевой алгебры. Отсюда и название модели – логическая.

Обычно для построения логической модели надо иметь структурную или структурно-функциональную модель объекта. На рисунке 2.15 показана упрощенная структурно-функциональная схема радиоприемника. Логическая модель, соответствующая этой схеме, представлена на рис. 2.16. Хотя сигналы в радио-

приемнике характеризуются амплитудой и частотой в данной модели (для упрощения), расщепление сигналов по параметрам не производилось. Однако понятно, что для моделирования процессов функции элементов должны это учитывать.

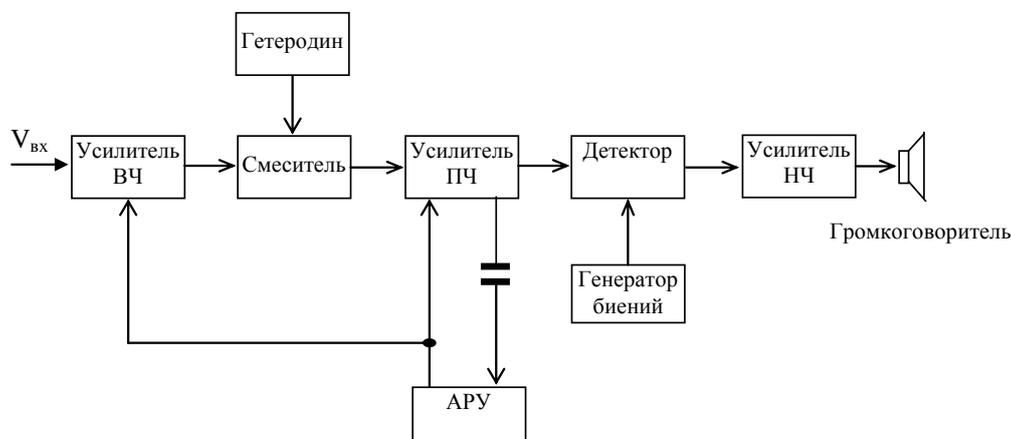


Рис. 2.15. Структурно-функциональная схема радиоприемника

В модели рис. 2.16 через z_i обозначен выходной сигнал i -го элемента. Усилитель промежуточной частоты (ПЧ) имеет в исходной схеме (рис. 2.15) два выхода, поэтому в модели он «расщеплен» на два элемента 3_1 и 3_2 .

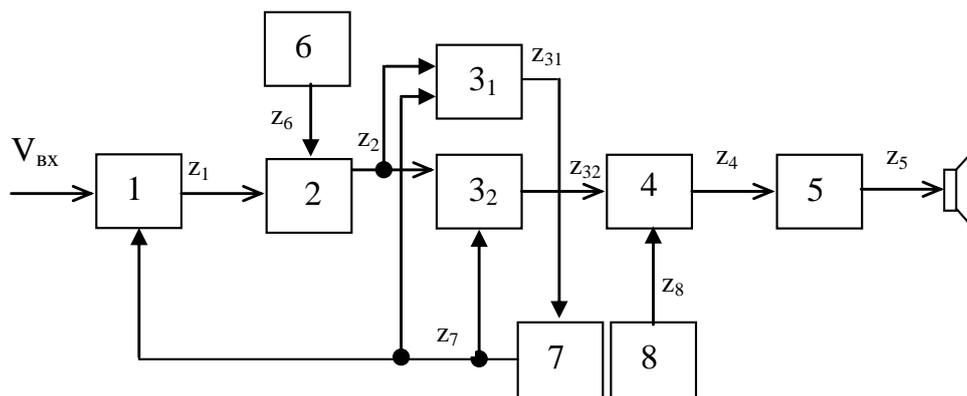


Рис. 2.16. Логическая модель, соответствующая схеме рис. 2.15: 1 – усилитель ВЧ, 2 – смеситель, 3_1 и 3_2 – усилитель ПЧ, 4 – детектор, 5 – усилитель НЧ, 6 – гетеродин, 7 – АРУ, 8 – генератор биений

Модели динамических систем

Эта модель чаще всего применяется для описания изменений во времени сигналов объектов, которые работают с аналоговыми сигналами и могут быть

представлены в виде функциональной и структурно-функциональной модели. К таким объектам относятся, например, различные электромеханические системы управления и регулирования. В теории управления объекты, функционирование которых рассматривается во времени, называют **динамическими системами**.

В общем виде изменения во времени выходных сигналов $Y(t)$ системы от ее входов $X(t)$ связаны сложной дифференциальной зависимостью, в которой присутствуют операторы дифференцирования. Функция $Y(t)$ может быть как **линейной**, так и **нелинейной**. Соответственно объект называется линейным либо нелинейным.

При расчетах и исследовании динамических систем обычно пытаются их линеаризовать, то есть заменить нелинейные уравнения линейными, приближенно описывающими процессы в системах (это упрощает описание и расчеты). Затем осуществляют учет вносимых нелинейностями особенностей.

Так, дифференциальное уравнение, определяющее зависимость выходного сигнала $y(t)$ от входной переменной $x(t)$ линейной системы с одним управляющим входом и выходом, может быть представлено как:

$$\begin{aligned} a_0 \frac{d^n y(t)}{dt^n} + a_1 \frac{d^{n-1} y(t)}{dt^{n-1}} + \dots + a_{n-1} \frac{dy(t)}{dt} + a_n y(t) &= \rightarrow \\ = b_0 \frac{d^m x(t)}{dt^m} + b_1 \frac{d^{m-1} x(t)}{dt^{m-1}} + \dots + b_0 \frac{dx(t)}{dt} + b_m x(t), \end{aligned}$$

где a_0, \dots, a_n и b_0, \dots, b_m – некоторые коэффициенты, значения которых в общем случае не являются постоянными.

Решение этого уравнения позволяет определять поведение объекта в динамических режимах работы. Наибольшая степень производной определяет степень объекта.

Системы дифференциальных уравнений можно записывать в символической форме с использованием так называемого оператора дифференцирования $p=d/dt$, что позволяет рассматривать дифференциальные уравнения как алгебраические. Приведенное выше общее уравнение в операторной форме записи имеет вид:

$$a_0 p^n y(t) + a_1 p^{n-1} y(t) + \dots + a_{n-1} p y(t) + a_n y(t) = b_0 p^m x(t) + \dots + b_{m-1} p x(t) + b_m x(t).$$

Полученное алгебраическое уравнение позволяет представить функцию объекта в виде

$$y(t) = \frac{b_0 p^m + b_1 p^{m-1} + \dots + b_{m-1} p^1 + b_m}{a_0 p^n + a_1 p^{n-1} + \dots + a_{n-1} p + a_n} \times x(t).$$

При многих входных и выходных сигналах объект обычно описывается системой уравнений. Так, некоторые объекты описывают в форме уравнений Коши,

которые представляют собой уравнения относительно первой производной переменных:

$$\left. \begin{aligned} \frac{dx_1}{dt} &= a_{11}x_1 + a_{21}x_2 + \dots + a_{n1}x_n + f_1(t) \\ \frac{dx_2}{dt} &= a_{12}x_1 + a_{22}x_2 + \dots + a_{n2}x_n + f_2(t) \\ \frac{dx_n}{dt} &= a_{1n}x_1 + a_{2n}x_2 + \dots + a_{nn}x_n + f_n(t) \end{aligned} \right\},$$

где x_1, x_2, \dots, x_n – собственные координаты системы, в которые входят ошибки системы, рабочие воздействия на объект, выходные сигналы; $a_{11}, a_{12}, \dots, a_{nn}$ – постоянные или переменные коэффициенты, определяемые структурой и параметрами объекта; $f_1(t), f_2(t), \dots, f_n(t)$ – сигналы внешнего воздействия на объект.

В теории управления очень часто изучаемый объект представляется из элементов (звеньев), каждый из которых описывается передаточной характеристикой либо **передаточной функцией**. При этом принято представлять передаточную функцию $W(s)$ как отношение преобразования Лапласа выходного сигнала $Y(s)$ к преобразованию Лапласа входного сигнала $X(s)$ при нулевых начальных условиях, т.е. $W(s) = Y(s) / X(s)$.

Важным свойством преобразования Лапласа является замена операций дифференцирования и интегрирования умножением и делением на оператор Лапласа, соответственно.

При переходе к преобразованию Лапласа далее рассматриваются не временные функции входной и выходной переменной, а их изображения по Лапласу. Существуют справочники, содержащие передаточные функции для типовых элементов (звеньев). Широкое применение передаточных функций с использованием преобразований Лапласа обусловлено также и тем, что в этом случае передаточную функцию параллельно (последовательно) соединенных элементов можно определить путем суммирования (произведения) передаточных функций элементов, что упрощает расчеты. Например, передаточная функция для безынерционного звена, реализующего функцию $y(t) = K \cdot x(t)$, будет равна $W(s) = K$; для интегрирующего звена $dy/dx = K \cdot x(t)$ передаточная функция $W(s) = K/s$, где K – коэффициент усиления; для апериодического звена $dy/dt = 1/T(KX - Y)$ передаточная функция $W(s) = K/(1 + Ts)$, где T – некоторая постоянная времени.

Кроме того, широко используется **метод пространства состояний**, который позволяет описать линейный объект системами уравнений в матричной форме. При этом в виде матриц представляются входные, выходные, внутренние и другие параметры, что позволяет применять хорошо автоматизируемые на ЭВМ алгоритмы матричного решения уравнений, описывающих функционирование объекта.

Описание в пространстве состояний для линейной системы имеет вид:

$$\dot{Z} = AZ + BX, Y = CZ + DX,$$

где X, Y и Z – вектор (матрица) входных, выходных и внутренних сигналов (переменных) соответственно; \dot{Z} – матрица производных внутренних сигналов; A, B, C, D – постоянные матрицы, характеризующие структуру объекта. Матрица A

описывает связи внутренних элементов между собой и внутренние обратные связи каждого элемента, матрица В – связи внутренних элементов с внешними входами, матрица С – связи внутренних элементов с внешними выходами, а матрица D отражает прямые связи между входами и выходами объекта.

Рассмотрим в качестве примера объект, изображенный на рис. 2.17, содержащий три звена первого порядка (A1, A2, A3), два сумматора, имеющий два входа x_1 и x_2 и два выхода y_1 и y_2 . Передаточные функции элементов приведены внутри графических изображений звеньев. Фактически это апериодические звенья с постоянной времени, равной 1.

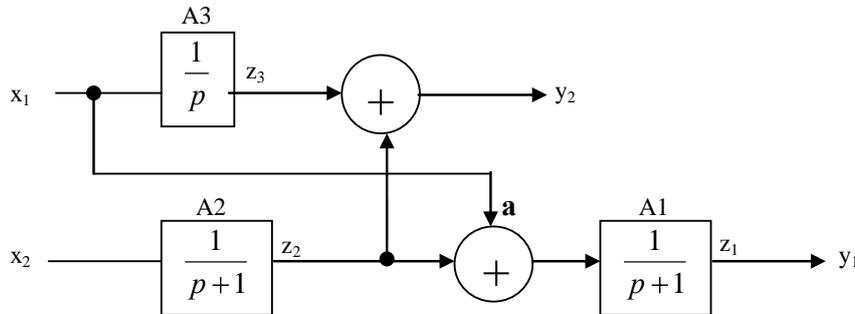


Рис. 2.17. Пример линейной динамической системы

Возьмем в качестве внутренних сигналов (сигналов состояния) выходы z_1, z_2, z_3 звеньев A1, A2, A3.

Передаточную функцию звена, например A3, можно записать в виде $1/p = z_3/x_1$. Далее, используя простейшие преобразования, можно записать систему уравнений:

$$\begin{aligned} z_1(p+1) &= z_2 + x_1, & y_2 &= z_2 + z_3 \\ z_2(p+1) &= x_2, & y_1 &= z_1, \\ z_3p &= x_1. \end{aligned}$$

Раскрывая скобки и перенося первые производные в левую часть уравнений, получим

$$\begin{aligned} z_1p &= -z_1 + z_2 + x_1, & y_1 &= z_1 \\ z_2p &= -z_2 + x_2, & y_2 &= z_2 + z_3 \\ z_3p &= x_1 \end{aligned}$$

Переходя к матричной форме уравнений и обозначениям $z_1p = \dot{z}_1, z_2p = \dot{z}_2, z_3p = \dot{z}_3$, получим искомое описание:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 10 \\ 01 \\ 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}; \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 100 \\ 011 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix},$$

$$\text{где } A = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 10 \\ 01 \\ 10 \end{bmatrix}, C = \begin{bmatrix} 100 \\ 011 \end{bmatrix}.$$

Заметим, в приведенной выше матричной форме отсутствует матрица D (т.е. D=0). Иными словами, переменные x_i не входят в правую часть уравнений y_i , что и видно по исходной системе уравнений.

Если между выходом j -го звена и входом i -го звена имеется связь (т.е. в уравнении z_i присутствует переменная z_j), то величина коэффициента при z_j отображается значением a_{ij} , а если связь отсутствует, то $a_{ij} = 0$. В нашем примере значение в матрице A коэффициента $a_{12}=1$ говорит о наличии прямой связи между звеньями A1 и A2 (на схеме 2.17 видно, что сигнал с A2 поступает через сумматор на элемент A1), коэффициенты $a_{11}=-1$ и $a_{22}=-1$ показывают наличие отрицательных обратных связей в звеньях A1 и A2, в то время как в A3 таких связей нет и это видно по коэффициенту $a_{33}=0$. Значение $a_{21}=0$ и $a_{23}=0$ отражает факт отсутствия связей между выходами A1 и A3 и входом A2. То же самое можно сказать о звене A3, что видно по нулевым значениям a_{31} и a_{32} .

Аналогичным образом коэффициенты матриц B и C показывают наличие или отсутствие связей входов и выходов схемы с ее внутренними звеньями. В коэффициенте b_{ij} индекс i соответствует номеру i -го звена, а индекс j соответствует номеру j -го входа, а в коэффициенте c_{ij} наоборот, i соответствует номеру выхода схемы, а j – номеру внутреннего звена. Так, коэффициент $c_{21}=1$ говорит о наличии связи A1 с выходом y_2 , а $b_{31}=1$ – о наличии связи входа x_1 с элементом A3.

2.2. Модели неисправных технических состояний

Причины неисправностей

Первопричиной нарушений в функционировании ОД являются различные **физические дефекты**: пробой полупроводникового перехода, окисление провода в месте пайки, обрыв проводника и т.п.

Под неисправностью понимают некоторую **модель функционального проявления физических дефектов** в ОД. При этом, возможно, несколько разных физических дефектов кристалла цифровой микросхемы приводят к тому, что, например, на выходном контакте микросхемы будет постоянно напряжение, соответствующее логическому уровню «0». Однако все они будут приводить (с точки зрения алгоритма функционирования ОД) к одной и той же неисправности: постоянное значение «0» на данном контакте.

Практика показывает, что при производстве значительная часть дефектов связана с процессами пайки (короткие замыкания, непропай, отсутствие или избыток припоя и т.п.). Возможна также установка неверных или со смещением компонентов (например, микросхем с малым шагом выводов). Указанные дефекты зачастую составляют до 90% всех возможных на этапе производства (рис. 2.18).

На этапе эксплуатации причинами дефектов чаще всего являются старение и износ элементов, проявление в ходе эксплуатации скрытых дефектов, являю-

щихся следствием некачественного изготовления, или несоблюдение условий эксплуатации. Старение и износ являются процессами постепенного, необратимого изменения электронных компонент и конструкции ОД. Действуя совместно, в некоторых случаях они могут вызывать резкое изменение функциональных параметров ОД. Старение происходит непрерывно и не зависит от того, в рабочем или нерабочем состоянии находится система. Износ происходит при применении объекта. У пластичных материалов из-за старения появляются остаточные деформации, а у упругих материалов теряются упругие свойства. Процессы старения и износа являются неизбежными. Их нельзя полностью предотвратить, можно только в некоторой степени уменьшить вызываемые ими последствия.



Рис. 2.18. Статистика причин неисправностей при производстве

Следует иметь в виду, что характер и статистика причин неисправностей при эксплуатации достаточно сильно отличаются от того, что имеет место при производстве (изготовлении) электронных изделий.

Основными внешними физическими факторами, приводящими к отказам в процессе эксплуатации ОД, являются:

- изменение температуры внутренней и внешней среды системы;
- высокая влажность;
- наличие агрессивных сред (газы, жидкости, пары, пыль, насекомые);
- влияние вибрации и ударов.

Все эти факторы приводят к естественному ухудшению параметров состояния элементов объекта.

Как правило, все объекты подвергаются атмосферным воздействиям, что связано с изменениями температуры и влажности. Повышение температуры и влажности вызывает как постепенные, так и внезапные изменения в материалах. При этом ускоряются химические реакции, происходят механические деформации. При высоких рабочих температурах некоторые изоляционные материалы разлагаются, изменяют свою внутреннюю структуру, выделяют из себя различ-

ные газы. С изменением температуры и влажности нелинейно изменяются величины электрического сопротивления и магнитной проницаемости для многих материалов, активизируются процессы окисления. Эти процессы существенно ускоряются под действием других физических факторов, например: ионизирующих излучений, вибраций, ультрафиолетового излучения, механических воздействий, электрических полей.

Все это приводит к тому, что в ходе эксплуатации повышается переходное сопротивление в местах пайки, изменяются параметры резисторов, полупроводниковых переходов и т.п. В результате происходят изменения электрических токов и напряжений, ухудшение изоляции, что в итоге приводит к отказу в работе ОД.

Обычно, чем больше число компонентов, из которых состоит ОД, тем больше вероятность отказов по причине их неисправности.

Например, пусть при изготовлении платы транзисторного приемника на предприятии осуществляется выборочный входной контроль транзисторов, обеспечивающий поступление на сборку не более 1% бракованных транзисторов. Если каждый приемник содержит по 10 транзисторов, то вероятность нахождения на плате хотя бы одного неисправного транзистора равна 0,1, и можно ожидать, что по крайней мере 9 изделий из 10 будут работоспособны.

Если собираемое устройство содержит 100 компонентов, то при тех же условиях, после сборки вероятность наличия хотя бы одного работоспособного устройства близка к нулю. Поэтому перед сборкой изделий из большого числа элементов проводят тщательный входной контроль комплектующих.

Аналогичная ситуация возникает при сборке объектов, состоящих из нескольких печатных плат. При среднем выходе годных плат 75% и отсутствии их контроля, даже при изготовлении сравнительно простого узла, состоящего из семи плат, девять из десяти собранных изделий могут оказаться неработоспособными.

Виды неисправностей и отказов

Принято различать одиночные и кратные неисправности. Под **одиночной** понимается неисправность, принимаемая в качестве элементарной, т.е. такой, которая не может быть представлена совокупностью нескольких других, более «мелких» неисправностей. **Кратная неисправность** является совокупностью одновременно существующих двух или большего числа одиночных неисправностей.

Появление неисправности может стать причиной перехода объекта в неработоспособное состояние. Тогда говорят, что произошел отказ в работе устройства.

Отказ – событие, заключающееся в нарушении работоспособного состояния объекта.

Неисправности бывают **устойчивые** и **неустойчивые**. Среди неустойчивых выделяют, в частности, сбой и перемежающийся отказ.

Сбой – самоустраняющийся отказ или однократный отказ, устраняемый незначительным вмешательством оператора.

Перемежающийся отказ – многократно возникающий самоустраняющийся отказ одного и того же характера.

Внезапный отказ – отказ, характеризующийся скачкообразным изменением значений одного или нескольких параметров объекта.

Постепенный отказ – отказ, возникающий в результате постепенного изменения значений одного или нескольких параметров объекта.

Некоторые неисправности не приводят к нарушению работоспособности объекта. Такие неисправности называют **избыточными**. Например, избыточные неисправности появятся, если задействуются не все выводы или функциональные возможности микросхем, применяемых в объекте диагностирования.

Критичность и тяжесть последствий отказов

С целью выработки более совершенных правил эксплуатации системы технического обслуживания и ремонта объекта, обеспечивающих предупреждение возникновения и/или ослабление тяжести возможных последствий отказов, а также для достижения требуемых характеристик безопасности, экологичности, эффективности и надежности рекомендуется делать анализ видов, последствий и критичности отказов.

Общие понятия, классификация отказов по их критичности и последствиям, методике проведения анализа даны в ГОСТ 27.310-95 «Анализ видов, последствий и критичности отказов. Основные положения», который соответствует международному стандарту МЭК 812(1985) в части определений, общих положений и методов анализа.

В соответствии с этим стандартом:

- **Тяжесть последствий отказа** – качественная или количественная оценка вероятного (наблюдаемого) ущерба от отказа элемента и/или системы.

- **Категория тяжести последствий отказов** – классификационная группа отказов по тяжести их последствий, характеризующаяся определенным, установленным до проведения анализа сочетанием качественных и/или количественных учитываемых составляющих ожидаемого (вероятного) отказа или нанесенного отказом ущерба (табл. 2.3).

Таблица 2.3

Категории отказов

Категория	Характеристика тяжести последствий отказов
I	Отказ, который может повлечь снижение качества функционирования объекта, но не представляет опасности для окружающей среды, самого объекта и здоровья людей
II	Отказ, который может повлечь задержку выполнения задачи, снижение готовности и эффективности объекта, но не представляет опасности для окружающей среды, самого объекта и здоровья людей
III	Отказ, который быстро и с высокой вероятностью может повлечь за собой значительный ущерб для самого объекта и/или для окружающей среды, срыв выполняемой задачи, но создает пренебрежимо малую угрозу жизни и здоровью людей
IV	Отказ, который быстро и с высокой вероятностью может повлечь за собой значительный ущерб для самого объекта и/или окружающей среды, гибель или тяжелые травмы людей, срыв выполнения поставленной задачи

- **Критический отказ** – отказ системы или ее элемента, тяжесть последствий которого в пределах данного анализа признана недопустимой и требует принятия специальных мер по снижению вероятности данного отказа и/или возможного ущерба, связанного с его возникновением.
- **Критичный элемент** – элемент системы, отказ которого может быть критическим.

Принципы формирования списка неисправностей

При решении задач диагностирования надо определить, что мы будем понимать под неисправностью объекта диагностирования. Это во многом зависит от того, что будет заменяемой единицей при ремонте, каковы цели диагностирования. Формальные методы построения тестов или поиска неисправностей, как правило, предполагают предварительное определение списка возможных (допустимых) неисправностей. Когда такой список сформулирован, то в дальнейшем задача решается в предположении, что фактическая неисправность является одной из этого списка.

Например, при решении задачи извлечения информации с неправильно работающего жесткого диска предварительно определяются наличие:

- дефектных секторов;
- нарушений в работе диска, связанных с сервоазметкой;
- неисправностей Блока магнитных головок;
- разрушения служебной информации;
- неисправностей платы электроники;
- разрушения динамического транслятора;
- нарушений логической структуры данных.

В зависимости от выявленных нарушений далее применяют те или иные способы и средства, позволяющие считать и восстановить информацию ненормально работающего диска.

Если объектом является печатная плата с микросхемами, то в качестве неисправности обычно рассматривают дефекты микросхем и проводников платы, которые отражаются на логике функционирования электронной схемы объекта или могут вывести параметры сигналов за допустимые пределы.

Если ОД представляется направленным графом, где дуги показывают направление и содержание передаваемой информации, то под неисправностью понимают искажения, приводящие к изменению функции, сопоставленной узлу графа, и прекращению или изменению направления передачи данных (сигналов).

Процесс диагностирования может осуществляться в несколько этапов, причем при переходе к новому этапу может меняться представление о неисправности, т.е. модель неисправности.

Наиболее формализованы и хорошо поддаются автоматизации задачи диагностирования объектов, которые можно представить в виде электрических схем. Как правило, каждой неисправности ставится в соответствие некоторое изменение связей или параметров элементов электрической схемы ОД. Например, часто в качестве допустимых рассматриваются неисправности, которые являются следствием обрывов и коротких замыканий линий связи в электрической схеме

ОД, выхода параметра элемента за пределы заданного допуска, пробоев полупроводниковых переходов и т.п.

Обычно модель неисправности зависит от элементной базы, а также от используемой модели ОД.

Некоторые неисправности могут приводить к таким нарушениям в схемах, при которых существенно усложняется поиск их места. Например, в схеме на рис. 2.19 при неисправности «короткое замыкание» диода D11 элемента D1 она будет проявляться по выходу элемента D2, а не D1. Пусть на входы abc подается комбинация логических сигналов 101. Поступивший на вход b сигнал логический «0» из-за закоротки диода D11 поступит и на диод D21 и далее на выход Y2. Иными словами, хотя дефект в элементе D1, но неправильное значение сигнала будет на выходе другого элемента (D2). При рассматриваемой неисправности в исправной схеме функции $Y1 = ab$ и $Y2 = bc$, а в неисправной – $Y1 = ab$ и $Y2 = abc$.

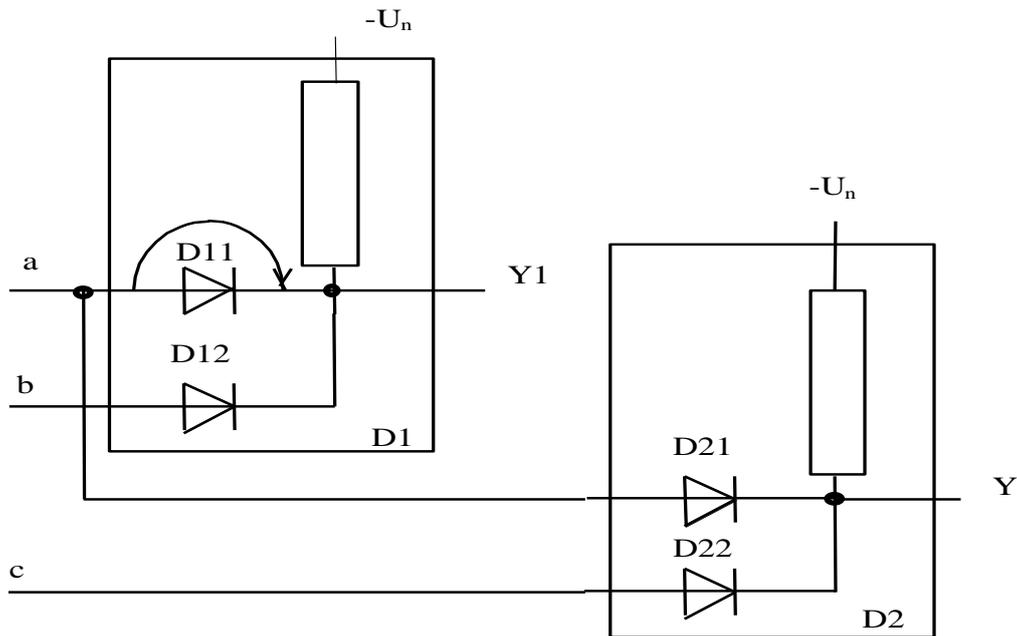


Рис. 2.19. Пример влияния дефекта одного элемента на поведение другого элемента

В схеме на рис. 2.20 короткое замыкание одного из диодов элемента D3 приводит к образованию обратной связи, т.е. превращению схемы из комбинационной в последовательностную. В результате поведение схемы начинает зависеть не только от текущих значений сигналов на внешних входах, но и от предшествующих состояний элементов, что усложняет процессы диагностирования.

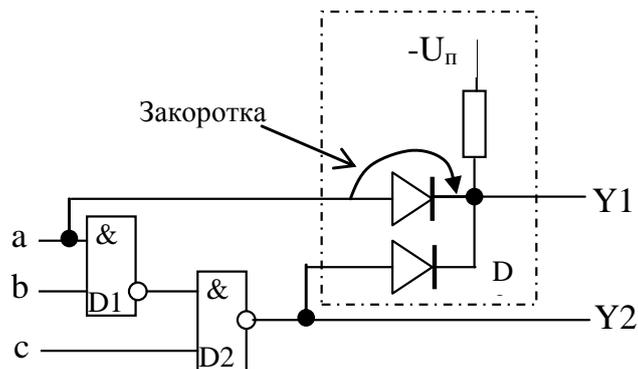


Рис. 2.20. Неисправность, при которой комбинационная схема превращается в последовательностную

Модель константной неисправности для цифрового устройства

Пусть имеется цифровой комбинационный элемент, реализующий функцию $f(x_1, \dots, x_m)$, где m – число входов элемента; x_1, \dots, x_m – входные переменные элемента, принимающие значения 0 или 1. Будем говорить, две функции $f_i(x_1, \dots, x_m)$ и $f_j(x_1, \dots, x_m)$ различимы, если $f_i(x_1, \dots, x_m) \neq f_j(x_1, \dots, x_m)$ хотя бы для одной комбинации значений переменных x_1, \dots, x_m .

От m двоичных аргументов возможно 2^m различных функций. Одна из них соответствует исправному элементу. Следовательно, теоретически, число возможных различных (по функции) неисправностей m -входового цифрового комбинационного элемента не может превышать $2^m - 1$.

Если рассматривать цифровой элемент памяти, то при определении числа возможных различных функций надо учитывать не только число входов, но и число внутренних переменных. Однако, как правило, число реальных различных неисправностей элемента существенно меньше $2^m - 1$. Были выполнены исследования поведения цифровых элементов при наличии в них неисправностей катастрофического типа (обрыв или короткое замыкание отдельных компонентов схемы элемента). Исследования показали, что наличие в элементе таких неисправностей эквивалентно, как правило, тому, что определенная комбинация переменных функций элемента зафиксирована константами 0 или 1.

В качестве иллюстрации рассмотрим элемент ИЛИ-НЕ, принципиальная и логическая схема которого представлены на рис. 2.21 (на рис. 2.21а приведена его электрическая схема, на рис. 2.21б – логическая). Он имеет три входа (a, b, c) и выход Y.

Пусть возможными физическими неисправностями элемента являются:

- 1) обрыв цепи эмиттера;
- 2) обрыв цепи базы;
- 3) обрыв цепи коллектора;
- 4) обрыв цепи резистора R4;
- 5) обрыв цепи резистора R5;
- 6) короткое замыкание между эмиттером и коллектором;

- 7) короткое замыкание между эмиттером и базой;
- 8) короткое замыкание между базой и коллектором;
- 9) обрыв цепи резистора R1;
- 10) обрыв цепи резистора R2;
- 11) обрыв цепи резистора R3.

Анализ функционирования элемента показывает, что наличие каждой неисправности 1, 2, 3 и 7 эквивалентно тому, что выход элемента зафиксирован константой «1»; при каждой неисправности 4, 5, 6 и 8 – константой «0». Неисправности 9, 10 и 11 эквивалентны фиксации константой «0» входов a, b и c, соответственно.

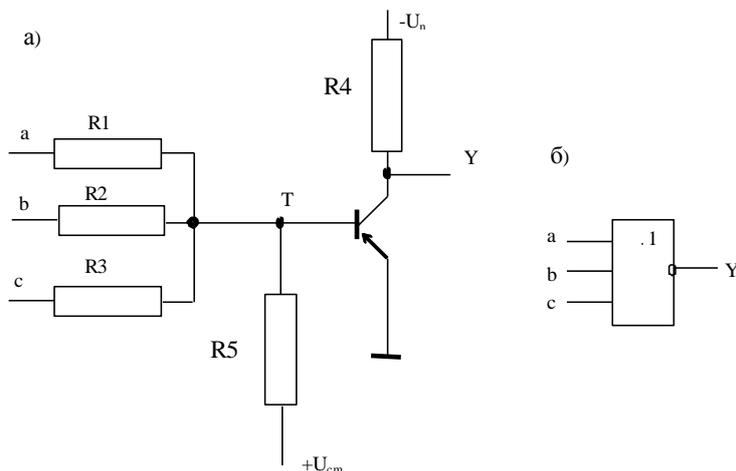


Рис. 2.21. Электрическая и логическая схемы элемента ИЛИ-НЕ

Хотя в примере рис. 2.21 могут быть 11 разных дефектов, при их наличии возможны всего 5 различных функций.

Неисправности, при которых функционирование цифрового элемента можно описать, фиксируя в значениях 0 и 1 переменные функции исправного элемента, называют неисправностями **«константного типа»**.

При решении ряда задач технической диагностики необходимо задавать список возможных неисправностей. Очевидно сокращение (разумное) списка возможных неисправностей существенно снижает трудоемкость решений. При выборе константной модели обычно принимают, что сигнал на каждом выводе может быть зафиксирован в значении «0» или «1». Тогда число неисправностей для элемента с m внешними выводами равно 2^m . Очевидно, это намного меньше, чем $(2^{2^m}-1)$.

Количество рассматриваемых неисправностей можно еще больше сократить, если выделить только различимые (по функции) неисправности¹. Доказано, что для комбинационных элементов типа И, ИЛИ, И-НЕ, ИЛИ-НЕ с m входами

¹ Неисправности, при которых элемент реализует одну и ту же функцию, называют эквивалентными. Обычно достаточно рассматривать только одну неисправность из группы эквивалентных.

число различных одиночных константных неисправностей равно $m+2$ (нетрудно заметить, что данное правило выполняется и для рис. 2.21).

Заметим, что имеются исследования, показывающие существование в цифровых схемах дефектов, которые не могут быть описаны константной моделью. Например, в схемах на МОП-транзисторах из-за дефектов возможно появление на выходе уровня напряжения между логическими 0 и 1, или превращение неисправного комбинационного элемента в схему, выходной сигнал которой зависит от его состояния перед подачей тестового воздействия (т.е. комбинационный элемент из-за дефекта превращается в элемент с памятью). В некоторых цифровых элементах из-за дефектов выполняемая функция становится инверсной к нормальной.

Однако многолетняя практика диагностирования цифровых ОД, в том числе программно-управляемых, выполненных с использованием больших интегральных схем, показывает высокую эффективность применения константной модели при решении различных задач диагностирования.

Неисправности в конечных автоматах

При использовании модели ОД в виде конечного автомата под неисправностью обычно понимают искажения в таблице переходов автомата. Часто считают, что при неисправностях число состояний автомата не увеличивается.

Неисправности в электрических цепях

В тех случаях, когда ОД описывается схемами из резисторов, конденсаторов, диодов, транзисторов неисправностям обычно ставится в соответствие обрывы и короткие замыкания резисторов, конденсаторов, пробой полупроводниковых переходов, отклонения параметров элементов схемы за пределы установленных допусков.

Пусть $r_{\text{ном}}$ – номинальное значение параметра r некоторого элемента и $\Delta r_{\text{ном}}$ – допустимое отклонение от номинала. С учетом допуска параметр r считается в норме, если значение r находится в пределах интервала $r_{\text{ном}} \pm \Delta r_{\text{ном}}$. В общем случае элемент считается неисправным, если $r > r_{\text{ном}} + \Delta r_{\text{ном}}$ либо $r < r_{\text{ном}} - \Delta r_{\text{ном}}$. Тогда число неисправностей элемента равно числу параметров, характеризующих элемент.

Возможны модели, когда каждому параметру ставится в соответствие несколько неисправностей. Например, значению параметра $r > r_{\text{ном}} + \Delta r_{\text{ном}}$ может сопоставляться одна неисправность, а $r < r_{\text{ном}} - \Delta r_{\text{ном}}$ – другая. Известны разработки, в которых отдельным неисправностям ставятся в соответствие некоторые конкретные отклонения от номинала: 50%, 100%, 200% и т.п. Еще один вариант модели данного вида предполагает для элемента 5 возможных технических состояний:

N – значение параметра в заданных пределах;

$N >$ – значение параметра выше заданного верхнего допустимого предела, но ниже верхнего предела диапазона измерения;

$H <$ – значение параметра ниже заданного нижнего допустимого предела, но выше нижнего предела диапазона измерения;

\emptyset – значение параметра ниже заданного нижнего диапазона измерения;

∞ – значение параметра выше заданного верхнего значения диапазона измерения.

Известны расширения данной модели неисправностей, учитывающие возможность установки другого элемента, короткого замыкания или обрыва и т.п.

Например, резистору R можно сопоставить одно из 12 технических состояний:

$R = R_n$ – установлен исправный резистор заданного сопротивления;

$R <$ – установлен резистор меньшего сопротивления;

$R >$ – установлен резистор большего сопротивления;

$R \ll$ – установлен резистор очень малого сопротивления;

$R \gg$ – установлен резистор очень большого сопротивления;

$R = 0$ – установлен резистор с дефектом «к.з.» (короткое замыкание);

$R = \infty$ – не установлен резистор;

$R = L$ – вместо резистора установлена индуктивность;

$R = C$ – вместо резистора установлен конденсатор;

$R = VD \rightarrow$ или $R = VD \leftarrow$ – вместо резистора установлен диод в прямом или обратном направлении соответственно;

$R = VD \leftrightarrow$ – вместо резистора установлен диод при наличии параллельного диода (т.е. при неисправности получается схема из двух параллельно включенных в разных направлениях диодов).

В некоторых системах автоматизации модели элементов модифицируют специально для упрощения процессов моделирования ОД с различными неисправностями. На рисунке 2.22 дана модель для моделирования неисправностей транзистора. Для моделирования неисправностей типа «обрывы и короткие замыкания полупроводниковых переходов» в нее внесено 6 резисторов.

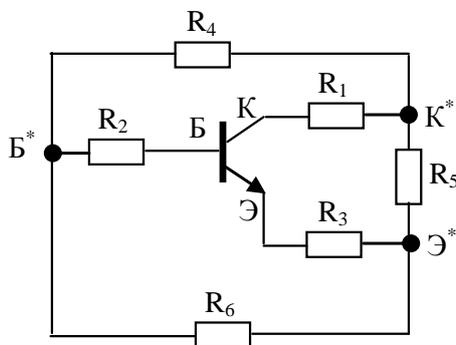


Рис. 2.22. Модель неисправностей транзистора

Например, при исправном транзисторе принимается, что $R_1 = R_2 = R_3 = 0,01$ Ом и $R_4 = R_5 = R_6 = 100$ Мом. Изменения в сопротивлениях резисторов при той или иной неисправности приведены в табл. 2.4.

Модификация сопротивлений при неисправностях

Неисправности	Значение сопротивления
Обрыв в цепи коллектора	$R_1 = 100 \text{ МОм}$
Обрыв в цепи базы	$R_2 = 100 \text{ МОм}$
Обрыв в цепи эмиттера	$R_3 = 100 \text{ МОм}$
Пробой перехода коллектор – база	$R_4 = 0,01 \text{ Ом}$
Пробой перехода коллектор – эмиттер	$R_5 = 0,01 \text{ Ом}$
Пробой перехода эмиттер – база	$R_6 = 0,01 \text{ Ом}$

Функциональные неисправности аналоговых устройств

Современные электронные устройства все чаще конструируются из микросхем высокого уровня интеграции. Детализация неисправностей микросхем при организации диагностирования таких устройств с практической точки зрения не имеет смысла. Поэтому для них зачастую рассматривают неисправности, которые можно отнести к категории функциональных.

Пусть устройство содержит множество $Q = \{q_1, \dots, q_r\}$ аналоговых элементов (микросхем) и в процессе тестовой проверки на его внешние входы могут подаваться тестовые воздействия (входные наборы) из множества $E = \{e_1, e_2, \dots\}$. Для простоты изложения примем, что все элементы $q_i \in Q$ являются одновыходными, и обозначим через u_{ij}^0 номинальное значение сигнала на выходе элемента $q_i \in Q$ на входном наборе $e_j \in E$ в исправном устройстве.

Неисправности аналоговых элементов часто приводят к изменению коэффициентов усиления элементов, фиксации значения выходного сигнала или его смещению на некоторую величину. Для описания таких неисправностей введем четыре типа моделей: s_A, s_B, s_C, s_D . Значение на выходе элемента q_i на входном наборе e_j при неисправности типа $s \in \{s_A, s_B, s_C, s_D\}$ обозначим через u_{ij}^s .

Модели неисправностей определим следующим образом.

Модель s_A : $u_{ij}^A = \text{const}$ для любого $e_j \in E$.

Модель s_B : $u_{ij}^B = u_{ij}^0 + d_i$, где $d_i = \text{const}$ для любого $e_j \in E$.

Модель s_C : $u_{ij}^C = u_{ij}^0 \varphi_i$, где $\varphi_i = \text{const}$ для любого $e_j \in E$.

Модель s_D : $u_{ij}^D = u_{ij}^0 \varphi_i + d_i$, где $\varphi_i = \text{const}$ и $d_i = \text{const}$ для любого $e_j \in E$.

Модель типа s_A позволяет описать поведение элемента при неисправностях, которые приводят к фиксации значения его выходного сигнала. **Модель s_B** отображает те неисправности, при которых значение выходного сигнала имеет фиксированное смещение по отношению к номинальному значению сигнала исправного элемента. Дефекты, приводящие к изменению коэффициента усиления, описывает **модель s_C** . И, наконец, **модель s_D** представляет собой комбинацию моделей s_B и s_C .

Имеются микросхемы, которые работают одновременно с аналоговыми и цифровыми сигналами. В этом случае, для каждого типа сигналов применяются свои модели.

Неисправности микропроцессорных систем и матриц памяти

В некоторых случаях для микропроцессорного ОД известна достаточно подробная электрическая схема на уровне логических элементов. Тогда оправдано использование константной модели неисправностей. Правда надо иметь в виду, что придется иметь дело со списком возможных неисправностей в десятки-сотни тысяч неисправностей.

Гораздо чаще разработчикам электронных изделий известна упрощенная схема ОД, состоящая из ряда функционально законченных узлов с указанием связей между ними и реализуемых функций. Так, если ОД представляется в виде графа регистровых передач (как на рис. 2.8), то рассматриваются неисправности, которые можно назвать функциональными, так как они не привязаны к какой-то конкретной микросхеме, а отражают нарушения при передаче и обработке информации в графе регистровых передач. При этом для каждого узла (механизма, функции) может формироваться свой класс неисправностей.

Например, в одной из методик построения тестов микропроцессорных устройств, в которой ОД рассматривается на структурно-функциональном уровне, при неисправности механизма выборки регистра, происходит перенос данных от источника не в заданный регистр, а в некоторое произвольное множество регистров; неисправности механизма адресации приводят к пропаданию, сдвигу во времени, а также к выработке неправильных управляющих сигналов; при неисправностях механизма дешифрации вместо операции I_k могут выполняться операции некоторого подмножества \mathfrak{Z}_k , причем итоговый результат одновременно выполнения операций подмножества \mathfrak{Z}_k формируется путем поразрядного логического сложения результатов выполнения всех операций; и т.п.

Многие современные программно-управляемые устройства содержат большие интегральные схемы (БИС) и сверхбольшие интегральные схемы (СБИС). Даже простейший микропроцессор – это БИС. При этом разработчики устройств обычно не имеют детальных электрических схем таких микросхем. Тогда для целей диагностирования создаются тестовые программы, направленные на проверку правильности работы ОД в типовых и критичных режимах, правильности выполнения некоторых последовательностей команд. При этом фактически нет необходимости перечислять возможные неисправности, и, по сути, решается задача проверки правильности функционирования. Для организации контроля достаточно знать правильный результат выполнения созданной тестовой программы. Любое отклонение свидетельствует о неисправности ОД, но, что конкретно неисправно при таком подходе, указать трудно. Очевидно, качество диагностирования при таком подходе фактически неизвестно.

Обязательными компонентами микропроцессорных объектов является наличие микросхем с памятью большого объема. Для сокращения числа внешних выводов и упрощения адресации они обычно имеют матричную структуру (рис. 2.23).

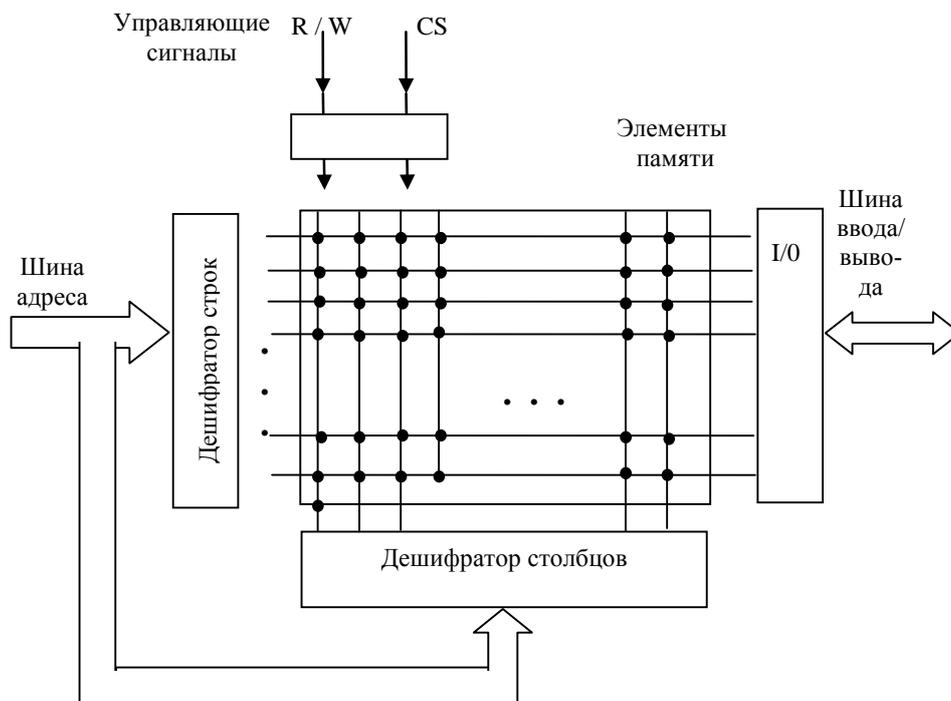


Рис. 2.23. Структура микросхемы памяти

Если рассматривать кристалл микросхемы как логическое устройство, то в нём можно выделить адресные дешифраторы строк и столбцов, интерфейс ввода/вывода I/O, матрицу запоминающих элементов. Последняя представляет собой двумерную матрицу $n \times n$ бит, в узлах которой располагаются ячейки памяти. Выбор ячейки осуществляется путем указания соответствующей строки и столбца матрицы с помощью дешифраторов строк и столбцов. Линия R/W служит для выбора режима работы: чтение или запись, линия CS – разрешение чтения/записи.

Для дешифраторов, линий управления, шин ввода/вывода обычно рассматривают уже упомянутые выше неисправности константного типа.

Неисправности матрицы запоминающих элементов можно разделить на две категории: константные неисправности и кодозависимые неисправности ячеек памяти. Константные неисправности запоминающих элементов (ячеек) – это фиксации в значениях «0» или «1» разрядов этих элементов. Независимо от того, что записывалось в ячейку, результат считывания – фиксированное значение.

Кодозависимыми неисправностями называют неисправности, при которых содержание некоторого запоминающего элемента изменяется в зависимости от значений соседних элементов. В том случае, когда соседними элементами считаются смежные элементы, неисправности называют смежными кодозависимыми неисправностями, а если в качестве соседних принимают элементы столб-

ца (строки), то неисправности называют столбцовыми (строчными) кодозависимыми неисправностями.

По характеру возникновения кодозависимых неисправностей различают **статические кодозависимые неисправности**, которые проявляются как зависимость от значений, ранее записанных в соседние элементы, и **динамические кодозависимые неисправности**, которые проявляются при изменении значений в соседних элементах. Проверка подобных неисправностей существенно удлиняет тесты, так как надо не только проверять правильность записи и чтения в отдельную ячейку, но также делать проверки при разных значениях в окружающих ячейках.

Неисправности в логической модели аналогового устройства

В случае использования логической модели аналогового устройства неисправности ОД должны удовлетворять следующим требованиям.

1. Элемент считается неисправным, если при допустимых входных сигналах на его выходе появляется сигнал, значение которого отличается от допустимого.
2. При выходе за пределы допустимых значений хотя бы одного из входных сигналов появляется выходной сигнал, отличающийся от допустимого.

Фактически в логической модели каждому выходу элемента сопоставляется только одна неисправность, состоящая в том, что выходной сигнал отличается от допустимого.

Неисправности динамических систем

Обычно здесь используются неявные модели неисправностей, т.е. по модели исправного объекта (изменяя ее константы и/или переменные) можно получить модель неисправности. При этом, очевидно, особенности преобразования в модель неисправности зависят от выбранной формы описания исправного объекта.

Рассмотрим особенности формирования моделей неисправностей при представлении исправного объекта в виде передаточных функций с использованием преобразований Лапласа и в пространстве состояний.

Пусть описание исправного объекта с помощью передаточной функции имеет вид:

$$W(p) = \frac{b_m p^m + \dots + b_1 p + b_0}{p^n + \dots + a_1 p + a_0} = \frac{B(p)}{A(p)}.$$

Понятно, что при появлении какой-либо i -й неисправности, где i – ее условный номер, объект будет описываться иной передаточной функцией. Обычно в линейных динамических системах передаточную функцию неисправного объекта можно представить функцией:

$$W_i(p) = B_i(p) / A_i(p) \text{ или } W_i(p) = [B(p) \pm \Delta B_i(p)] / [A(p) \pm \Delta A_i(p)],$$

где $\Delta A_i(p)$ и $\Delta B_i(p)$ – отклонение от нормальных значений числителя и знаменателя передаточной функции при i -й неисправности.

Пример: Пусть объект (рис. 2.24) представляет собой два последовательно соединенных элемента, являющихся аperiodическими звеньями первого поряд-

ка (т.е. они описываются дифференциальными уравнениями с производными первой степени). Схема имеет один вход (x) и один выход (y), передаточные функции элементов приведены внутри графических изображений звеньев.

Передаточная функция последовательно соединенных элементов равна произведению их передаточных функций. Тогда передаточная функция схемы рис. 2.24:

$$W(p) = \frac{1}{p+1} \times \frac{p+3}{p+2} = \frac{p+3}{p^2+3p+2}.$$

Пусть в схеме рис. 2.24 имеет место неисправность – изменение коэффициента обратной связи (K_{OC}) в элементе \mathcal{E}_2 . Такая неисправность возможна, например, если элемент \mathcal{E}_2 реализован с использованием операционного усилителя с обратной связью. Значение K_{OC} определяется вторым слагаемым знаменателя передаточной функции элемента \mathcal{E}_2 . В исправном состоянии $K_{OC}=1$. Предположим, что в результате неисправности $K_{OC}=2$. При такой неисправности объект описывается передаточной функцией:

$$W_1(p) = \frac{1}{(p+2)} \times \frac{p+3}{p+2} = \frac{p+3}{p^2+4p+4} = \frac{p+3}{(p^2+3p+2)+(p+2)}.$$

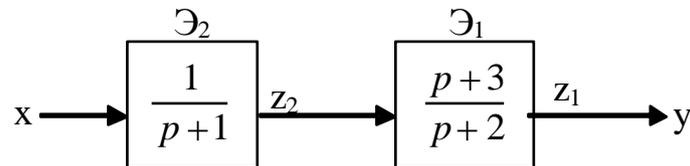


Рис. 2.24. Динамическая система с двумя звеньями первого порядка

По последней формуле видно, что передаточную функцию схемы с рассматриваемой неисправностью можно представить в соответствии с ранее приведенной формулой $W_i(p)=[B(p)+\Delta B_i(p)]/[A(p)+\Delta A_i(p)]$. В нашем случае $\Delta B_1(p)=0$ и $\Delta A_1(p)=p+2$.

Рассмотрим теперь особенности представления неисправностей в **пространстве состояний**.

Ранее указывалось (см. пп. 2.1), что описание исправного линейного объекта в пространстве состояний (матричной форме) имеет вид

$$\dot{Z} = AZ + BX, Y = CZ + DX.$$

При такой форме модели объекта действие i -й неисправности можно описать изменениями значений элементов в матрицах A , B , C , D . Иными словами, неисправный объект в матричной форме описывается уравнениями

$$\dot{Z} = [A+\Delta A_i] Z + [B+\Delta B_i] X, Y = [C+\Delta C_i] Z + [D+\Delta D_i] X,$$

где ΔA_i , ΔB_i , ΔC_i , ΔD_i – изменения матриц, вызванные действием i -й неисправности.

В качестве примера рассмотрим объект рис. 2.17. Он описывается матричными уравнениями:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 10 \\ 01 \\ 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 100 \\ 011 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}.$$

Предположим, что k -й неисправности соответствует неисправность «обрыв линии **a**» (рис. 2.17), связывающей вход x_1 с нижним сумматором и через него с элементом A_1 . В исходном уравнении эта линия задается коэффициентом $b_{11}=1$ матрицы B . При обрыве $b_{11}=0$, т.е. $\Delta b_{11}=-1$. Таким образом, при k -й неисправности, матрицу коэффициентов B_k можно представить как следующие изменения матрицы B :

$$B_k = B + \Delta B_k = \begin{bmatrix} 10 \\ 01 \\ 00 \end{bmatrix} + \begin{bmatrix} -1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 00 \\ 01 \\ 00 \end{bmatrix}.$$

Остальные матрицы коэффициентов не изменятся. Таким образом, для рассматриваемой неисправности изменения матриц коэффициентов:

$$\Delta A_k = 0, \Delta B_k = \begin{bmatrix} -1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \Delta C_k = 0, \Delta D_k = 0.$$

В итоге, имеем $\dot{Z}_k = A_k Z_k + B_k X = A Z_k + B_k X$ и $Y_k = C_k Z_k + D_k X = C Z_k + D X$.

Рассмотренное описание неисправности приводит к дискретным изменениям в матрицах коэффициентов.

Более общий способ описания неисправности – применение **сигнальной модели** неисправности.

Пусть в одном из внутренних элементов системы произошла неисправность. При этом выходной сигнал неисправного элемента $z^*(t) \neq z(t)$, где $z(t)$ – выходной сигнал исправного элемента (рис. 2.25 а).

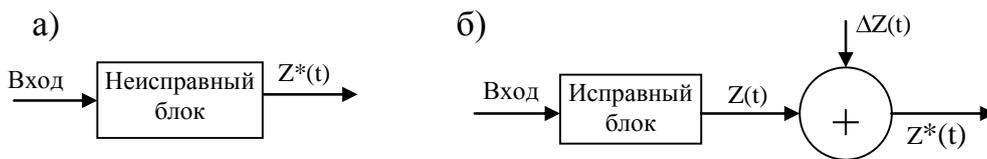


Рис. 2.25. Сигнальная модель неисправности

В сигнальной модели предполагается, что сигнал при неисправности можно представить в виде $z^*(t) = z(t) + \Delta z(t)$, где $\Delta z(t)$ – составляющая, вызванная дей-

ствием неисправности. В результате неисправный элемент можно представить из исправного элемента и дополнительного сумматора, на вход которого поступает $\Delta z(t)$ (рис. 2.25 б).

Переходя к обобщению, можно сказать, что неисправный объект диагностирования (ОД) в сигнальной модели можно представить как исправный, у которого появились дополнительные входы (рис. 2.26), на которые подаются заранее неизвестные сигналы $V(t)$, недоступные для измерения по сравнению с входными сигналами $X(t)$.

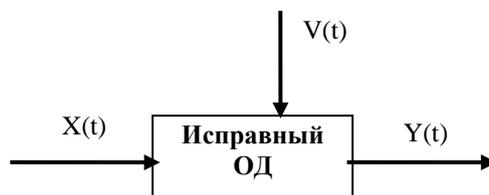


Рис. 2.26. Обобщенная сигнальная модель неисправного объекта

В пространстве состояния такой объект описывается уравнениями

$$\dot{Z} = AZ + BX + LV, \quad Y = CZ + DX + KV,$$

где L и K – матрицы, описывающие действие неисправностей, причем число столбцов матриц определяется числом неисправностей.

Пусть в схеме рис. 2.17 имеет место неисправность элемента A_2 . Если неисправность приводит к искажению выходной переменной, то ее новое значение $z_2^* = z_2 + v_2$, где v_2 – составляющая, вызванная действием неисправности. Таким образом, описание неисправного объекта представляется в виде

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 + v_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 10 \\ 01 \\ 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 100 \\ 011 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 + v_2 \\ z_3 \end{bmatrix}.$$

Выполняя стандартные преобразования с матрицами эти матричные уравнения можно записать в виде:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 10 \\ 01 \\ 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} v_2, \quad \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 100 \\ 011 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} v_2.$$

Представленные в правых частях уравнений матрицы $L = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}$ и $K = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ ото-

бражают действие неисправности на объект. С использованием данной матричной формы можно описать одновременно несколько неисправностей. Тогда матрицы L и K будут иметь столько столбцов, сколько неисправностей.

2.3. Таблица функций неисправностей и таблица неисправностей

Таблица функций неисправностей (ТФН) и таблица неисправностей (ТН) – это специальные явные формы представления поведения ОД в исправном и неисправном состояниях.

Методы решения задач диагностирования на основе ТФН и ТН довольно просты, но их построение требует значительных затрат вычислительных ресурсов. Методы на основе ТФН и ТН обычно целесообразно применять при среднем и крупносерийном производстве электронной техники, когда может быть выделено ограниченное число возможных неисправностей, а затраты на подготовку необходимой информации окупаются за счет диагностирования большого числа однотипных объектов. Кроме этого, ТФН и ТН очень наглядны и удобны для иллюстраций результатов теоретических исследований.

Таблица функций неисправностей

ТФН содержит сведения о поведении исправного ОД, а также ОД с каждой из допустимых неисправностей.

Введем следующие обозначения:

$S = \{ s_1, \dots, s_r \}$ – множество допустимых неисправностей, $r = 1, \dots, r$;

$W = \{ w_1, \dots, w_m \}$ – множество контролируемых параметров (КП);

$E = e_1, \dots, e_n$ – последовательность проверок, выполняемых над объектом.

Структура ТФН приведена в табл. 2.5. Она состоит из $r+1$ подтаблиц (далее матриц) M_0, M_1, \dots, M_r . Строки ТФН соответствуют отдельным проверкам последовательности $E = e_1, \dots, e_n$. Матрица M_i ($i = 0, 1, \dots, r$) задает поведение ОД в исправном (s_0) и каждом неисправном (s_1, \dots, s_r) состояниях. Число столбцов каждой матрицы равно числу контролируемых параметров (КП) в множестве W , причем i -й столбец сопоставлен $w_i \in W$.

На пересечении k -й строки и l -го столбца матрицы M_i ($i = 0, 1, \dots, r$) проставляется допустимое значение t_i^{kl} параметра w_l на наборе e_k при диагностировании ОД, находящегося в техническом состоянии s_i . В случаях цифровых КП обычно значение $t_i^{kl} \in \{0, 1\}$, при аналоговых КП значение параметра – некоторый непрерывный интервал или номинальное значение. Для интервала обычно указываются значения его верхней и нижней границ. Иногда для задания значений аналоговых КП используется многозначный конечный алфавит, в котором каждому символу сопоставляется определенный интервал значений.

Структура таблицы функций неисправностей

Проверки	M ₀			M ₁			...			M _r		
	w ₁	...	w _m	w ₁	...	w _m				w ₁	...	w _m
e ₁	t ₀ ¹¹		t ₀ ^{1m}	t ₁ ¹¹		t ₁ ^{1m}				t _r ¹¹		t _r ^{1m}
•	•		•	•		•				•		•
•	•		•	•		•		...		•		•
•	•		•	•		•				•		•
e _n	t ₀ ⁿ¹		t ₀ ^{nm}	t ₁ ⁿ¹		t ₁ ^{nm}				t _r ⁿ¹		t _r ^{nm}

На рисунке 2.27 приведена ТФН для схемы из одного двухвходового цифрового элемента «И» в предположении, что у него возможны одиночные неисправности константного типа, т.е. неисправности «константа 0» или «константа 1» на выводах элемента. Контролируемым выходом является вывод 3 элемента. В ТФН неисправность «константа 0(1)» на j-м выводе обозначена s_{j-0(1)}. Входные сигналы элемента обозначены символами a и b.

В данном примере каждая матрица M_i (i = 0, 1, ..., 6) представляет один столбец, так как у элемента только один контролируемый выход (вывод 3). Элемент матрицы M_i равен 0(1), если на соответствующем входном наборе значение сигнала на выводе 3 равно 0(1).

При наличии двоичной ТФН процессы контроля и поиска места неисправности могут быть организованы следующим образом.

Входы	M ₀	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆
a b	s ₀	s ₁₋₁	s ₁₋₀	s ₂₋₁	s ₂₋₀	s ₃₋₁	s ₃₋₀
0 0	0	0	0	0	0	1	0
0 1	0	1	0	0	0	1	0
1 0	0	0	0	1	0	1	0
1 1	1	1	0	1	0	1	0

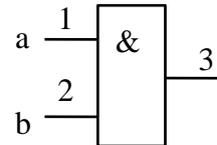


Рис. 2.27. Таблица функций неисправностей элемента «И»

Контроль. Подаем заданные в ТФН тестовые воздействия E и измеряем контролируемые параметры W. По результатам измерений формируем матрицу M_x такого же формата, что и матрицы M₀. Сравниваем поэлементно матрицы M_x и M₀. Если M_x = M₀, то считаем ОД исправным. Если M_x ≠ M₀, то ОД неисправен.

Поиск неисправности. Для определения конкретной неисправности ОД выполняем следующие действия. Матрицу M_x поочередно сравниваем с каждой M_i, где i = 1, ..., r. Если M_x = M_i, то неисправность s_i заносим в список подозреваемых неисправностей (СПН).

В этот СПН может попасть несколько неисправностей, при которых ОД имеет одну и ту же выходную реакцию. Так, в примере на рис. 2.27 дают одинаковые вы-

ходные сигналы каждая из неисправностей s_{1-0} , s_{2-0} и s_{3-0} . Иными словами, при наличии, скажем, неисправности s_{2-0} в СПН попадут три неисправности: s_{1-0} , s_{2-0} и s_{3-0} .

Чтобы сократить объем ТФН (и ТН), неисправности с одинаковой выходной реакцией объединяются в группы эквивалентных неисправностей и под такую группу формируется только одна матрица. Обычно оставляют неисправность, сопоставленную выходу элемента.

Для цифровых ОД с памятью часто применяют троичные ТФН с алфавитом $\{0,1,x\}$, где $x = \{0,1\}$. Символ «x» применяется для обозначения случаев, когда возможна неоднозначность поведения ОД, например, из-за состязаний или поведения элементов памяти вследствие неисправностей или схемных особенностей.

В таблице 2.6 дан пример троичной ТФН для случаев $E = e_1, e_2, e_3$; $W = \{w_1, w_2, w_3\}$ и $S = \{s_1, \dots, s_4\}$.

Таблица 2.6

Пример троичной ТФН

E	M_0	M_1	M_2	M_3	M_4
e_1	1 0 x	0 0 1	0 0 1	1 0 1	0 0 1
e_2	x 1 1	x 1 1	1 1 1	1 1 1	1 x 1
e_3	0 x x	x x 1	0 1 1	0 x 1	x 1 1

Таблица неисправностей

При решении некоторых задач диагностирования надо знать, на каких проверках и какие КП отличаются от нормальных (эталонных) при той или иной неисправности. В этом случае ТФН не очень удобна, так как для выявления вышеуказанных ситуаций надо проводить сравнение значений в M_0 и в соответствующей матрице M_i . Наглядное представление о проверках и КП, на которых значения сигналов каждой допустимой неисправности отличаются от нормальных, дает **таблица неисправностей**.

Структура ТН дана в табл. 2.7.

Таблица 2.7

Структура таблицы неисправностей

Проверки	\tilde{M}_1				\tilde{M}_r		
	w_1	...	w_m		w_1	...	w_m
e_1	\tilde{t}_1^{11}		\tilde{t}_1^{1m}		\tilde{t}_r^{11}		\tilde{t}_r^{1m}
•	•		•		•		•
•	•		•	•••	•		•
•	•		•		•		•
e_n	\tilde{t}_1^{n1}		\tilde{t}_1^{nm}		\tilde{t}_r^{n1}		\tilde{t}_r^{nm}

Она отличается от ТФН, во-первых, отсутствием матрицы, описывающей исправный ОД. Во-вторых, значение элемента \tilde{t}_i^{kl} в ее матрице \tilde{M}_i ($i=1, \dots, r$) представляет собой результат сравнения элементов t_i^{kl} и t_o^{kl} из ТФН. В случае цифрового КП значение \tilde{t}_i^{kl} (при двоичном и трехзначном алфавите кодирования сигналов) определяется в соответствии с табл. 2.8.

Таблица 2.8

		t_o^{kl}		
		0	1	x
t_i^{kl}	0	0	1	x
	1	1	0	x
	x	x	x	x

Заметим, что символ «x» применяют для обозначения ситуаций, когда точно нельзя предсказать, какое будет в данной точке значение: «0» или «1». Такое может быть, например, если возможны состязания (более подробно о состязаниях см. в подразделе 3.5) или если при включении питания элементы памяти устанавливаются в случайное состояние.

Варианты кодирования результатов сравнения для аналогового КП более разнообразны. Так, можно использовать трехзначный алфавит $\{0, 1, x\}$, присваивая \tilde{t}_i^{kl} значение 1, x или 0, если интервалы t_o^{kl} и t_i^{kl} не пересекаются, частично пересекаются или все значения t_i^{kl} входят в t_o^{kl} . Можно \tilde{t}_i^{kl} присваивать значения 1, 2 или 0, если значения t_i^{kl} меньше, больше или входят в t_o^{kl} . Применяются и другие алфавиты.

Применение (в аналоговом случае) алфавитов с ограниченным числом символов для кодирования значений элементов матриц позволяет сократить объем ТФН и ТН. Однако очевидно, что это сокращение происходит за счет загробления первоначальных данных.

Пример ТН для цифрового элемента «ИЛИ-НЕ» приведен на рис. 2.28.

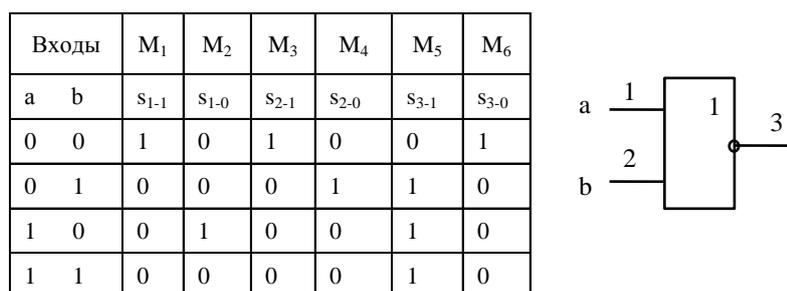


Рис. 2.28. ТН для элемента «ИЛИ-НЕ»

В ней элементы матриц принимают значения из алфавита $\{0,1\}$. В этом случае значение элемента \tilde{t}_i^{kl} можно получить по формуле $\tilde{t}_i^{kl} = t_o^{kl} \oplus t_i^{kl}$.

ТН удобна для использования с целью определения возможных неисправностей.

В методическом плане поиск неисправности с использованием двоичной ТН выполняется аналогично тому, как это делается с использованием двоичной ТФН. Единственное отличие состоит в том, что матрица M_x предварительно заменяется на матрицу $\tilde{M}_x = M_x \oplus M_o$. В матрице \tilde{M}_x , как и в любой матрице \tilde{M}_i ($i=1, \dots, r$), значение «1» означает, что при неисправности s_i на соответствующих входном наборе и контролируемом выходе в исправной и неисправной схеме разные значения.

Особенности построения ТФН и ТН для аналоговых устройств

Для аналоговых ОД, кроме приведенных выше форм ТФН и ТН, возможны и другие формы этих таблиц.

В таблице 2.9 представлена ТФН для модели неисправностей резистора, приведенной в пп. 2.2. Она построена в предположении, что $\Pi = [\pi_1, \pi_2, \dots, \pi_n]$ – множество элементарных проверок; $S = \{s_0, s_1, \dots, s_r\}$ – множество технических состояний, где s_0 – исправное состояние и s_i – неисправное состояние ($i=1, \dots, r$); $I = \{H, N, N(\emptyset, \infty)\}^1$ – возможные результаты измерений.

Таблица 2.9

ТФН для неисправностей резистора

Элементарные проверки	Состояние СД	Состояние объекта диагностирования											
		R=R _n	R<	R>	R<<	R>>	R=0	R=∞	R=L	R=C	R= \vec{VD}	R= \overleftarrow{VD}	R= \overleftrightarrow{VD}
1	2	3	4	5	6	7	8	9	10	11	12	13	14
π_1	H	1	1	1	0	0	0	0	0	0	0	0	0
	\emptyset	0	0	0	1	0	1	0	1	0	0	0	0
	∞	0	0	0	0	1	0	1	0	1	1	1	1
π_2	H	1	0	0	0	1	0	0		0	0	0	0
	\emptyset	0	1	1	1	0	1	0	1	0	0	0	0
	∞	0	0	0	0	0	0	0	0	1	1	1	1
π_3	H	1	0	0	1	0	0	0	0	0	0	0	0
	\emptyset	0	0	0	0	0	1	0	1	0	0	0	0
	∞	0	1	1	0	1	0	1	0	1	1	1	1

¹ Физический смысл этих значений дан в пп. 2.2.

1	2	3	4	5	6	7	8	9	10	11	12	13	14
π_4	H	1	0	0	0	0	0	0	0	0	0	1	1
	\emptyset	0	0	0	1	0	1	0	1	0	0	0	0
	∞	0	1	1	0	1	0	1	0	1	1	0	0
π_5	H	1	0	0	0	0	0	0	0	0	1	0	1
	\emptyset	0	0	0	1	0	1	0	1	0	0	0	0
	∞	0	1	1	0	1	0	1	0	1	0	1	0
π_6	H	1	0	0	0	0	0	0	1	1	0	0	0
	\emptyset	0	0	0	0	0	1	0	0	0	0	0	0
	∞	0	1	1	1	1	0	1	0	0	1	1	1

В таблице 2.9 каждой элементарной проверке π_k сопоставлено 3 строки для результатов измерений H, \emptyset и ∞ соответственно (H – значение сопротивления резистора в заданных пределах; \emptyset – ниже заданного нижнего диапазона измерения; ∞ – выше заданного верхнего значения диапазона измерения). Значение «1» в ячейке таблицы означает, что результат измерения показывает на возможность неисправности, указанной в заголовке соответствующего столбца. Каждой неисправности сопоставлен один столбец, так как в данном случае один контролируемый параметр (предполагается, что в ходе диагностирования измеряется сопротивление).

Первый столбец в таблице ($R=R_H$) задает поведение исправного объекта, а остальные столбцы – сопоставлены неисправным состоянием. Таблица 2.9 построена для следующих проверок:

π_1 – подача на резистор постоянного напряжения с амплитудой $U=0,1$ В и включение на измерительном приборе диапазона измерения, включающего нормальные значения резистора;

π_2 – подача на резистор постоянного напряжения с амплитудой $U=0,1$ В и включение максимального диапазона измерения значений сопротивлений;

π_3 – подача на резистор постоянного напряжения с амплитудой $U=0,1$ В и включение минимального диапазона измерения значений сопротивлений;

π_4 – подача на резистор постоянного напряжения с амплитудой $U=+1$ В и включение диапазона измерений до 100 Ом;

π_5 – подача на резистор постоянного напряжения с амплитудой $U=-1$ В и включение диапазона измерений до 100 Ом;

π_6 – подача на резистор переменного напряжения с амплитудой $U=0,1$ В и последовательное включение всех диапазонов измерений.

В таблице 2.9 отражены результаты измерений в трехзначном алфавите $I=\{H, \emptyset, \infty\}$. Для заданных проверок оказывается, что на проверке π_1 исправное состояние отличается от всех неисправных состояний за исключением состояний $R<$ и $R>$. По таблице 2.9 видно, что для установления факта исправности элемента достаточно двух проверок, например π_1 и π_2 . Для различения неисправностей надо ис-

пользовать все проверки π_1 - π_6 . При этом можно воспользоваться информацией в виде табл. 2.10 (словарь дефектов), которая, по сути, является таблицей неисправностей. Таблица имеет небольшой объем, так как в данном случае для различения неисправностей достаточно рассматривать двухзначный алфавит результатов $I=\{\emptyset, \infty\}$.

Таблица 2.10

Словарь дефектов

Технические состояния	Элементарные проверки							
	π_2		π_4		π_5		π_6	
	\emptyset	∞	\emptyset	∞	\emptyset	∞	\emptyset	∞
R=0	1	0	1	0	1	0	1	0
R=L	1	0	1	0	1	0	0	0
R<<	1	0	1	0	1	0	0	1
R>>	0	0	0	1	0	1	0	1
R= ∞	0	1	0	1	0	1	0	1
R=C	0	1	0	1	0	1	0	0
R= \overrightarrow{VD}	0	1	0	1	0	0	0	1
R= \overleftarrow{VD}	0	1	0	0	0	1	0	1
R= \overleftrightarrow{VD}	0	1	0	0	0	0	0	1

В принципе, ТФН может быть построена для любой модели объекта, в которой можно выделить элементы и задать множество допустимых неисправностей. Так, в разделе 2.2 была рассмотрена модель неисправностей при использовании логической модели аналогового устройства. Введем обозначения: 0 – недопустимое значение, 1 – допустимое значение, z_i – выход i -го элемента функциональной схемы, s_j – неисправность j -го элемента функциональной схемы, s_0 – исправное техническое состояние (в этом состоянии все элементы исправны и сигналы на всех линиях являются допустимыми).

При вышеприведенных допущениях ТФН для неисправностей элементов логической модели можно представить в виде таблицы, столбцы которой соответствуют выходам элементов, а строки – техническим состояниям s_0, s_1, \dots, s_r (r – число элементов). На пересечении s_j строки и z_i столбца проставляется 0 (1), если при неисправности s_j значение z_i недопустимо (допустимо). Очевидно, что в строке s_j в столбце z_i должно быть значение «0». Например, для модели рис. 2.16 ТФН будет иметь вид, представленный табл. 2.11.

Логическая модель очень проста. Для конкретной логической модели фактически существует только одно входное воздействие (на внешних входах все единицы), а число одиночных неисправностей фактически равно числу выходов элементов. Это, конечно, ограничивает ее возможности для целей диагностики. Повысить эффективность можно, если строить свою модель для каждого отдельного режима работы ОД.

Однако эта ТФН достаточно эффективно может использоваться для построения дерева поиска неисправностей, т.е. алгоритма поиска места неисправности.

Таблица 2.11

Таблица функций неисправностей для модели рис. 2.16

Техническое состояние	Выход элемента								
	z_1	z_2	z_{31}	z_{32}	z_4	z_5	z_6	z_7	z_8
s_0	1	1	1	1	1	1	1	1	1
s_1	0	0	0	0	0	0	1	0	1
s_2	1	0	0	0	0	0	1	0	1
s_{31}	1	1	0	0	0	0	1	0	1
s_{32}	1	1	0	0	0	0	1	0	1
s_4	1	1	1	1	0	0	1	1	1
s_5	1	1	1	1	1	0	1	1	1
s_6	1	0	0	0	0	0	0	0	1
s_7	0	0	0	0	0	0	1	0	1
s_8	1	1	1	1	0	0	1	1	0

2.4. Проверяемость и различимость неисправностей

Формальные определения

При подборе воздействий, обеспечивающих обнаружение неисправности, необходимо сформулировать условия обнаружения. Если же решается задача поиска места неисправности, то надо уметь различать поведение ОД при разных неисправностях.

Сформулируем формальные условия проверки (обнаружения) и различимости неисправностей.

Предварительно введем операцию пересечения троичных матриц $C = A \cap B$, где значение элемента $c_{ij} = a_{ij} \cap b_{ij}$ определяется по нижеприведенным правилам (табл. 2.12).

Операция пересечения

		b_{ij}		
		0	1	x
a_{ij}	0	0	\emptyset	0
	1	\emptyset	1	1
	x	0	1	x

Будем говорить, что:

- 1) $C = \emptyset$ (C равно пусто), если в матрице C хотя бы один элемент $c_{ij} = \emptyset$;
- 2) $A \subset B$ (A принадлежит B), если A можно получить из B путем фиксации всех или некоторых «x» матрицы B константами 0 и 1.

Например, матрица $C = \emptyset$, если

$$C = \begin{vmatrix} 0 & 1 \\ 1 & x \end{vmatrix} \cap \begin{vmatrix} 0 & x \\ 0 & 0 \end{vmatrix} = \begin{vmatrix} 0 & 1 \\ \emptyset & 0 \end{vmatrix}.$$

Матрица $A \subset B$, если

$$A = \begin{vmatrix} 0 & 1 \\ 1 & x \end{vmatrix}, B = \begin{vmatrix} x & 1 \\ 1 & x \end{vmatrix}.$$

Контроль и поиск неисправностей с использованием троичной ТФН можно организовать следующим образом.

Подаем на ОД последовательность (тест) E и измеряем контролируемые параметры из W. По результатам измерений формируем матрицу измерений M_x (заметим, что в M_x всегда конкретные значения 0 или 1 и нет значений «x»).

Контроль. Проверяем принадлежность $M_x \subseteq M_0$. Если да, то ОД исправен. Если $M_x \not\subseteq M_0$, то ОД считаем неисправным.

Поиск неисправности. Поочередно для всех $k = 1, \dots, r$ проверяем принадлежность $M_x \subseteq M_k$. Неисправность s_k включаем в СПН, если $M_x \subseteq M_k$.

Пример. Пусть в ОД (табл. 2.6) имеется неисправность s_4 и в результате измерений получена матрица

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

Выполним процедуру контроля. Объект неисправен, так как $M_x \not\subseteq M_0$. Выполним процедуру поиска неисправности. Поочередно проверяем принадлежность матрицы M_x для всех M_k ($k = 1, 2, 3, 4$). Получим $M_x \subset M_1$, $M_x \not\subset M_2$, $M_x \not\subset M_3$, $M_x \subset M_4$. Следовательно, в СПН войдут неисправности s_1 и s_4 .

Одними из основополагающих понятий в технической диагностике являются понятия о проверке и различимости неисправностей. Рассмотрим сначала эти понятия применительно к цифровым ОД, а затем распространим их на объект с аналоговыми контролируруемыми параметрами (КП).

Определение 2.1. Неисправность $s_k \in S$ называется **проверяемой**, если $M_o \cap M_k = \emptyset$.

В примере табл. 2.6 к проверяемым неисправностям относятся s_1, s_2 и s_4 , так как $M_o \cap M_1 = \emptyset, M_o \cap M_2 = \emptyset$ и $M_o \cap M_4 = \emptyset$.

Практически это означает, что при наличии в ОД одной из неисправностей s_1, s_2 или s_4 значение КП w_1 на e_1 будет отличаться от его значения в исправном ОД.

Определение 2.2. Неисправность $s_k \in S$ называется **непроверяемой**, если $M_k \subseteq M_o$.

В нашем примере к непроверяемым относится неисправность s_3 , так как $M_3 \subset M_o$.

Определение 2.3. Неисправность $s_k \in S$ называется **условно проверяемой**, если $M_o \cap M_k \neq \emptyset$ и $M_k \not\subseteq M_o$.

При условно проверяемой неисправности результаты измерений КП могут в некоторых случаях совпадать с возможными значениями исправного ОД, а в других – нет. Так, если матрицы M_o и M_k имеют вид (в примере предполагается, что столбцы сопоставлены КП w_1 и w_2 ; строки – проверкам e_1 и e_2)

$$M_o = \begin{vmatrix} 1 & x \\ x & 0 \end{vmatrix} \text{ и } M_k = \begin{vmatrix} x & 1 \\ x & 0 \end{vmatrix},$$

то неисправность s_k будет условно проверяемой. Если при измерении КП w_2 на e_1 будет получено значение «0», то можно обнаружить наличие неисправности (так как в исправном ОД должно быть «1»). Однако при значении «1» этого КП в неисправном ОД (что возможно), мы не сможем обнаружить неисправность.

Определение 2.4. Неисправности s_i и s_j называются **различимыми**, если $M_i \cap M_j = \emptyset$.

Для таблицы 2.6 различимыми являются пары неисправностей s_1 и s_3, s_2 и s_3, s_3 и s_4 .

Определение 2.5. Неисправности s_i и s_j называются **неразличимыми**, если $M_i = M_j$.

Определение 2.6. Неисправности s_i и s_j называются **условно различимыми**, если $M_i \cap M_j \neq \emptyset$ и $M_i \neq M_j$.

Примерами условно различимых пар неисправностей для табл. 2.6 являются s_1 и s_2, s_1 и s_4, s_2 и s_4 .

Проверяемость и различимость при цифровых сигналах

При наличии условно различимых неисправностей СПН существенно зависит от результатов измерения КП и от того, какая из условно различимых неисправностей фактически имеется в ОД.

Пусть в ОД (табл. 2.6) имеется неисправность s_4 и получена

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

Тогда в СПН попадут s_1, s_2 и s_4 . Если же

$$M_x = \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{vmatrix},$$

то в СПН войдут неисправности s_1 и s_4 .

Ниже, в табл. 2.13 и 2.14, приведены возможные комбинации значений элементов матриц ТФН и соответствующие им условия проверяемости или различимости.

Таблица 2.13

M_o	M_k	Проверяемость неисправности
0 1	1 0	Проверяемая
0 1	x x	Условно проверяемая
0 1 x x x	0 1 x 0 1	Непроверяемая

Таблица 2.14

M_i	M_j	Различимость неисправностей
0 1	1 0	Различимые
0 1 x x	x x 0 1	Условно различимые
0 1 x	0 1 x	Неразличимые

Неисправность является проверяемой (неисправности различимы), если в одноименных элементах матриц M_o и M_k (M_i и M_j) присутствует **хотя бы одна комбинация значений**, указанных в первой группе строк таблиц 2.13 и 2.14, и непроверяемой (неразличимыми) – если **все значения** из строки «непроверяемая» («неразличимая»). Если для сравниваемых элементов матриц **хотя бы один раз** имеет место комбинация значений из строки «условно проверяемая» («условно различимые»), а остальные значения из этих же строк и/или из строк «непроверяемая» («неразличимые»), то неисправность является условно проверяемой (неисправности условно различимые).

Проверяемость и различимость при аналоговых сигналах

Рассмотренные выше понятия проверки и различимости неисправностей могут быть распространены и на случай аналоговых КП. Однако при этом надо учитывать ту особенность, что аналоговый сигнал характеризуется некоторым интервалом допустимых значений. Поэтому, исследуя проверяемость и различимость неисправностей в аналоговых объектах, надо исходить из того, что возможные значения аналоговых КП представлены интервалами.

В таблице 2.15 показаны условия проверяемости для аналогового КП. В таблице через a_o и b_o , a_k и b_k обозначены нижняя и верхняя границы интервала возможных значений для исправного ОД и с неисправностью s_k соответственно.

Таблица 2.15

Условия проверяемости неисправности для аналоговых КП

Номер пункта	Соотношение интервалов	Проверяемость неисправности
1		Проверяемая
2	а) б) $a_o = a_k$	Непроверяемая
3	а) б)	Условно проверяемая

В таблице 2.16 показаны условия различимости неисправностей для аналоговых КП.

Таблицы 2.15 и 2.16 составлены в предположении, что объект имеет один контролируемый параметр и выполнена только одна проверка (т.е. матрица M_i имеет одну строку и один столбец).

Таблица 2.16

Условия различимости неисправностей для аналоговых КП

Номер пункта	Соотношение интервалов	Результат \cap -операции	Различимость неисправностей
1		\emptyset	Различимые
2			Неразличимые
3	а) б)		Условно различимые

Нетрудно заметить, что операции пересечения и принадлежности (\cap - и \subset -операции), использовавшиеся выше в определениях проверяемости и различимости неисправностей, легко развить на случай интервалов. Результат \cap -операции для интервалов показан в третьем столбце табл. 2.16, а операцию принадлежности интервала В интервалу А ($B \subset A$) можно сформулировать следующим образом: интервал $B \subset A$, если $B \cap A = A$.

С учетом введенных для интервалов операций пересечения и принадлежности сформулированные ранее формальные определения проверяемости и различимости неисправностей вполне применимы и для случая аналоговых сигналов.

Контрольные вопросы

1. В чем различие явной и неявной моделей?
2. Какая модель называется структурной?
3. Какая модель называется функциональной?
4. Какая модель называется структурно-функциональной?
5. Понятие «конечный автомат». Способы представления конечных автоматов.
6. Представьте функцию схемы рис. 2.29 в эквивалентной нормальной форме. Как она получается?

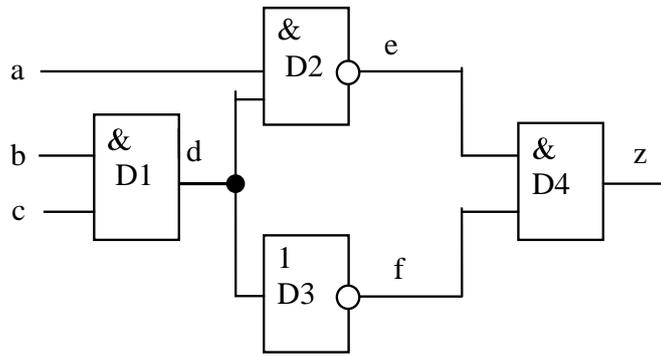


Рис. 2.29. Схема для контрольных вопросов 6, 8, 10, 11

7. Запишите общую формулу для булевой производной по переменной x_i для функции $F(x)$.
8. Определите булеву производную dz/dc для схемы рис. 2.29.
9. Что из себя представляет граф регистровых передач программно управляемого устройства?
10. Запишите сингулярное покрытие для элемента D2 схемы рис. 2.29.
11. Запишите d-кубы неисправностей для элемента D2 схемы рис. 2.29.
12. Составьте уравнения Кирхгофа для схемы рис. 2.30.

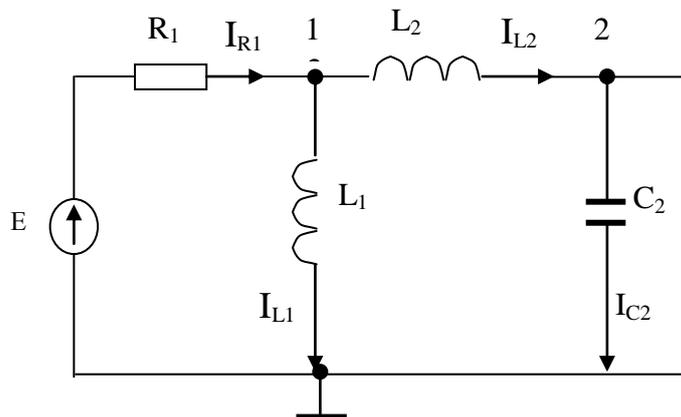


Рис. 2.30. Схема электрической цепи к контрольному вопросу 12

13. Условия представления объекта в виде логической модели.
14. Что представляет собой модель Эберса-Мола и какой электронный элемент она описывает?
15. Что представляет собой передаточная функция? Представьте функцию $y(t) = a_1 dz^3/dt + a_2 dz/dt + 1$ с использованием оператора дифференцирования.
16. Что описывает таблица функций неисправностей, ее структура?
17. Что описывает таблица неисправностей? В чем ее отличие от таблицы функций неисправностей?

18. Какую неисправность называют проверяемой, а какую непроверяемой?
 19. Какую неисправность называют условно проверяемой?
 20. Какие неисправности называют различимыми, а какие неразличимыми?
 21. Какие неисправности называют условно различимыми?
 22. Какой является неисправность s_k (проверяемой, непроверяемой, условно проверяемой), если в таблице функций неисправностей матрицы M_0 и M_k имеют вид, как показано ниже:

$$M_0 = \begin{vmatrix} 0 & 0 & x \\ 1 & 1 & 1 \\ 0 & x & 1 \end{vmatrix}, M_k = \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ x & x & 1 \end{vmatrix}.$$

23. Какими являются неисправности s_i и s_j (различимыми, неразличимыми, условно различимыми), если в таблице функций неисправностей матрицы M_i и M_j имеют вид, как показано ниже:

$$M_i = \begin{vmatrix} 1 & 0 & x \\ 1 & 1 & 1 \\ 0 & x & 1 \end{vmatrix}, M_j = \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ x & x & 1 \end{vmatrix}.$$

Рекомендуемая литература

Алексейчик, А. Применение автоматической оптической инспекции в общей стратегии тестирования на современном производстве / А. Алексейчик // Компоненты и технологии. – 2004. – № 9.

Байда, Н.П. Микропроцессорные системы поэлементного диагностирования РЭА / Н.П. Байда, И.В. Кузьмин, В.Т. Шпилевой. – М.: Радио и связь, 1987. – 256 с.

Бэндлер, Д.У. Диагностика неисправностей в аналоговых цепях / Д.У. Бэндлер, А.Э. Салама // ТИИЭР. – 1985. Т. 73. – № 8. – С. 35–87.

Карибский, В.В. Основы технической диагностики / В.В. Карибский, П.П. Пархоменко, Е.С. Согомонян, В.Ф. Халчев. Кн. 1. – М.: Энергия, 1976. – 464 с.

Мальшенко, Ю.В. Автоматизация диагностирования электронных устройств / Ю.В. Мальшенко, В.П. Чипулис, С.Г. Шаршунов. – М.: Энергоатомиздат, 1986. – 216 с.

Сигорский, В.П. Алгоритмы анализа электронных схем / В.П. Сигорский, А.И. Петренко. – М.: Сов. радио, 1976. – 608 с.

Чипулис, В.П. Анализ и построение тестов цифровых программно- управляемых устройств / В.П. Чипулис, С.Г. Шаршунов. – М.: Энергоатомиздат, 1992. – 224 с.

Malysenko Y. Functional Fault Models for Analog Circuits // IEEE Design&Test. – 1998. V. 15. – No. 2. – P. 80–85.

Thatte S.M., Abraham J.A. Test generation for microprocessors // IEEE transactions on computers. – 1980. Vol. C-29. – № 6. – P. 429–441.

Тема 3. КРИТЕРИИ ОЦЕНКИ ТЕСТОВ И СИСТЕМ ДИАГНОСТИРОВАНИЯ

3.1. Виды критериев оценки

При организации процессов диагностирования, естественно, возникают вопросы оценки качества (эффективности) используемых тестов, отдельных диагностических средств и процедур, а также системы диагностирования в целом. Оценка эффективности позволяет количественно судить о полезности создаваемого или используемого средства диагностирования (или теста), сравнить их различные реализации.

В зависимости от целей возможны разные критерии оценки. Это могут быть, например, оценки затрат на создание средств диагностирования и их эксплуатацию, вероятностные оценки правильности результата диагностирования и т.д. Формулы для численной оценки свойств оцениваемого объекта обычно называют показателями.

Различают частные (единичные), интегральные и комплексные критерии (показатели). **Частный показатель** характеризует некоторое конкретное свойство оцениваемого объекта. Так как частных показателей может быть достаточно много, то возникает необходимость иметь некоторый обобщенный единый показатель, который и называют **интегральным**. Он представляет собой некоторую функцию от частных показателей. Чаще всего интегральный показатель – это экономический показатель, который характеризует уровень затрат и доходов. **Комплексный показатель** есть функция от нескольких (но не всех) частных показателей.

В ГОСТ 20911-89 даются некоторые показатели, которые обычно используют в технической диагностике.

Продолжительность диагностирования (контроля) – интервал времени, необходимый для проведения диагностирования (контроля) объекта.

Условная вероятность необнаруженного отказа (неисправности) при диагностировании (контроле) – вероятность того, что неисправный (неработоспособный) объект в результате диагностирования (контроля) признается исправным (работоспособным).

Условная вероятность ложного отказа (неисправности) при диагностировании (контроле) – вероятность того, что исправный (работоспособный) объект в результате диагностирования (контроля) признается неисправным (неработоспособным).

Условная вероятность необнаруженного отказа (неисправности) в данном элементе (группе) – вероятность того, что при наличии отказа (неисправности) в результате диагностирования принимается решение об отсутствии отказа (неисправности) в данном элементе (группе).

Условная вероятность ложного отказа (неисправности) в данном элементе (группе) – вероятность того, что при отсутствии отказа (исправности) в результате диагностирования принимается решение о наличии отказа (неисправности) в данном элементе (группе).

Полнота контроля – характеристика, определяющая возможность выявления отказов (неисправностей) в объекте при выбранном методе его диагностирования (контроля).

Глубина поиска места неисправностей – характеристика, задаваемая указанием составной части объекта с точностью, до которой определяется место отказа (неисправности).

Достоверность контроля – степень объективного соответствия результатов диагностирования (контроля) действительному техническому состоянию объекта.

Кроме вышеуказанных критериев, на этапе проектирования систем диагностирования обычно делают оценку затрат на средства и процессы диагностирования. Как правило, затраты зависят от ряда параметров, и при создании системы диагностирования ставится задача оптимизировать затраты в зависимости от этих параметров. Оценивать **суммарные затраты на создание системы диагностирования** для поиска с глубиной i и запасных элементов на ремонт можно по формуле

$$C(i) = R_{п.д.} + Q_{зип.}$$

где $R_{п.д.}$ – общая стоимость средств диагностирования при поиске с глубиной i , $Q_{зип.}$ – стоимость запасных элементов замены, используемых для ремонта.

С увеличением глубины поиска растет величина $R_{п.д.}$ и уменьшается значение $Q_{зип.}$ (рис. 3.1). Если нам известны функции $R_{п.д.}$ и $Q_{зип.}$, то по ним можно построить функцию $C(i)$, экстремум которой (точка А на рис. 3.1) определяет глобальный минимум $C(i)$. Пользуясь графиками рис. 3.1, для необходимой глубины поиска легко определить значения показателей $R_{п.д.}$ и $Q_{зип.}$.

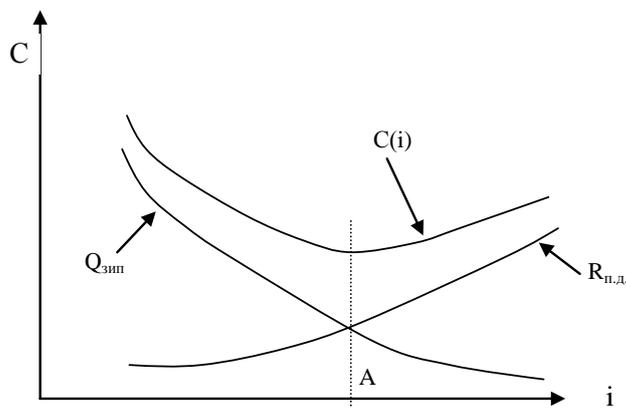


Рис. 3.1. Графическое изображение функций $R_{п.д.}$, $Q_{зип.}$, $C(i)$

Если известны затраты от ошибочных результатов диагностирования (R_1) и средние затраты на эксплуатацию системы диагностирования (R_2), то **экономическую эффективность** можно подсчитать как $C=R_1+R_2$. Возможна оценка по **информационной эффективности**, которая определяет количество информации о техническом состоянии объекта диагностирования, получаемой с помощью системы диагностирования.

Показатели могут определяться как в абсолютных, так и в относительных единицах. Например, **относительную эффективность** можно определять по формуле $K_3=\mathcal{E}/(Z_p+Z_c)$, где \mathcal{E} – суммарный полезный эффект от применения системы, Z_p – затраты на разработку, изготовление и ввод в эксплуатацию; Z_c – затраты на сопровождение при эксплуатации.

Сегодня общепризнанно, что спроектированное изделие должно быть контролепригодным. Критерии контролепригодности довольно подробно обсуждаются в разделе 6 настоящего учебного пособия.

ГОСТ 25518-93 (Межгосударственный стандарт. Диагностирование изделий. Общие требования) устанавливает характеристики и показатели, рекомендуемые для использования при создании диагностического обеспечения. При этом все показатели разбиваются на две группы: **показатели достоверности и точности диагностирования, показатели технико-экономические**. Перечень первых приведен в табл. 3.1.

Таблица 3.1

Показатели достоверности и точности диагностирования

Задачи диагностирования	Результат диагностирования	Показатели достоверности и точности
1	2	3
Определение вида технического состояния	1. Изделие исправно и (или) работоспособно	Вероятность того, что в результате диагностирования изделие признаётся исправным (работоспособным) при условии, что оно неисправно (неработоспособно)
	2. Изделие неисправно и (или) неработоспособно	Вероятность того, что в результате диагностирования изделие признаётся неисправным (неработоспособным) при условии, что оно исправно (работоспособно)
Поиск места отказа или неисправностей	Наименование элемента (сборочной единицы) или группы элементов, которые имеют неисправное состояние и место отказа или неисправностей	Вероятность того, что в результате диагностирования принимается решение об отсутствии отказа (неисправности) в данном элементе (группе) при условии, что данный отказ имеет место
		Вероятность того, что в результате диагностирования принимается решение о наличии отказа в данном элементе (группе) при условии, что данный отказ отсутствует

1	2	3
Прогнозирование технического состояния	Численное значение параметров технического состояния на задаваемый период времени, в том числе и на данный момент времени	Среднеквадратическое отклонение прогнозируемого параметра
	Численное значение остаточного ресурса (наработки)	Среднеквадратическое отклонение прогнозируемого остаточного ресурса
	Численное значение остаточного ресурса (наработки), соответствующего заданной вероятности	Вероятность безотказной работы, показатели изменения прогнозируемого диагностического параметра по нормативно-техническому документу
	Нижняя граница вероятности безотказной работы по параметрам безопасности на задаваемый период времени	Доверительная вероятность

Наиболее важными и практически часто используемыми оценками первой группы при построении тестов и общей оценке диагностического обеспечения электронных устройств являются: полнота и глубина поиска неисправностей, достоверность контроля. Поэтому более подробно рассмотрим, как определить их численные значения.

3.2. Оценка полноты контроля

Понятие полноты контроля

Полнота контроля – основной критерий оценки качества тестов, процедур контроля и систем диагностирования, применяемых на практике. Обычно она определяется как доля или процент проверяемых (обнаруживаемых) неисправностей от всех возможных одиночных неисправностей.

Известны подобные оценки на основе показателей надежности.

Например, она может оцениваться по формуле

$$\Pi = \sum_{i=1}^n \lambda_i / \sum_{k=1}^m \lambda_k ,$$

где λ_i – сумма интенсивностей неисправностей¹, которые обнаруживаются тестом (системой диагностирования);

¹ Интенсивность отказов (неисправностей) – условная плотность вероятности возникновения отказа объекта, определяемая при условии, что до рассматриваемого момента отказ не возникал.

λ_k – сумма интенсивностей всех возможных неисправностей (n и m – число обнаруживаемых и всех возможных неисправностей, соответственно).

Для теста, на котором проверяются все возможные неисправности, по этой формуле $\Pi=1$. Однако определить значения интенсивностей отказов элементов для современных объектов диагностирования практически невозможно.

Рассмотрим более практичный способ вычисления полноты. Будем определять ее в виде **процента проверяемых неисправностей** по формуле:

$$\Pi_1 = [(G - G_1)/G] \cdot 100\%,$$

где G – число всех допустимых неисправностей, G_1 – число непроверяемых неисправностей.

С увеличением сложности ОД обычно возрастает число условно проверяемых неисправностей. Причем их поиск – наиболее неприятная и сложная проблема.

При наличии условно проверяемых неисправностей для оценки полноты контроля целесообразно дополнительно применять еще две формулы:

$$\Pi_2 = \{[G - (G_1 + \sum_{k=1}^{G_2} m_k)]/G\} \cdot 100\%;$$

$$\Pi_3 = \{[G - (G_1 + G_2)]/G\} \cdot 100\%,$$

где G_2 – число условно проверяемых неисправностей, m_k – вероятность проверки k -й неисправности из числа условно проверяемых (понятия проверяемых, непроверяемых и условно проверяемых неисправностей обсуждались в п. 2.4).

Фактически оценки Π_1 и Π_3 дают верхнюю и нижнюю границы возможных значений полноты проверки. Значение Π_2 ($\Pi_3 < \Pi_2 < \Pi_1$) определяется с учетом вероятностей непроверки условно проверяемых неисправностей. При большом числе условно проверяемых неисправностей Π_2 может стать основной оценкой.

Если имеются данные (например в виде ТФН) о поведении исправного ОД и с неисправностью, то достаточно легко определить число непроверяемых и условно проверяемых неисправностей и вычислить Π_1 и Π_3 . Для вычисления Π_2 необходимо дополнительно уметь определять значения m_k .

Определение вероятности непроверки при аналоговых контролируемых параметрах

Рассмотрим, как определять значения m_k для цифровых и аналоговых КП. При этом будем считать значения КП случайными величинами, так как нам неизвестны техническое состояние ОД и точные значения характеристик его элементов.

В последующих рассуждениях будем использовать следующие обозначения: $\mathbf{E} = e_1, \dots, e_n$ – последовательность тестовых наборов, подаваемых на внешние входы ОД; $\mathbf{S} = \{s_1, \dots, s_r\}$ – множество допустимых неисправностей; $\mathbf{W} = \{w_1, \dots, w_m\}$ – множество КП.

В процессе диагностирования результат каждого измерения сравнивается с соответствующим допуском. Если хотя бы один из результатов измерений находится вне границ заданного для него допуска, то ОД считаем неисправным.

У разных экземпляров одного и того же устройства элементы имеют некоторое различие в характеристиках, что является следствием технологического разброса при их изготовлении. Поэтому аналоговые сигналы и соответственно аналоговые КП, характеризуются некоторым полем рассеивания (интервалом значений). Значения нижней и верхней границ такого интервала j -го КП на i -й проверке для исправного устройства обозначим a_o^{ij} и b_o^{ij} , для устройства с неисправностью $s_k - a_k^{ij}$ и b_k^{ij} . Очевидно, что поле допуска для проверки e_i и КП w_j должно включать по крайней мере часть интервала $[a_o^{ij}, b_o^{ij}]$.

В дальнейшем для определенности будем считать, что нижняя и верхняя границы допуска на наборе e_i при КП w_j равны a_o^{ij} и b_o^{ij} , т.е. интервал $[a_o^{ij}, b_o^{ij}]$ задает допустимые значения, если объект диагностирования исправен.

Очевидно, для проверяемой и непроверяемой неисправностей $m_k = 0$ и $m_k = 1$ соответственно. Для условно проверяемой неисправности $0 < m_k < 1$.

Сначала предположим, что в процессе контроля измеряется только один КП w_j и только на одной проверке e_i . Тогда для условно проверяемой неисправности s_k

$$m_k^{ij} = m_k = \int_a^b f_k^i(w_j) dw_j,$$

где $f_k^i(w_j)$ – функция плотности вероятности значений $w_j \in W$ на проверке $e_i \in E$ при наличии неисправности $s_k \in S$; a и b – нижняя и верхняя граница области общих значений для исправного (s_o) и неисправного (s_k) устройств.

Интеграл в формуле берется по области значений, которые являются общими для исправного и неисправного технических состояний. Фактически значение m_k есть вероятность того, что при наличии в объекте контроля неисправности s_k результат измерения попадет в интервал $[a, b]$.

Например, пусть границы интервалов допустимых значений для s_o и s_k такие, как показано на рис. 3.2.

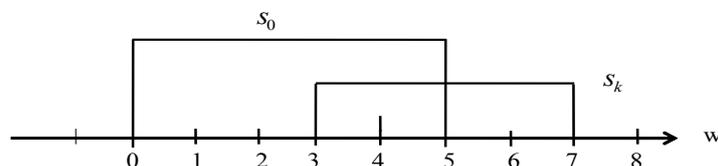


Рис. 3.2. Пример пересечения интервалов допустимых значений

Тогда $m_k = \int_3^5 f_k^i(w_j) dw_j$.

Пусть теперь выполняется несколько проверок и измеряется несколько КП. Если предположить, что значения КП некоррелированы, то

$$m_k = \prod_{i=j=1}^{i=n, j=m} m_k^{ij}.$$

Следовательно,

$$\Pi_2 = [G - (G_1 + \sum_{k=1}^{G_2} \prod_{i=j=1}^{i=n, j=m} m_k^{ij}) / G] 100\%,$$

где k соответствует k -й неисправности из числа условно проверяемых.

В практических расчетах может оказаться удобным организовать вычисления по формуле:

$$\Pi_2 = [(G - \sum_{k=1}^G \prod_{i=j=1}^{i=n, j=m} m_k^{ij}) / G] 100\%.$$

Определение функции $f_k^i(w_j)$, как правило, сопряжено со значительными трудностями. Вычисления m_k значительно упрощаются, если допустить, что для $f_k^i(w_j)$ имеют место равномерный или нормальный закон распределения. Так, при равномерном законе значение m_k^{ij} есть отношение длины интервала значений, общих для технических состояний s_0 и s_k , к длине интервала поля рассеивания при s_k . Предположим, что соотношения полей рассеивания такие, как на рис. 3.2.

Тогда:

$$m_k^{ij} = (b_0^{ij} - a_k^{ij}) / (b_k^{ij} - a_k^{ij}) = (5 - 3) / (7 - 3) = 1/2.$$

Проиллюстрируем предлагаемый способ оценки полноты контроля на примере, представленном табл. 3.2. Для этого примера $E = e_1, e_2$; $W = \{w_1, w_2\}$ и $S = \{s_1, s_2, s_3\}$. Интервалы $[a_0^{ij}, b_0^{ij}]$ и $[a_k^{ij}, b_k^{ij}]$ даны в подтаблицах (M_0 определяет интервалы значений КП для s_0 ; M_k – для $s_k \in S$). Интервалы общих значений для исправного и каждого неисправного технического состояния показаны в табл. 3.3 (подтаблица T_k содержит результат пересечения интервалов матриц M_0 и M_k).

Таблица 3.2

Проверка	M_0		M_1		M_2		M_3	
	w_1	w_2	w_1	w_2	w_1	w_2	w_1	w_2
e_1	[1,4]	[0,5]	[1,5]	[2,4]	[1,4]	[1,3]	[1,4]	[2,3]
e_2	[3,7]	[1,4]	[1,4]	[3,6]	[3,6]	[2,4]	[3,7]	[2,8]

Таблица 3.3

Проверки	T_1		T_2		T_3	
	w_1	w_2	w_1	w_2	w_1	w_2
e_1	[1,4]	[2,4]	[1,4]	[1,3]	[1,4]	[2,3]
e_2	[3,4]	[3,4]	[3,6]	[2,4]	[3,7]	[2,4]

Для упрощения вычислений примем гипотезу о равномерном распределении значений КП. Тогда получим

$$\begin{array}{lll} m_1^{11} = (4-1)/(5-1)=3/4 & m_2^{11} = 1 & m_3^{11} = 1; \\ m_1^{12} = 1 & m_2^{12} = 1 & m_3^{12} = 1; \\ m_1^{21} = 1/3 & m_2^{21} = 1 & m_3^{21} = 1; \\ m_1^{22} = 1/3 & m_2^{22} = 1 & m_3^{22} = 1/3. \end{array}$$

Отсюда:

$$m_1 = \prod_{i=j=1}^{i=j=2} m_1^{ij} = 3/4 \cdot 1 \cdot 1/3 \cdot 1/3 = 1/12;$$

$$m_2 = 1 \cdot 1 \cdot 1 \cdot 1 = 1;$$

$$m_3 = 1 \cdot 1 \cdot 1 \cdot 1/3 = 1/3.$$

Анализируя полученные значения, нетрудно установить, что неисправность s_2 относится к числу непроверяемых, а s_1 и s_3 – к условно проверяемым.

Таким образом, в рассматриваемом случае $G = 3$, $G_1 = 1$, $G_2 = 2$.

Следовательно,

$$П_1 = [(3 - 1) / 3] \cdot 100\% \approx 67\%.$$

$$П_3 = \{[3 - (1 + 2)] / 3\} \cdot 100\% = 0\%;$$

Заметим, что данный пример иллюстрирует ситуацию, когда оценки $П_1$ и $П_3$ мало информативны. Так, если включение новых проверок в последовательность E или изменение состава W скажется только на степени пересечения полей рассеивания в технических состояниях s_o и s_k , то оценки $П_1$ и $П_3$ останутся неизменными.

Используя ранее вычисленные значения m_k , определим значение $П_2$:

$$П_2 = \{[G - (1 + \sum m_k)]/G\} \cdot 100\% = \{[3 - (1 + 5/12)]/3\} \cdot 100\% \approx 53\%.$$

Определение вероятности проверки при цифровых контролируемых параметрах

Сформулируем правила вычисления m_k для цифровых КП при следующих допущениях:

- все неисправности из S равновероятны;
- в ТФН возможны значения из алфавита $\{0, 1, x\}$;
- если матрица M_o или M_k ($k = 1, \dots, r$) имеет h символов x , то все 2^h возможных двоичных вариантов исходов равновероятны.

Обозначим через $|M_k|$ число возможных двоичных матриц, получаемых из M_k путем фиксации символов « x » в значениях 0 и 1. Очевидно, $|M_k| = 2^h$. В процессе диагностирования производятся измерения КП множества W на входных наборах E . Результаты измерений можно представить в виде матрицы M_x , строки которой соответствуют наборам из E , а столбцы – контролируемым параметрам из W . Если в объекте имеется неисправность s_k , то вероятность того, что матрица измерений M_x будет равна некоторой двоичной матрице, которую можно получить из M_k , равна $1/|M_k|$.

Матрица $S = M_o \cap M_k$ определяет все возможные варианты матриц измерений M_x , при которых неисправность s_k не проверяется. С учетом всего вышеизложенного вероятность непроверки неисправности s_k можно записать в виде выражения:

$$m_k = |M_o \cap M_k| / |M_k|,$$

где $|M_o \cap M_k|$ – число различных вариантов двоичных матриц, которые можно получить из матрицы $S = M_o \cap M_k$ путем фиксации символов «х» в матрице S значениями 0 и 1.

Например, определим значения m_k , если заданы матрицы:

$$M_o = \begin{vmatrix} 01x \\ x01 \\ 110 \end{vmatrix} \quad \text{и} \quad M_k = \begin{vmatrix} x1x \\ x01 \\ 11x \end{vmatrix}.$$

Тогда имеем

$$M_o \cap M_k = \begin{vmatrix} 01x \\ x01 \\ 110 \end{vmatrix}.$$

Так как $|M_k| = 2^4 = 16$ и $|M_o \cap M_k| = 2^2 = 4$,
то $m_k = |M_o \cap M_k| / |M_k| = 4/16 = 1/4$.

Объект диагностирования может иметь одновременно цифровые и аналоговые КП. Тогда следует отдельно, по вышеприведенным формулам, подсчитать для каждой $s_k \in S$ вероятность непроверки m_k' для цифровых КП и m_k'' – для аналоговых КП, а затем вычислить итоговую вероятность как произведение $m_k = m_k' \times m_k''$ (если результаты измерений некоррелированы). Далее при вычислении Π_2 использовать полученные итоговые вероятности.

3.3. Оценка глубины поиска неисправностей

Формальные оценки и их связь с технологиями ремонта

Для численной оценки глубины поиска неисправностей, которую можно получить при диагностировании, применяются различные формулы.

Рассмотрим три оценки, отражающие различные технологии (стратегии) ремонта:

$$R = \sum_{j=1}^r p_j \mu_j \text{ – разрешающая способность диагностирования;}$$

$$N(I) = \sum_{j=1}^r p_j \gamma_j \text{ – риск оператора;}$$

$$H = \sum_{j=1}^r p_j \log_2 \mu_j \text{ – энтропийный показатель.}$$

В приведенных формулах p_j – вероятность неисправности $s_j \in S$ ($j = 1, \dots, r$); μ_j – среднее число неисправностей в СПН при наличии неисправности s_j ; I – заданная последовательность анализа неисправностей в СПН; γ_j – среднее число неисправностей, анализируемых в СПН до неисправности s_j .

Общим для всех анализируемых критериев является то, что с уменьшением их значений увеличивается глубина поиска неисправности. Это означает, что из множества анализируемых вариантов оптимальным является тот, для которого получено минимальное значение.

Процессы диагностирования и ремонта объектов на практике могут осуществляться по-разному. Поэтому целесообразно выбирать такую формулу для численной оценки, которая учитывает технологию ремонта.

Первый критерий (R) хорош при процессах, когда после определения СПН заменяются все компоненты, неисправности которых вошли в СПН. Заметим, что некоторые из заменяемых компонентов могут быть исправными, например, в силу неразличимости неисправностей или из-за несовершенства теста. При этом, очевидно, трудоемкость и экономичность такого способа ремонта определяется количеством неисправностей в СПН. Разрешающая способность R диагностирования численно равна среднему числу неисправностей в СПН. Чем меньше R , тем меньше число заменяемых в процессе ремонта компонентов объекта. Значение R может быть в пределах от 1 (все неисправности различимы) до r (все неисправности неразличимы).

В объектах, содержащих большое число печатных плат, возможно использование модификации критерия R , при которой подсчитывается средняя доля случаев, когда число заменяемых плат не больше заданного числа m :

$$R^* = \frac{1}{N} \sum_{i=1}^N d_i,$$

где N – всего плат в объекте; $d_i=1$, если при i -й неисправности число блоков в СПН не больше m , иначе – 0. Этот критерий теоретически может иметь значения в интервале от 0 до 1. Значение $R^*=1$, если при любой неисправности число заменяемых компонент не больше m .

Значение критерия $N(I)$, названного риском оператора, равно среднему числу неисправностей в одном СПН с номерами, меньше номера фактической неисправности объекта. Это значение изменяется от 0 (все неисправности различимы) до $(r-1)/2$ (все неисправности неразличимы).

Второй критерий (в отличие от первого) предполагает, что замене подлежат только действительно неисправные компоненты. Для этого необходимо после нахождения СПН уточнить место фактической неисправности объекта. На практике определение фактической неисправности может заключаться в поочередном просмотре элементов, вошедших в СПН, до тех пор, пока неисправность не будет установлена. При этом просмотр неисправностей осуществляется в некотором заданном порядке, например, в порядке возрастания схемных номеров подозреваемых на неисправность элементов. Значение N

численно равно среднему числу неисправностей из СПН, которые необходимо проанализировать оператору для точного установления неисправного элемента. Значение N (при равной вероятности неисправностей) может находиться в пределах от $(r-1)/2$ до 0 .

Допустим, выделение из СПН фактической неисправности выполняется в несколько шагов. На каждом шаге очередная контрольная точка для просмотра выбирается таким образом, что множество подозреваемых неисправностей разделяется на две равные части, в одной из которых гарантированно находится фактическая неисправность. Критерий N численно равен среднему числу шагов, за которое можно выделить любую неисправность путем последовательного деления (на каждом шаге) множества подозреваемых неисправностей пополам. Если определение фактической неисправности из СПН выполняется именно таким образом и затраты на реализацию каждого шага (деления множества неисправностей пополам) равноценны, то оценку качества теста целесообразно проводить с использованием критерия N . Его значение равно среднему значению логарифма по основанию 2 от числа неисправностей в одном СПН и находится в пределах от 0 (все неисправности различимы) до $\log_2 r$ (все неисправности неразличимы).

Остановимся более подробно на правилах вычисления R , $N(I)$ и N . В их формулы входят аргументы μ_j и γ_j , которые, в свою очередь, могут быть вычислены по формулам:

$$\mu_j = \sum_{i=1}^r m_{ij} \quad \text{и} \quad \gamma_j = \sum_{s_i \in C_j} m_{ij},$$

где m_{ij} – вероятность включения s_i в СПН при наличии в ОД неисправности $s_j \in S$; C_j – список неисправностей, перечисленных в СПН до s_j (например, пусть $СПН = \{s_5, s_4, s_3, s_2, s_1\}$, тогда $C_3 = \{s_5, s_4\}$).

Таким образом, для вычислений всех рассматриваемых оценок необходимо уметь вычислять m_{ij} .

Рассмотрим правила вычисления значения m_{ij} . В методическом плане они во многом аналогичны тем, которые применялись при вычислении вероятности m_k (т.е. вероятности непроверки неисправности s_k , использовавшейся для оценки полноты контроля).

Очевидно, что $0 < m_{ij} < 1$. Если неисправности s_i и s_j различимы, то $m_{ij} = 0$; если неисправности неразличимы – $m_{ij} = 1$.

Определение вероятности различимости неисправностей при цифровых контролируемых параметрах

Эта вероятность определяется относительно пар неисправностей. Здесь может использоваться формула, подобная формуле для вычисления вероятности непроверки неисправности, если вместо исправного состояния рассматривать второе неисправное состояние.

При известной ТФН вероятность включения неисправности s_i в СПН при наличии в ОД неисправности s_j

$$m_{ij} = |M_i \cap M_j| / |M_j|,$$

где $|M_j|$ – число двоичных матриц, которые можно получить из матрицы M_j ; $|M_i \cap M_j|$ – число двоичных матриц, которые можно получить из матрицы $M_i \cap M_j$. Эта формула получена при предположении, что равновероятны все возможные двоичные исходы для матриц M_i и M_j .

Определение вероятности различимости неисправностей при аналоговых контролируемых параметрах

Если предположить, что результаты измерений некоррелированы, то

$$m_{ij} = \prod_{k=1}^{k=m, l=n} m_{ij}^{kl},$$

где m_{ij}^{kl} – вероятность включения s_i в СПН при наличии в ОД неисправности s_j , если измеряется значение только одного КП w_1 и только на проверке e_k .

В свою очередь,

$$m_{ij}^{kl} = \int_a^b f_j^k(w_1) dw_1,$$

где $f_j^k(w_1)$ – функция плотности вероятности значений w_1 на проверке e_k при наличии в ОД неисправности s_j , a и b – границы интервала общих значений для обеих неисправностей s_i и s_j . Если допустить, что $f_j^k(w_1)$ удовлетворяет равномерному закону, то $m_{ij}^{kl} = (b - a) / (b_j^{kl} - a_j^{kl})$.

Например, для неисправностей s_1 и s_2 (табл. 3.2) при допущении о равномерном распределении значений КП $m_{12}^{21} = (4 - 3) / (6 - 3) = 1/3$ (в этом примере интервал допустимых значений КП w_1 на e_2 для неисправности s_1 равен $[1,4]$, для s_2 – $[3,6]$).

Вероятности появления неисправностей, как правило, бывают неизвестны. Поэтому при вычислениях все неисправности обычно считают равновероятными. Тогда формулы для вычислений примут вид:

$$R = 1/r \cdot \sum_{j=1}^r \sum_{i=1}^r m_{ij}; \quad N(I) = 1/r \cdot \sum_{j=1}^r \gamma_j; \quad H = 1/r \cdot \sum_{j=1}^r \log_2 \mu_j.$$

Пример вычисления глубины поиска

Вычислим значения R и $N(I)$ для ОД с аналоговыми КП, ТФН которого представлена в табл. 3.4. В ней отсутствует матрица M_0 , так как она не нужна для вычислений.

Выполним сначала пересечения матриц, соответствующих различным парам s_i и s_j неисправностей из $S = \{s_1, s_2, s_3\}$. Результаты выполнения \cap -операции приведены в табл. 3.5. Очевидно, что нет необходимости выполнять пересечения M_j и M_j , так как $m_{jj} = 1$ для любого j .

Таблица 3.4

Проверки	M ₁		M ₂		M ₃	
	w ₁	w ₂	w ₁	w ₂	w ₁	w ₂
e ₁	[2,7]	[0,7]	[4,7]	[0,4]	[8,11]	[5,9]
e ₂	[2,11]	[0,10]	[3,7]	[0,4]	[6,10]	[5,9]
e ₃	[3,7]	[2,8]	[3,7]	[3,7]	[8,12]	[2,9]

Для \cap -операции справедлив переместительный закон, т.е. $M_i \cap M_j = M_j \cap M_i$. Поэтому в табл. 3.5 приведены результаты пересечения только трех пар матриц. Из таблицы видно, что $M_1 \cap M_3 = \emptyset$ и $M_2 \cap M_3 = \emptyset$.

Таблица 3.5

Проверки	M ₁ \cap M ₂		M ₁ \cap M ₃		M ₂ \cap M ₃	
	w ₁	w ₂	w ₁	w ₂	w ₁	w ₂
e ₁	[4,7]	[0,4]	\emptyset	[5,7]	\emptyset	\emptyset
e ₂	[3,7]	[0,4]	[6,10]	[5,9]	[6,7]	\emptyset
e ₃	[3,7]	[3,7]	\emptyset	[2,8]	\emptyset	[3,7]

Отсюда $m_{13} = m_{31} = m_{23} = m_{32} = 0$. Теперь остались неизвестными только m_{12} и m_{21} . Значения m_{12}^{kl} и m_{21}^{kl} , необходимые для вычисления m_{12} и m_{21} , приведены в табл. 3.6. Предполагая, что для функций плотности вероятностей значений КП выполняется равномерный закон, получим следующие значения вероятностей:

$$\begin{array}{lll}
 m_{11} = 1 & m_{12} = 1 & m_{13} = 0 \\
 m_{21} = 3/5 \cdot 4/9 \cdot 4/7 \cdot 1; 2/5 \cdot 2/3 \approx 0,04 & m_{22} = 1 & m_{23} = 0 \\
 m_{31} = 0 & m_{32} = 0 & m_{33} = 1.
 \end{array}$$

Таблица 3.6

Проверки	m ₂₁ ^{kl}		m ₁₂ ^{kl}	
	w ₁	w ₂	w ₁	w ₂
e ₁	3/5	4/7	1	1
e ₂	4/9	2/5	1	1
e ₃	1	2/3	1	1

После подстановки значений m_{ij} в формулу для разрешающей способности диагностирования получим

$$R = 1/3 \cdot \sum_{j=1}^r \sum_{i=1}^r m_{ij} \approx 1,35.$$

Следовательно, среднее число неисправностей по всем СПН, которые будут получаться при диагностировании данного устройства, равно 1,35.

Для этого же примера вычислим значение $N(I)$ для двух последовательностей $I_1 = s_1, s_2, s_3$ и $I_2 = s_3, s_2, s_1$ анализа неисправностей. В случае, если неисправности в СПН будут анализироваться в последовательности I_1 , получим

$$\gamma_1 = 0, \gamma_2 = 1, \gamma_3 = 0.$$

Тогда при допущении о равной вероятности появления неисправностей

$$N(I_1) = 1/3 \cdot \sum_{j=1}^3 \gamma_j = 1/3.$$

Для последовательности I_2 получим

$$\gamma_1 \approx 0,04, \gamma_2 = 0, \gamma_3 = 0.$$

Тогда $N(I_2) \approx 1/3 \cdot 0,04 \approx 0,013$. Сравнивая значения $N(I_1)$ и $N(I_2)$, легко убедиться, что последовательность I_2 анализа неисправностей в СПН предпочтительнее с точки зрения риска оператора.

Если ОД имеет одновременно цифровые и аналоговые КП, то вероятность включения s_i в СПН равна $m_{ij} = m_{ij}' \times m_{ij}''$, где m_{ij}' и m_{ij}'' – вероятности включения s_i в СПН, вычисленные отдельно для цифровых и аналоговых КП.

3.4. Достоверность контроля

Любая система контроля работает с ошибками. Кроме того, контролю подвергается только часть параметров ОД. Поэтому получаемая в результате контроля информация содержит неопределенность. Достоверность контроля зависит от точности измерений и объема контроля. Достоверность контроля есть мера определенности результатов контроля.

Решение о техническом состоянии ОД принимается на основе сравнения показателя качества с допусками. Показатель качества вычисляется по измеренным значениям контролируемых параметров. Поэтому достоверность контроля есть достоверность принятия решений по показателям качества.

Наряду с достоверностью по показателю качества рассматривают понятие достоверности по параметру. Достоверность по показателю качества выражается через достоверности по параметрам.

Существует несколько различных численных оценок достоверности (абсолютная, относительная, методическая и т.п.). Мы рассмотрим формулы для определения абсолютной достоверности и ее основных составляющих, которые используются при вычислениях и других видов достоверности.

Абсолютная достоверность результатов контроля – вероятность принятия правильного решения:

$$D = 1 - P_{\text{ош}} = 1 - (\alpha + \beta),$$

где α – риск изготовителя (вероятность того, что работоспособный объект признан негодным);

β – риск заказчика (вероятность того, что неработоспособный объект признан годным).

В процессе определения технического состояния сложной системы при контроле каждого параметра возможны следующие независимые и единственно возможные события: годный параметр оценивается системой контроля как годный; годный параметр оценивается системой контроля как негодный; негодный параметр оценивается системой контроля как негодный; негодный параметр оценивается системой контроля как годный.

Под годным понимается параметр, находящийся в пределах установленного допуска, под негодным – параметр, вышедший за пределы допуска.

Введем следующие обозначения: A_i – i -й параметр годный, A_i^* – i -й параметр негодный, B_i – i -й параметр оценивается системой контроля как годный, B_i^* – i -й параметр оценивается системой контроля как негодный.

Тогда перечисленные выше сложные события можно записать в виде $A_i B_i$, $A_i B_i^*$, $A_i^* B_i$, $A_i^* B_i^*$, а их вероятности соответственно $P(A_i B_i)$, $P(A_i B_i^*)$, $P(A_i^* B_i)$, $P(A_i^* B_i^*)$. Выражение $A_i B_i^*$ означает, например, что годный параметр оценивается системой контроля как негодный.

Указанные события составляют полную группу событий, поэтому сумма их вероятностей равна единице:

$$P(A_i B_i) + P(A_i B_i^*) + P(A_i^* B_i) + P(A_i^* B_i^*) = 1.$$

Неправильная оценка параметров системой контроля имеет место при наступлении событий $A_i B_i^*$ и $A_i^* B_i$. Тогда достоверность D_i контроля параметра, под которой понимается вероятность правильной оценки системой контроля состояния i -го параметра, выражается формулой:

$$D_i = P(A_i B_i) + P(A_i^* B_i^*).$$

Вероятность неправильной оценки состояния i -го параметра представляет собой сумму ошибок первого рода $P(A_i B_i^*)$ – риск изготовителя, и второго рода $P(A_i^* B_i)$ – риск заказчика.

Вышеуказанные вероятности можно определить по характеристикам параметра и системы контроля (величине допуска, среднему квадратическому отклонению, точности измерения).

Например, вероятность того, что годный параметр оценивается как годный

$$P(A_i B_i) = \int_{a_i}^{b_i} f(x_i) \left[\int_{a_i - x}^{b_i - x} f_1(t_i) dt_i \right] dx_i,$$

где $f(x_i)$ и $f_1(t_i)$ – соответственно плотности распределения параметра x_i и ошибки измерений t_i , a_i и b_i – соответственно нижняя и верхняя границы допуска на i -й параметр.

3.5. Анализ на состязания тестов цифровых схем

Понятие состязания

Все элементы и линии связи вносят задержки на распространение сигналов. Так, если на вход элемента-повторителя подать импульсный сигнал $U_{вх}$ (рис. 3.3), то на его выходе сигнал $U_{вых}$ (пунктирная линия) меняется с некоторой задержкой относительно входного сигнала.

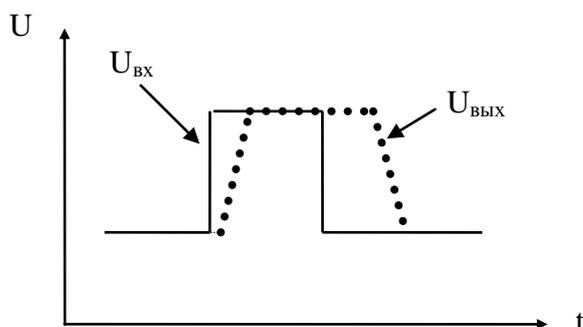


Рис. 3.3. Формы входного и выходного импульсов

Если сигнал проходит через несколько элементов, то в итоге может быть довольно существенная задержка. Разброс временных параметров реальных элементов, а также различная длина цепей прохождения сигналов могут являться причиной неодновременного изменения состояний входов некоторых элементов схем.

Пусть на элемент «И» (рис. 3.4) последовательно подаются комбинации сигналов 01 и 10, т.е. сигнал на первом входе меняется с 0 на 1, а на втором – с 1 на 0. Установившиеся значения выходного сигнала на обеих этих комбинациях одинаковы и равны нулю.

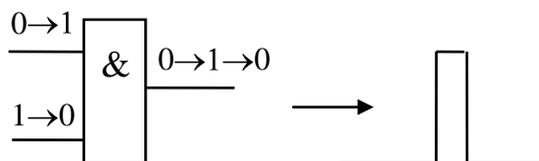


Рис. 3.4. Пример появления кратковременного импульса из-за неодновременного изменения входных сигналов

Однако если сигнал на первом входе изменится раньше, чем на втором, то на входах может кратковременно возникнуть комбинация 11. Соответственно, на выходе кратковременно появится единичный импульс. На рисунке 3.4 (справа) показана форма выходного сигнала элемента в этом случае. Если же задержка по второму входу меньше, то при смене входной комбинации сигнал на выходе элемента не изменится. В результате этого при тестовой проверке на оче-

редном входном наборе выходные сигналы логических элементов могут принимать значения, не предусмотренные логикой функционирования схемы (без учета задержек). В таких случаях говорят, что на данном наборе существуют так называемые состязания между сигналами.

При проектировании цифровых устройств, как правило, принимаются меры для устранения влияния состязаний на алгоритм функционирования устройства. Однако если проектируемое устройство в целом устойчиво к состязаниям, при тестовой проверке отдельных конструктивно-логических узлов необходимо считаться с возможностью состязаний.

Как показывает практика наладки и ремонта, тесты, синтезированные без учета состязаний, могут приводить к неоднозначности результатов проверки. К тому же в устройстве, свободном от состязаний в исправном состоянии, могут возникать состязания при наличии в нем неисправности. Все это существенно усложняет процессы диагностирования.

В зависимости от места возникновения состязаний сигналов различают:

- состязания между изменениями сигналов на внешних входах;
- состязания между входными сигналами и сигналами элементов памяти;
- состязания между сигналами элементов памяти;
- состязания между сигналами элементов схемы.

Установившиеся значения сигналов в комбинационных схемах определяются исключительно значениями сигналов на внешних входах. В таких схемах достаточно легко избавиться от «ложных» импульсов, появляющихся вследствие состязаний. Например, сигналы с выходов схемы можно снимать с некоторой задержкой.

В схемах с памятью борьба с состязаниями является более сложной, так как «ложные» импульсы могут запоминаться и влиять на последующую работу схемы. Допустим, что в цифровой схеме на рис. 3.5 входной набор 011 изменяется на 101, а на выходах элементов 2 и 3 (образующих триггер) первоначально были значения 0 и 1 соответственно. Если входной сигнал x_1 изменится раньше, чем x_2 , то на выходе элемента 1 появится кратковременный нулевой импульс, что приведет к смене сигналов на элементах 2 и 3. В случае же, когда на выходе элемента 1 не будет единичного импульса, триггер не изменит своего состояния. Иными словами, установившиеся значения на выходах элементов 2 и 3 будут зависеть от соотношений моментов изменений сигналов на x_1 и x_2 .

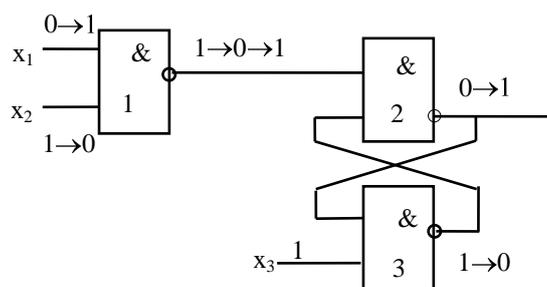


Рис. 3.5. Иллюстрация состязаний в схеме с памятью

Классификация состязаний в цифровых схемах

На рисунке 3.6 приведена классификация состязаний. По характеру реакции схемы на состязания в ней выделяют две группы. Если состязание может вызвать переход схемы в неправильное устойчивое состояние, то оно называется опасным состязанием. Если эффект состязания заключается в возможности появления на некоторых выходах элементов схемы кратковременных импульсов (при правильном установившемся состоянии), то состязание называется неопасным.

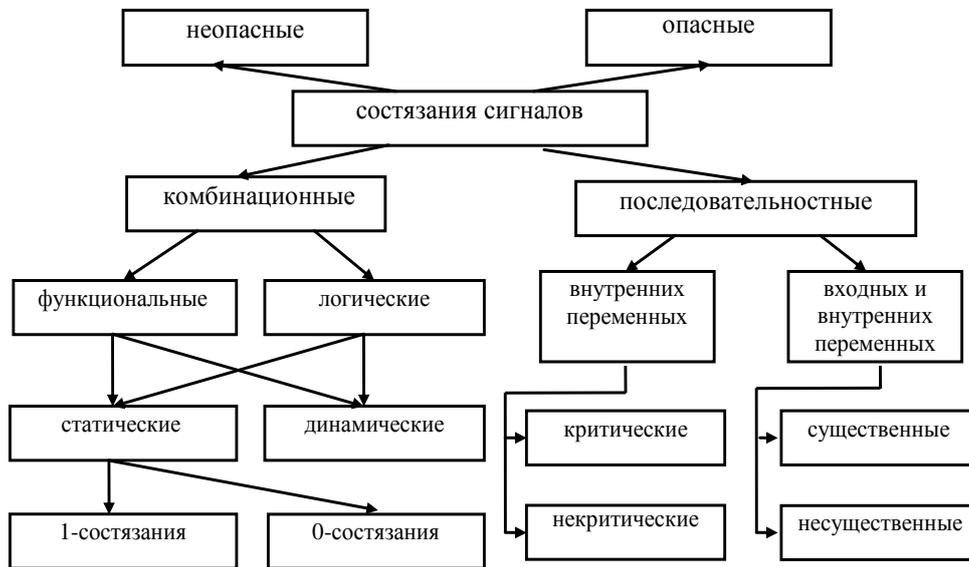


Рис. 3.6. Классификация состязаний

Кроме того, все состязания разделяются по типу схем: на комбинационные и последовательные.

В свою очередь, комбинационные разделяются на функциональные и логические.

Как функциональные, так и логические состязания могут быть двух типов: статические или динамические.

Обозначим через $f(e)$ значение функции схемы на входном наборе e . Схема, реализующая (на одном из выходов) функцию f , содержит **статическое** [**динамическое**] **состязание** на переходе $e_j \rightarrow e_{j+1}$ (т.е. при смене входного набора e_j на e_{j+1}), если $f(e_j) = f(e_{j+1})$ [$f(e_j) \neq f(e_{j+1})$], и во время перехода на выходе схемы может возникнуть один или несколько кратковременных импульсов.

Иными словами, при статическом состязании установившиеся значения выходных сигналов комбинационной схемы на e_j и e_{j+1} одинаковы, а при динамическом – разные.

Статическое состязание называется **1-состязанием**, если $f(e_j) = f(e_{j+1}) = 1$, и **0-состязанием**, если $f(e_j) = f(e_{j+1}) = 0$.

Состязания, называемые функциональными, иллюстрируются картой Карно¹, представленной табл. 3.7. Такое состязание возможно на входном переходе $000 \rightarrow 110$, что соответствует переходу из клетки «а» в клетку «с» (табл. 3.7). При таком переходе возможно кратковременное попадание в клетку «b», если x_2 изменится несколько раньше, чем x_1 . Тогда на выходе возможно появление нулевого импульса, то есть происходит статическое 1-состязание. При переходе из клетки «а» в клетку «d» меняется значение функции с 1 на 0, причем по таблице видно, что при переходе возможна последовательность смены входных сигналов $000 - 001 - 011 - 111$. В ходе этих переходов функция последовательно принимает значения 1-0-1-0, т.е. происходит динамическое состязание.

Более строго понятие **функционального статического состязания** можно определить следующим образом. Рассмотрим переход $e_j \rightarrow e_{j+1}$ с изменением r входных переменных, где

$$e_j = (x_1, \dots, x_r, x_{r+1}, \dots, x_n) \text{ и } e_{j+1} = (\bar{x}_1, \dots, \bar{x}_r, x_{r+1}, \dots, x_n).$$

Булева функция f содержит функциональное статическое состязание на переходе $e_j \rightarrow e_{j+1}$, если и только если $f(e_j) = f(e_{j+1})$ и среди 2^r входных наборов с зафиксированными значениями x_{r+1}, \dots, x_n (изменяются переменные x_1, \dots, x_r) найдутся хотя бы два набора e_i и e_k такие, что $f(e_i) \neq f(e_k)$.

Функциональные состязания отражают свойства заданной переключающей функции. Доказано, что никакая схемная реализация функции (при допущении произвольных и заранее неизвестных задержек) не может устранить функциональные состязания.

Второй тип состязаний, называемый **логическим**, зависит от структурной реализации функции схемы. Логические состязания, как правило, устраняются выбором соответствующей структуры схемы на этапе ее проектирования, а также ограничениями на допустимые входные переходы.

Рассмотрим, например, функцию, заданную картой Карно в табл. 3.8. Пусть эта функция реализуется схемой на рис. 3.7а. Судя по табл. 3.8, на соседних входных наборах со значениями входных сигналов $011 \rightarrow 111$ функциональное состязание отсутствует (переход от клетки «а» в «b»). Однако если сигнал на выходе элемента И2 (рис. 3.7а) изменится раньше сигнала на выходе элемента И1, то на выходе схемы возможно появление кратковременного нулевого сигнала. В данном примере происходит логическое статическое 1-состязание. Это состязание можно устранить, добавив элемент И3 (не изменяющий функции), как показано на рис. 3.7б.

¹ Карта Карно – это одна из табличных форм описания функции. В нашем примере она описывает некоторую функцию трех переменных: x_1, x_2, x_3 . Каждому столбцу сопоставляются значения пары переменных x_1 и x_2 , а строке – x_3 . В клетках таблицы проставляются соответствующие значения функции.

Таблица 3.7

		x_1x_2			
		00	01	11	10
x_3	0	1(a)	0(b)	1(c)	1
	1	0	1	0(d)	1

Таблица 3.8

		x_1x_2			
		00	01	11	10
x_3	0	0	0	1	0
	1	1	1(a)	1(b)	1

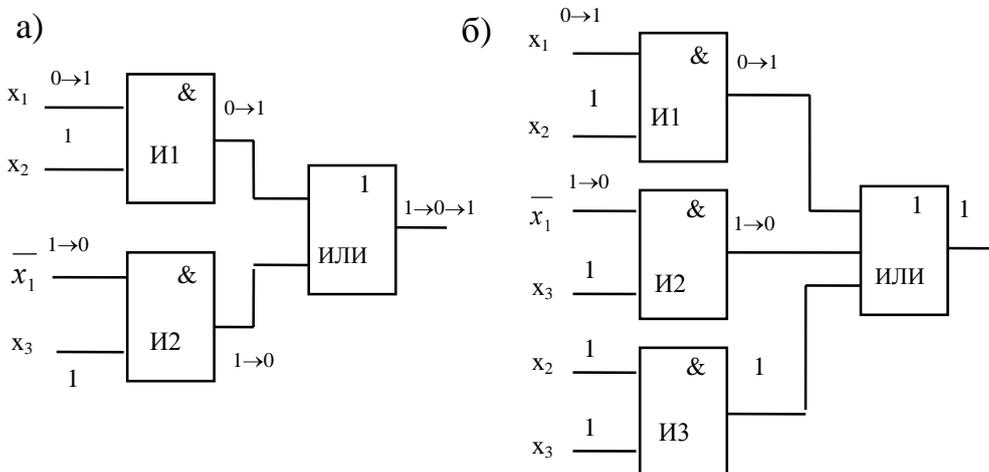


Рис. 3.7. Преобразование схемы для устранения состязания логического типа

В отличие от функционального, при логическом статическом состязании среди 2^r входных наборов с зафиксированными значениями x_{r+1}, \dots, x_n (изменяются переменные x_1, \dots, x_r) нет наборов e_i и e_k таких, что $f(e_i) \neq f(e_k)$, однако возможно появление кратковременного импульса.

При анализе поведения устройств с памятью следует учитывать как комбинационные состязания, так и состязания, присущие только последовательным схемам (рис. 3.6). Среди них выделяют состязания между внутренними состояниями и между входными сигналами и внутренними состояниями.

Критические и существенные состязания относятся к опасным, а некритические и несущественные – к неопасным.

Анализ на состязания тестов цифровых схем с использованием моделирования

В инженерной практике для анализа на состязания тестов (комбинационных и последовательных схем) эффективно используется аппарат троичного моделирования. При этом предполагается, что функционирование каждого элемента схемы описывается в трехзначном алфавите $\{0, 1, x\}$.

Основываясь на анализе троичных функций логических элементов, была выдвинута гипотеза о том, что если изменившимся входным сигналам присвоить

значения «х», то значение троичной функции равно 1 или 0 только в том случае, когда выход элемента не зависит от изменившихся входных сигналов, в противном случае значение функции равно «х». При вычислении функции элемента в алфавите {0, 1, х} используются следующие правила:

$0 \vee 0 = 0$	$0 \vee x = x$	$0 \wedge 0 = 0$	$0 \wedge x = 0$	$\overline{\overline{x}} = x$
$0 \vee 1 = 1$	$x \vee 0 = x$	$0 \wedge 1 = 0$	$x \wedge 0 = 0$	
$1 \vee 0 = 1$	$1 \vee x = 1$	$1 \wedge 0 = 0$	$1 \wedge x = x$	
$1 \vee 1 = 1$	$x \vee 1 = 1$	$1 \wedge 1 = 1$	$x \wedge 1 = 0$	
	$x \vee x = x$		$x \wedge x = x$	

Введем обозначения:

$\alpha = (x_1, \dots, x_p, x_{p+1}, \dots, x_n)$ и $\beta = (\overline{x_1}, \dots, \overline{x_p}, x_{p+1}, \dots, x_n)$ – некоторая пара входных наборов схемы;

$\alpha | \beta$ – входной набор, в котором значения первых p переменных (x_1, \dots, x_p) имеют значения «х», а остальные имеют значения как в α и β . Тогда на основании приведенной выше гипотезы можно сформулировать следующее утверждение.

Утверждение 3.1. Комбинационная схема содержит статическое состязание входных переменных при переходе от набора α к β , если

$$f(\alpha) = f(\beta) \neq x \text{ и } f(\alpha | \beta) = x.$$

Для иллюстрации проверим возможность состязаний в схеме рис. 3.8 при переходе от набора $e_1 = 000$ к набору $e_2 = 101$. Результаты моделирования функционирования схемы на этих наборах приведены в табл. 3.9. Видно, что для z_1 и z_2 первое условие выполняется, т.е.

$$z_1(e_1) = z_1(e_2) = 1 \text{ и } z_2(e_1) = z_2(e_2) = 1.$$

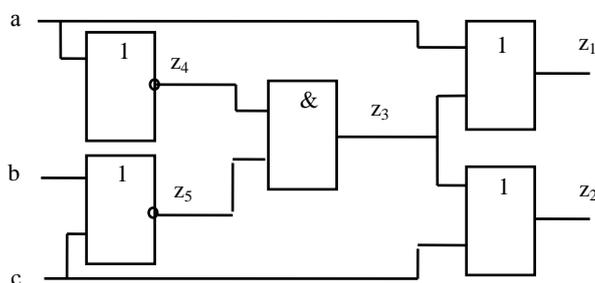


Рис. 3.8. Комбинационная схема с состязанием

Проверим второе условие.

На переходе $e_1 \rightarrow e_2$ сменились значения входных переменных a, c . Заменяем значения a и c на наборе e_2 на «х» и проведем троичное моделирование работы схемы на этом наборе (третья строка в табл. 3.9).

Троичное моделирование (т.е. определение выходных сигналов элементов по их троичным функциям) дает $z_1 = z_2 = x$. Следовательно, в схеме возможны состязания. В реальной схеме причиной состязаний может быть неодновременная смена входных сигналов а и с.

Таблица 3.9

Номер пункта	Набор	Значения выходных сигналов элементов				
		z_1	z_2	z_3	z_4	z_5
1	0 0 0	1	1	1	1	1
2	1 0 1	1	1	0	0	1
3	x 0 x	x	x	x	x	1

Анализ на состязания последовательностных схем при смене входного набора e_j на e_{j+1} выполняется в два этапа.

На первом этапе (процедура А) изменившимся при подаче e_{j+1} входным переменным приписывается значение «х» и выполняется троичное моделирование. Задача процедуры А – выявить элементы памяти (обратные связи), на состояния которых могут влиять изменения входных переменных.

На втором этапе (процедура В) определяются установившиеся состояния элементов памяти (обратных связей) схемы при подаче e_{j+1} . Для этого входным переменным схемы назначаются их нормальные значения 1 или 0 (такие, как в e_{j+1}) и повторяется троичное моделирование, причем исходное состояние схемы при моделировании совпадает с состоянием схемы после выполнения процедуры А.

Обозначим через Y_k состояние k -го элемента памяти на данном наборе. Тогда справедливо следующее утверждение.

Утверждение 3.2. Если $Y_k = 1(0)$ после применения процедур А и В, то Y_k должно принять на данном наборе значение 1(0) независимо от задержек элементов схемы.

Теперь становится очевидным, что состязания не оказывают влияния на установившееся состояние схемы, если после выполнения процедур А и В для любого k выполняется соотношение $Y_k \neq x$.

Изложенный метод анализа тестов на состязания предполагает, что возможны любые задержки элементов схемы. Поэтому некоторые состязания, обнаруженные этим методом, могут отсутствовать в реальной схеме.

В системах автоматизации построения тестов могут использоваться различные модификации методов анализа на состязания с использованием моделирования. В частности, для повышения адекватности анализа могут применяться алфавиты с более чем тремя символами.

Состязания могут возникать не только на тестовых (созданных специально для целей диагностирования), но и на рабочих воздействиях. Поэтому проектирование сложных цифровых устройств, как правило, ведут с учетом проблемы состязаний.

Для устранения состязаний проектировщики аппаратуры используют следующие приемы:

- 1) введение специальных элементов для создания задержек в заданных частях схемы;
- 2) изменение структуры схемы с целью устранения состязаний логического типа;
- 3) анализ функционирования на рабочих и тестовых входных наборах и их изменение с целью устранения состязаний;
- 4) съём выходных сигналов с задержкой (для устранения влияния опасных состязаний);
- 5) использование синхронизации при разработке схем с памятью.

Широкое распространение синхронных схем и тактируемых элементов памяти (D-триггеров и JK-триггеров) не в последнюю очередь обусловлено стремлением разработчиков избавиться от проблемы опасных состязаний, так как синхронизация позволяет избавиться от детального анализа влияния задержек на функционирование проектируемого изделия.

Контрольные вопросы

1. Дайте определения понятия полнота контроля и приведите общие формулы для ее вычисления.
2. Дайте определение понятия глубина поиска неисправностей.
3. Какие оценки глубины поиска используются, если:
 - при ремонте заменяются все неисправные компоненты, попавшие в список подозреваемых элементов (СПН),
 - осуществляется просмотр сигналов на внутренних контрольных точках с целью выделения фактической неисправности из СПН,
 - в процессе просмотра очередная точка выделяется из условия деления множества подозреваемых неисправностей пополам.
4. Дайте определение и приведите общую формулу для вычисления достоверности контроля. Что понимают под риском заказчика и изготовителя?
5. Как вычисляется вероятность непроверки неисправности s_k при цифровых контролируемых сигналах?
6. Как вычисляется вероятность непроверки неисправности s_k при аналоговых контролируемых сигналах?
7. Пусть в таблице функций неисправностей матрицы M_0 и M_k , описывающие поведение цифрового исправного и неисправного объектов, имеют вид:

$$M_0 = \begin{vmatrix} 01x1 \\ 101x \\ 1111 \end{vmatrix} \text{ и } M_k = \begin{vmatrix} x101 \\ 101x \\ x1x1 \end{vmatrix}$$

Чему равна для этих матриц вероятность m_k непроверки неисправности s_k ?

8. Вычислите вероятности непроверки неисправностей и полноту контроля для ТФН с аналоговыми контролируемыми сигналами, представленной табл. 3.10.

Таблица 3.10

Проверка	M ₀		M ₁		M ₂		M ₃	
	w ₁	w ₂						
e ₁	[1,4]	[0,5]	[1,5]	[2,4]	[1,4]	[1,3]	[1,4]	[2,3]
e ₂	[3,7]	[1,4]	[1,4]	[3,6]	[3,6]	[2,4]	[3,7]	[2,8]

9. Как вычисляется вероятность включения неисправности s_i в СПН при наличии в объекте неисправности s_j и аналоговых контролируемых сигналах?

10. Как вычисляется вероятность включения неисправности s_i в СПН при наличии в объекте неисправности s_j и цифровых контролируемых сигналах?

11. Пусть в таблице функций неисправностей матрицы M_i и M_j , описывающие поведение цифрового объекта с неисправностями s_i и s_j , имеют вид:

$$M_i = \begin{vmatrix} 0 & 1 & x & 1 \\ 1 & 0 & 1 & x \\ 1 & 1 & 1 & 1 \end{vmatrix} \quad \text{и} \quad M_j = \begin{vmatrix} x & 1 & 0 & 1 \\ 1 & 0 & 1 & x \\ x & 1 & x & 1 \end{vmatrix}$$

Чему равна для этих матриц вероятность m_{ij} включения неисправности s_i в СПН при наличии в объекте неисправности s_j ?

12. Вычислите вероятность m_{ij} включения неисправности s_i в СПН при наличии в объекте неисправности s_j для ТФН с аналоговыми контролируемыми сигналами, представленной табл. 3.10.

13. Что является причиной состязаний?

14. В чем отличие функциональных и логических состязаний?

15. Нарисуйте временные диаграммы, иллюстрирующие форму сигнала при статическом и динамическом состязаниях и установившемся состоянии «логический 0».

16. Нарисуйте временные диаграммы, иллюстрирующие форму сигнала при статическом и динамическом состязаниях и установившемся состоянии «логическая 1».

17. Какие состязания называют опасными?

18. Какие способы используют проектировщики электронных схем для устранения состязаний?

Рекомендуемая литература

ГОСТ 20911-89. Техническая диагностика. Термины и определения. – М.: Изд-во стандартов, 1990.

ГОСТ 27002-89. Надежность в технике. Основные понятия. Термины и определения. – М.: Изд-во стандартов, 1990

Карибский, В.В. Основы технической диагностики / В.В. Карибский, П.П. Пархоменко, Е.С. Согомонян, В.Ф. Халчев. Кн. 1. – М.: Энергия, 1976. – 464 с.

Клюев, В.В. Технические средства диагностирования: справочник / В.В. Клюев, П.П. Пархоменко и др.; под общ. ред. В.В. Ключева. – М.: Машиностроение, 1989. – 672 с.

Мальшенко, Ю.В. Автоматизация диагностирования электронных устройств / Ю.В. Мальшенко, В.П. Чипулис, С.Г. Шаршунов. – М.: Энергоатомиздат, 1986. – 216 с.

Мальшенко, Ю.В. Оценка полноты теста при наличии аналоговых контролируемых параметров / Ю.В. Мальшенко // Автоматика и телемеханика. – 1996. – № 7. – С. 179–189.

Мальшенко, Ю.В. Техническая диагностика / Ю.В. Мальшенко. – Владивосток: Изд-во ВГУЭС, 1998. Ч. 1. – 68 с.

Тема 4. МЕТОДЫ ПОСТРОЕНИЯ ТЕСТОВ ЦИФРОВЫХ УСТРОЙСТВ

4.1. Понятия контролирующего и диагностического тестов

Общая классификация видов тестов

Задача построения теста состоит в том, чтобы найти такую совокупность и последовательность входных воздействий, при подаче которой на объект диагностирования получаемые ответы объекта в заданных контрольных точках позволяют делать заключение о его реальном техническом состоянии.

Без хорошего теста невозможно обеспечить высокую достоверность и полноту контроля, быстро находить неисправные компоненты.

В общем случае **тестом называют** любую последовательность сигналов, подаваемую на входы объекта диагностирования (ОД) для целей диагностирования.

Как уже отмечалось, диагностирование во времени можно разбить на ряд интервалов (проверок), каждому из которых сопоставляется некоторое тестовое воздействие (входной набор) и результат измерений.

Пусть E – множество всех возможных последовательностей входных наборов, любую из которых можно подать на ОД в процессе диагностирования; $S = \{s_1, \dots, s_r\}$ – множество допустимых неисправностей; T_k – некоторая конкретная последовательность входных наборов $T_k \in E$; P_m – максимально возможная полнота контроля на последовательностях входных наборов из E .

Определение 4.1. Последовательность $T_k \in E$ называется **контролирующим тестом**, если на последовательности входных наборов T_k обеспечивается полнота контроля, равная P_m .

Иными словами, контролирующий тест – это такой тест, который обеспечивает максимально возможную полноту контроля.

Допустим, что на тесте T отсутствуют условно-проверяемые неисправности, а $M_0(T)$ и $M_i(T)$ – матрицы таблицы функции неисправностей (ТФН) на тесте T для исправного состояния s_0 и при неисправности $s_i \in S$ соответственно. Тогда можно утверждать, что T является контролирующим тестом, если для любой проверяемой на E неисправности $s_i \in S$ выполняется условие $M_0(T) \cap M_i(T) = \emptyset$.

Пусть R_m – минимально возможная глубина поиска неисправностей с использованием последовательностей из E .

Определение 4.2. Последовательность $T_d \in E$ называется **диагностическим тестом**, если на ней обеспечивается глубина поиска неисправностей, равная R_m .

Иными словами, диагностическим является тест, на котором обеспечивается максимально возможная различимость неисправностей.

Допустим, что на тесте T отсутствуют условно-различимые неисправности. Тогда можно утверждать, что T является диагностическим, если для любой пары s_i, s_j неисправностей, различимых на E , выполняется условие: $M_i(T) \cap M_j(T) = \emptyset$.

Очевидно, что свойства определений 4.1 и 4.2 могут выполняться для нескольких тестов, обладающих разной длиной (обычно длину теста принимают равной числу входящих в него входных наборов).

Определение 4.3. Контролирующий (диагностический) тест T называют **тупиковым контролирующим (диагностическим)**, если удаление из него любого входного набора приводит к изменению полноты контроля (глубины поиска неисправностей).

Заметим, что в конкретных случаях возможно несколько тупиковых контролирующих (диагностических) тестов, различающихся составом и количеством входных наборов.

При проведении диагностирования желательно иметь тесты как можно меньшей длины. Это сокращает время диагностирования и объем необходимого информационного обеспечения.

Случайные и детерминированные методы построения тестов

Задача построения теста одна из главных и наиболее сложных в технической диагностике. Как правило, современные системы автоматизации проектирования электронных устройств имеют специальные подсистемы автоматизации построения тестов.

В большинстве случаев на практике не удается построить тесты со 100% полнотой контроля или глубиной поиска до одной конкретной неисправности, хотя существуют методы, гарантирующие достижение таких показателей. Причины этого – высокие временные и вычислительные затраты или наличие избыточных компонент. Неисправности последних в принципе непроверяемы по внешним контролируемым выходам без доступа к внутренним узлам ОД.

В связи с высокой сложностью и трудоемкостью процесса построения тестов широко применяются **интерактивные системы** построения тестов, которые предполагают активное участие человека в этом процессе.

Различают методы случайного и детерминированного формирования тестов.

Первые предполагают использование алгоритмов случайного подбора кандидатов для включения в создаваемый тест. Выбранные кандидаты в тест анализируются с целью оценки их проверяющих возможностей. Если новый случайно выбранный входной набор проверяет дополнительные неисправности, то он включается в тест. Возможно, что для части внешних входов устанавливается определенный (не случайный) порядок изменения сигналов. Например, это могут быть входы цепей синхронизации.

Алгоритмы вторых ориентированы на целенаправленный подбор кандидатов для включения в создаваемый тест. При этом значения сигналов выбираются

исходя из функций компонент и структуры ОД. Обычно на очередном этапе работы алгоритма выбирается некоторая из возможных неисправностей и конкретно для нее подбирается проверяющий входной набор.

При использовании детерминированных методов процесс построения теста в общем случае состоит из следующих этапов:

- 1) определение списка допустимых неисправностей;
- 2) подбор тестового набора для проверки очередной неисправности из списка;
- 3) моделирование схемы на тестовом наборе для выявления подмножества проверяемых на данном наборе неисправностей;
- 4) исключение проверяемых неисправностей из исходного списка допустимых и включение выбранного набора в состав теста;
- 5) определение полноты проверки схемы на построенном тесте;
- 6) если полнота удовлетворяет разработчика, то завершение алгоритма, иначе переход на п. 2.

В теории технической диагностики наиболее глубоко формализованы и проработаны задачи построения тестов для цифровых схем. Ниже мы рассмотрим основные методологии построения тестов для цифровых схем. Заметим, что при соответствующих модификациях некоторые из них могут использоваться для аналого-цифровых и аналоговых устройств.

Хотя методы случайного подбора входных наборов довольно просты для компьютерной реализации, но они используются, скорее, в качестве дополнений к детерминированным методам. Современные системы автоматизации построения тестов реализуют некоторую группу методов. Например, первоначально может использоваться случайный метод подбора кандидатов для включения в тест. В ходе формирования теста постоянно оценивается его полнота (глубина поиска). Если эффективность случайного подбора падает, то переходят на какой-либо детерминированный метод, задавая неисправности, проверка (различимость) которых улучшает показатель качества теста.

4.2. Получение тупиковых тестов с использованием таблиц неисправностей

Общий подход

Первыми теоретическими работами, с которых собственно и ведется отчет образования технической диагностики как самостоятельного научного направления, являются исследования по получению тупиковых и минимальных тестов для релейно-контактных схем (Чегис, Яблонский, 1955).

В качестве модели ОД при этом использовалась двоичная ТН, в которой все столбцы различны (одинаковые столбцы объединялись в один).

Нетрудно заметить, что тогда контролирующему тесту соответствует совокупность строк ТН, в которых в каждом столбце имеется хотя бы одна единица, а диагностическому – совокупность строк, на которых любая пара столбцов хотя бы в одной из этих строк имеет разные значения.

Предложенный общий подход к определению тупиковых и минимальных тестов состоял в следующем.

Пусть ТН содержит r столбцов, причем i -й столбец сопоставлен i -й неисправности ($i=1, \dots, r$).

1. Для каждого столбца определяем множество строк L_i ($i=1, \dots, r$), в которых в i -м столбце имеется 1.

2. Рассматривая обозначения строк как некоторые логические переменные, составляем логические выражения, где \sum_i – логическая сумма переменных, сопоставленных строкам из L_i .

3. Составляем логическое выражение $\prod_{i=1}^r \sum_i$.

4. Полученное выражение с применением операций алгебры логики приводим к выражению $\sum \Pi$ (т.е. к дизъюнктивной форме). При этом с переменными выполняются преобразования вида

$$A A = A; A \vee AB = A; (A \vee B)(A \vee C) = A \vee BC.$$

5. Переменные, входящие в одну конъюнкцию результирующего выражения, образуют тупиковый тест. Конъюнкции с наименьшим числом букв определяют минимальные тупиковые тесты.

Рассмотрим изложенную методику на примере ТН, представленной табл. 4.1.

Таблица 4.1

ТН для иллюстрации методики определения тупиковых тестов

Наборы	Номера неисправностей					
	1	2	3	4	5	6
a	1	1	0	0	0	0
b	1	0	1	0	1	0
c	1	0	0	1	0	0
d	1	0	1	0	0	1
e	0	1	0	0	0	0
f	0	1	0	0	1	0
g	0	0	1	1	0	0
h	0	0	1	0	0	1

1. Для данных табл. 4.1 имеем:

$$L_1 = \{a, b, c, d\}; L_2 = \{a, e, f\}; L_3 = \{b, d, g, h\};$$

$$L_4 = \{c, g\}; L_5 = \{b, f\}; L_6 = \{d, h\}.$$

2. Логические суммы Σ_i для каждого столбца имеют вид:

$$\Sigma_1 = a \vee b \vee c \vee d;$$

$$\Sigma_2 = a \vee e \vee f;$$

$$\Sigma_3 = b \vee d \vee g \vee h;$$

$$\Sigma_4 = c \vee g;$$

$$\Sigma_5 = b \vee f;$$

$$\Sigma_6 = d \vee h.$$

3. Составляем логическое произведение:

$$\Pi\Sigma = (a \vee b \vee c \vee d)(a \vee e \vee f)(b \vee d \vee g \vee h)(c \vee g)(b \vee f)(d \vee h).$$

4. После раскрытия скобок и преобразований выражения $\Pi\Sigma$ получаем:

$$\Sigma\Pi = dcf \vee dfg \vee cfh \vee afgh \vee bfg \vee abcd \vee abdg \vee abch \vee abgh \vee bcde \vee bdeg \vee bceh \vee begh.$$

5. Каждая конъюнкция итогового выражения, полученного при выполнении п. 4, определяет тупиковый тест. При этом конъюнкции dcf , dfg и cfh определяют минимальные (по длине) тупиковые тесты.

Проверим, например, является ли совокупность наборов dfg тупиковым контролирующим тестом. Для этого выпишем неисправности, которые проверяются на каждом из этих наборов: d проверяет неисправности $\{1, 3, 6\}$, f – $\{2, 5\}$, g – $\{3, 4\}$. Объединение полученных списков неисправностей содержит все 6 возможных неисправностей. Следовательно, совокупность наборов dfg является контролирующим тестом. Этот тест является тупиковым, так как удаление из него любого набора приведет к уменьшению числа проверяемых неисправностей.

Аналогичная методика может использоваться для определения по ТН и тупиковых диагностических тестов. Для этого предварительно надо преобразовать ТН в таблицу (назовем ее таблицей различимости неисправностей), в которой столбцы соответствуют всем возможным парам неисправностей из исходной ТН. В клетках столбца проставляется результат сравнения по $\text{mod}2$ значений из столбцов соответствующей пары неисправностей. Результат такого преобразования для табл. 4.1 представлен в табл. 4.2 (здесь и далее в клетках таблицы проставлены только единицы, пустым клеткам сопоставлены нули). Любая совокупность строк, покрывающая единицами все столбцы табл. 4.2, является диагностическим тестом. Поэтому, применив к данным табл. 4.2 вышеизложенный метод, получим выражение $\Sigma\Pi$, каждая конъюнкция которого определяет тупиковый диагностический тест.

В общем случае, исходная ТН содержит для каждой неисправности несколько столбцов (по числу КП) для каждой неисправности. Однако для рассматриваемой задачи ее легко привести к ТН с одним столбцом для каждой неисправности. В новой ТН строка столбца i содержит «1», если в этой строке в исходной ТН содержится хотя бы одна «1».

Таблица различимости неисправностей

Наборы	Пары неисправностей														
	1,2	1,3	1,4	1,5	1,6	2,3	2,4	2,5	2,6	3,4	3,5	3,6	4,5	4,6	5,6
a		1	1	1	1	1	1	1	1						
b	1		1		1	1		1		1		1	1		1
c	1	1		1	1		1			1			1	1	
d	1		1	1		1			1	1	1			1	1
e	1					1	1	1	1						
f	1			1		1	1		1		1		1		1
g		1	1			1	1				1	1	1	1	
h		1			1	1			1	1	1			1	1

Упрощение процесса определения тупикового теста

Изложенный выше формальный метод определения тупиковых тестов является трудоемким даже для небольших таблиц. При этом определяются все возможные тупиковые тесты. Однако в практическом плане достаточно знать только один тупиковый тест. Поэтому имеет смысл рассматривать задачу получения только одного тупикового теста, что потенциально существенно упрощает трудоемкость решения.

Рассмотрим один из методов, ориентированных на определение только одного тупикового теста. Этот метод предполагает возможность предварительного уменьшения размера таблицы, которая используется для определения теста. Очевидно, чем менее объемна исходная таблица, тем менее трудоемко решение.

Строки и столбцы ТН удаляются по следующим правилам (особенность этих правил в том, что после удаления оставшаяся информация достаточна для определения, по крайней мере, одного тупикового теста).

1. Из таблицы удаляются все поглощаемые строки. Строка называется поглощаемой, если все ее единицы покрываются единицами другой строки. Например, из трех строк

```
010101010
101010110
110111010
```

поглощаемой является первая строка. Она поглощается третьей строкой.

2. Из таблицы удаляются все поглощающие столбцы. Столбец называется поглощающим, если он покрывает единицами все единицы другого столбца. Например, из трех столбцов

```
111
001
110
000
101
```

первый столбец является поглощающим. Он содержит единицы в тех же строках, где они имеются во втором столбце.

3. Если k -я строка включена в тест, то выбор остальных строк может выполняться по таблице, полученной из исходной, путем удаления k -й строки и столбцов, которые она покрывает.

Рассмотрим применение этих правил для определения тупикового теста по ТН, представленной табл. 4.1.

Строка e поглощается строками a и f , столбец 3 поглощает столбец 6. В результате применения правил 1 и 2 получаем табл. 4.3. Теперь надо выбрать строку (набор), которая будет включена в текст. Обычно выбирают строку с наибольшим числом единиц. В нашем примере все строки содержат по две единицы, поэтому можно выбрать любую. Допустим, что это строка a . Включаем ее в тест. Тогда, применяя правило 3, получаем табл. 4.4.

Таблица 4.3

ТН после удаления строки e и столбца 3

Наборы	Номера неисправностей				
	1	2	4	5	6
a	1	1			
b	1			1	
c	1		1		
d	1				1
f		1		1	
g			1		
h					1

Таблица 4.4

ТН после удаления столбцов с единицами в строке a

Набор	Номера неисправностей		
	4	5	6
b		1	
c	1		
d			1
f		1	
g	1		
h			1

После удаления поглощенных строк f , g и h однозначно получаем тест $T = \{a, b, c, d\}$, являющийся тупиковым. Нетрудно заметить, что с помощью рассмотренного способа мы в общем случае получим тупиковый тест, который лишь случайно может оказаться одним из минимальных. Если рассмотреть все возможные варианты удаления строк и столбцов, то можно получить все тупиковые тесты, включая минимальные.

Исследуем, например, вариант включения в тест строки f . После удаления из табл. 4.3 столбцов, покрываемых строкой f , получим табл. 4.5.

Таблица 4.5

ТН после включения в тест строки f

Набор	Номера неисправностей		
	1	4	6
a	1		
b	1		
c	1	1	
d	1		1
g		1	
h			1

В результате поглощения строк a , b , g строкой c и строки h строкой d однозначно получим тест $T = \{c, d, f\}$. Он короче предыдущего и для данного примера является минимальным.

Приближенные методы минимизации ТН, как правило, представляют собой итерационные процедуры выбора строк. При этом на каждом шаге выбирается «наилучшая» строка в соответствии с определенным критерием. Эти критерии могут быть различными.

Например, можно на каждом шаге выбирать набор, обеспечивающий проверку максимального числа еще непроверенных неисправностей. Другой способ предполагает на каждом шаге определение столбца, содержащего наименьшее число единиц. Если столбец содержит только одну единицу, то строка с единицей безусловно включается в тест. Если все строки содержат по несколько единиц, то в тест включается строка с наибольшим числом единиц.

4.3. Метод активизации одномерного пути

Этот метод является одним из исторически первых структурных подходов к построению тестов цифровых ОД. Основная идея метода заключается в построении пути от места неисправности, например, от элемента D_i (если неисправен D_i) через последовательность некоторых элементов D_2, \dots, D_N до одного из внешних контролируемых выходов объекта. Входные сигналы элементов

D_2, \dots, D_N , блокирующие прохождение сигнала от D_i , выбираются таким образом, чтобы по значению сигнала на выбранном внешнем контролируемом выходе можно было судить о состоянии элемента D_i .

При этом на входы неисправного элемента D_i необходимо подать сигналы, при которых неисправность проявляется на его выходе. Создаваемый вышеуказанным образом путь называют активизированным, что и определило название метода.

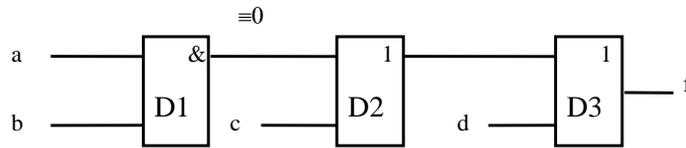


Рис. 4.1. Формирование активизированного пути

Пусть в цифровой схеме рис. 4.1 имеется неисправность, эквивалентная фиксации выхода элемента D1 в значении «лог.0». Чтобы активизировать путь от D1 до выхода f надо подать $c=0$ и $d=0$. Для проявления неисправности по выходу элемента D1 надо подать $a=b=1$. Таким образом входной набор, на котором проверяется рассматриваемая неисправность $a=b=1, c=d=0$.

Процедуру формирования тестового воздействия для проверки некоторой неисправности можно сформулировать в виде следующих операций.

1. Определяем условия, при которых заданная неисправность проявляется в месте ее возникновения.
2. Выбираем последовательность элементов (путь), через которые неисправность (точнее, сигнал с выхода неисправного элемента) будет транспортироваться до контролируемого выхода.
3. Определяем условия активности выбранного пути. При этом для каждого элемента пути (выбранного в п. 2) задаем входные сигналы таким образом, чтобы его выход зависел от выходного сигнала предыдущего элемента пути.
4. Определяем комбинацию значений входных сигналов схемы, при которых неисправность проявляется в месте ее возникновения и транспортируется на контролируемый выход схемы.

Пункты 1–3 приведенного алгоритма часто называют прямой фазой метода активизации одномерного пути, а п. 4 – обратной фазой. Чтобы построить тест для устройства в целом, данный алгоритм применяется последовательно для каждой неисправности из множества допустимых. Тест устройства представляет собой последовательность тестовых наборов, полученных для каждой неисправности.

На рисунке 4.2 показаны значения сигналов на линиях схемы в результате выполнения рассмотренного алгоритма для неисправности «линия $i=0$ ». В процессе выполнения «прямой фазы» алгоритма линиям присваиваются значения: $a=1, b=1, k=0, e=1$. В «обратной фазе» алгоритма надо определить значения c и d , при которых $k=0$. Такими будут значения: $c=0, d=0$ (в данном случае можно вы-

брать и другую комбинацию значений, обеспечивающих $k=0$, например $c=1$ и $d=0$).

Таким образом, заданную неисправность («линия $i=0$ ») проверяет тестовый набор $abcde=11001$.

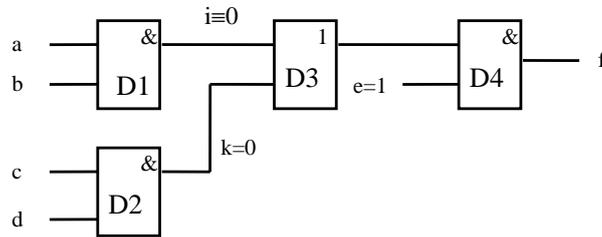


Рис. 4.2. Схема для иллюстрации метода активизации одномерного пути

Метод прост и удобен для ручного использования, однако возможны неисправности, для проверки которых необходимо активизировать несколько путей. Обычно это следствие наличия в схеме сходящихся разветвлений. На рисунке 4.3 приведена схема, в которой для проверки неисправности «1 на выходе элемента D1» необходимо активизировать одновременно два пути: через элементы D2, D4 и через элементы D3, D4.

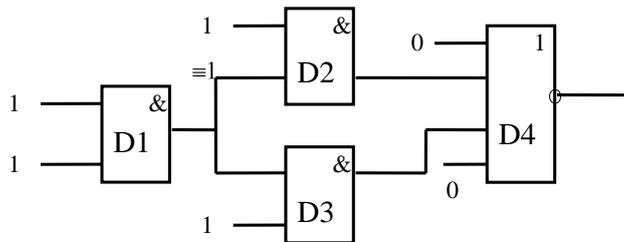


Рис. 4.3. Схема, в которой надо активизировать одновременно два пути

4.4. d-алгоритм

Понятие логического куба

Один из популярных формальных методов реализации структурного подхода к построению тестов, при котором обеспечивается активизация множества путей, является **d-алгоритм**.

Общее решение задачи построения теста на основе идеи активизации пути наиболее удачно воплощено в предложенном Ротом (Roth J.P., 1967) d-алгоритме. В основе его лежат понятия логических кубов различного вида и правила действия над кубами, что делает удобным реализацию алгоритма на ЭВМ.

Под логическим кубом понимается вектор размерностью n , каждая координата которого имеет одно из пяти значений: $0, 1, x, d, d^*$.

Здесь 0 и 1 представляют обычные булевы значения, x – символ неопределенного или безразличного значения. Символ d может иметь значение логическая «1» либо «0», конкретное значение устанавливается в процессе работы алгоритма, d^* – инверсия значения d .

Над парой логических кубов $A = (a_1, a_2, \dots, a_n)$ и $B = (b_1, b_2, \dots, b_n)$ можно производить операцию пересечения, которая выполняется поразрядно по следующим правилам:

$$a_i \cap b_j = \begin{cases} a_i, & \text{если } b_j = x \text{ или } a_i = b_j \\ a_i \cap b_j, & \text{если } a_i = x, \\ \emptyset, & \text{в остальных случаях} \end{cases}$$

Исходная информация для выполнения алгоритма

По рассматриваемому алгоритму описание каждого элемента схемы содержит три вида кубов, которые далее используются при построении тестов: **вырожденные (сингулярные) кубы, d-кубы элементов и d-кубы неисправностей элементов.**

Примеры сингулярных кубов, d-кубов элементов и d-кубов неисправностей для элементов И, ИЛИ-НЕ и M2 приведены в подразделе «Описание объекта с помощью кубов» (тема 2, рис. 2.7, табл. 2.2).

Пример выполнения d-алгоритма

d-алгоритм предполагает, что тестовое воздействие (входной набор) определяется отдельно для каждой неисправности. Совокупность полученных наборов образует тест для всего множества допустимых неисправностей.

В d-алгоритме можно выделить следующие четыре основные операции построения тестового набора для проверки заданной неисправности.

1. Выбирается d-куб заданной неисправности.
2. Активируются возможные пути от элемента с выбранной неисправностью к контролируемому выходу. При этом применяется операция пересечения d-куба неисправности с d-кубами всех элементов путей от неисправности до выхода схемы (прямая фаза d-алгоритма).
3. Выполняется пересечение полученного в п. 2 d-куба с сингулярными кубами остальных элементов с целью обеспечения условий активизации, заданных этим d-кубом (обратная фаза d-алгоритма).

Пример. Необходимо определить тестовое воздействие для неисправности «0» на втором входе (линия 8) элемента D3 схемы рис. 4.4.

В рассматриваемой схеме пронумеровано 12 линий. Удобно вести вычисления, представляя кубы в виде 12-разрядных векторов.

Прежде чем начать выполнение d-алгоритма, следует подготовить исходную информацию, которая понадобится при его исполнении. Это сингулярные и d-кубы элементов (были приведены в теме 2), а также d-куб заданной неисправности. Последний имеет вид (в верхней строке даны номера линий схемы рис. 4.4):

7 8 9 11
0 1 0 d*

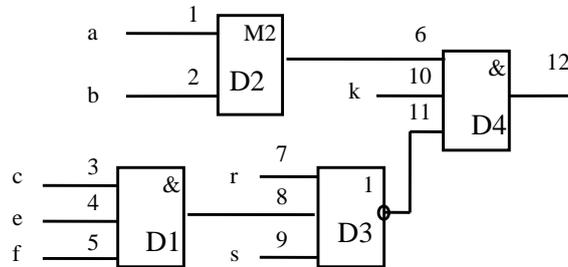


Рис. 4.4. Схема для иллюстрации d-алгоритма

Координате (11), соответствующей выходу элемента D3, присваиваем значение d^* , так как в исправном состоянии на выходе элемента при подаче на вход комбинации 010 будет значение «0».

Если для заданной неисправности существует несколько d-кубов неисправности, то первоначально можно выбрать любой из них.

После подготовки исходной информации можно перейти к операциям d-алгоритма.

1. Выбираем d-куб неисправности « = 0 на втором входе элемента D3» и представляем его в формате 12-разрядного логического куба (в верхней строке даны номера линий):

1 2 3 4 5 6 7 8 9 10 11 12

$CO = x x x x x 0 1 0 x d^* x$.

2. Активизируем путь от неисправного элемента до выхода схемы. В данном случае от выхода D3 имеется только один путь к выходу схемы (через элемент D4). Выполняем пересечение куба CO с d-кубом C1 элемента D4, имеющим символ d^* на третьем входе:

1 2 3 4 5 6 7 8 9 10 11 12

$CO = x x x x x 0 1 0 x d^* x$

$C1 = x x x x x 1 x x x 1 d^* d^*$

 $C2 = x x x x x 1 0 1 0 1 d^* d^*$

Появление в результирующем кубе символа d^* на выходной координате (т.е. на линии 12) свидетельствует об окончании прямой фазы d -алгоритма.

3. Выполняем обратную фазу d -алгоритма. Для рассматриваемой схемы она будет состоять в последовательном пересечении куба $C2$ с сингулярными кубами элементов $D1$ и $D2$. При этом из сингулярных кубов элемента $D1$ необходимо выбрать тот, в котором на его выходе значение 1, так как в кубе $C2$ координата 8 определена равной 1.

$$\begin{array}{r}
 \mathbf{1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12} \\
 C2 = x\ x\ x\ x\ x\ 1\ 0\ 1\ 0\ 1\ d^*\ d^* \\
 C3 = x\ x\ 1\ 1\ 1\ x\ x\ 1\ x\ x\ x\ x \\
 \text{-----} \\
 C4 = x\ x\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ d^*\ d^*
 \end{array}$$

Из сингулярных кубов элемента $D2$ следует выбрать тот, в котором на выходе значение 1, так как в кубе $C4$ координата 6 имеет значение 1. Таких кубов несколько. Первоначально можно взять любой.

$$\begin{array}{r}
 \mathbf{1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12} \\
 C4 = x\ x\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ d^*\ d^* \\
 C5 = 0\ 1\ x\ x\ x\ 1\ x\ x\ x\ x\ x\ x \\
 \text{-----} \\
 C6 = 0\ 1\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ d^*\ d^*
 \end{array}$$

В полученном кубе $C6$ нет неопределенных (т.е. равных «х») координат. Поэтому выполнение d -алгоритма закончено.

4. На последнем шаге по координатам 1–5, 7, 9, 10 определяем входной набор, проверяющий заданную неисправность («0» на линии 8):

$$\begin{array}{r}
 \mathbf{a\ b\ c\ e\ f\ r\ s\ k} \\
 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1.
 \end{array}$$

Возможно, что при выполнении п. 2 или п. 3 в результате пересечения будет получено значение \emptyset («пусто»). Тогда следует взять другой куб для пересечения (если такой существует). Если все же после перебора всех кубов (в том числе всех вариантов d -кубов рассматриваемой неисправности) не удалось избавиться от значения \emptyset , то это означает, что данная неисправность не может быть проверена, т.е. в схеме имеется избыточность.

Выше мы рассмотрели принципы работы d -алгоритма применительно к комбинационным схемам. Существует модификация алгоритма применительно к схемам с элементами памяти. При этом основные операции и состав исходной информации сохраняются, но процесс поиска воздействий, проверяющих заданную неисправность, усложняется, так как приходится учитывать особенности функционирования схем с памятью.

Если схема с памятью имеет обратные связи, то надо делать условные разрывы и вводить псевдовходы и псевдовыходы. Возможно, в процессе функцио-

нирования схемы сигналы на линиях обратной связи изменяются несколько раз, пока не установятся. Поэтому методика применения алгоритма при разрыве обратных связей предполагает введение копий блоков (элементов), охваченных обратной связью (i -я копия имитирует работу схемы в момент $t+i$, где t – момент подачи тестового воздействия). Вообще говоря, каждый элемент памяти – это схема с обратной связью. Однако если элемент памяти рассматривается как макроэлемент, обратные связи которого находятся внутри этого элемента, то можно использовать методику определения тестового воздействия, применяемую для комбинационных схем. Однако надо иметь в виду, что подобранное тестовое воздействие, возможно, проверяет неисправность только при определенных состояниях элементов памяти. Тогда требуется решить задачу синтеза установочной последовательности. При диагностировании реального объекта надо будет подать на объект сначала эту последовательность, а затем уже подобранное тестовое воздействие. Иными словами, для обеспечения проверки некоторой неисправности может потребоваться подать на объект несколько тестовых воздействий.

4.5. Построение теста по методу булевой производной

Условия проверки неисправности

В пп. 2.1 указывалось, что под **булевой производной** функции $f(x) = f(x_1, x_2, \dots, x_n)$ по x_i понимают функцию $df(x)/dx_i = f(x_1, x_2, \dots, x_i, \dots, x_n) \oplus f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n)$, где \oplus – сумма по модулю 2.

Булева производная, если ее значение равно «1», определяет значения логических переменных x_1, \dots, x_n , при которых изменение значения переменной x_i приводит к изменению значения функции $f(x)$. Допустим, на комбинации A этих переменных $df(x)/dx_i = 1$. Тогда из вышеуказанного свойства булевой производной следует, что при комбинации значений A значение $f(x)$ зависит от значения переменной x_i .

Неисправность приводит к фиксации x_i в конкретное значение 1 или 0. Естественно, для того чтобы проверить неисправность, при которой $x_i=1(0)$, комбинация A должна обеспечить на этом входе значение $x_i=0(1)$.

Таким образом, **условие проверки неисправности** $x_i=0$ можно записать в виде $x_i \cdot df(x)/dx_i = 1$. Аналогично, условие проверки неисправности $x_i=1$ записывается в виде $\bar{x}_i \cdot df(x)/dx_i = 1$.

Обычно функция $f(x)=f(x_1, x_2, \dots, x_n)$ записывается как функция от входных переменных (входов) схемы. Если рассматривать неисправности на внутренних линиях схемы, то при записи функции надо вводить переменные, сопоставляемые с местом появления неправильных сигналов (их называют внутренними переменными).

Обозначим такую внутреннюю переменную через z_i . Тогда условие проверки неисправности $z_i=0$ можно записать в виде $z_i \cdot df(x)/dz_i = 1$. Аналогично, условие проверки неисправности $z_i=1$ записывается в виде $\bar{z}_i \cdot df(x)/dz_i = 1$.

Алгоритм и примеры определения тестовых наборов

Как и в случае d-алгоритма, построение теста ведется путем последовательного подбора проверяющих входных наборов для отдельных неисправностей. Тест образуется путем объединения выбранных наборов.

Алгоритм выбора проверяющих наборов для отдельной неисправности комбинационной схемы можно представить в виде следующих операций.

1. Составляем функцию $f(x)$, в которой в качестве переменной присутствует z , сопоставленная линии, фиксацию которой в 1(0) надо проверить.
2. Определяем частную булеву производную $df(x) / dz$, приводим полученное выражение к дизъюнктивной форме (ДФ).
3. Выбираем один из термов (например, t), полученной в п.2 ДФ.
4. Неисправность $z = 0$ проверяется на воздействии, при котором значения переменных x_1, \dots, x_n обеспечивают условие $z \cdot t = 1$.
5. Неисправность $z = 1$ проверяется на \bar{z} воздействии, при котором значения переменных x_1, \dots, x_n обеспечивают условие $\bar{z} \cdot t = 1$.

Приведем несколько примеров вычисления тестов методом булевой производной.

Пример. Дана схема (рис. 4.5), реализующая функцию $f(x) = x_1 x_2 \vee x_3$. Найти тесты неисправностей $x_1 = 0$ и $x_1 = 1$.
Найдем булеву производную $df(x) / dx_1$:

$$df(x) / dx_1 = (1 \cdot x_2 \vee x_3) \oplus (0 \cdot x_2 \vee x_3) = (x_2 \vee x_3) \oplus x_3 = (x_2 \vee x_3) \bar{x}_3 \vee (\bar{x}_2 \bar{x}_3) x_3 = x_2 \bar{x}_3.$$

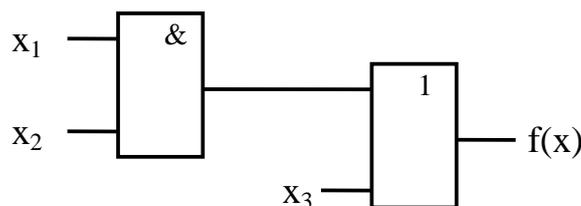


Рис. 4.5. Древоидная схема с неисправностью входа x_1

Тест для $x_1 = 0$ определим из условия $x_1 \cdot df(x) / dx_1 = 1$, то есть $x_1 x_2 \bar{x}_3 = 1$. Следовательно, $x_1 = 1, x_2 = 1, x_3 = 0$.

Тест для $x_1 = 1$ определим из условия $\bar{x}_1 df(x) / dx_1 = 1$, то есть $\bar{x}_1 \bar{x}_2 \bar{x}_3 = 1$.

Следовательно, $x_1 = 0, x_2 = 1, x_3 = 0$.

Пример. Для схемы рис. 4.6 найти тестовые наборы для проверки неисправностей $x_2 = 0$ и $x_2 = 1$.

Схема реализует функцию $f(x) = x_1 x_2 \vee x_1 \bar{x}_2$.

Пользуясь формулой для вычисления булевой производной $df(x) / dx_2$, найдем ее значение:

$$df(x) / dx_2 = (x_1 \cdot 1 \vee x_1 \cdot 0) \oplus (x_1 \cdot 0 \vee x_1 \cdot 1) = 0.$$

Это означает, что $f(x)$ не зависит от x_2 , то есть неисправности $x_2 = 0$ и $x_2 = 1$ являются непроверяемыми.

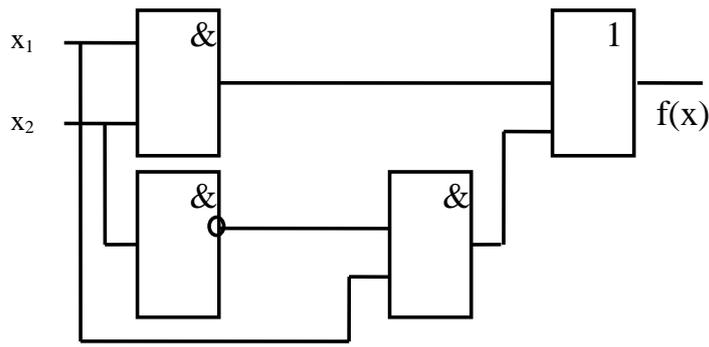


Рис. 4.6. Схема с разветвлениями и неисправностью входа x_2

Пример. Дана схема рис. 4.7. Найти тестовый набор для проверки неисправности $y_6 = 0$. В данном примере выбрана неисправность внутренней линии схемы. Выразим $f(x)$ через внутренние переменные y_5 и y_6 :

$$f(x) = \overline{y_5 \vee y_6}.$$

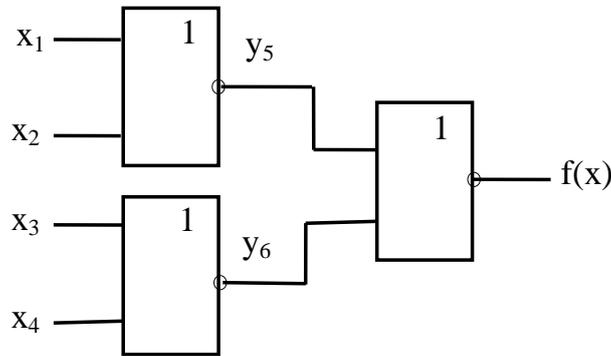


Рис. 4.7. Схема с неисправностью внутренней линии

Найдем булеву производную $df(x)/dy_6$:

$$df(x)/dy_6 = \overline{(y_5 \vee 0)} \oplus \overline{(y_5 \vee 1)} = \overline{y_5} \oplus 0 = \overline{y_5} = x_1 + x_2.$$

Тест для $y_6 = 0$ найдем из условия $y_6 \cdot df(x)/dy_6 = 1$, то есть

$$\overline{(x_3 \vee x_4)}(x_1 \vee x_2) = \overline{x_3} \overline{x_4}(x_1 \vee x_2) = x_1 \overline{x_3} \overline{x_4} \vee x_2 \overline{x_3} \overline{x_4} = 1.$$

Следовательно, тестом для проверки рассматриваемой неисправности является набор $x_1=1, x_2=1, x_3=0$ и $x_4=0$.

Если от проверяемой точки имеются несколько путей (что возможно при наличии точек разветвления) к контролируемому выходу, то дополнительно

необходимо определять входные наборы для проверки всех путей от точки разветвления. В точке разветвления переменная x_i «расщепляется» на каждый путь. Например, в примере рис. 4.8 переменная x_i расщепляется на две переменные x_{i1} и x_{i2} . В данном случае надо определять производную и проверяющие наборы не только для точки x_i , но и для каждой из переменных x_{i1} и x_{i2} , точнее, путей от этих переменных до контролируемого выхода схемы. При этом каждое ответвление от рассматриваемой точки может требовать своего входного набора для проверки. Производная для конкретного пути определяется как произведение производной x_i на производные внутренних переменных, лежащих на рассматриваемом пути.

Пусть схема (рис. 4.8) реализует функцию f , а путь P_{i1} от x_i проходит через внутренние точки y_a, y_c, y_d . Тогда условие проверки правильности прохождения сигнала от точки разветвления x_i по пути P_{i1} будет определяться условием: $x_i \cdot df(P_{i1})/dx_i = 1$ (для неисправности « $x_{i1}=0$ ») и $\bar{x}_i \cdot df(P_{i1})/dx_i = 0$ (для неисправности « $x_{i1}=1$ »).

При этом производная пути P_{i1} определяется выражением:

$$df(P_{i1})/dx_i = (df/dy_d)(dy_d/dy_c)(dy_c/dy_a)(dy_a/dx_i).$$

От x_i сигнал поступает на выход схемы и по другому пути, который проходит через элемент с выходом y_b . Поэтому надо определить входной набор для проверки линии от x_i до y_b , для чего потребуется определить производную x_i по данному пути. В итоге, получим отдельные тестовые наборы для проверки обоих ответвлений от x_i .

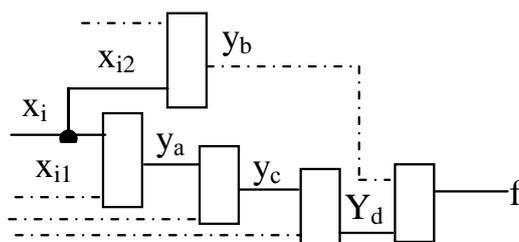


Рис. 4.8. Выбор пути в схеме с разветвлением

Метод может применяться для случая схем с элементами памяти.

В общем виде процесс определения входных наборов для проверки x_i в схеме с элементами памяти и разветвлениями можно представить из следующих операций:

- 1) описываем схему системой уравнений, где каждой линии обратной связи сопоставляется некоторая переменная;
- 2) выбираем некоторый путь P_{i1} от x_i до некоторого выхода z ;
- 3) определяем производную $dz(P_{i1})/dx_i$ и приводим ее к дизъюнктивной форме (ДФ);
- 4) для определения входного набора для проверки $x_i=0$ ($x_i=1$) выбираем из полученной ДФ терм t , для которого $x_i t = 1$ ($\bar{x}_i t = 1$);

5) если выбранный терм содержит переменные, сопоставленные линиям обратной связи, то требуется обеспечить установку этих переменных в необходимые значения. Иными словами, требуется подобрать входной набор (возможно, для этого потребуются несколько специальных установочных наборов), устанавливающий соответствующие элементы памяти в необходимые состояния;

6) тест для проверки x_i составляется из установочных наборов и набора, определяемого по результатам выполнения п. 5.

4.6. Построение теста по методу эквивалентной нормальной формы

Построение модели объекта диагностирования

Этот метод основан на представлении булевой функции в виде **эквивалентной нормальной формы (ЭНФ)**. Функция в виде ЭНФ отличается от дизъюнктивной формы (ДФ) тем, что ее переменные сопоставлены не входам схемы, а всем возможным путям распространения сигналов.

Эквивалентная нормальная форма, как и обычная нормальная, может быть вычислена **методом подстановки** с той лишь разницей, что избыточные термы не исключаются, так как они характеризуют конкретную реализацию схемы.

Так для схемы рис. 4.9. ЭНФ может быть получена путем следующих подстановок (применяется процедура обратной подстановки: от выхода схемы к ее входам):

$$y = (\bar{m} \bar{n})_1;$$

$$y = ((\bar{a} \vee \bar{k})_2 (1)_3)_1;$$

$$y = ((\bar{a} \vee (\bar{b} \bar{c})_5)_2 (k \vee d)_4)_3)_1;$$

$$y = ((\bar{a} \vee (\bar{b} \bar{c})_5)_2 (((b \vee c)_5 \vee d)_4)_3)_1;$$

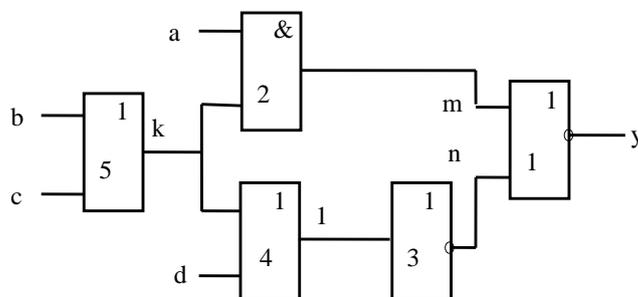


Рис. 4.9. Схема для иллюстрации метода ЭНФ

В ходе обратной подстановки последовательно определяются функции на выходных линиях элементов через переменные, сопоставленные их входным

линиям. Каждая скобка описывает функцию некоторого элемента. Индекс скобки соответствует номеру элемента, функцию которого описывает скобка.

Раскрыв скобки, получим

$$y = \overline{a_{21}} \overline{b_{5431}} \vee \overline{a_{21}} \overline{c_{5431}} \vee \overline{a_{21}} \overline{d_{431}} \vee \overline{b_{521}} \overline{c_{521}} \overline{b_{5431}} \vee \overline{b_{521}} \overline{c_{521}} \overline{c_{5431}} \vee \overline{b_{521}} \overline{c_{521}} \overline{d_{431}}.$$

Переменная и ее индексы в выражении определяют путь прохождения (т.е. список номеров элементов, через которые проходит сигнал) соответствующего входного сигнала до выхода схемы.

Закодируем последовательности элементов путей схемы следующим образом: 2,1-1; 5,2,1-2; 5,4,3,1-3; 4,3,1-4.

С учетом введенных обозначений:

$$y = \overline{a_1} \overline{b_3} \vee \overline{a_1} \overline{c_3} \vee \overline{a_1} \overline{d_4} \vee \overline{b_2} \overline{c_2} \overline{b_3} \vee \overline{b_2} \overline{c_2} \overline{c_3} \vee \overline{b_2} \overline{c_2} \overline{d_4}.$$

Аргументы (переменные) ЭНФ называют буквами. Число букв ЭНФ в общем случае больше числа входных переменных схемы, так как одна и та же входная переменная может быть связана с выходом схемы несколькими путями.

Наряду с ЭНФ можно использовать при построении тестов также обратную ЭНФ (ОЭНФ). Обратная ЭНФ для схемы рис. 4.8 имеет вид:

$$\overline{y} = a_1 b_2 \vee a_1 c_2 \vee b_3 c_3 d_4.$$

Методика построения теста

Идея построения теста неисправности по ЭНФ (или ОЭНФ) основана на следующем.

При неисправности константного типа функцию, реализуемую схемой, можно получить из исходной, зафиксировав некоторые буквы в значениях 0 и 1. Например, функцию схемы (рис. 4.9) при неисправности «линия m=1» можно получить из исходной ЭНФ, зафиксировав буквы a_1 и b_2 в значении 1, либо буквы a_1 и c_2 в значении 1. После подстановки этих значений в функцию исправной схемы получим функцию, реализуемую при заданной неисправности: $y = 0$.

Будем говорить, что на некотором наборе входных сигналов схемы проверяется (или обнаруживается) фиксация буквы $a_k=1$ (0), если на этом наборе значения выходного сигнала исправной схемы и с зафиксированной буквой a_k различны.

Вышеуказанная связь между неисправностью и ЭНФ дает возможность определить входной набор для проверки неисправности путем определения входного набора, проверяющего фиксацию букв ЭНФ этой неисправностью.

Поскольку ЭНФ представляет собой сумму логических произведений, она соответствует гипотетической схеме из нескольких элементов И, сигналы с которых поступают на элемент ИЛИ (рис. 4.10). Каждой схеме И соответствует один терм ЭНФ. Из такого представления ЭНФ становится очевидным, что для проверки неисправности, связанной с переменной (буквой) «а», входящей в какой-либо терм ЭНФ, необходимо выполнение следующих условий:

1) равенство нулю всех термов¹, кроме содержащего переменную "а" (применительно к рис. 4.10 для этого надо обеспечить равенство нулю хотя бы одной переменной на входах элементов D_2, \dots, D_n);

2) равенство единице всех компонент выбранного терма, в который входит переменная «а» (на рис. 4.10 это переменные x_1, \dots, x_k).

Выполнение вышеуказанных условий обеспечивает равенство $y=a$ и, как следствие, неисправности, изменяющие значение переменной «а», изменяют и значение выхода «у» схемы.

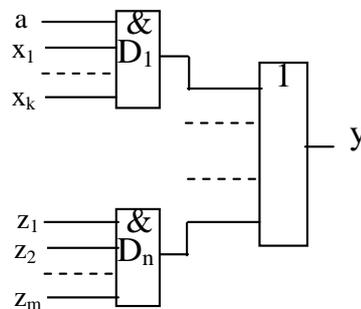


Рис. 4.10. Схема реализации ЭНФ

Тест, проверяющий фиксацию всех букв в значениях 0 и 1, является тестом, проверяющим все неисправности константного типа.

С учетом вышеизложенного, методику построения теста по методу ЭНФ можно представить в виде следующих операций:

1) пронумеровав линии схемы и элементы, строим ЭНФ;
2) выбираем букву ЭНФ и терм с этой буквой;
3) для проверки буквы² на неисправность типа 1(0) присваиваем ей значение 0(1). Всем остальным буквам выбранного терма присваиваем значения так, чтобы они давали значение 1 в выбранном терме.

4) в остальных термах хотя бы одной букве присваиваем такое значение, чтобы терм стал равным нулю. Если оказалось невозможным выполнить это условие для выбранного терма, то возвращаемся к п. 3 и берем другой терм с рассматриваемой буквой и т.д.;

5) определяем значения переменных ЭНФ, сопоставленных входам схемы. Они устанавливаются в соответствии с назначенными значениями букв и задают входной набор для проверки, выбранной в п. 2 буквы. При этом надо иметь в виду, что одноименным буквам с разными индексами соответствует одна и та же входная переменная (так, сигнал, поданный на вход «с» определяет значения как буквы c_2 , так и буквы c_3);

¹ Каждому терму в схеме 4.10 сопоставлен один из элементов И.

² Методика построения теста изложена с некоторыми упрощениями, так как теоретически возможны случаи, когда надо проверять фиксацию не отдельной буквы, а сразу нескольких.

б) повторяем п. 2–4 до тех пор, пока не будут рассмотрены все буквы ЭНФ и их фиксации.

Каждой букве сопоставлена группа линий и элементов схемы. Поэтому тестовый набор, проверяющий букву, фактически проверяет правильность прохождения сигнала через соответствующую группу линий и элементов, а следовательно проверяет некоторое множество неисправностей. Заметим, что возможны ситуации, когда для проверки неисправности необходимо искать набор, проверяющий фиксацию некоторой группы букв. Однако такие ситуации достаточно редки.

Построим тест для схемы рис. 4.9. При этом воспользуемся обратной ЭНФ, которая несколько проще.

Буква $a_1=1$ проверяется на наборе $a=0, b=1, c=1, d=1$; $a_1=0$ – на наборе $a=1, b=1, c=1, d=1$; буква $b_2=1$ проверяется на наборе $a=1, b=0, c=0, d=1$; и т.д. Результирующий тест, проверяющий фиксацию всех букв, включает наборы:

a	b	c	d
0	1	1	1
1	1	1	1
1	0	0	1
1	1	0	1
1	0	0	1
1	0	1	1
0	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Некоторые наборы одновременно проверяют несколько букв. Так, набор 0000 проверяет буквы $b_3=1, c_3=1, d_4=1$. Схема на рис 4.9 является комбинационной и ее поведение на конкретном воздействии не зависит от предыдущих воздействий. Поэтому повторяющиеся наборы можно исключить из теста, что и сделано в рассмотренном примере.

Развитие этой методики на схемы с элементами памяти, как и в случае метода с булевой производной, предполагает введение отдельных букв для состояний элементов памяти. Если выясняется, что для необходимой проверки требуется установка элементов памяти в конкретные состояния, то необходимо подобрать соответствующую установочную последовательность.

4.7. Использование модели конечного автомата для построения тестов

Существует ряд методов построения тестов по автоматной модели, когда объект диагностирования представлен в виде таблицы и графа переходов. Они удобны для реализации на ЭВМ. Однако надо иметь в виду, что построение графа и таблицы переходов затруднительно для схем с большим числом элементов памяти. Кроме того, в такой модели отсутствует информация о конкретной схемной реализации объекта диагностирования, что затрудняет оценку полноты

контроля получаемых тестов и использование модели для практического решения задач поиска неисправностей.

Рассмотрим один из таких методов применительно к автомату, таблица и граф переходов которого приведены на рис. 4.11.

Входные сигналы	Состояния			
	1	2	3	4
X_0	$3/y_0$	$3/y_0$	$3/y_1$	$1/y_1$
X_1	$2/y_0$	$4/y_0$	$4/y_1$	$2/y_1$

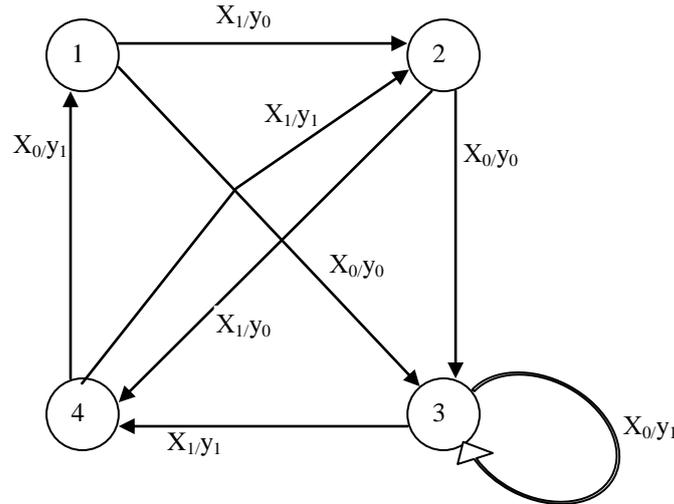


Рис. 4.11. Граф и таблица переходов автомата

По этому методу строится тест, обладающий тем свойством, что под его воздействием автомат должен «побывать» во всех состояниях и совершить все возможные переходы. Автомат на рис. 4.11 имеет четыре возможных состояния. Над дугой графа через символ «/» приведены **входное воздействие/выходной сигнал**. В таблице переходов через символ «/» дается **следующее состояние/выходной сигнал**.

Построение теста можно представить из следующих операций.

1. Определяем синхронизирующую последовательность, позволяющую установить автомат в некоторое начальное состояние Z_0 .

Последовательность $X_{\text{синх}}$ входных сигналов называется синхронизирующей, если при ее подаче автомат переходит в одно и то же состояние (например Z_0) независимо от того, в каком состоянии он находился до этого момента. Последовательность $X_{\text{синх}}$ легко определить по таблице переходов. Допустим (в нашем примере), в качестве Z_0 рассматривается состояние 3 (рис. 4.11). По таблице видно, что после подачи входного набора X_0 автомат

переходит в состояние 3, если он находился в одном из состояний 1, 2 или 3. Если же он был в состоянии 4, то после подачи X_0 окажется в состоянии 1. Так как последовательность $X_{\text{синх}}$ должна переводить автомат из любого состояния в конкретное состояние 3, то надо еще раз подать входной набор X_0 .

Таким образом, синхронизирующей последовательностью, устанавливающей автомат на рис. 4.11 в состояние 3, в данном примере является последовательность из двух входных наборов $X_0 X_0$. Заметим, если после подачи первого X_0 автомат уже находился в состоянии 3, то подача второго X_0 оставит автомат в состоянии 3 (это следует из графа переходов автомата).

Проверка любой электронной схемы начинается с подачи синхронизирующей последовательностью, что позволяет избежать неоднозначности в поведении объекта диагностирования. Стандартным решением на практике является введение специального входа «Сброс» («Reset»), при подаче сигнала на который все элементы памяти устанавливаются в некоторое определенное состояние.

2. Начинаем построение теста, для чего выбираем переход из состояния, в котором был установлен автомат синхронизирующей последовательностью, в некоторое другое состояние. Определяем входной набор, обеспечивающий выбранный переход и соответствующий выходной сигнал. Входной набор включаем в тест, а соответствующий выходной сигнал принимаем за эталонное (правильное) значение на данном наборе.

Например, если выбрать переход из 3 в состояние 2, то надо подать X_1 . При этом на выходе автомата появится сигнал y_0 .

3. Выбираем переход, переводящий автомат в новые состояния, и так до тех пор, пока автомат не пройдет через все переходы и состояния. Записываем и включаем в тест значения входных/выходных сигналов, соответствующих каждому из этих переходов.

4. Возможно, что из исходного состояния Z_0 нельзя выйти на некоторый переход. Тогда надо автомат перевести в любое состояние (возможно, это состояние автомат уже проходил), из которого возможен ранее не пройденный переход¹. Затем выбираем последовательность входных наборов, позволяющих пройти через ранее не пройденные переходы и состояния. Необходимые для таких переходов входные наборы включаем в тест. Повторяем п. 4, пока не будут пройдены все переходы и состояния.

Пользуясь вышеизложенным методом, построим тест для автомата на рис. 4.11.

После подачи $X_{\text{синх}} = X_0 X_0$ автомат будет находиться в состоянии 3. Начнем с того, что включим в тест входной набор (X_1), переводящий автомат в состояние 4. При этом на выходе появится сигнал y_0 . Для перехода в следующее (новое) состояние 1 надо подать X_0 , а для последующего перехода в состояние 2 – X_1 . Последовательность $X_1 X_0 X_1$ позволяет автомату совершить переходы $3 \rightarrow 4 \rightarrow 1 \rightarrow 2$ между всеми состояниями, но при этом остались переходы, которые не выполнялись. Например, судя по графу рис. 4.11, возможны также переходы

¹ В рассматриваемом примере пришлось еще дважды проходить через состояние 4, чтобы обеспечить проход по всем переходам графа.

непосредственно между состояниями 2–4, 2–3 и 1–3. После подбора входных наборов, позволяющих проверить все возможные переходы, построение теста заканчивается. Последовательность входных/выходных сигналов, обеспечивающих все возможные переходы, показана на рис. 4.12. Эта последовательность входных наборов вместе с синхронизирующей последовательностью $X_{\text{синх}} = X_0 X_0$ и представляет тест.

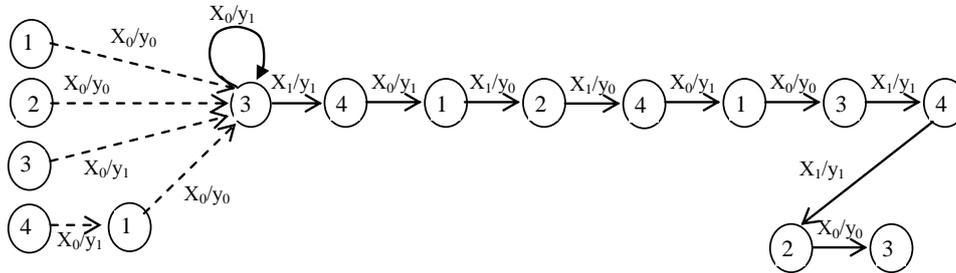


Рис. 4.12. Последовательность проверяемых переходов

Таким образом, в ходе тестирования на автомат надо будет предварительно подать синхронизирующую последовательность $X_{\text{синх}} = X_0 X_0$, а затем последовательность $X_1 X_0 X_1 X_1 X_0 X_0 X_1 X_1 X_0$. При правильной работе автомата при подаче синхронизирующей последовательности на его выходе должна быть последовательность $y_0 y_1$, если он находился в одном из состояний 1, 2 или 3. Если же автомат был в состоянии 3, то после подачи $X_{\text{синх}}$ на выходе появятся сигналы $y_1 y_1$. При подаче остальных входных наборов на выходе правильно функционирующего автомата должна быть последовательность сигналов $y_1 y_1 y_0 y_0 y_1 y_0 y_1 y_1 y_0$.

4.8. Построение тестов для микропроцессорных систем

Современная технология проектирования и производства электронных схем позволяет реализовать в одном кристалле миллионы транзисторов. В настоящее время на одном кристалле реализуются достаточно сложные микропроцессорные системы. Их тестирование представляет очень серьезную проблему.

Если микропроцессор можно представить **на вентильном уровне** (т.е. в виде электрической схемы, компонентами которой являются элементы И, ИЛИ, триггеры и т.п.), то, в принципе, для построения теста можно использовать рассмотренные выше методы. При этом в качестве модели неисправностей вполне возможно использовать константную модель. Однако подбор входных воздействий надо вести с учетом архитектуры и структуры команд, с которыми работает микропроцессор, а также учитывать особенности выполнения различных команд. Иными словами, надо понимать, что предлагаемые для включения в тест коды должны задавать команды и данные в форматах, принятых для данного микропроцессора.

Разработчики программно-управляемых систем используют готовые микросхемы (микропроцессоры, модули памяти, интерфейсные схемы и др.), которые все время усложняются. Многие создаваемые электронные изделия имеют такую сложность, что нереален какой-то их анализ в целом на вентиляльном уровне. К тому же зачастую разработчику печатной платы неизвестны вентиляльные схемы сложных микросхем. Поэтому часто построение теста ведется на основе информации об архитектуре системы команд, описаний структуры и способов адресации команд, протоколов, реализуемых в интерфейсных схемах.

Один из распространенных подходов предполагает представление программно-управляемого объекта диагностирования **в виде функциональной модели из отдельных механизмов** (см. п. 2.1). Тесты разрабатываются отдельно для каждого механизма (функции) с учетом его взаимодействия с другими механизмами. Выбор механизмов, представленных в модели, – неформальная задача. Например, могут выделяться такие механизмы, как дешифрация операций, адресации, межрегистрового обмена данными, выборки регистров и др. При разработке теста надо знать алгоритм работы механизма. Построение теста на основе моделей из механизмов (функций) требует хорошего понимания «внутреннего» функционирования объекта диагностирования.

В общем виде, процесс построения тестов по функциональной модели можно представить из следующих операций:

- построение графа функциональной модели;
- определение всех путей между входной (IN) и выходной (OUT) вершинами графа;
- построение для каждого пути подпрограммы, обеспечивающей продвижение входных данных до выхода по данному пути.

Полученная подпрограмма дополняется командами для считывания продвигаемых данных, сравнения полученных результатов с заранее подготовленными эталонными данными и отображения результатов.

Так, для графа модели рис. 2.8 возможно 9 разных путей от входной вершины IN до выходной вершины OUT (рис. 4.13). В граф-модели над дугами этих путей проставлены команды и условия, при которых происходит передача и преобразование данных по направлению дуги. Данная информация используется для формирования подпрограммы обработки и перемещения данных от вершины IN до вершины OUT. Например, для пути P_1 пересылку входных данных в регистр R_1 можно выполнить с помощью команды I_0 (рис. 2.8), а пересылку данных из регистра R_1 на выход – с помощью команды I_2 или I_7 (при условии нулевого значения в регистре R_4).

Еще один подход к разработке тестов микропроцессорных устройств исходит из того, что процесс функционирования объекта – это **процесс выполнения некоторой программы**, который предполагает последовательное выполнение отдельных команд. При этом подходе сначала выполняется анализ системы команд объекта. Далее для каждой команды микропроцессора объекта строится макрос, который представляет короткую последовательность команд, активизирующий эту команду и проверяющий правильность ее выполнения путем распространения на выходы одного или более результирующих машинных слов, полученных при выполнении команды (для визуализации результатов выполне-

ния макроса могут использоваться светодиоды, экран монитора, некоторое внешнее устройство отображения и т.п.). Макросы строятся для каждой команды и далее из них формируют итоговую тестовую программу (возможно, их будет несколько).

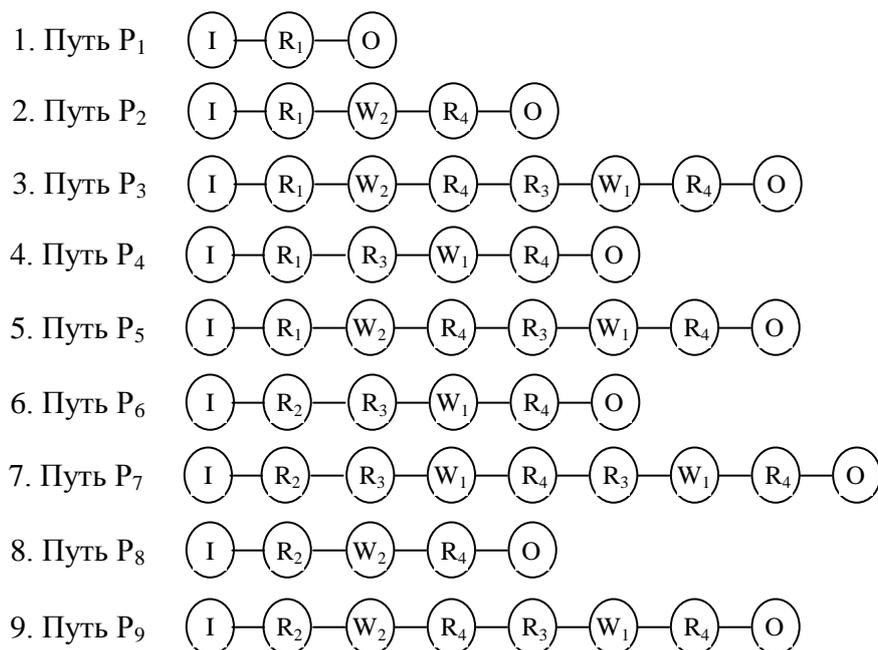


Рис. 4.13. Возможные пути в графе рис. 2.8 от вершины IN (I) до вершины OUT (O)

Современные СБИС, в частности микросхемы – микропроцессоры, микросхемы памяти, схемы различных интерфейсов, проектируются со встроенными схемами их контроля. При этом вводится несколько дополнительных внешних выводов микросхемы, через которые можно осуществлять управление процессами контроля и выводить результаты контроля для анализа. Существуют различные рекомендации и даже стандарты на проектирование микросхем со встроенными схемами контроля. Один из распространенных подходов к такому проектированию состоит в организации так называемого периферийного сканирования (более подробно о методе периферийного сканирования см. в разделе 7 настоящего учебного пособия). Тогда через стандартный порт компьютера или через специальное внешнее дополнительное устройство – эмулятор можно организовать проверку микросхемы, исключив влияние на проверку других компонент объекта диагностирования. Таким образом, наиболее сложные компоненты объекта диагностирования можно проверить отдельно, зачастую используя уже имеющиеся тестовые программы. В итоге снижается сложность построения тестов. Важно отметить, что производители микросхем со встроенными

механизмами периферийного сканирования обычно одновременно поставляют соответствующие тестовые программы для проверки.

В последние годы активно формируется особое направление в методах создания тестов, предназначенных для отработки вновь созданных микропроцессоров или программ¹. Тогда важно проверить работу процессора при выполнении произвольной последовательности команд и входных данных, даже не предусмотренных в нормальных рабочих режимах. Одна из конструктивных идей автоматизации этого направления состоит в том, что разработчик теста создает некоторую базовую тестовую программу, в которую вставляют специальный макрос (подпрограмму). Далее автоматически генерируются произвольные коды, которые вставляются в макрос как команды и данные. При запуске тестовой программы на исполнение процессор получает произвольно составленные команды, что может вызвать останов или зависание. Тогда определяется правильность реакции процессора и автоматически меняется (на новую) команда макроса, вызвавшая отказ в работе, и т.д.

В заключение отметим, что на практике обычно используются интерактивные средства автоматизации построения тестов программно-управляемых устройств, т.е. синтез теста идет в диалоге ЭВМ-человек. Это связано с тем, что до сих пор построение теста для микропроцессора и вообще для любого современного программно-управляемого устройства – это, скорее, искусство, чем алгоритм.

Для практических приложений зачастую берут готовые серийные печатные платы. При этом ее приспособление к конкретному приложению сводится к вводу программы пользователя в установленный на плате микропроцессор, установке на плате согласующих элементов, подключению платы к датчикам и отладке введенной программы. Тогда в качестве тестов используются некоторые программы, аналогичные тем, которые будут исполняться при реальной работе объекта, дополненные командами отображения выходных и промежуточных результатов работы. Иными словами, тогда для создания тестов надо просто знать основные режимы функционирования и алгоритмы, наиболее часто выполняемые микропроцессором данной платы. Возможность применения такого подхода к диагностированию обусловлена высокой надежностью современных микропроцессоров и электронных печатных плат в целом.

Контрольные вопросы

1. Дайте определения понятий контролирующего, диагностического и тупикового тестов.
2. Для таблицы неисправностей (табл. 4.6), используя методику, изложенную в пп. 4.1, получите дизъюнктивную форму, задающую все тупиковые тесты. Определите по ней тупиковый тест минимальной длины.
3. Используя методику упрощений (удаление поглощаемых строк и поглощающих столбцов), получите тупиковый тест по данным табл. 4.6.

¹ Вопросы тестирования программ обсуждаются в теме 8 настоящего учебного пособия.

Таблица 4.6

Наборы	Номера неисправностей					
	1	2	3	4	5	6
a	1	1	0	0	0	0
b	0	0	1	0	1	0
c	1	0	0	1	0	0
d	1	0	1	0	0	1
e	0	1	0	0	0	0
f	0	1	0	0	1	0

4. Что значит «активизировать путь»? Определите назначение этой операции при построении теста по методу активизации чувствительного пути.
5. Определите по методу активизации чувствительного пути тестовый набор для проверки неисправности «константа 0» на выходе элемента 4 схемы рис. 4.9.
6. Что представляет собой d-куб и какие значения могут принимать его разряды (координаты)?
7. Перечислите сингулярные кубы и d-кубы элемента И с тремя входами.
8. Составьте d-кубы неисправностей для элемента И с тремя входами.
9. В чем сущность обратной фазы d-алгоритма?
10. Запишите общую формулу для булевой производной функции F по переменной x_i .
11. Определите булеву производную функции $F=x_1x_2 \vee x_3x_4$ по переменной x_3 .
12. Какое условие должно выполняться при получении тестового набора для проверки фиксаций переменной x_i по методу булевой производной?
13. Получите ЭНФ для схемы на рис. 4.7.
14. Постройте тест, проверяющий все буквы ЭНФ схемы на рис. 4.6, ассоциируемые с путями, исходящими от входа x_1 .
15. Какое назначение синхронизирующей последовательности?
16. Постройте тест, проверяющий все переходы автомата на рис. 4.11, если отсутствует переход из состояния 1 в состояние 2.
17. Правила построения графа функциональной модели микропроцессорного устройства.

Рекомендуемая литература

- Карибский В.В. Основы технической диагностики / В.В. Карибский, П.П. Пархоменко, Е.С. Согомонян, В.Ф. Халчев. Кн. 1. – М.: Энергия, 1976. – 464 с.
- Коган, Б.М. Основы эксплуатации ЭВМ / Б.М. Коган, И.Б. Мкртумян. – М.: Энергоатомиздат, 1988. – 432 с.
- Мальшенко, Ю.В. Автоматизация диагностирования электронных устройств / Ю.В. Мальшенко, В.П. Чипулис, С.Г. Шаршунов – М.: Энергоатомиздат, 1986. – 216 с.
- Мальшенко, Ю.В. Техническая диагностика / Ю.В. Мальшенко. Ч. 1. – Владивосток: Изд-во ВГУЭС, 1998. – 68 с.
- Чипулис, В.П. Анализ и построение тестов цифровых программно-управляемых устройств / В.П. Чипулис, С.Г. Шаршунов. – М.: Энергоатомиздат, 1992. – 224 с.

Тема 5. ВНУТРИСХЕМНОЕ (ПОЭЛЕМЕНТНОЕ) ДИАГНОСТИРОВАНИЕ

5.1. Назначение и принципы внутрисхемного диагностирования

Понятие и назначение

Термин **внутрисхемное тестирование** (ICT) означает проверку отдельных компонентов непосредственно на плате.

В основе методов и систем поэлементного контроля лежит идея о том, что объект исправен, если параметры всех его элементов находятся в заданных допусках и линии связи не имеют обрывов и коротких замыканий.

В последние годы активно развивается идеология проектирования и эксплуатации электронных изделий с развитыми средствами встроенного диагностирования, позволяющая автоматически определить неисправный узел. При этом конструкция изделия позволяет легко заменить неисправный узел на исправный. Если же этот узел имеет небольшую стоимость, то его можно «выбросить» и не заниматься восстановлением. Однако до сих пор большое число электронных устройств подлежит восстановлению (по крайней мере на этапе производства), и для уточнения фактически неисправных элементов используются методы и средства внутрисхемного диагностирования.

В системах поэлементного диагностирования поочередно проверяются все элементы и линии связи. Причем при тестировании компонента необходимо исключить влияние параллельных цепей. Например, при проверке резистора надо определить именно его сопротивление, а не сопротивление цепи, к которой он подключен.

Внутрисхемное тестирование выполняется после установки компонентов на плату и давно применяется в ходе серийного производства электронных изделий в качестве надежного метода проверки правильности установки элементов на печатной плате и оценки параметров установленных элементов. С помощью методов и средств внутрисхемного диагностирования можно локализовать неисправности при ремонте электронных модулей, не имея их принципиальных схем (например, часто отсутствуют схемы на импортное оборудование).

Считается, что использование этих методов позволяет обнаружить до 80% всех дефектов сборки.

Классификация видов внутрисхемного диагностирования

Внутрисхемное диагностирование, в свою очередь, подразделяется на аналоговое и цифровое.

При аналоговом внутрисхемном диагностировании обычно проверяют:

- наличие коротких замыканий и обрывов, в том числе в результате некачественной пайки;
- фактические значения дискретных компонентов (резисторов, конденсаторов, индуктивностей, дискретных полупроводниковых приборов) и их отклонение от номиналов;
- правильность установки элементов (например, ориентация диодов);
- правильность установки и функционирование микросхем.

При цифровом внутрисхемном диагностировании проверяются цифровые микросхемы на соответствие таблице истинности. Для исключения влияния других микросхем, подключенных к проверяемой, на вход тестируемой микросхемы могут подаваться импульсы большого тока с ограниченной длительностью. Пусть, например, при включении питания на вход проверяемой микросхемы от предшествующих микросхем поступает сигнал «0», а нам надо иметь значение «1». Подав в данную точку импульс большого тока, можно кратковременно «подавить» значение «0». Важно, чтобы его длительность не превышала время, которое выдерживает подобную перегрузку данный тип микросхем. Такой метод диагностирования называется «**backdriving**».

Выбор оборудования и метода доступа к проверяемому изделию зависит от конструктивных особенностей изделия и его элементов, условий и целей диагностирования.

С точки зрения степени автоматизации доступа к точкам измерений в системах внутрисхемного диагностирования выделяют следующие методы.

Метод клипс и пробников¹ предполагает, что оператор вручную с помощью специальных контактных зондов подсоединяет измерительное оборудование к некоторым точкам на печатной плате. Он универсален и недорог, но требует больших временных затрат и сравнительно высокую квалификацию персонала, поэтому его обычно применяют при единичном производстве и ремонте.

Метод «поле контактов» или «**ложе из гвоздей**» обычно предполагает изготовление специального тестового адаптера, содержащего множество контактных игл. На адаптер накладывается печатная плата, а через контактные иглы производится подключение измерительного оборудования к контактным площадкам на плате. Такой метод обеспечивает очень высокую производительность. Его применяют при средне- и крупносерийном производстве. При этом адаптер делается отдельно для каждого диагностируемого изделия.

Метод подвижных пробников является безадаптерной альтернативой доступа к диагностируемому изделию. Оборудование с подвижными пробниками имеет несколько тестовых головок с приводами по осям X, Y и Z. Тестовые головки (обычно от четырех до восьми) перемещаются над испытываемой платой с большой скоро-

¹ Более подробно методы внутрисхемного диагностирования и соответствующее оборудование для их реализации рассматриваются в последующих подразделах.

стью до точек, к которым надо подключить измерительный узел. Электрические зонды, расположенные на каждой головке, вступают в контакт с соответствующими контактными площадками на печатной плате. Таким образом обеспечивается доступ к контрольным точкам, в которых надо сделать измерения.

Вместе с тем данный метод не обеспечивает особо высокой производительности при тестировании. Однако он достаточно эффективен для предприятий с небольшой программой выпуска, но широкой номенклатурой изделий.

5.2. Принципы измерений параметров элементов с помощью вольтметров и амперметров

Схемы подключения приборов

Оценка параметров элементов при внутрисхемном диагностировании предполагает измерение напряжений и/или токов, протекающих через элементы с последующим вычислением необходимого параметра на основе закона Ома.

Простейшими приборами для измерения напряжений и токов являются аналоговые вольтметр и амперметр соответственно. В быту и в мастерских по ремонту радио и телеаппаратуры для ручного поиска неисправностей используют также универсальные приборы – мультиметры, которые в зависимости от выбранного режима измерения могут измерять несколько параметров: напряжение, ток, сопротивление, емкость, частоту и др.

Для измерения падения напряжения U_x на элементе (рис. 5.1а) вольтметр V надо подключить параллельно контролируемому элементу R_x . При этом часть общего тока I_0 цепи отводится на вольтметр (ток I_v), в результате чего уменьшается ток и соответственно падение напряжения на контролируемом элементе. Чтобы измеренное значение меньше отличалось от действительного (т.е. меньше была погрешность измерения), надо использовать вольтметр как можно с большим сопротивлением.

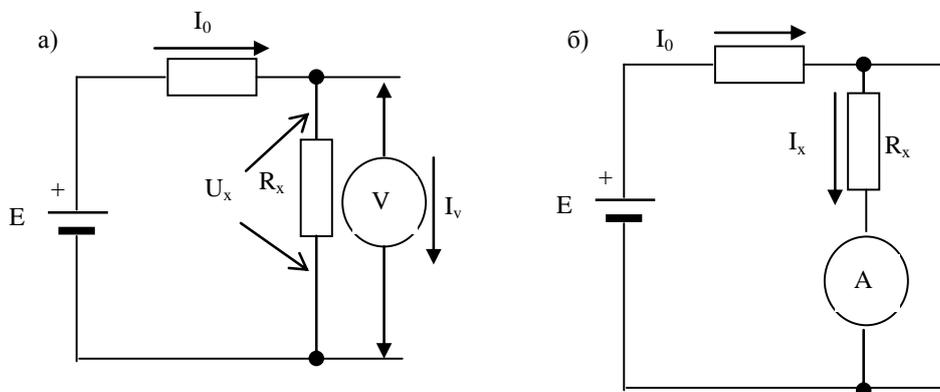


Рис. 5.1. Схемы подключения вольтметра и амперметра

Для измерения тока, проходящего через контролируемый элемент R_x , измерительный прибор (амперметр) надо включить последовательно с R_x (рис. 5.1б). Амперметр измеряет ток, непосредственно проходящий через контролируемый элемент. Однако надо иметь в виду, что амперметр сам обладает некоторым сопротивлением, что сказывается на величине тока. Чтобы измеренное значение меньше отличалось от действительного, надо использовать амперметр как можно с меньшим сопротивлением.

О работоспособности элемента зачастую можно судить по величине сопротивления, которое он оказывает проходящему через него току. При этом измерения могут проводиться на постоянном или переменном токе.

Измерения на постоянном токе позволяют определить значения сопротивлений постоянных резисторов, определить токи утечки конденсаторов и полупроводниковых переходов, обрывы в линиях связи, короткие замыкания в элементах разных типов (резисторы, диоды, конденсаторы, трансформаторы и др.).

В частности, для измерений сопротивлений на постоянном токе в бытовых условиях широко применяют простые и компактные, так называемые аналоговые омметры (рис. 5.2). Они имеют очень небольшое энергопотребление и могут длительное время работать от обычной батарейки.

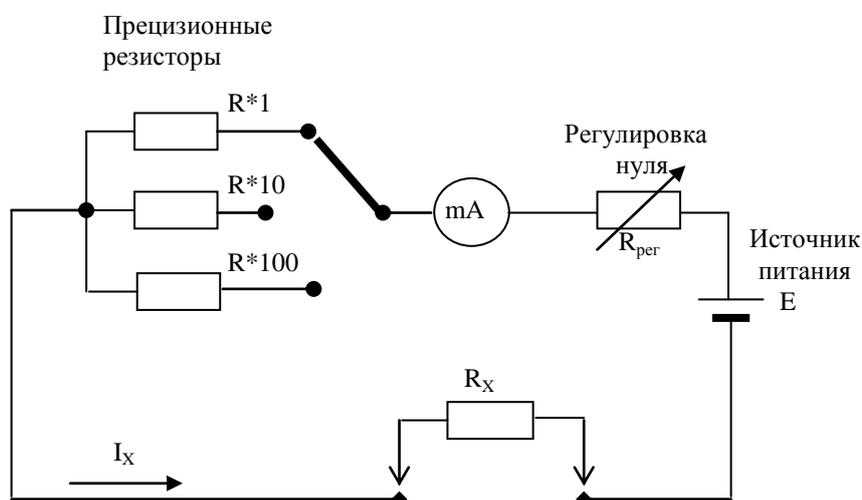


Рис. 5.2. Функциональная схема аналогового омметра

В аналоговых омметрах измеряемый резистор R_x включается последовательно со стрелочным измерителем тока (mA) и прецизионным резистором ($R*1$, $R*10$ или $R*100$), а также дополнительным регулировочным резистором. Величина прецизионного резистора выбирается переключателем в зависимости от предполагаемого значения R_x . Подключение к измеряемому элементу производится с помощью щупов. Перед подключением производится калибровка, для чего щупы замыкают друг на друга. При этом с помощью регулировочного резистора на стрелочном измерителе (mA) устанавливают значение «0». Только

затем щупами подключаются к R_x . По величине тока определяется сопротивление резистора R_x . Величина тока определяется выражением

$$I_x = E / (R_x + R^* + R_{\text{рег}}),$$

где R^* и $R_{\text{рег}}$ – значения прецизионного и регулировочного резисторов соответственно. Измерительная шкала имеет нелинейную градуировку, что является следствием нелинейной зависимости I_x от R_x .

Измерения на переменном токе

Измерения на переменном токе используются для оценки параметров разных катушек, конденсаторов, трансформаторов, усилителей и др. В этом случае на проверяемый элемент подается переменное напряжение (ток) обычно синусоидальной формы либо в виде прямоугольных импульсов.

Переменный периодический сигнал характеризуется периодом T (время одного цикла изменения сигнала), амплитудой A и частотой $f=1/T$. Если среднее значение амплитуды A не равно нулю, то говорят, что сигнал имеет постоянную составляющую $U_{\text{п}}$ (рис. 5.3). Для схем, работающих с переменными сигналами и содержащих усилители, постоянная составляющая напряжения является важным диагностическим параметром. Чтобы измерить только постоянную или только переменную составляющую переменного напряжения, вольтметр подключают к контрольным точкам с помощью специальных пробников-фильтров (рис. 5.4).

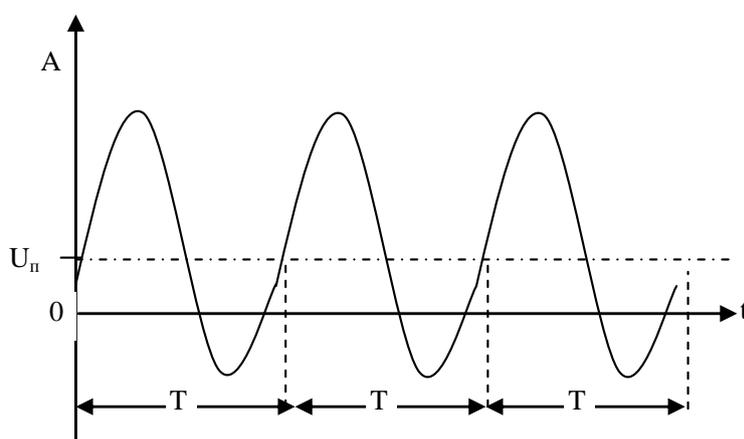


Рис. 5.3. Характеристики периодического сигнала

Пробник на рис. 5.4а используется для измерения постоянной составляющей переменного напряжения¹. Поступающее на пробник напряжение попадает на LC-фильтр. С некоторым упрощением его работу можно объяснить следующим образом.

¹ На рис. 5.4 приведены примеры простейших фильтров.

Поступающий ток сначала проходит через индуктивность L , сопротивление которой имеет значительную реактивную составляющую и небольшую активную (часто выполняется в виде небольшой катушки из медного провода). Она оказывает повышенное сопротивление переменной составляющей тока, но очень небольшое – постоянной. Далее ток попадает на конденсатор C . Он не пропускает постоянную составляющую тока, но хорошо пропускает переменную, которая через него попадает на земляной вывод. В результате на вход измерительного прибора проходит только постоянная составляющая измеряемого тока. Если фильтр не полностью устраняет переменную составляющую, то можно соединить несколько подобных фильтров.

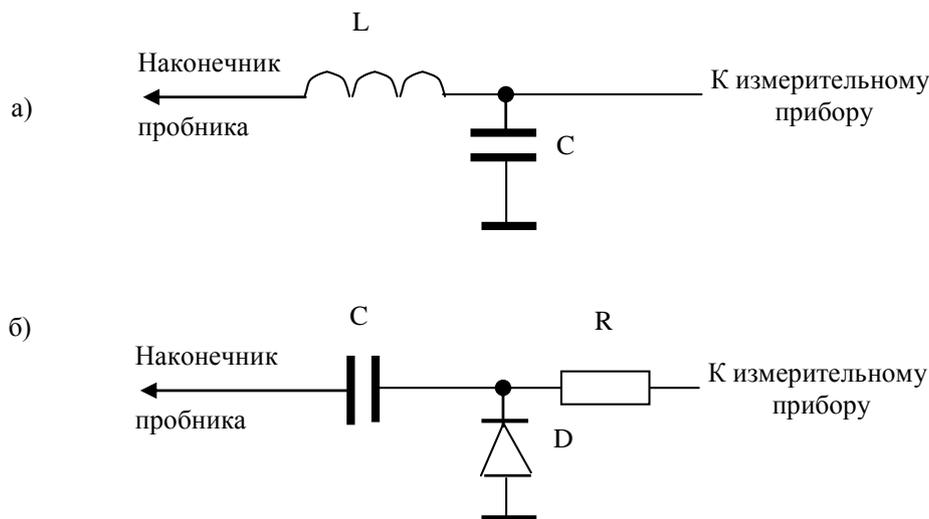


Рис. 5.4. Варианты пробников

Пробник на рис. 5.4б служит для выделения положительных значений высокочастотной составляющей. Конденсатор C не пропускает постоянную составляющую входного напряжения. Переменная составляющая проходит через конденсатор и попадает на диод D . Для отрицательных значений переменного напряжения сопротивление диода близко к нулю и весь входной ток идет через диод и, наоборот, при положительных – практически весь ток направляется через резистор R , так как сопротивление диода в обратном направлении очень велико.

Существуют и более «экзотические» пробники, которые эффективно используются при регулировке и наладке бытовой радиоаппаратуры. Один из таких пробников представлен на рис. 5.5а. По сути, это фильтр, настроенный на подавление некоторой частоты (вид его амплитудно-частотной характеристики дан на рис. 5.5б).

Подавляемая частота определяется по формуле $f_n = 1/2 \pi R_1 C_1$. При этом должны выполняться соотношения: $R_1 = R_2 = 2R_0$, $C_1 = C_2 = 1/2 C_0$, $R_1 = 1/2 \pi f_n C_1$. Данный фильтр может применяться для выявления, например, искажений в уси-

лителе низкой частоты. Пусть фильтр настроен на подавление частоты $f_{\Pi}=1000$ Гц, и на вход проверяемого усилителя низкой частоты подается такая же частота. Подключим фильтр к выходу усилителя. Если усилитель не вносит искажений, то амплитуда сигнала на выходе фильтра будет близка к нулю.

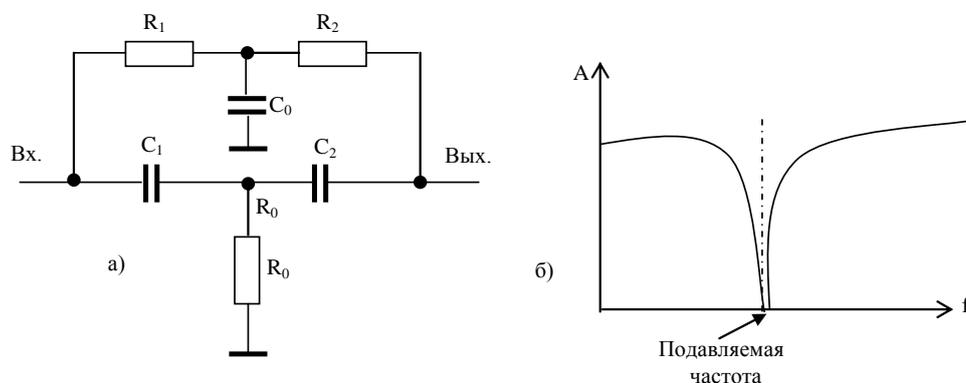


Рис. 5.5. Фильтр для подавления частот

Современные варианты омметров и мультиметров представляют собой цифровые измерительные приборы. Они отображают результаты измерений в цифровой и/или графической форме, имеют существенно более высокую точность измерений, могут подключаться к ЭВМ и т.п. Повышенная точность достигается, в том числе, за счет использования иных методов измерений. Так, для подачи тока на измеряемый элемент используют специальный источник постоянного тока (ИПТ), который автоматически поддерживает фиксированный ток I_0 в измерительной цепи независимо от величины R_x (рис. 5.6). В такой схеме падение напряжения на резисторе R_x пропорционально его величине.

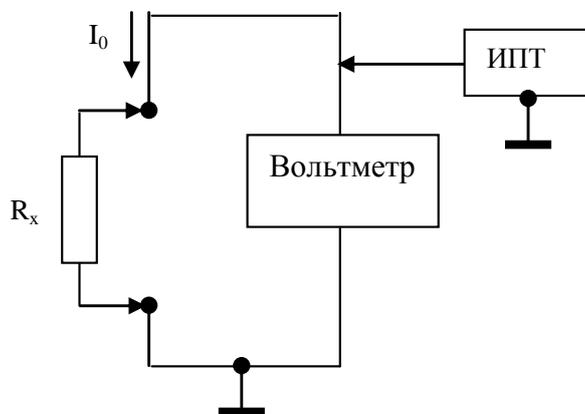


Рис. 5.6. Структурная схема цифрового омметра

5.3. Особенности определения работоспособности различных элементов с помощью простейших приборов

Хотя разработчики электронных схем постепенно переходят на использование микросхем все более высокой интеграции, но до сих пор много схем, в которых используются так называемые дискретные элементы: резисторы, конденсаторы, катушки индуктивности, трансформаторы, диоды, транзисторы и др.

Их отказы обычно имеют катастрофический характер (очень часто происходят обрывы или короткие замыкания) и могут быть выявлены, например, простейшим измерительным прибором – омметром. При этом проверку дискретных элементов осуществляют в следующей последовательности:

- внешний осмотр (при осмотре следует обращать внимание на целостность корпуса, отсутствие на поверхности элемента пятен, трещин или вздутий, изменение цвета печатной платы под элементом или в месте пайки выводов);

- проверка надежности пайки выводов (очень часто вследствие окисления или нагрева нарушается прочность пайки, что можно проверить, подергав вывод элемента);

- измерение параметров элемента (измерений может быть несколько: при прямом и обратном включении элемента, между разными выводами, при создании перемычек, при включенном или отключенном питающем напряжении и т.д.).

Практика показывает, что предварительный осмотр и проверка качества паяк зачастую позволяет сразу указать элементы, которые надо проверить на неисправность.

Проверка резисторов

Собственно резисторы делят на проволочные и непроволочные, постоянные и переменные. Основные технические параметры резисторов следующие:

- номинальная величина омического сопротивления;
- номинальная мощность рассеивания;
- допустимое отклонение сопротивления от номинальной величины;
- температурный коэффициент (характеризует изменение сопротивления при изменении температуры на 1C°);
- предельное рабочее напряжение и др.

Маркировка на корпусе резисторов обычно указывает три первых из вышеперечисленных параметров.

Реальная величина сопротивления резистора может отличаться от номинальной в пределах его класса точности.

Мощность, которую может рассеивать резистор при длительной непрерывной работе в неподвижном воздухе при температуре $+20\text{C}$ без повреждения проводящего элемента, называется номинальной и измеряется в ваттах. Непроволочные резисторы изготавливаются на номинальную мощность от десятых долей до десятков ватт.

Наиболее часто встречающаяся неисправность резистора – перегорание токопроводящего слоя. Может быть вызвано прохождением через резистор недопустимо большого тока, в результате чего возникает обрыв (тогда рассеиваемая мощность становится выше номинальной), но иногда может иметь место и короткое замыкание из-за расплавления и создания из расплавленного материала перемычки. Реже, но возможно, частичное разрушение резистивного слоя, что приводит к резкому изменению сопротивления в сторону уменьшения или увеличения. В результате старения или при длительном нагреве величина сопротивления резистора может измениться на несколько десятков процентов.

Проволочные резисторы довольно редко выходят из строя. Основные их неисправности – обрыв, что происходит вследствие перегорания проволоки.

При осмотре неисправного резистора можно обнаружить обуглившиеся поверхности лакового или эмалевого покрытия, а в ряде случаев изменение цвета, иногда проявляющееся в виде колечек. Однако возможно, что обрыв токопроводящего элемента не вызывает изменений внешнего вида резистора.

Перед измерением сопротивления резисторов в схеме аппаратуры следует отключить питание и разрядить конденсаторы большой емкости (замкнуть коротко). Проверять сопротивление резисторов, входящих в состав сложных цепей, где возможно дополнительное прохождение тока через другие резисторы, электролитические конденсаторы или полупроводниковые приборы, желательно отпаять предварительно один конец испытываемого резистора от схемы.

Если измерение сопротивления резистора осуществляется без выпаивания из схемы, необходимо учитывать влияние шунтирующих цепей.

Резистор исправен, если его значение в пределах «номинальное значение \pm допустимое отклонение» с учетом класса точности.

Переменный резистор имеет три вывода. При проверке исправности переменных резисторов один щуп омметра подключают к среднему выводу резистора, а другой – к одному из крайних. Ось переменного резистора при этом вращают очень медленно. Если резистор исправен, то при вращении его оси стрелка (или показания цифрового индикатора) омметра будет плавно отклоняться. Измеряемое значение должно при этом изменяться от нуля до номинального значения. Дрожание, рывки свидетельствуют о плохом контакте щетки среднего вывода с токопроводящим элементом. Если стрелка омметра вообще не отклоняется, то резистор неисправен. Проверку рекомендуется повторить, переключив другой щуп омметра ко второму крайнему лепестку резистора.

Проверка конденсаторов

К электрическим неисправностям конденсаторов относятся: пробой, короткое замыкание пластин, изменение номинальной емкости сверх допуска (обычно это происходит из-за старения диэлектрика, попадания на него влаги, перегрева, деформации), повышение тока утечки из-за ухудшения изоляции. Для старых электролитических конденсаторов характерна частичная потеря емкости, что происходит в результате высыхания электролита.

Наиболее часто имеют место неисправности электролитических конденсаторов. В низковольтной радиоэлектронной аппаратуре керамические, пленочные и другие

типы конденсаторов крайне редко выходят из строя. Большой частью причиной их неисправности бывают механические повреждения: малозаметный перелом вывода около корпуса конденсатора, трещина в керамике и т.п. Поэтому при ремонте электронных блоков не следует без особой необходимости отгибать прижатые к плате керамические конденсаторы и лишний раз перегибать выводы.

Если при внешнем осмотре дефекты не обнаружены, проводят электрическую проверку. Если прибор позволяет измерять емкости, то достаточно измерить величину емкости и сравнить ее с номинальной. Рекомендуется перед проверкой отпаять испытываемый конденсатор (или хотя бы один из его выводов) от схемы.

Работоспособность конденсатора большой емкости можно проверить с помощью простейшего стрелочного тестера-омметра.

В зависимости от номинальной емкости проверяемого конденсатора выбирается предел измерения омметра. Для конденсаторов емкостью порядка сотен тысяч микрофард подходящими пределами являются « $\times 10$ » или « $\times 100$ » Ом. Конденсаторы емкостью единиц-десятков мкФ проверяются на пределах « $\times 1$ » и « $\times 10$ » кОм.

Стрелка прибора после небольшого толчка (при заряде конденсатора) должна вновь вернуться в исходное положение и показать очень большое сопротивление (сотни кОм). В противном случае конденсатор имеет утечку и подлежит обязательной замене. При проверке электролитических конденсаторов необходимо следить за тем, чтобы «плюс» омметра был подключен к положительному выводу конденсатора. С помощью нескольких заведомо исправных конденсаторов, замечая предельное отклонение стрелки, можно приблизительно отградуировать прибор для ориентировочного измерения емкостей. Тогда по величине первоначального отклонения стрелки можно примерно определить величину емкости. Следует отметить, что проверка конденсаторов при помощи одного только омметра не всегда достаточна.

Стрелка прибора не должна отклоняться при подключении обоих щупов омметра к выводам исправного конденсатора небольшой емкости.

Конденсаторы средней емкости (от 500 пФ до 1 мкФ) радиолюбители часто проверяют с помощью последовательно подключенных к выводам конденсатора телефонов и источника тока. При исправном конденсаторе C_x в момент замыкания цепи в телефонах прослушивается кратковременный щелчок (рис. 5.7а).

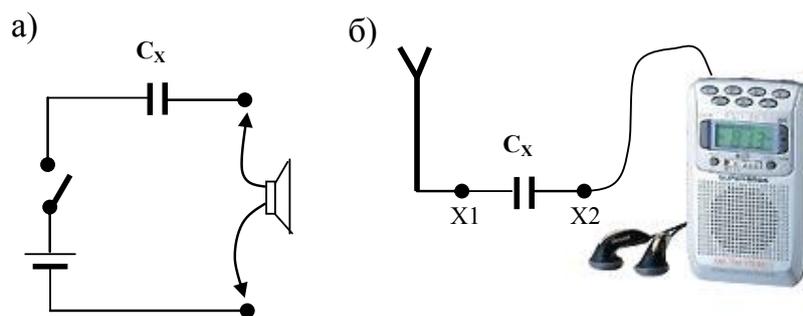


Рис. 5.7. Простые способы проверки конденсаторов малой емкости

Конденсаторы малой емкости (до 500 пФ) можно проверить в цепи тока высокой частоты. Например, конденсатор включают между антенной и приемником к точкам X1 и X2 (рис. 5.7б). Если громкость приема не уменьшится, значит, обрыва нет.

Проверка конденсатора в схеме без выпаивания должна производиться с обязательным учетом шунтирующих его элементов. Если параллельно проверяемому конденсатору включен резистор с небольшим сопротивлением, результат измерения емкости окажется недостоверным – зарядный бросок может быть мал или вообще отсутствует.

Особую осторожность следует соблюдать при проверке конденсаторов в цепях, связанных с выводами микросхем. При большом напряжении батареи омметра или большом токе можно повредить микросхему.

В радиолюбительской практике широко применяются простейшие приборы, в которых проверяемый конденсатор включается в контур генератора. На рисунке 5.8 показана схема такого прибора для проверки конденсаторов в пределах от 0,1 до 1000 мкф.

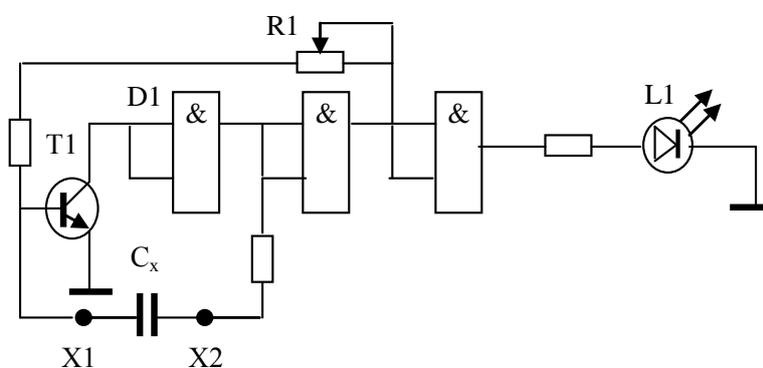


Рис. 5.8. Схема прибора для проверки конденсаторов

Проверяемый конденсатор C_x подключают к зажимам X1 и X2 (с соблюдением полярности) в цепь между входом транзистора T1 и выходом логического элемента И (D1). Так как коллектор транзистора подключен ко входу этого элемента, то через конденсатор C_x образуется положительная обратная связь. В результате элемент D1 начинает работать как генератор, частота которого зависит от величины C_x . Индикатором является светодиод L1.

Если проверяемый конденсатор C_x «пробит», т.е. замкнуты его пластины, то светодиод гаснет. В случае неисправности типа «обрыв» светодиод горит постоянно. С исправным конденсатором светодиод периодически вспыхивает. Частоту вспышек в некоторых пределах можно изменять переменным резистором R1.

Проверка катушек индуктивности

Проверка исправности катушек индуктивности начинается с внешнего осмотра, в ходе которого убеждаются в исправности каркаса, экрана, выводов;

в правильности и надежности соединения всех деталей катушки между собой; в отсутствии видимых обрывов проводов, замыканий, повреждений изоляции и покрытий. Особое внимание следует обращать на места обугливания изоляции, места соединения проводов с выводами, почернение или оплавление заливки.

Электрическая проверка катушек индуктивности включает проверку на обрыв, обнаружение короткозамкнутых витков и определение состояния изоляции обмотки.

Проверка на обрыв выполняется с помощью омметра. В большинстве случаев индуктивность имеет небольшое сопротивление по постоянному току. Поэтому перед измерением омметр устанавливают на самый низкоомный диапазон. Большое сопротивление означает обрыв или плохой контакт одной или нескольких жил провода индуктивности. Малое сопротивление означает наличие межвиткового замыкания. При коротком замыкании выводов сопротивление равно нулю. Для более точной оценки состояния катушки индуктивности необходимо измерить ее индуктивность.

Проверка трансформатора

Очень часто отказы возникают в блоках питания. При этом причиной отказа могут быть неисправности трансформаторов и силовых дросселей. По конструкции и технологии изготовления силовые трансформаторы и дроссели низкой частоты (НЧ) имеют много общего. Те и другие состоят из сердечника и обмоток, выполненных изолированным проводом.

Неисправности трансформаторов и дросселей НЧ делятся на механические и электрические. К механическим неисправностям относятся: поломки экрана, сердечника, выводов, каркаса и крепежной арматуры, к электрическим – обрывы обмоток, замыкания между витками одной или нескольких обмоток; короткое замыкание обмотки на сердечник или экран.

Проверку исправности трансформаторов и дросселей НЧ начинают с внешнего осмотра. В ходе его выявляют и устраняют все видимые механические дефекты.

Для проверки замыкания на корпус (пластины) надо проверить сопротивление между корпусом и выводом обмотки (рис. 5.9а). Если коротких замыканий нет, то омметр должен показывать сопротивление в десятки-сотни Ом.

Проверка на короткое замыкание между двумя обмотками производится путем подключения омметра Ω между любыми выводами этих обмоток (рис. 5.9б). Если коротких замыканий нет, то омметр должен показывать очень высокое сопротивление. Сопротивление обмотки проверяют, подключив омметр к выводам обмотки (рис. 5.9в). Если сопротивление равно бесконечности, то имеет место обрыв. Обмотка может иметь несколько средних отводов. Тогда надо измерить сопротивления между одним из крайних выводов и каждым отводом. Сопротивление между точками измерений при этом должно быть пропорционально числу витков обмотки между выводами, к ко-

торым подключается омметр (так как вся обмотка обычно выполняется одним и тем же проводом).

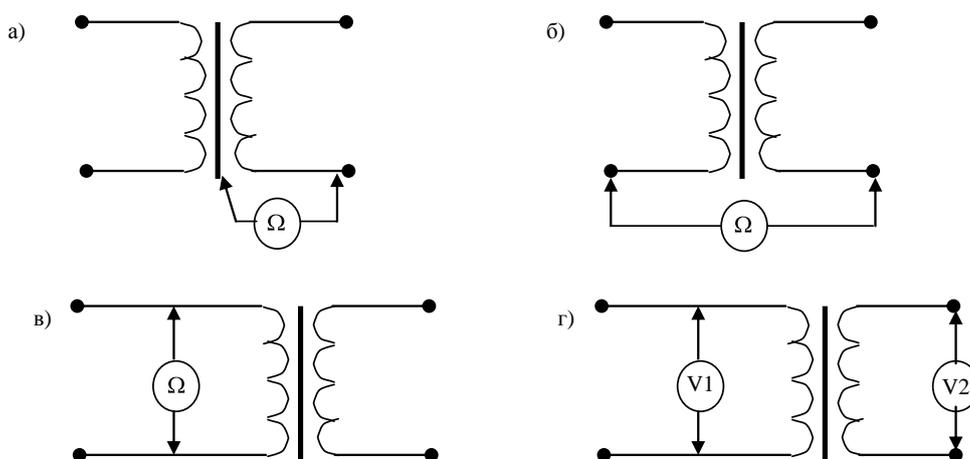


Рис. 5.9. Схемы измерений при проверке трансформаторов

Сложнее обнаружить наличие короткозамкнутых витков. Обычно для этого измеряют (на самом низкоомном пределе) сопротивление обмотки и сравнивают результаты с паспортными данными. Способ простой, но неточный, особенно при малой величине сопротивления обмоток и малом числе короткозамкнутых витков. Другой способ состоит в проверке коэффициентов трансформации на холостом ходу (рис. 5.9г). Тогда выводы вторичных обмоток надо отсоединить от нагрузки. Обмотки, подключаемые к питающей электрической сети, называют первичными (W_1), остальные – вторичными. Коэффициент трансформации определяется как $k=W_2/W_1$, где W_i – количество витков i -й обмотки. Этот коэффициент определяет и отношение напряжений между первичной и вторичной обмотками. Поэтому k должно быть равно отношению напряжений, показываемых вольтметрами V_2 и V_1 (рис. 5.9г). При наличии межвитковых замыканий коэффициент трансформации будет меньше паспортного значения.

У силовых трансформаторов одним из признаков наличия короткозамкнутых витков является чрезмерный нагрев обмотки, в том числе на холостом ходу. Существуют специальные приборы для обнаружения наличия короткозамкнутых витков.

Проверка диодов

Диод – двухэлектродный электровакуумный, полупроводниковый или газоразрядный прибор с резко выраженной односторонней проводимостью электрического тока. Диоды применяются в электро- и радиоаппаратуре для выпрямления и стабилизации переменного тока, детектирования, переключения электрических цепей.

Диод хорошо пропускает через себя ток в одном направлении и очень плохо – в другом.

В подавляющем большинстве современных электронных устройств применяются полупроводниковые диоды. Их действие основано на использовании свойств электронно-дырочного перехода (р-n-перехода), который образуется на стыке двух специально подобранных веществ р и n (рис. 5.10).

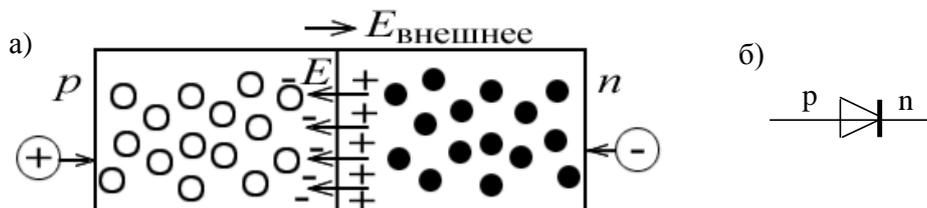


Рис. 5.10. Полупроводниковый диод: а) принцип работы, б) изображение на электрических схемах

Если к р-n-переходу диода приложить напряжение $E_{\text{внешнее}}$ в прямом направлении, т.е. подать на его р-область положительный потенциал, а на n-отрицательный, то потенциальный барьер E , соответствующий переходу, понижается и сопротивление току снижается – течёт большой прямой ток. Если приложить напряжение в обратном направлении (обратное смещение), то потенциальный барьер перехода, наоборот, повышается (сопротивление току увеличивается) и через р-n-переход протекает очень малый ток.

Указанные свойства, отражает **вольтамперная характеристика** (рис. 5.11), показывающая зависимость величины протекающего через диод тока I от величины и полярности приложенного к его выводам внешнего напряжения U . Слева от оси I на рисунке показано изменение тока (обратного тока) при подаче отрицательного напряжения на вывод диода, соединенный с р-областью. Видно, что обратный ток $I^*_{\text{обр}}$ очень небольшой, но при повышении отрицательного напряжения начиная с -2В резко увеличивается, т.е. происходит пробой перехода – резко возрастает обратный ток. Как правило, после этого характеристики диода не восстанавливаются, т.е. он переходит в неисправное состояние. В то же время на рис. 5.11 видно, что при подаче на р-область даже небольшого положительного напряжения прямой ток через диод резко возрастает. Однако слишком большой прямой ток также опасен для диода.

Таким образом, диод хорошо пропускает ток в одном направлении и плохо в другом. Иными словами, в прямом направлении диод имеет небольшое сопротивление, а в обратном – очень большое. Эти особенности полупроводниковых переходов и используются при проверке диодов с помощью омметров.

Наиболее часто встречающиеся неисправности диодов – короткое замыкание (пробой р-n перехода) и обрыв. Пробой характеризуется существенным увеличением обратного тока. Такие неисправности полупроводниковых диодов легко выявить с помощью омметра, измеряя прямое и обратное сопротивление диода.

В процессе измерения омметр одновременно является источником тока. Омметры создают небольшой ток, поэтому измерения с их помощью обычно

неопасны для исправных диодов. Для измерения в другом направлении надо просто поменять местами щупы (зонды), с помощью которых омметр подсоединяется к контролируемому элементу. При коротком замыкании (пробое) имеет место резкое уменьшение сопротивления в обратном направлении. В случае пробоя диода прямое и обратное сопротивления будут небольшими и составят несколько Ом или десятков Ом, а при обрыве в обоих направлениях они будут очень большими.

Исправные диоды имеют прямое сопротивление в пределах (типичные значения): германиевые точечные – 50–100 Ом; кремниевые точечные – 150–500 Ом и плоскостные (германиевые и кремниевые) – 20–50 Ом.

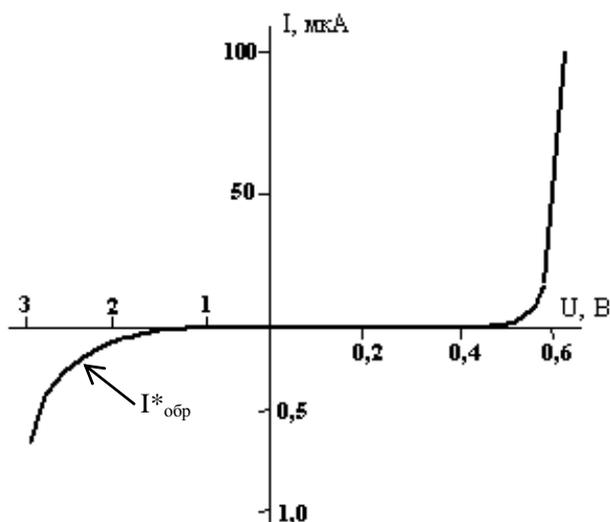


Рис. 5.11. Вольтамперная характеристика диода

В некоторых случаях при измерении сопротивления диода, имеющего утечку, показание прибора при подключении может медленно уменьшаться и, достигнув определенного значения, останавливаться. При повторном подключении процесс должен повторяться.

Результаты измерений могут существенно зависеть от напряжения источника питания омметра, его входного сопротивления. Если имеется заведомо исправный аналогичный диод, то рекомендуется сверить результаты измерений проверяемого и заведомо исправного диодов. В случае исправности проверяемого диода результаты измерений должны быть близкими.

Проверка транзисторов

В отличие от диода, транзистор состоит из двух полупроводниковых переходов. При этом каждый из переходов можно рассматривать как аналог обычного диода. Транзистор p-n-p представляет собой для постоянного тока как бы два последовательно соединенных диода (рис. 5.12), причем катоды диодов, т.е.

п-области, соединены вместе и подключены к выводу базы, а р-области (аноды) подключены к выводам эмиттера и коллектора. Возможны n-p-n транзисторы. В них база находится в месте сочленения р-областей, а эмиттер и коллектор присоединяются к n-областям.

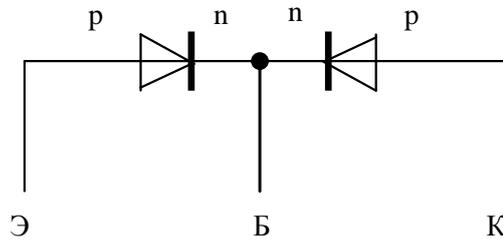


Рис. 5.12. Простейшая модель р-п-р транзистора

Если в исправном транзисторе типа р-п-р к базе (Б) подключить положительный полюс внутренней батареи омметра, то переходы будут заперты и омметр покажет большое сопротивление между базой и коллектором или эмиттером. Если же к базе подключить отрицательный полюс источника, то омметр покажет малое сопротивление с любым из выводов эмиттера или коллектора. Таким образом, р-п-переходы между эмиттер-базой и базой-коллектором могут проверяться как диоды. Обратное сопротивление отличается от прямого в десятки раз. Для n-p-n транзисторов полярности подаваемых напряжений надо поменять на обратные. Однако надо иметь в виду, что для ряда транзисторов предельно допустимое напряжение на переходах может быть меньше, чем напряжение батареи тестера.

На рисунке 5.13 для р-п-р и n-p-n транзисторов показаны пары выводов, между которыми должно быть низкое (высокое) сопротивление.

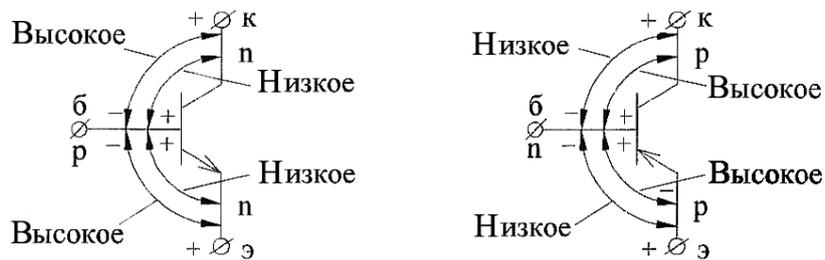


Рис. 5.13. Уровни величины сопротивлений переходов при разных направлениях включения

Типичные неисправности транзисторов: обрывы и пробой полупроводниковых переходов. Возможна также частичная потеря усилительных свойств.

Проверку переходов транзисторов с помощью омметра в аппаратуре рекомендуется производить, отключив питающее напряжение и отпаяв базовый вы-

вод из схемы. Прямые сопротивления переходов транзисторов составляют десятки-сотни Ом, обратные – сотни килоомм, единицы мОм.

Еще один быстрый способ проверки транзистора без его выпайки и при отключенном питании – измерение омметром сопротивления между выводами эмиттера и коллектора при замыкании базы с коллектором и при соединении базы с эмиттером. При исправном транзисторе в первом случае омметр покажет малое сопротивление, во втором – порядка нескольких сотен тысяч или десятков тысяч Ом. Для мощных транзисторов измеренное обратное сопротивление может быть несколько десятков Ом, а обратное – несколько кОм.

Если имеет место частичная потеря усилительных свойств, то надо проверить коэффициент усиления β (рис. 5.14а).

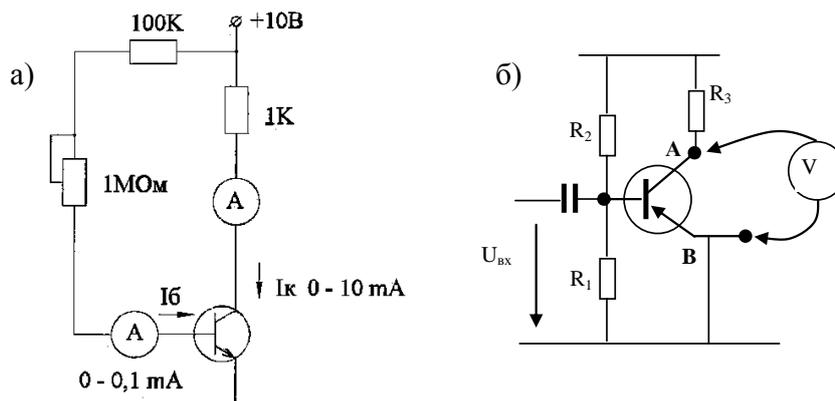


Рис. 5.14. Схемы для проверки коэффициента усиления (а) и пробы (замыкания) переходов транзистора (б)

Практически любой омметр имеет режим измерения напряжений. Документация на электронные устройства, изготовленные с использованием транзисторов, диодов, резисторов и т.п., может содержать специальные **Карты напряжений**. В них указаны напряжения в отдельных точках схемы относительно корпуса аппаратуры при нормальной работе, в том числе на выводах транзисторов. Измеряя напряжения в этих точках, можно примерно определить место неисправного элемента (не только транзистора). Для подозреваемого транзистора затем проверяются сопротивления переходов.

Для проверки транзистора, включенного в схему и работающего в режиме усилителя, необходимо измерить постоянные напряжения коллектор-эмиттер $U_{кэ}$ и база-эмиттер $U_{бэ}$. Напряжение $U_{кэ}$ должно находиться в интервале между 0 и $U_{пит}$ – напряжением источника питания. Обычно напряжение $U_{бэ}$ имеет значение в несколько десятых вольт: положительное относительно эмиттера для p-n-p и отрицательное – для n-p-n транзисторов. Если это напряжение равно нулю, переход закорочен (пробой), если он выше указанного значения, то имеет место обрыв.

В общем случае любой транзисторный каскад имеет схему, представленную на рис. 5.14б.

Во включенной схеме, используя вольтметр, проверку транзисторов можно также выполнить следующим образом (применительно к схеме рис. 5.14б):

- между эмиттером и коллектором (точки А и В) включается вольтметр V;
- параллельно резистору R₁ подсоединяется еще один резистор такого же номинала, что вызывает увеличение отрицательного напряжения на базе транзистора. При этом исправный транзистор увеличивает свою проводимость, и включенный вольтметр заметно уменьшит свои показания;
- убираем ранее подсоединенный резистор и замыкаем накоротко резистор R₁. Смещение на базу станет равным нулю, транзистор закрывается. Тогда, при исправном транзисторе, вольтметр должен показать напряжение, близкое к величине напряжения питания каскада. Если при этих проверках показания вольтметра будут не такими, как указано выше, значит, транзистор неисправен и его следует заменить.

Существует большое число радиолюбительских простых приборов, позволяющих проверить работоспособность транзисторов в реальных режимах работы. Примеры таких схем приведены на рис. 5.15 и 5.16. Их можно применять, не выпаивая проверяемый транзистор (при этом схема, содержащая проверяемый транзистор, должна быть обесточена).

В схеме рис. 5.15 выводы проверяемого транзистора подключают к точкам Э, Б и К, а коллектор транзистора V₂ замыкают переключателем S2 на высокоомный телефонный капсюль В1. Транзистор V₁ совместно с проверяемым образуют симметричный мультивибратор, колебания с которого подаются на телефонный капсюль В1 и при исправном проверяемом транзисторе слышен фон с частотой 400÷800 Гц.

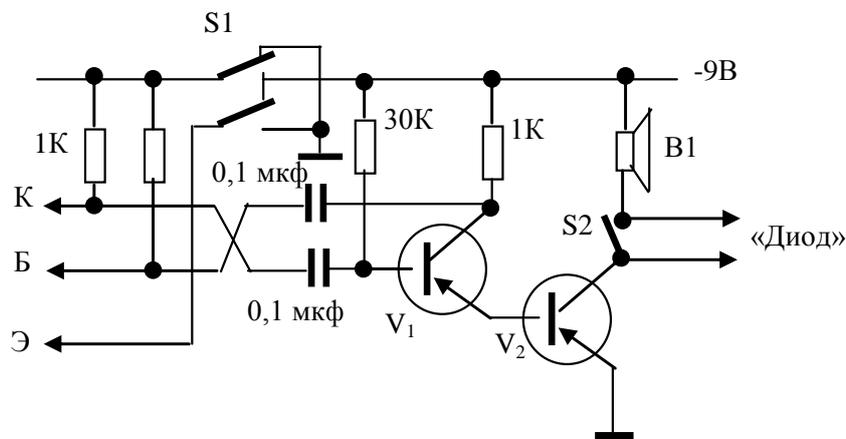


Рис. 5.15. Схема для проверки транзисторов и диодов

Переключателем S1 изменяют полярность питания в соответствии со структурой проверяемого транзистора (р-п-р – нижнее положение или п-р-п – верхнее положение). Для проверки диода контакты переключателя S2 размыкают и к вы-

водам «Диод» подсоединяют выводы исследуемого диода, а к щупам Э, Б, К – заведомо исправный транзистор. Если колебания мультивибратора прослушиваются с одинаковой громкостью при изменении полярности подключения испытуемого диода, – он пробит; отсутствие звука означает обрыв в диоде. В случае исправного диода уровень громкости значительно меняется при изменении полярности подключения диода.

В схеме тестера на рис. 5.16 для проверки транзистора его выводы надо подключить к соответствующим выводам (К, Б, Э).

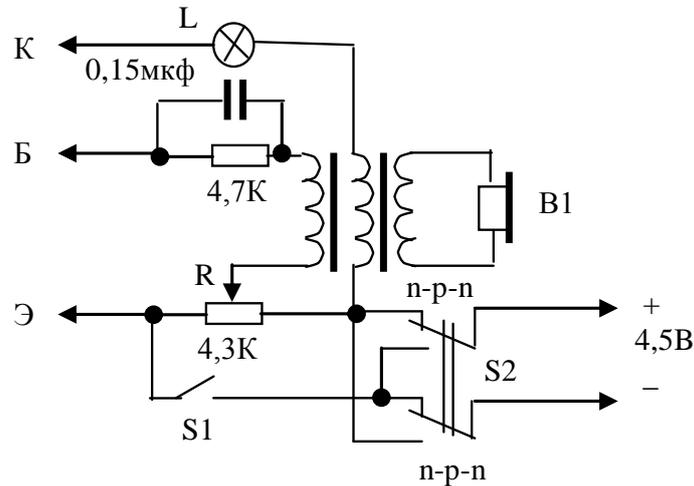


Рис. 5.16. Схема для проверки транзисторов

При подключении транзистора образуется генератор звуковой частоты и из динамика В1 раздается звук. Генерации добиваются переменным резистором R, перемещая его движок. При пробое переходов транзистора вспыхивает лампа L, а при их обрывах – лампа не горит, а телефон не издает звуков.

Выключателем S1 подают питание на тестер, а переключателем S2 изменяют полярность поступающего на генератор напряжения в зависимости от структуры проверяемого транзистора (n-p-n или p-n-p).

Некоторые особенности имеет проверка полевых транзисторов. Чтобы предотвратить выход из строя полевого транзистора, во время проверки очень важно соблюдать некоторые правила безопасности. Дело в том, что полевые транзисторы очень чувствительны к статическому электричеству, поэтому их рекомендуется проверять, предварительно организовав заземление. У них возможен пробой между стоком и истоком из-за статического электричества. Чтобы снять с себя накопленные статические электрические заряды, необходимо надеть на руку заземляющий антистатический браслет, а при хранении полевых транзисторов, особенно маломощных, их выводы должны быть замкнуты между собой.

У исправного маломощного полевого транзистора между всеми его выводами должно быть большое сопротивление (мегаомы). Следует заметить, что име-

ются некоторые исключения. Если при проверке приложить положительный щуп тестового прибора к затвору (G) транзистора n-типа, а отрицательный – к истоку (S), зарядится емкость затвора и транзистор откроется. Тогда при последующем замере сопротивления между стоком (D) и истоком (S) прибор покажет некоторое значение сопротивления, которое зависит от ряда факторов. Поэтому перед измерением между стоком и истоком надо кратковременно замкнуть накоротко все ножки транзистора, чтобы разрядить емкость затвора. После этого сопротивление сток-исток должно стать бесконечным. В противном случае транзистор признается неисправным.

В мощных полевых транзисторах переход между стоком и истоком может вести как обычный диод, т.е. в одном направлении иметь большое сопротивление, а в другом – малое.

Примеры переносных тестеров

Промышленность выпускает большое число различных приборов, позволяющих измерять ток, напряжение, сопротивление, емкость или индуктивность. Часто их называют тестерами. Они различаются по функциональным возможностям, точности и методам измерений, конструктивному исполнению др. Современные приборы имеют память для хранения измерений, могут подключаться к ЭВМ.

На рисунке 5.17 показан простой стрелочный прибор (тестер) для выполнения таких измерений. Он позволяет измерять постоянный и переменный ток, напряжение и сопротивление.

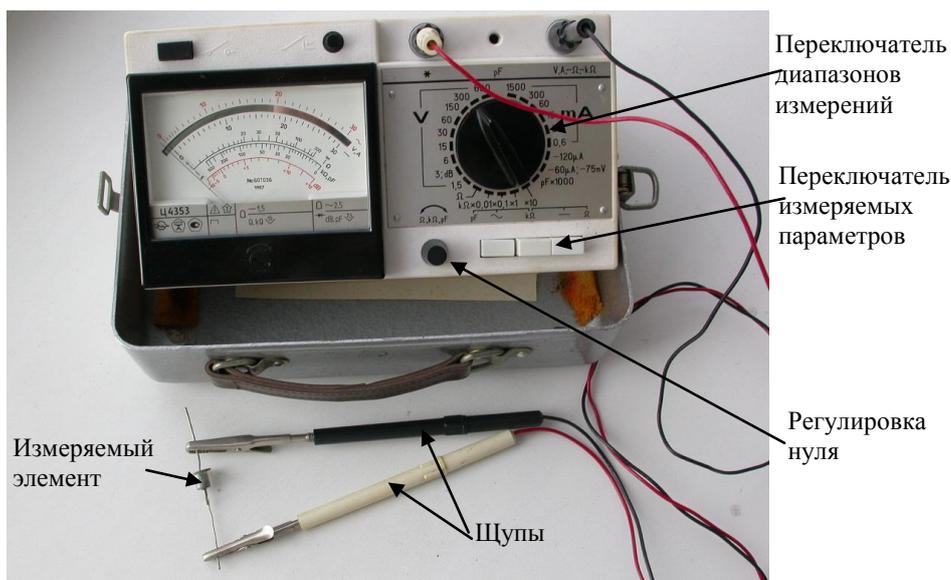


Рис. 5.17. Стрелочный тестер

Переключатель измеряемых параметров содержит три кнопки для выбора режимов измерений сопротивлений, постоянного или переменного напряжений

(тока). При измерении сопротивлений предварительно надо закоротить концы щупов и ручкой регулировки нуля установить стрелку на ноль.

Стрелочные приборы сегодня почти не выпускаются. Их современные аналоги представляют собой электронные цифровые приборы, в которых результат измерения выводится на цифровой дисплей. Они имеют небольшие габариты, могут выполнять несложную обработку результатов измерений. На рисунке 5.18 показаны три таких прибора.

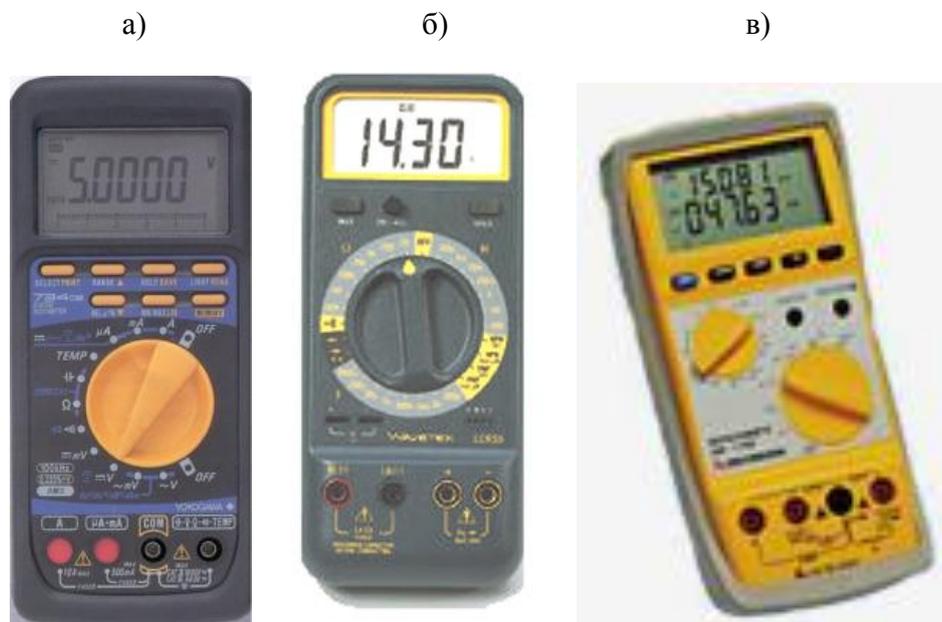


Рис. 5.18. Электронные тестеры: а) мультиметр серии 734 фирмы Yokogawa Electric, б) мультиметр LCR55 фирмы Fluke, в) мультиметр AM-1109

Мультиметры серии 734 снабжены встроенной памятью и последовательным интерфейсом RS-232C с инфракрасным разъемом. Кроме собственно измерений, он имеет функции запоминания и выделения максимального (пикового) значения, определения и вывода на дисплей минимальных/максимальных и средних значений, отображения величины в процентах и в дБ и др.

Модель LCR55 имеет 28 диапазонов для измерения индуктивности, емкости и сопротивления, специальные гнезда для установки конденсаторов и транзисторов при проведении измерений; может запоминать результаты измерений и выделять максимальные значения.

С помощью мультиметра AM-1109 можно измерять постоянное и переменное напряжение, измерять параметры резисторов, емкостей, проводимость полупроводниковых переходов, частоту синусоидальных и импульсных сигналов, определять скважность импульсов и среднеквадратичные значения токов и напряжений. Особенностью прибора является возможность измерения одновременно по двум каналам.

5.4. Погрешности методов и средств измерений

Единицы измерения

Проведение измерений предполагает наличие системы единиц для оценки значений измеряемых параметров, которая должна быть всеобщей, понятной и удобной для применения, а также эталонов единиц измерений.

Эталон единицы величины – средство измерения, предназначенное для воспроизведения и хранения единицы величины (или кратных либо дольных значений единицы величины) с целью передачи ее размера другим средствам измерения данной величины¹.

Первыми начали применять универсальные системы единиц специалисты в механике. В прошлом существовало несколько вариантов систем единиц, как, например, метр, грамм, секунда и миллиметр, миллиграмм, секунда, но постепенно общепринятой стала система СГС (сантиметр, грамм, секунда). Затем была разработана система МКС (метр, килограмм, секунда). Система «механических» единиц стала основой для единиц измерения, введенных для оценки параметров электрического поля и вообще электрических величин.

В 1960 г. XI Генеральная конференция по мерам и весам утвердила Международную систему единиц СИ. Она включает в себя систему единиц МКС (механические единицы) и систему МКСА или систему Джорджи (электрические единицы). В системе СИ первоначально было шесть основных единиц, которые в 1971 г. были дополнены величиной для определения количества вещества (табл. 5.1).

Таблица 5.1

Основные единицы системы СИ

Величина	Символ	Наименование	Обозначение	Размерность
Длина	l	Метр	м	L
Масса	m	Килограмм	кг	M
Время	t	Секунда	с	T
Электрический ток	I	Ампер	A	I
Температура	T, θ	Кельвин	K	θ
Сила света	I _v	Кандела	кд	J
Количество вещества	ν	Моль	моль	mol

Через основные единицы системы СИ определяются все другие единицы. Например, единица мощности «Ватт» определена через размерности ML^2T^{-3} .

¹ Закон РФ «Об обеспечении единства измерений».

Системой СИ введены также стандартные наименования и обозначения десятичных кратных и дольных единиц (табл. 5.2).

Таблица 5.2

Десятичные кратные и дольные единицы

Множитель	Наименование приставки	Обозначение приставки	Множитель	Наименование приставки	Обозначение приставки
10^{18}	Экса	Э	10^{-1}	Деци	д
10^{15}	Пета	П	10^{-2}	Санتي	с
10^{12}	Тера	Т	10^{-3}	Милли	м
10^9	Гига	Г	10^{-6}	Микро	мк
10^6	Мега	М	10^{-9}	Нано	н
10^3	Кило	к	10^{-12}	Пико	п
10^2	Гекто	г	10^{-15}	Фемто	ф
10	Дека	да	10^{-18}	Атто	а

Существуют и другие системы единиц. Так, в Англии и Северной Америке до сих пор широко распространена английская система (в ней для измерения длины используются дюйм и фут, веса – фунт и т.д.). Однако система СИ постепенно вытесняет другие системы единиц.

Чтобы обеспечить совместимость единиц измерений при их применении в разных местах, создана система эталонов единиц измерений.

Существуют четыре различных уровня эталонов: рабочие, вторичные, первичные и международные.

Рабочие и вторичные эталоны хранятся в отраслевых лабораториях. Рабочие используются для контроля и калибровки лабораторного инструмента (в том числе различных технических средств контроля) в данной отрасли производства, а также для выполнения некоторых точных измерений. Вторичные служат для контроля и калибровки рабочих эталонов.

Первичные эталоны хранятся в специальных национальных лабораториях. Они могут использоваться для калибровки присланных в лабораторию вторичных эталонов. Сами первичные эталоны калибруются путем проведения специальных измерений, результаты которых выражаются в фундаментальных единицах. Результаты таких измерений в разных национальных лабораториях затем сопоставляются для получения среднего значения первичного эталона.

Международные эталоны хранятся в Международном бюро мер и весов (Франция). Они не подлежат использованию для измерений или калибровки и периодически проверяются путем выполнения абсолютных измерений в соответствующих фундаментальных единицах. Что из себя представляют эти эталоны? Так, эталон метра менялся несколько раз. Первоначально он был определен как десятиmillionная часть меридиана земли, проходящего через Париж, и был

выполнен в виде платино-иридиевого стержня. Сегодня метр представляет расстояние, которое пробегает свет в вакууме за время $1/299792458$ с (из этого понятия видно, что по международным стандартам скорость света $299\,792\,458$ м/с). В качестве международной секунды в 1967 г. на XIII Генеральной конференции по мерам и весам определили продолжительность $9\,192\,631\,770$ периодов излучения, соответствующих переходу между двумя определенными уровнями основного состояния атома ^{133}Cs .

Классификация погрешностей измерений

Измерения проводят с помощью средств измерений.

Средство измерений – техническое устройство, предназначенное для измерений.

Средство измерений, предназначенное для выработки сигнала по измеренным значениям в форме, доступной для непосредственного восприятия наблюдателем, называют **измерительным прибором**. Результат измерения отображается на индикаторе: стрелочном, цифровом и др.

Все измерения разделяются на прямые и косвенные.

При проведении **прямых измерений** непосредственно регистрируется численное значение интересующей нас величины Y_k . Например, механические стрелочные весы выполняют прямое измерение веса объекта контроля. Под весом объекта прогибается пружина, связанная со стрелкой. Положение стрелки на проградуированном циферблате сразу показывает вес.

При **косвенных измерениях** интересующая нас величина недоступна прямому измерению. В этом случае наблюдается и измеряется другая величина Z_k , которая является некоторой функцией от интересующей нас величины Y_k . Например, аналоговый омметр на рис. 5.2 первоначально измеряет ток, величина которого зависит от величины сопротивления диагностируемого элемента. Существует обратно пропорциональная зависимость $I=U/R_x$, устанавливаемая законом Ома, между значением измеренного тока I и сопротивлением R_x диагностируемого элемента. Поэтому, проградуировав соответствующим образом шкалу амперметра, можно по его показаниям определять значение измеряемого сопротивления, хотя фактически прибор определяет ток.

Любому измерению присущи ошибки, как бы тщательно ни был изготовлен измерительный прибор и выполнены все этапы измерения. Опасность ошибок состоит в том, что они влияют на качество решений, которое принимает человек на основе полученных результатов измерений. Поэтому очень важно оценить и указать точность результата измерения. Если ее не учитывать, то можно принять неправильное решение о фактическом значении измеряемого параметра и соответственно о техническом состоянии элемента.

Точность, которой следует добиваться при измерении, зависит от его цели. Вовсе не всегда следует стремиться провести измерение как можно более точно, тем более что проведение особо точных измерений, как правило, требует повышенных материальных и временных затрат. Например, если заводской допуск на сопротивление резистора 20%, то измерение можно провести и с 5% точностью.

Однако требования к точности измерительного прибора резко возрастут, если резистор изготовлен с точностью 0,1%.

Изучением различных видов ошибок и причин их возникновения, разработкой методов уменьшения ошибок и их влияния на конечный результат деятельности человека занимаются теория измерений и теория ошибок.

Одна из классификаций ошибок, используемая в теории измерений, показана на рис 5.19.

Личные ошибки связаны с действиями человека, проводящего измерения. Он может иметь нарушенное обоняние, цветоощущение, пониженный уровень зрения и др. Каждый по-своему оценивает положение даже стрелки весов или часов. Личные ошибки зависят также от уровня знаний и практической подготовленности, степени утомляемости, ответственности за выполняемую работу и т.п.

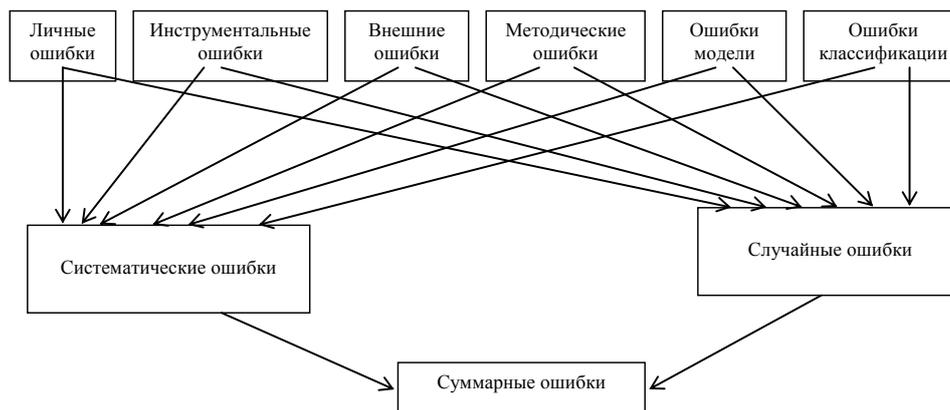


Рис. 5.19. Основные виды ошибок

Инструментальные ошибки возникают вследствие невозможности изготовить измерительное устройство идеально точно. Даже нанесение шкалы на линейку или циферблат измерителя невозможно выполнить с абсолютно одинаковым шагом, не говоря уже о неидеальности изготовления механических частей измерительных механизмов. У электронных измерительных устройств, реализующих цифровые алгоритмы обработки, возможны ошибки округления при оцифровке аналогового сигнала или при вычислениях.

Внешние ошибки связаны с воздействием окружающей среды на измерительное устройство и процесс измерения. Такими воздействиями могут быть колебания напряжения в сети, механические толчки и вибрации, изменения температуры, влажности и давления воздуха, электрические разряды искусственного или естественного происхождения и т.п.

Методические ошибки и **ошибки модели** связаны с реализуемым в приборе методом измерений.

Ошибки классификации возникают, когда решение принимается путем сравнения измеренного значения с некоторыми априори заданными значениями, описывающими эталонные объекты, к одному из которых надо отнести контролируемый объект.

По своему характеру любая из перечисленных выше ошибок может быть случайной или систематической. **Случайными** являются ошибки, которые невозможно учесть и воспроизвести при повторных измерениях. **Систематические ошибки** зачастую обусловлены неучетом важной, существенной связи, присущей наблюдаемому объекту или процессу измерений.

Резкой границы между различными видами ошибок не существует. Одна и та же ошибка может считаться в одних случаях систематической, а в других – случайной. Это, например, может быть связано с длительностью мешающего внешнего воздействия. Если мешающий фактор одинаково действует при всех измерениях, то его можно рассматривать в качестве систематической ошибки.

Все вышерассмотренные виды ошибок могут приводить к неправильным результатам измерения.

Качество измерений, отражающее отклонение результата измерений от истинного значения измеряемой величины, называют **погрешностью системы измерения**. Фактически погрешность измерения – это число, указывающее возможные границы неопределенности полученного при измерении значения. Ее величина зависит от характеристик самого измерительного прибора, используемого метода измерений, влияния внешних условий на процесс измерений и др.

В теории измерений различают несколько десятков видов погрешностей.

При создании новых измерительных приборов очень важно выделить инструментальные и методические погрешности.

Инструментальными (аппаратными) называют погрешности, которые определяются конструкцией и схемой самого измерительного прибора. Они могут быть определены при испытаниях и занесены в паспорт прибора.

Методическими называют погрешности, которые связаны с методом проведения измерений. Примером такой погрешности может служить погрешность измерения напряжения стрелочным вольтметром (рис. 5.1). При подключении вольтметра его сопротивление шунтирует измеряемую цепь. Поэтому погрешность показания вольтметра зависит от сопротивления измеряемой цепи, так как часть тока, проходящая по катушке вольтметра, определяется не только величиной сопротивления катушки (которое постоянно), но и сопротивлением измеряемой цепи. Чтобы уменьшить погрешность измерения стрелочным вольтметром, его катушку делают с максимально большим сопротивлением. Тогда колебания тока в катушке вольтметра из-за изменения сопротивления измеряемой цепи будут меньше влиять на результат измерения.

В эксплуатационной документации дается 1–2 вида погрешностей, характеризующих качество измерений данным прибором (обычно это некоторые интегральные погрешности).

Погрешности прибора определяют в ходе его поверки, при этом используют образцовые средства измерений с очень низкими погрешностями.

Чаще всего в эксплуатационной документации на измерительный прибор указывают относительную или приведенную погрешности либо их варианты.

Абсолютная погрешность $\Delta A = A_{п} - A_{д}$ выражается в единицах измеряемой величины и показывает отклонение результата измерения от действительной величины ($A_{п}$ – показание прибора, $A_{д}$ – истинное значение измеряемой величины).

Относительная погрешность вычисляется (в процентах) как отношение абсолютной погрешности к истинному значению измеряемой величины: $\delta = (\Delta A / A_d) 100\%$.

Погрешность может меняться при изменении действительного значения измеряемой величины. В таких случаях для прибора указывают приведенную погрешность.

Приведенная погрешность – отношение (в процентах) абсолютной погрешности к значению A_v верхнего предела измерений: $\delta_{п} = (\Delta A / A_v) 100\%$.

Погрешности, возникающие в нормальных условиях работы измерительного прибора, называют **основными**. Изменение основной погрешности, обусловленное изменением внешних условий относительно нормальных, вызывает появление **дополнительных** погрешностей.

При нормировании погрешностей прибора устанавливают пределы допускаемых погрешностей (основной и всех дополнительных), а также нормальные условия и допускаемые отклонения от нормальных значений для всех влияющих факторов. Обобщенной характеристикой пределов допускаемых погрешностей является класс точности измерительного прибора.

Наибольшее значение приведенной погрешности, соответствующее максимально возможной ΔA_{\max} абсолютной погрешности, допускаемой данным прибором, называется **классом точности**:

$$K = (\Delta A_{\max} / A_m) \cdot 100\%,$$

где A_m наибольшее значение, которое можно измерить по шкале прибора.

По ГОСТ 1845-52, электроизмерительные приборы делятся на **семь классов точности**: 0,1; 0,2; 0,5; 1,0; 1,8; 2,5; 4,0.

Значение класса точности помещается на лицевой стороне прибора. Зная K , можно найти наибольшую абсолютную погрешность: $\Delta A_{\max} = (K \cdot A_m) / 100$.

Все погрешности можно разделить на два типа – систематические и случайные.

Систематические обычно остаются постоянными на протяжении всей серии измерений. **Случайные** – хаотически изменяются и в равной степени могут быть как положительными (увеличивают значение измеряемой величины по отношению к ее истинному значению), так и отрицательными (уменьшают значение измеряемой величины).

Разновидность случайных ошибок – грубые ошибки или **промахи**. Их источник – неправильные отсчеты по прибору, неправильная запись отсчета и т.п. В большинстве случаев промахи хорошо заметны, так как соответствующие им отсчеты резко отличаются от других. При обработке результатов измерений такие отсчеты следует отбрасывать. Однако промах можно заметить только в том случае, если проделано несколько измерений одной и той же величины. Поэтому, если есть основания полагать, что на результат измерения воздействуют случайные факторы, нельзя ограничиваться одним измерением, обязательно следует провести их несколько раз. Тогда можно уменьшить погрешность путем статистической обработкой совокупности измерений (обычно определяют среднее значение нескольких результатов измерения).

Оценка измерений с учетом погрешностей

Как правило, в методических указаниях в документации к прибору, если его измерения подвержены случайным ошибкам, указывается на необходимость многократных измерений.

Предположим, что мы провели серию измерений некоторой физической величины x . Результат отдельного i -го измерения обозначим через x_i , а общее число измерений n . Если систематическая ошибка отсутствует, то разумно предположить, что значения измерений расположатся вблизи неизвестного нам истинного значения A_d измеряемой величины, причем отклонения в сторону больших и меньших значений будут равновероятными. Опыт показывает, что во многих случаях такое предположение справедливо. Тогда в качестве наилучшего приближения к истинному значению обычно берут среднее арифметическое \bar{x} , а возможное отклонение от истинного – как среднеквадратичное S_x отклонение по всем n отдельным измерениям:

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^n x_i \quad \text{и} \quad S_x = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}} \approx \frac{1}{n} \sqrt{\sum_{i=1}^n (x_i - \bar{x})^2}.$$

Систематические ошибки возникают вследствие погрешностей измерительной аппаратуры (спешит или отстаёт секундомер, сбилась настройка электронного прибора, сместился регулировочный груз в механических весах, и т.д.), а также из-за того, что условия измерения отличаются от нормальных для работы прибора, а поправку на это несоответствие не делают. Например, при определении плотности вещества по объёму тела надо учитывать, что размеры тел при повышении температуры обычно увеличиваются, а масса остается одной и той же. Поэтому плотность нагретых тел ниже, чем холодных.

Возможно, роль случайных факторов незначительна. Тогда измерение каждой величины проводится, как правило, только один раз, и точность его определяется погрешностью используемого прибора (ее называют **приборной**).

При оценке результата измерений интервал, в пределах которого с определенной вероятностью может оказаться истинное значение измеряемой величины, часто указывают в виде выражения $A_n \pm \Delta A$. Величина ΔA называется **погрешностью** или **ошибкой** измерения, а интервал значений от $(A_n - \Delta A)$ до $(A_n + \Delta A)$ – **доверительным интервалом**.

Доверительный интервал не является исчерпывающей характеристикой точности измерения. Нужна количественная характеристика его достоверности, показывающая, насколько можно быть уверенным в том, что истинное значение измеряемой величины окажется в пределах доверительного интервала. Такая характеристика – вероятность того, что значение A_n отличается от истинного не более, чем на ΔA – называется **доверительной вероятностью**.

Пусть результат серии измерений записан в виде $X=25\pm 2$ и приведенный доверительный интервал (от 23 до 27) соответствует доверительной вероятности $\alpha=0,95$. Что это означает?

Пусть измерения производятся большое число раз. Например, сделаем $n=100$ однотипных измерений. Результаты будут отличаться друг от друга. При указанной доверительной вероятности примерно в 95 случаях результаты будут отличаться от истинного значения измеряемой величины не более чем на $\Delta x=2$, а результаты остальных измерений выйдут за пределы доверительного интервала.

Погрешность может описываться достаточно сложной формулой. Так, для дозиметра РМ 1621 основная относительная погрешность измерения мощности эквивалентной дозы описывается выражением $\delta = \pm(15+0,0015/N +0,01N)\%$, где N – измеренная мощность.

Оценка погрешностей при косвенных измерениях имеет некоторую особенность. При косвенных измерениях искомая физическая величина A является функцией величин X, Y, Z, \dots , значения которых были определены с помощью прямых измерений. Результат косвенного измерения записывается в виде:

$$A \pm \Delta A,$$

где $A = f(X, Y, Z, \dots)$ – значение искомой величины, рассчитанное по средним значениям параметров X, Y, Z, \dots , если каждый из них измерялся по несколько раз; ΔA – абсолютная погрешность косвенного измерения, зависящая от погрешностей измерения параметров X, Y, Z, \dots (т.е. от $\Delta X, \Delta Y, \Delta Z, \dots$).

В простейших случаях абсолютную и относительную погрешность косвенных измерений нетрудно подсчитать. Рассмотрим несколько примеров.

Пусть $A = X+Y$. Если известны погрешности ΔX и ΔY , то

$$A \pm \Delta A = (X \pm \Delta X) + (Y \pm \Delta Y).$$

Максимальное значение погрешности равно при этом $\Delta A = \Delta X + \Delta Y$.

Такой же будет максимальная абсолютная погрешность, если $A=X-Y$.

Отсюда следует, что относительные погрешности величин, являющихся суммой или разностью двух параметров, равны соответственно:

$$E = \frac{\Delta X + \Delta Y}{X + Y} \quad \text{и} \quad E = \frac{\Delta X + \Delta Y}{X - Y}.$$

При косвенных измерениях контролируемый параметр может являться некоторой функцией от нескольких измерений. В таких случаях при определении возможных погрешностей надо учитывать вид функции. Например, площадь (A) обычно определяется путем перемножения результатов измерений линейных размеров длины (X) и ширины (Y) объекта, т.е. $A = X \cdot Y$.

Тогда

$$A \pm \Delta A = (X \pm \Delta X) (Y \pm \Delta Y) \approx X \cdot Y \pm X \cdot \Delta Y \pm Y \cdot \Delta X + \Delta X \cdot \Delta Y$$

Обычно, последнее слагаемое $\Delta X \cdot \Delta Y$ этой формулы значительно меньше остальных и им можно пренебречь. Тогда абсолютная погрешность:

$\Delta A = X \cdot \Delta Y + Y \cdot \Delta X$, а относительная

$$E = \frac{X \cdot \Delta Y + Y \cdot \Delta X}{X \cdot Y} \quad \text{или} \quad E = \frac{\Delta X}{X} + \frac{\Delta Y}{Y}.$$

5.5. Принципы и методы организации измерений параметров элементов в автоматизированных системах поэлементного диагностирования

Общие принципы организации диагностирования

Для организации внутрисхемного диагностирования необходимо:

- обеспечить доступ к внутренним контрольным точкам объекта диагностирования;
- исключить влияние других элементов при проверке очередного элемента;
- обеспечить защиту активных электрорадиоэлементов (транзисторы, микросхемы и др.);
- определить необходимые тестовые воздействия;
- выбрать метод оценки результатов измерений с целью определения технического состояния проверяемого элемента.

Подаваемые на элементы напряжения и проходящие токи в той или иной мере зависят от окружающих элементов. Это создает серьезные проблемы в определении точных значений параметров проверяемого элемента по результатам измерений. Поэтому проблема измеряемости характеристик отдельных элементов является центральной при применении внутрисхемного диагностирования.

Обычно измерения организуются в несколько этапов:

- измерения без включения питания диагностируемого объекта с подачей от измерительной системы на объект постоянных или переменных напряжений (токов);
- включение питания диагностируемого объекта и измерение токов потребления;
- измерения при включенном питании диагностируемого объекта.

Электрическая цепь, содержащая проверяемый элемент, может иметь две и более точек (полюсов) для подключения измерительной системы. Очень часто путем создания закороток и/или заземления некоторых полюсов эта цепь с точки зрения «измерителя» становится двух-, трех- или четырехполюсником. Общая модель трехполюсника приведена на рис. 5.20.

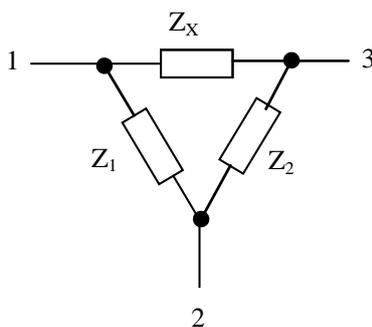


Рис. 5.20. Схема трехполюсника

На рисунке 5.20 полюсы, к которым может подключаться измерительная схема имеют номера 1,2,3; Z_X – сопротивление проверяемого элемента, Z_1 и Z_2 – сопротивления шунтирующих цепей. Если трехполюсник имеет реактивные элементы, то сопротивления будут иметь комплексный характер.

В принципе, можно составить математическую модель электрических цепей, содержащих проверяемый элемент, подставить результаты измерений в модель и вычислить параметры элементов проверяемой цепи. Например, ниже приведена система уравнений, представляющая математическую модель трехполюсника и позволяющая вычислить по результатам трех измерений значения Z_1 , Z_2 и Z_X :

$$\begin{aligned} Z_{RX} &= Z_X Z_1 / (Z_X + Z_1), \\ Z_{R1} &= Z_X Z_2 / (Z_X + Z_2), \\ Z_{R2} &= Z_1 (Z_X + Z_2) / (Z_X + Z_1 + Z_2). \end{aligned}$$

Здесь Z_{RX} – сопротивление между полюсами 1 и 2 (рис. 1.20) при замкнутых полюсах 2 и 3, Z_{R1} – сопротивление между полюсами 2 и 3 при замкнутых 1 и 2, Z_{R2} – сопротивление между полюсами 1 и 2 без замыкания каких-либо полюсов. Сделав измерения Z_{RX} , Z_{R1} и Z_{R2} , можно подставить их в уравнения и, сделав соответствующие вычисления с помощью встроенного в систему внутрисхемного диагностирования вычислительного устройства, определить значения Z_1 , Z_2 и Z_X .

Однако в действующих системах внутрисхемного диагностирования широко применяются специальные аппаратные измерительные схемы, которые не только избавляют от необходимости составлять достаточно сложные математические модели, но и позволяют существенно снизить погрешности выполняемых измерений и расчетов.

Схемы измерений

В системах внутрисхемного диагностирования схемы измерений обычно строятся с использованием операционных усилителей, могут изменяться в зависимости от условий и целей измерений.

В системах внутрисхемного диагностирования измерительная схема может подключаться к объекту измерений с помощью нескольких проводов: обычно от 2 до 6. На рисунке 5.21 показаны измерительные схемы для измерения сопротивления элемента Z_X , шунтированного элементами Z_1 и Z_2 . Это трехпроводные схемы измерений.

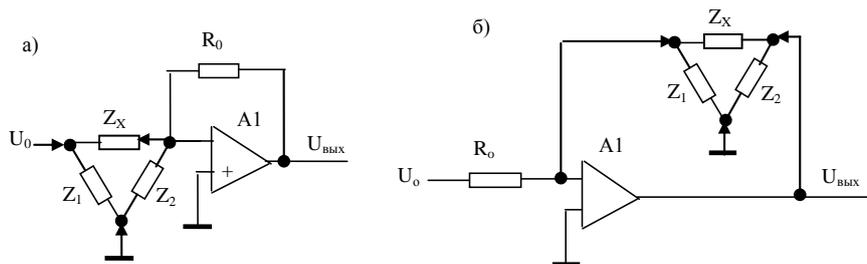


Рис. 5.21. Трехпроводные схемы измерения Z_X : а) по методу заданного напряжения, б) по методу постоянного тока

О параметре проверяемого элемента обычно судят по результатам измерения напряжения или тока. Возможны схемы измерений, где на проверяемый элемент подается известное напряжение и измеряется (прямо или косвенно) ток, проходящий через него (рис. 5.21а). В других схемах через проверяемый элемент проходит известный ток и фиксируется падение напряжения на элементе (рис. 5.21б). Известны схемы измерений, в которых измеряется как проходящий через элемент ток, так и падение напряжения на нем.

Основной критерий выбора схемы измерений – необходимый уровень погрешности измерений. Использование более двух проводов для подключения измерительной схемы к объекту диагностирования, как правило, имеет целью снизить погрешности измерений.

В схеме измерений рис. 5.21а элемент Z_X одним выводом (полюсом) подключается к инвертирующему входу операционного усилителя А1, вторым – к источнику эталонного напряжения U_0 . Третий провод измерительной схемы подключает узел соединения элементов Z_1 и Z_2 к «земле» (т.е. к точке с нулевым потенциалом). Одна из особенностей функционирования операционного усилителя А1 – нулевой потенциал на инвертирующем входе. В результате оба вывода элемента Z_2 (рис. 5.21 а) имеют одинаковый (нулевой) потенциал и через элемент не проходит ток. Тем самым устраняется его влияние на измерения. Два других элемента Z_X и Z_1 в схеме рис. 5.21а подключены параллельно к источнику напряжения U_0 . Поэтому ток через проверяемый элемент Z_X не зависит и от элемента Z_1 . Таким образом, с некоторыми упрощениями, реализуемую схемой функцию можно представить выражением $U_{\text{вых}} = -U_0 \cdot R_0/Z_X$. Если известны U_0 и R_0 , то, измерив $U_{\text{вых}}$, по приведенной выше формуле легко вычислить значение Z_X . Данную схему измерений рекомендуется применять при измерениях сопротивлений в несколько и более килоом.

Схему рис. 5.21б применяют для измерения небольших сопротивлений (десятки – сотни Ом). В ней также устраняется влияние на результат измерений элементов Z_1 и Z_2 . Оба вывода элемента Z_1 находятся под нулевым потенциалом и его параметры не влияют на ток через Z_X . Элемент Z_2 , по сути, выполняет роль нагрузки для операционного усилителя и не влияет на значение $U_{\text{вых}}$, где $U_{\text{вых}} = -U_0 Z_X/R_0$.

Рассмотренные схемы измерений очень просты, но в ряде случаев дают большие погрешности измерений. В частности, для их снижения может понадобиться перед измерениями закортить некоторые точки диагностируемого объекта; значительный вклад в погрешность дают переходные сопротивления зондов, с помощью которых измерительная схема подключается к объекту диагностирования. Для снижения погрешностей измерений применяют многопроводные схемы измерений. Пример шестипроводной схемы измерений приведен на рис. 5.22.

Линии со стрелками соответствуют наконечникам зондов, подключаемых к проверяемому элементу Z_X , который шунтируется элементами Z_1 и Z_2 . Операционный усилитель А1 работает в режиме повторителя и создает ток в со-

ответствии с выражением $I=U_0/R_2$. При большом коэффициенте усиления усилителя А3 весь этот ток направляется на элемент Z_X . Проверяемый элемент Z_X включен в обратную связь усилителя А2, т.е. находится в режиме постоянного тока. Чтобы устранить влияние переходных сопротивлений при подключении зондов от входа и выхода усилителя А3 к выводам элемента Z_X , с помощью двух дополнительных зондов напряжение снимают непосредственно с элемента Z_X . При этом для снижения влияния переходных сопротивлений при съеме падения напряжения с элемента Z_X снимаемые сигналы поступают на дифференциальный усилитель А4 через достаточно большие сопротивления R_4 и R_5 . Переходные сопротивления имеют значения в несколько Ом и их включение в измерительную схему последовательно с большими R_4 и R_5 нивелирует влияние переходных сопротивлений на погрешность измерения. Для устранения влияния переходного сопротивления в точке подключения элементов Z_1 и Z_2 к «земле» в данной схеме использован операционный усилитель А2. Он работает в режиме повторителя с большим коэффициентом усиления и обеспечивает во всех точках обратной связи потенциал «0».

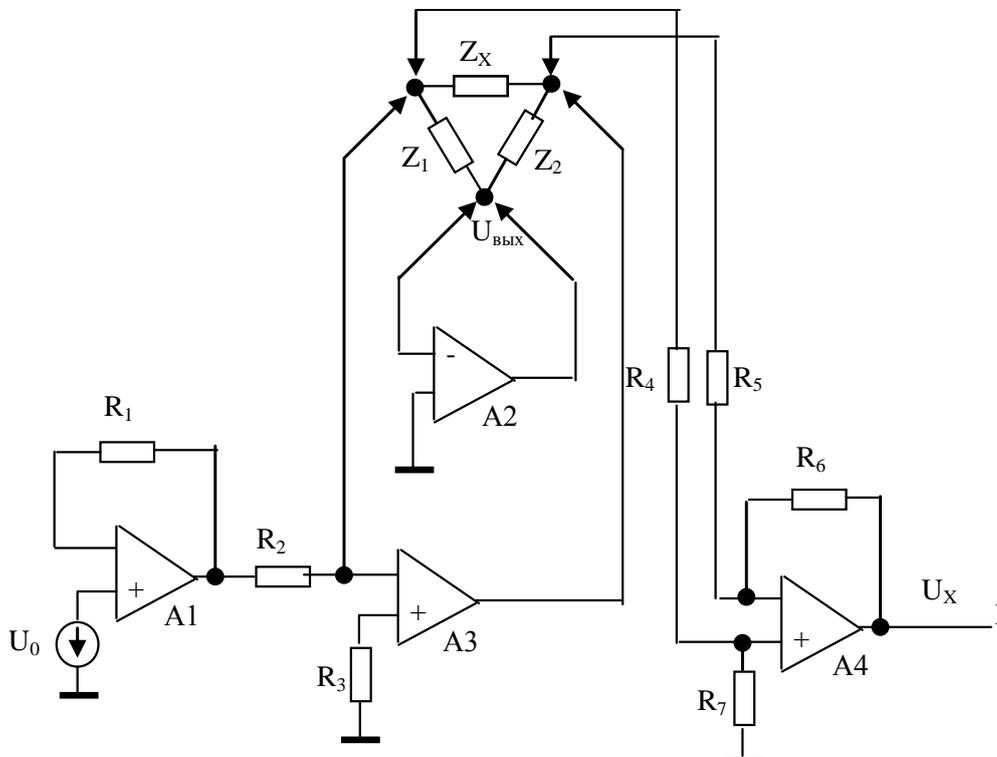


Рис. 5.22. Шестипроводная схема измерений

Возможны схемы измерений, в которых истинное значение Z_X оценивается как по току, так и по падению напряжения на элементе (рис. 5.23).

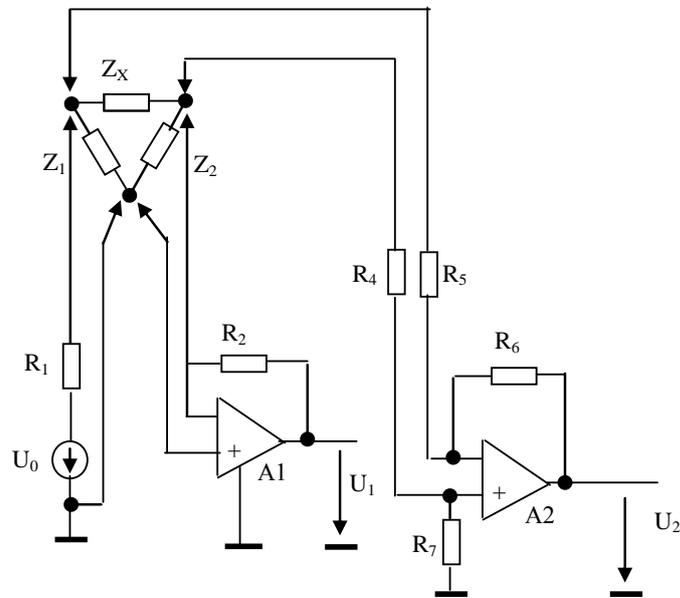


Рис. 5.23. Комбинированная шестипроводная измерительная схема

Схемы измерений для отдельных типов элементов

Приведенные выше измерительные схемы с небольшими доработками могут применяться для самых разных типов элементов.

Так, трехпроводная схема измерений, показанная на рис. 5.24, может использоваться для проверки транзисторов. Она позволяет измерять сразу два параметра: падение напряжения на переходе эмиттер–база и статический коэффициент передачи тока (коэффициент усиления $h_{21Э}$). Последний приблизительно можно оценить по формуле $h_{21Э} = U_2 R_1 / E_0 R_2$. Падение напряжения на переходе эмиттер – база определяется по значениям U_1 , R_1 и E_0 .

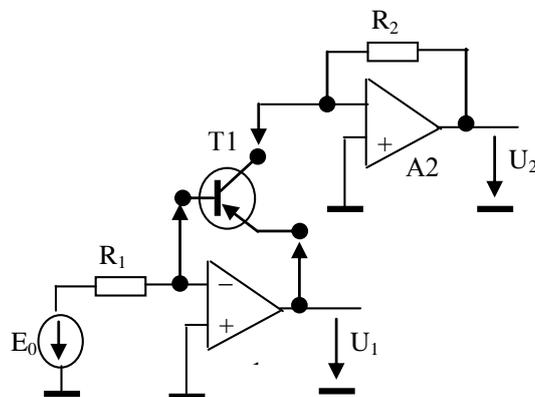


Рис. 5.24. Измерительная трехпроводная схема для проверки транзистора

В большинстве случаев об исправности диода можно судить по его сопротивлению в прямом и обратном направлении. Схемы, представленные на рис. 5.21, могут достаточно эффективно использоваться для проверки прямых и обратных токов диодов (рис. 5.25), которые определяют прямое и обратное сопротивление диода. В них, как уже пояснялось, влияние шунтирующих элементов Z_1 и Z_2 устранится за счет особенностей функционирования операционных усилителей. Схема рис. 5.25а позволяет оценить обратный ток диода. Однако если параллельно диоду подключен шунтирующий резистор небольшой величины, то возникает очень большая погрешность.

Вторая схема (рис. 5.25б) применяется для оценки падения напряжения на диоде при прямом включении. Обычно для оценки прямого падения напряжения с помощью источника U_0 и эталонного резистора R_0 создается некоторый ток и по вольтамперной характеристике диода определяют «исправное» ($U_{и}$) падение напряжения. При исправном диоде по результатам измерения должно получиться $U_{вых} \approx U_{и}$. Если бы не было сопротивления $R_{ш}$, то ток через диод $I_0 = E_0/R_0$. Однако наличие $R_{ш}$ приводит к тому, что часть тока ответвляется в $R_{ш}$. В результате $I_0 = I_{пр} + I_{ш}$, где $I_{пр}$ – ток через диод, $I_{ш}$ – ток через резистор $R_{ш}$. Поэтому $U_{вых}$ будет ниже, чем установленное по вольтамперной характеристике.

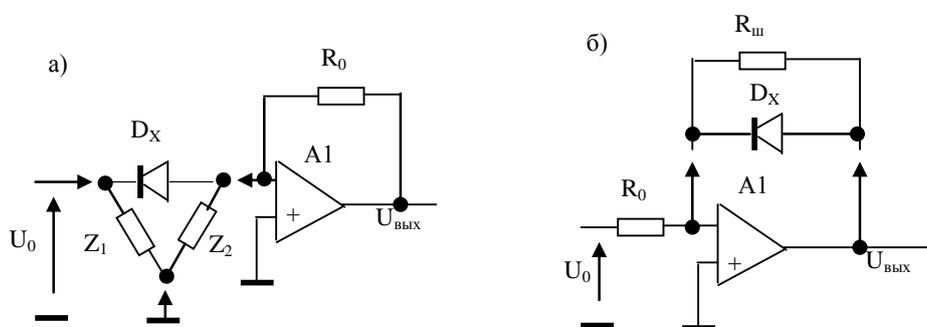


Рис. 5.25. Измерительные схемы для проверки диодов

Для проверки некоторых элементов (конденсаторы, индуктивности) надо подавать переменное или импульсное напряжение. При этом требуются, обычно, более сложные схемы измерений, содержащие преобразователи аналог-код, счетчики импульсов и др. Однако и в этих случаях широко применяются измерительные схемы на основе операционных усилителей.

Ниже приведена пятипроводная схема (рис. 5.26) для оценки емкости конденсатора C_X . При некоторых допущениях выходное напряжение в этой схеме можно принять равным $U_{вых} = 2\pi f_0 C_X R_2 K U_0$, где f_0 и U_0 – частота и амплитуда эталонного источника синусоидального напряжения, K – коэффициент, учитывающий влияние усилителя-преобразователя.

Внутрисхемное диагностирование применяется к печатным платам, в которых уже впаяны электронные компоненты. При этом, как правило, каждый проверяемый компонент является частью некоторой электрической цепи, другие

компоненты которой шунтируют и вносят искажения в процесс измерения. Чтобы снизить влияние этого фактора и снизить погрешности измерений, надо каким-то образом заблокировать или компенсировать влияние других компонент электрической цепи на токи, протекающие через проверяемый элемент. Схемы на рис. 5.21–5.26 позволяют устранить влияние компонент, подключенных между выводами проверяемого элемента и «землей». Однако возможны более сложные варианты шунтирования. Тогда делаются дополнительные закоротки, в заданные точки подаются компенсирующие напряжения.

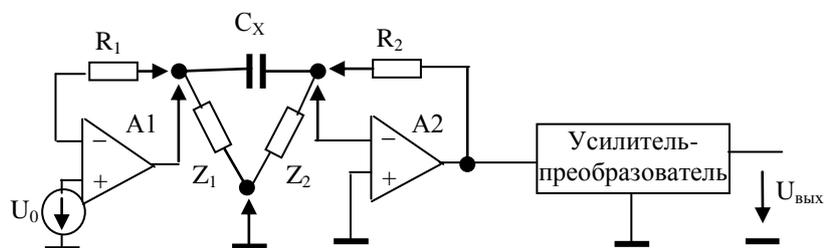


Рис. 5.26. Измерительная пятипроводная схема для определения емкости конденсатора

На рисунках 5.27 и 5.28 приведены примеры устранения влияния шунтирующих элементов в реальных схемах.

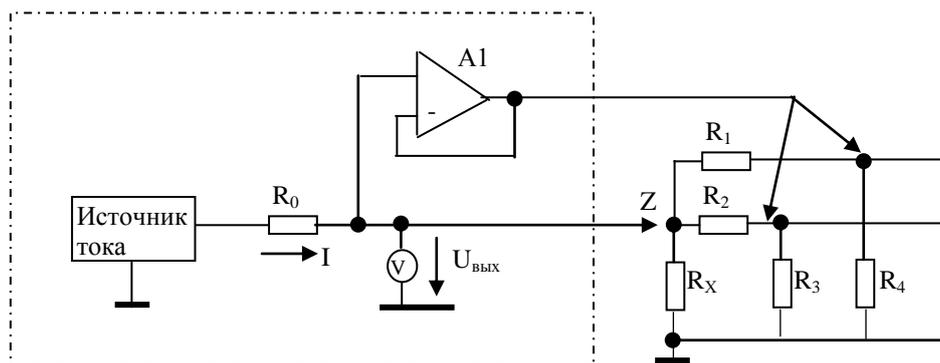


Рис. 5.27. Устранение влияния шунтирующих элементов путем подачи компенсирующего напряжения

На рисунке 5.27 показана схема подключений для измерения падения напряжения на элементе R_x . Часть элементов, входящая в состав собственно аппаратуры системы диагностирования, обведена пунктирной линией. Она содержит источник тока, создающий ток I . Этот ток полностью поступает в точку Z , так как операционный усилитель $A1$ имеет большое входное сопротивление. Если не принимать специальных мер, то ток I пойдет не только через R_x , но и через

R_1, R_2, R_3, R_4 . С помощью операционного усилителя А1 в точки соединений R_1 и R_4, R_2 и R_3 подается компенсирующее напряжение. Усилитель А1 работает как повторитель с глубокой обратной связью. Поэтому на его выходе напряжение такое же, как в точке Z. Таким образом, после подачи компенсирующего напряжения оба вывода резисторов R_1, R_2, R_3, R_4 окажутся под одним и тем же потенциалом, т.е. токов через них не будет. В результате весь ток I пойдет через R_X . Если ток I известен, то после измерения $U_{\text{вых}}$ можно определить значение R_X по формуле $R_X = U_{\text{вых}}/I$.

В схеме на рис. 5.28 измеряется значение резистора R_X в коллекторной цепи транзистора Т1. Измеряемый резистор включается в цепь обратной связи операционного усилителя А1. В данном случае использована пятипроводная схема измерения. Два провода используются для подключения к выводам резистора R_1 , остальные три – для устранения влияния на результат измерения шунтирующих элементов и защиты транзистора Т1 от электрических повреждений. При закорачивании вывода резистора R_2 на «землю», оба его вывода оказываются под нулевым потенциалом и ток через резистор не идет.

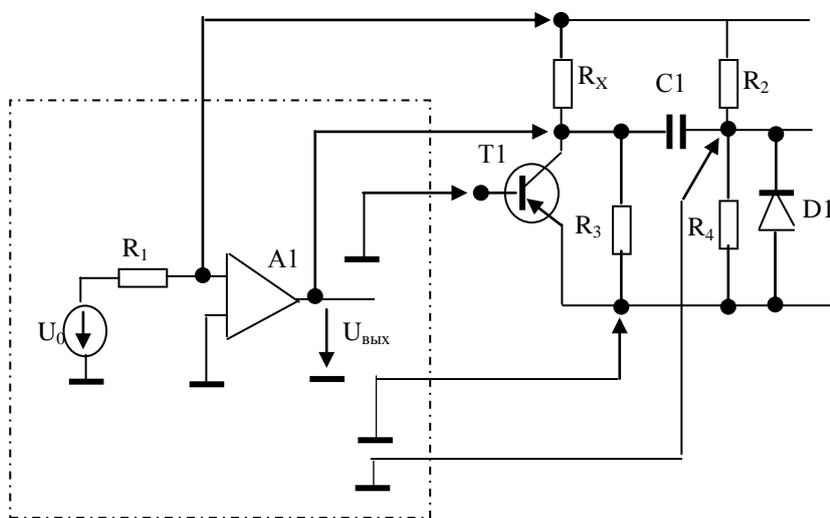


Рис. 5.28. Устранение влияния шунтирующих элементов и защита транзистора путем подключения к шине «земля»

При подключении нижнего вывода резистора R_3 к «земле», он начинает работать как нагрузка на выход усилителя А1 и не влияет на ток в R_X . Таким образом, можно считать, что при указанных на рис. 5.28 закоротках

$$R_X = -U_{\text{вых}} R_1 / U_0.$$

Подключение базы транзистора Т1 к «земле» позволяет защитить полупроводниковые переходы транзистора от электрического пробоя.

Многие электронные ОД содержат цифровые микросхемы. Внутрисхемное диагностирование предполагает выполнение проверки, не выпаивая микросхемы

из платы. Для этого на контакты тестируемой микросхемы подаются мощные кратковременные импульсы, способные установить заданный уровень логического сигнала независимо от логического состояния соединенных с этим выводом других микросхем. Проверяемая микросхема подключается к тестеру при помощи специального тестового зажима (клипсы), имеющего вид прищепки, который надевается сверху на корпус микросхемы (рис. 5.29). Величина и длительность импульса тока, подаваемого на контакт микросхемы, должна быть достаточной для принудительной установки выхода цифровой микросхемы в заданное состояние и, в то же время, не выводить ее из строя. Длительность подачи импульса зависит от технологии изготовления микросхем. Существуют стандарты, определяющие допустимые параметры импульсных воздействий для такого диагностирования.



Рис. 5.29. Подключение к микросхемам с помощью разъема типа «клипса»

5.6. Автоматизированные системы поэлементного диагностирования

Промышленностью выпускается большое число разнообразных систем, специально предназначенных для внутрисхемного диагностирования. В зависимости от назначения и условий применения (серийное производство, ремонт в условиях сервисного центра и др.) они могут различаться по габаритам и конструкции, производительности, стоимости и функциональным возможностям, степени автоматизации, программным обеспечением и др.

В этом разделе мы рассмотрим некоторые из характерных представителей систем, используемых для автоматизации процессов поэлементного диагностирования.

Стационарные системы с адаптером типа «ложе из гвоздей»

Стационарные системы имеют довольно большие габариты, обычно применяются при серийном производстве, обладают высокой производительностью и разнообразными функциональными возможностями.

Структуру и принцип действия подобных систем поэлементного контроля рассмотрим на примере стационарной отечественной системы 1013, серийно выпускавшейся еще в начале 80-х годов прошлого века. Структура ее аппаратных средств представлена на рис. 5.30. Система обеспечивает контроль проводников объекта диагностирования (ОД), контроль параметров отдельных элементов, контроль на соответствие таблице напряжений, контроль токов, потребляемых ОД по шинам питания.

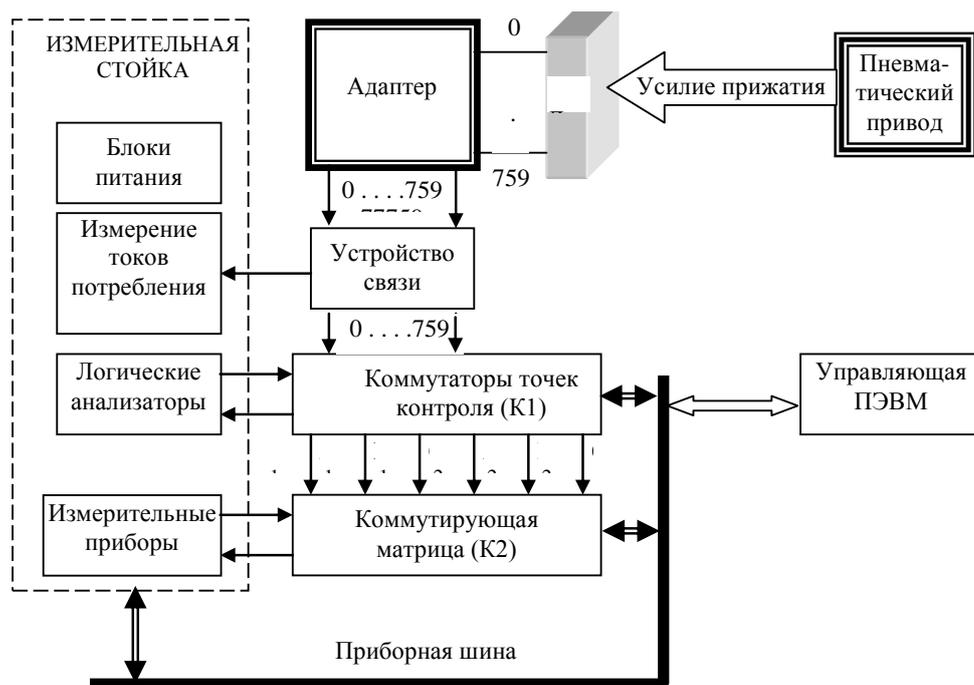


Рис. 5.30. Структура аппаратной части системы 1013

Адаптер системы представляет собой плату, на которой с некоторым шагом установлены игольчатые контакты (т.е. используется контактное устройство типа «ложе из гвоздей»). Печатная плата ОД укладывается сверху на эти контакты. Пневматический привод развивает необходимое усилие для прижатия контролируемой печатной платы к игольчатому полю адаптера, который с помощью устройства связи подключается к входам коммутатора контрольных точек К1. Программно управляемый коммутатор К1 через контакты адаптера коммутирует соответствующие точки измерений к необходимым входам А1, В1, С1, А2, В2, С2 коммутирующей матрицы К2, что позволяет реализовать многопроводные

схемы подключения к ОД. Коммутирующая матрица К2 обеспечивает подключение выводов контролируемых элементов и других точек платы ОД к измерительной схеме, создаваемой для измерений. Количество входных каналов коммутатора К1 может достигать 8192, а количество выходов обычно не менее 6, чтобы можно было организовать шестипроводные схемы измерений. Кроме того, через коммутатор К1 к шинам питания ОД подключается измеритель токов питания. Управление системой осуществляется ПЭВМ с помощью комплекта прикладных программ через приборную шину.

Измерительная часть системы выполнена в виде измерительной стойки, в которой установлены измерительные приборы, блок управления для включения необходимых режимов работы и отображения текущих результатов контроля. Для обеспечения большинства видов измерений в качестве измерителя в системе применялся серийный цифровой вольтметр В2 – 32/1.

Диагностирование ОД может осуществляться в двух режимах. В первом (без подачи питания на ОД) проверяется топология, сопротивление утечки между проводниками, измеряются значения сопротивлений, емкостей и индуктивностей, параметры диодов, стабилитронов и транзисторов. Во втором режиме (с подачей питания на ОД) проверяется работоспособность микросхем, проводятся измерения напряжений (в том числе на соответствие заданной таблицы напряжений в указанных контрольных точках), измеряются токи потребления.

При измерении параметров R, L, C как на переменном, так и на постоянном токе выходные напряжения измеряются вольтметром, а ЭВМ вычисляет контролируемые параметры. Повышение точности измерения на постоянном токе обеспечивается измерением «нуля» источника рабочего напряжения с соответствующей автоматической коррекцией данных путём программирования: проведения двух измерений с изменением полярности рабочего напряжения с расчётом среднего значения результата измерения, что исключает влияние паразитных напряжений постоянного тока в цепях и уход «нуля» схемы измерения.

Программа контроля составляется пользователем на специализированном языке СТ-13, ориентированном на внутрисхемный контроль.

Помимо программных средств подготовки тест-программ и управления диагностированием, система 1013 обладает программной самодиагностикой. Исполнение программ самодиагностики осуществляется подключением к системе вместо адаптера устройства самоконтроля, в состав которого включены необходимые средства, обеспечивающие метрологический контроль системы и проверку работоспособности системы и её составных частей.

Одной из сложных задач при разработке систем внутрисхемного диагностирования остаётся обеспечение надёжного контактирования с ОД. В системе 1013 использовано контактное приспособление, называемое «ложом из гвоздей». Оно представляет собой плату, на которой установлены подпружиненные игольчатые штырьки. Их наконечники прижимаются к контактным площадкам проводников проверяемой печатной платы, через них провода измерительной схемы подключаются к выбранным точкам на проверяемой печатной плате.

Существуют адаптеры, в которых контролируемую плату к неподвижным подпружиненным штырям прижимают рычагом с механическим приводом. Однако более предпочтительным является использование вакуумного механизма

прижатия. В них ручной механический привод заменён пневматическим, что позволяет достаточно легко регулировать усилие прижатия, регулируя давление воздуха. Для хорошего контакта требуется давление 120–160 г на один контактный штырь и соответствующее давление воздуха. Применение пневматического привода дороже, но позволяет уменьшить вероятность повреждений «ложе из гвоздей», которое может произойти при ручном механическом механизме прижатия.

Рисунок 5.31 иллюстрирует типичный механизм прижатия печатной платы ОД к игольчатым контактам адаптера с помощью пневматического привода. Проверяемая плата удерживается в контакте с «ложем из гвоздей» при помощи вакуумной системы, откачивающей воздух из адаптера.

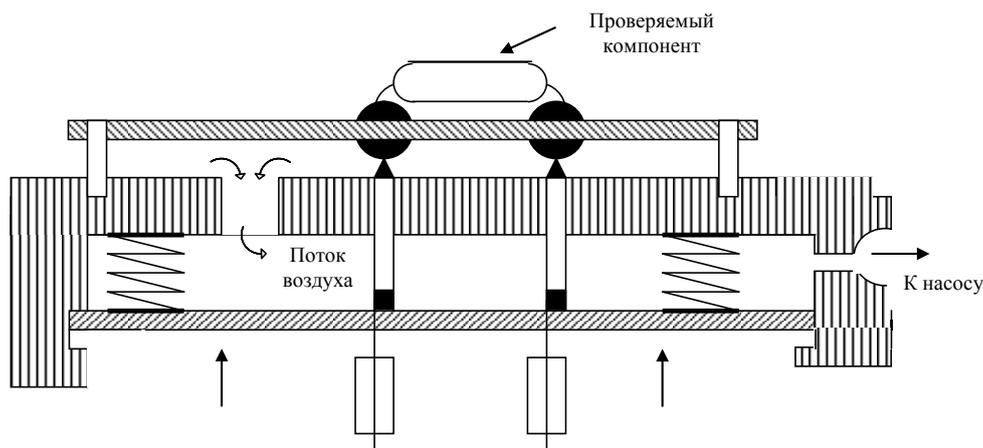


Рис. 5.31. Механизм прижатия печатной платы ОД к контактам адаптера

Контактирование с нужными точками ОД обеспечивается подпружиненными штырями, которые являются одним из ответственных компонентов адаптера. Каждый штырь осуществляет физический и электрический контакт с некоторой точкой печатной платы ОД и через него информационный сигнал с платы поступает в измерительную систему.

Штыри приходится периодически менять, поэтому они имеют специальную конструкцию, облегчающую этот процесс (рис. 5.32). Основными элементами конструкции являются: корпус, в котором двигается плунжер с наконечником. Чтобы обеспечить мягкое прижатие наконечника к контрольной точке, в корпусе имеется пружина, прижимающая наконечник и обеспечивающая необходимый рабочий ход плунжера. На плате адаптера устанавливают гильзы, в которые и вставляется корпус штыря. Для замены надо просто вытащить из гильзы старый штырь и вставить новый.

На рисунке 5.32 наконечник имеет два зубца, что обеспечивает надежную фиксацию контакта, однако возможны и другие формы наконечников.

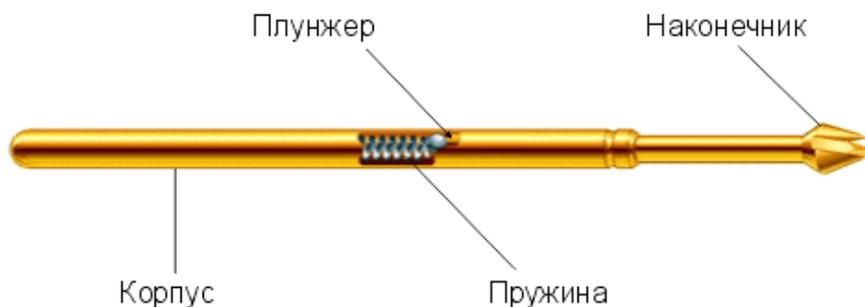


Рис. 5.32. Конструкция штыря

В зависимости от особенностей пайки и типов установленных элементов контактная часть штырей может иметь разную форму. На рисунке 5.33 (а-е) показаны различные формы наконечников.

Наконечники в виде гвоздя (а, д) могут прокалывать защитное лаковое покрытие; пилообразные (б) – создают довольно большую поверхность контактирования; выпуклые (в) – используют в случаях, когда в пайке или выводе элемента имеются полости; вогнутые (г) – хороши при округлённых пайках; цилиндрические (е) – обычно применяют для подключения к плоским контактным площадкам.

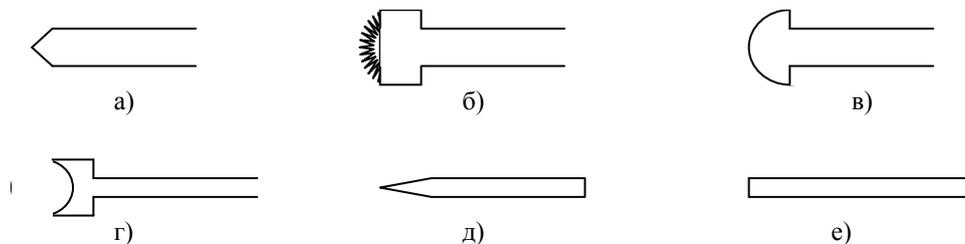


Рис. 5.33. Формы наконечников контактных штырей

Диаметр штыря обычно зависит от шага координатной сетки диагностируемой печатной платы. Он составляет 1,0 мм и менее. Диаметр гильз определяет максимальное число штырей, которые можно установить на плате адаптера. Чем тоньше штыри и гильзы, тем в большем количестве их можно установить на единице поверхности. В универсальном варианте адаптера на его контактной плате («ложе из гвоздей») устанавливают максимально возможное число гильз и штырей с учетом координатной сетки печатной платы ОД. Однако при серийном производстве, как правило, для каждого ОД делают отдельную плату адаптера, на которой штыри находятся только в тех местах, в которых надо будет делать измерения. Такой адаптер дешевле и надежнее.

Система внутрисхемного диагностирования TestStation TS12X фирмы TERADYNE (США) также относится к стационарным (рис. 5.34). Для доступа к

диагностируемому объекту в тестере используется адаптер «ложе из гвоздей» (bed-of-nails). В адаптере контактная плата со штырями является сменной. Вся необходимая информация для ее изготовления генерируется в автоматическом режиме при разработке тестовой программы.



Рис. 5.34. Внешний вид системы TestStation TS12X

Система обладает расширенными возможностями. Кроме обычных возможностей внутрисхемного диагностирования система обеспечивает:

- функциональное тестирование узлов и изделия в целом,
- использование технологии «backdriving» для проверки цифровых микросхем (путем кратковременной подачи импульсов тока),
- безопасное тестирование КМОП компонентов,
- внутрисхемное программирование FLASH, EEPROM и контроллеров в оперативном режиме,
- тестирования методом периферийного сканирования (Boundary Scan),
- мониторинг с удаленных рабочих станций результатов тестирования в реальном режиме времени (производительность, выход годного, загрузка тестера),
- возможность интеграции в автоматическую производственную линию.

Тестовые программы генерируются с помощью программного пакета ATG Xpress, исходя из данных САПР.

Системы с подвижными зондами

На рисунке 5.35 показан внешний вид стационарных установок (тестеров) внутрисхемного диагностирования с подвижными зондами, применяемых в условиях серийного производства. В адаптерах таких систем не нужны контактные платы с большим числом штырей.



Рис. 5.35. Тестеры S40 Pilot и S40 Pilot LX с подвижными пробниками

В тестерах S40 PILOT проверяемая печатная плата устанавливается в специальный адаптер, который имеет несколько подвижных тестовых головок с пробниками и с приводами по осям X, Y и Z (рис. 5.36 а). Тестовые головки (обычно четыре) могут перемещаться над испытываемой платой в нужное место с довольно большой скоростью. В головках находятся металлические иглы – электрические зонды («подвижные пробники»), которые вступают в контакт с переходными отверстиями и контактными выводами объекта диагностирования (рис. 5.36 б). Таким образом обеспечивается доступ к контрольным точкам диагностируемого изделия. Шаг между соседними контактными площадками – до 0,4 мм.

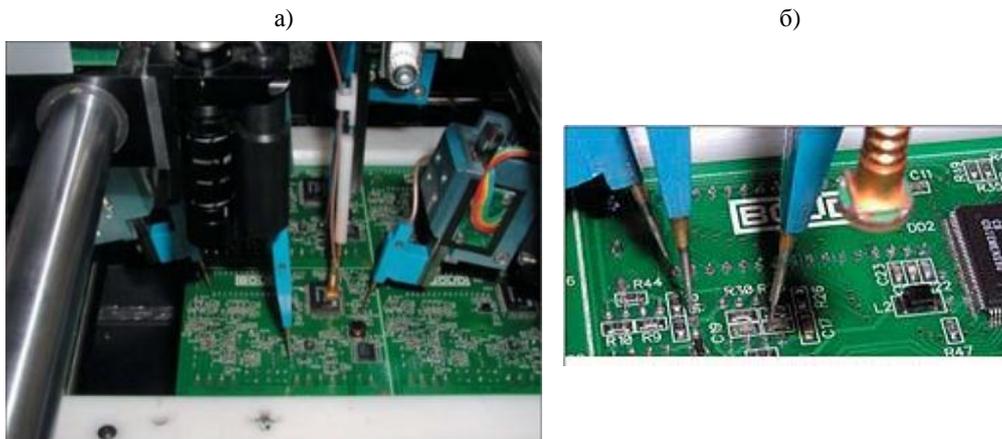


Рис. 5.36. Система подвижных пробников тестера S40 PILOT:
а) общий вид, б) зонды

Помимо четырех подвижных пробников, осуществляющих съем сигналов с верхней стороны платы, предусмотрена возможность установки фиксированных

пробников (до 1024), обеспечивающих контактирование с контрольными точками, расположенными с нижней стороны тестируемой платы.

Как и большинство современных стационарных систем, тестер S40 Pilot допускает интеграцию в процесс тестирования дополнительных внешних приборов, управляемых через интерфейсы IEEE, RS-232C, PCI и др.

Тестер позволяет устанавливать под нижней стороной печатной платы ОД зонды с подпружиненными штырями, которые для фиксации своего положения имеют магнитную подставку (рис. 5.37).

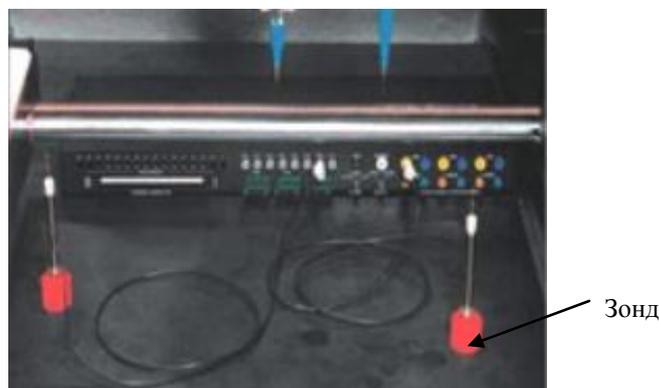


Рис. 5.37. Зонды на магнитной (красной) подставке



Рис. 5.38. Окно программы при диагностировании в режиме on-line

В тестовой системе S40 Pilot имеется видеосистема с подвижной видеокамерой, что позволяет оператору контролировать наличие, правильность ориентации и полярность установленных на тестируемых платах компонентов (интегральных микросхем, электролитических конденсаторов и т.д.). С помощью этой видеосистемы и специальных языковых средств можно управлять позиционированием подвижных пробников при диагностировании в режиме реального времени. Программируя место точек установки подвижных зондов на мониторе оператора, можно создать окно с изображением местоположения подвижных пробников (рис. 5.38).

Настольные системы внутрисхемного диагностирования

Достаточно большую часть рынка систем внутрисхемного диагностирования составляют настольные системы. Они эффективны и экономически оправданы для применения в сервисных и ремонтных центрах или при мелкосерийном производстве. Их современные варианты позволяют осуществлять не только традиционное внутрисхемное диагностирование, но и выполнять функциональные проверки ОД, диагностировать объекты с большим количеством цифровых входных/выходных каналов, содержащих микропроцессоры и элементы памяти.

На рисунке 5.39 приведена система SFL1500, позволяющая тестировать цифровые и аналоговые платы без демонтажа компонентов и без включения питания платы. Ее особенностью является использование метода **аналогового сигнатурного анализа** для проверки элементов.



Рис. 5.39. Настольный тестер SFL1500

Суть метода заключается в следующем. Проходящий через элемент ток зависит от приложенного напряжения и электрических характеристик этого компонента. График изменения тока от прикладываемого напряжения (вольтамперная характеристика) индивидуален для каждого компонента и может использоваться для оценки технического состояния компонента. Таким образом, техническое состояние элементов можно определять, сравнивая полученные в ходе диагностирования вольтамперные характеристики с эталонными (которые получены заранее на заведомо исправном ОД). Такой метод определения технического состояния элемента называют «**аналоговый сигнатурный анализ**», а саму вольтамперную характеристику, характеризующую проверяемый элемент – **сигнатурой**.

Например, сигнатура для чистого сопротивления – прямая линия, так что изменение ее наклона от угла наклона сигнатуры аналогичного устройства на

заведомо исправной плате указывает на неправильное значение сопротивления. Конденсатор имеет эллиптическую сигнатуру, ее диаметр зависит от емкости конденсатора. Неправильная величина емкости приводит к изменению диаметра эллипса, наличие утечки проявляется в наклоне эллипса (рис. 5.40а). Микросхемы содержат полупроводниковые элементы и их сигнатуры зависят от типа микросхемы (аналоговая, цифровая), диапазона изменения амплитуд входных сигналов (рис. 5.40б).

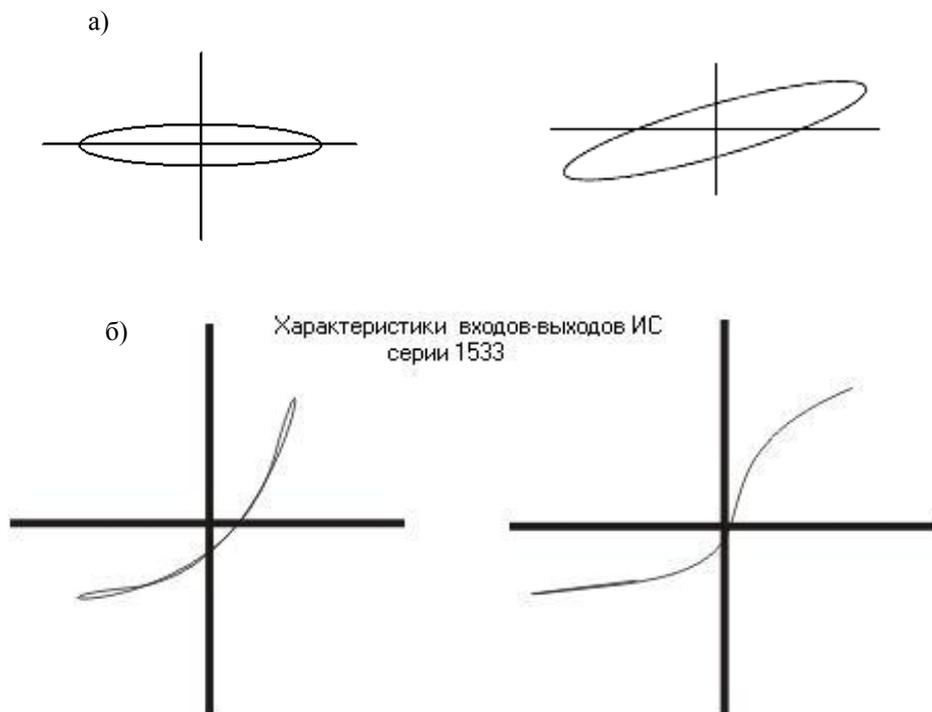


Рис. 5.40. Формы сигнатур при проверке конденсатора (а) и микросхемы (б)

Тестер SFL1500 вручную с помощью специальных зондов (щупов) подключают к выбранным точкам на диагностируемой плате. Через них система подает сигналы безопасного уровня и частоты, необходимые для построения сигнатуры, измеряет и выводит в графическом виде на цветной жидкокристаллический экран вольтамперную характеристику (сигнатуру) анализируемой цепи, которая сравнивается с эталонной. На проверяемый объект подаются импульсы с изменяющейся амплитудой. Их частота, диапазоны изменения амплитуды программируются. Хотя прибор имеет всего два канала для приема и отображения сигналов, но, используя мультиплексор, можно снимать сигналы с многоконтактных внешних разъемов печатных плат (до 128 каналов).

Эталонная сигнатура берется из специально создаваемой базы данных эталонных сигнатур по измерениям в заведомо исправном объекте. Для хранения

эталонных сигнатур прибор имеет встроенную память. Возможна работа с базой сигнатур, находящейся на внешней Flash-памяти.

Наклоны сигнатур, диаметры эллипсов зависят от номинала установленных на плате элементов, при любой неисправности сигнатуры резко меняют форму, поэтому процесс определения неисправности довольно прост и сводится к сравнению сигнатур, получаемых от проверяемого изделия, с эталонным. Сравнение выполняется автоматически, т.е. прибор сообщает результат сравнения (ГОДЕН/НЕ ГОДЕН). Порог различия (допуск на отклонение) сигнатур задается пользователем тестера.

На рисунке 5.41 показана настольная система универсального назначения QT200.

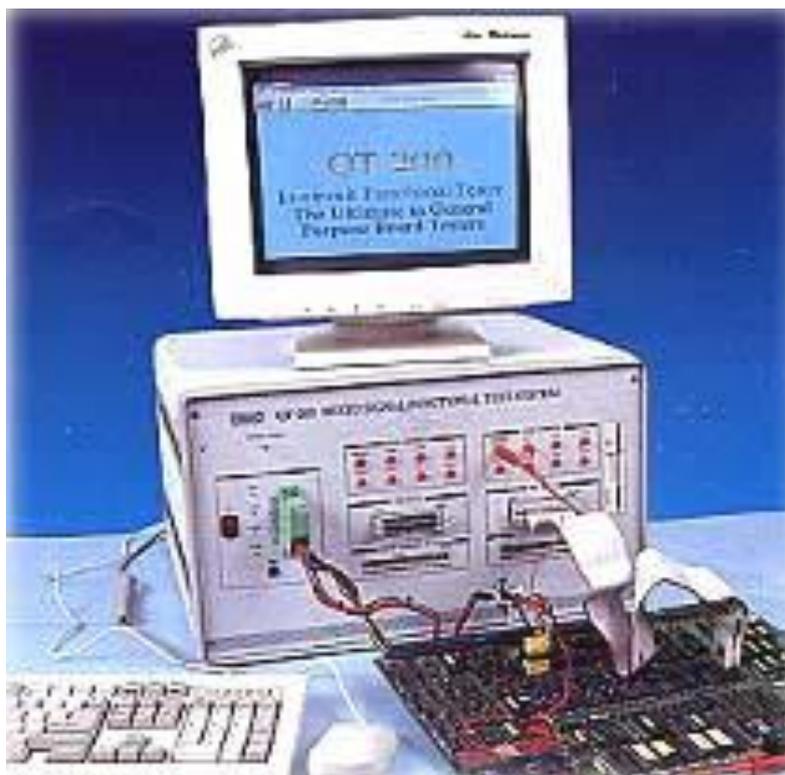


Рис. 5.41. Настольная система внутрисхемного диагностирования QT200

Система позволяет проводить внутрисхемное диагностирование многоконтактных цифровых микросхем путем подачи кратковременных мощных импульсов, аналоговый сигнатурный анализ всех типов элементов, проверять токи утечки, находить короткие замыкания и обрывы. Имеется большая библиотека сигнатур (эталонных значений) для стандартных тестовых воздействий аналоговых и цифровых микросхем, транзисторов и диодов. Для проверки элементов система может подавать разные по форме и фазе напряжения (синусоидальное, прямоугольное, треугольное, импульсное).

В настольных системах внутрисхемного диагностирования имеется специальный режим предварительного определения эталонных сигнатур на заведомо исправном ОД и формирования на основе проведенных измерений программной библиотеки сигнатур.

Современные системы строятся по принципу конструктора. Они могут содержать сменные блоки, позволяющие дополнить стандартные способности систем внутрисхемного диагностирования (определять параметры резисторов, конденсаторов, транзисторов, диодов или индуктивностей) возможностью подавать и снимать цифровые сигналы, выполнять аналоговый сигнатурный анализ, организовывать внутрисхемный контроль цифровых микросхем, подавая кратковременные мощные импульсы тока, проверять микросхемы памяти и др. К подобным системам, в частности, относятся отечественные системы «ТЕСТ-Д» и «АСК-В1» (рис. 5.42, 5.43).

Автоматизированная система диагностики «ТЕСТ-Д» предназначена для диагностики и ремонта промышленных электронных устройств, в частности печатных плат для станков с программным управлением. Система представляет собой управляемый от персонального компьютера диагностический стенд, к которому при помощи специальных переходных адаптеров можно подключить различные электронные модули. На рисунке 5.42 показана конфигурация системы в режиме снятия вольтамперных характеристик.



Рис. 5.42. Система «ТЕСТ-Д» с модулем внутрисхемного диагностирования

Система АСК-В1 (рис. 5.43) создавалась для диагностирования при производстве и ремонте радиотелевизионной аппаратуры.



Рис. 5.43. Автоматизированная система внутрисхемного контроля «АСК-В1»

Осциллографы

Следует сказать еще об одном виде измерительных приборов, традиционно используемых в наиболее сложных случаях, когда возникает необходимость детально изучить изменение сигналов во времени. Это **осциллографы**.

Даже сегодня он обязательный атрибут ремонтной мастерской и разработчика новых электронных схем.

Осциллограф служит для точного отображения на экране изменения во времени напряжения или тока в заданной точке или одновременно в нескольких точках (обычно в 2-х или 4-х).

Современные осциллографы не только измеряют и отображают аналоговые сигналы, но и осуществляют их цифровую обработку, запоминают результаты измерений в цифровой форме, имеют интерфейсы для взаимодействия с компьютером и с локальной вычислительной сетью, разъемы подключения к принтеру и Flash-памяти.

Внешний вид одного из осциллографов фирмы Tektronix показан на рис. 5.44.

Осциллографы серии DPO 4000 имеют до 4-х входных измерительных каналов, полосу пропускания до 1ГГц, частоту квантования входных аналоговых сигналов до 5 ГГц. Прибор позволяет определять и отображать до 25 различных характеристик сигналов: временных, амплитудных, статистических. Результаты измерения переводятся в цифровой код и могут запоминаться. Стандартная длина записи в осциллографах серии DPO4000 составляет 10 млн выборок.



Рис. 5.44. Передняя панель осциллографа серии DPO 4000

Результаты измерений и обработки отображаются на экране. Современные осциллографы содержат микропроцессор, отображают изображения в цвете, имеют довольно большие по размеру экраны. Так, осциллографы серии DPO 4000 имеют жидкокристаллический экран с диагональю более 26 см (10,4 дюйма) и разрешением XGA (1024 x 768 точек).

В схемах современных электронных изделий часто предусматриваются последовательные шины для передачи/приема цифровых кодов. В последовательной шине (содержащей не более 2–4-х информационных каналов – проводов) каждый отдельный пакет содержит адрес, сигналы синхронизации и управления и непосредственно данные. Некоторые осциллографы снабжаются аппаратно-программными средствами, позволяющими регистрировать, анализировать и отображать сигналы на таких шинах. Так, функциональные возможности осциллографов серии DPO4000 позволяют:

- с помощью дополнительных модулей выполнять запуск по содержимому пакета на последовательной шине данных;
- выполнять автоматическое декодирование пакетов и отображение данных в доступной и понятной форме;
- выполнять поиск нужного участка с данными и отмечать отдельные события в информации, передаваемой по последовательной шине.

Осциллографы снабжаются самыми разнообразными функциями анализа сохраненных результатов первичных измерений, которые могут иметь большой объем. Так, ручки управления, показанные в правой части рис. 5.44, позволяют изменять масштаб показа и вести «прокрутку» осциллограмм. При этом на экране осциллографа показывается место выделенного и отображаемого участка в исходной временной последовательности (рис. 5.45).

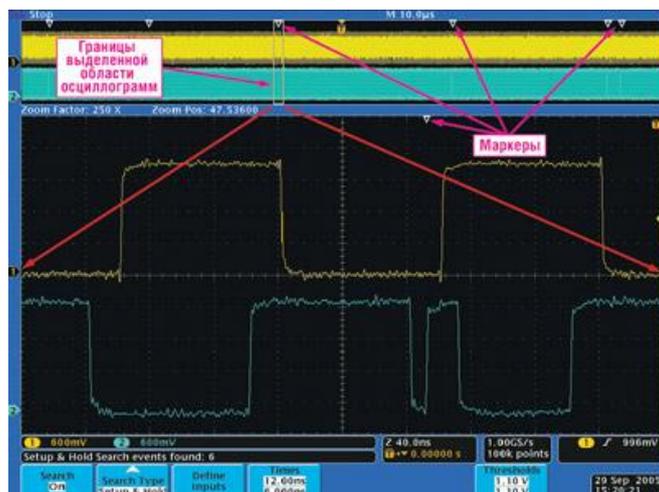


Рис. 5.45. Экран осциллографа в режиме «прокрутки» и просмотра диаграмм

Соответствующие функции осциллографа позволяют осуществлять автоматический поиск событий с заданными пользователем параметрами (например, при пересечении положительным фронтом заданного уровня напряжения) или пакетов данных с заданной адресной или управляющей информацией. При обнаружении событий в заданных точках автоматически устанавливаются маркеры, которые пользователь может видеть в ходе «прокрутки».

Для компьютерного анализа архивов измерений и удаленного обмена осциллографы снабжаются специальными программными средствами.

Контрольные вопросы

1. Классификация систем внутрисхемного контроля по способу подключения к объекту диагностирования.
2. Почему стрелочные омметры имеют нелинейную градуировку шкалы, а цифровые – линейную?
3. Как сопротивление стрелочных вольтметров и амперметров влияет на погрешность измерений?
4. Нарисуйте и поясните схему фильтра, который задерживает постоянную составляющую сигнала и пропускает переменную.
5. Нарисуйте и поясните схему фильтра, который задерживает переменную составляющую сигнала и пропускает постоянную.
6. Нарисуйте и поясните схему фильтра, который выделяет отрицательную полуволну синусоидального сигнала, нарисуйте график выходного сигнала для предложенного фильтра.
7. Какая из емкостей 1000 пф или 1 мкф оказывает большее сопротивление переменному току?
8. Какая из индуктивностей 10 мГн или 1 мГн оказывает большее сопротивление переменному току?
9. Первичная обмотка трансформатора содержит 10000 витков, а вторичная – 250. Какой коэффициент трансформации?

10. Пусть при измерениях сопротивлений переходов БЭ и БК на базу транзистора подается положительное напряжение. Какое сопротивление покажет прибор для сопротивлений переходов (высокое или низкое)?
11. Дайте определение понятия погрешности.
12. В чем отличие прямых измерений от косвенных?
13. Запишите формулы абсолютной и относительной погрешности.
14. Какая погрешность называется приборной?
15. В чем различие инструментальной и методической погрешностей?
16. Пусть производятся независимые измерения величин А и К. Как будут выглядеть формулы абсолютной и относительной погрешностей для суммы этих измерений (если известны абсолютные погрешности измерений величин А и К)?
17. Как устроены адаптеры типа «ложе из гвоздей»? Их основные конструктивные элементы.
18. В схемах измерений на основе операционного усилителя рекомендуется при измерении малых сопротивлений измеряемый элемент подключать вместо резистора обратной связи. Почему?
19. В схемах измерений на основе операционного усилителя рекомендуется при измерении больших сопротивлений измеряемый элемент подключать вместо входного резистора. Почему?
20. Зачем используются многопроводные схемы подключений к объекту диагностирования в системах поэлементного диагностирования?
21. В чем сущность метода аналоговых сигнатур?

Рекомендуемая литература

Байда, Н.П. Микропроцессорные системы поэлементного диагностирования РЭА / Н.П.Байда, И.В.Кузьмин, В.Т.Шпилевой. – М.: Радио и связь, 1987. – 256 с.

Карр, Дж. Диагностика и ремонт аппаратуры радиосвязи и радиовещания / Дж. Карр; пер. с англ. . – М., 1991. – 400 с.

Лихтциндер, Б.Я. Внутрисхемное диагностирование узлов радиоэлектронной аппаратуры / Б.Я. Лихтциндер. – Киев: «ТЭХНИКА», 1988. – 168 с.

Мидлтон, Р.Г. Наладка и ремонт радиоэлектронных устройств, не имеющих технического описания / Р.Г. Мидлтон; пер. с англ.; под ред. Ф.Н. Покровского. – М.: Энергоатомиздат, 1994. – 304 с.

www.compitech.ru

www.izme.ru

www.pcbfab.ru

www.radvam.hut1.ru

www.skordon.ru

www.sovtest.ru

www.techno.ru

Тема 6. ОРГАНИЗАЦИЯ ДИАГНОСТИРОВАНИЯ И ОБСЛУЖИВАНИЯ МИКРОПРОЦЕССОРНЫХ УСТРОЙСТВ

6.1. Особенности микропроцессора как объекта диагностирования

Сегодня кажется уже нет такой области применения, где бы не использовались электронные устройства с микропроцессором.

Микропроцессор – это программно-управляемое устройство, работа которого заключается в последовательном выполнении некоторых команд. Поэтому построение теста для него, по сути, представляет собой процесс формирования некоторой программы, а контроль – процесс выполнения этой тестовой программы. Ее правильное исполнение, означает исправность (работоспособность) микропроцессора.

Электрическая схема даже простейшего микропроцессора состоит из тысяч-десятков тысяч логических элементов с большим числом элементов памяти. Уже поэтому задачи диагностирования объектов, содержащих микропроцессор, относятся к наиболее сложным в технической диагностике.

К числу **основных особенностей**, которые надо учитывать при организации диагностирования таких объектов, относят:

- высокую сложность электрических схем, содержащих большое число недоступных для прямого контроля элементов памяти и обратных связей;
- магистральный принцип работы с шинами большой разрядности;
- использование микросхем и модулей памяти большого объема (десятки-сотни мегабайт);
- высокие рабочие частоты (сотни-тысячи мегагерц);
- исполнение микропроцессора и некоторых других микросхем программно-управляемых объектов в виде СБИС с большим числом выводов (до нескольких сот);
- использование микросхем памяти с возможностью перепрограммирования;
- выполнение схемы объекта на одной печатной плате (в некоторых случаях блок питания, устройства ввода, вывода и отображения информации могут выполняться в виде отдельных конструктивных узлов);
- отсутствие подробной и понятной информации о структуре и функционировании микропроцессора и других СБИС.

При диагностировании программно-управляемых объектов наиболее широкое распространение получил принцип раскрутки, или принцип расширяющихся областей. Диагностирование по принципу раскрутки предполагает, что первоначально некоторая часть оборудования (называемая ядром или системным ядром) должна быть в заведомо работоспособном состоянии. В него обычно входят блок питания, собственно микропроцессор, системный генератор синхронизации, системная шина. Диагностирование выполняется в несколько этапов, на каждом из которых проверяется очередная часть оборудования объекта. Сначала ядро проверяет аппаратуру первой области, затем проверяется аппаратура второй области с использованием ядра и уже проверенной первой области и т.д. Реализация принципа раскрутки предполагает использование встроенных средств самодиагностирования.

Типичным примером объекта, при организации диагностирования которого используется принцип раскрутки, является персональная ЭВМ. С целью обеспечения быстрого выявления отказов и высокой вероятности правильного функционирования современные ЭВМ содержат мощные средства встроенного самоконтроля, охватывающие контролем практически все действия компьютера. Этими средствами контроль выполняется на нескольких уровнях.

Выделяют четыре уровня контроля:

логический,
функциональный,
системный,
пользовательский.

На логическом уровне выполняется контроль правильности выполнения простейших операций: переполнение, деление на ноль, правильность чтения/записи отдельных байт и слов и т.п. Контроль на логическом уровне ведется непрерывно, поэтому он реализуется с помощью простых и быстродействующих схем.

На функциональном уровне контролируется, например, корректность протоколов обмена по внутренним шинам, правильность передачи блоков памяти путем использования специальных методов кодирования, превышение времени выполнения машинных циклов с помощью таймеров и т.п.

На системном уровне контролируется корректность протоколов обмена с внешними устройствами, корректность форматов данных и команд, нарушения защиты памяти, обращения к неиспользуемой или несуществующей области памяти и т.п.

Пользовательский уровень предполагает контроль на допустимость входных и выходных данных, использование двойного счета для проверки правильности вычислений и т.п.

Знание возможностей и особенностей функционирования встроенных средств контроля объекта может быть очень полезным при создании тестов.

Проектирование программно-управляемых (микропроцессорных) устройств сегодня ведется с учетом требований контролепригодности, так как в противном случае не удастся обеспечить нужный уровень надежности. Существуют даже специальные стандарты такого проектирования.

Вышеуказанные особенности программно-управляемых устройств инициировали создание специальных средств диагностирования, которые будут рассмотрены ниже. Они могут регистрировать многоуровневые потоки данных, работать на высокой частоте, представлять результаты диагностирования в компактной и удобной для визуального контроля форме, производить отладку вновь создаваемых программ, имитировать работу блоков памяти и собственно микропроцессора, а также взаимодействие с внешними устройствами.

6.2. Общие принципы организации технического обслуживания ПЭВМ

Понятие технического обслуживания

Техническое обслуживание (ТО) – это комплекс операций или операция по поддержанию работоспособности или исправности изделия при использовании по назначению, ожидании, хранении и транспортировании (по ГОСТ 18322-78).

Техническое обслуживание включает регламентированные в конструкторской документации операции для поддержания работоспособности или исправности изделия в течение его срока службы. Под **операцией технического обслуживания** понимают законченную часть технического обслуживания изделия, выполняемую на одном рабочем месте. В техническое обслуживание могут входить мойка изделия, контроль его технического состояния, очистка, смазывание, калибровка, крепление соединений и т.п.

Объект диагностирования и необходимые для организации и проведения обслуживания средства, документация, персонал и др. образуют систему технического обслуживания и ремонта.

Система технического обслуживания – совокупность взаимосвязанных средств, документации технического обслуживания и ремонта и исполнителей, необходимых для поддержания и восстановления качества изделий, входящих в эту систему (по ГОСТ 18322-78).

Техническое обслуживание характеризуется некоторой **периодичностью**, т.е. интервалом времени или наработки между данным видом технического обслуживания (ремонта) и последующим таким же видом или другим большей сложности.

Стандарт ГОСТ 18322-78 выделяет различные **виды и методы технического обслуживания**. Так, по видам техническое обслуживание возможно:

- при использовании,
- при ожидании,
- при хранении,
- при транспортировании,
- периодическое,
- сезонное,
- в особых условиях,
- регламентированное,

с периодическим контролем,
с непрерывным контролем.

По методам технического обслуживания выделяют поточный, централизованный, децентрализованный, а также методы технического обслуживания эксплуатационным или специализированным персоналом, эксплуатирующей или специализированной организацией, либо предприятием-изготовителем.

Основные компоненты системы технического обслуживания ПЭВМ

Наиболее массовым продуктом, содержащим микропроцессор, является персональная ЭВМ (ПЭВМ).

Быструю популярность ПЭВМ начали получать после выпуска в 1976 г. компанией Apple Computer ПЭВМ Apple и Apple-2 в 1977 г.

В 1981 г. фирма IBM выпустила IBM PC с операционной системой PC DOS, после чего началось массовое использование ПЭВМ. Конструкторские и программные решения IBM PC оказались настолько успешными, что на долгие годы послужили базовым стандартом для последующих разработок ПЭВМ.

В связи с тем, что ПЭВМ выпускались в большом количестве и являлись качественно совершенно новым и специфичным техническим устройством, фирма IBM создала соответствующую систему технического обслуживания персональных ЭВМ, принципы которой сохранили актуальность до сих пор. Основные компоненты этой системы:

- 1) средства встроенного контроля (обеспечивают поддержку средств самоконтроля, самодиагностики и самовосстановления);
- 2) техническая документация;
- 3) система гарантийного и послегарантийного обслуживания;
- 4) технологическое оборудование.

Тестирование при включении питания

Основная функция первой компоненты – контроль функционирования после включения и в процессе работы ПЭВМ. Средства встроенного контроля обеспечивают диагностирование практически всех узлов ПЭВМ: центральный процессор (микропроцессор); дисплей, клавиатура, ОЗУ, шины, дисководы гибких и лазерных дисков, винчестер, блок питания и др.

Встроенные средства диагностирования позволяют ЭВМ самостоятельно проводить предварительное тестирование сразу после включения ПЭВМ, вести контроль выполнения операций в процессе работы, локализовать возникающие неисправности.

Важная роль реализации этих функций отводится специальному постоянно-запоминающему устройству (ПЗУ), которое называют ROM BIOS. В этом ПЗУ хранится ряд программных модулей, которые предназначены для управления некоторыми процессами и узлами в компьютере, в том числе для проведения самоконтроля. Заметим, что в современных ПЭВМ функции BIOS распределены по нескольким микросхемам. Например, отдельной микросхемой может

быть видео-BIOS, программы которого используются для тестирования и организации последующей работы видеоплаты с процессором.

При включении компьютера сначала запускается специальный программный модуль **Power On Self Test (POST)** – самотестирование при включении питания, который находится в ROM BIOS.

Этот модуль управляет тестированием блока питания, процессора, ОЗУ, контроллеров дисков, дисплея, клавиатуры и других компонент ПЭВМ. Программа **POST** запускается автоматически после нажатия кнопки **⟨ПУСК⟩**. Диагностирование выполняется по принципу раскрутки. Процедура тестирования может занимать от нескольких до десятков секунд, в зависимости от тактовой частоты и объёма памяти компьютера. Только при успешном завершении программы **POST** начинается загрузка операционной системы и других программных модулей, необходимых для дальнейшей работы ПЭВМ.

Через пункт системного меню «BIOS Features Setup» (Установка характеристик BIOS) пользователь может ускорить выполнение процедуры самотестирования. Для этого в подпункте «Quick Power On Self Test» надо выбрать «Enabled» (быстрое самотестирование). Тогда BIOS сократит или пропустит некоторые проверки в процессе самотестирования.

Под управлением программы POST последовательно проверяются основные узлы компьютера. В ходе самотестирования определяется наличие дополнительных микросхем BIOS и, при необходимости, управление тестированием передается хранящимся в них программным модулям. Перед началом каждой проверки программа POST генерирует специальный код размером один байт (от 00 до FFh), называемый POST-кодом, и записывает его значение в специальный диагностический порт в адресном пространстве устройств ввода/вывода (в персональных ЭВМ для этого обычно выделяется порт с адресом 80h или 81h). В случае обнаружения неисправности POST-код, предварительно выведенный в диагностический порт, однозначно определяет операцию (проверку), при выполнении которой обнаружена неисправность. Узел, который проверялся на этой операции, считается неисправным.

В таблице 6.1 приведена последовательность проверок при включении питания для одной из первых отечественных ПЭВМ ЕС-1840, которая являлась аналогом ПЭВМ PC XT, первой в ряду персональных ЭВМ фирмы IBM.

Выполнение самотестирования после включения питания сопровождается выводом на экран сообщений о ходе проверки, в том числе звуковыми сигналами. В частности, выдается информация о типе процессора, подключенных устройствах, при проверке оперативной памяти на экран выводятся цифры, показывающие объем проверенной в данный момент памяти и др. В случае обнаружения неисправности на экран выводится соответствующее сообщение, при этом может выдаваться код ошибки, указывающий точное место неисправности.

Таблица 6.1

Последовательность проверок при включении компьютера

Номер проверки	Выполняемые операции
1	2
1	ВКЛЮЧЕНИЕ И ПРОВЕРКА ПИТАНИЯ
2	ФОРМИРОВАНИЕ СИГНАЛА «ПИТАНИЕ ИСПРАВНО». ЗАПУСК СИНХРОНИЗАЦИИ
3	ОЧИСТКА РЕГИСТРОВ
4	САМОТЕСТИРОВАНИЕ МИКРОПРОЦЕССОРА
5	ЗАПРЕТ ПРЕРЫВАНИЯ
6	УСТАНОВКА ФЛАГОВ ЦП. ТЕСТ ЧТЕНИЯ / ЗАПИСИ РЕГИСТРОВ ЦП
7	ПРОВЕРКА КОНТРОЛЬНОЙ СУММЫ ПЗУ BIOS. Сравнение суммы байтов, содержащихся в BIOS'е с известной хранимой величиной
8	ИНИЦИАЛИЗАЦИЯ КОНТРОЛЛЕРА ПРЯМОГО ДОСТУПА К ПАМЯТИ. Проверка правильности временных операций ПДП; инициализация таймера регенерации ОЗУ; запись фрагмента во все регистры и сверка с прочитанным; инициализация регистров; начало процесса регистрации ОЗУ ПДП
9	ЗАПУСК ТЕСТА ПАМЯТИ. Если старт «тёплый», пропустить
10	ТЕСТ ПЕРВЫХ 16 К ОЗУ. Запись пяти фрагментов в память, проверка правильности
11	ИНИЦИАЛИЗАЦИЯ КОНТРОЛЛЕРА ПРЕРЫВАНИЯ. Установка векторов прерываний; проверка контроллера прерываний; тест чтения / записи в регистр маскирования прерываний
12	ТЕСТ ТАЙМЕРА. Проверка скорости счёта
13	ИНИЦИАЛИЗАЦИЯ И ЗАПУСК КОНТРОЛЛЕРА МОНИТОРА. Считывание типа монитора; установка режима видео; тест чтения / записи видео ОЗУ
14	ОТОБРАЖЕНИЕ КУРСОРА
15	ИДЕНТИФИКАЦИЯ УСТАНОВЛЕННЫХ В СЛОТАХ БЛОКОВ. Тест чтения / записи на шинах адреса и данных
16	ПОЛНЫЙ ТЕСТ ОЗУ. Тест чтения / записи ОЗУ выше банка 0
17	ТЕСТ КЛАВИАТУРЫ. Формирование сигнала сброса клавиатуры; разрешение работы клавиатуры; проверка правильности кода сканирования (ОААН); очистка буфера клавиатуры; проверка отсутствия кода сканирования; установка таблицы векторов прерывания
18	ТЕСТ ДИСКОВОДОВ

1	2
19	ТЕСТ ДОПОЛНИТЕЛЬНОГО ПЗУ (С8000Н – F4000Н). Выполнение теста чтение /запись в порт принтера; запоминание адреса действующих портов; установка времени ожидания для портов принтера и интерфейса RS-232
20	РАЗРЕШЕНИЕ НЕМАСКИРУЕМЫХ ПРЕРЫВАНИЙ
21	ГЕНЕРАЦИЯ ЗВУКОВОГО СИГНАЛА И ВЫДАЧА КОДА ОШИБКИ В СЛУЧАЕ ОБНАРУЖЕНИЯ НЕИСПРАВНОСТИ
22	ЗАГРУЗКА ОПЕРАЦИОННОЙ СИСТЕМЫ С ДИСКОВОДА. ОЖИДАНИЕ КОМАНДЫ С КЛАВИАТУРЫ

Для удобства идентификации неисправностей коды разбивают на группы. Классификация кодов для ПЭВМ ЕС-1840 приведена в табл. 6.2. Вместо символов «х» компьютер проставляет цифры, указывающие на конкретную неисправность соответствующего устройства. Так код 601-09 означал неспособность дисководов сформировать дорожку, 606-27 – неправильная скорость диска и т.д.

Таблица 6.2

Классификация кодов для ПЭВМ ЕС-1840

Группа	Неисправное устройство
0xx	Источник питания
1xx	Системная плата
2xx	ОЗУ (до микросхемы)
3xx	Клавиатура
4xx	Адаптер монохромного алфавитно-цифрового дисплея (MDA)
5xx	Адаптер монохромного графического дисплея
6xx-xx	Дисковод гибких дисков и его контроллер
7xx	Сопроцессор
9xx	Параллельный порт
11xx	Последовательный порт
14xx	Графическое печатающее устройство
15xx	Адаптер синхронной передачи данных
17xx-xx	Накопитель на жестких дисках и его контроллер
18xx	Блоки расширения
19xx-xx	Стример-накопитель на магнитной ленте
24xx-xx	Графический редактор EGA

Средства самоконтроля, самодиагностики и восстановления вычислительного процесса

Дальнейшая работа ПЭВМ возможна, только если программа POST не обнаружила неисправностей. После тестирования оборудования загружается операционная система и при ее успешной загрузке ПЭВМ готова к выполнению задач пользователя. Заметим, что на этом работа встроенных средств контроля не заканчивается. В дальнейшем встроенными средствами контролируются практически все действия компьютера и пользователя. При появлении отказов на экран выдаются соответствующие сообщения, говорящие пользователю о характере и месте отказа.

Функции самоконтроля и самодиагностики реализуются с помощью соответствующих аппаратных и программных средств, при контроле правильности пересылки данных (чтения/записи) широко применяются различные способы кодирования данных (более подробно см. в теме 7).

При возникновении отказа ЭВМ пытается самостоятельно его устранить. Средства самовосстановления сначала пытаются определить характер отказа, в зависимости от которого применяются те или иные способы восстановления вычислительного процесса. Так, практически во всех ЭВМ применяются такие способы, как маскирование, повторение операции, возврат к контрольной точке, программный рестарт.

Под маскированием понимают устранение ошибки с помощью корректирующих кодов или резервирования, а также мажоритарных схем. Этот способ позволяет продолжать вычислительный процесс при многих устойчивых ошибках.

Восстановление повторением операции, при выполнении которой обнаружена ошибка, эффективно в случае сбоев. Однако здесь важно обнаружить ошибку сразу в момент ее появления, чтобы она не успела исказить используемые данные.

Способ возврата к контрольной точке может эффективно действовать, даже при сильных искажениях данных. Под контрольной точкой понимают точку (место в выполняемой программе), для которой сохранены промежуточные результаты вычислений и состояний процессора, что позволяет восстановить вычислительный процесс с данной точки. Реализация способа предполагает периодическое автоматическое назначение очередной контрольной точки и сохранение данных, необходимых для возобновления вычисления процесса с данной точки.

Средства самодиагностики и восстановления должны обеспечивать:

- непрерывность контроля;
- блокировку распространения ошибок;
- размещение точек контроля в узловых точках вычислительного процесса;
- регистрацию состояния ЭВМ в момент появления ошибки в специальных регистрах;
- точное указание команды (этапа ее выполнения), на которой была ошибка;

- сохранение предыстории выполнения команд после появления ошибки и до ее регистрации;
- повторение вычислительного процесса с соответствующей контрольной точки.

Заметим, что современные операционные системы ПЭВМ предоставляют неквалифицированному пользователю возможность в диалоговом режиме уточнить характер неисправности и получить указания по действиям с целью восстановления работоспособности компьютера. Например, в операционной системе Windows XP диагностику нарушений в работе монитора можно провести, нажав кнопку «ПУСК» и последовательно выбрав пункты «**Настройка – Панель управления – Экран**». В появившемся окне «**Свойства: Экран**» надо щелкнуть на кнопке «**Диагностика: ...**» (рис. 6.1). После этого появится окно со списком возможных признаков неисправности (рис. 6.2), в котором пользователь должен указать соответствующий признак и щелкнуть на кнопке «**Далее >**» и т.д. В итоге, пользователь получает конкретные указания о действиях, которые надо произвести для устранения неисправности (рис. 6.3).

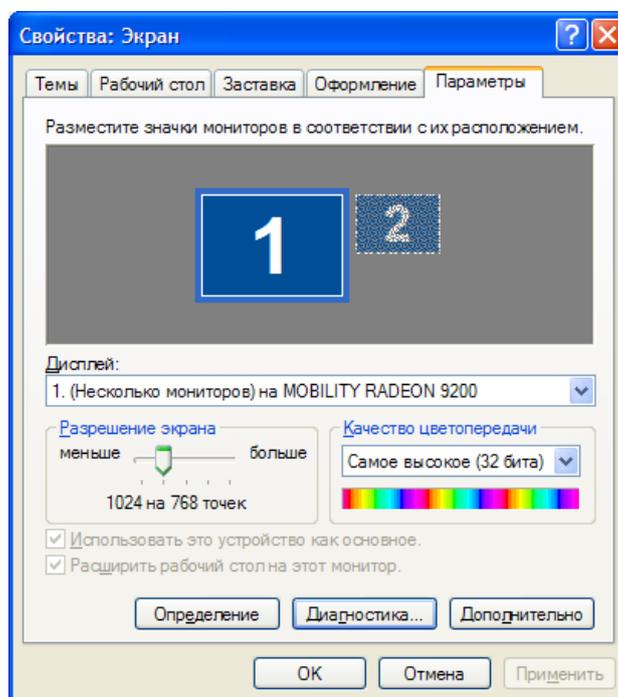


Рис. 6.1. Окно «Свойства: Экран»

Собственно базовая система ввода/вывода используется для управления различными узлами ЭВМ, в том числе для обработки прерываний, которые, в частности, вырабатываются при возникновении неисправностей.

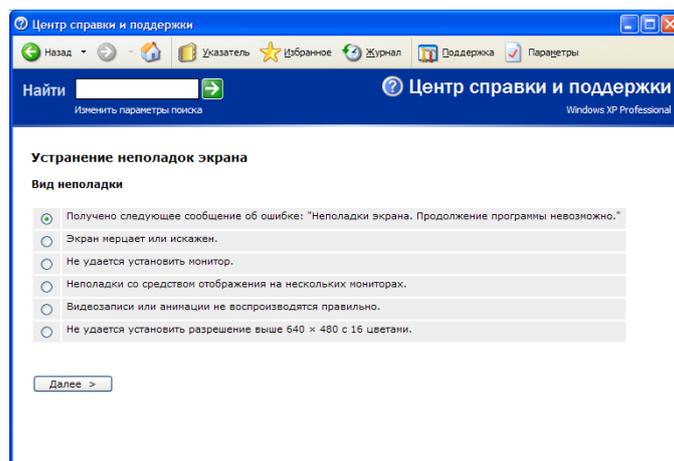


Рис. 6.2. Окно для указания признаков неисправности монитора

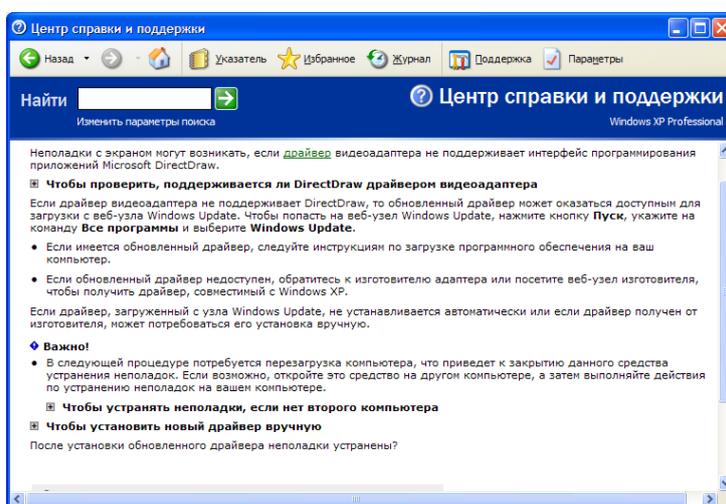


Рис. 6.3. Окно справки

Гарантийное и послегарантийное обслуживание

Сегодня мало кто решится купить компьютер, если продавец не дает гарантийных обязательств.

Фирма IBM, организовав массовое производство персональных компьютеров, одновременно создала систему их гарантийного и послегарантийного обслуживания, которая в основе используется до сих пор.

Типичная система гарантийного и послегарантийного обслуживания имеет иерархическую структуру:

- корпоративный центр поддержки;
- местное (региональное) отделение обслуживания;
- агент по продажам.

Для взаимодействия с клиентами организуется «Горячая линия», обслуживаемая квалифицированными специалистами. Пользователь по телефону или по E-mail обращается в корпоративный центр или местное отделение со своими проблемами, сообщает о результатах работы стандартных программ диагностики, установленных на его компьютере, тут же получает консультации.

Гарантийный срок обычно 12 месяцев, но может достигать нескольких лет. При покупке техники могут предлагаться варианты гарантийного обслуживания (выезд на место, время прибытия специалиста после вызова и др.).

Послегарантийное обслуживание происходит по договорам на абонентское обслуживание, в которых оговариваются виды и объем услуг, форма обслуживания. За фирмой, подписавшей договор, закрепляют технического специалиста, который осуществляет плановые и срочные выезды на обслуживаемую фирму.

Типичный набор услуг по таким договорам включает:

тестирование работы компьютеров, поиск и устранение неисправностей;

срочный ремонт;

подбор конфигурации, установка и настройка технического и программного обеспечения;

модернизация компьютеров;

обновление антивирусных и иных программ;

администрирование сети;

настройка и администрирование доступа в Интернет;

консультирование сотрудников по работе с программным обеспечением и внешними устройствами;

расширение сети (подключение новых компьютеров);

замена картриджей и тонеров.

Сегодня фирмы-производители ПЭВМ в части организации гарантийного и послегарантийного обслуживания все чаще применяют «Аутсорсинг», т.е. перепоручение невыгодных или непрофильных для компании работ другой организации. Крупные производители тщательно подходят к подбору фирм, которым доверяется создание **сервисных центров** для обслуживания. Они получают сертификаты, подтверждающие их полномочия и определяющие область авторизации – перечень заказчиков или регион, обслуживаемый данным сервисным центром.

Техническое обслуживание включает регламентированные в конструкторской (проектной) и/или эксплуатационной документации операции по поддержанию работоспособного и исправного состояния. Например, этими документами для ПЭВМ и ее внешних устройств могут устанавливаться ежедневное профилактическое обслуживание, еженедельное, месячное и полугодовое обслуживание. Для каждого вида обслуживания устанавливается перечень работ, который при этом надо выполнить. Так, один из вариантов еженедельного обслуживания включает следующие работы:

проверка работоспособности устройств на тестах в ускоренном режиме;

профилактика устройств внешней памяти, что необходимо для уменьшения износа элементов считывания информации, повреждения носителей информации и повышения их надежности;

проверка и удаление компьютерных вирусов с устройств внешней памяти ПЭВМ с целью исключения их воздействия на информацию пользователя и работоспособность устройств ПЭВМ;

проведение дефрагментации накопителей на жестких магнитных дисках, что повышает эффективность их работы и увеличивает срок службы;

проверка линий и устройств локальной вычислительной сети (ЛВС) с помощью автономных тестов.

Эксплуатационная документация

Диагностирование и ремонт таких сложных объектов, как компьютер невозможно без хорошей эксплуатационной документации.

Эксплуатационные документы – документы, предназначенные для использования при эксплуатации, обслуживании и ремонте изделия в процессе эксплуатации.

Техническую документацию, применяемую при эксплуатации ПЭВМ (по зарубежной классификации), условно можно разбить на три группы:

– руководство по эксплуатации типа «быстрая справка» (содержит основные инструкции по установке, эксплуатации, тестированию с помощью средств стандартного программного обеспечения, установке опций, установке переключателей на материнской плате, диагностический диск «покупательского» уровня);

– техническое руководство по применению (довольно подробная информация о технических и программных средствах, режимах их работы);

– руководство по обслуживанию технических средств (содержит информацию по действиям для выявления и устранения неисправностей; диагностические таблицы для ручного поиска неисправностей; диск со специальными программами для углубленного диагностирования и т.п.).

Ниже (в качестве примера) приведены фрагменты таблиц (табл. 6.3) для ручного поиска неисправностей из справочного руководства «Персональные компьютеры семейства Compaq Deskpro 2000».

В отечественных стандартах дается подробный перечень документов, которые должны создаваться при разработке и производстве технических объектов, в том числе касающиеся вопросов испытаний, технического обслуживания, ремонта и др. Это, например, документы:

– **программа и методика испытаний** (документ, содержащий технические данные, подлежащие проверке при испытании изделий, а также порядок и методы их контроля);

– **ремонтные документы** (документы, содержащие данные для проведения ремонтных работ на специализированных предприятиях);

– **инструкции** (документы, содержащие указания и правила, используемые при изготовлении (сборке, регулировке, контроле и т.п.);

– **ведомость держателей подлинников** (документ, содержащий перечень предприятий, на которых хранят подлинники документов, разрабатываемых и (или) примененных для данного изделия и др.

Пример таблицы неисправностей для компьютеров семейства Compaq Deskpro 2000

Неисправность	Причина	Способ устранения
1	2	3
Устранение неисправностей питания		
Компьютер не включается	Компьютер не подключен к внешнему источнику питания	Подключите компьютер к внешнему источнику питания
	Отсутствие контакта в разъемах кабелей, подключенных к внешнему источнику питания	Проверьте разъемы кабелей, подключающих компьютер к внешнему источнику питания
	Неисправна вновь установленная плата PCI или ISA	Замените дефектную плату
Компьютер не воспроизводит автоматически дату и время	Элемент питания RTC (часов реального времени) нуждается в замене. Срок службы элемента питания не превышает 5 лет	Замените элемент питания RTC
Компьютер автоматически выключается	Превышена рабочая температура устройства. Возможна блокировка вентилятора	Устройство работает в условиях слишком высоких температур. Дайте ему время, чтобы остыть. Или необходимо обратиться к специалисту по обслуживанию
Устранение сбоев изображения		
Пустой экран	Монитор не включен и не горит индикатор монитора	Выключите монитор и убедитесь, что загорелся индикатор монитора
	Неисправность проводных соединений	Проверьте проводные соединения между монитором, компьютером и сетевой розеткой
	Включена функция экономии электроэнергии	Нажмите любую клавишу или щелкните кнопкой мыши (если она подключена), наберите свой пароль

Окончание табл. 6.3

1	2	3
	Неправильно выбрано положение входного переключателя RGB (красный, зеленый, синий), установленный на задней стенке монитора	Установите входной переключатель монитора RGB на 75 Ом, если используется переключатель синхронизации, установите его в положение External
	Вы используете монитор с постоянной синхронизацией, который не поддается синхронизации при выбранной разрешающей способности	Проверьте, что монитор может принимать ту же частоту развертки, что и выбранная разрешающая способность.
Работа монитора нарушается при включении функции экономии электроэнергии	При включении функции экономии электроэнергии используется монитор, не снабженный такой возможностью	Отключите функцию экономии электроэнергии
Текст на экране содержит искаженные символы	В файле CONFIG.SYS не указан драйвер ANSI.SYS	Необходимо добавить драйвер ANSI.SYS к файлу CONFIG.SYS. Добавьте в файл следующую строку: DEVICE=C:\ANSI.SYS
Неправильные цвета видеобразия	Указывает на некорректное значение импеданса монтажных соединений или монитора	При использовании проводов BNC убедитесь, что красный, зеленый, синий провода BNC подсоединены к соответствующим разъемам монитора. Далее проверьте, что входы RGB монитора установлены на 75 Ом
Устранение неисправностей CD-ROM		
Система не загружается с накопителя на CD-ROM	При выполнении утилиты Computer Setup загрузка с CD-ROM не производится	Запустите утилиту Computer Setup. Установите приоритеты для накопителей
Устройства CD-ROM не распознаются, не загружен драйвер	Накопитель на CD-ROM не подключен или к нему не подключена балансная загрузка	Обратитесь к документации на CD-ROM

Специальные технические средства для обслуживания микропроцессорных устройств

Специфика функционирования электронных устройств, содержащих микропроцессор, сложность их диагностирования с помощью традиционных средств послужили причинами создания специальных приборов для их диагностирования и наладки. Это специализированное оборудование далее будем называть технологическим оборудованием.

Наиболее **важными компонентами** технологического оборудования являются:

- 1) логический анализатор,
- 2) сигнатурный анализатор,
- 3) внутрисхемный эмулятор,
- 4) диагностическая плата,
- 5) стенды технологического контроля.

Особенности требований к размещению и эксплуатации средств вычислительной техники

В России вопросы безопасности труда при использовании ЭВМ регулируются техническими регламентами, стандартами, государственными санитарно-эпидемиологическими правилами и нормативами.

На момент издания настоящего учебного пособия эти вопросы, в частности, регулировались **Санитарно-эпидемиологическими правилами и нормативами СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы»**. В СанПиН 2.2.2/2.4.1340-03 довольно подробно изложены требования к помещениям для работы с ПЭВМ, требования к обустройству рабочих мест, к уровням шума, вибрации, уровням электромагнитных полей, освещенности и др. При этом отдельно выделены требования к оснащению рабочих мест разных категорий пользователей: дети дошкольного возраста, обучающиеся в системе начального и высшего образования, взрослые пользователи.

Ниже приведены некоторые из базовых требований, относящиеся к безопасности труда при работе с ПЭВМ:

- мощность экспозиционной дозы мягкого рентгеновского излучения в любой точке на расстоянии 0,05 м от экрана и корпуса видеодисплейного терминала (на электроннолучевой трубке) при любых положениях регулировочных устройств не должна превышать 1 мкЗв/час (100 мкР/час);
- не допускается размещение мест пользователей ПЭВМ во всех образовательных и культурно-развлекательных учреждениях для детей и подростков в цокольных и подвальных помещениях;
- площадь на одно рабочее место пользователей ПЭВМ с видеодисплейным терминалом на базе электроннолучевой трубки должна составлять не менее 6 м², в помещениях культурно-развлекательных учреждений и с терминалом на базе плоских дискретных экранов (жидкокристаллические, плазменные) – 4,5 м²;

- при использовании ПЭВМ с терминала на базе электроннолучевой трубки (без вспомогательных устройств – принтер, сканер и др.), отвечающих требованиям международных стандартов безопасности компьютеров, с продолжительностью работы менее 4-х часов в день допускается минимальная площадь 4,5 м² на одно рабочее место пользователя (взрослого и учащегося высшего профессионального образования);
 - в помещениях, оборудованных ПЭВМ, проводится ежедневная влажная уборка и систематическое проветривание после каждого часа работы на ПЭВМ;
 - рабочие столы следует размещать таким образом, чтобы видеодисплейные терминалы были ориентированы боковой стороной к световым проемам, чтобы естественный свет падал преимущественно слева;
 - освещенность на поверхности стола в зоне размещения рабочего документа должна быть 300–500 лк. Освещение не должно создавать бликов на поверхности экрана. Освещенность поверхности экрана не должна быть более 300 лк;
 - при размещении рабочих мест с ПЭВМ расстояние между рабочими столами с видеомониторами (в направлении тыла поверхности одного видеомонитора и экрана другого видеомонитора), должно быть не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов – не менее 1,2 м;
 - рабочие места с ПЭВМ в помещениях с источниками вредных производственных факторов должны размещаться в изолированных кабинках с организованным воздухообменом;
 - рабочие места с ПЭВМ при выполнении творческой работы, требующей значительного умственного напряжения или высокой концентрации внимания, рекомендуется изолировать друг от друга перегородками высотой 1,5–2,0 м;
 - экран видеомонитора должен находиться от глаз пользователя на расстоянии 600–700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов;
 - требования к организации и оборудованию рабочих мест с ПЭВМ для взрослых пользователей:
 - высота рабочей поверхности стола для взрослых пользователей должна регулироваться в пределах 680–800 мм; при отсутствии такой возможности высота рабочей поверхности стола должна составлять 725 мм;
 - модульными размерами рабочей поверхности стола для ПЭВМ, на основании которых должны рассчитываться конструктивные размеры, следует считать: ширину 800, 1000, 1200 и 1400 мм, глубину 800 и 1000 мм при нерегулируемой его высоте, равной 725 мм;
 - рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной – не менее 500 мм, глубиной на уровне колен – не менее 450 мм и на уровне вытянутых ног – не менее 650 мм;
 - клавиатуру следует располагать на поверхности стола на расстоянии 100–300 мм от края, обращенного к пользователю или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.
- Кроме того, специальные требования предъявляются к конструкции и размерам рабочего стула, организации системы освещения помещения и др.

Эксплуатационная документация обычно содержит некоторые общие рекомендации, связанные с включением компьютера и его подготовкой к самотестированию и работе, например:

- проверьте все проводные соединения на предмет выявления нарушений контакта или неисправного соединения;
- убедитесь, что компьютер и монитор подключены к работающей розетке электросети;
- при включении системы выньте диски из дисководов;
- при включении компьютера должен загореться зеленый индикатор питания;
- убедитесь, что все необходимые драйверы устройств установлены (в современном компьютере при включении автоматически определяются и устанавливаются необходимые драйверы);
- при тусклом экране монитора или искажении (сдвига) изображения с помощью ручек регулировки выполните необходимые регулировки.

Обычно необходимые программные и аппаратные настройки делает поставщик компьютера, однако пользователю полезно знать о необходимости таких настроек.

Современные ПЭВМ очень надежные устройства, редко отказывают, даже если нарушаются условия эксплуатации. Однако при нарушении условий эксплуатации повышается вероятность сбоев и отказов.

Существует ряд простых практических правил, которые рекомендуется соблюдать при установке и эксплуатации ПЭВМ:

- колебания температуры неблагоприятно сказываются на состоянии компьютера. Поэтому, чтобы компьютер работал надежно, температура должна быть постоянной (если комната сильно нагревается в солнечные дни, рекомендуется установить кондиционер). Для любых электронных устройств указывается допустимый диапазон температур. Большинство фирм-изготовителей приводят эти данные в паспорте изделия;
- следует максимально исключить электрические воздействия, радиочастотные помехи, помехи и колебания напряжения в цепях питания. Желательно подключать компьютер к сети 220 В трехштырьковой вилкой с хорошим заземлением;
- старайтесь подключать компьютеры к отдельным линиям питания со своими предохранителями. Не пользуйтесь без крайней необходимости удлинителями, поскольку при этом возрастает уровень помех в сети. Контактные гнезда удлинителей часто не обеспечивают хороший контакт. В результате возникают помехи, происходит нагрев в месте контакта с вилкой шнура питания. Число устройств подключенных к розетке, к которой подключен компьютер, должно быть как можно меньше;
- надежность работы компьютера резко снижается в запыленных и задымленных помещениях. Вентилятор блока питания втягивает имеющиеся в воздухе частицы внутрь, что приводит к нежелательным последствиям. Пыль, оседая внутри компьютера, ухудшает охлаждение системы. В результате сокращается срок службы компонентов и увеличивается перепад температур при прогреве компьютера. В пыли содержатся проводящие частицы, которые могут привести

к возникновению утечек и даже коротких замыканий. В качестве профилактики рекомендуется периодически снимать корпус компьютера и проводить удаление пыли;

- нельзя допускать образование конденсата или попадания воды и других жидкостей на оборудование. Если влага попала внутрь компьютера, немедленно прекратите работу и выключите питание. Эксплуатация во влажных помещениях, как правило, снижает надежность и со временем вызывает отказы винчестеров и блока питания.

- если компьютерную технику предполагается эксплуатировать в неблагоприятных условиях, то следует покупать изделие, разработанное специально для работы в подобных условиях.

На работу компьютера влияет не только его аппаратная часть, но и эксплуатируемое системное и программное обеспечение. Так, производительность компьютера можно существенно увеличить за счет соответствующего выбора и настройки драйверов. Если решаемые задачи требуют использования больших программ и больших объемов данных, то повысить производительность можно, увеличив объем оперативной памяти. Немаловажный момент, влияющий на скорость работы компьютера, – это организация данных на жестком диске. По мере того, как вы записываете файлы на жесткий диск, редактируете и удаляете их, многие из них фрагментируются, т.е. разбиваются на множество разбросанных по всему диску частей. Считывание с диска такого «фрагментированного» файла требует дополнительного времени на перемещение головок дисковвода и поиск фрагментов.

Профилактическое обслуживание программной части компьютера главным образом заключается в проверке на наличие вирусов, выявлении и исправлении нарушений в системных файлах, оптимизации файла подкачки и файлов реестра, удалении ненужных файлов, обновлении драйверов устройств, дефрагментации файлов, а также в выявлении и блокировании сбойных участков жесткого диска.

Частота проведения профилактических работ, определяется пользователем и зависит от загруженности и «возраста» компьютера, серьезности обрабатываемых на нем задач.

Показатели систем технического обслуживания и ремонта

Показатели системы технического обслуживания и ремонта позволяют оценить затраты времени, труда и средств на техническое обслуживание и ремонты и содержат затраты, обусловленные конструкцией и техническим состоянием изделия (оперативные затраты), и затраты, обусловленные организацией, технологией выполнения технического обслуживания и ремонтов, материально-техническим обеспечением, квалификацией персонала, условиями окружающей среды и т.д.

Рекомендуемые показатели перечисляются, в частности, в ГОСТ 18322-78 (Система технического обслуживания и ремонта техники. Термины и определения).

Средняя продолжительность технического обслуживания (ремонта) – математическое ожидание продолжительности одного технического обслужива-

живания (ремонта) данного вида за определенные период эксплуатации или наработку.

Средняя трудоемкость технического обслуживания (ремонта) – математическое ожидание трудоемкости одного технического обслуживания (ремонта) данного вида за определенные период эксплуатации или наработку.

Средняя стоимость технического обслуживания (ремонта) – математическое ожидание стоимости одного технического обслуживания (ремонта) данного вида за определенные период эксплуатации или наработку.

Вышеприведенные показатели предназначены для оценки одного технического обслуживания. В то же время за период «жизни» объект, как правило, многократно подвергается техническому обслуживанию. Тогда стандарт предлагает возможность оценивать **среднюю суммарную** или **удельную суммарную** трудоемкость, стоимость и/или продолжительность по всем техническим обслуживаниям. Удельная суммарная, например, продолжительность обслуживания определяется как отношение средней суммарной продолжительности технических обслуживаний (ремонтов) к заданной наработке.

Кроме того, в число основных показателей включают **коэффициенты готовности и технического использования** (см. п. 1.4), **готовность парка изделий**.

Готовность парка изделий – отношение числа работоспособных изделий к общему числу изделий парка в рассматриваемый момент времени.

6.3. Логические анализаторы и пробники

Назначение и принцип работы

Один из подходов к поиску неисправностей в микропроцессорных устройствах и системах состоит в контроле текущей информации на системных шинах: шинах адреса, данных и управления. При этом информация снимается одновременно по нескольким линиям и фиксируется в определенные моменты времени (возможен съем одновременно с нескольких сот линий). Например, если мы снимаем информацию с адресной шины и фиксируем её в последовательные моменты, то тем самым мы можем проверить правильность перебора адресов (в соответствии с программой). Если снимается информация и с шины данных, то мы можем проверить правильность поступающих для считывания команд и данных, а также результат выполнения отдельных команд. Информация с шины управления позволяет проверить правильность выработки управляющих сигналов и протокол обмена на системной шине.

Сигналы в ЭВМ относятся к цифровым, т.е. характеризуются только двумя уровнями напряжения (условно обозначаемыми «0» и «1»). Анализ двоичной информации с целью проверки работоспособности устройства и поиска неисправностей называется логическим анализом. Для такого анализа были разработаны специальные приборы – **логические анализаторы** (рис. 6.4).

Логический анализатор (ЛА) выполняет функции многоканальной регистрации, запоминания и отображения информации о поведении контролируемого

устройства в моменты времени, предшествующие какому-либо событию или в следующие за ним. Позволяет наблюдать одновременно на нескольких линиях однократные и периодические сигналы. Информация отображается на экране дисплея в виде временных диаграмм или таблиц состояний. ЛА применяют при разработке, изготовлении и эксплуатации средств вычислительной техники. Они являются эквивалентом многоканального осциллографа, специально приспособленным для работы с микропроцессорными устройствами.

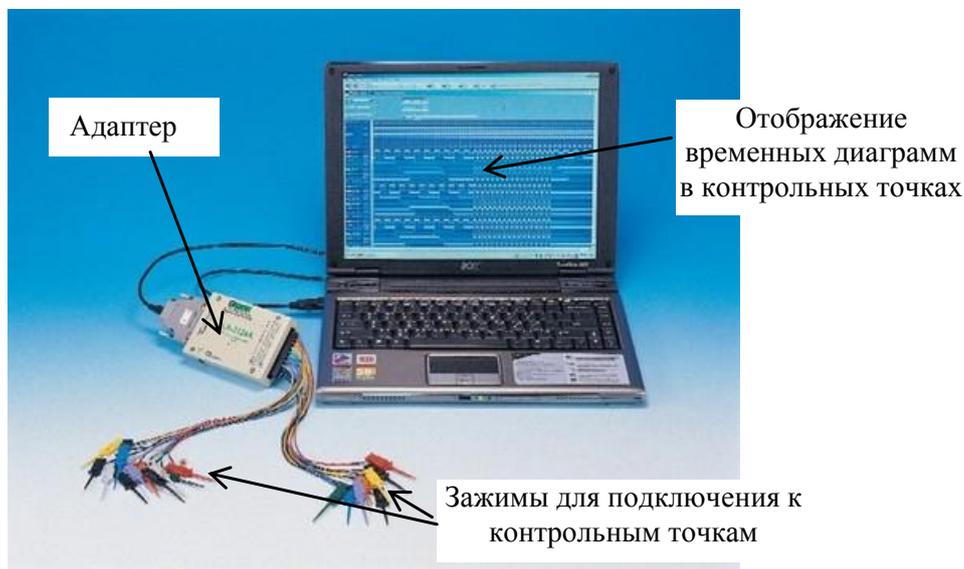


Рис. 6.4. Логический анализатор LA-2124 фирмы Flite Electronics International

В логических анализаторах информация запоминается не непрерывно, а в определенные моменты времени. С этой целью в ЛА имеется тактирующий вход. Тактовые импульсы, по которым ведется отсчет времени, могут подаваться как от внутреннего генератора через регулируемый делитель частоты, так и от внешнего источника (например, от контролируемой микропроцессорной системы).

Возможно подключение ЛА прямо к внешнему разъему на объекте диагностирования. Для подключения ЛА к внутренним контрольным точкам используются специальные щупы с зажимами. Есть щупы для подключения к отдельным точкам (рис. 6.5а); для подключения одновременно к группе точек (например, к выводам микросхемы) используются зонды-клипсы, которые надеваются на корпус микросхемы и обеспечивают контакт с каждым ее выводом (рис. 6.5б и 6.5в).

С точки зрения схемотехники логический анализатор представляет собой быстродействующую буферную оперативную память. Буфер этот односторонний: сначала в буферную память с большой тактовой частотой последовательно записываются значения входных сигналов, а затем эта информация последовательно читается из буфера.

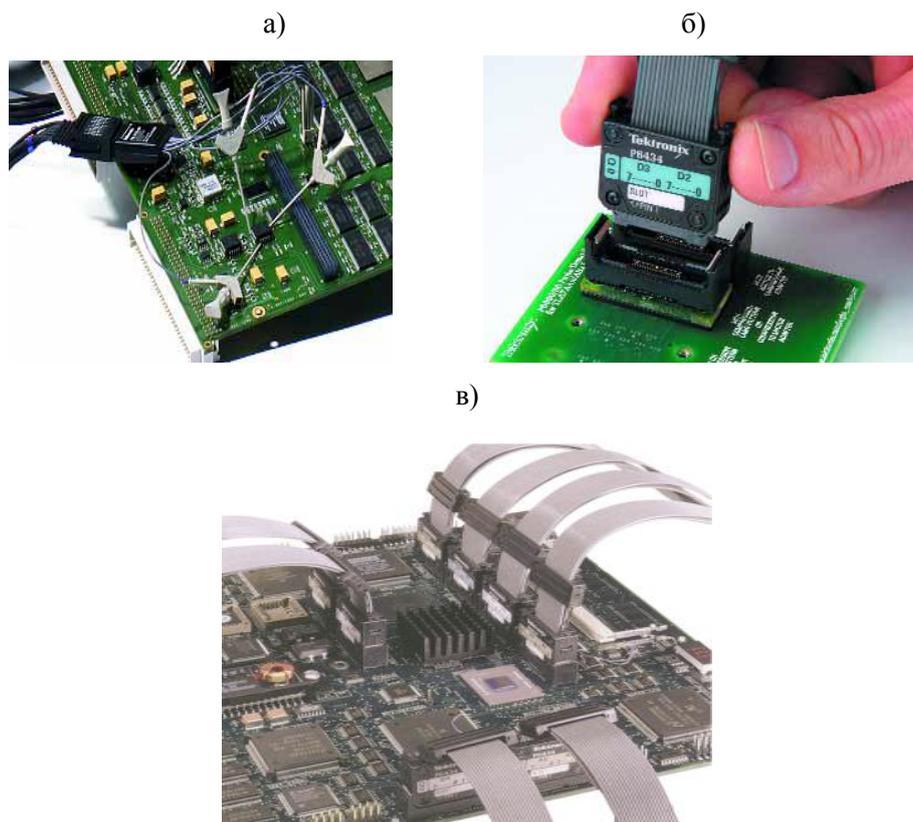


Рис. 6.5. Примеры щупов для подключения ЛА к контрольным точкам

Основные параметры и функции

Логические анализаторы характеризуются числом одновременно контролируемых каналов, емкостью буферной памяти для хранения снимаемых данных, максимальной частотой выборки, способами синхронизации и запуска, частотами внешней и внутренней синхронизации, интерфейсом с компьютером, формами представления данных.

Микропроцессорные устройства работают на высоких частотах. Поэтому одно из важнейших требований к ЛА – минимизация влияния на диагностируемый объект. ЛА должны обладать высоким входным сопротивлением (сотни-десятки сотен килоом в зависимости от частоты измерений и элементной базы объекта) и малой входной емкостью (порядка 10–25 пФ). Для обеспечения этих требований компараторы логических уровней делаются выносными.

Функционально ЛА подразделяют на два типа: **анализаторы логических состояний (АЛС)** и **анализаторы временных диаграмм (АВД)**. АЛС фиксируют состояние контрольных точек проверяемой схемы во время тактовых сигналов, задаваемых самим проверяемым устройством, и записывают процесс изменения состояний синхронно с его работой.

АВД фиксирует состояние контрольных точек проверяемой схемы в моменты времени, которые задаются независимо работающим внутренним тактовым генератором самого ЛА, и состояния контрольных точек фиксируются в дискретные моменты времени, задаваемые этим внутренним генератором (его частота в несколько раз выше, чем у объекта диагностирования).

Современные ЛА зачастую совмещают возможности АЛС и АВД.

Последовательности значений логических состояний, полученные с каждой контрольной точки, заносятся во внутреннюю память анализатора. В первом случае данные заносятся в память синхронно с изменением логических состояний диагностируемого узла, а во втором – по тактовым сигналам самого ЛА, т.е. асинхронно по отношению к диагностируемому узлу. Поэтому очень часто АЛС называют синхронными, а АВД – асинхронными анализаторами.

В последующем пользователь ЛА может, пользуясь специальными командами, запрограммировать вывод на экран диаграммы изменения сигналов на выбранных контрольных точках в заданные периоды времени. Обычно экран ЛА позволяет одновременно вывести диаграммы в 16–32 контрольных точках (рис. 6.6).

В реальных системах могут возникать импульсы и всплески напряжений, длительность которых меньше периода тактовых сигналов. Такие ложные сигналы считаются помехами.

Существуют два способа обнаружения помех: увеличение тактовой частоты съема сигналов с контрольных точек по отношению к частоте работы диагностируемого объекта (по крайней мере в 5–6 раз) и применение «ловушек». В первом случае возможно фиксировать появление помехи между тактами работы объекта и показать их на временных диаграммах как данные, во втором – просто фиксируется появление ложных импульсов.

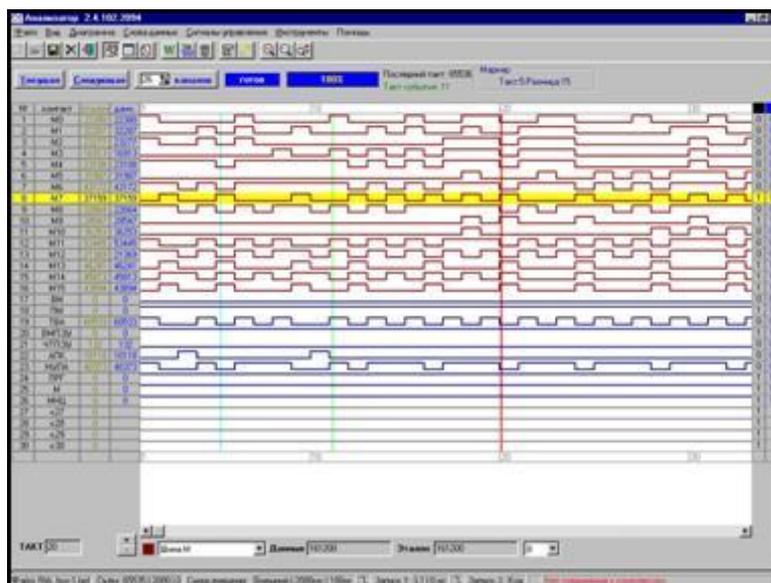


Рис. 6.6. Пример вывода результатов измерений сигналов на 32-разрядной шине

Промышленность выпускает самые разнообразные варианты конструктивного исполнения ЛА. На рисунке 6.4 показан ЛА, в котором управление, обработка измерений и вывод результатов выполняются с помощью портативного компьютера. Выпускаются ЛА, выполненные в виде вставных плат (рис. 6.7б) в компьютер или в виде автономного прибора (рис. 6.7а), с возможностью обмена с компьютером через интерфейс USB или параллельный порт.

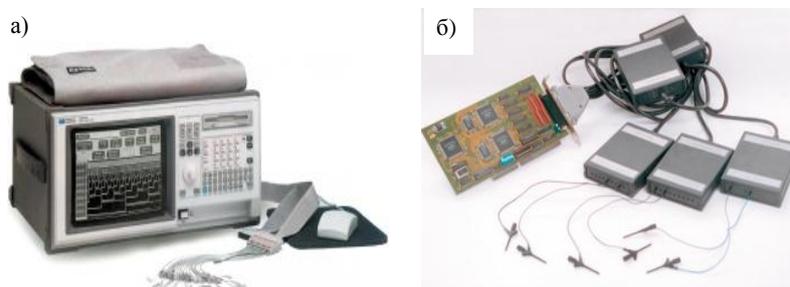


Рис. 6.7. Примеры конструктивного исполнения логических анализаторов

Основное разнообразие в семейство логических анализаторов вносят способы запуска и формы представления данных. Съем сигналов с контрольных точек обычно выполняется через выносной адаптер, который выступает в качестве усилителя-преобразователя снимаемых напряжений в цифровые коды, поступающие далее для обработки и запоминания в памяти ЛА. Анализаторы на основе ПЭВМ имеют очевидное достоинство, связанное с наличием больших объемов дисковой памяти, что позволяет организовать хранение, в том числе долговременное, результатов измерений и эталонных значений для больших тестов и многих объектов.

Особенности регистрации отображения сигналов

ЛА имеют два основных режима работы: регистрации и отображения.

Регистрация – процесс записи состояния сигналов, поступающих по входным каналам анализатора в его запоминающие устройства. Регистрация начинается по сигналу запуска, который может быть внешним сигналом, кодовым словом либо последовательностью кодовых слов.

Отображение – процесс индикации на экране (дисплее) временных диаграмм или логических состояний, записанных в ЗУ в процессе регистрации.

Регистрация (запоминание) информации выполняется при появлении сигнала «Запуск». Как правило, возможны следующие способы формирования запускающего импульса:

- а) от кнопки «Запуск»;
- б) от внешнего сигнала, подаваемого на вход «запуск»;
- в) от запускающего слова.

Запускающим словом называется двоичный код, выставляемый оператором с помощью переключателей или программно. Запуск ЛА осуществляется с того момента, когда информация на некоторых входах, к которым подключены щупы ЛА, совпадает с запускающим словом. Возможен запуск с задержкой – через определенное время, определяемое числом тактов задержки.

Пусть, например, ЛА подключен к адресной шине и мы хотим просмотреть информацию на этой шине с того момента, когда процессор выставит на шину адрес $008F_{16}$. С этой целью мы с помощью переключателей устанавливаем запускаящее слово « $008F_{16}$ ». Как только на входе ЛА появится указанный адрес, идёт запоминание информации по каждому тактовому импульсу.

Объём запоминаемой информации зависит от объёма внутренней памяти ЛА и может быть сотни и более килобит на каждый канал. Уменьшая число измеряемых каналов, обычно можно увеличить число запоминаемых на канал сигналов.

В режиме отображения информация выводится на экран в виде логических временных диаграмм, таблиц, графического изображения, буквенных выражений. При отображении в виде таблиц информация может кодироваться в двоичном, восьмиричном, шестнадцатиричном или алфавитно-цифровом коде. Для вывода оператор должен указать каналы и такты, для которых надо вывести значения сигналов. Можно указать начало вывода за несколько тактов до некоторого события (например, конкретного адреса на шине или момента фиксации сбоя в работе диагностируемого объекта).

При необходимости исследования формы сигналов ЛА могут дополнительно оснащаться программным осциллографом, реализованном на базе вставной платы с АЦП. В этом случае на экране можно показывать не только логические уровни сигналов в моменты поступления тактовых импульсов, но и их форму в реальном времени, как на экране обычного осциллографа. Программная оболочка такого осциллографа включает в себя набор сервисных услуг, таких, как экранный указатель, поиск заданных блоков данных, сравнение, масштабирование. Кроме того, предоставляется возможность работы с цветом, шрифтами и т.д.

В ряде случаев бывает необходимо проконтролировать в некоторой точке лишь наличие сигнала и его логическое значение. Это, конечно, можно сделать и с помощью логического анализатора, но он довольно велик и дорог. Более удобно использовать в таких случаях так называемые логические пробники. Ими можно пользоваться, держа в руке, как обычный карандаш или авторучку. Результат измерения показывается на жидкокристаллическом или светодиодном индикаторе, обычно расположенном на боковой поверхности корпуса пробника (рис. 6.8).

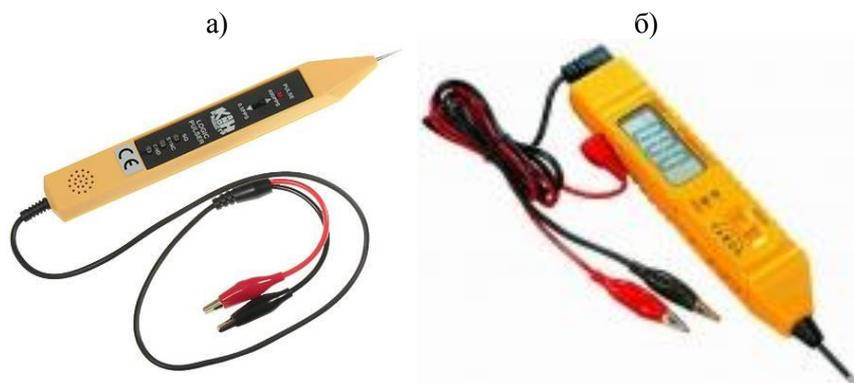


Рис. 6.8. Примеры логических пробников: а) LP540H, б) M3211D

Такие пробники, несмотря на их простоту, позволяют регистрировать и отображать: уровень напряжения в измеряемой точке (логический 0 или 1, вне уровня), генерацию, одиночный импульс и др. Известны пробники, которые, кроме функций измерения, могут иметь режим подсчета числа импульсов или генерации кратковременных импульсов тока, что может потребоваться для точного определения места некоторых типов неисправностей в смонтированных печатных платах.

Головная часть пробника выполняется в виде иглы, в корпус может устанавливаться звуковой индикатор. Для подключения к источнику питания, создания общей «земли» с диагностируемым объектом или передачи результатов измерения во внешнее устройство обработки (например ПЭВМ) пробник может снабжаться специальными кабелями. Необходимость подключения пробника к компьютерной системе обработки возникает, например, если он является частью автоматизированной системы пошагового поиска неисправностей, в которой результаты ручных измерений используются компьютером для выдачи информации об очередной точке измерений.

Промышленность выпускает и более сложные ручные пробники, которые могут регистрировать и отображать не только логические уровни, но и измерять напряжение, сопротивление, частоту и т.п.

6.4. Внутрисхемные эмуляторы

Назначение и конструкция

Принцип внутрисхемной эмуляции состоит в замене некоторого узла диагностируемого объекта (микропроцессор, модуль памяти, внешнее устройство и т.п.) неким устройством, поведение которого с точки зрения отлаживаемой системы соответствует поведению заменяемого узла. Это устройство, называемое **внутрисхемным эмулятором** (рис. 6.9), должно не только адекватно отражать поведение заменяемого узла, но и предоставлять разработчику возможность управления процессом отладки и контроля за состоянием всего диагностируемого объекта.

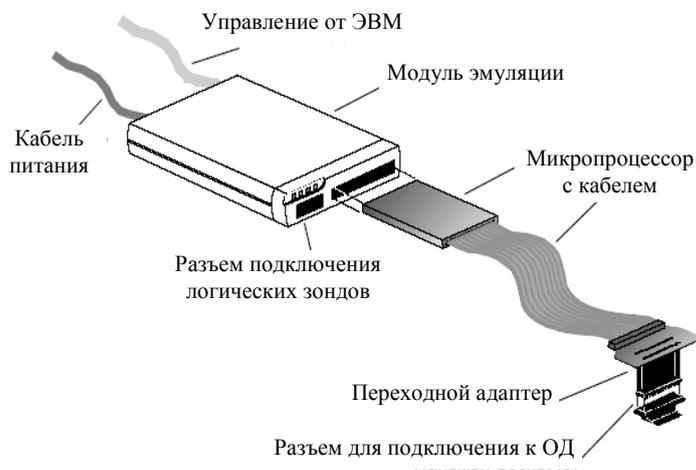


Рис. 6.9. Типичная конструкция комплекса эмуляции

При этом функции и режим работы микропроцессорной системы сохраняются почти такими же, как в реальном применении, обеспечивается программно-управляемый доступ к ее узлам для задания различных отладочных и диагностических режимов работы, внесения изменения, оценки состояния и т.д.

Эмуляторы обычно используются для поиска неисправностей и отладки новых программ.

Чтобы сделать конструкцию эмулятора более универсальной, заменяющий узел и схему сопряжения с ним (модуль эмуляции) зачастую выполняют в виде отдельных конструктивных узлов. При этом заменяющий узел подключается с помощью разъемов к модулю эмуляции и к объекту диагностирования (ОД), в котором замещается соответствующий элемент (рис. 6.9). Для подключения к сигналам синхронизации (такт, чтение/запись и др.) предусматриваются специальные входы (на схеме рис. 6.9 это разъем для подключения логических зондов).

Возможны эмуляторы, подключение которых предполагает физическое удаление замещаемого элемента, вместо которого подключается эмулятор. Если этот элемент несъемный, т.е. его невозможно удалить из отлаживаемой системы, то использование эмулятора возможно, если система имеет отладочный режим работы, при котором все его выходы переведены в третье состояние (высокий импеданс). Тогда после перевода системы в отладочный режим для подключения эмулятора используют специальный адаптер-клипсу, с помощью которого он подключается непосредственно к выводам замещаемого элемента.

Внутрисхемные эмуляторы делятся на управляемые от внешней ЭВМ и функционирующие автономно. Внешняя (управляющая) ЭВМ служит для обеспечения диалога с пользователем, ввода, редактирования и хранения программ, отображения текущего состояния отлаживаемой системы и управления процессом отладки. Автономные внутрисхемные эмуляторы имеют индивидуальные вычислительные ресурсы, средства ввода-вывода, не требуют для своей нормальной работы стыковки с какими-либо внешними вычислительными средствами, но за это пользователю приходится расплачиваться либо существенно более высокой ценой, либо пониженными функциональными и сервисными возможностями.

В общем случае эмулятор позволяет:

- осуществлять системное тестирование ОД с измененной частотой;
- выполнять функции всех или отдельных блоков ОЗУ и ПЗУ ОД;
- просматривать и редактировать содержимое эмулируемой памяти;
- имитировать функционирование устройств ввода-вывода, подачу внешних сигналов;
- выполнять трассировку исполняемых программ¹;
- определять количество обращений к различным участкам программы; время, затраченное на выполнение различных участков программы и циклов, и др.

¹ Трассировка – сохранение последовательности исполнения инструкций (команд) микропроцессором.

Основные функции

После подключения эмулятора в его память из управляющего компьютера загружается специальная программа – **отладочный монитор**. В последующем он выполняется на процессоре диагностируемой системы. Возможности эмулятора во многом определяются возможностями монитора.

Основные функции отладочного монитора:

- загрузка программы пользователя или специальной диагностической программы в свободную от монитора память;
- задание точек останова;
- запуск и останов загруженной программы в реальном времени;
- проход программы пользователя по шагам;
- просмотр, редактирование содержимого памяти и управляющих регистров.

С помощью отладочного монитора загружается информация об адресах, по которым установлены точки останова, о распределении адресного пространства между эмулятором и отлаживаемой системой и др. Затем управление передается целевому процессору и начинается выполнение программы.

Обычно выполнение программы пользователя возможно как минимум в одном из трех режимов:

- режим реального времени (программа выполняется, как будто эмулятор отсутствует);
- пошаговый режим;
- режим с остановками по заданным условиям.

При выполнении в режиме реального времени программа выполняется полностью адекватно целевой системе без эмулятора. В двух последних случаях управление периодически передается резидентной программе – отладочному монитору.

По ходу выполнения программы целевой процессор выставляет на внешнюю шину текущий адрес и, как только он совпадает с адресом точки останова, вырабатывается сигнал прерывания, по которому управление передается монитору. Под управлением отладочного монитора происходит обмен данными (в управляющую ЭВМ поступает информация о состоянии внутренних регистров и ячеек памяти отлаживаемой системы, при необходимости ЭВМ их изменяет и записывает новые значения в отлаживаемую систему).

В пошаговом режиме программа пользователя выполняется последовательно по одной команде с возможностью передачи информации о состоянии **отлаживаемой системы в управляющую ЭВМ**. После выполнения каждой команды происходят вызовы отладочного монитора.

Устройства для эмуляции микропроцессоров могут использовать память отлаживаемой (диагностируемой) системы либо использовать свою память, т.е. одновременно осуществлять эмуляцию памяти. Однако существуют внутрисхемные эмуляторы, основной функцией которых является эмуляция (подмена) постоянной памяти (ПЗУ или ППЗУ) или оперативной памяти (рис. 6.10). При этом можно эмулировать как всю память, так и ее отдельные участки.

Эмулятор ПЗУ – программно-аппаратное средство, позволяющее замещать ПЗУ на отлаживаемой плате и подставляющее вместо него ОЗУ, в которое может быть загружена программа с управляющего компьютера через один из стандартных каналов связи. Это устройство позволяет пользователю избежать многократных циклов перепрограммирования ПЗУ.

Эмуляторы, приведенные на рис. 6.10, подключаются на плату вместо микросхемы ПЗУ через специальную колодку, могут работать с компьютером через стандартный порт ПЭВМ. Эмуляция происходит в реальном времени, посредством компьютера можно просматривать и редактировать содержимое памяти во время эмуляции, имеется буфер трассировки. В реальном времени можно:

- менять и устанавливать необходимые переменные количества циклов, временные константы и др.;
- просматривать таблицы трассировки;
- моделировать подачу внешних сигналов.

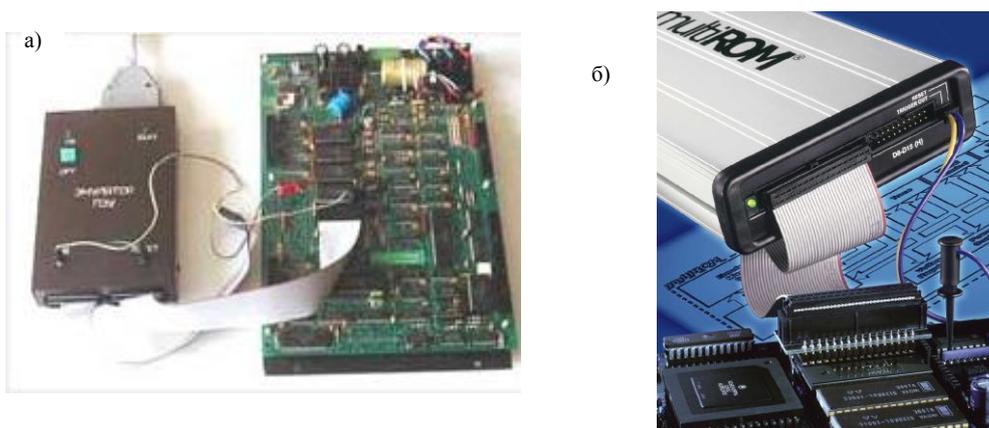


Рис. 6.10. Эмуляторы памяти: а) модуль системы «ТЕСТ-Д»,
б) модуль серии multiROM

Ранние эмуляторы ПЗУ позволяли только загружать программу, запускать ее и останавливать, используя общий сброс. Затем появились усложненные модели с возможностью задания неограниченного числа точек останова, аппаратной выработкой сигналов трассировки по достижении определенного адреса и вывода на осциллограф или монитор и др.

Современные эмуляторы ПЗУ представляют собой гибрид из обычного эмулятора ПЗУ, монитора отладки и схем быстрого переключения шины с одного на другой, что позволяет не только контролировать работу памяти, но и отслеживать состояние отлаживаемой системы в целом. В качестве источника тактового сигнала может быть выбран генератор, расположенный на плате отлаживаемого устройства, либо встроенный во внутрисхемный эмулятор. В последнем случае предусматривается возможность программного управления частотой тактового сигнала.

Отладочные интерфейсы

В 1985 г. группой ведущих производителей электронной техники был создан комитет JTAG, который принял стандарт на технологию периферийного сканирования при тестировании цифровых устройств. Стандарт предусматривает наличие у каждой микросхемы пяти дополнительных выводов для последовательного ввода управляющих сигналов и вывода состояний регистров диагностируемого объекта. В настоящее время выпускаются серии логических микросхем, а также процессоры с поддержкой этого стандарта. Использование схемы интерфейса JTAG в универсальном эмуляторе позволит унифицировать методы подключения и отладки соответствующих процессоров. Также при использовании JTAG можно производить отладку процессоров, не вынимая их из отлаживаемых устройств, что весьма важно для высоко интегрированных систем с затрудненным доступом к электронным цепям. При этом управление процессами отладки и контроля может происходить через стандартный порт ПЭВМ (например СОМ-порт).

Заметим, что идеи JTAG в дальнейшем получили развитие в других стандартах. Это развитие направлено на возможность вывода состояний объекта в реальном времени (например, интерфейсы улучшенный JTAG – EJTAG и NEXUS), а также применение технологии JTAG применительно к объектам с аналоговыми сигналами (стандарт IEEE 1149.4). Применение интерфейсов типа JTAG позволяет, в частности, организовать контроль за техническим состоянием микропроцессорной системы в фоновом режиме, т.е. непосредственно в процессе ее применения по назначению.

В качестве аппаратного ядра внутрисхемных эмуляторов все шире применяются так называемые отладочные кристаллы. Отладочный кристалл представляет собой микросхему, содержащую то же процессорное ядро и порты для внешнего взаимодействия, как и у проверяемой микропроцессорной системы. Однако отладочный кристалл дополнительно содержит узлы, позволяющие осуществлять доступ к внутренним ресурсам ОД извне, обычно по интерфейсу JTAG, и таким образом контролировать и управлять процессом отладки. В результате существенно упрощаются дополнительные средства отладки.

6.5. Сигнатурный анализ

Назначение и способ использования

Микропроцессорные устройства работают на высоких тактовых частотах, их диагностирование может потребовать съема контролируемых сигналов в течение тысяч-сотен тысяч тактов. Последнее обстоятельство приводит к тому, что эталоны, с которыми надо сравнивать результаты измерений, могут иметь длину тысячи-сотни тысяч бит. Похожая проблема возникает при контроле данных, считываемых с винчестера или лазерного диска. Поэтому существенной проблемой при организации производства и технического обслуживания микропроцессорных узлов является проблема уменьшения объема информации о поведении исправного ОД (эталонной), а также объема информации, снимаемой с ОД в процессе диагностирования.

Специально для микропроцессорных устройств разработаны методы, позволяющие проводить контроль с эталонными кодами небольшой длины. На их основе разработаны довольно простые устройства и методики, которые могут использоваться для ручного диагностирования. Фактически они базируются на идее «сжатия» измеряемой длинной последовательности бит в короткий (обычно в несколько десятков бит) код. Подобные методы получили в отечественной технической литературе название методы «компактного тестирования» или «сигнатурного анализа».

Так, при измерениях может подсчитываться число логических переключений выходного контролируемого сигнала (с 0 на 1 или с 1 на 0) и далее сравниваться с указываемым в документации эталонным значением. Другой способ состоит в подсчете контрольной суммы выходных векторов и сравнении с указанным в технической документации значением. Недостаток этих методов в том, что при подсчете числа переключений или контрольной суммы контрольная аппаратура, реализующая эти подсчеты, должна иметь цепи распространения переносов при суммировании, что замедляет процесс проверки и усложняет аппаратуру, а также относительно невысокая достоверность контроля.

Наиболее востребованным из методов компактного тестирования оказался **сигнатурный анализ**, предполагающий преобразование (сжатие) длинного контролируемого двоичного кода $P(x)$ с помощью сдвигового регистра в короткий код фиксированной длины $R(x)$, обычно 16÷32 разряда (рис. 6.11). Этот сжатый код R , называемый **сигатурой**, далее сравнивается с указанным в документации для каждой контролируемой точки эталонным кодом (эталонной сигатурой). При несовпадении полученной и эталонной сигатур можно с очень высокой долей вероятности утверждать, что объект диагностирования (ОД) работает неправильно.

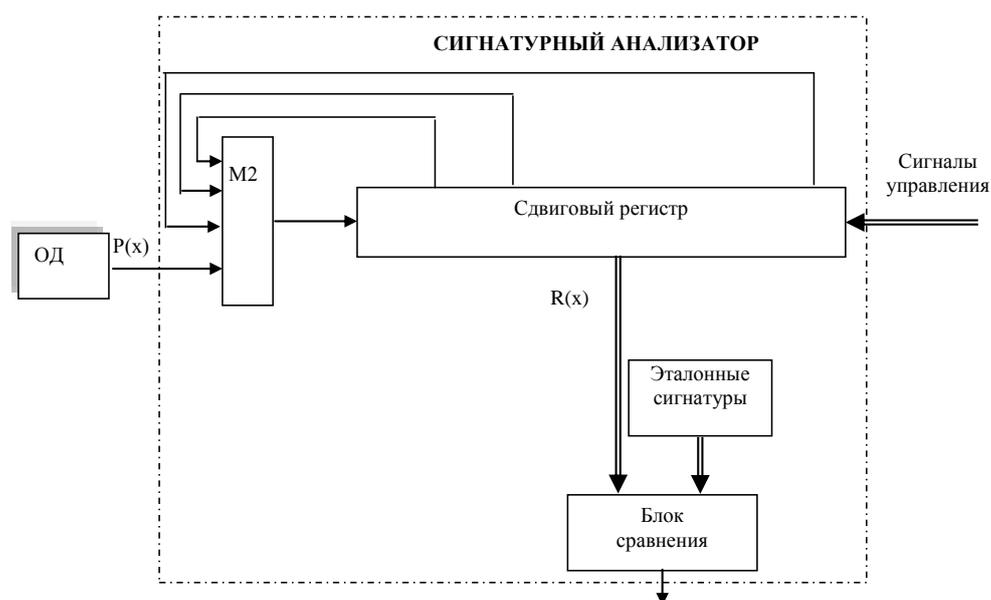


Рис. 6.11. Организация диагностирования с использованием сигнатурного анализатора

Сигнатурный анализатор представляет собой довольно простое устройство, с помощью которого осуществляется деление поступающего кода $P(x)$ на некоторый образующий многочлен Q . Он выбирается с использованием теории циклических избыточных кодов таким, чтобы обеспечивалась максимальная вероятность получения неверной сигнатуры при ошибке в контролируемом коде. Основным элементом сигнатурного анализатора является сдвиговый регистр. Его длина (число разрядов) определяет длину сигнатуры, т.е. остаток R от деления $P(x)$ на образующий многочлен.

При выполнении диагностической процедуры оператор последовательно просматривает в тестовом режиме сигнатуры на контрольных точках ОД. При обнаружении расхождений с указанными в документации сигнатурами он переходит к просмотру сигнатур, двигаясь от точки с неправильной сигнатурой к входу, пока не найдет неисправный элемент. При использовании сигнатурного анализа поиск неисправностей в микропроцессорной аппаратуре становится похожим на обслуживание телевизора, когда мастер, ремонтирующий его с помощью тестера и щупа, сравнивает просматриваемые сигналы с сигналами, указанными на чертеже схемы, а при обнаружении расхождения выполняет просмотр сигналов, двигаясь от выхода к входу схемы (для определения места неисправности).

Теоретические основы сигнатурного анализа

С математической точки зрения сигнатура – результат деления двоичного полинома $P(x)$ на некоторый полином $Q(x)$. Полином $P(x)$ описывает поток бит с контрольной точки (поступает на вход сигнатурного анализатора), а $Q(x)$ – определяется числом разрядов сдвигового регистра и разрядами, к которым подключаются линии обратных связей. После подачи на вход анализатора всех бит $P(x)$ состояние разрядов регистра представляет собой остаток $R(x)$ от результата деления по формуле

$$P(x) = q(x) Q(x) \oplus R(x),$$

где $q(x)$ – целая часть (частное) от деления $P(x)$ на $Q(x)$. Код $R(x)$ это и есть сигнатура. Ее длина равна числу разрядов регистра. Обобщенная схема сигнатурного анализатора, реализующего полином $Q(x) = x^3 + x^2 + 1$, показана на рис. 6.12. На схеме каждый элемент памяти $T1$, $T2$ и $T3$ реализует один из разрядов регистра, $M2$ – элементы сложения по модулю 2 ($\text{mod } 2$). Цепи тактирования работы регистра на рисунке не показаны.

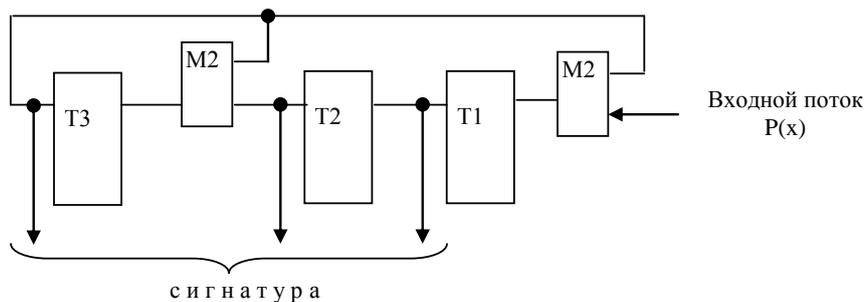


Рис. 6.12. Сигнатурный анализатор, реализующий деление на полином $Q(x) = x^3 + x^2 + 1$

В данном случае сигнатура состоит из трех разрядов, значения которых определяются состояниями элементов T1, T2 и T3 памяти регистра. Заметим, что схема сигнатурного анализатора на рис. 6.12, фактически является схемой получения циклического кода.

На практике применяются также сигнатурные анализаторы, в схемах которых все выходы обратных связей заводятся прямо на вход регистра через элемент mod 2 (рис. 6.13). Принято говорить, что сигнатурный анализатор, имеющий схему, как на рис. 6.12, реализует метод деления на полином, а на рис. 6.13 – метод свертки.

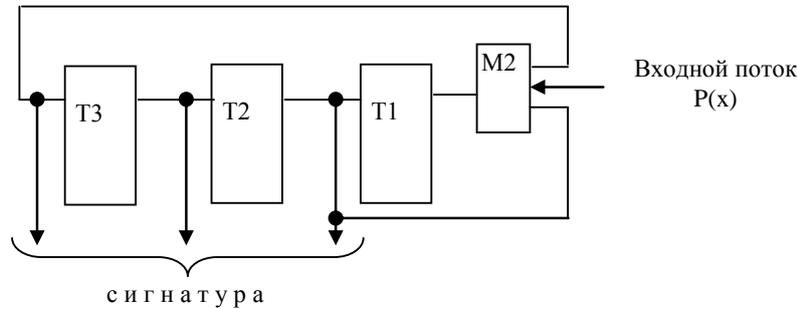


Рис. 6.13. Сигнатурный анализатор, реализующий деление на полином $W(x) = x^3 + x^1 + 1$

В результате работы анализатора на рис. 6.13, полученная сигнатура $C(x) \neq R(x)$. Однако математически доказано, что существует однозначная связь между сигнатурами обеих схем, если полином $Q(x)$ является обратным по отношению к полиному $W(x)$, т.е. $Q(x) = W^{-1}(x)$. Связь между остатками (сигнатурами) таких двух схем определяется выражением:

$$R(x) = C(x) \cdot \begin{vmatrix} a_m & 0 & 0 & 0 & 0 \\ a_{m-1} & a_m & 0 & 0 & 0 \\ \dots & a_{m-1} & a_m & 0 & 0 \\ a_2 & \dots & a_{m-1} & a_m & 0 \\ a_1 & a_2 & \dots & a_{m-1} & a_m \end{vmatrix}.$$

В матрице выше a_i – коэффициенты при слагаемых полинома $Q(x)$. В нашем примере он имеет вид: $Q(x) = a_3 x^3 + a_2 x^2 + a_1 x^1 + 1$, где $a_3 = a_2 = 1$ и $a_1 = 0$. Полином $Q(x) = W^{-1}(x)$, т.е. является обратным по отношению к $W(x)$ (заметим, что в обратном полиноме коэффициенты от a_{m-1} до a_1 имеют инверсное значение по отношению к исходному). Тогда связь между остатками (сигнатурами) схем на рис. 6.12 и 6.13 определяется выражением:

$$R(x) = C(x) \cdot \begin{vmatrix} a_3 & 0 & 0 \\ a_2 & a_3 & 0 \\ a_1 & a_2 & a_3 \end{vmatrix} = C(x) \cdot \begin{vmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{vmatrix}.$$

Схема и особенности функционирования одноканального анализатора

Основным элементом сигнатурного анализатора является сдвиговый регистр и один или несколько элементов, реализующих функцию $\text{mod}2$ (рис. 6.13). Элементы по $\text{mod}2$ используются для создания обратных связей. На их входы заводятся выходы отдельных триггеров (разрядов Q_i) регистра. Номера разрядов, с которых берутся отводы, зависят от выбранного для реализации порождающего полинома. На рисунке 6.14 показана схема, реализующая деление поступающей для контроля последовательности P бит на многочлен $x_{16}+x_{14}+x_1+1$. Соответственно на элемент $\text{mod}2$ заведены обратные связи с разрядов 1, 14 и 16 (на схеме через Q_i обозначен i -й разряд сдвигового регистра).

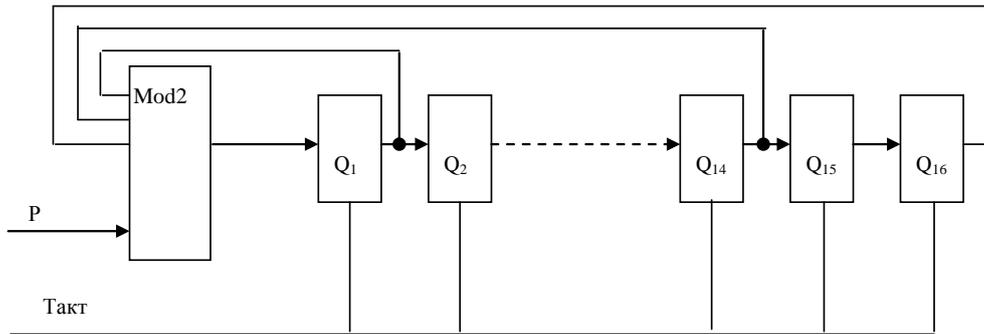


Рис. 6.14. Принцип построения схемы сигнатурного анализатора

Один из первых серийных сигнатурных анализаторов был выпущен фирмой Хьюлетт-Паккард. Принципы его построения фактически стали неофициальным стандартом на сигнатурные анализаторы. Это был портативный прибор, содержащий 16-разрядный регистр сдвига с сумматором по $\text{mod} 2$ на входе (рис. 6.15). Он реализует метод свертки.

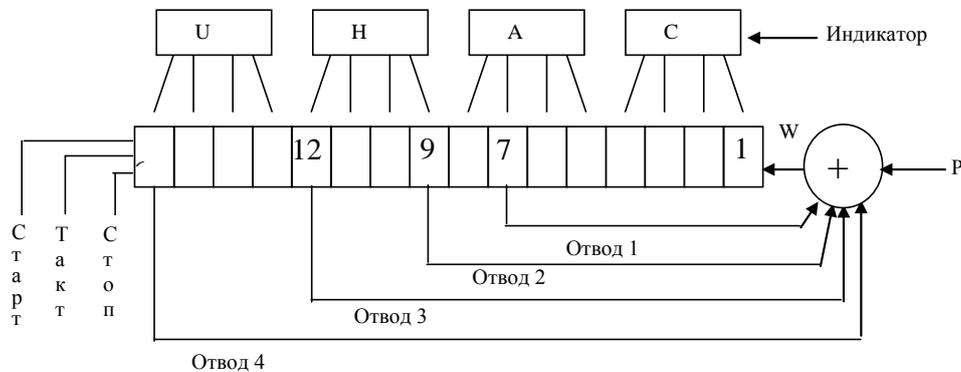


Рис. 6.15. Структурная схема сигнатурного анализатора фирмы Хьюлетт-Паккард

Принцип его работы можно описать следующим образом. С помощью щупа вход сигнатурного анализатора присоединяется к контролируемому выходу ОД. Процесс последующего получения сигнатуры предполагает задание некоторого фиксированного интервала времени – «окна», соответствующего определенному числу тактовых сигналов, а следовательно, и такому же числу бит в входной последовательности Р, поступающей на вход анализатора.

Снятие с контролируемой точки последовательности бит Р, подача их на вход элемента mod 2 и работа сдвигового регистра синхронизируются одной и той же серией тактовых сигналов. Анализатор содержит схему формирования «окна», на входы которой поступают сигналы ПУСК, ТАКТ, СТОП. По сигналу ПУСК биты последовательности Р поступают на вход анализатора, а сигнал синхронизации ТАКТ от проверяемого объекта сдвигает данные в регистре. По сигналу СТОП на сдвиговом регистре остается 16-битный остаток, который отображается на 4-х сегментных индикаторах в виде четырех шестнадцатиричных цифр. Полученный на индикаторах код (сигнатура) сравнивается с эталонной сигнатурой для данной контролируемой точки. Эталонные сигнатуры могут быть получены предварительно путем проверки с помощью сигнатурного анализатора заведомо исправного ОД или расчетным путем (моделированием) на ЭВМ.

Фирма Hewlett-Packard спроектировала свой анализатор для ручного применения. Чтобы избежать ошибок при просмотре символов, отображаемых на индикаторах, была принята нестандартная схема обозначения шестнадцатиричных кодов на сегментных индикаторах: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, А, С, F, H, P, U (в порядке возрастания значений).

В таблице 6.4 показан процесс формирования сигнатуры анализатора на рис. 6.15 для входной последовательности Р = 1111110000111111111, при «окне» длительностью 19 тактов. При этом полагается, что начальное состояние регистра (Такт «0») – все нули.

На протяжении первых семи тактов по цепям обратной связи поступают «нули» и в регистр вдвигаются без каких-либо изменений первые 7 бит входной последовательности (сигнал на входе регистра обозначен в табл. 6.4 символом W). После седьмого такта по обратной цепи от первого отвода (седьмой разряд регистра) на сумматор по модулю 2 поступает 1, которая в восьмом такте заносится в первый разряд регистра (в восьмом разряде входной последовательности Р стоит 0). Аналогичным образом работает анализатор в последующие такты.

После закрытия «окна» (после 19 тактов) в регистре образуется 16-битный остаток 1111 1101 1010 1011, соответствующий сигнатуре УНАС.

Метод сигнатурного анализа обладает высокой достоверностью, которая в данном случае определяется вероятностью того, что различающиеся двоичные векторы, поступающие на вход анализатора, имеют неодинаковые сигнатуры. Любая одиночная ошибка в контролируемом векторе обнаруживается с вероятностью, равной единице.

Формирование сигнатуры

Такт	P	W	Состояние разрядов регистра															
			16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
3	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
4	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
5	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
6	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
7	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0
8	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1
9	0	1	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1	1
10	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1	1	0
11	1	1	0	0	0	0	0	1	1	1	1	1	1	0	1	1	0	1
12	1	1	0	0	0	0	1	1	1	1	1	1	0	1	1	0	1	1
13	1	0	0	0	0	1	1	1	1	1	1	0	1	1	0	1	1	0
14	1	1	0	0	1	1	1	1	1	1	0	1	1	0	1	1	0	1
15	1	0	0	1	1	1	1	1	1	0	1	1	0	1	1	0	1	0
16	1	1	1	1	1	1	1	1	0	1	1	0	1	1	0	1	0	1
17	1	0	1	1	1	1	1	0	1	1	0	1	1	0	1	0	1	0
18	1	1	1	1	1	1	0	1	1	0	1	1	0	1	0	1	0	1
19	1	1	1	1	1	1	1	1	0	1	1	0	1	0	1	0	1	1

При числе ошибок более одной вероятность обнаружения ошибок¹ равна $1 - 1/2^n$, где n – число разрядов сигнатурного анализатора. При $n = 16$ эта вероятность равна 0,99998, иными словами, можно ожидать, что не более 0,002% различных последовательностей P будут иметь одинаковые сигнатуры.

При необходимости, для повышения достоверности результатов сигнатурного анализа, можно увеличить длину регистра. Например, уже при $n = 20$ веро-

¹ При равной вероятности каждой возможной ошибки.

ятность обнаружения многократной ошибки в проверяемой двоичной последовательности составляет 0,999999.

Совпадение сигнатур возможно только в том случае, когда при формировании сигнатуры последующие ошибки в выходном векторе компенсируют влияние предыдущих ошибок (искажений). Можно показать, что при двойной ошибке компенсация имеет место, когда эти ошибки разделены 2^{n-1} нулями, что является маловероятным событием.

Возможно, что значения некоторых бит Р зависят от исходных состояний элементов памяти или состязаний в контролируемой схеме, либо в точке, с которой снимается Р, возможно высокоимпедансное состояние. В первом случае необходимо задавать несколько эталонных сигнатур (всего 2^k , где k – число неопределенных бит). Во втором случае применяют дополнительные схемы преобразования Р в W. Например, если сигнал Р имеет высокий импеданс, соответствующий бит в W (рис. 6.15) сохраняет предыдущий логический уровень (0 или 1). Используемые преобразователи просты и основываются на том факте, что при высоком импедансе напряжение сигнала находится между уровнями логического 0 и 1.

Многоканальные сигнатурные анализаторы

Современные ОД, как правило, содержат несколько контролируемых выходов. Поэтому были разработаны многоканальные анализаторы, которые могут одновременно снимать сигналы с нескольких контрольных точек. Особенно это актуально для микропроцессорных ОД, в которых из-за шинной организации зачастую надо контролировать не просто значения сигналов, но и их сочетания в каждый момент времени. В принципе, можно сделать устройство, состоящее из набора одноканальных анализаторов, но это усложняет контрольную аппаратуру.

Однако оказалось возможным создание эффективных многоканальных анализаторов, которые можно рассматривать как некоторые модификации обычных одноканальных анализаторов. При этом число разрядов сигнатуры получается таким же, как и в одноканальном анализаторе, а вероятность необнаружения ошибки может быть даже несколько выше.

На рисунке 6.16 показана схема пятиканального сигнатурного анализатора. Он может формировать единую 16-разрядную сигнатуру, получая сигналы одновременно с пяти контрольных точек y_1, y_2, y_3, y_4, y_5 . Доказано, что при некоторых допущениях вероятность необнаружения ошибки этим анализатором:

$$P = (2^{qk-m} - k) / (2^{qk} - k) \approx 1/2^m,$$

где k – количество контролируемых каналов (т.е. число контролируемых сигналов, одновременно поступающих на анализатор);

q – длина контролируемых последовательностей бит;

m – число разрядов регистра.

Промышленность серийно выпускает различные по конструкции сигнатурные анализаторы. Они могут быть встроены в универсальные системы диагностирования либо выполняться в виде отдельных автономных устройств (рис. 6.17).

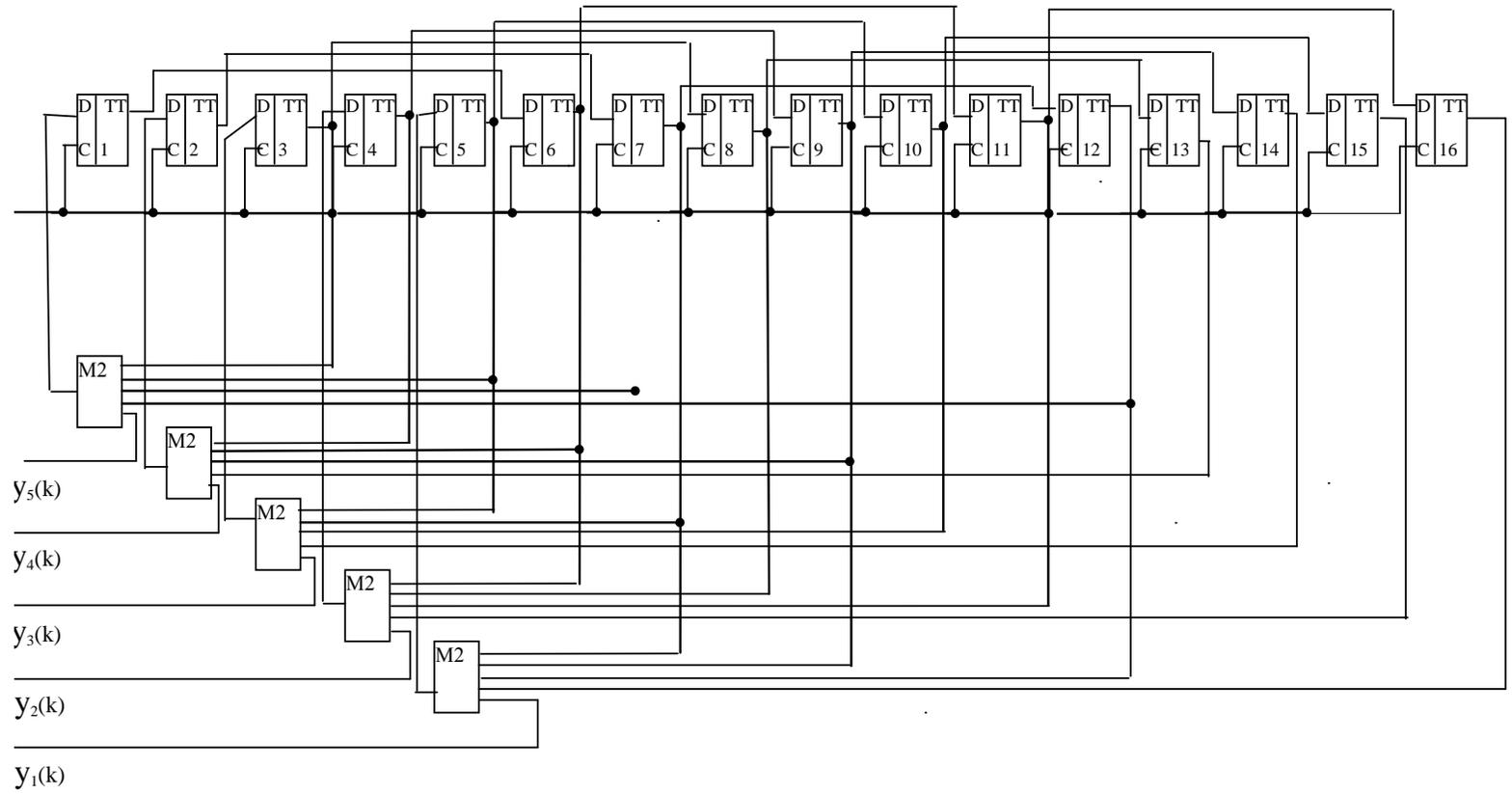


Рис. 6.16. Схема 5-канального сигнатурного анализатора



Рис. 6.17. Примеры сигнатурных анализаторов, выполненных в виде автономных устройств

6.6. Диагностические платы

Назначение и конструкции

Современные ПЭВМ имеют довольно совершенную и эффективную систему самоконтроля и самодиагностики. При ее срабатывании на монитор выдается информация об обнаруженном отказе в работе оборудования. Однако возможно, что из-за неисправности нарушен вывод или монитор вообще не функционирует. Тогда для определения места неисправности используют специальные аппаратные средства контроля – диагностические платы. Даже в случаях, когда большая часть оборудования компьютера работоспособна и операционная система загружается, диагностическая плата оказывается очень полезной при диагностике таких неисправностей, которые не могут быть точно указаны или обнаружены стандартными программными утилитами.

Обычно диагностическая плата устанавливается в свободный разъем материнской платы ЭВМ (рис. 6.18а, в). Однако имеются варианты ее исполнения встроенной в материнскую плату (рис. 2.18б) или в виде отдельного внешнего устройства, которое кабелем подключается к одному из разъемов (портов) контролируемой ЭВМ (рис. 6.18г).

В основу работы диагностической платы положены некоторые особенности исполнения процедуры (программы) POST (см. п. 6.2). Поэтому диагностическую плату часто называют POST-картой.

POST выполняется при каждом включении питания компьютера до начала загрузки операционной системы. Эта же процедура выполняется также при нажатии на кнопку RESET. Программа POST хранится в микросхеме ROM BIOS.

Основной целью процедуры POST является проверка правильности функционирования основных устройств компьютера (таких как процессор, память, видеоконтроллер, клавиатура, гибкий и жесткий диски и т.д.) и необходимая для дальнейшей работы компьютера начальная инициализация системных устройств перед загрузкой операционной системы (всего около 100 операций).

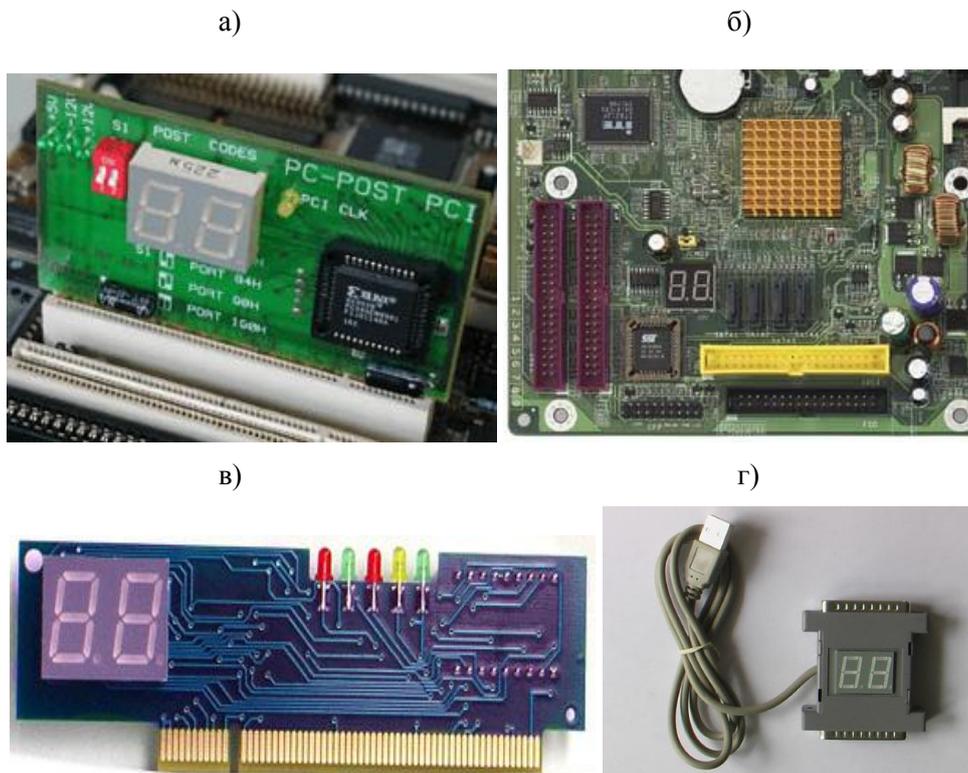


Рис. 6.18. Примеры конструктивного исполнения POST-карт

Перед началом каждой операции процедура POST генерирует специальный код размером один байт (от 00 до FFh), называемый POST-кодом, и записывает его значение в специальный диагностический порт в адресном пространстве устройств ввода/вывода (в компьютерах типа IBM PC/AT для этого выделен порт с адресом 80h, который используется и в большинстве других IBM-совместимых компьютеров). В случае обнаружения неисправности в тестируемом устройстве процедура POST просто «зависает», а POST-код, предварительно выведенный в диагностический порт, однозначно определяет операцию, на которой произошло «зависание» процедуры POST, а следовательно, и неисправный узел компьютера.

POST-коды и их отображение

POST-карта представляет собой относительно простое устройство, предназначение которого через некоторую шину считывать из диагностического порта и выводить на светодиодные или сегментные индикаторы POST-коды. В нормальной ситуации цифры кодов быстро бегут на индикаторах, завершаясь значением FF, что говорит об успешном прохождении тестирования, инициализации и о готовности к загрузке операционной системы. Однако если какое-то устройство компьютера вызовет остановку программы диагностики, то на индикаторах будет постоянно гореть POST-код последнего отработанного шага, указывающий на операцию, а следовательно, и на то устройство, при проверке ко-

торого произошла остановка программы. POST-код расшифровывается по таблице кодов, которую составляет производитель BIOS. Оригинальные и достоверные таблицы POST-кодов можно найти в эксплуатационной документации на соответствующий компьютер и сайтах производителей BIOS.

Дополнительно POST-карта может выводить на светодиоды состояния линий «адрес/данные» и некоторых линий с управляющими сигналами.

Однако следует иметь в виду, что таблицы POST-кодов различны для BIOS разных производителей, а в связи с постоянным совершенствованием микропроцессоров и чипсетов несколько отличаются даже для различных версий BIOS одного и того же производителя.

В качестве примера в табл. 6.5 приведено несколько POST-кодов из справочного руководства «Персональные компьютеры семейства Compaq». При работающем мониторе код и его краткая текстовая интерпретация выводятся на экран (собственно POST-код и его текстовое пояснение приведены в левом столбце табл. 6.5).

Таблица 6.5

Сообщения об ошибках и POST-коды

Код ошибки	Возможный источник неисправности	Рекомендуемое действие
1	2	3
101 -Rom Error (Ошибка ПЗУ)	Контрольная сумма системного ПЗУ	Обратитесь к представителю компании-производителя компьютера по обслуживанию
101 -I/O Rom Error (ошибка ПЗУ ввода-вывода)	Контрольная сумма дополнительных ПЗУ	Обратитесь к представителю компании-производителя компьютера по обслуживанию
102 -System Board Failure (сбой системной платы)	Модули прямого доступа к памяти, таймеры и т.п.	Замените плату процессора. Вызовите утилиту Computer Setup или утилиту Windows
162 -System Options Error (ошибка настройки системы)	Отсутствие накопителя на дискетах или несоответствие типа дисководов	Вызовите утилиту Computer Setup или утилиту Windows
164 -Memory Size Error (ошибка размера памяти)	Неверно задана конфигурация памяти	Вызовите утилиту Computer Setup или утилиту Windows
205 -Memory Error (ошибка памяти)	Ошибка КЭШ-памяти	Вызовите утилиту диагностики
301 -Keyboard Error (ошибка клавиатуры)	Сбой клавиатуры	Убедитесь в отсутствии зажатых клавиш или нечто похожего. При необходимости заменить клавиатуру

1	2	3
501 -Display adapter Failure (Сбой адаптера дисплея)	Контроллер видеосигнала дисплея	Замените модуль DRAM
702 -A coprocessor has been detected that is not reported in CMOS. (Обнаружен сопроцессор, не представленный в CMOS)	Ошибка в настройке конфигурации	Вызовите утилиту Computer Setup или утилиту Windows
1771 -Primary Disk Port Address Assignment Conflict (Конфликт назначения адреса порта основного диска)	Контроллерам внутреннего и внешнего накопителя на жестких дисках присвоен основной адрес	Вызовите утилиту Computer Setup или утилиту Windows

Наиболее эффективно используется POST-карта в том случае, когда на материнской плате сохранена работоспособность устройств, обеспечивающих выполнение самой процедуры POST: процессора, системы синхронизации, ROM BIOS, узлов формирования питающих напряжений, контроллеров внутрисистемных шин и шин расширения. В иных случаях ее возможности существенно ограничиваются. Тогда помочь могут режимы работы платы, предполагающие замену стандартной микросхемы ROM BIOS проверяемой ПЭВМ на специальную БИС.

Пример схемы POST-карты

В простейшем варианте POST карта просто отображает POST-код на светодиодах или сегментном индикаторе. Структурная и принципиальная схемы такой простой карты для случая подключения к ISA-шине приведены на рис. 6.19 и 6.20. На структурной схеме индикатора POST-кодов (рис. 6.19):

ISA-bas (ISA-шина);

RG – восьмиразрядный параллельный регистр, предназначенный для съема с шины (контакты Data0 – Data7) и хранения очередного значения POST-кода;

Reset-сигнал сброса регистра RG;

DC1 – дешифратор разрешения записи в регистр RG;

IOWR – сигнал разрешения чтения данных с шины;

DC2 – дешифратор-преобразователь двоичного POST-кода в код семи сегментного индикатора;

HG – двухразрядный семисегментный индикатор для отображения значения кода ошибки в виде шестнадцатеричных символов – 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

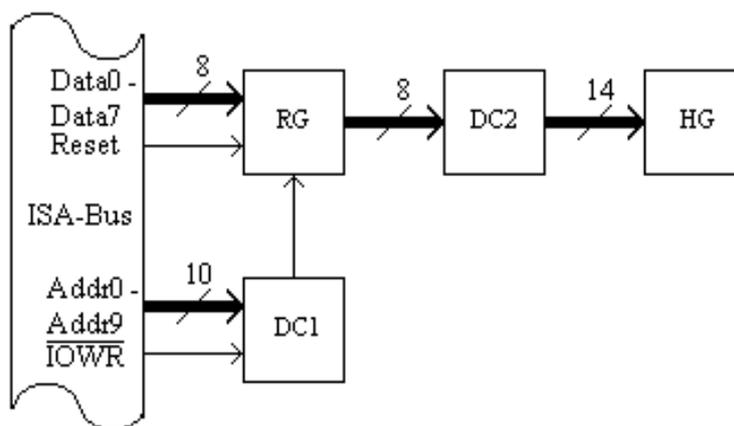


Рис. 6.19. Структурная схема индикатора POST-кодов

Дешифратор разрешения записи в регистр RG (элемент DD1 на рис. 6.20) собран на элементах DD2.1, DD2.3, DD2.4, DD3.1 и DD3.2. На его адресные входы поступают адресные сигналы Addr0, ..., Addr9 с ISA-шины, а также сигнал IOWR разрешения записи (контакт B13 шины ISA). Дешифратор адреса формирует сигнал «1» на выводе 11 регистра DD1 при появлении на шине ISA адреса 80h и низкого уровня сигнала IOWR. Адрес 80h формируется на шине каждый раз, когда BIOS выдает очередной POST-код на шину. При поступлении «1» на вход WR регистра разрешается занесение в него информации с линий данных Data0 – Data7, т.е. в регистр DD1 заносится POST-код.

Сигнал Reset обнуляет регистр DD1 каждый раз, когда происходит включение или перезапуск компьютера.

Дешифратор-преобразователь двоичного кода в код семисегментного индикатора выполнен на двух соответствующим образом запрограммированных микросхемах ПЗУ DD4 и DD5. Поступивший из регистра DD1 код преобразуется в соответствии с запрограммированным алгоритмом и отображается на двухразрядном семисегментном индикаторе HG1.

Напряжение питания на плату индикатора поступает с контактов разъема XP1 шины ISA – B3, B29 (+5V) и B1, B31 (земля). В цепи питания каждой микросхемы установлен помехоподавляющий конденсатор (C1-C5).

Функции современных диагностических плат

Современные диагностические платы реализуют значительно более сложные функции. Они, в частности, могут сами инициировать некоторые собственные тестовые и управляющие сигналы, что позволяет выполнять проверки, которые невозможны в обычном режиме работы ЭВМ. Производители вычислительной техники наладили выпуск ряда специализированных диагностических плат, предназначенных для тщательного тестирования различных типов шин и портов (PCI, USB, LPT, COM и др.), внешних устройств ЭВМ (дисководов для жестких, лазерных, гибких дисков и др.), отдельных типов процессоров и т.п.

вид слота ISA сверху XP1

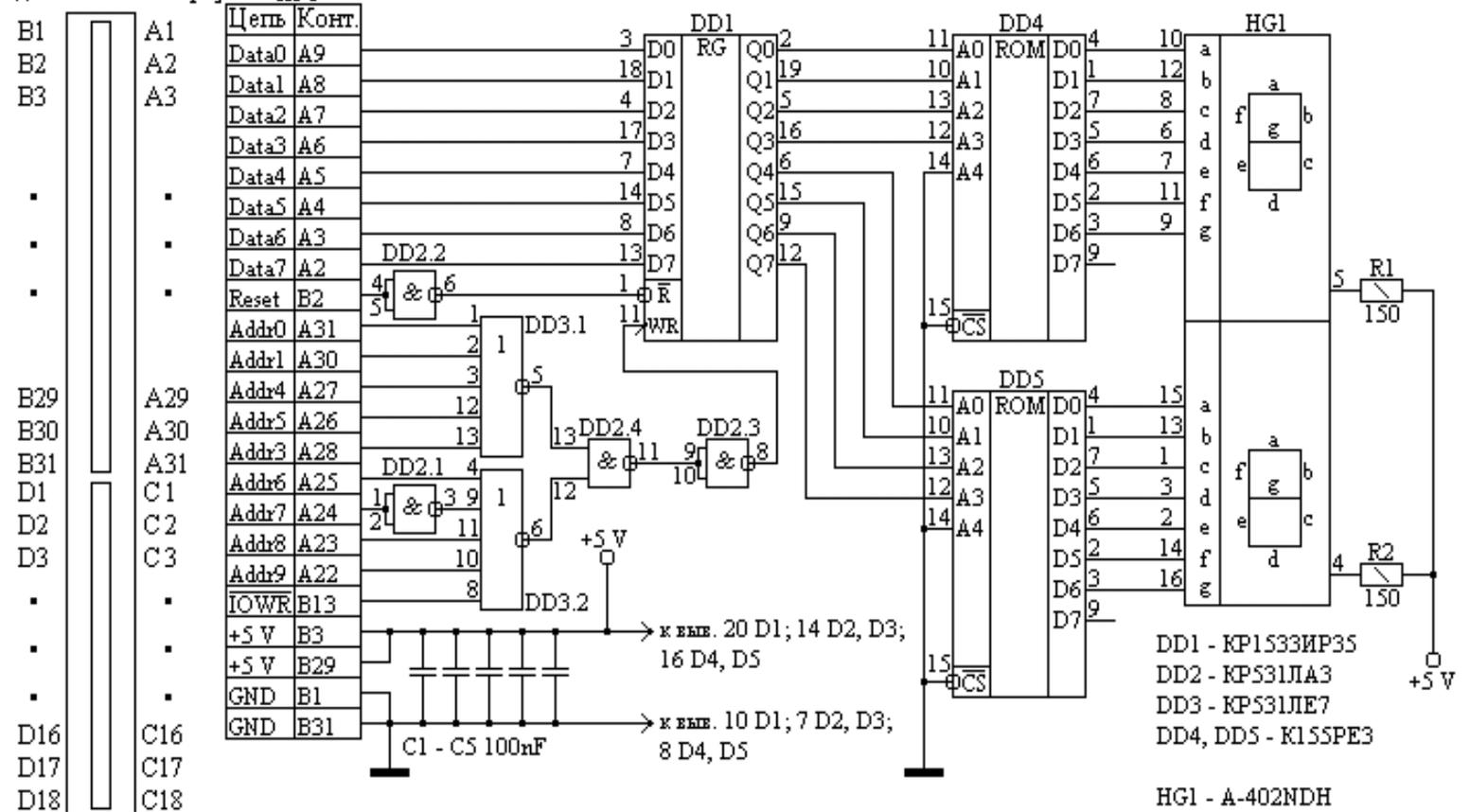


Рис. 6.20. Принципиальная электрическая схема индикатора POST-кодов

При этом для углубленной диагностики они могут предполагать замену стандартной микросхемы BIOS на специально предназначенную для диагностирования. Программа POST исходной BIOS останавливает работу процессора при первой же ошибке в его работе. Специальная BIOS содержит драйверы, модифицированные с целью устранения остановов в процессе диагностирования, что позволяет даже при ошибках выполнения отдельных проверок продолжить диагностирование и получить более подробную информацию о состоянии ПЭВМ.

В обобщённом виде типовая последовательность выполнения процедуры POST состоит из следующих основных этапов:

- тест процессора;
- проверка контрольной суммы ПЗУ (ROM) BIOS;
- проверка и инициализация контроллера прямого доступа к памяти, контроллера прерываний и системного таймера;
- предварительная проверка памяти (проверка регенерации и первых 64 кб ОЗУ);
- загрузка векторов прерывания;
- инициализация видеоконтроллера;
- тестирование всего объёма ОЗУ;
- тестирование клавиатуры;
- тестирование CMOS памяти;
- инициализация LPT- и COM-портов;
- инициализация и тестирование контроллера гибкого диска;
- инициализация и тестирование контроллера жесткого диска;
- поиск расширений ROM BIOS и их инициализация;
- вызов процедуры загрузки операционной системы (Bootstrap).

Рассмотрим особенности функционирования диагностической платы на примере платы PHD PC I2 (рис. 6.21). Она используется в технических центрах для сложного и квалифицированного тестирования систем, имеющих шину PCI (при процессорах Pentium II и выше), AMD и VIA процессоры поддерживаются полностью.

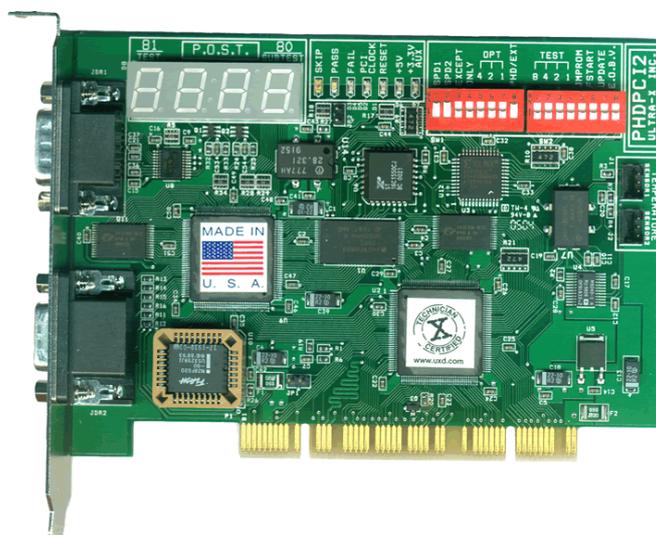


Рис. 6.21. Диагностическая плата PHD PC I2

Плата имеет несколько режимов работы:
 диагностики,
 принудительного старта,
 перехода на дополнительный ROM.

В **режиме диагностики** плата PHD позволяет осуществить тестирование на уровне компонентов материнской платы и вывести результаты на монитор компьютера (рис. 6.22). Режим используется, когда POST выполняется без проблем.

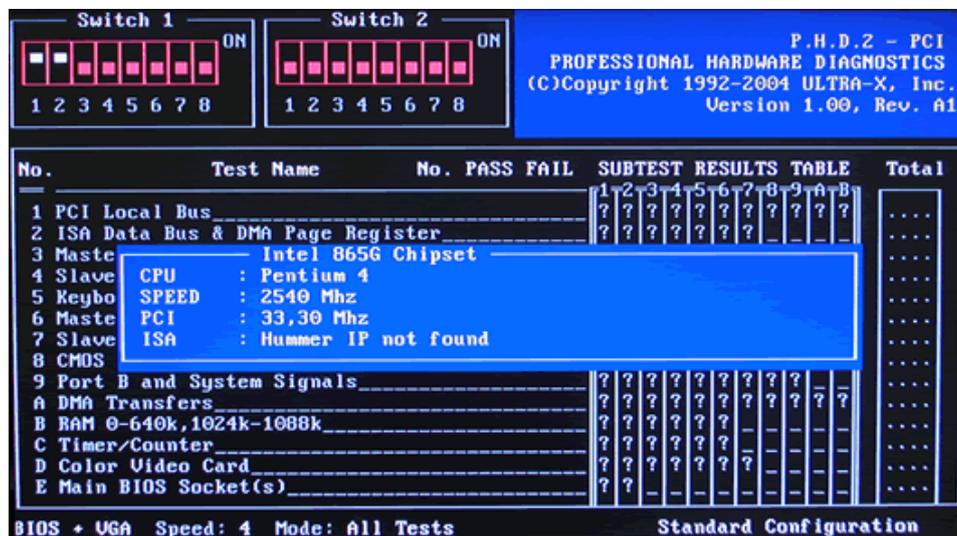


Рис. 6.22. Отображение результатов тестирования

В данном режиме на мониторе можно вывести текстовое сообщение о характере и месте неисправности, что снимает необходимость в ручной дешифровке кода. PHD организует выполнение набора диагностических тестов, позволяющих более тщательно проверить:

- локальную шину PCI,
- регистры прямого доступа к памяти (DMA),
- контроллер DMA,
- контроллер клавиатуры и клавиатуру,
- контроллер прерывания,
- CMOS RAM и часы реального времени,
- таймер,
- системную часть RAM,
- видеоадаптер.

Если загружается операционная система, то плату PHD можно использовать вместе со специальной управляющей программой. Ее меню позволяет задавать наборы тестов для тщательной диагностики отдельных устройств ЭВМ. Так, через пункт **System Board** (системная плата) запускаются тесты, которыми проверяются основные функции системной платы: процессор, устройство для выполнения операций с плавающей точкой, каналы прямого доступа к памяти, кон-

троллеры прерывания, шина PCI, функция «plug and play», звуковая плата и динамик; **Peripherals** (периферийное оборудование) – тесты, проверяющие подсистему ввода/вывода: порты RS-232, LPT (рис. 6.23) и USB; каналы связи с модемом, принтером, звуковой канал, Ethernet, а также PCI модем; **Video** (видео) – тесты проверки видеоадаптера (тестируется: графический адаптер, монитор, Видео Ram); **Burn-in** (испытание на отказ) – запускается режим длительного тестирования, например для определения наработки на отказ, и др.

Режим принудительного старта используется, когда POST проходит со сбоями (т.е. когда загрузка системы происходит через раз или не осуществляется вовсе). К таким случаям относятся ситуации, когда система просто умерла, зависла в процессе выполнения POST или периодически дает сбой и т.п. В этом режиме PHD сама инициализирует систему, и тестирование осуществляется независимо от BIOS. При этом «изолируются» неисправные компоненты, которые препятствуют загрузке системы.

Test Mode	Current	Pass	Fail
LPT Type Detection.....			
Ext. Loopback IRQ Generation...			
LPT General Control Register...			
SPP/BPP/EPP Data Port.....			
BPP Bi-Directional Operation...			
EPP Address Register.....			
EPP Data Ports 0-3.....			
ECP Control Register.....			
ECP Internal FIFO Sizing.....			
ECP FIFO Testing.....			
ECP IRQ Assignment.....			
ECP ISA IRQ Line Monitoring...			
ECP Configuration Registers...			
External Loopback.....			

Рис. 6.23. Форма вывода результатов тестирования LPT порта

Режим перехода на дополнительный ROM очень похож на режим принудительного старта. Единственное отличие состоит в том, что стандартная микросхема ROM BIOS как бы заменяется постоянным запоминающим устройством, установленным прямо на диагностической плате (отсюда и название режима). В этом случае PHD инициализирует систему, используя свой собственный BIOS. В режиме принудительного старта PHD требует доступности специальных сигналов на чипсете, чтобы обойти стандартный BIOS и обеспечить контроль шины. Если эти сигналы не предусмотрены схемой материнской платы или чипсета, то PHD генерирует эти сигналы искусственно и получает доступ к шине.

Перед работой плату PHD надо настроить с помощью специальных переключателей, установленных на плате в правом верхнем углу. С их помощью можно: задать совокупность тестов, которые должны быть осуществлены, скорость выполнения тестов; установить мониторинг температуры; вывести результаты тестирования на параллельный порт; направить результаты на удаленную систему.

Плату PHD можно использовать для многократного выполнения специальных тестов, предназначенных для обнаружения причин сбоев или просто тренировочных прогонов после ремонта, для проверки на совместимость и соответствие стандартам шины PCI аппаратной части и системных функций программного обеспечения.

6.7. Средства отладки микропроцессорных систем

Организация производства микропроцессорных систем и ПЭВМ

Многие производители электронного оборудования специализируются на массовом производстве отдельных стандартных компонент ПЭВМ и микропроцессорных управляющих систем, что позволяет им обеспечить низкую себестоимость производства. Поэтому большинство производителей микропроцессорных систем и ПЭВМ, по сути, занимаются их сборкой, приобретая на других предприятиях необходимые комплектующие. При этом они обычно сами изготавливают кабели, некоторые нестандартные или относительно простые компоненты, выполняют монтаж и отладку собранного изделия. Для наиболее ответственных применений собранные изделия или их составляющие могут проходить механические и климатические испытания.

Заметим, что производители компонент добились очень высокой надежности поставляемой продукции. Поэтому сборкой и поставкой, например ПЭВМ, для бытовых и офисных приложений часто занимаются мелкие фирмы не имеющие собственного производства и испытательного оборудования.

При изготовлении микропроцессорных изделий необходимо контролировать едва ли не каждую техническую операцию, чтобы своевременно выявить неработоспособные элементы и узлы. Причем «интеллект» стендового оборудования должен быть не ниже «интеллекта» проверяемых устройств – самой ПЭВМ и ее составных частей.

На рисунке 6.24 показана укрупненная схема технологического маршрута изготовления микропроцессорной системы, в которой показаны основные этапы изготовления и контроля. На ней цифрами обозначены:

1. Подготовка печатающего устройства, жесткого и гибкого дисков, клавиатуры, дисплея и других компонент к сборке.
2. Изготовление и проверка кабелей, других компонент, которые изготавливаются на предприятии.
3. Программирование ПЗУ, сборка и проверка вычислительного блока.
4. Сборка системы в целом, инсталляция базового программного обеспечения.
5. Первая проверка работоспособности ПЭВМ.
6. Технологический прогон.
7. Копирование с эталонных дисков на рабочие диски программного обеспечения, оговоренного контрактом на поставку.
8. Вторая проверка работоспособности.
9. Передача изделия на склад готовой продукции.

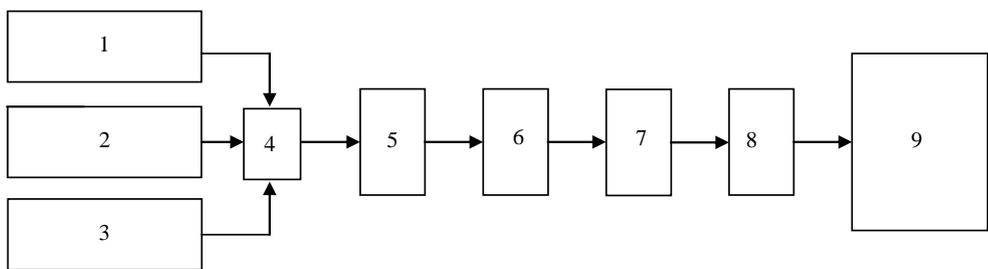


Рис. 2.24. Схема технологического процесса изготовления микропроцессорной системы

В общем случае, технологический маршрут включает десятки операций контроля, в которых задействуется разнообразное оборудование.

Состав применяемых на конкретном предприятии стандов существенно зависит от требований заказчиков, финансовых возможностей, культуры производства и схемы организации производства (в частности, от сочетания покупных и самостоятельно изготовленных компонент).

На рисунке 6.25 показан состав стандовой аппаратуры, используемой в процессе изготовления микропроцессорной системы и для контроля её узлов.

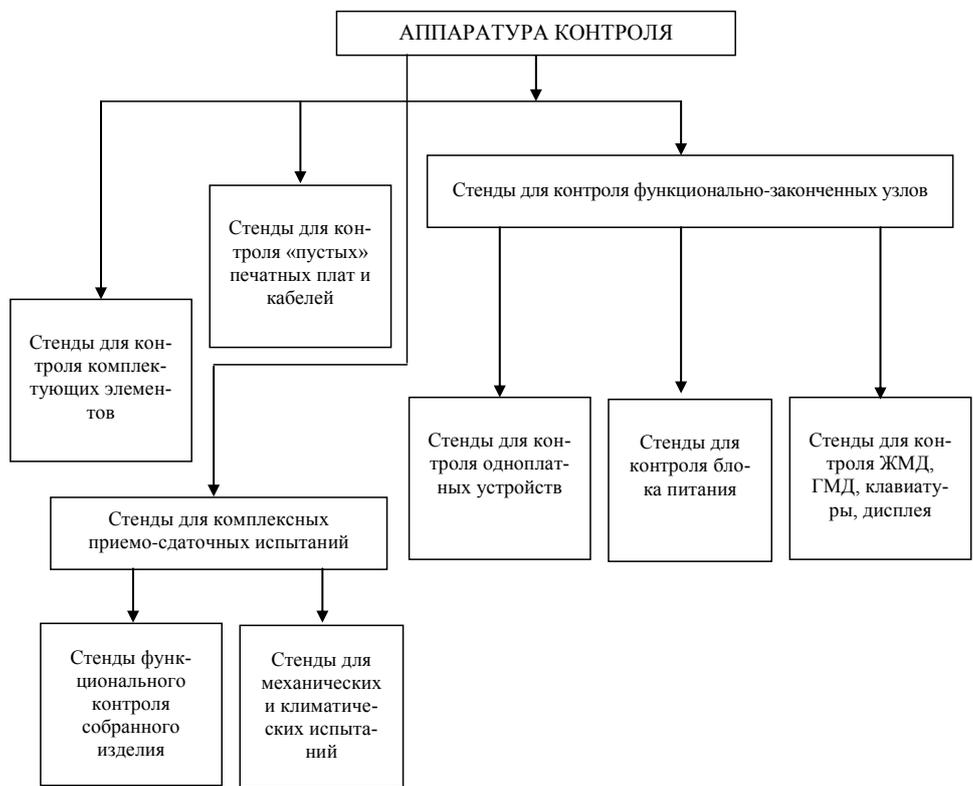


Рис. 6.25. Стенды производственного контроля

Назначение и условия применения средств отладки микропроцессоров

Микропроцессор – элемент почти любой электронной схемы современного промышленного изделия (от игрушки и бытовых приборов до роботов).

Микропроцессорная система – программно управляемая система. Ее использование, в большинстве случаев, предполагает, что каждый конкретный пользователь работает с собственным набором программ, сам определяет и создает конфигурацию не только программных, но и аппаратных средств (объем и виды памяти, состав внешних устройств). Микропроцессорная система – это система реального времени, т.е. окончательное мнение о ее работоспособности можно получить только после проверки функционирования во всех реальных режимах при рабочих скоростях работы аппаратуры. Даже если аппаратная часть исправна, при возникновении отказа очень часто невозможно сказать из-за чего это произошло. Возможно, причина в некорректном использовании программой ресурсов микропроцессорной системы, но не исключено, что в данном режиме функционирования стал проявляться ранее не выявленный дефект программы¹.

Достаточно условно микропроцессоры можно разделить на две группы: универсальные и специализированные. Первые используются в универсальных вычислительных устройствах, к которым относятся компьютеры. Вторые обычно имеют меньшие вычислительные возможности, они дешевле и приспособлены для создания различных систем управления бытовыми и промышленными приборами, механизмами и системами. К числу вторых, в частности, относятся так называемые микроконтроллеры. Это микропроцессоры со специализированными интерфейсами, упрощающими подключение к управляемым от них приборам, механизмам и датчикам.

Один и тот же микроконтроллер может иметь большое число разнообразных применений. Многие применения микроконтроллеров в устройствах управления промышленным или бытовым оборудованием предполагают разработку для каждого применения уникальной программы. При этом возникает проблема не только проверки правильности программирования исходного алгоритма, но и его доработки с учетом временных параметров взаимодействия микропроцессора с внешними датчиками и устройствами, учета влияния различных помех на каналы связи и саму аппаратуру и т.д. Поэтому даже если есть уверенность, что аппаратная часть системы исправна, вновь созданные программы необходимо отработать в реальных условиях функционирования. Система считается отлаженной после того, как рабочие программы правильно функционируют на действительной аппаратуре системы в реальных условиях.

Вышеуказанные особенности функционирования и использования микропроцессорных систем привели к созданию специальной **методологии и средств отладки** управляющих микропроцессорных систем на базе микроконтроллеров. Эти средства ориентированы не столько на поиск места неисправности, сколько

¹ Более подробно о тестировании программ см. в теме 8 настоящего учебного пособия.

на отработку разрабатываемых схем на базе микроконтроллеров и «зашиваемых» в них программ.

Так, многие производители универсальных микроконтроллеров, выпускают для своих разработок специальные отладочные комплексы, которые позволяют произвести отработку управляющих программ для их различных применений. Особенностью использования микроконтроллеров является то, что для каждого конкретного применения создается своя конфигурация «обрамляющей» части микроконтроллера (т.е. пользователем самостоятельно разрабатываются и подключаются к микроконтроллеру интерфейсные схемы для взаимодействия с датчиками и выдачи управляющих напряжений); создается необходимая для управления микроконтроллером программа, которая заносится в его перепрограммируемую память. В ходе отладки **решаются задачи** предварительной проверки работоспособности разработанных интерфейсных схем и программы в рабочих режимах функционирования и устранения выявленных дефектов, отрабатывается корректность взаимодействия разработанной программы с операционной системой и другими прикладными программами.

Микропроцессорная система представляет собой объединение некоторых функционально-законченных узлов: процессор, память (ОЗУ, ПЗУ), узел отображения информации, блок питания, АЦП и др. Соответственно выделяют методологии автономной и комплексной отладки.

Автономная отладка аппаратуры предполагает проверку работоспособности отдельных узлов микропроцессорной системы. При этом организуется подача рабочих или тестовых входных воздействий и прием ответных реакций с отлаживаемого узла. Воздействия и ответные реакции определяются, исходя из спецификаций на отлаживаемую систему, а также структурных схем устройств.

После отладки отдельных узлов проверяется их работа во взаимодействии. Это комплексная отладка. **Особенностями средств комплексной отладки**, по сравнению со средствами автономной отладки, является возможность управления поведением микропроцессорной системы в целом и сбора информации о ее поведении в реальном времени. Фактически при этом проверяется правильность совместного функционирования компонент системы при выполнении некоторой тестовой и рабочих программ. Процессор системы работает с шинами адресов, данных и управления. Анализируя их сигналы, можно проконтролировать выполнение программы в процессоре.

При автономной отладке аппаратуры могут потребоваться приборы, умеющие:

а) выполнять функции аналогового прибора, т.е. измерять напряжение и ток; отображать форму сигнала, подавать импульсы определенной формы и т.д.;

б) подавать последовательность сигналов одновременно на несколько входов в соответствии с заданной временной диаграммой или заданным алгоритмом функционирования аппаратуры;

в) собирать и отображать значения сигналов на многих линиях в течение определенного промежутка времени, задаваемого пользователем.

Для автономной отладки аппаратуры широко используются осциллографы, вольтметры, амперметры, частотомеры, генераторы импульсов, позволяющие отлаживать аппаратуру на схемном уровне. В ходе комплексной отладки приме-

няют логические анализаторы, эмуляторы, диагностические платы, т.е. такие же средства, которые применяются для диагностирования аппаратной части компьютеров.

Отладочные эмуляторы микропроцессоров

Основой современных средств отладки управляющих систем на базе микроконтроллеров являются специализированные программно-управляемые устройства (узлы). В технической литературе их зачастую называют эмуляторами.

В качестве базовых компонент в таких устройствах обычно выделяют:

**эмуляционную память,
узел эмуляции микроконтроллера,
отладчик,
подсистему точек останова.**

Эмуляционная память используется для замещения в процессе отладки штатного ПЗУ и/или ОЗУ микроконтроллера, а в некоторых отладочных комплексах позволяет отлаживать программу без использования реальной системы или ее макета. Эмулятор позволяет использовать вместо штатного ПЗУ отлаживаемого объекта эмуляционную память. При необходимости внесения изменений в отлаживаемую программу достаточно загрузить новую или модифицированную программу в память эмулятора, вместо того чтобы заниматься перепрограммированием и заменой штатного ПЗУ при выявлении очередной ошибки. Использование эмуляционной памяти позволяет упростить отслеживание промежуточных данных в процессе выполнения отлаживаемой программы.

Узел эмуляции микроконтроллера в системах отладки часто представляет собой копию того же микроконтроллера, на базе которого строится микропроцессорная система. Наличие такого узла в сочетании с эмуляционной памятью позволяет вести отладку новой прикладной программы вне рабочей микропроцессорной системы, в которой в последующем эта программа будет использоваться.

Для имитации могут использоваться так называемые **симуляторы**. Они отличаются от эмуляторов более простым устройством аппаратной части. Вместо реального микроконтроллера используются ресурсы внешнего компьютера (его называют инструментальным), который моделирует поведение микроконтроллера и определяет необходимые значения сигналов на его выводах. Эти значения сигналов передаются в аппаратную часть симулятора как сигналы, выработанные микроконтроллером. Недостатком симулятора является невозможность отладки систем в реальном времени, преимуществом – низкая стоимость.

Заметим, что при использовании симулятора существенно упрощается имитация памяти микроконтроллера, ввод текста программы и её трансляция в двоичный код. При этом достаточно просто организовать имитацию выполнения программы по шагам и работу на пониженной частоте, остановки ее выполнения в контрольных точках или по состоянию регистров микроконтроллера, отображение на экране и модификацию всех программно-доступных ресурсов микроконтроллера в момент останова.

Под отладчиком и подсистемой точек останова обычно понимают некоторые программные средства (набор программных модулей) со специфическими функциями.

В функции отладчика входит:

- загрузка отлаживаемой программы в память системы;
- управление процессом эмуляции;
- определение количества обращений к различным участкам программы и времени, затраченного на выполнение различных участков программы;
- вывод и отображение содержимого всех регистров и другой памяти.

Отладчик позволяет контролировать ход выполнения отлаживаемой программы и видеть соответствие между ее исходным текстом и состоянием всех ресурсов эмулируемого микроконтроллера. Для визуализации состояния системы современные отладчики используют многооконный графический интерфейс.

В число базовых функций **подсистемы точек останова** входит:

- формирование условий останова выполнения программы (выбор точек останова):
 - останавливать выполнение программы или выполнять иные действия в заданных точках ее выполнения, в том числе в реальном масштабе времени;
 - синхронно с работой процессора фиксировать последовательность выполняемых инструкций и состояний выбранных сигналов.

Отладочные платы

В ряде случаев фирмы-разработчики микроконтроллеров предлагают своим клиентам специальные макетные платы, называемые в зарубежной литературе – **оценочные платы** (Evaluation Boards) или **платы развития**. В последнее время, при выпуске новой модели кристалла микроконтроллера, фирма-производитель обязательно выпускает и соответствующую плату развития. Физически это печатная плата с установленным на ней микроконтроллером, плюс все необходимое стандартное «обрамление», в том числе схемы для связи с внешним компьютером. Как правило, там же имеется свободное поле для монтажа дополнительных схем пользователя. На плате может быть готовая разводка для установки или подключения дополнительных элементов, рекомендуемых фирмой: ПЗУ, ОЗУ, ЖКИ-дисплей, клавиатура, АЦП и др. Зачастую эти доработанные пользователем платы используют в качестве одноплатных контроллеров, встраиваемых в мало серийную продукцию. Программная часть отладочных средств может «зашиваться» непосредственно в ПЗУ, установленную на плату развития, и/или находиться на внешнем компьютере, подключаемом к плате.

Возможности по отладке, предоставляемые с платой развития, ограничены, но наличие законченного набора готовых программно-аппаратных средств, позволяющих без потери времени приступить к монтажу и отладке прикладной системы, во многих случаях является решающим фактором, особенно если учесть, что стоимость такого комплекта меньше, чем стоимость более универсального эмулятора, а сама плата после отладки может использоваться как рабочее изделие.

В реальной системе микроконтроллер обычно занимается считыванием информации с подключенных внешних устройств (датчиков), обработкой этой информации и выдачей управляющих воздействий на исполнительные узлы. Наличие на отладочной плате интерфейсов с внешней средой позволяет пользователю с помощью внешнего компьютера и специальных программных модулей имитировать поступление сигналов от внешних датчиков, принимать и отображать на экране выходные сигналы в удобной для визуального просмотра форме.

Ниже приведены примеры выполнения отладочных плат (рис. 6.25–6.27).

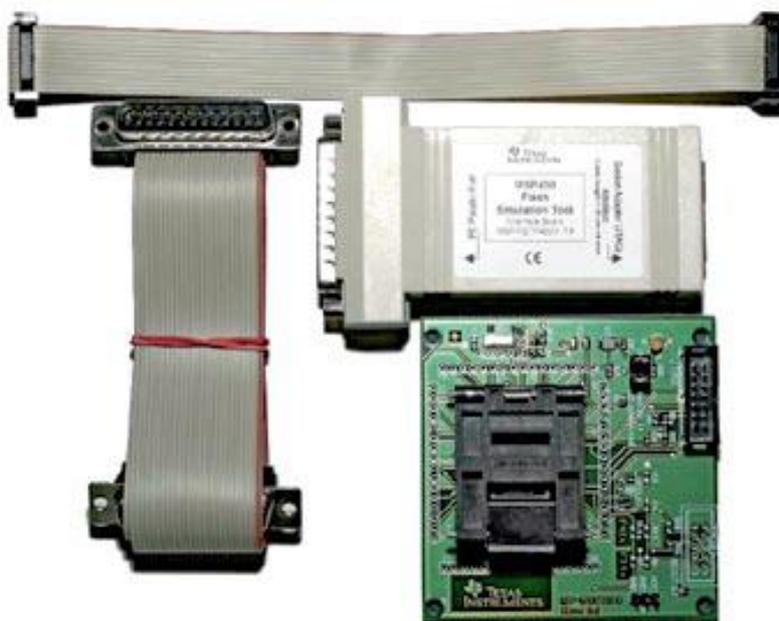


Рис. 6.25. Комплект для отладки и программирования микроконтроллеров MSP430F41X компании Texas Instruments

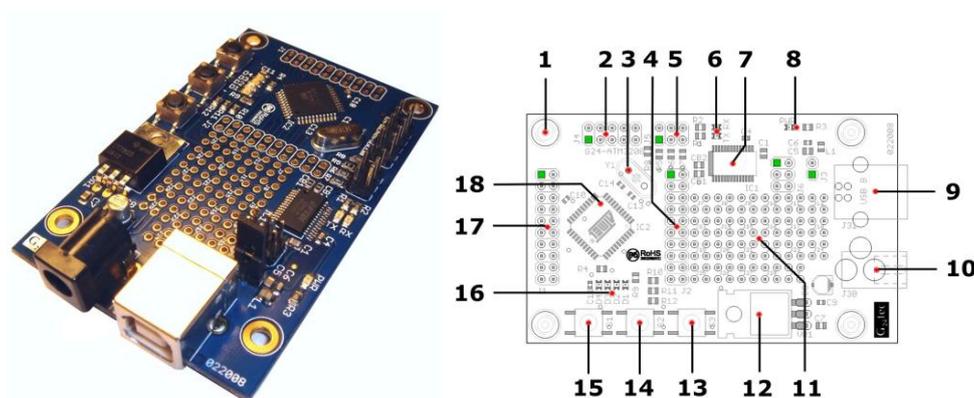


Рис.6.26. Отладочная плата для AVR микропроцессоров фирмы Atmel

В состав комплекта рис. 6.25 входят два микроконтроллера MSP430F413, отладочная плата с ZIF-сокетом, адаптер для подключения отладочной платы к LPT-порту внешнего компьютера, необходимые соединительные кабели, CD-диск с набором документации и программным обеспечением (среда разработки, демо-версия C-компилятора, ассемблер, программный симулятор).

На рисунке 6.26 справа показано расположение компонент на плате: 1 – вывод «земля»; 2 – разъем подключения JTAG-интерфейса; 3 – кварцевый резонатор; 4 – разъем расширения J2; 5 – разъем ISP-интерфейса для подключения к Flash-памяти; 6 – светодиод индикация состояния порта USB, 7 – преобразователь интерфейсов USB – FT232RL; 8 – светодиод «Питание»; 9 – USB порт; 10 – разъем питания; 11 – место для дополнительного монтажа; 12 – регулятор напряжения; 13 – кнопка S3; 14 – кнопка S2; 15 – кнопка сброса; 16 – четыре светодиода общего назначения; 17 – разъем расширения J1; 18 – микроконтроллер ATmega32.

Наличие разъемов подключения JTAG-интерфейса и ISP-интерфейса позволяет организовать режим фоновой отладки и программирование памяти с внешнего компьютера.

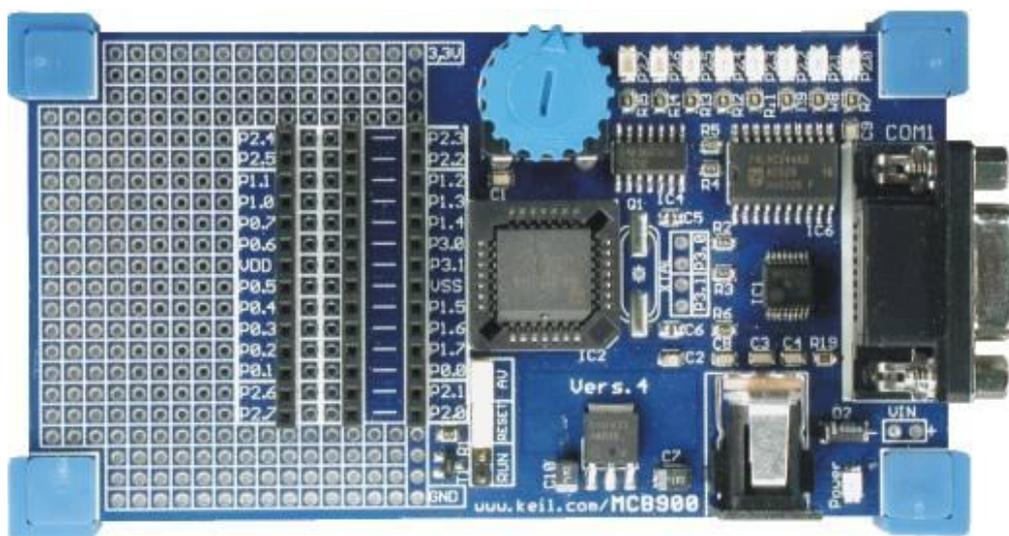


Рис. 6.27. Отладочная плата для отработки устройств на базе микроконтроллера LPC935

Отладочная плата на рис. 6.27 предназначена для отладки схем на базе микроконтроллера LPC935. На ней установлены необходимые для отладки переключатели, светодиоды и другие элементы, имеется площадка для дополнительного монтажа и последовательный COM-порт для подключения к внешнему компьютеру. В комплекте поставляется интегрированная среда разработки с симулятором/отладчиком, а также программным модулем FlashMagic, позволяющим организовать взаимодействие с Flash-памятью в режиме ISP через COM-порт.

В стремлении создать универсальные отладочные средства некоторые фирмы выпускают эмуляторы, которые могут «собираться» под конкретный тип микропроцессора. На рисунке 6.28а показан эмулятор **PICE-51**, который представляет собой некоторую сборную конструкцию, подбираемую под конкретный объект.

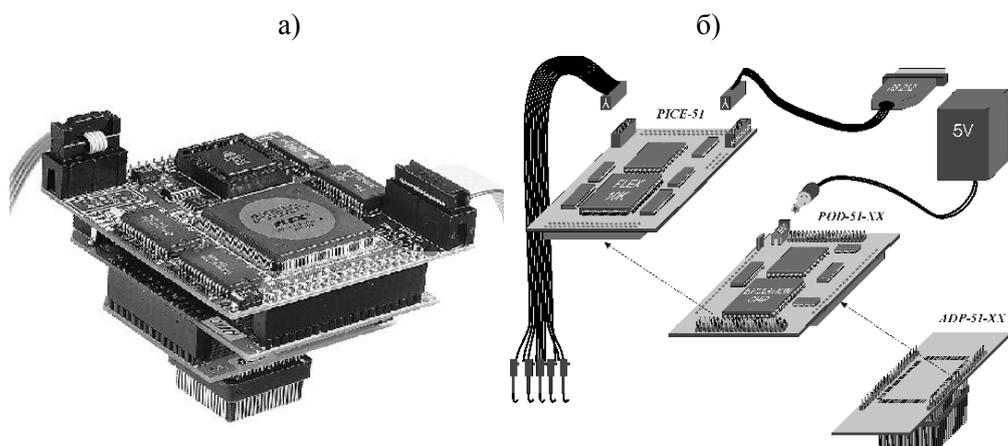


Рис. 6.28. PICE-51 – внутрисхемный эмулятор 8-разрядных микроконтроллеров семейства 8051

Эмулятор состоит из **основной платы PICE-51** размером 80x76 мм, сменного **адаптера POD-51** под конкретный процессор и сменной **эмуляционной головки ADP-51** под конкретный тип корпуса (рис. 6.28б). На основной плате реализуются функции подсистемы точек останова, установлены разъемы для подключения к внешнему компьютеру, для подключения к дополнительным контрольным точкам на объекте контроля. Связь с компьютером – по гальванически развязанному каналу **RS-232C**.

Плата сменного адаптера содержит **эмулирующий процессор** под конкретный тип микроконтроллера.

Эмуляционная головка подбирается под колодку, через которую эмулятор подключается к плате пользователя. Питание эмулятора осуществляется от блока питания или непосредственно от отлаживаемого устройства. Для применения по назначению вышеуказанные узлы эмулятора через специальные разъемы собираются в единую конструкцию (эмулятор в сборе показан на рис. 6.28 а).

Для конкретизации возможностей отладочных эмуляторов ниже приведены характеристики эмулятора PICE-51.

Характеристики аппаратуры:

- отсутствие каких-либо ограничений на использование программой пользователя ресурсов микроконтроллера;
- до 256К эмулируемой памяти программ и данных. Поддержка банкованной модели памяти. Распределение памяти между эмулятором и устройством пользователя с точностью до 1-го байта;

- до 512К аппаратных точек останова по доступу к памяти программ и данных;
- аппаратная поддержка для отладки программ на языках высокого уровня;
- трассировка 8-ми произвольных внешних сигналов;
- 4 выхода синхронизации аппаратуры пользователя;
- трассировщик реального времени с буфером объемом от 16К до 64К фреймов по 64 бита с доступом «на лету». Трассировка адреса, данных, сигналов управления, таймера реального времени и 8-ми внешних сигналов пользователя;
- программируемый фильтр трассировки;
- аппаратный процессор точек останова с возможностью задания сложного условия останова эмуляции по комбинации сигналов адреса, данных, управления, 8-ми внешних сигналов, таймера реального времени, счетчиков событий и таймера задержки;
- 4 комплексных точки останова, которые могут быть использованы независимо или в комбинациях по условиям AND/OR/IF-THEN;
- 48-разрядный таймер реального времени;
- прозрачная эмуляция – доступ «на лету» к эмулируемой памяти, точкам останова, процессору точек останова, буферу трассировки, таймеру реального времени;
- управляемый генератор тактовой частоты для эмулируемого процессора. Возможность плавного изменения тактовой частоты от 500 кГц до 40 МГц;
- гальванически развязанный от компьютера канал связи RS-232C;
- встроенная система самодиагностики аппаратуры эмулятора.

Характеристики программного обеспечения:

- программное обеспечение ориентировано на работу в среде Windows на IBM-совместимых компьютерах;
- встроенный многооконный **редактор** предназначен для написания исходных текстов программ. Редактор поддерживает операции с блоками текста, поиск/замену, цветовое выделение синтаксических конструкций языка ассемблера и Си;
- встроенный **менеджер проектов** обеспечивает автоматическую компиляцию программ. Все опции задаются в диалоговой форме. Переход от редактирования исходного текста к отладке и обратно происходит «прозрачно», т.е. менеджер проектов автоматически запускает компиляцию проекта при необходимости;
- символьная отладка и отладка по исходному тексту для программ, созданных с помощью следующих компиляторов:
 - ассемблер **ASM51** фирмы **Intel**,
 - ассемблер **MCA-51** фирмы **Фитон/МикроКосм**,
 - компилятор **PL/M** фирмы **Intel**,
 - ассемблер и компилятор **Си** фирмы **IAR Systems**,
 - ассемблер и компилятор **Си** фирмы **Avocet Systems Inc./HiTech**,
 - ассемблер и компилятор **Си** фирмы **Keil Software Inc.**;

– автоматическое сохранение и загрузка файлов конфигурации аппаратуры, интерфейса и опций отладки. Обеспечивается совместимость файлов конфигурации с симулятором PDS-51;

– возможность настройки цветов, шрифтов и других параметров для всех окон одновременно и для каждого окна в отдельности.

Совокупность программно-аппаратных и программных средств, поддерживающих все этапы отладки и обеспечивающих простое и быстрое взаимодействие с другими инструментальными средствами, называют **интегрированной средой разработки**.

Фирмы-разработчики наиболее распространенных микроконтроллеров, как правило, обеспечивают потребителей своей продукции соответствующими средствами. Кроме рассмотренных выше функций, особенностями современной интегрированной среды разработки является наличие развитых средств для ввода текста отлаживаемой программы и условий ее выполнения, многочисленных оконных форм отображения хода выполнения программы и состояния микроконтроллера и его памяти. Например, на экран управляющего компьютера можно вывести текст программы и курсором указать место останова. При запуске программы на выполнение будет показываться каждая выполняемая команда, состояния регистров. После останова можно вручную (с клавиатуры) изменить состояние переменных, ячеек памяти и т.п.

Контрольные вопросы

1. Особенности программно-управляемых устройств как объектов диагностирования.
2. Дайте определения понятий «техническое обслуживание» и «система технического обслуживания».
3. Основные компоненты технического обслуживания ПЭВМ.
4. Как ведется тестирование под управлением программы POST, ее основные функции?
5. Средства и способы самодиагностики и самовосстановления ПЭВМ.
6. Виды эксплуатационных документов, используемых при эксплуатации ЭВМ.
7. Требования к помещениям и размещению в них ПЭВМ.
8. Основные показатели систем технического обслуживания и ремонта.
9. Назначение, основные параметры и функции логических анализаторов.
10. Назначение, основные параметры и функции внутрисхемных эмуляторов.
11. Назначение сигнатурного анализатора, организация диагностирования с использованием сигнатурного анализатора.
12. Нарисуйте схему и поясните работу сигнатурного анализатора фирмы Хьюлетт-Паккард.
13. Назначение, конструкции и основные функции диагностических плат.
14. Назначение и основные функции отладочных средств микропроцессоров.

Рекомендуемая литература

Гигиенические требования к персональным электронно-вычислительным машинам и организации работы. Санитарно-эпидемиологические правила и нормативы СанПиН 2.2.2/2.4.1340-03.

ГОСТ 20911-89. Техническая диагностика. Термины и определения. – М.: Изд-во стандартов, 1990.

ГОСТ 27.310-95. Анализ видов, последствий и критичности отказов. Основные положения. – М.: Изд-во стандартов, 1996.

ГОСТ 27002-89. Надежность в технике. Основные понятия. Термины и определения. – М.: Изд-во стандартов, 1990.

Казьмина, С.К. Компактное тестирование / С.К. Казьмина // Автоматика и телемеханика. 1982. № 3. С. 173–189.

Карибский, В.В. Основы технической диагностики / В.В. Карибский, П.П. Пархоменко, Е.С. Согомонян, В.Ф. Халчев. Кн. 1. – М.: Энергия, 1976. – 464 с.

Клюев, В.В. Технические средства диагностирования: справочник / В.В. Клюев, П.П. Пархоменко, В.Е. Абрамчук и др. – М.: Машиностроение, 1989. – 672 с.

Коган, Б.М. Основы эксплуатации ЭВМ / Б.М. Коган, И.Б. Мкртумян. – М.: Энергоатомиздат, 1988. – 432 с.

Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию персональных электронно-вычислительных машин и организационной техники и сопровождению программных средств. Постановление № 28 от 23 июля 1998 г. Министерство труда и социального развития.

Справочное руководство. Персональные компьютеры семейства Compaq Deskpro 2000. Compaq Computer Corporation.

Фергусон, Дж., Обслуживание микропроцессорных систем / Дж. Фергусон, Л. Макари, П. Уилльямс; пер. с англ. – М.: Мир, 1989. – 336 с.

Ярмолик, В.Н. Генерирование и применение псевдослучайных сигналов в системах испытаний и контроля / В.Н. Ярмолик, С.Н. Демиденко. – Минск: Наука и техника, 1986. – 200 с.

www.g24-tec.com

www.kazus.ru

www.sovtest.ru

www.techno.ru

www.tektronix.com/logic_analyzers

Тема 7. ПРОЕКТИРОВАНИЕ КОНТРОЛЕПРИГОДНЫХ УСТРОЙСТВ

7.1. Понятие и критерии контролепригодности

Что понимают под контролепригодностью?

В книге Р.Дж. Беннетса, известного английского специалиста в области проектирования контролепригодных устройств, дается следующее определение контролепригодности: «Схема является контролепригодной, если процедуры генерации множества тестовых наборов, оценки их эффективности и реализации тестового диагностирования могут быть выполнены при условии соблюдения в установленных пределах финансовых затрат, затрат времени и значений показателей, характеризующих приспособленность схемы к обнаружению неисправностей, поиску места неисправностей и реализации диагностирования».

В отечественном стандарте даются более короткие определения.

Контролепригодность – свойство объекта, характеризующее его приспособленность к проведению контроля заданными средствами (по ГОСТ 19919-74).

Приспособленность к диагностированию – свойство изделия, характеризующее его пригодность к проведению контроля заданными методами и средствами технического диагностирования (по ГОСТ 26656-85).

Чем лучше ОД приспособлен для целей диагностирования, т.е. чем выше уровень его контролепригодности, тем проще (дешевле, быстрее) решаются задачи диагностирования, что, в свою очередь, сказывается на надежности и экономических показателях применения объекта диагностирования.

Для контролепригодных объектов характерным является простота построения тестов, высокие значения показателей диагностирования, высокая ремонтнопригодность.

Для обеспечения высокой готовности и ремонтнопригодности аппаратуры широко применяют встроенные средства диагностирования. Они обеспечивают функции самоконтроля, самодиагностики, самовосстановления.

Самоконтроль – способность устройства контролировать правильность выполнения заданных функций.

Использование схем контроля порождает проблему контроля правильности работы самих схем контроля. В последнее время эту проблему пытаются решить путем построения таких схем контроля, которые обладают свойством самоконтроля правильности своего процесса функционирования.

Самодиагностика – способность устройства в случае выявления неправильного функционирования локализовать неисправность входящих в устройство составных частей.

Самовосстановление – способность восстановить заданные функции посредством автоматической коррекции неправильных результатов работы или посредством автоматического подключения новых узлов вместо неисправных.

Реализация подобных функций связана с применением **принципа избыточности**, предполагающего использование той или иной избыточности при разработке аппаратуры (временной, информационной, аппаратурной или алгоритмической).

Временная избыточность предполагает дополнительные затраты времени на выполнение контрольных операций. Например, периодически запускается тестовая задача с известным результатом решения.

Информационная избыточность предполагает представление команд и данных кодами с дополнительными разрядами для контроля и коррекции ошибок.

Алгоритмическая избыточность предполагает выполнение решения задачи по разным алгоритмам (программам) с проверкой получаемых результатов на совпадение.

Аппаратурная избыточность состоит в применении дополнительной аппаратуры для реализации функций контроля и коррекции ошибок. Например, в кристалл микропроцессора часто встраивают элементы для реализации функций периферийного тестирования (более подробно см. в п. 7.4).

Зачастую используемая методология диагностирования базируется на применении нескольких принципов избыточности. Так, реализация циклических кодов для контроля правильности передачи данных обычно предполагает применение специальных электронных схем, т.е. сочетание принципов информационной и аппаратурной избыточности.

В современных ЭВМ общего назначения средства автоматического контроля обеспечивают контроль правильности функционирования до 90–95% оборудования машины.

Специальные дополнительные средства для повышения надежности обработки информации в компьютерах первыми стали применять военные еще в 50-е годы прошлого века. Это для них создавались многомашинные вычислительные комплексы – прообразы современных параллельных вычислительных систем и сетей. В таких комплексах не менее двух копий одной и той же задачи одновременно запускались не менее чем на двух ЭВМ, результаты сравнивались в так называемых контрольных точках. Если результаты не совпадали, фиксировалась неисправность, осуществлялась ее диагностика программными средствами, отключался неисправный компьютер, затем снова запускалось несколько копий задачи. Заметим, что такой подход предполагает использование аппаратурной избыточности (несколько компьютеров для выполнения одной задачи), наличие дополнительных средств взаимодействия между операционными системами нескольких ЭВМ, а также необходимость прерывать процесс решения задачи на время выяснения причин неисправности, поиска и устранения ее источника. Позднее он получил название структурного резервирования процессов и аппаратуры.

В 60–70-е годы стали активно применять временное резервирование (временная избыточность) процессов для обнаружения сбоев даже в однопроцессор-

ных компьютерах. При этом одна и та же часть программы (между контрольными точками) выполнялась на компьютере несколько раз (с последующим сравнением результатов); вероятность того, что компьютер «собьется» несколько раз в одной и той же «точке» программы, принималась бесконечно малой. Ныне для аналогичных целей применяется иной, более эффективный прием, получивший название многовариантного программирования: одна и та же задача или ее фрагмент реализуется по различным алгоритмам и программам. Однако во всех этих случаях резервирование процессов существенно увеличивает время выполнения каждого приложения, что соответствует уменьшению реальной производительности компьютерной системы.

Разумеется, создатели вычислительных систем и других сложных электронных систем всегда заботились о повышении надежности «слабых мест» своих разработок путем использования дополнительных аппаратурных и программных средств. В этих целях широко применялись схемы помехоустойчивого кодирования для взаимодействия с оперативной памятью и передачи информации в системах коммуникаций; самопроверяемые схемы контроля работы сумматоров, дешифраторов и других модулей; в наиболее ответственных случаях нередко использовалось дублирование различных блоков; в последние годы разработчики БИС и СБИС стали встраивать в них элементы для организации периферийного тестирования и др.

Показатели приспособленности к диагностированию

Стандартом ГОСТ 26656-85 (Техническая диагностика. Контролепригодность. Общие требования) рекомендуется контролепригодность оценивать количественными характеристиками, называемыми **показателями приспособленности к диагностированию**.

В качестве конкретных показателей приспособленности к диагностированию рекомендуется рассматривать:

- **среднюю оперативную трудоемкость** данного вида диагностирования (S_d);
- **коэффициент безразборного диагностирования** ($K_{б.д.}$).

Средняя оперативная трудоемкость данного вида диагностирования – средняя суммарная оперативная трудоемкость операций, необходимых для определения технического состояния изделия при данном виде диагностирования. Определяется по формуле

$$S_d = \sum_{j=1}^N S_{dj},$$

где N – число операций данного вида диагностирования, необходимых для определения технического состояния изделия; S_{dj} – оперативная трудоемкость j -й операции диагностирования, чел.-ч; $S_{dj} = S_{oj} + S_{vj}$, где S_{oj} – основная трудоемкость j -й операции диагностирования, чел.-ч; S_{vj} – средняя вспомогательная трудоемкость j -й операции диагностирования, чел.-ч.

В этих формулах **основная трудоемкость диагностирования** характеризует затраты труда на непосредственное диагностирование (установление необходимых режимов работы изделия и средств диагностирования, измерения, сравнение ре-

зультата измерения с эталоном, регистрация и отображение результата); **вспомогательная трудоемкость** характеризует среднюю трудоемкость установки и снятия устройств, необходимых для выполнения операций диагностирования, а также трудоемкость доступа к контрольным точкам и приведения объекта в исходное состояние после диагностирования.

Коэффициент безразборного диагностирования – отношение контролируемых параметров изделия данного вида диагностирования, для измерения которых не требуются демонтно-монтажные работы, к общему числу контролируемых параметров изделия. Определяется по формуле:

$$K_{б.д.} = \Pi_k / \Pi_n,$$

где Π_k – число контролируемых параметров изделия данного вида диагностирования, для измерения которых не требуются демонтно-монтажные работы; Π_n общее число контролируемых параметров данного вида диагностирования.

Кроме того, в зависимости от специфики применения объекта, вместо средней оперативной трудоемкости могут применяться показатели:

– **средняя оперативная продолжительность** данного вида диагностирования (T_d) – средняя суммарная оперативная продолжительность операций, необходимых для определения технического состояния изделия при данном виде диагностирования;

– **удельная суммарная оперативная трудоемкость** данного вида диагностирования ($S_{уд}$) – отношение средней суммарной оперативной трудоемкости диагностирования изделия за цикл периодического технического обслуживания к заданной наработке изделия.

Управляемость и наблюдаемость

На стадии проектирования объекта для определения уровня контролепригодности разрабатываемой схемы (в первую очередь относительно задачи построения теста) часто оценивают ее **управляемость** и **наблюдаемость**.

Понятия управляемости и наблюдаемости пришли из теории управления.

Говорят, что система является **управляемой по состоянию**, если она может быть переведена из любого состояния $z(t_0)$ при $t=t_0$ в любое другое желаемое состояние $z(t_1)$ за конечный интервал времени τ ($\tau=t_1-t_0$) путем приложения входного воздействия $u(t)$. Это понятие может быть использовано для определения **простоты установки элементов памяти** схемы в заданное состояние, что часто требуется при построении тестов последовательностных схем.

Возможна также управляемость по выходу схемы. Пусть $y(t_0)$ – значение (состояние) выхода схемы в момент $t=t_0$. Говорят, что система является управляемой по выходу, если она может быть переведена из любого состояния $y(t_0)$ при $t=t_0$ в любое другое желаемое состояние $y(t_1)$ за конечный интервал времени τ ($\tau=t_1-t_0$) путем приложения входного воздействия $u(t)$.

Понятие наблюдаемости дополняет понятие управляемости. Если управляемость требует (в идеале), чтобы каждое состояние системы было чувствительно к воздействию входного сигнала, то наблюдаемость требует, чтобы каждое состояние системы влияло на измеряемый выходной сигнал.

Система наблюдаема, если ее состояние можно непосредственно или косвенно определить по выходным (измеряемым) сигналам системы.

Поэтому, когда определенное состояние (или изменение этого состояния) не влияет на выходные сигналы, система не наблюдаема. Отсутствие влияния входных сигналов на некоторое состояние означает, что система неуправляема.

При построении тестов возникают задачи обеспечения в заданной точке (линии) необходимого значения сигнала (для проявления неисправности по выходу неисправного элемента) и продвижения неправильного сигнала до контролируемого выхода. Иными словами, возникают задачи управляемости линии (т.е. установки линии в нужное состояние) и наблюдаемости линии (т.е. «транспортирования» сигнала с данной линии до контролируемого выхода схемы). Очевидно, чем «проще» решаются задачи управляемости и наблюдаемости линии, тем легче (проще) будет строить тест для схемы.

В технической диагностике используется несколько способов численной оценки управляемости и наблюдаемости.

Так, один из способов предполагает подсчет числа линий, на которых надо установить конкретные значения $0(1)$. Обозначим линию с неисправностью через z . В зависимости от неисправности на линии надо установить 0 либо 1 . **$1(0)$ – управляемость линии z** определяется минимальным числом линий схемы, которые надо установить в конкретное состояние для установки на линии z конкретного состояния $1(0)$. **Наблюдаемость линии z** есть число линий схемы, которые надо установить в конкретные состояния (0 или 1), чтобы обеспечить «транспортирование» сигнала с линии z до контролируемого выхода схемы.

Пусть W – множество линий схемы, а $k_{y0}(i)$ и $k_{y1}(i)$ – 0 и 1 управляемость линии $i \in W$ соответственно. Тогда общую $0(1)$ управляемость схемы можно оценить по формулам:

$$K_{y0} = \sum k_{y0}(i) \text{ и } K_{y1} = \sum k_{y1}(i) \text{ для всех } i \in W.$$

Аналогично общую наблюдаемость K_n схемы можно оценить по формуле:

$$K_n = \sum k_n(i),$$

где $k_n(i)$ – наблюдаемость линии $i \in W$.

При данном подходе к оценке управляемости и наблюдаемости контролепригодность $K_{кп}$ схемы, в целом, естественно определять как

$$K_{кп} = K_{y0} + K_{y1} + K_n.$$

Подсчитаем коэффициент $K_{кп}$ для схемы на рис. 7.1. Очевидно, $0(1)$ управляемости линий, являющихся входами схемы, равны 1 . Иными словами, $k_{y0}(4) = k_{y0}(5) = k_{y0}(6) = k_{y0}(7) = k_{y1}(4) = k_{y1}(5) = k_{y1}(6) = k_{y1}(7) = 1$. Рассмотрим теперь элемент $D1$ схемы на рис. 7.1. Это элемент, реализующий функцию ИЛИ-НЕ. Для того, чтобы на линии (2) установить 0 , достаточно установить 1 хотя бы на одной из линий (6) или (4), т.е. формула для вычисления 0 -наблюдаемости линии (2) имеет вид $k_{y0}(2) = \min\{k_{y1}(6), k_{y1}(4)\} + 1$. Для обеспечения на линии (2) значения 1 , надо обеспечить на обоих входах элемента $D1$ значения 0 . Иными словами, $k_{y1}(2) = k_{y0}(6) + k_{y0}(4) + 1$. Таким образом, имеем $k_{y0}(2) = 1 + 1 = 2$, а $k_{y1}(2) = 1 + 1 + 1 = 3$. Действуя аналогичным образом, вычислим $0(1)$ – управляемости линии (3). Зная

управляемость линий (2) и (3), нетрудно определить управляемости линии (1). Например, 0- управляемость линии (1) можно определить по формуле $k_{y0}(1) = \min\{k_{y0}(2), k_{y0}(3)\} + 1$ (элемент D2 реализует функцию И, поэтому для установки на линии (1) достаточно обеспечить на одном из входов значение 0).

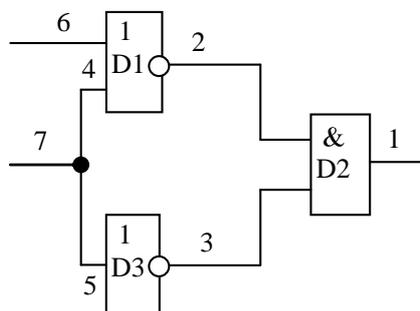


Рис. 7.1. Пример для вычисления показателя контролепригодности

Подобным образом можно определять и наблюдаемость линий. Так, очевидно, наблюдаемость выходной линии (1) равна 0. Наблюдаемость линии (6) можно определить по формуле $k_n(6) = k_n(2) + k_{y0}(4) + 1$. В этой формуле одним из слагаемых является $k_{y0}(4)$, так как для «транспортирования» сигнала с линии (6) на линии (4) надо установить значение 0. Если некоторая линия (L) расщепляется на несколько линий L^1, L^2, \dots, L^m (например, на рис. 7.1 линия 7 расщепляется на линии 4 и 5), то ее наблюдаемость $k_n(L) = \min\{k_n(L^1), k_n(L^2), \dots, k_n(L^m)\}$, т.е. она равна минимальному значению наблюдаемости линий, на которые разделилась исходная линия (L).

Значения управляемостей и наблюдаемости линий схемы рис. 7.1 приведены в табл. 7.1. В крайнем правом столбце этой таблицы приведены также суммарные значения K_{y0}, K_{y1}, K_n .

Таблица 7.1

Значения управляемостей и наблюдаемости для линий схемы рис. 7.1

Показатель	Номера линий							Суммарное значение
	1	2	3	4	5	6	7	
0-управляемость	5	2	2	1	1	1	1	$K_{y0}=13$
1-управляемость	3	3	2	1	1	1	1	$K_{y1}=12$
Наблюдаемость	0	3	3	5	4	5	4	$K_n=24$

Используя полученные для схемы рис. 7.1 данные, легко подсчитать показатель ее контролепригодности: $K_{кп} = K_{y0} + K_{y1} + K_n = 13 + 12 + 24 = 49$.

Известен также способ оценки показателей контролепригодности, при котором все возможные значения находятся в интервале $0 \div 1$. При таком подходе максимальную управляемость, равную 1, имеют входные линии схемы; линия, на которой

невозможно установить значение 0 или 1, имеет управляемость 0. В общем случае управляемость линии L на выходе элемента можно вычислить по формуле:

$$K_y(L) = C(L) \times f\{X\},$$

где $C(L)$ – коэффициент передачи управляемости для линии L, $f\{X\}$ – некоторая функция от управляемостей множества входных линий элемента. Коэффициент передачи определяется выражением:

$$C(L) = 1 - |L(0) - L(1) / L(0) + L(1)|,$$

где $L(0)$ – число всех вариантов установки 0 на линии L, $L(1)$ – число всех вариантов установки 1 на линии L. Из формулы видно, что $C(L) = 1$, когда $L(0) = L(1)$ и $C(L) = 0$, когда $L(0) = 0$ либо $L(1) = 0$. Число вариантов установки выходной линии L можно определять как количество комбинаций входных сигналов элемента, позволяющих установить на линии значение 0(1). Так, для элемента И с двумя входами $L(0) = 3$, $L(1) = 1$ и $C(L) = 1 - |3 - 1 / 3 + 1| = 0,5$; для элемента И с тремя входами $L(0) = 7$, $L(1) = 1$ и $C(L) = 1 - |7 - 1 / 7 + 1| = 0,25$. Что касается функции $f\{X\}$, то ее можно вычислять, например, как среднее арифметическое управляемостей на входных линиях элемента. Наблюдаемость призвана оценить легкость транспортировки неправильного сигнала от неисправности до контролируемых выходов. По рассматриваемому подходу наблюдаемость определяется как некоторая функция от числа путей (активизируемых и не активизируемых) до контролируемых выходов, а также числа комбинаций сигналов, позволяющих активизировать каждый путь. Итоговый показатель контролепригодности оценивается по формуле

$$K_{кп} = K_y \times K_n.$$

Так, если $K_y = 0,7$ и $K_n = 0,5$, то $K_{кп} = 0,7 \times 0,5 = 0,35$.

Показатели надежности как критерии контролепригодности

В качестве частных критериев уровня контролепригодности на стадии эксплуатации можно рассматривать показатели ремонтпригодности.

Вероятность восстановления – вероятность того, что время восстановления работоспособности объекта не превысит заданное значение.

Средняя время восстановления – математическое ожидание времени восстановления работоспособного состояния объекта после отказа.

Средняя трудоемкость восстановления – математическое ожидание трудоемкости восстановления работоспособного состояния объекта после отказа.

На практике уровень контролепригодности часто определяют по такому комплексному показателю надежности, как коэффициент готовности, а также по затратам, связанным с ремонтом. **Коэффициент готовности** – показатель надежности ремонтируемых объектов, характеризующий вероятность того, что объект будет работоспособен в произвольно выбранный момент времени в промежутках между выполнениями планового технического обслуживания:

$$K_g = T / (T + T_b),$$

где T – наработка на отказ; T_b – среднее время восстановления работоспособности.

7.2. Контроль и восстановление информации с использованием кодирования

Понятие кода, ошибки и основные методы контроля

При работе с оперативной памятью ЭВМ, обмене данными между процессором и внешними устройствами ЭВМ, перемещении пакетов данных по локальной сети или сети Интернет и т.д. возможно искажение пересылаемой информации. При этом могут измениться значения некоторых бит, пропасть или добавиться биты (байты). Основные причины искажений – помехи в линиях связи, дефекты и сбои в работе программных и технических средств, возможны и умышленные искажения информации. Поэтому в ЭВМ, вычислительных сетях и системах используются специальные методы контроля правильности передачи информации, проверки отсутствия различий в записанной и считываемой информации.

Для вышеуказанных целей в современных компьютерах, компьютерных сетях и системах широко применяют методы контроля, основанные на использовании специальных способов кодирования двоичных данных. В простейшем случае может осуществляться перекодирование исходных слов данных в более длинные слова, например, 4-битовых на 8-битовые. Тогда некоторые из 8-битовых комбинаций будут запрещенными. При получении запрещенной комбинации можно утверждать, что произошло искажение при передаче. В частности, фирма IBM уже в первых своих ЭВМ использовала так называемый код M из N, в котором исходный код преобразуется в более длинный новый код длины N. В нем любому допустимому исходному коду был сопоставлен код с числом единиц, равным M.

Наиболее широко используются следующие методы:

контроль чётности,

коды коррекции ошибок (Error Correcting Code – ECC),

вычисление контрольной суммы,

циклические избыточные коды (Cyclic Redundancy Check – CRC).

Эти методы используются для того, чтобы обнаружить и, если возможно, исправить ошибки, возникающее при хранении, передаче и обработке данных. В частности, все модемы используют подобные методы, что закреплено международными стандартами. Все вышеуказанные методы используются при передаче данных по сети Интернет. Любой компьютер, записывая файл, добавляет к нему число с контрольной суммой, которое затем используется для контроля чтения.

Известно довольно много вариантов реализаций методов кодирования. В данном разделе рассматриваются общие принципы контроля с использованием кодирования на примере некоторых конкретных реализаций.

Информацию, которая хранится, передается и обрабатывается с помощью компьютеров, всегда можно рассматривать как некоторые двоичные коды. Их длина измеряется в битах или байтах.

Если длина кода n разрядов, то таким двоичным кодом можно представить максимум 2^n различных слов. Так, с помощью кода длиной в один байт (т.е. 8 бит) можно представить $2^8=256$ разных двоичных слов.

Если все разряды слова служат для представления информации, код называется **простым** (неизбыточным). Коды, в которых только часть кодовых слов используется для представления информации, называются **избыточными**. Часть слов в избыточных кодах являются запрещёнными, и их появление при передаче информации свидетельствует о наличии ошибки. Принадлежность к разрешённым или запрещённым словам определяется правилами кодирования, и для различных кодов эти правила различны.

Коды разделяются на **равномерные** и **неравномерные**. В равномерных все слова содержат одинаковое число разрядов. В неравномерных число разрядов в словах может быть различным. В вычислительных устройствах и системах применяются преимущественно равномерные коды.

Равномерные избыточные коды делятся на **разделимые** и **неразделимые**. Разделимые коды всегда содержат постоянное число информационных (представляющих передаваемую информацию) и избыточных разрядов, причём избыточные разряды занимают одни и те же позиции в кодовом слове. В неразделимых разряды кодового слова невозможно четко разделить на информационные и избыточные.

Ошибки приводят к искажению значений разрядов кода. Если искажается g разрядов, то говорят, что имеет место **ошибка кратности g** .

Очень важным параметром, характеризующим способность выбранной системы кодирования обнаруживать и исправлять ошибки, является кодовое расстояние.

Кодовым расстоянием между двумя словами называется число разрядов, в которых символы слов не совпадают. Если длина слова n , то кодовое расстояние может принимать значение от 1 до n . Например, слова 00001111 и 10001110 различаются в двух разрядах – первом и последнем. Следовательно, кодовое расстояние между этими двумя словами $d=2$.

Способность кода (системы кодирования) обнаруживать или исправлять ошибки определяется так называемым **минимальным кодовым расстоянием**.

Минимальным кодовым расстоянием данного кода (d_{\min}) называется минимальное кодовое расстояние между двумя любыми словами в коде. Если имеется хотя бы одна пара слов, отличающихся друг от друга только в одном разряде, то минимальное расстояние такого кода равно 1.

Пусть используется 8-разрядный код, в котором разрешенными являются кодовые слова:

00001111, 11110000, 00110011, 11001100, 00000011.

В нем кодовое расстояние между первой парой слов и рядом других $d=4$. Однако имеются пары слов, например, 00001111 и 00000011, между которыми кодовое расстояние $d=2$. Это значение и будет минимальным кодовым расстоянием для рассматриваемого кода.

Простой (неизбыточный) код имеет минимальное расстояние $d_{\min} = 1$. Для разделимых избыточных кодов $d_{\min} > 1$. Если $d_{\min} \geq 2$, то любые два слова в данном коде отличаются не менее чем в двух разрядах, следовательно, любая одиочная ошибка (т.е. искажение только одного разряда) приведёт к появлению запрещённого слова и будет обнаружена.

В общем случае, чтобы избыточный код позволял обнаруживать ошибки кратностью r , должно выполняться условие

$$d_{\min} \geq r + 1.$$

Действительно, одновременная ошибка в r разрядах слова даёт новое слово, отстоящее от первого на расстояние r . Чтобы слово совпало с каким-либо другим разрешённым словом, минимальное расстояние между двумя разрешёнными словами должно быть хотя бы на единицу больше, чем r . Нетрудно проверить, что в примере выше будет обнаруживаться искажение любого одного разряда, так как используемый код имеет $d_{\min}=2$.

Коды, которые позволяют обнаруживать и одновременно исправлять ошибки в кодах, называют **корректирующими**.

Для исправления r -кратной ошибки необходимо, чтобы новое слово, полученное в результате ошибки, не только не совпадало с каким-либо разрешённым словом, но и имело только одно наиболее близкое (по кодовому расстоянию) к нему разрешённое слово. Исправление предполагает замену слова с ошибкой на наиболее близкое правильное (разрешённое) слово.

Допустим, что искажённое слово отстоит от правильного на расстояние r . Тогда от любого другого разрешённого слова оно должно отстоять не менее чем на $r+1$ (чтобы была возможность исправления ошибки кратности r), а минимальное кодовое расстояние должно быть

$$d_{\min} \geq 2r + 1.$$

Если $d_{\min} \geq 3$, то одиночная ошибка создаёт запрещённое слово, отличающееся от правильного в одном разряде, а от любого другого разрешённого – не менее чем в двух разрядах. Заменяя запрещённое слово наиболее близким к нему (в смысле кодового расстояния) разрешённым словом, можно исправить одиночную ошибку.

В качестве простого примера кода с исправлением ошибок рассмотрим код с четырьмя допустимыми кодовыми словами:

0000000000, 0000011111, 1111100000 и 1111111111.

Его минимальное кодовое расстояние равно 5. Из условия $d_{\min} \geq 2r+1$ вытекает, что при данном коде можно исправить одиночные и двойные ошибки.

Пусть посылается код 0000000000. Допустим, что в результате воздействия помех принято слово 0000000011. Оно не входит в число разрешённых, т.е. в ходе передаче был искажен исходный код. При сравнении полученного слова с каждым разрешённым будут вычислены кодовые расстояния 2, 3, 7, 8 соответственно. Следовательно, полученное слово надо заменить на правильное слово 0000000000, так как оно отличается от 0000000011 на наименьшее число (2) разрядов. При искажении трёх и более разрядов при данных разрешённых кодовых словах гарантировать правильное восстановление невозможно. Так, если слово 0000000000 изменилось на 0000000111, то ближайшим по кодовому расстоянию будет разрешённое слово 0000011111. Однако это неправильное слово.

Контроль четности

В свое время этот метод был введен фирмой IBM в качестве стандарта для контроля компьютерной информации, в соответствии с которым информация хранится фрагментами по девять битов, причём восемь из них (байт) предназначены собственно для данных, а девятый является **битом чётности** и предназначен для контроля. Значение бита чётности задаётся таким, чтобы количество единиц в разрядах данных (восемь разрядов) и разряде чётности было **нечётным**¹.

Идея контроля состоит в следующем. Пусть надо сохранить в памяти байт данных. Перед занесением в память к байту добавляется контрольный разряд с таким значением, чтобы общее число единичных битов в очередном сохраняемом слове (с учетом контрольного разряда) было нечетным. Так, если число единичных значений в информационной части (8 бит) оказалось четным, то контрольный бит слова устанавливается в 1, иначе – в значение 0. После этого все 9 бит (8 бит данных + соответствующий контрольный бит) записываются в ячейку памяти.

При чтении ячейки подсчитывается число единичных битов в считанном слове (8 бит данных + соответствующий контрольный бит) и, если оно нечетное, то слово считается принятым правильно.

Например, надо записать в ячейку памяти десятичное число 15. Двоичный код этого числа 00001111 содержит четное число единиц. Тогда контрольный, девятый разряд (добавим его к информационным разрядам слева), устанавливается в значение 1 (при этом общее число единиц в байте будет нечетным), т.е. будет записываться кодовое слово 100001111. При чтении подсчитывается число единиц в считанном слове. Если оно нечетное, то принимается, что искажения информации при чтении не было.

Существуют очень простые схемные (аппаратные) решения для определения значения контрольного разряда и подсчета на четность числа единиц в кодах, что способствует широкому использованию этого метода контроля. Использование девятого контрольного бита позволяет схемам управления памятью на аппаратном уровне контролировать целостность каждого байта данных. Так, этот метод используется при пересылке между регистрами ЭВМ; модули (банки) оперативной памяти, устанавливаемые на системной плате персонального компьютера, предусматривают использование бита чётности для контроля правильности чтения/записи, хотя возможен режим работы, когда контроль на чётность отключается. При обнаружении ошибки схема контроля чётности на системной плате ПЭВМ формирует так называемое немаскируемое прерывание (Non-maskable Interrupt – NMI), по которому основная работа прекращается и инициируется специальная процедура, записанная в BIOS компьютера.

Минимальное расстояние кода с проверкой чётности $d_{\min}=2$, поэтому он обнаруживает все одиночные ошибки (т.е. ошибки кратности 1) и, кроме того, все случаи нечётного числа ошибок (3, 5 и т.д.). При одновременном возникновении двух или любого другого чётного числа ошибок код с проверкой чётности не обнаруживает ошибок.

¹ Контроль четности может производиться не только при 8, но и при любом числе информационных разрядов.

рый переход к высокому уровню. При приеме последовательности одинаковых бит их число определяется по длительности принимаемого импульса.



Рис. 7.3. Формат данных при передаче через последовательный порт

В конце блока передаваемых данных следует контрольный бит P (или несколько бит), используемых для организации контроля переданной информации. Затем вновь устанавливается высокий потенциал (стоповый сигнал S₂), что свидетельствует о завершении передачи блока данных.

Современные модемы при передаче данных обязательно добавляют контрольные биты для повышения помехоустойчивости передачи. Они могут автоматически менять способы и уровень контроля в зависимости от помех. О частоте ошибок передающий модем узнает из сообщений приемного модема. Обычно, если увеличивается частота ошибок, то передающий модем автоматически переходит на более помехоустойчивые способы кодирования. Они требуют, конечно, дополнительного времени. Поэтому при высоком уровне помех скорость передачи уменьшается. Существуют международные стандарты, регламентирующие порядок работы модемов в части контроля передаваемой информации.

В инструкциях по настройке модемов при описании способа кодирования передаваемых данных можно встретить, например, указание вида 8-N-1 или 7-E-1. Эти цифры и буквы определяют число битов данных, метод контроля четности и число стоповых битов. 8-N-1 означает, например, что в слове восемь битов данных, контроль четности отсутствует и используется один стоповый бит. В свою очередь 7-E-1 означает семь битов данных в слове, проверку на четность и один стоповый бит.

Коды коррекции ошибок

В ЭВМ, например, при работе с оперативной памятью, при организации обмена с запоминающими устройствами на магнитных лентах и дисках, организации обмена данными в вычислительных сетях применяют корректирующие коды, позволяющие не только обнаруживать, но и исправлять ошибки. К таким кодам относятся код Хэмминга, групповые корректирующие коды, циклические коды и др.

Коды коррекции ошибок (Error Correcting Code – ECC) позволяют не только обнаружить ошибку, но и исправить ее, если искажился один разряд. Компьютер, в

котором используется подобный код, в случае ошибки в одном разряде сможет ее исправить и работать без прерывания. Конечно, возможны ошибки в большем числе разрядов. Но приблизительно 98% сбоев памяти вызвано ошибкой именно в одном разряде, т.е. они исправляются с помощью данного типа кодов.

Рассмотрим применение корректирующих кодов на примере широко распространенного кода Хэмминга.

Предварительно информационные разряды передаваемого слова делятся на группы, которые частично пересекаются. Затем между информационными разрядами слова (по определенным правилам) вставляются несколько контрольных разрядов, значения которых формируются перед передачей информации на основе подсчёта суммы единиц для выбранных групп информационных разрядов.

Контрольная аппаратура на приёмном конце образует корректирующее число на основе принятых информационных и контрольных разрядов, которое равно «0» при отсутствии ошибки либо указывает место ошибки, например, двоичный порядковый номер ошибочного разряда в слове. Ошибочный разряд автоматически корректируется путём изменения его состояния на противоположное.

Требуемое число контрольных разрядов в коде Хэмминга (разрядность корректирующего числа) определяется из следующих соображений. Пусть кодовое слово длиной n разрядов имеет m информационных и $k=n-m$ контрольных разрядов, а корректирующее число должно указать номер неправильного разряда. Если его длина равна k разрядов, то можно указать 2^k разных номеров.

Таким образом, должно соблюдаться соотношение

$$2^k \geq n + 1 \text{ или } 2^k - k - 1 \geq m.$$

Из этого неравенства следует, например, что пять контрольных разрядов позволяют передавать в коде Хэмминга от 1 до 26 информационных разрядов, и т.д. (табл. 7.2). Таким образом, избыточность кода Хэмминга значительно выше избыточности кода с проверкой чётности.

Таблица 7.2

Число контрольных разрядов для кода, способного исправлять одиночные ошибки

Размер слова	Количество контрольных разрядов	Общий размер	Процент увеличения длины слова
8	4	12	50
16	5	21	31
32	6	38	19
64	7	71	11
128	8	136	6
256	9	265	4
512	10	522	2

Контроль по коду Хэмминга реализуется с помощью набора схем подсчёта чётности, которые при кодировании формируют контрольные разряды, а при декодировании – корректирующее число.

В коде Хэмминга к слову, состоящему из m битов, добавляется r битов чётности, при этом образуется слово длиной $m+r$ битов. Биты нумеруются с единицы (а не с нуля) слева направо. Все биты, номера которых – степени двойки, являются битами чётности; остальные – пересылаемые данные.

Допустим, исходная информация, которую надо передать, представлена 16-ю битами. Для представления ее кодом Хемминга нужно добавить 5 контрольных бит. В результирующем коде биты с номерами 1, 2, 4, 8 и 16 – будут битами чётности (контрольными битами), а все остальные – биты данных. Всего передаваемое слово будет содержать 21 бит (16 битов данных и 5 битов чётности).

Каждый бит чётности предназначен для контроля определённой группы разрядов передаваемого кода. Общее число битов со значением 1 в контролируемой группе (с учетом контрольного бита) должно быть чётным. Ниже для каждого контрольного бита приведены контролируемые разряды (указаны номера соответствующих разрядов в пересылаемом кодовом слове):

- 1 – проверяет биты 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21;
- 2 – проверяет биты 2, 3, 6, 7, 10, 11, 14, 15, 18, 19;
- 4 – проверяет биты 4, 5, 6, 7, 12, 13, 14, 15, 20, 21;
- 8 – проверяет биты 8, 9, 10, 11, 12, 13, 14, 15;
- 16 – проверяет биты 16, 17, 18, 19, 20, 21.

Присвоим группе разрядов, которая контролируется битом с номером p , вес равный p . Особенность кода Хэмминга в том, что сумма весов групп, в которые попадает информационный бит, равна номеру его места в передаваемом кодовом слове. Иными словами, если информационный бит находится на месте разряда с номером b , то $b = n_1 + n_2 + \dots + n_k$, где n_i ($i=1, \dots, k$) – веса групп, содержащих разряд b . Так, правильность передачи бита 5 контролируется битами 1 и 4, поскольку $1+4=5$. Бит 6 контролируется битами 2 и 4, так как $2+4=6$ и т. д.

Рассмотрим построение кода Хэмминга для 16-битного слова 1111000010101110. После добавления контрольных бит получим 21-битное слово 001011100000101101110 (рис. 7.4). Прямоугольниками на рисунке выделены контрольные разряды, которые (как легко убедиться) находятся на местах, соответствующих числам $2^0, 2^1, \dots, 2^4$. Чтобы увидеть, как происходит исправление ошибки, рассмотрим, что произойдёт, если бит 5 в этом слове изменит значение из-за помех в линии связи. Тогда вместо правильного слова 001011100000101101110 в месте приема данных будет получено слово 001001100000101101110.

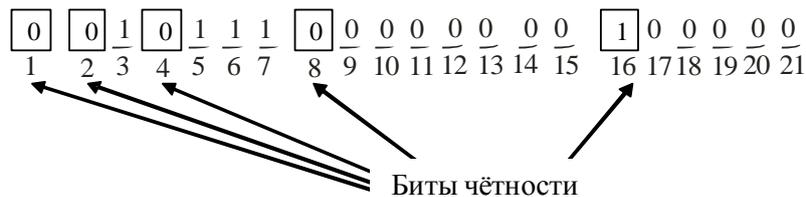


Рис. 7.4. Построение кода Хэмминга для слова 1111000010101110

Проверим на четность соответствующие группы бит. Получим:

бит 1 неправильный (биты 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21 содержат пять единиц),

бит 2 правильный (биты 2, 3, 6, 7, 10, 11, 14, 15, 18, 19 содержат шесть единиц),

бит 4 неправильный (биты 4, 5, 6, 7, 12, 13, 14, 15, 20, 21 содержат пять единиц),

бит 8 правильный (биты 8, 9, 10, 11, 12, 13, 14, 15 содержат две единицы),

бит 16 правильный (биты 16, 17, 18, 19, 20, 21 содержат четыре единицы).

Очевидно, что неправильный бит входит в первый и третий списки. Общими в этих списках являются биты 5, 7, 13, 15 и 21, т.е. неправильный бит среди них. Из полученного списка следует исключить биты, попавшие в списки, сопоставленные контрольным битам с номерами 2, 8 и 16, так как в этих списках нет неправильного бита. После анализа списков из подозреваемых на ошибку битов можно исключить биты с номерами 7, 13, 15, 21, так как все они попали в списки с правильными контрольными битами. В итоге получаем, что был искажен бит номер 5. Поскольку в полученном коде этот бит имеет значение 1, то его правильное значение 0. Именно таким образом исправляются ошибки при использовании кода Хэмминга.

В реальности процедура определения искаженного бита гораздо проще.

Для каждой группы подсчитывается число единиц, и определяются контрольные биты, в списках которых нечетное число единиц. Если таких нет, то искажения кода не произошло. Если обнаружилось группы с нечетным числом единиц, то нужно сложить номера соответствующих контрольных бит. Сумма, полученная в результате, даст номер позиции неправильного бита. Например, в рассмотренном примере неправильными были контрольные биты с номерами 1 и 4. Тогда ошибка произошла в бите с номером $N=1+4=5$.

Вычисление контрольной суммы

При записи/чтении файлов в ЭВМ, пересылке данных по сети Интернет используются методы контроля с вычислением контрольной суммы.

Идея этих методов состоит в следующем. Передаваемые байты группируются в слова, например по 32 бит (4 байта). Каждое слово рассматривается как некоторое целое число, т.е. передаваемая информация представляется последовательностью чисел. Эти числа суммируются (при этом игнорируется переполнение). Результат суммирования инвертируют и заносят (обычно) в заголовок сохраняемого или передаваемого блока информации. Это итоговое число и называют **контрольной суммой**. При проверке полученных данных получатель вычисляет контрольную сумму для полученных данных, включая в число суммируемых чисел и контрольную сумму отправителя. Если искажений не было, то в результате вычислений получатель должен получить слово, в котором все разряды равны единице.

Например, пусть надо передать 4 байта: 00000011, 10000000, 01010101, 10101010. Рассматривая каждый байт как отдельное число, проведем арифметическое суммирование: $00000011+10000000+01010101+10101010=10000010$. Следовательно, контрольное число будет равно инверсии числа 10000010, т.е.

01111101. Таким образом, к отправляемому пакету из 4 байт надо добавить контрольное число 01111101. Получатель пакета производит суммирование пришедших в его адрес байт, рассматривая каждый байт как некоторое число. Если искажений не было, то сумма 4-х исходных байт будет 10000010. Если к нему добавить контрольное число получим $10000010+01111101=11111111$. При искажении байта (включая контрольное число) очень высока вероятность, что результат суммирования на приемной стороне будет не равен 11111111.

Контрольная сумма используется, в частности, при передаче данных по интернетовским протоколам TCP, UDP, XTP и др. на транспортном уровне. Как правило, этот метод контроля реализуется программно. Принятый в Интернет метод вычисления контрольной суммы описан в документе REC 1071.

Контроль при помощи циклического избыточного кода

Эти методы (Cyclic Redundancy Check-CRC) контроля используются, в частности, на канальном уровне при контроле пересылаемых данных в компьютерных сетях. Они достаточно просто реализуются в аппаратном варианте. При этом данные рассматриваются как некоторые многочлены (полиномы). Например, двоичный код 110011001 можно представить в виде многочлена $x^8+x^7+x^4+x^3+1$.

Пусть надо отправить блок данных длиной m разрядов. При этом надо гарантировать обнаружение ошибок кратности не более r . Если отправитель и получатель используют метод контроля на основе циклического избыточного кода, то они предварительно должны оговорить, какой будет применяться **образующий многочлен G**. Он имеет длину $r+1$ разрядов, причем старший (самый левый) бит образующего многочлена должен быть равен 1.

Обмен информацией с использованием контроля на основе циклического кода осуществляется следующим образом.

Отправитель направляет получателю данные, добавив к ним справа k контрольных разрядов, т.е. посылается двоичный код D из $m+k$ разрядов. Левые m разрядов представляют посылаемую информацию. Значения контрольных k разрядов устанавливаются такими, чтобы D делилось на многочлен G без остатка. **Получатель**, приняв код D , делит его на многочлен G . Если результат деления не содержит остатка (т.е. равен нулю), то считается, что при передаче данных не было искажений.

Доказано, что такой способ контроля при соответствующем подборе G позволяет гарантированно обнаружить ошибки кратности r и любые нечетные ошибки. В общем случае для ошибки кратности больше r вероятность ее обнаружения $p=1-1/2^r$. Так, при $r=16$ вероятность $p=(1-2^{-16})=0.9999848$.

Рассмотрим на конкретном примере, как осуществляется формирование отправляемого кода D и его контроль получателем с использованием данного метода.

Пусть надо отправить 16-разрядное слово $D_0=1111000010101110$. В качестве образующего многочлена используем двоичный код $G=110101$, т.е. число контрольных разрядов равно 5. Тогда отправляемое слово будет иметь длину $16+5=21$ разряд.

Чтобы определить значения контрольных разрядов, надо сделать следующее. Добавим к исходному коду справа 5 нулевых разрядов (5 – это число контрольных разрядов). Получим код $D_1 = 11110000101011100000$, который надо разделить на G :

$$\begin{array}{r}
 11110000101011100000 \quad 110101 \\
 110101 \quad 10111100 \quad 0010000 \\
 \hline
 10010 \\
 00000 \\
 \hline
 100100 \\
 110101 \\
 \hline
 100011 \\
 110101 \\
 \hline
 101100 \\
 110101 \\
 \hline
 110011 \\
 110101 \\
 \hline
 1100 \\
 0000 \\
 \hline
 11001 \\
 00000 \\
 \hline
 110011 \\
 110101 \\
 \hline
 1101 \\
 0000 \\
 \hline
 11010 \\
 00000 \\
 \hline
 110100 \\
 110101 \\
 \hline
 10 \\
 00 \\
 \hline
 100 \\
 000 \\
 \hline
 1000 \\
 0000 \\
 \hline
 10000 \\
 00000 \\
 \hline
 10000 \\
 \hline
 \boxed{10000} \longrightarrow \text{Остаток}
 \end{array}$$

Следует иметь в виду, что операции по вычислению CRC кодов выполняются в арифметике по модулю 2 без переносов в соседние разряды, причем операции сложения и вычитания эквивалентны и выполняются путем поразрядного сложения по модулю 2. Таким образом, на каждом шаге вместо арифметической операции вычитания производится сложение по модулю 2.

В результате деления мы получили целую часть, равную 1011110010010000, и остаток 10000. Полученный остаток представляет собой значение, которое надо занести в контрольные разряды, т.е. мы добавляем младшим разрядам кода D_1 число 10000, в результате чего получаем код $D=11110000101011110000$. Этот код, собственно, и направляется получателю. Очевидно, что D , в отличие от D_1 , делится на G без остатка.

Получатель кода D делит его на многочлен G . Если деление происходит без остатка, то считается, что при передаче не было искажений.

Вообще говоря, вероятность обнаружения ошибки зависит от образующего многочлена. Выбор порождающего полинома G зависит от характера возможных ошибок. Международные стандарты устанавливают значения наиболее оптимальных многочленов, которые должны использовать разработчики вычислительных систем и сетей. Так, для контроля правильности передачи 5-байтных заголовков АТМ-ячеек используется 8-разрядный код CRC. В ряде IEEE-протоколов канального уровня используется 32-разрядный образующий многочлен $G=100000100110000010001110110110111$.

Циклические коды, в общем случае, обладают следующими свойствами:

- ошибки в одном бите обнаруживаются, если G содержит более одного члена;
- ошибки в двух битах обнаруживаются, если G содержит три члена;
- ошибки в любом нечетном числе бит обнаруживаются, если G содержит множитель $x+1$;
- при соответствующем выборе G можно обеспечить обнаружение любых ошибок кратности r и ниже;
- если G_r гарантирует обнаружение ошибок кратности r , то с вероятностью $1-(1/2)^{r-1}$ можно обеспечить обнаружение ошибок кратности $r+1$ и с вероятностью $1-(1/2)^r$ любых ошибок кратностью более $r+1$.

Как видно, циклические коды обладают довольно неплохими характеристиками с точки зрения обнаружения ошибок при передаче кодов.

Данный метод контроля может быть реализован в виде достаточно простых аппаратных схем на основе сдвигающего регистра. Поэтому он широко используется при создании отказоустойчивых компьютерных систем обработки информации, на канальном уровне в вычислительных сетях и в качестве средства самоконтроля записи/чтения данных в память ЭВМ.

Ниже (рис. 7.5) приведена схема реализации контроля с помощью циклического кода с использованием образующего многочлена X^4+X+1 .

Особенность данной схемы в том, что она может использоваться как для формирования циклического кода, так и для его проверки. Сдвиговый регистр состоит из 4-х триггеров (Т). Сигнал «Сдвиг» в схеме играет роль тактового сигнала, по фронту которого на вход регистра поступает очередной бит кода D и осуществляется сдвиг находящегося в регистре кода.

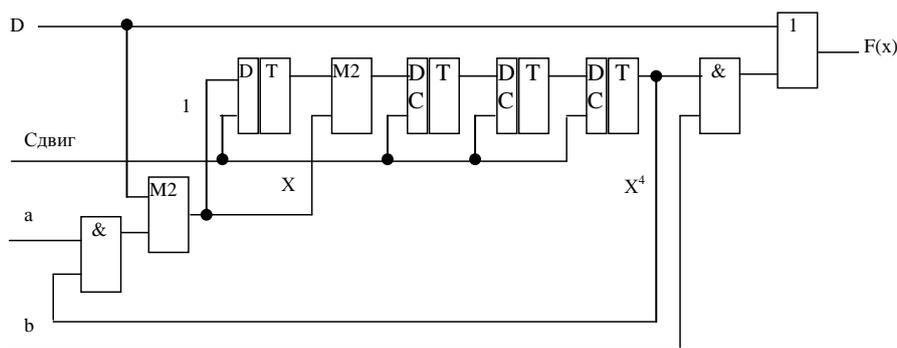


Рис. 7.5. Пример схемной реализации контроля на основе циклического кода

Пусть надо передать код D , имеющий длину m бит. Тогда в течение m тактов $a=1$ и $b=0$. При этом биты кода D поступают на вход регистра и одновременно на выход $F(x)$. В результате в регистре окажется остаток от деления D на многочлен X^4+X+1 (т.е. на код 10011), а на выходе $F(x)$ последовательно появятся все биты кода D . Затем делаем $a=0$ и $b=1$, D равным «0». Тогда при поступлении сигнала «Сдвиг» на выход $F(x)$ последовательно станут поступать биты регистра, т.е. к $F(x)$ добавятся биты остатка, накопленного в регистре. Таким образом, после $m+r$ тактов на $F(x)$ будет циклический код исходной последовательности D (в данном примере $r=4$).

Допустим теперь, что надо принять код $F(x)$ и проверить: не исказился ли он в процессе передачи? Код $F(x)$ подается на вход сдвигового регистра (вместо D), при этом устанавливаем $a=1$ и $b=0$. Если после $m+r$ сдвигов в регистре будут все нули, то ошибок при передаче не было.

В заключение отметим, что вышерассмотренные методы кодирования в той или иной мере используются во всех микропроцессорных системах управления (от схем управления бытовыми приборами до сложнейших систем управления в авиационной и космической технике). Так, широко используемый в современных системах автоматики протокол Modbus обмена с внешними датчиками (приборами) предполагает использование методов контроля четности и CRC.

7.3. Схемные решения повышения контролепригодности

Общие принципы

Во многих случаях хорошую контролепригодность объекта диагностирования невозможно обеспечить без преобразования структуры (схемы) объекта к виду, удобному для проведения диагностирования. Для этого в объект еще на этапе его проектирования и конструирования вводят дополнительные элементы:

- дополнительные контрольные точки (используются для подключения в режимах диагностирования внешней аппаратуры для подачи дополнительных тестовых сигналов либо измерения сигналов);

- дополнительные входы для блокировки сигналов или управления извне;
- аппаратные средства, которые при диагностировании позволяют изменять структуру объекта;
- схемы для функционального встроенного самоконтроля и индикации, которые служат для автоматического обнаружения отказов и указания места (вида) отказа;
- аппаратные средства, которые генерируют тесты и анализируют результаты;
- схемы кодирования, обнаружения и исправления искажений в передаваемых кодах;
- устройства для индикации отказов.

Практически все современные электронные приборы и системы имеют встроенные аппаратные средства для решения задач диагностирования в процессе их работы или технического обслуживания. Большинство схемных решений в той или иной степени имеют целью обеспечить самоконтроль на рабочих воздействиях непосредственно в ходе применения объекта по назначению.

Примером такого прибора является ПЭВМ (см. тему 6). Многие сложные микросхемы также содержат узлы специально для упрощения контроля их работоспособности.

К сожалению, не существует универсальных схемных решений, используемых для упрощения задач самоконтроля и вообще диагностирования. В каждом конкретном случае применяются свои решения, хотя для ряда часто используемых разработчиками функциональных узлов разработаны типовые схемы их контроля.

Всегда важно выбрать наиболее информативные точки для целей диагностирования, обеспечивающие повышение наблюдаемости и управляемости. Доступ к этим точкам позволяет повысить наблюдаемость и управляемость. После этого, зачастую используя некоторые стандартные решения, схема изделия дорабатывается для получения желаемого уровня диагностирования.

К таким точкам относятся:

- выходы генераторов, сигналы синхронизации и сброса;
- шины различного назначения (адреса, данных, управления третьим состоянием и др.);
- цепи обратной связи;
- сигналы задания режимов (блокировка, чтение, запись и т.д.);
- состояния регистров и счетчиков;
- сигналы управления мультиплексорами и др.

Если методика точного определения места неисправности и ремонта предполагает возможность подачи сигналов или измерений с помощью внешних средств (не встроенных в объект), то кроме чисто схемных решений надо предусмотреть соответствующие конструкторские решения, позволяющие достаточно просто осуществлять доступ к необходимым для диагностирования точкам (создать специальные контактные площадки, установить дополнительные разъемы и др.).

Рассмотрим несколько типичных методов и схем для организации контроля.

Дублирование и мажоритарные схемы контроля

Контроль путем дублирования (рис. 7.6) является наиболее универсальным и основан на побитном сравнении выходных сигналов основной и дублирующей схем с последующим объединением выходных сигналов на схемах сравнения в один общий сигнал ошибки. Правда, для его реализации требуются большие аппаратные затраты.

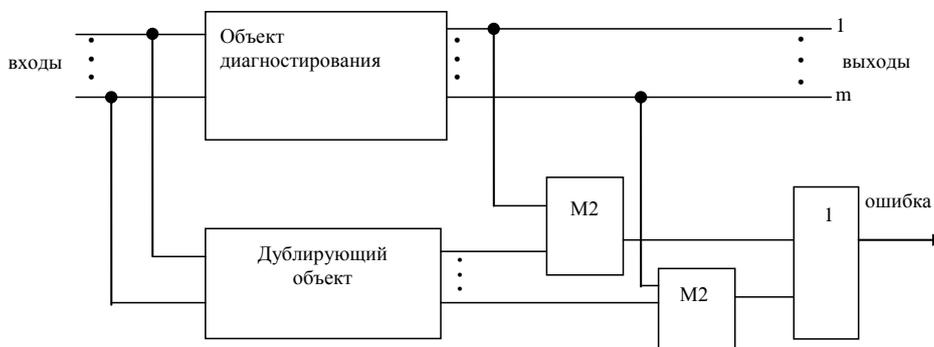


Рис. 7.6. Контроль с использованием дублирования

При этом способе контроля кроме собственно объекта диагностирования надо иметь дублирующий объект, входные и выходные сигналы которых в исправном состоянии должны быть одинаковыми (дублирующий объект может быть упрощенной моделью основного, возможна реализация его в виде программной модели). Выходы основного и дублирующего объектов попарно подключаются к элементам, реализующим функцию «по модулю 2», выходы которых, в свою очередь, подключены к входам элемента ИЛИ. При нормальной работе, когда сигналы пар выходов совпадают, сигналы на выходах всех элементов «по модулю 2» (M2) и выходе элемента ИЛИ равны «0». При возникновении неисправности в основном или дублирующем объекте, хотя бы на одном выходе сигнал станет отличаться от правильного и на выходе соответствующего элемента «по модулю 2» сигнал станет равным «1». Соответственно и на выходе элемента ИЛИ появится сигнал «1», что свидетельствует о возникновении отказа в работе основного или дублирующего объекта. Рассмотренный метод контроля хорош, когда дублирующий объект можно сделать гораздо проще и надежнее основного.

Мажоритарными называются системы, в которых несколько одинаково работающих устройств, а для выделения «правильных» результатов работы используется специальный узел, определяющий правильные данные по большинству одинаковых сигналов.

Мажоритарные методы контроля можно рассматривать как развитие методов дублирования. В мажоритарных системах не два, а несколько одинаковых объектов. Их одноименные выходы объединяются на мажоритарном элементе и направляются по назначению. Каждый мажоритарный элемент имеет столько входов, сколько одинаковых объектов. Выходной сигнал элемента определяется по большинству одинаковых входных сигналов.

На рисунке 7.7 приведен мажоритарный элемент для троированных мажоритарных систем. Он имеет три входа А, В и С (т.е. в системе три одинаково работающих объекта). При нормальной работе объектов А, В, С с них поступают одинаковые сигналы 0 или 1. Выходной сигнал мажоритарного элемента равен «1», если на двух или трех входах «1», иначе – значение «0». При отказе одного из объектов, например А, выходной сигнал, будет определяться значениями сигналов на входах В и С.

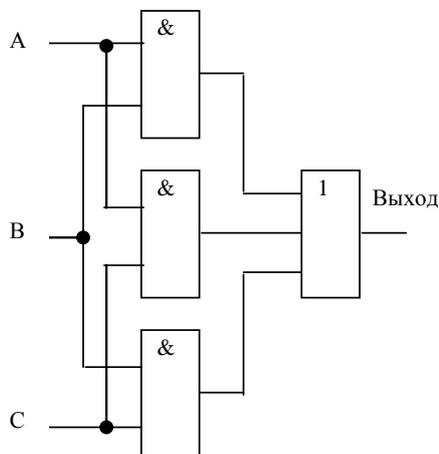


Рис. 7.7. Мажоритарный элемент на три входа

Таким образом, при отказе одного из объектов на выходах мажоритарных элементов все равно будут правильные значения. Часто мажоритарные элементы добавляются специальными схемами, отключающими неисправный объект и информирующими оператора о необходимости ремонта. При допустимости отказов n объектов система должна содержать не менее $2n+1$ одинаковых объектов. Чтобы повысить эффективность мажоритарных систем, они содержат механизм адаптации, который отключает неисправные объекты и переводит мажоритарные элементы на «голосование» по меньшему числу входов. Например, пусть отказали 2 из 5 объектов. Тогда надо отключить два отказавших объекта, организовать на базе исправных троированную мажоритарную систему и тем самым продолжить контроль по мажоритарному методу. Мажоритарные методы затратные и используются в очень ответственных системах: в космических аппаратах, на пассажирских самолетах и т.п. Системы, в которых реализуются мажоритарные методы контроля, относятся к отказоустойчивым.

Самопроверяемые схемы контроля

Сами схемы контроля тоже могут отказывать. Поэтому стараются применять так называемые самопроверяемые схемы контроля.

Схема контроля называется **самопроверяемой**, если она обнаруживает не только неисправности контролируемого объекта, но также и свои собственные. Так, самопроверяемые схемы контроля широко применяются в вычислительных системах и устройствах. Для многих типов узлов вычислительных устройств

(регистры, сумматоры, арифметико-логические устройства) разработаны специальные самопроверяемые схемы контроля.

В случае цифровых объектов диагностирования широко распространен и наиболее формализован способ построения самопроверяемых схем контроля, имеющих два выхода f_1 и f_2 (рис. 7.8). При этом предполагается, что на входы схемы контроля (СК) поступают те же сигналы, что и на объект, а также выходные сигналы объекта диагностирования. Значения сигналов 01, 10 на выходах f_1 , f_2 обычно свидетельствуют о правильном функционировании контролируемого устройства и СК, а значения 00, 11 – о наличии одиночной неисправности в контролируемом объекте или СК.

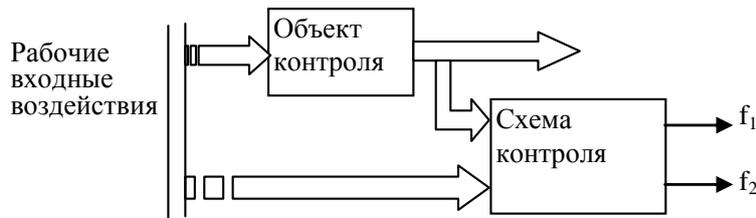


Рис. 7.8. Подключение самопроверяемой схемы контроля

Одним из требований при построении самопроверяемой схемы контроля (ССК) является требование раздельной реализации функций f_1 и f_2 . Тогда можно гарантировать, что любая одиночная константная неисправность на входах и выходах ССК не приведет к инвертированию сигналов на обоих выходах схемы.

В качестве примера ССК рассмотрим схему контроля по нечетности байтового регистра (восемь информационных x_1, \dots, x_8 и один контрольный бит x_k), изображенную на рис. 7.9. Эта ССК состоит из двух отдельных схем контроля. Первая реализует функцию $f_1 = x_1 \oplus x_2 \oplus x_3 \oplus x_4$, вторая – $f_2 = x_5 \oplus x_6 \oplus x_7 \oplus x_8 \oplus x_k$, где x_k – контрольный бит.

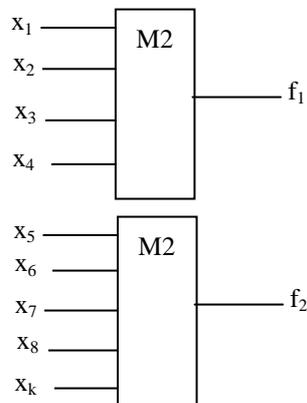


Рис. 7.9. Самопроверяемая схема контроля по нечетности

При передаче байта с контролем по нечетности любой неискаженный поступивший на ССК байт с контрольным битом имеет нечетное число единиц, причем одна из групп (f_1) будет содержать их четное число, а другая (f_2) – нечетное (либо наоборот). Таким образом, при правильной (без искажений) передаче информации выходной сигнал ССК всегда будет принимать значение 01 или 10. При искажении в процессе пересылки любого бита общее число единиц станет четным, и обе функции f_1 и f_2 будут иметь один и тот же выходной сигнал "0" или "1". Очевидно, тот же результат будет при возникновении одиночной неисправности самой ССК.

Другим примером самопроверяемой схемы контроля является схема контроля дешифратора, приведенная на рис. 7.10. Особенность работы дешифратора в том, что при нормальной работе только на одном выходе должен быть сигнал "1". Нетрудно заметить, что при нормальной работе (без неисправности) выходные сигналы ССК принимают значения 10 или 01. При возникновении одиночной неисправности в дешифраторе или в самой схеме контроля выходные сигналы ССК принимают значения 00 или 11.

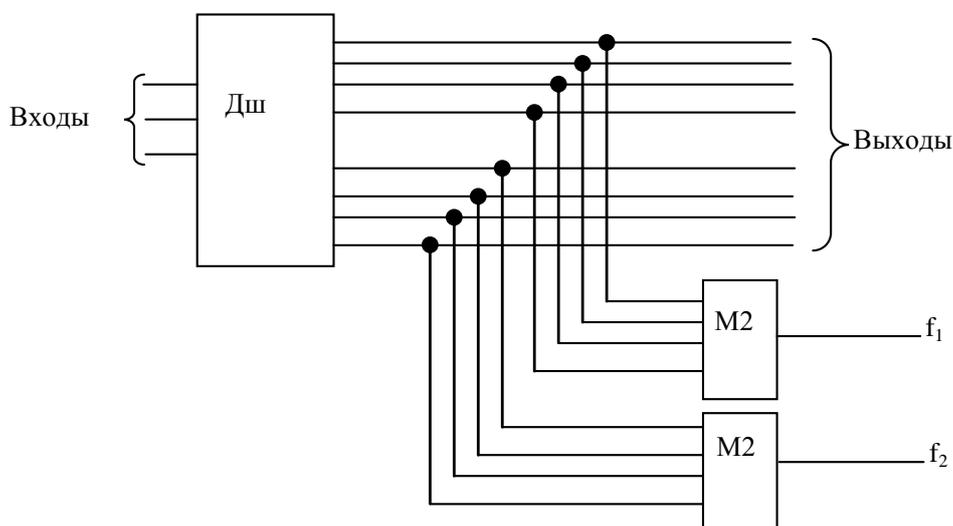


Рис. 7.10. Самопроверяемая схема контроля дешифратора

Основой арифметико-логического устройства, осуществляющего арифметические операции в ЭВМ, является сумматор. Он поразрядно складывает пары двоичных чисел. Таблица истинности, описывающая работу одноразрядного сумматора, приведена в табл. 7.3. В таблице использованы обозначения: a_i и b_i – i -й разряд складываемых чисел A и B , c_{i-1} – сигнал переноса от сложения предыдущего разряда, s_i – результат логического сложения a_i и b_i , c_i – сигнал переноса при сложении разрядов a_i и b_i . Для контроля такого сумматора (SM) можно использовать схему, приведенную на рис. 7.11 и обладающую свойством самопроверяемости.

Таблица истинности одноразрядного сумматора

Входы сумматора			Выходы сумматора	
a_i	b_i	c_{i-1}	s_i	c_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

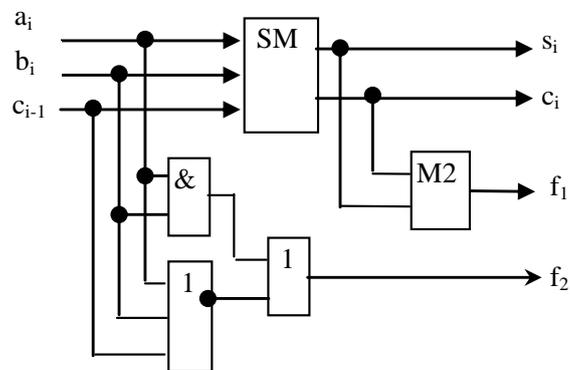


Рис. 7.11. Самопроверяемая схема контроля одноразрядного сумматора

Введение схем контроля предполагает введение дополнительных выводов для управления ими или индикации результатов работы схем контроля. Обычно имеются ограничения на количество выводов в электронном изделии. Поэтому проектировщики при разработке схем контроля стараются уменьшить число выводов, дополнительно вводимых в целях контроля.

Для минимизации числа дополнительных выводов применяют самопроверяемые схемы сжатия. Схема сжатия, приведенная на рис. 7.12, обеспечивает объединение сигналов с выходов двух ССК в один общий сигнал. Если на одном из входов схемы сжатия сигналы f_{11} , f_{12} либо f_{21} , f_{22} принимают значение 00 или 11 (что является признаком отказа), то сигнал на выходе также будет свидетельствовать о наличии отказа.

Для программно-управляемых объектов, содержащих большое число регистров и счетчиков, широко применяются методы сканирования, позволяющие

получать для целей диагностирования информацию о состояниях элементов памяти, в том числе непосредственно в процессе применения объекта по назначению (т.е. в режиме on-line). Более подробно об аппаратурных реализациях метода сканирования см. в пп. 7.4.

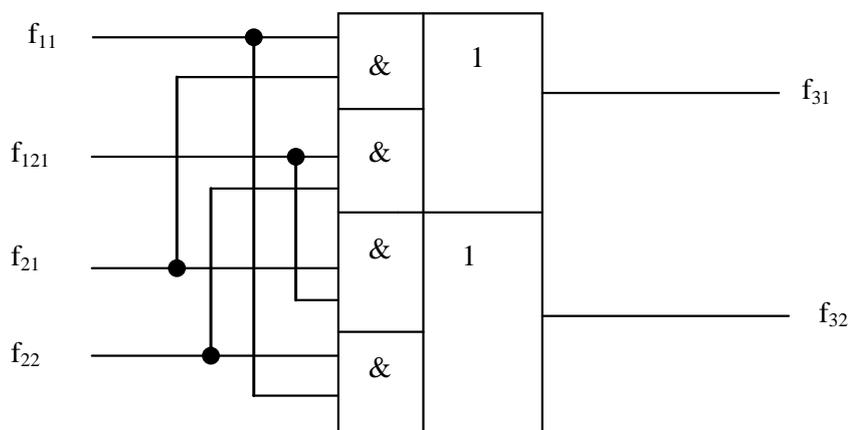


Рис. 7.12. Самопроверяемая схема сжатия

Контроль объектов с аналоговыми сигналами

Сигнал называется аналоговым (непрерывным), если в заданном интервале он может иметь бесконечное число значений. Первично измеряемыми в целях контроля параметрами аналоговых сигналов чаще всего являются напряжение, ток, частота, фаза, период (или длительность).

Практическая разработка схем встроенного контроля схем с аналоговыми входными и выходными сигналами менее формализована, чем в случае объектов с цифровыми сигналами, а зачастую носит эвристический характер. Причинами этого являются:

- случайные электромагнитные воздействия, токи в соседних проводниках, изменения из-за старения или нагрева электронных компонент объекта диагностирования и т.п., которые влияют на значения аналоговых сигналов;
- при повторной подаче одних и тех же входных воздействий значения выходных аналоговых сигналов будут несколько иными;
- для проверки «правильности» аналогового сигнала, результат его измерения сравнивают с некоторым допуском, точное определение которого само по себе является сложной задачей;
- измерение и сравнение с допуском происходит с некоторыми погрешностями;
- существенно более сложные формальные модели и вычислительная трудоемкость расчетов.

Рассмотрим некоторые методы построения встроенных систем контроля объектов с аналоговыми контролируруемыми сигналами.

Часто в них можно выделить точки, уровень сигнала в которых должен иметь определенное значение при нормальном функционировании объекта кон-

троля (в частности, это напряжения питания, ток в антенне и др.). Тогда контроль функционирования можно организовать, введя специальные (достаточно простые) схемы контроля уровней напряжений (токов) в таких точках. Для индикации того, что питание имеет нормальный уровень, обычно к схеме контроля подключают светодиод, устанавливаемый на передней панели объекта.

Для постоянного контроля таких параметров, как частота, ток, напряжение на передней панели могут устанавливать простые стрелочные или цифровые измерители.

Из формальных методов организации встроенного самоконтроля следует указать методы контроля по обобщенным характеристикам или метод избыточных переменных.

Контроль по обобщенной характеристике предполагает выделение некоторого параметра (возможно нескольких), который в достаточно полной мере отражает техническое состояние объекта диагностирования. Например, если объект диагностирования в ходе обработки данных производит вычисления $\sin \alpha$ и $\cos \alpha$, то в качестве обобщенной характеристики может использоваться сумма $\sin \alpha + \cos \alpha = 1$. Если это равенство выполняется (в пределах установленного допуска), то имеет место правильное функционирование объекта диагностирования.

Метод избыточной переменной предполагает введение некоторой дополнительной функции (переменной), которая постоянно вычисляется средствами контроля. В случае, если ее значение равно нулю (с учетом некоторого допуска), считается, что объект правильно функционирует.

В общем виде метод избыточной переменной можно представить следующим образом.

Пусть объект диагностирования описывается системой уравнений:

$$y_1 = f_1(x_1, \dots, x_m, t)$$

.....

.....

$$y_n = f_n(x_1, \dots, x_m, t)$$

Дополним ее еще одним, избыточным, уравнением:

$$y_{n+1} = f_{n+1}(x_1, \dots, x_m, t),$$

где $y_{n+1} = - \sum_{i=1}^n m_i y_i$; m_i – масштабирующие коэффициенты.

Изменяя эти коэффициенты, можно менять чувствительность системы контроля к изменениям тех или иных характеристик объекта диагностирования. Это уравнение называют **уравнением избыточной переменной**.

Очевидно, если вычисления вышеприведенной системы функций происходят правильно, то $\Delta = \sum_{i=1}^{n+1} m_i y_i = 0$. Выражение Δ называют **контрольным условием**. Контроль по методу избыточных переменных состоит в непрерывной или периодической проверке равенства нулю контрольного условия Δ .

Рассмотрим случай, когда объект диагностирования состоит из двух инвертирующих суммирующих усилителей (рис. 7.13а), реализующих функции:

$$y_1 = -a_1x_1 - a_2x_2$$

$$y_2 = -a_3x_1 - a_4x_3,$$

где x_i – сигнал на i -м входе объекта, a_i – коэффициент усиления по i -му входу.

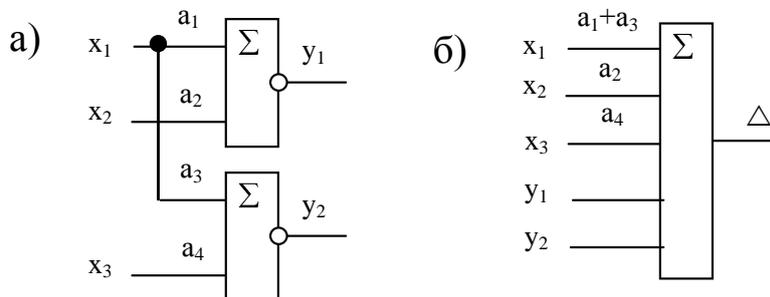


Рис. 7.13. Объект диагностирования (а) и схема контроля (б) по методу избыточной переменной

Для контроля по методу избыточных переменных введем дополнительно функцию $y_{n+1} = -\sum_{i=1}^n m_i y_i$. Примем все $m_i=1$. Тогда уравнение избыточной переменной примет вид: $y_3 = -(y_1+y_2)$. В результате получим контрольное условие:

$$\Delta = -\sum_{i=1}^{n+1} m_i y_i = -[(-a_1x_1 - a_2x_2) + (-a_3x_1 - a_4x_3) - (y_1+y_2)] = a_1x_1 + a_2x_2 + a_3x_1 + a_4x_3 +$$

$+y_1+y_2=0$ или $\Delta = x_1(a_1+a_3) + a_2x_2 + a_4x_3 + y_1 + y_2 = 0$. Элемент, реализующий это контрольное условие, показан на рис. 7.13б. Если на некотором такте функционирования объекта значение $\Delta \neq 0$, то это является признаком отказа (ошибки). Учитывая особенности аналоговых сигналов и их измерений, обычно значение Δ сравнивается не с нулем, а с некоторым допуском – интервалом (т.е. считается, что сигнал рассогласования Δ при нормальной работе может несколько, в пределах допуска, отличаться от нулевого значения).

Практические правила построения контролепригодных схем

Разработчики аппаратуры выработали целый ряд практических рекомендаций, следование которым повышает уровень контролепригодности создаваемых электронных изделий. Некоторые из этих рекомендаций приведены ниже.

1. **Обеспечивайте установку элементов памяти (триггеров, регистров, счетчиков) в заданное состояние** при включении питания. Если при включении питания элементы памяти устанавливаются в произвольные заранее неизвестные состояния, то существенно усложняются вопросы диагностирования.

Существуют разные варианты решения проблемы установки. На рисунке 7.14 показан прием автоматической установки в состояние «0» с помощью

RC-цепи. При включении питания на обкладках конденсатора С напряжение всегда ноль вольт и через элемент И2 на вход R триггера Т подается сигнал «0», который принудительно устанавливает триггер в состояние «0» (т.е. на его выходе устанавливается значение «0»). Через некоторое время конденсатор заряжается и элемент 2 начинает работать в режиме повторителя.

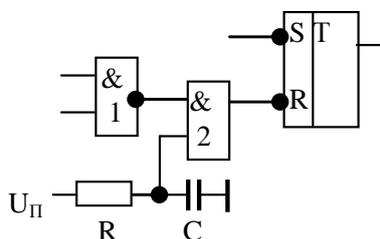


Рис. 7.14. Схема принудительного сброса триггера

2. Для повышения управляемости и наблюдаемости **вводите дополнительные контрольные точки**. Это могут быть специальные контактные площадки или штыри на печатной плате, специальные дополнительные разъемы и т.п. Через них могут подаваться тестовые воздействия или измеряться значения сигналов в соответствующих точках электрической схемы объекта диагностирования.

3. **Используйте свободные выводы** имеющихся **разъемов** для целей диагностирования.

4. **Чтобы сократить количество внешних выводов**, используемых для целей диагностирования, применяйте мультиплексоры (рис. 7.15а) и демультиплексоры (рис. 7.15б). Они позволяют в режиме диагностирования переключать внешние выходы, к которым подсоединены входы/выходы объекта диагностирования, на контрольные точки. Таким образом, одни и те же выводы внешнего разъема объекта используются как в обычном режиме работы, так и в режиме диагностирования. Соответствующим выбором контрольных точек можно существенно повысить управляемость и наблюдаемость.

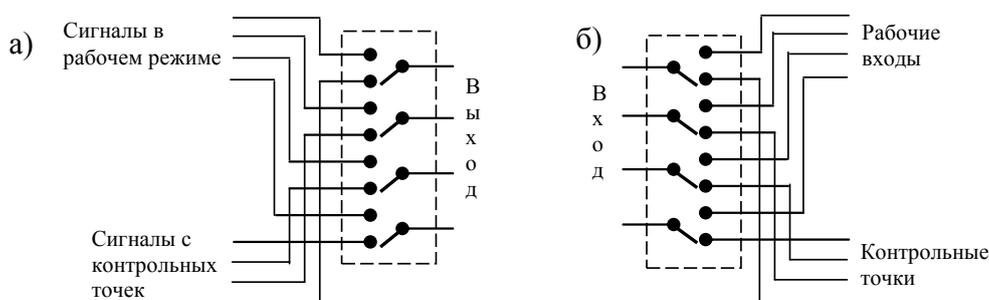


Рис. 7.15. Применение мультиплексора (а) и демультиплексора (б) для подключения контрольных точек к внешним выводам

5. При наличии трудноуправляемых подсхем (функциональных узлов) для улучшения управляемости используйте линии блокировки и дополнительные линии управления (рис. 7.16).

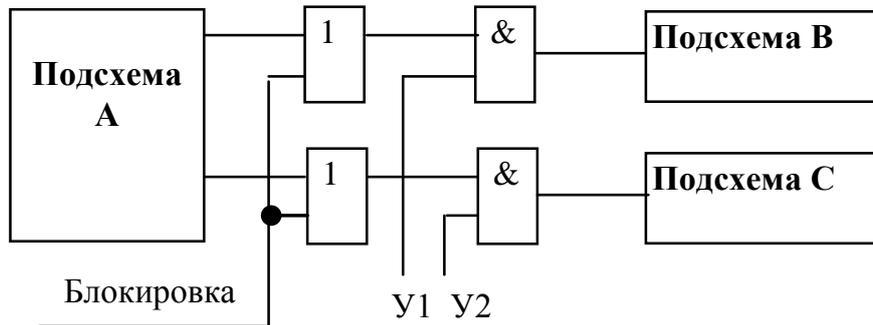


Рис. 7.16. Повышение управляемости при наличии подсхем

На рисунке сигнал «1» на линии «Блокировка» отключает выходы подсхемы А от входов подсхем В и С. Дополнительные линии управления У1 и У2 позволяют подать на входы подсхем В и С необходимые для диагностирования сигналы.

6. Для улучшения контролепригодности многоразрядных счетчиков создавайте возможность их разделения на части и отдельного управления работой в режиме диагностирования. Разрыв можно создать с помощью логического элемента, управляемого сигналом «Блокировка», а через линию У1 можно имитировать подачу счетных импульсов на вход части 2 счетчика (рис. 7.17). В нормальном режиме работы через линию «Контрольная точка» можно дополнительно контролировать сигнал на выходе части 1 счетчика.

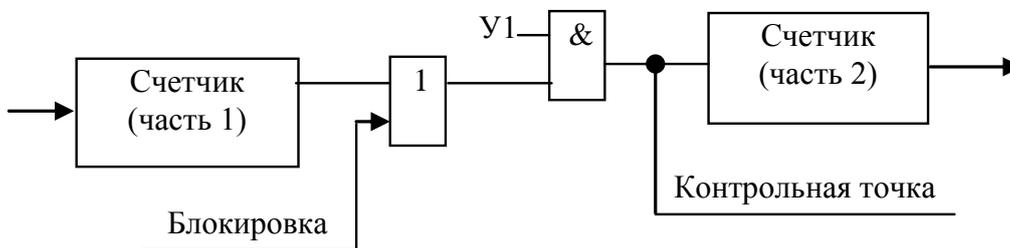


Рис. 7.17. Разделение многоразрядного счетчика на части

7. Предусматривайте возможность разрыва петель обратной связи и имитацию ее сигналов извне. Разрыв создают с помощью перемычек или специальной логики (рис. 7.18). В режиме диагностирования перемычка снимается либо подается сигнал «Блокировка». Тогда обратная связь разрывается, и к концам обратной связи возможен доступ извне через выводы А и В

внешнего разъема (при использовании переключки, рис. 7.18а) либо по линии «Управление» (рис. 7.18б).

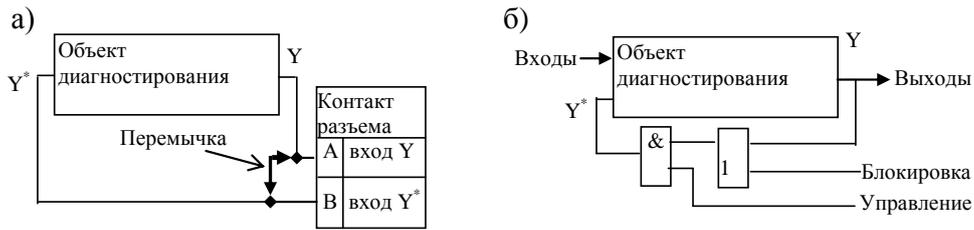


Рис. 7.18. Разрыв обратной связи с помощью
а) переключки и б) логических элементов

8. При наличии в объекте диагностирования встроенного тактового генератора обеспечивайте возможность его отключения и подачи тактовых сигналов извне. Как и в случае обратной связи, для этого ставят переключки (рис. 7.19а) или используют дополнительные логические элементы-ключи (рис. 7.19б). Подключив внешний генератор, можно уменьшить тактовую частоту или перейти к пошаговому выполнению алгоритма работы объекта диагностирования, что зачастую облегчает процессы нахождения места неисправности.

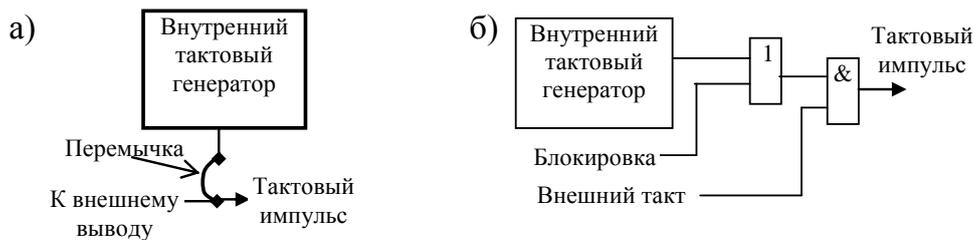


Рис. 7.19. Способы подмены тактового генератора: с использованием переключки (а) или логических элементов (б)

9. При выборе элементной базы микропроцессорных систем отдавайте предпочтение микросхемам, приспособленным для периферийного сканирования (более подробно см. в пп. 7.4) или других способов диагностирования.

10. Проектируйте схемы из функционально-законченных подсхем, что упрощает процессы поиска неисправностей и автоматизацию построения тестов.

11. Аналоговые и цифровые подсхемы старайтесь размещать на разных печатных платах или их участках. Это связано с тем, что проверки аналоговых и цифровых узлов обычно требуют применения разного оборудования и методик для проверок.

12. На печатной плате располагайте элементы (микросхемы, резисторы, конденсаторы и др.) **однотипных узлов** одинаковым образом, старайтесь унифицировать расположение выводов на разъемах (например, отводить под шину адреса группу последовательно расположенных выводов в одном ряду контактов разъема). Для установки наиболее сложных для проверки микросхем (микропроцессоры, микросхемы памяти большого объема и т.п.) используйте колодки.

13. Если разрабатываемое изделие будет выпускаться серийно, обязательно **учитывайте особенности диагностического оборудования**, которое будет применяться при производственном контроле.

14. На печатной плате и/или передней панели объекта **устанавливайте индикаторы** (жидкокристаллический дисплей, светодиоды), которые информируют о наличии питания, отказах или логических значениях сигналов в ключевых точках (например на шине управления). Наличие таких индикаторов позволяет иметь какую-то информацию о характере отказа даже при неработающем основном узле отображения информации.

Заметим, что существуют решения, предполагающие запоминание схемой встроенного контроля кода ошибки и установку специального разъема для подключения внешнего переносного считывателя-индикатора (рис. 7.20).



Рис. 7.20. Считыватель кодов неисправностей Creader (для встроенных систем контроля автомобилей)

15. В случае, когда объект диагностирования является программно-управляемым, **предусматривайте специальную программу – обработчик отказов**, зафиксированных средствами встроенного контроля. Одна из ее основных функций – вывод на имеющееся в объекте устройство отображения подробной информации о характере обнаруженного отказа.

16. Если функционирование объекта предполагает удаленный обмен информацией, **используйте кодирование для обнаружения искажений кодов** и исправления этих искажений при передаче (более подробно о таком кодировании см. в пп. 7.2).

17. Всегда уделяйте большое внимание качеству и объему конструкторской, технологической и эксплуатационной документации, в том числе касающейся вопросов организации диагностирования. Это особенно важно, если к наладке, эксплуатации и ремонту не привлекается разработчик.

7.4. Стандарты периферийного сканирования

Общие принципы организации периферийного сканирования

Современные технологии проектирования изделий электроники предполагают, что тестопригодность должна быть обеспечена еще на стадии разработки. В случае применения микросхем высокой степени интеграции, в частности микропроцессоров, разработчики все чаще делают выбор в пользу применения микросхем, поддерживающих так называемое **периферийное (граничное) сканирование**. Технология и принцип аппаратурной реализации **периферийного сканирования** (Boundary Scan – **BS**) были впервые предложены в 1985 г., а в 1990 г. оформлены стандартом **IEEE 1149.1** (IEEE Standard Test Access Port and Boundary-Scan Architecture).

Периферийное сканирование используется как средство доступа к внутренним регистрам микросхем для наблюдения за их состоянием в процессе отладки и диагностирования, что существенно упрощает определение причин отказов и процессы отладки программно-аппаратных систем. В последующем эта идея получила развитие и в других стандартах: IEEE 1194.4 и др.

Технология периферийного сканирования была апробирована, в частности, в 80-х годах прошлого столетия в компьютерах серии IBM 370. Придуманно оно было с целью упростить диагностирование устройств, содержащих большое число элементов памяти, недоступных для непосредственного контроля. Периферийное сканирование позволяет получать информацию о состояниях внутренних элементов памяти диагностируемой схемы, что существенно упрощает решение задач диагностирования. Несомненно, положительной особенностью периферийного сканирования является то, что схемы для его реализации предполагают введение всего нескольких дополнительных выводов. Так, достаточно ввести всего один дополнительный выход, чтобы получать информацию о состояниях любого числа внутренних элементов памяти контролируемого объекта.

Схемы на рис. 7.21 и 7.22 иллюстрируют идеологию периферийного сканирования.

Допустим, синхронную схему, для которой надо организовать периферийное тестирование, можно представить в виде структуры рис. 7.21. На рисунке M – комбинационная часть схемы; Y_1, \dots, Y_n – элементы памяти. Видно, что состояния элементов памяти непосредственно не контролируются. В результате, в случае неисправности, путем анализа значений выходных сигналов может оказаться невозможным определить точное место неисправности, так как неизвестны действительные состояния элементов памяти.

Периферийное сканирование организуется с целью получения информации о состояниях элементов памяти, которые непосредственно не наблюдаются. Для этого исходная схема видоизменяется, как показано на рис. 7.22. В ней между элементами памяти (Y_1, \dots, Y_n) и схемой М ставятся ключи. Кроме того, дополнительно вводятся два канала управления («Управление сканированием», «Управление режимом») и один дополнительный выход «Выход сканирования».

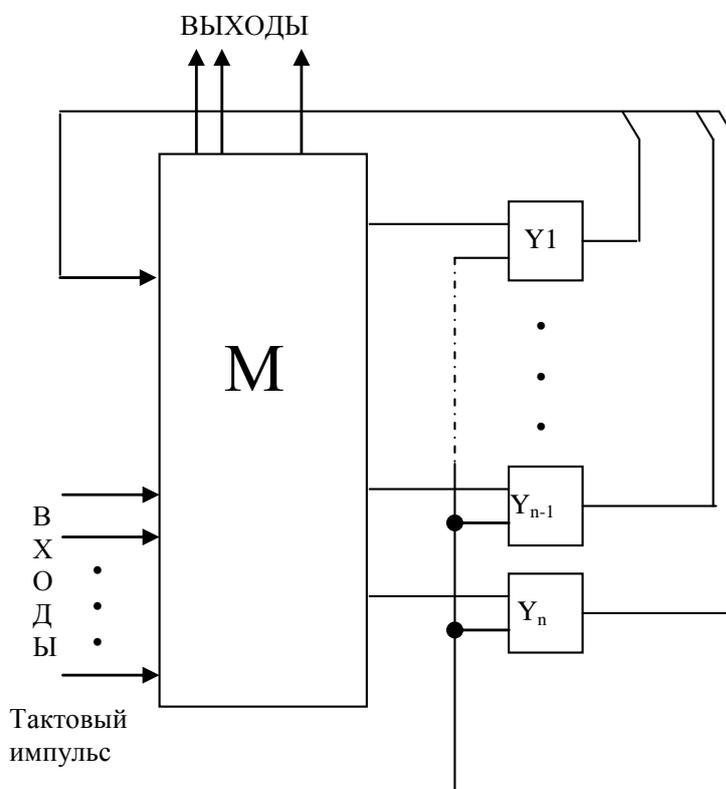


Рис. 7.21. Модель исходной схемы

Для нормальной работы по сигналу «Управление режимом» ключи устанавливаются в состояние (верхнее положение), при котором элементы памяти Y_1, \dots, Y_n непосредственно подключены к выходам схемы М. В случае перехода в режим сканирования ключи устанавливаются в нижнее положение, и элементы памяти соединяются в последовательную цепь, создавая сдвиговый регистр. В последующем по тактовым импульсам состояния отдельных разрядов созданного регистра последовательно поступают на вывод «Выход сканирования». В результате, наблюдая за сигналом на дополнительном выходе «Выход сканирования», можно получить информацию о состояниях ранее недоступных элементов памяти. Через вход «Управление сканированием» в регистр можно записать любой необходимый код.

Схема на рис. 7.22 сильно упрощена. Практические схемы реализации идеи периферийного сканирования значительно сложнее. В частности, они содержат специальные дополнительные регистры для «копирования» состояний элементов памяти контролируемой логики для последующего вывода этих состояний через выход сканирования, что позволяет вести «сканирование», не нарушая нормальную работу объекта.

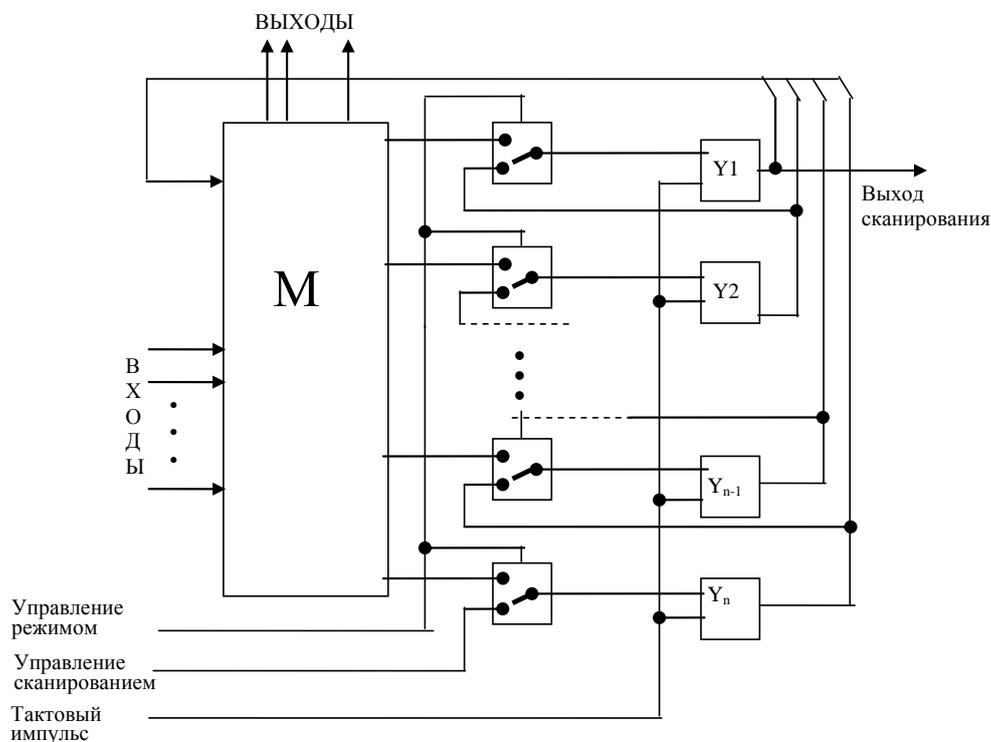


Рис. 7.22. Модернизация схемы рис. 7.21 с целью обеспечения режима периферийного сканирования

Стандарты периферийного сканирования представляют описание специализированного аппаратного интерфейса, разработанного для тестирования цифровых процессоров. Наиболее известны стандарты под аббревиатурой **JTAG** (сокращение англ. Joint Test Action Group от названия группы ведущих производителей электронных компонентов, которая была создана для решения проблем тестирования сложных электронных устройств). Возможности JTAG и его варианты повсеместно используются для отладки и программирования микропроцессорных устройств, в частности микроконтроллеров.

Сегодня эти стандарты все шире используют производители микросхем при разработке электрических схем микросхем. Так, практически все современные микроконтроллеры имеют интерфейсы для организации периферийного тестирования.

Стандарт IEEE 1149.1/JTAG периферийного сканирования

Базовый стандарт IEEE 1149.1/JTAG определяет относительно простую шину с несложным протоколом. В соответствии со стандартами JTAG контролируемый объект (микросхема, печатная плата, микроконтроллер) должна иметь дополнительно специализированный тестовый порт TAP (Test Access Port). Управление портом осуществляет специальный контроллер. Он может устанавливаться на самом контролируемом объекте или вне его. Управление собственно контроллером обычно выполняется через компьютер через один из его интерфейсов. Стандарт развивается, в частности появились версии IEEE 1149.4, IEEE 1149.6.

Контроллер JTAG и соответственно тестовый порт TAP имеют четыре или пять выводов: TCK, TMS, TDI, TDO, TRST (рис. 7.23).

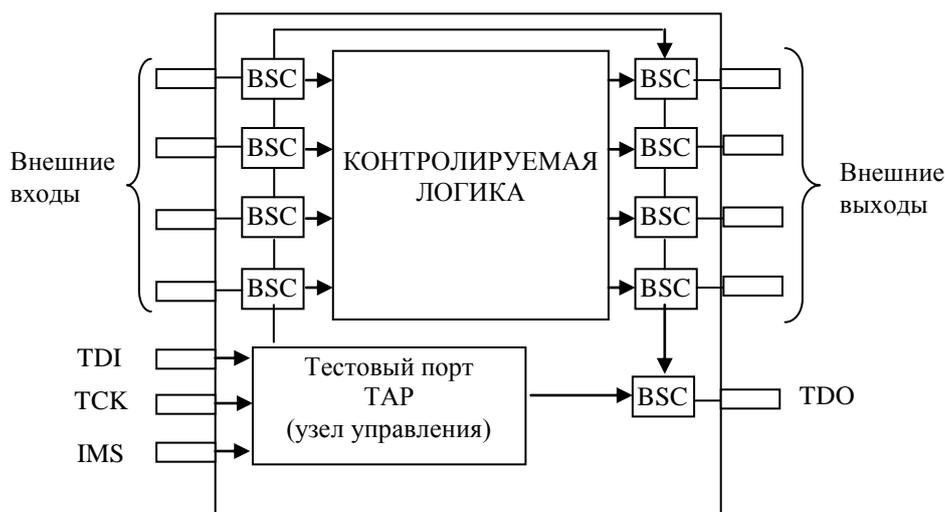


Рис. 7.23. Структура микросхемы со встроенными элементами интерфейса JTAG

Функциональное назначение этих выводов:

TDI (вход тестовых данных) – входная последовательность бит. Поступают в микросхему с этого вывода по переднему фронту сигнала TCK;

TDO (выход тестовых данных) – выходные данные, выдаваемые в виде последовательности бит. Команды и данные выдвигаются из микросхемы с этого вывода по заднему фронту сигнала TCK;

TCK (вход тестового тактирования) – сигнал синхронизации (тактирования) работы схем управления периферийным сканированием. Максимальная частота сканирования периферийных ячеек равна 16 МГц;

TMS (вход управления тестированием) – обеспечивает выбор режима тестирования;

TRST (сигнал для инициализации порта тестирования) – может отсутствовать, т.к. инициализация возможна путем подачи соответствующей последовательности сигналов на вход **TMS**.

В аппаратном плане тестовый порт **TAP** – это некоторый набор регистров различного назначения и схем управления. Его основу составляют специальные блоки ввода/вывода (ячейки) – **BSC**. Через них обеспечивается доступ к входам и выходам контролируемой логики. Их не меньше, чем число логических входов и выходов, причем все они соединяются не только с выводами контролируемой логики, но и между собой (рис. 7.23).

Ячейки **BSC** устанавливаются на границе внешние выводы – контролируемая логика, при этом с их помощью можно запомнить значения сигналов на этих выводах и вывести их через специальный дополнительный вывод (**TDO**) для внешнего анализа. Отсюда и название технологии контроля: периферийное сканирование (**Boundary Scan**).

Если объект контроля содержит двунаправленные выводы, то к такому выводу подключается дополнительная ячейка **BSC** для обеспечения двунаправленного движения сигналов.

Последовательно соединяемые ячейки **BSC** образуют цепочку, называемую **JTAG-путь** (**JTAG scan path**), по сути, представляющую сдвиговый регистр, что позволяет организовать вывод информации из ячеек **BSC** через один внешний вывод **TDO**. Режим их работы определяется инструкциями **TAP**. В обычном состоянии **BSC** не оказывают никакого влияния на работу контролируемого устройства, и сигналы свободно проходят через них. В режиме тестирования можно ввести (задать) через вход **TDI** значения тестовых сигналов и через **BSC** подать их на входы контролируемой логики. В других режимах в ячейки **BSC** могут «переписываться» значения сигналов с выводов контролируемой логики и затем выводятся в последовательном коде через вывод **TDO**.

Стандарты **JTAG** допускают наличие в объекте контроля нескольких микросхем (узлов) с портами **TAP**. Тогда их выводы, **TSM**, **TCK** и **TRST** соединяются параллельно, а выводы **TDO**, **TDI** – последовательно. В объекте могут быть и элементы (узлы), не имеющие портов **TAP** (рис. 7.24), при этом входы и выходы элементов могут соединяться в соответствии с обычной логикой их функционирования.

Разработчики стандартов позаботились о том, чтобы можно было достаточно «просто» организовать отдельное тестирование каждой микросхемы или узла, приспособленного под **JTAG**, автоматически определять вариант реализации **JTAG** и применять стандартные процедуры тестирования, созданные разработчиками микросхем.

TAP порт – по сути, некоторый контроллер, включающий несколько регистров и схемы управления их работой:

IR – регистр инструкций (**Instruction Register**). В него заносится код, поступающий через вход **TDI**;

BPR – регистр обхода (**Bypass Register**). Служит для организации работы объекта в нормальном режиме;

DID – регистр идентификации устройства (**Device Identification Register**) длиной 32 бита. Содержит идентификатор производителя, код устройства

и номер версии, по которым TAP может определить особенности тестируемой микросхемы;

BSR – сдвигающий регистр, в котором каждая ячейка BSC представлена битом.

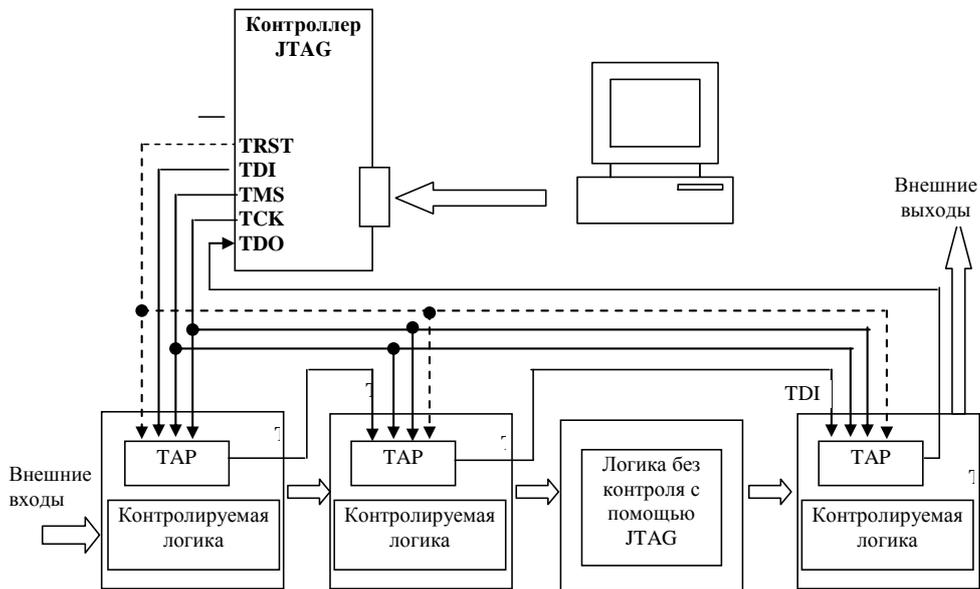


Рис. 7.24. Организация периферийного сканирования в соответствии со стандартом IEEE 1149.1

Управление работой тестового порта TAP осуществляется по сигналам TDI, TMS, TCK от контроллера JTAG. Алгоритм работы зависит от состояния его схемы управления и инструкций, которые через вывод TDI вводятся в регистр IR.

Оператор вводит программу тестирования на специальном языке BSDL (Boundary Scan Description Language). Производители микросхем предоставляют информацию о реализации JTAG-контроллера и соответствии тестового регистра внешним выводам устройства в специальных файлах в стандартизованных форматах, например на том же языке BSDL. Существуют специальные прикладные программы для организации тестирования в соответствии со стандартами JTAG. Они позволяют автоматически генерировать тестовые программы, интегрировать их в системы производственного внутрисхемного контроля, использовать С-подобные языковые средства для организации нестандартных режимов тестирования, отображать ход тестирования и др.

Стандарт IEEE 1149 определяет набор обязательных команд (инструкции) контроллера JTAG и функционирование TAP при их выполнении, которые должны поддерживаться всеми устройствами любых производителей. Это команды инициализации, задания режимов работы, считывания статуса, считывания идентификатора устройства и другие. Кроме того, для каждого тестируемо-

го устройства могут быть определены свои специфичные расширения системы команд.

К числу обязательных инструкций относят:

BYPASS – переход в режим, при котором схемы интерфейса JTAG не влияют на прохождение сигналов через контролируемую логику;

SAMPLE – копирование в память порта TAP значений на внешних входах/выходах контролируемой логики;

PRELOAD – загрузка данных для последующей выдачи через ячейки BSC на выходы контролируемой логики;

EXTEST – проверка цепей между выходами и входами микросхем (узлов), выполненных в соответствии с требованиями JTAG.

Функционирование TAP и ячеек BSC зависит не только от инструкций, генерируемых контроллером JTAG, но и от внутреннего состояния TAP. В стандарте даются правила перехода из одного состояния в другое и режимы работы в них TAP и его регистров.

В одну цепочку могут подключаться разные устройства разных производителей со своими специфичными особенностями. Так как стандарт предусматривает наличие регистра-идентификатора, то по коду в регистре DID тестовая программа может определить особенности каждого тестируемого узла и учесть их при выборе управляющих инструкций.

Современные микроконтроллеры содержат перепрограммируемую память, в том числе типа флэш-памяти. Разработчики микроконтроллеров используют возможности интерфейса JTAG для перепрограммирования памяти, подключая ячейки BSC к выводам памяти процессора или микросхемы флэш-памяти.

Заметим также, что некоторые шины ПЭВМ (например PCI) предусматривают линии для работы по стандарту JTAG.

Стандарт IEEE 1149.4 для схем с аналоговыми сигналами

В 1999 г. был принят стандарт IEEE 1149.4, в котором ранее существовавший стандарт IEEE 1149.1 развивался применительно к объектам, имеющим не только цифровые, но и аналоговые входные/выходные сигналы. Общая схема организации тестирования в соответствии со стандартом IEEE 1149.4 приведена на рис. 7.25.

В соответствии со стандартом в разрыв каждого цифрового вывода контролируемой схемы включится элемент (ячейка) типа «D», а аналоговых – типа «ABM» (рис. 7.25). На ячейку типа D возлагаются функции, аналогичные функциям ячейки BSC (рис. 7.23). Ячейки ABM подключаются исключительно к выводам с аналоговыми сигналами. При этом они дополнительно подключаются к двум внутренним аналоговым шинам AB1 и AB2. Кроме того, все ячейки D и ABM последовательно соединены с друг другом.

Для управления работой встроенных схем контроля в объект встроены тестовый порт TAP и **блок интерфейса шин тестирования ТВИС** (Test Bus Interface Circuit). Блок **ТВИС** введен для коммутации двух внутренних аналоговых шин на два внешних вывода AT1 и AT2.

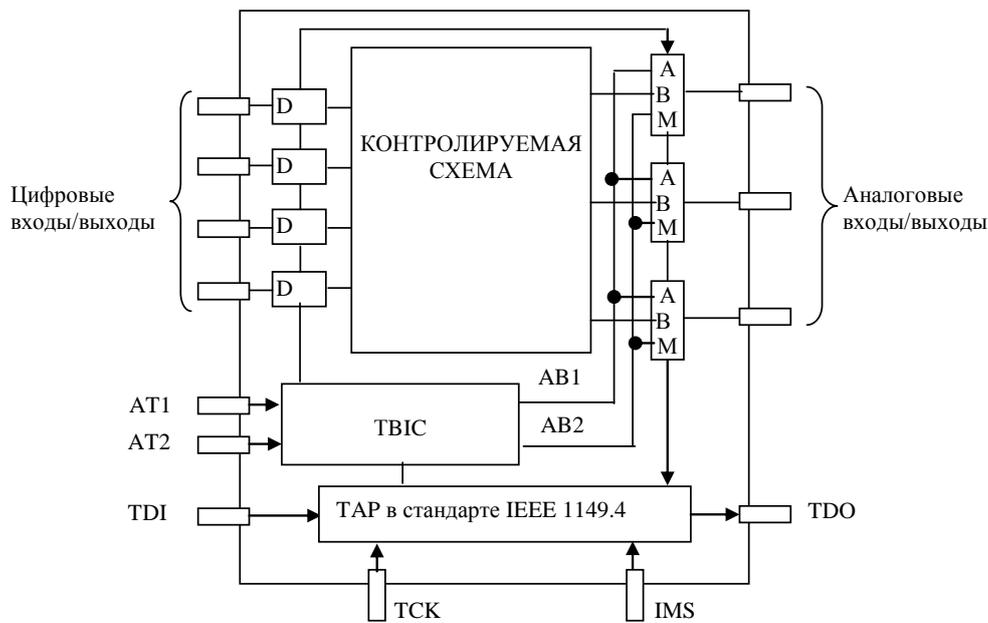


Рис. 7.25. Организация контроля по стандарту IEEE 1149.4

Как и в стандарте IEEE 1149.1, управление осуществляется извне сигналами TDI, TMS и TCK, а цифровую информацию о сигналах на входах/выходах контролируемой схемы можно вывести через вывод TDO. Однако особенностью стандарта IEEE 1149.4 является введение дополнительно для работы с аналоговыми сигналами двух выводов AT1 и AT2 (рис. 7.25). С помощью блока ТВІС линии внутренних шин AB1 и AB2 можно коммутировать на выходы AT1 и AT2. При этом AT1 и AT2 могут использоваться как для ввода, так и для вывода аналоговых сигналов в реальном времени. Иными словами, например, можно одновременно контролировать значения аналоговых сигналов на двух аналоговых выходах контролируемой схемы, причем не нарушая нормальной работы схемы. Выбор ячеек ABM, подключаемых к внутренним аналоговым шинам AB1 и AB2, осуществляется путем подачи соответствующих кодов на выходы TDI, TMS и TCK. В режиме обычного функционирования контролируемого объекта ячейки D и ABM не оказывают никакого влияния на прохождение сигналов.

Режимы работы определяются инструкциями и состоянием порта TAP. В части работы с сигналами на цифровых входах/выходах в стандартах IEEE 1149.1 и IEEE 1149.4 много общего. В то же время специально для работы с аналоговыми сигналами стандартом IEEE 1149.4 введена дополнительная инструкция PROBE. Ее назначение – коммутация аналоговых входов/выходов на внутренние аналоговые шины и внешние выходы AT1 и AT2.

Контрольные вопросы

1. Дайте определение контролепригодности (по ГОСТ 19919-74).
2. Какими показателями оценивается приспособленность к диагностированию (по ГОСТ 26656-85)?
3. Дайте определение управляемости и наблюдаемости.
4. Для линии e на рис. 7.26 определите $l(0)$ управляемость и наблюдаемость (по числу линий, которые надо установить в конкретные состояния).

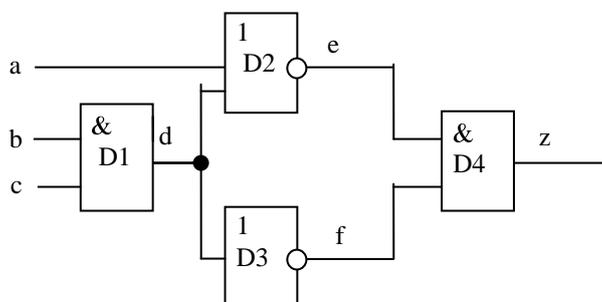


Рис. 7.26. Схема для контрольного вопроса 4

5. Дайте определение понятия «кодированное расстояние». Какое кодированное расстояние между кодированными словами 01010101 и 10100101?
6. Условия обнаружения и условия исправления ошибки кратности г.
7. При передаче 6-разрядных кодов осуществляется контроль четности. Добавьте слева к нижеперечисленным кодам контрольный разряд: 101010, 111111, 000000, 110001, 010000, 111010, 010101.
8. Определите, сколько контрольных разрядов потребуется для кодирования кодом Хэмминга информации (слова) длиной: 16, 32, 64 и 128 бит. Укажите номера разрядов, которые будут отведены под контрольные разряды?
9. Укажите, контроль каких разрядов передаваемого слова осуществляется контрольными разрядами, которые находятся в первом и втором разрядах передаваемого слова, если контроль осуществляется с помощью кода Хэмминга и исходная информация для передачи (без контрольных бит) содержит 16 или 32 разряда.
10. Определите контрольное число для файла 0100011111111111010101010101010111111, если при его определении файл разбивается на слова длиной один байт.
11. Какова вероятность обнаружения ошибки кратности $r+1$ с помощью циклического кода?
12. Какие элементы вводятся в проектируемое устройство для повышения контролепригодности?
13. В чем отличие контроля по методу дублирования и с применением мажоритарных схем?
14. Какая схема контроля называется самопроверяемой?
15. Как организуется контроль по методу избыточной переменной?

16. Перечислите основные правила, которые следует соблюдать при разработке схем с целью повышения контролепригодности.
17. Основные аппаратные дополнения в контролирующую схему при использовании стандарта IEEE 1149.1 (JTAG).
18. Назначение дополнительных линий, вводимых в схему в соответствии со стандартом IEEE 1149.1.
19. Назначение ячеек BSC, вводимых в контролирующую схему по стандарту IEEE 1149.1.
20. Как организуется контроль аналоговых сигналов по стандарту IEEE 1149.4?

Рекомендуемая литература

- Беннеттс, Р.Дж. Проектирование тестопригодных логических схем / Р.Дж. Беннеттс; пер. с англ. – М.: Радио и связь, 1990. – 176 с.
- Городецкий, А. Бесконтактное внутрисхемное тестирование аналого-цифровых схем / А. Городецкий // Радиоэлектроника и информатика. – 2005. – № 5.
- Гук, М. Аппаратные интерфейсы ПК: энциклопедия / М. Гук. – СПб.: Питер, 2003. – 528 с.
- Коган, Б.М. Основы эксплуатации ЭВМ / Б.М. Коган, И.Б. Мкртумян. – М.: Энергоатомиздат, 1988. – 432 с.
- Пархоменко, П.П., Основы технической диагностики. Оптимизация алгоритмов диагностирования, аппаратные средства) / П.П. Пархоменко, Е.С. Согомоян; под ред. П.П. Пархоменко. – М.: Энергия, 1981. – 320 с.
- Согомоян, Е.С. Самопроверяемые устройства и отказоустойчивые системы / Е.С. Согомоян, Е.В. Слабаков. – М.: Радио и связь, 1989. – 208 с.
- www.jtag.com

Тема 8. ТЕСТИРОВАНИЕ ПРОГРАММ

8.1. Основные понятия и цели тестирования

Жизненный цикл программы

Очень многие электронные устройства относятся к программно-управляемым, например устройства, содержащие микропроцессор. При этом причиной отказа при работе устройства могут быть дефекты не только аппаратуры, но и программ. Более того, задача построения тестов для диагностирования микропроцессорного устройства, по сути, есть задача создания некоторой программы, правильное выполнение которой свидетельствует об исправности или работоспособности диагностируемого объекта.

В программно-управляемых объектах при диагностировании зачастую трудно определить, что было причиной отказа – исполняемая программа или аппаратная часть объекта. Поиск причин отказа существенно упрощается, если есть уверенность в отсутствии дефектов в исполняемой программе. Поэтому в процессе создания программы она в обязательном порядке подвергается тестированию, возможно неоднократно.

Как и в случае технических изделий, в «жизни» (жизненного цикла) любой программы или базы данных можно выделить несколько этапов, начиная с разработки обоснования и задания на разработку до завершения эксплуатации.

В международных стандартах **жизненный цикл (ЖЦ)** рассматривают как набор этапов, работ и операций, регламентирующих ведение работ по созданию и эксплуатации программного продукта. В зависимости от требований к разработке и условий эксплуатации жизненный цикл может быть представлен с различной степенью детализации этапов.

На укрупненном уровне ЖЦ может включать небольшое число этапов, например:

- 1) анализ и выработка требований,
- 2) проектирование,
- 3) реализация,
- 4) внедрение и эксплуатация.

Одна из более детальных схем жизненного цикла приведена на рис. 8.1. В ней уже 8 этапов.

В общем случае тестирование необходимо на каждом этапе ЖЦ.

Несмотря на существенные достижения в области проектирования программных продуктов, решение задач их тестирования – до сих пор во многом

неформальный процесс. Причем статистические исследования показывают, что выявление дефектов программ и их устранение на первых этапах жизненного цикла в десятки-сотни раз менее затратное, чем на этапах применения программы по назначению.

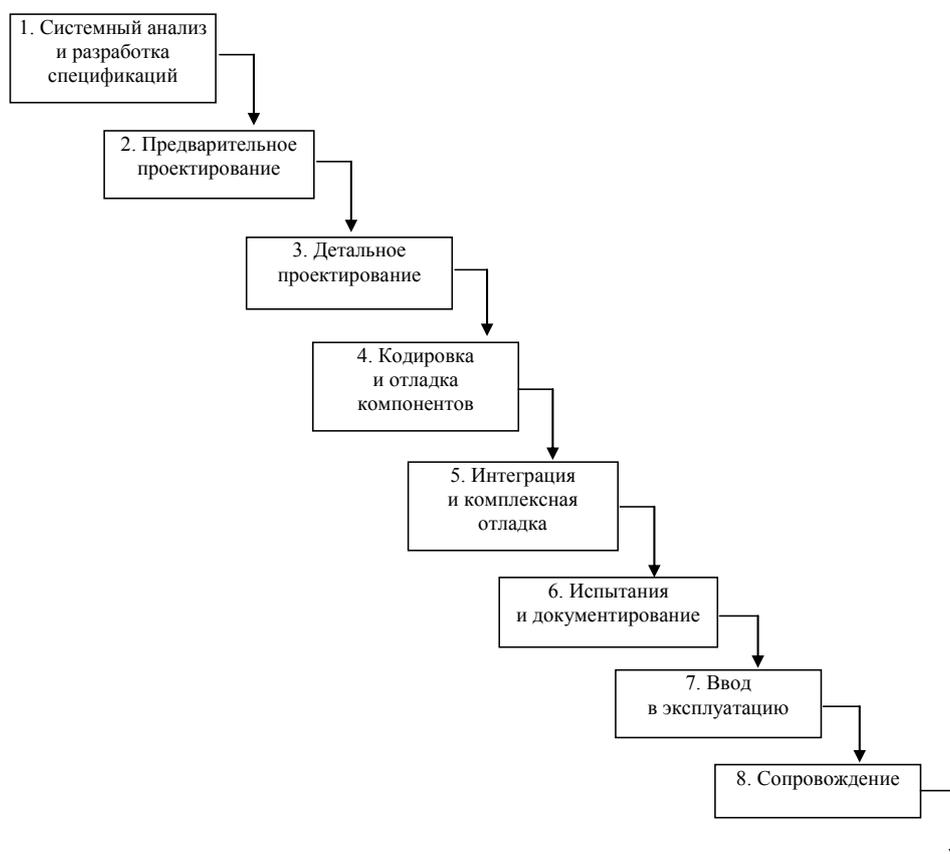


Рис. 8.1. Этапы жизненного цикла

Наиболее трудоемким и обязательным является тестирование уже созданной программы. Поэтому основное внимание уделим вопросам тестирования собственно программ.

Понятие и виды тестирования программ

Научно-техническая и учебная литература по проектированию программ содержит самые разнообразные трактовки понятия «тестирование». Мы рассмотрим это понятие с точки зрения теории технической диагностики, в рамках которой разработана довольно стройная и общая теория проверки, а также диагностики (определения места неисправностей) различных объектов. Тесты и тестирование широко используются в технике для установления соответствия изде-

лий заданным правилам построения, техническим условиям или заданиям на них, а также для определения достигнутых показателей качества.

Использование теории технической диагностики является вполне обоснованным: большинство терминов, используемых при обсуждении задач тестирования программных продуктов «пришли» из технической диагностики, а наибольшие достижения эта наука получила применительно к вычислительной технике. К тому же авторы достаточно большой части публикаций в области тестирования программных продуктов существенно используют результаты технической диагностики.

Не следует путать термин «тестирование» с «отладкой». В последнем случае обычно речь идет об интерактивном (с участием человека) процессе доработки кода программы. При этом тестирование является частью процесса отладки.

Тестирование программ имеет целью проверить и гарантировать правильную (безошибочную в определенном смысле) работу программы, а при неправильной работе определить место возможного дефекта (причину ошибки в работе).

В процессе тестирования в программу поступают некоторые данные и воздействуют некоторые управляющие команды, а результаты тестирования сравниваются с эталонными (правильными) результатами. Несовпадение полученных при тестировании результатов с эталонными говорит о наличии ошибки в работе тестируемой программы.

Используемую для тестирования совокупность управляющих команд, а также входных и эталонных выходных данных называют **тестом**.

В некоторых публикациях **тест определяется** как «**совокупность входных данных для программы, а также точное описание всех результатов, которые должна выработать программа на этих данных**».

Тест можно представить в виде последовательности **элементарных проверок**, каждая из которых характеризуется некоторым набором управляющих команд и входных данных (**тестовое воздействие**) и правильных выходных данных (**эталон**).

В теории технической диагностики по целям выделяют три вида проверок:
проверка исправности;
проверка работоспособности,
проверка правильности функционирования.

Применительно к программам можно говорить, что в ходе **проверки исправности** ставится цель проверить, соответствует ли программа всем требованиям нормативно-технической документации.

При проверке работоспособности надо определить выполняет ли программа во всех режимах работы все функции, для которых она предназначена.

Проверка работоспособности является более «слабой», чем проверка исправности. Работоспособная программа может быть неисправной. Например, в документации, прилагаемой к программе, могут отсутствовать некоторые эксплуатационные документы или в них не были внесены некоторые исправления. Программист мог ошибочно запрограммировать лишнее число циклов вычислений, но при этом конечный результат оказывается правильным, хотя, возможно, увеличилось время решения. Это нарушение нормативно-технических требова-

ний, т.е. такие программы должны быть отнесены к разряду неисправных. Однако если при запуске на ЭВМ программа выполняет вычисления с необходимой точностью и в приемлемое для пользователя время, то она будет считаться работоспособной.

Наконец, **проверка правильности функционирования** имеет целью проверить, выполняет ли программа правильно свои функции в ходе работы в данном конкретном режиме с конкретными данными в конкретный момент времени. Заметим, что в каждой ЭВМ есть специальные аппаратные узлы, которые контролируют правильность работы программы и соответствующих аппаратных узлов в настоящий момент (нет ли переполнения ячейки при сложении чисел, не произошло ли ошибки при считывании данных с гибкого диска и т.п.).

Будем называть **отказом** событие, заключающееся в нарушении нормального функционирования программы.

Кроме задач проверки в технической диагностики, выделяют задачу диагностики или поиска неисправности. Применительно к программам эта задача предполагает определение конкретных причин отказов при ее выполнении или конкретного несоответствия нормативно-техническим требованиям.

Некоторые рекомендации и стандарты выделяют тестирование в отдельный этап жизненного цикла программы, другие считают, что это естественное приложение к каждому этапу, так как тестирование проводится практически на всех этапах разработки.

Разработчики программ зачастую выделяют альфа- и бета-тестирование.

Альфа-тестирование – тестирование готового программного продукта на специально созданных задачах. Оно проводится у заказчика при участии разработчика.

Бета-тестирование – опробование программного продукта, бесплатно предоставленного потенциальным пользователям, на реальных задачах. Проводится без прямого участия разработчика. О найденных ошибках и замечаниях пользователь сообщает разработчику.

Информация об обнаруженных при альфа- и бета-тестировании отказах используется далее разработчиком для исправления программ.

Во всех случаях должны быть известны и, желательно, формализованы эталонные правила или значения результатов, которым должен соответствовать тестируемый объект. Результаты тестирования должны сравниваться с требованиями технического задания, спецификаций или иных эталонов для определения степени соответствия требованиям, полученным разработчиком от заказчика.

Как правило, тестирование и подготовка данных для его проведения, происходят при ограниченных вычислительных и временных ресурсах. Это определяет целесообразность тщательного планирования и документирования вопросов тестирования.

Существует довольно большое число рекомендаций, в том числе в виде стандартов, в которых рассматриваются вопросы тестирования программ. В них даются не только общие рекомендации по организации и документации тестирования, но и конкретизируется возможный состав и содержание документации.

8.2. Программа как объект тестирования

Дефекты и неисправности в программах

Наряду с понятием теста, базовыми понятиями в теории технической диагностики являются понятия **дефекта** и **неисправности**. В классической теории технической диагностики под дефектом понимают физическую первопричину, являющуюся причиной неправильной работы элемента или устройства в целом (пробой полупроводникового перехода, выгорание токопроводящей дорожки и т.п.), а под неисправностью – некоторую модель влияния дефектов на функционирование элементов объекта тестирования. Модель неисправности зависит от выбранной для решения задач тестирования модели самого объекта тестирования и его элементов.

Для программы дефектом можно считать неправильную запись операнда, ссылку на не ту метку, неверно заданный формат переменной, неверно заданный переход в алгоритме обработки и т.п.

Одной и той же неисправности может соответствовать несколько разных дефектов. Например, будем рассматривать в качестве «элемента» модели программы строку с записью оператора. Тогда неисправность типа «переполнение в ячейке» при выполнении некоторого оператора может происходить из-за задания неправильного формата одного из операндов либо из-за неправильной записи операции в самом операторе.

Для программ в качестве эквивалента понятия «неисправность» обычно используется термин **«ошибка»**. Однако в случае программ понятие «ошибка» имеет более широкую трактовку. Например, ошибкой программы может быть неправильная последовательность команд или указание неправильного перехода, неправильное использование типов переменных в тексте (коде) программы. С другой стороны, программист может абсолютно правильно запрограммировать заданный алгоритм обработки данных, однако не учесть при этом некоторые нюансы взаимодействия с операционной системой или другими программами. В результате при запуске программы будут возникать отказы.

Общеизвестно, что сложное техническое устройство или программа почти всегда после изготовления имеют дефекты. Они могут выявляться и появляться и в период эксплуатации. Поэтому в ходе разработки, перед передачей в эксплуатацию и в ходе нее изделие проходит многократное тестирование с целью определения и устранения ошибок и соответствующих дефектов.

Опытные программисты говорят, что при большом числе операторов нет исправных программ. В процессе отладки сложных программных средств практически невозможно обнаружить и ликвидировать все ошибки. В результате в программах остается некоторое количество скрытых ошибок. Они могут вызвать неверное функционирование программ при определенных сочетаниях входных данных и/или режимов работы. Поэтому пользователи сталкиваются с тем, что периодически в процессе эксплуатации выявляются все новые и новые дефекты. В качестве реального примера можно привести операционную систему Windows. Хотя ее разработкой занимаются тысячи высокопрофессиональных

программистов, применяются самые передовые технологии разработки, однако постоянно появляются сообщения о новых найденных в системе дефектах.

Особенности тестирования и проявления дефектов

Если рассматривать программу как независимый объект тестирования, то количество N дефектов в ней во времени t можно представить графиком, представленным на рис. 8.2.

На рисунке 8.2 видно, что на первых этапах жизненного цикла программы выявляется большинство дефектов (обычно это происходит до этапа ввода в эксплуатацию). На этапе эксплуатации периодически также выявляются и устраняются дефекты, но уже реже и в меньшем количестве. Таким образом, со временем количество ошибок программы постепенно снижается от некоторого начального значения N и стремится к нулю. Возрастание количества ошибок возможно, только если производилась корректировка программного кода (пик на рис. 8.2 соответствует внесению новых дефектов в момент t_k корректировки программы).

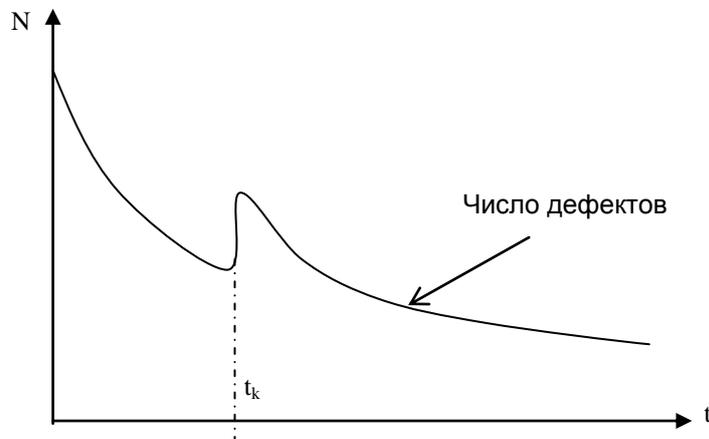


Рис. 8.2. Изменение количества дефектов в программе с течением времени

В технических изделиях число дефектов во времени иллюстрирует график на рис. 8.3. На нем видно, что на первых этапах жизненного цикла изделия большая часть дефектов устраняется, и частота их появления стабилизируется на некотором низком уровне. Однако с течением времени (обычно из-за старения и изменения характеристик составляющих его аппаратных элементов) число дефектов вновь возрастает.

Здесь мы подходим к одному из фундаментальных отличий задач тестирования технических и программных объектов. Каждая новая копия технического устройства может иметь свои дефекты, причем их число со временем может даже возрасть. Все же программные копии, как правило, имеют одни и те же

дефекты, а с течением времени их число уменьшается (если, конечно, не производится корректировка или изменение условий применения программы).



Рис. 8.3. Изменение количества дефектов в техническом устройстве с течением времени

Команды программ выполняются с помощью некоторых аппаратных (электронных) узлов ЭВМ, которые, в свою очередь, могут иметь дефекты. Поэтому при нарушении правильного функционирования аппаратной части ЭВМ зачастую трудно понять, что явилось причиной отказа – программа или аппаратура.

Программы имеют также ряд других особенностей, которые отличают процесс их тестирования от традиционного, применяемого для проверки технических изделий. К таким особенностям относят:

- практическая невозможность (в большинстве случаев) перечислить возможные ошибки и, как следствие, формально получить тесты, гарантирующие исчерпывающую проверку программы;
- для ряда сложных программ практически невозможно определить эталонные значения для всех изменений в процессе выполнения программы;
- относительно невысокая степень формализации критериев качества тестов и процесса тестирования.

При разработке тестов для проверки технических устройств обычно строят модель устройства (математическую или физическую) и далее используют ее для подбора тестов либо получения эталонных значений. Для программ хорошие модели трудно построить, поэтому рассматриваются упрощенные модели, а в качестве эталонов для тестирования часто используются косвенные данные, которые не отражают результаты всех промежуточных вычислений.

Наличие в программе дефекта можно обнаружить, если происходит отклонение в выполнении программой заданных функций, т.е. **отказ**. В зависимости от степени серьезности последствий отказов в программе эти отклонения можно классифицировать следующим образом:

- полное прекращение выполнения функций на длительное или неопределенное время;
- кратковременное нарушение хода вычислительного процесса.

Наиболее **типичными симптомами проявления ошибок** при выполнении программы являются:

- отклонение выходных данных программы от эталонных значений;
- преждевременное (аварийное) завершение выполнения программы;
- недопустимое увеличение времени выполнения программы вплоть до полной остановки ее выполнения;
- заикливание на выполнении некоторой последовательности команд;
- нарушение последовательности вызова отдельных программ, в результате чего происходит пропуск необходимых программ либо непредусмотренное обращение к программам.

Аварийное завершение прикладных программ, как правило, легко идентифицируется, так как операционная система тогда выдает сообщение, содержащее соответствующий аварийный код. Типичными причинами появления кодов аварийного завершения являются сбои в работе аппаратной части ЭВМ или дефекты программы, приводящие к обращению к отсутствующим или запрещенным для доступа ячейкам памяти, переполнению в ячейке при выполнении операции, и др.

На практике все больше задач, решение которых предполагает создание интеллектуальных информационных систем. В них алгоритмы работы и начальные значения переменных могут меняться в каждом конкретном применении. Это существенно усложняет задачи тестирования.

8.3. Организационно-методические аспекты тестирования

Процесс тестирования

В научно-технической литературе процесс выявления конкретного места дефекта в программе и его устранения часто называют **отладкой**.

Тестирование есть часть процесса устранения дефектов (отладки). Результаты тестирования используются для обнаружения и локализации ошибок, после чего делаются исправления в коде программы (или в исходных данных) и вновь проводится тестирование, пока функционирование программы не будет происходить в соответствии с установленными требованиями.

В организационно-методическом плане этот процесс можно представить из следующих этапов:

- анализ объекта тестирования;
- выбор методов тестирования;
- выбор контролируемых параметров, разработка тестов и соответствующих эталонов;
- оценка полноты тестирования выбранным методом на разработанных тестах, доработка теста (при низкой полноте);
- реализация процесса тестирования и получение результатов функционирования объекта на подготовленных тестах;
- сравнение результатов тестирования с эталонами;
- обнаружение отклонений от эталонных значений, принятие решений о проведении дополнительного тестирования и разработки дополнительных тестов с целью определения причин ошибок;

– документирование результатов тестирования и корректировка кода программы.

После корректировки программы процесс повторяется, пока результаты прогона программы не будут соответствовать заданным эталонам.

Для созданных программ обычно выделяют три вида тестирования: модульное тестирование (выявление и устранение ошибок в программных модулях), интегральное тестирование (выявление и устранение ошибок, которые проявляются при взаимодействии модулей), приемо-сдаточное тестирование (комплексная проверка при сдаче заказчику). В организационно-методическом плане это, с одной стороны, независимые виды испытаний создаваемого программного продукта. С другой – они имеют при реализации ряд общих черт, причем выполняются последовательно, могут использовать одни и те же методы, а при разработке тестов для каждого вида тестирования учитываются вопросы совместного функционирования модулей и требования заказчика.

Обобщенная схема, иллюстрирующая организацию процесса обнаружения и устранения причин ошибок, показана на рис. 8.4.

Методы тестирования

Схема на рис. 8.4 отображает логику выявления и устранения ошибок, когда последовательно применяется несколько методов тестирования (статическое, детерминированное, стохастическое, в реальном времени)

В ходе тестирования результаты выполнения теста сравниваются с эталонами и определяется часть программы, содержащая дефект и подлежащая в дальнейшем корректировке. Очевидно, чем точнее и быстрее определяется причина ошибки и место дефекта, тем эффективнее используемый тест и метод тестирования. После выявления дефекта проводится корректировка программы с целью его устранения, после чего проводится повторное тестирование тем же или иным методом. При этом предварительно может меняться или корректироваться тест.

Статическое тестирование проводится без исполнения объектного кода программы путем формального анализа текста программы, написанного на языке программирования. Операторы и операнды текста программ при этом анализируются в символьном виде, поэтому такой метод иногда называют также символьным тестированием. В качестве эталонов применяются правила структурного построения программных модулей на используемом языке программирования, а также правила записи операторов и операндов языка.

Детерминированное тестирование предполагает предварительную подготовку тестов и стратегии тестирования, гарантирующих некоторый результат, например, выявление и определение дефектов заданного класса. Возможных дефектов очень много, поэтому такое тестирование используется для небольших программ (модулей) либо после других методов, когда уже ограничена область нахождения дефекта.

Стохастическое тестирование – тестирование, при котором исходные тестовые данные берутся случайным образом. Стохастическое тестирование применяется в основном для обнаружения ошибок, а для локализации их причин приходится переходить к детерминированному тестированию.

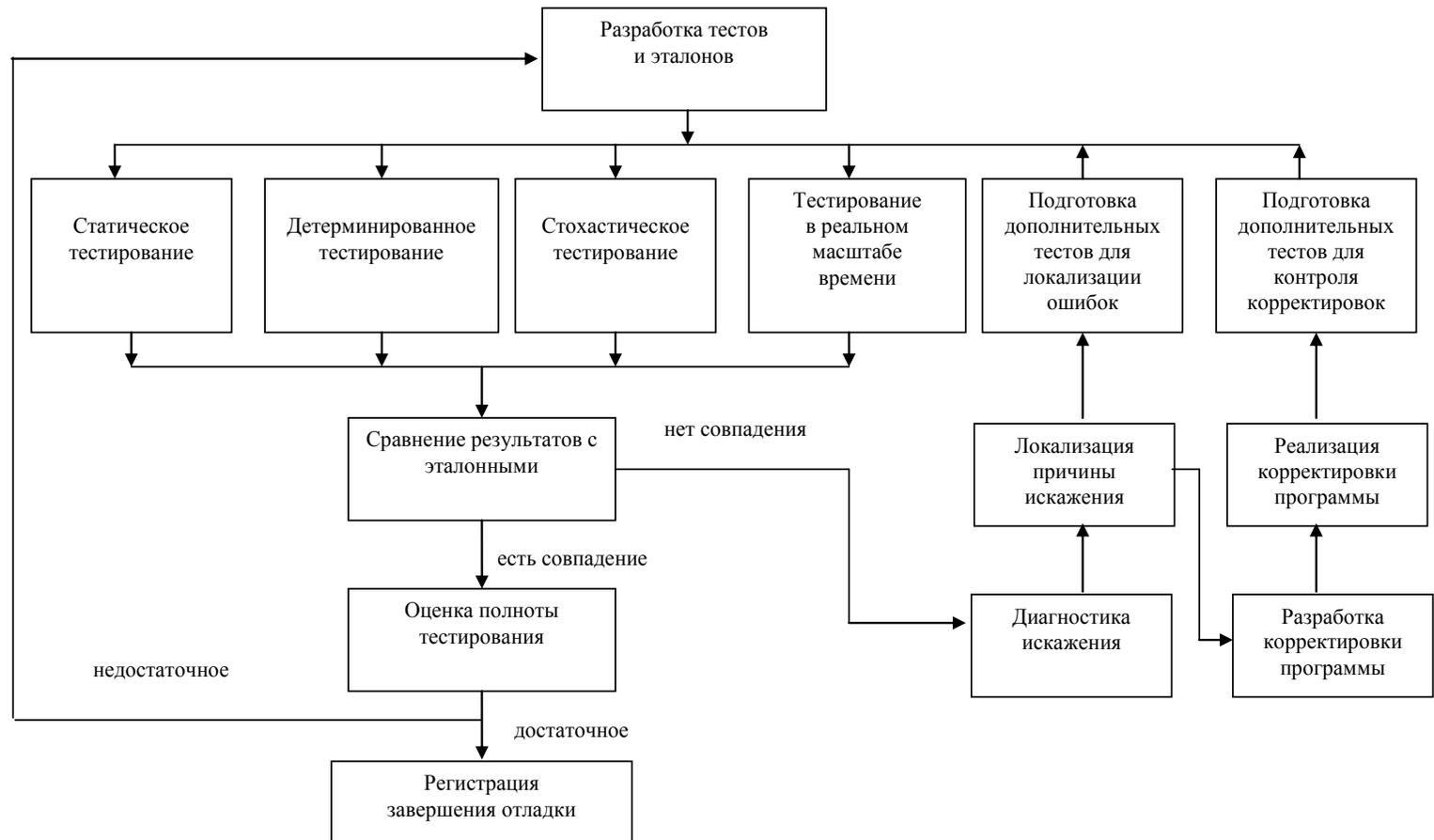


Рис. 8.4. Организация процессов обнаружения и устранения ошибок

При тестировании в реальном масштабе времени проверяется исполнение программы и обработка исходных данных с учетом времени их поступления, длительности и приоритетности обработки, динамики использования памяти и взаимодействия с другими программами и т.д. При обнаружении отказов для локализации дефекта часто приходится переходить к детерминированному тестированию, причем в качестве эталонных могут применяться временные характеристики работы исправной программы.

Современная теория программирования рекомендует проектировать сложные информационные системы как комплекс взаимодействующих программных модулей. В программных системах с такой архитектурой для выявления и устранения ошибок могут использоваться технологии восходящего или нисходящего тестирования.

При **восходящем** тестировании прежде всего проверяются модули нижних иерархических уровней, к которым постепенно подключаются вызывающие их модули. Основные трудности состоят в необходимости полного обновления тестовых наборов при подключении каждой новой программы более высокого уровня.

При **нисходящем** тестировании проверки начинаются с модулей управления и организации вычислительного процесса, размещенных на высших иерархических уровнях. К ним постепенно подключаются для тестирования программы последующих более низких иерархических уровней. Если программы нижних уровней не разработаны или не протестированы, то вместо них включаются программные имитаторы – «заглушки». В результате при тестировании осуществляется проверка некоторой группы модулей с имитаторами на нижних иерархических уровнях. Преимуществом такого метода является возможность сохранения и развития наборов тестовых исходных данных по мере подключения программ нижних уровней. С самого начала тестирования могут использоваться исходные данные, соответствующие реальному функционированию программного продукта, с последовательно снимаемыми ограничениями. При этом уже на ранних этапах проверяется межмодульный интерфейс. Однако тестирование групп программ с заглушками может требовать повышенных затрат на обнаружение ошибок в модулях нижних иерархических уровней по мере их подключения, если они до этого не тестировались.

На практике оба метода используются совместно с учетом сложности тестируемых программ и реальных планов разработки.

Для отладки могут использоваться специальные программные или программно-аппаратные средства.

Средства отладки программ должны:

- а) управлять исполнением программ (запускать, останавливать, изменять порядок выполнения команд и т.д.);
- б) собирать информацию о ходе выполнения программы;
- в) отображать в удобной для наладчика форме состояния указанных переменных и массивов данных в заданные моменты времени или при выполнении определенных условий;
- в) показывать трассы (т.е. последовательность выполняемых программой команд и состояния регистров после выполнения каждой команды);

г) имитировать работу отсутствующих аппаратных средств микропроцессорной системы и взаимодействие с другими программами.

Сегодня практически любой прибор (холодильник, видеокамера, автомобиль, телевизор, навигационная система, система контроля работы дизеля и т.п.) имеет в своем составе специализированный микроконтроллер, содержащий микропроцессор. Один и тот же микроконтроллер может использоваться для исполнения разных функций в зависимости от введенной в его память программы. Для отладки программ, вводимых в микроконтроллер, создаются специальные средства отладки. Некоторые из таких отладочных средств рассматривались в пп. 6.7 настоящего учебного пособия.

8.4. Построение тестов

Особенности построения тестов при различных моделях программ

Тест почти всегда можно представить в виде последовательности элементарных проверок. В самом общем виде процесс проектирования теста для конкретной программы (системы программ) представляется как последовательность шагов. На очередном шаге выбирается очередной кандидат в элементарную проверку, который после оценки его эффективности (соответствия заданным требованиям) включается в тест или исключается из кандидатов в тест. В качестве критерия оценки очередного кандидата может быть, например, число дополнительно обнаруживаемых ошибок с помощью данной проверки. Говорят, что ошибка обнаруживается (выявляется, проверяется) на данной проверке, если при наличии в программе ошибки измеряемый результат функционирования программы будет отличаться от эталона.

В идеале тестирование желательно провести на всех вариантах входных данных во всех режимах работы программы. Такой тест называют исчерпывающим. Однако в силу большого числа вариантов это зачастую невозможно. Поэтому возникает задача построения (проектирования) ограниченного числа проверок, обеспечивающих необходимый уровень тестирования.

Формализация процессов проектирования теста предполагает задание в явном или не явном виде модели объекта тестирования и списка (или видов) возможных ошибок.

Модели собственно программ можно разбить на три большие группы: структурные («белый ящик»), функциональные («черный ящик») и структурно-функциональные («серый ящик»).

Структурная модель содержит подробную информацию о внутренней структуре программы. Наиболее часто она представляет собой описание алгоритма программы в виде блок-схемы (рис. 8.5) или графа. В качестве возможных ошибок при этом обычно рассматривают искажения данных при их перемещении по отдельным путям блок-схемы или графа, либо ошибки выполнения каждой операции (команды), сопоставленной элементу блок-схемы или узлу графа. Тогда задачу построения теста можно сформулировать как задачу подбора последовательности проверок, позволяющих обнаружить любую из возможных ошибок или даже различить их. Заметим, что в теории технической диагностики

тест, позволяющий обнаружить максимальное число возможных неисправностей, называют контролирующим, а различающий максимальное число пар неисправностей – диагностическим.

Рассмотрим построение теста для случая блок-схемы на рис. 8.5, описывающей алгоритм вычисления суммы R элементов матрицы размером $n \times p$. Значение R будем считать выходной (измеряемой) величиной.

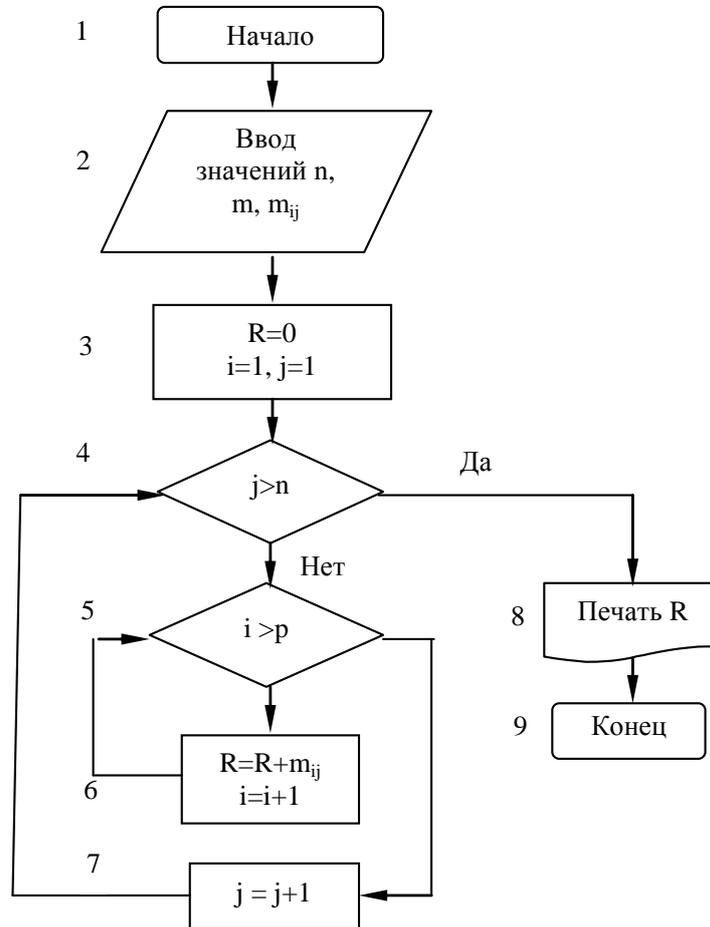


Рис. 8.5. Структурная модель, представленная в виде блок-схемы

После ввода значений n и p , а также элементов m_{ij} матрицы производится процедура инициализации, в ходе которой устанавливаются начальные значения переменных и индексов циклов (R , i , j). Внешний цикл (индекс j) обеспечивает последовательный выбор строк матрицы. Внутренний цикл (индекс i) позволяет организовать суммирование элементов очередной строки. Результат суммирования накапливается в переменной R . Условие $j > n$ является условием завершения алгоритма.

Пусть необходимо построить тест, проверяющий прохождение данных по каждому пути от блока 1 («Начало») до блока 9 «Конец»). Сначала определим все возможные пути. Это пути через блоки:

путь 1 – 1,2,3,4,5,6,5,7,4,8,9;

путь 2 – 1,2,3,4,5,7,4,8,9;

путь 3 – 1,2,3,4,8,9.

Будем формировать тест, подбирая проверки для каждого из трех полученных путей. Чтобы проверить прохождение данных по первому пути, надо задать $n=r=1$ и m_{11} . При этих условиях матрица состоит из одного элемента и эталонного значения (правильный результат проверки) $R= m_{11}$. Таким образом, первую элементарную проверку, включаемую в искомый тест, составляет набор значений $\pi_1=\{n=r=1, R= m_{11}\}$. Для проверки пути 2 в тест надо включить элементарную проверку $\pi_2=\{n=1, r=0, R=0\}$; пути 3 – $\pi_3=\{n=0, R=0\}$ (в проверке π_2 результат R не зависит от значения m_{ij} , в проверке π_3 – от значений r и m_{ij} , т.е. их можно задать любыми). Таким образом, результирующий тест будет состоять из трех элементарных проверок π_1, π_2, π_3 .

Для рассматриваемого примера (рис. 8.5) можно выбрать другую модель ошибки. Например, указать список возможных ошибок для каждого отдельного элемента блок-схемы и далее строить тест в предположении, что список возможных ошибок объекта тестирования есть сумма списков ошибок, сформированных для каждого элемента. Так, для блока 3 (рис. 8.5) в качестве возможных можно определить ошибки вида: $R \neq 0, i \neq 1, j \neq 1$; для блока 4: $j < n, j = n, n < 0, n \neq n^*$ (n^* – значение n при выполнении операции блока 2) и т.д. Чтобы построить тест надо будет подобрать элементарные проверки, позволяющие обнаружить каждую ошибку из созданного списка (если она случится).

Достаточно очевидно, что состав получаемого теста, его возможности по выявлению дефектов существенно зависят от выбранных моделей объекта и ошибок. Поэтому выбор этих моделей – очень важный элемент, предшествующий разработке теста и процессов тестирования.

Элементарная проверка, подобранная для выявления некоторой ошибки, может обнаруживать наличие и ряда других ошибок. Поэтому практические системы построения тестов предполагают анализ каждой выбранной проверки, с целью оценки, какие еще ошибки она может выявлять. На следующем шаге выбирается проверка для ошибок, не обнаруживаемых на ранее выбранных проверках. Возможно, что для обнаружения ошибки недостаточно одной элементарной проверки. Существование таких ошибок серьезно усложняет процесс построения теста.

Данные о вероятности или частоте исполнения маршрутов или операторов помогают выделить интенсивно используемые компоненты программ, которые целесообразно подвергать наиболее тщательной проверке. Априорно на базе представлений разработчика о динамике функционирования создаваемой программы может быть составлен перечень сведений о частоте и длительности выполнения каждого оператора, вероятность выполнения каждого логического условия, предельные значения некоторых переменных, количество итераций в циклах и т.д. Эти данные для каждой программы могут получаться автоматиче-

ски в процессе ее исполнения или обобщаться, уточняться по мере тестирования программ.

Процессы проектирования тестов на основе структурной модели наиболее формализованы.

Функциональные модели объектов тестирования предполагают, что внутренняя структура программы неизвестна. Имеется только функциональное описание, которое позволяет для заданных входных данных и управляющих команд определить правильные выходные результаты работы тестируемой программы. Такое описание может представлять собой математическую функцию или таблицу, задающую связь между значениями входных данных и результатом вычислений, осуществляемых программой; описание требований к алгоритму программы в техническом задании или в форме отдельной спецификации и т.д.

Так, для рассматриваемого выше примера в качестве функциональной модели можно взять математическое выражение $R = \sum_{j=1}^n \sum_{i=1}^p m_{ij}$. Прежде чем переходить к построению теста, надо определить модель ошибки. В данном случае естественно считать ошибкой отклонение переменных функции R от их правильных значений. Например, можно считать, что вследствие дефектов программы каждая переменная может находиться в одном из трех состояний: «постоянное значение», «больше правильного значения», «меньше правильного значения».

Теперь можно начать формировать тест, подбирая для каждой возможной ошибки элементарную проверку, при выполнении которой гарантируется обнаружение ошибки.

Как и в случае структурной модели, можно рассматривать и другие модели ошибок.

Структурно-функциональная модель занимает промежуточное положение между структурной и функциональной. Объект представляют в виде функционально законченных частей (модулей, подпрограмм) с указанием связей между ними.

При использовании структурно-функциональной модели в определенной мере сглаживаются недостатки структурной и функциональной моделей. Так, по функциональной модели место дефекта можно указать очень грубо. Чем подробнее описывает модель объект тестирования, тем потенциально более «хорошие» тесты можно построить. Поэтому при использовании структурной модели можно строить тесты, обеспечивающие наибольшую полноту проверки и наиболее точно указывающие место дефекта, но она очень громоздка. Структурно-функциональная модель описывает программу в виде совокупности некоторых частей-модулей. Применение структурно-функциональной модели потенциально позволяет указывать место дефекта с точностью до модуля. При этом в процессе отладки можно организовать двухэтапное тестирование: сначала на тесте, который позволяет выделить неправильно работающий модуль, а затем, используя тест, ориентированный на более тщательное тестирование конкретного модуля.

Идеология построения теста по тестовым случаям

Многие современные программы в силу разных причин (алгоритмическая сложность, применение высокоуровневых языков программирования или специальных инструментальных средств и др.) не имеют хороших описаний. Поэтому сегодня широко применяется идеология построения тестов по **тестовым случаям**. По этой идеологии сначала определяют наиболее важные и часто используемые функции, возможные режимы и критические (предельные) параметры работы программы (фактически при этом в неявном виде формулируется класс возможных ошибок). Затем подбираются проверки для этих тестовых случаев. С точки зрения типа модели объекта тестирования здесь используется функциональная модель.

Хотя каждая программа уникальна, но удалось выделить ряд целей тестирования по тестовым случаям, которые позволяют говорить об этом подходе, как об одном из способов построения тестов, пригодных для широкого применения.

При подборе тестовых проверок по тестовым случаям обычно выделяют:

тестирование функций – проверка правильности выполнения основных функций программы;

стрессовое тестирование – проверки исполнения программ в нештатных ситуациях, при граничных значениях входных данных, в критических ситуациях по условиям и логике решения задач;

тестирование корректности использования ресурсов – проверки качества и безопасности исполнения программ при ограниченности ресурсов памяти и производительности вычислительной системы;

тестирование параллельного исполнения программ – проверки согласованности использования исходных и промежуточных данных общей памяти, а также процессоров при параллельном исполнении нескольких программ;

тестирование эффективности защиты от искажений исходных данных (случайных или преднамеренных);

тестирование эффективности защиты от сбоев – проверка реакции программного обеспечения при сбоях в работе аппаратуры и/или невыявленных дефектах программ;

тестирование диалоговых средств взаимодействия с пользователем;

тестирование конфигураций – проверка работоспособности при различных конфигурациях оборудования и программной системы, и др.

Проектирование тестов по тестовым случаям трудно формализовать. Однако в научно-технической литературе можно найти многочисленные практические рекомендации по принципам выбора проверок для различных тестовых случаев. Так, при стрессовом тестировании предварительно следует определить, когда и при каких условиях возникают «стрессовые» ситуации в работе программы. Для этого разработчик теста должен сформулировать для себя ряд вопросов, чтобы определить требования к элементарным проверкам. Например:

– В каких режимах работы требуется максимальный объем вычислительных ресурсов?

– Можно ли и как перегрузить систему?

– Что случится, когда другие процессы потребуют совместного использования одного и того же ресурса?

- Какие существуют ограничения на обработку данных и на ресурсы?
- Можно ли одновременно запускать несколько процессов?
- Что будет, если ввести данные, со значениями вне допустимого диапазона?

Для создания «напряжений» (стрессовых ситуаций) в работе тестируемой программы можно:

- уменьшить доступную область памяти или дискового пространства,
- запустить параллельно несколько экземпляров приложения,
- создать значительное количество прерываний,
- очень быстро нажимать клавиши,
- превысить граничные значения и параметры,
- сгенерировать множество асинхронных управляемых событиями процессов,
- выполнить большое число вводов или повторений определенных действий,
- отправить большой сложный запрос в систему баз данных,
- запустить приложение, когда система выполняет резервирование, поиск данных или копирование, и др.

Контрольные вопросы

1. Что понимается под жизненным циклом программы, и из каких этапов он состоит?
2. Цель тестирования программы.
3. Дайте определение теста и отказа для программы.
4. В чем различие альфа- и бета-тестирования?
5. Нарисуйте графики изменения во времени частоты выявления дефектов в аппаратуре и в программах.
6. Перечислите типичные симптомы появления ошибки в работе программы.
7. Перечислите основные этапы тестирования.
8. Виды тестирования программ. Дайте характеристику этих видов.
9. В чем отличие статического тестирования от других видов тестирования?
10. Дайте характеристики технологий нисходящего и восходящего тестирования.
11. Что означает «тестирование по тестовым случаям»?
12. Какие виды проверок выделяют при тестировании по тестовым случаям?

Рекомендуемая литература

Калбертстон, Р. Быстрое тестирование / Р. Калбертстон, К. Браун, Г. Кобб; пер. с англ. – М.: Изд. дом. «Вильямс», 2002. – 384 с.

Липаев, В.В. Отладка сложных программ / В.В. Липаев. – М.: Энергоатомиздат, 1993.

Орлов, С.А. Технологии разработки программного обеспечения: учебник для вузов. 3-е изд. / С.А. Орлов. – СПб.: Питер, 2004. – 527 с.

Тампре, Л. Тестирование программного обеспечения / Л. Тампре; пер. с англ. . – М.: Изд. дом. «Вильямс», 2003. – 368 с.

Фатрепп, Р.Т. Управление программными проектами: достижение оптимального качества при минимуме затрат / Р.Т. Фатрепп, Д.Ф. Шафер, Л.И. Шафер; пер. с англ. – М.: Изд. дом «Вильямс», 2003. – 1136 с.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
Тема 1. ОСНОВНЫЕ ПОНЯТИЯ И ЗАДАЧИ ТЕХНИЧЕСКОЙ ДИАГНОСТИКИ	7
1.1. Технические состояния, надежность и задачи диагностирования	7
1.2. Средства и системы диагностирования	13
1.3. Использование различных видов систем диагностирования в процессах производства электронных изделий	17
Тема 2. МОДЕЛИ ОБЪЕКТОВ И НЕИСПРАВНОСТЕЙ	21
2.1. Модели исправных объектов	21
2.2. Модели неисправных технических состояний	43
2.3. Таблица функций неисправностей и таблица неисправностей	60
2.4. Проверимость и различимость неисправностей	67
Тема 3. КРИТЕРИИ ОЦЕНКИ ТЕСТОВ И СИСТЕМ ДИАГНОСТИРОВАНИЯ	75
3.1. Виды критериев оценки	75
3.2. Оценка полноты контроля	78
3.3. Оценка глубины поиска неисправностей	83
3.4. Достоверность контроля	88
3.5. Анализ на состязания тестов цифровых схем	90
Тема 4. МЕТОДЫ ПОСТРОЕНИЯ ТЕСТОВ ЦИФРОВЫХ УСТРОЙСТВ	100
4.1. Понятия контролирующего и диагностического тестов	100
4.2. Получение тупиковых тестов с использованием таблиц неисправностей	102
4.3. Метод активизации одномерного пути	107
4.4. d-алгоритм	109
4.5. Построение теста по методу булевой производной	113
4.6. Построение теста по методу эквивалентной нормальной формы	117
4.7. Использование модели конечного автомата для построения тестов	120
4.8. Построение тестов для микропроцессорных систем	123

Тема 5. ВНУТРИСХЕМНОЕ (ПОЭЛЕМЕНТНОЕ)	
ДИАГНОСТИРОВАНИЕ	128
5.1. Назначение и принципы внутрисхемного диагностирования	128
5.2. Принципы измерений параметров элементов с помощью вольтметров и амперметров	130
5.3. Особенности определения работоспособности различных элементов с помощью простейших приборов.....	135
5.4. Погрешности методов и средств измерений	149
5.5. Принципы и методы организации измерений параметров элементов в автоматизированных системах поэлементного диагностирования	157
5.6. Автоматизированные системы поэлементного диагностирования.....	165
Тема 6. ОРГАНИЗАЦИЯ ДИАГНОСТИРОВАНИЯ И ОБСЛУЖИВАНИЯ МИКРОПРОЦЕССОРНЫХ УСТРОЙСТВ.....	181
6.1. Особенности микропроцессора как объекта диагностирования	181
6.2. Общие принципы организации технического обслуживания ПЭВМ....	183
6.3. Логические анализаторы и пробники	199
6.4. Внутрисхемные эмуляторы.....	205
6.5. Сигнатурный анализ	209
6.6. Диагностические платы.....	218
6.7. Средства отладки микропроцессорных систем.....	227
Тема 7. ПРОЕКТИРОВАНИЕ КОНТРОЛЕПРИГОДНЫХ УСТРОЙСТВ	239
7.1. Понятие и критерии контролепригодности.....	239
7.2. Контроль и восстановление информации с использованием кодирования.....	246
7.3. Схемные решения повышения контролепригодности	258
7.4. Стандарты периферийного сканирования	272
Тема 8. ТЕСТИРОВАНИЕ ПРОГРАММ	282
8.1. Основные понятия и цели тестирования	282
8.2. Программа как объект тестирования	286
8.3. Организационно-методические аспекты тестирования	289
8.4. Построение тестов	293

Учебное издание

Мальшенко Юрий Вениаминович

Стыцюра Людмила Федоровна

Саяпин Владимир Иванович

ТЕХНИЧЕСКАЯ ДИАГНОСТИКА

Учебное пособие

Редактор С.Г. Масленникова

Компьютерная верстка М.А. Портновой

Лицензия на издательскую деятельность ИД № 03816 от 22.01.2001

Подписано в печать 22.06.10. Формат 70×100/16.
Бумага писчая. Печать офсетная. Усл. печ. л. 35,0.
Уч.-изд. л. 23,5. Тираж 600 экз. Заказ

Издательство Владивостокский государственный университет
экономики и сервиса
690600, Владивосток, ул. Гоголя, 41
Отпечатано: множительный участок ВГУЭС
690600, Владивосток, ул. Державина, 57