

МИКРОПРОЦЕССОРЫ

В 3 книгах

Книга 2

СРЕДСТВА СОПРЯЖЕНИЯ • КОНТРОЛИРУЮЩИЕ И ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

Под редакцией
члена-корреспондента АН СССР
Л. Н. ПРЕСНУХИНА

Допущено Министерством высшего
и среднего специального образования СССР
в качестве учебника
для студентов высших технических
учебных заведений

**МИНСК
«ВЫШЭЙШАЯ ШКОЛА»
1987**

ББК 32.973.2 я73

М59

УДК 681.325.5:621.382.049.77 (075.8)

Печатается по изданию:

Микропроцессоры. В 3 кн. Кн. 2. Средства сопряжения. Контролирующие и информационно-управляющие системы.— М.: Высш. шк., 1986.

Авторы: В. Д. Вернер, Н. В. Воробьев, А. В. Горячев, Г. И. Фролов, А. А. Шишкевич.

Рецензент — кафедра автоматизированных систем управления МВТУ им. Н. Э. Баумана (зав. кафедрой — проф. В. Н. Четвериков)

Микропроцессоры: В 3 кн. Кн. 2: Средства сопряжения. М 59 Контролирующие и информационно-управляющие системы: Учеб. для техн. вузов /В. Д. Вернер, Н. В. Воробьев, А. В. Горячев и др.; Под ред. Л. Н. Преснухина.— Мн.: Выш. шк., 1987.— 303 с.: ил.

Учебник является второй книгой из серии, выпускаемой под общим названием «Микропроцессоры». В нем рассмотрены вопросы схемотехники и организации связей элементов и узлов микроЭВМ; аппаратные средства сопряжения систем сбора и преобразования информации, вопросы теории и проектирования информационно-управляющих вычислительных систем (ИУВС) на базе микроЭВМ с изложением материалов математического обеспечения и организации вычислительного процесса специализированных ИУВС.

Впервые книга вышла в издательстве «Высшая школа» (Москва) в 1986 г.

М 2405000000—151 40—87
М304(03)—87

ББК 32.973.2я73

© Издательство «Высшая школа», 1986.

© Оформление. Издательство
«Высшая школа», 1987.

ВВЕДЕНИЕ

Научно-технический прогресс, определяющий мощный подъем общественного производства, в значительной степени обусловлен широким внедрением достижений электроники во все отрасли народного хозяйства. Прогресс в области вычислительной техники и радиоэлектроники связан с достижениями микроэлектроники в создании интегральных схем малой, средней, большой и сверхбольшой степени интеграции. Появление микропроцессорных БИС позволило из-за их дешевизны, малых габаритов, массы, мощности потребления и свойства программируемости функций решить проблему разработки малого числа БИС для большого числа применений, внедрить вычислительную технику в те области, в которых ранее она не использовалась.

Необходимость повышения технического уровня вычислительной техники, приборов и средств автоматизации, связи, робототехники на основе новейших достижений микроэлектроники специально отмечалась в директивных документах партии и правительства.

Промышленностью освоены и выпускаются много типов микропроцессоров, благодаря которым обеспечены исключительные преимущества цифровым методам обработки информации. Однако существуют аналоговые сигналы, которые надо принимать, обрабатывать, хранить и выдавать пользователю. Важное значение приобретает проблема связи аналоговых объектов с цифровыми управляющими машинами, в частности вопросы преобразований, нормализации сигналов, методы и средства передачи аналоговых сигналов по линиям связи при наличии помех и т. д. Применение микропроцессоров и микроЭВМ для сбора данных и управления производственными процессами вызывает ряд проблем аналого-цифрового преобразования сигналов, которые должен решать пользователь.

Для обработки аналоговых и цифровых сигналов разработана большая номенклатура микросхем, среди которых можно отметить генераторы, усилители, аналого-цифровые и цифроаналоговые преобразователи, модуляторы, компараторы, переключатели тока и напряжения, элементы выборки и хранения, фильтры, вторичные источники питания, центральные процессорные элементы, устройства управления вводом-выводом, программируемые параллельные и последовательные интерфейсы, контроллеры прямого доступа к памяти, магистральные приемопередатчики, блоки микропрограммного управления, приоритетного прерывания, арифметические расширители, запоминающие устройства, многофункциональные синхронизирующие устройства, программируемые таймеры и т. п.

Большинство перечисленных схем и устройств являются функциональными составными частями микропроцессорных комплектов, в значительной степени определяя архитектуру микроЭВМ. Однако практически любая микроЭВМ, кроме основных функциональных БИС, содержит и значительное число микросхем малой и средней степени интеграции (табл. В.1), особенно в периферийном оборудовании, датчиках первичной информации, устройствах встроенного контроля и диагностики, устройствах отображения информации и т. д.

Учебник в основном будет посвящен возможностям использования схем средней степени интеграции, большинство из которых можно рассматривать как многофункциональные узлы (см. уровень межрегистровых передач в табл. В.1). Излагаемый материал подкрепляется примерами применения отечественных микросхем для обработки аналоговых и цифровых сигналов, в ряде случаев дается расчет анализируемых схем.

Контроль сложных изделий — трудоемкий и длительный процесс, требующий усилий большого числа лиц для получения достоверной информации. Системы контроля на базе микроЭВМ обладают большими преимуществами в отношении стоимости контроля, уменьшения времени его проведения и обработки результатов, надежности, обеспечивают значительную гибкость, высокую компактность и модульную простоту, а также открывают широкие возможности для новых приложений, ранее требовавших использования дорогих специализированных ЭВМ.

В основу решения задачи автоматизированного контроля технического состояния объектов положены следующие принципы: максимальная степень автоматизации процесса контроля и сведение к минимуму числа ручных операций, повышение достоверности резуль-

Т а б л и ц а В. 1

Уровень иерархии	Составные части	
	Система	Элементы системы
Системный	Вычислительная система, ЭВМ	Устройства ЭВМ (АУ, УУ, ОЗУ, ПЗУ, УВВ, каналы связи, ...)
Уровень межрегистровых передач	Устройства ЭВМ	Узлы ЭВМ (сумматоры, счетчики, регистры, дешифраторы, АЦП, ЦАП, ...)
Уровень логических схем	Узлы ЭВМ	Цифровые и аналоговые элементы ЭВМ (И, ИЛИ, И — НЕ, ИЛИ — НЕ, И — ИЛИ — НЕ, ..., триггеры, элементы ЗУ, операционные усилители, ключи, ...)
Схемотехнический	Цифровые и аналоговые элементы ЭВМ	Электронные компоненты (транзисторы, диоды, резисторы, конденсаторы, ...)
Компонентный	Электронные компоненты	Материалы (магнитные материалы, металлы, полупроводники, диэлектрики, ...)

татов контроля, автоматическая выдача протокола результатов испытаний; высокая надежность системы, основанная на использовании встроенной микроЭВМ и модульной структуры; максимальная простота и доступность программного обеспечения, благодаря которым система контроля может обслуживаться лицами, не являющимися специалистами в области вычислительной техники.

В наибольшей степени этим требованиям отвечают одноплатные микроЭВМ «Электроника МС1201.01», обладающие высокой надежностью, обусловленной использованием интегральных микросхем высокой степени интеграции, широкими функциональными возможностями и развитым программным обеспечением, совместимым с рядом отечественных и зарубежных микро- и мини-ЭВМ (например, «Электроника 60», СМ-3, СМ-4, «Электроника 100-25», а также широко известное семейство мини- и микроЭВМ системы PDP-11).

Магистрально-модульный принцип, положенный в основу организации этого семейства ЭВМ и распространенный на аппаратные и программные средства, обеспечивает совместимость модулей на конструктивном, электрическом, функциональном и программном уровнях, однородность структуры, гибкость в выборе аппаратных и программных модулей, возможность расширения структуры, модернизацию модулей и постоянное совершенствование системы в целом. Системы с общей магистралью хорошо приспособлены для работы в АСУТП, в системах информационно-измерительных, для автоматизации научных исследований, автоматизированного контроля (АСК), когда требуются высокая производительность, гибкость и работа в реальном масштабе времени.

Стандартизация связей делает систему легко перестраиваемой в зависимости от предъявленных к ней требований и обеспечивает возможность ее расширения за счет подключения новых или замены старых внешних устройств ВУ. В качестве примера на рис. В.1 показана схема одной из возможных АСК с применением микроЭВМ и общей магистрали, где ВдТ — видеотерминал; УВВ — устройство ввода-вывода на перфоленту; УВыв — устройство вывода на печать; АЦП — аналого-цифровой преобразователь; УВыв и П — устройство вывода и приема релейных сигналов; УС с ОК — прочие устройства связи с объектом контроля. Ядром системы является микроЭВМ, выполняющая функции приема, обработки и передачи информации. Видеотерминал работает в качестве оперативного

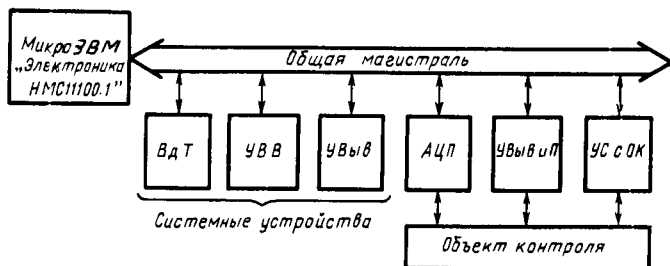


Рис. В.1. Пример схемы автоматизированной системы контроля.

средства связи между человеком-оператором и системой контроля. Устройство ввода-вывода на перфоленту используется для ввода программ контроля в оперативную память микроЭВМ перед началом работы, а также для вывода скорректированных программ. Для получения протокола результатов контроля объекта служит устройство вывода на печать.

Для осуществления контроля параметров объект должен быть оснащен системой измерительных датчиков. Большинство датчиков преобразуют некоторую физическую величину в электрическую. Примерами датчиков с электрическими выходами являются терморезисторы для измерения температуры, тензодатчики для измерения давления или незначительного смещения, потенциометры для измерения положения и т. п. Сигнал на выходе датчика (напряжение или ток) необходимо преобразовать в форму, приемлемую для использования в ЭВМ. Такое преобразование выполняется с помощью различного рода аналого-цифровых преобразователей (АЦП).

Как показывает отечественный и зарубежный опыт, системы сопряжения с датчиками и приемниками информации, как правило, значительно превышают микроЭВМ по объему и стоимости электронного оборудования.

Реализация преимуществ автоматизированного контроля на основе микроЭВМ возможна в полной мере лишь при соответствующей организации программного обеспечения. Основное ограничение на использование микроЭВМ в АСК связано с необходимостью разрабатывать программное обеспечение для каждого конкретного применения. Одним из вариантов, позволяющих преодолеть трудности программирования, является применение диалоговых языков высокого уровня. В настоящей работе для автоматизированного контроля предлагается известный диалоговый язык программирования ФОКАЛ, расширенный и дополненный специальными функциями [1]. Выбор именно этого языка объясняется удачным сочетанием следующих его важных качеств: а) функциональная простота и гибкость, обеспечивающие эффективную реализацию программного обеспечения; б) наличие средств управления общей магистралью; в) простота, легкость изучения и использования; г) компактность интерпретатора языка и программ пользователя; д) возможность работы в режиме диалога.

Проектирование системы контроля — комплексная задача, требующая от инженера-разработчика глубоких знаний как в области создания аппаратурных средств, так и в области программного обеспечения. В существующей в настоящее время технической литературе вопросу единого подхода к описанию и проектированию аппаратурных и программных средств уделяется явно недостаточное внимание. Материал, излагаемый в книге, построен таким образом, чтобы всюду, где это возможно, прослеживалась связь между аппаратурным и программным обеспечением. Наряду с общими вопросами организации микроЭВМ, необходимыми для понимания функционирования всей системы, приводятся конкретные примеры написания про-

грамм связи с внешними устройствами на языке машинного уровня, дается представление об организации интерпретирующей программы, описана методика включения новых функций в состав интерпретатора, дается пример организации и написания управляющей программы на языке высокого уровня.

Достигнутый отечественной электронной промышленностью высокий уровень технологии элементной базы средств вычислительной техники, характеризующий производством спектра МПК БИС (табл. В.2), БИС запоминающих устройств (табл. В.3), БИС интерфейса, интегральных схем аналоговой и цифроаналоговой техники, в совокупности с новыми архитектурными решениями открывает широкие возможности к массовому созданию информационно-управляющих вычислительных систем с более совершенными тактико-техническими характеристиками.

Т а б л и ц а В. 2

Обозначение МПК	Количество БИС в наборе	Разрядность	Тактовая частота, МГц	Схемотехника	Система команд	Напряжение питания, В	Тип корпуса
KP580	7	8	0,5	nМДП	Постоянная	+5; +12; -5	2123.40-2
K587	4	4n	0,5	КМДП	Произвольная	+9	429.42-1
KP588	5	16	1,0	КМДП	Постоянная	+5	2124.42-1
K589	9	2n	6,0	ТТЛДШ	Произвольная	+5	
K1800	4	4n	36,0	ЭСЛ	»	-5,2; -2	2207.48-1
K1801	3	16	8,0	nМДП	Постоянная	+5	429.42-5
KP1802	6	8n	8,0	ТТЛДШ	Произвольная	+5	2206.42-1
KP1804	6	4n	8,0	ТТЛДШ	»	+5	2123.40-3

Т а б л и ц а В. 3

№ п/п	Обозначение БИС ЗУ	Тип ЗУ	Информационная емкость	Время чтения записи, нс	Схемотехника	Напряжение питания, В	Мощность потребления, мВт/бит
1	K541РУ1	ОЗУ статическое	4096×1	400	И ² Л	+5	0,11
2	KP537РУ2	То же	4096×1	540	КМДП	+5	0,07
3	K565РУ1	ОЗУ динамическое	4096×1	400	nМДП	+5, +12, -5	0,175
4	K565РУ3	То же	16384×1	700	nМДП	+5, +12, -5	0,03
5	K588РР11	ППЗУ	1024×1	500	И ² Л	+5	0,3

В данном учебнике под *информационно-управляющей вычислительной системой* (ИУВС) будем понимать комплекс аппаратурных и программных средств, который под воздействием оператора или автоматически управляет состоянием какого-либо объекта. Работа ИУВС при управлении реальными объектами связана, как правило, с необходимостью обмена информацией с большим числом внешних устройств, с решением многих задач в реальном масштабе времени при их относительной неизменности в течение всего времени эксплуатации, с большой продолжительностью непрерывной работы при

Таблица В. 4

Технические характеристики	4P./CP-2	AP-101	MOD/LS1-10	SUMC (32)	SCP-050
Формат операндов	16—32	32	16/32/48	16/32/48	16/32
Представление чисел	Фиксированная точка	Плавающая точка	Плавающая точка	Плавающая точка	
Управление	Микрокомандное	Микрокомандное	Нет данных	Нет данных	Микрокомандное
Время выполнения операций, мкс:					
сложения	5—10	2,4	1,7	2,5	1
умножения	19	7,2	5,8	10,0	3
деления	47	Нет данных	11,7	35,0	54 (32)
Элементная база	ТТЛ ИС	ТТЛ ИС	ТТЛ, БИС, СИС	КМДП БИС	МП БИС, КМДП/КНС
Информационная емкость запоминающих устройств, бит:					
минимальная	8К	32К	4К	8К	8К
максимальная	32К	256К	256К	256К	64К
Число каналов устройства ввода-вывода информации	16-разрядный мультиплексный канал	16-разрядный мультиплексный канал	2	5	Нет данных
Мощность потребления, Вт	250	Нет данных	20	10	6
Масса, кг	23	23,4	7,7	5,3	2,7
Показатели надежности	Наработка на отказ 4000 ч	Наработка на отказ 1325 ч	Нет данных	Нет данных	Вероятность безотказной работы 0,8814 в течение двух лет

сохранении устойчивости к отказам. Это приводит к некоторым особенностям архитектуры (структуры и системного программного обеспечения) ИУВС по сравнению с универсальными средствами вычислительной техники.

Из структурных особенностей ИУВС следует отметить: 1) развитую систему сопряжения с большим количеством внешних устройств (как правило, датчиков информации); 2) использование различного рода каналов ввода-вывода и стандартного интерфейса; 3) развитую систему средств управления и отображения информации; 4) возможность прерываний работы ИУВС; 5) специализацию системы команд, ориентированную на выполнение алгоритмов определенного класса; 6) развитую систему памяти (применение наряду с ОЗУ ПЗУ и ППЗУ большой информационной емкости и быстродействия); 7) наличие системы аппаратурного контроля; 8) наличие счетчиков реального времени.

Системное программное обеспечение ИУВС должно быть ориентировано на: 1) выполнение вычислений в реальном масштабе времени; 2) обеспечение обмена информацией во время решения задач; 3) большой объем используемых данных и прикладных программ; 4) большое количество операций логического преобразования, поиска и упорядочения информации; 5) использование программно-логических способов защиты как от сбоев и отказов ИУВС, так и от сбоев во входной информации.

Примерами задач, решаемых ИУВС, могут служить задачи: 1) управления электромеханическими устройствами в реальном масштабе времени; 2) обработки сигналов в системах связи; 3) селекции сигналов в условиях шумов; 4) коммутации сообщений; 5) сбора и отображения информации. В частности, по данным [2], решение задач последнего типа может потребовать ИУВС с производительностью $40 \cdot 10^6$ операций/с и памятью 10^9 бит.

Некоторые технические характеристики (время выполнения команд, число команд, разрядность данных, габаритно-весовые и надежность показатели) ряда зарубежных ЭВМ, применяемых в авиационно-космических ИУВС, приведены в табл. В.4. Анализ тенденций совершенствования вычислительных средств для ИУВС [2] показывает, что если раньше целесообразным считалось оптимизировать характеристики ЭВМ в соответствии с конкретными задачами ИУВС, то в настоящее время более перспективно использование в ИУВС ЭВМ универсального типа. Тем не менее в ИУВС, к которым предъявляются весьма жесткие требования по быстродействию, надежности, большое значение по-прежнему будут иметь специализированные ЭВМ.

В целом в проектировании ЭВМ ИУВС настоящего поколения проявляются следующие тенденции: 1) применение БИС микропроцессорных наборов, БИС ЗУ и нестандартных (заказных) ИС высокой степени интеграции; 2) микропрограммный принцип управления; 3) представление данных в форме с плавающей точкой, что на 50 % сокращает время подготовки задач; 4) приближение ЭВМ ИУВС к универсальным ЭВМ, которое выражается в использовании языков

высокого уровня, арифметических операций в форме с плавающей точкой, модульности конструкции и т. д.; 5) стандартизация ЭВМ для ИУВС; 6) переход к адаптивным самоорганизующимся структурам; 7) стандартизация системы команд; 8) модульность программного обеспечения.

Примеры структурных решений ИУВС, построенных на базе микроЭВМ или БИС микропроцессорных наборов — в дальнейшем их будем называть микропроцессорными ИУВС (МП ИУВС), рассматриваются в § 2.3, однако необходимо отметить, что основу рассматриваемых МП ИУВС составляют одна или несколько микроЭВМ (микропроцессоров), устройство сопряжения с объектом, датчики входной информации и средства отображения информации и управления состоянием системы, предоставляемые оператору.

Авторы сделали попытку систематизировать имеющиеся в литературе рекомендации по проектированию МП ИУВС, поясняя их примерами из опыта собственных разработок МП ИУВС. В частности, предпринята попытка изложить методику назначения технических требований на разработку ИУВС при различных формах исходного задания; рассмотрены методика проектирования однопроцессорной ИУВС и архитектурные особенности и интерфейсы МП ИУВС; изложены методики оценки надежностных характеристик, рекомендации по конструированию ИУВС; описываются структурные и схемотехнические особенности построения ИУВС на базе МПК БИС серии К589.

Учитывая трудность выполнения поставленной задачи из-за многообразия подходов к созданию ИУВС в силу принципиальной зависимости ее архитектуры от класса решаемых ею задач, авторы понимают, что оно может служить лишь введением в проблему. Тем не менее они надеются, что для студентов вузов и для неспециалистов в области вычислительной техники, начинающих применять средства микропроцессорной вычислительной техники в ИУВС для народного хозяйства, учебник поможет разобраться в ключевых вопросах: с чего начать, как и какие технические вопросы необходимо решить.

I. СХЕМОТЕХНИКА СРЕДСТВ СОПРЯЖЕНИЯ

Проектируя ту или иную систему с использованием микроЭВМ, необходимо рассмотреть особенности схемотехники цифровых и аналоговых элементов сопряжения, а также схемы средней степени интеграции, являющиеся многофункциональными узлами ЭВМ. В разделе разбираются варианты структур устройств сопряжения с объектом, в том числе на базе аналого-цифрового микропроцессора; даются классификация и примеры организации интерфейсов; обсуждаются характеристики датчиков первичной информации.

1. СРЕДСТВА СОПРЯЖЕНИЯ МИКРОЭВМ С ОБЪЕКТОМ

1.1. Средства автоматического ввода и вывода данных в системах сбора и обработки информации

Структура устройств автоматического ввода данных. Рост сложности объектов управления, контроля, измерения, совершенствование технологических процессов обусловили необходимость высокой степени автоматизации процессов вычисления параметров управления, проведения процедур контроля, измерения элементов технологических процессов. Все эти различные по своей физической сущности задачи могут быть решены на основе использования автоматизированных систем сбора и обработки данных (система СОД). Основная тенденция проектирования современных СОД направлена на создание децентрализованных конфигураций систем, которые базируются на применении в качестве аппаратуры обработки данных МП или мини-ЭВМ.

Технические средства для реализации автоматизированных систем СОД должны содержать комплекс устройств, реализующих задачу ввода информации в МП без участия человека-оператора. Эта задача решается на основе введения датчиков первичной информации для преобразования неэлектрических величин в пропорциональные электрические. На основе изложенного можно сказать, что автоматизированная система сбора и обработки данных представляет собой сложное техническое устройство, которое в наиболее общем случае осуществляет автоматическое вычисление совокупных параметров управления, контроля, измерения объектов, различных по своей физической сущности, и регистрацию результатов вычислений.

Характерные особенности системы СОД целесообразно рассмотреть во взаимодействии с объектом (рис. 1.1).

самоконтроля СОД, а функциональные зависимости $\varphi_1, \varphi_2, \dots, \varphi_k$ образуют контрольные тесты СОД.

Обычно совокупность допустимых значений $C_1, C_2, \dots, C_n; P_1, P_2, \dots, P_m; r_1, r_2, \dots, r_k$ интерпретируют как многомерное пространство (C -мерное, P -мерное, r -мерное), в пределах которого существование объекта и системы СОД имеет практическую ценность. В этом случае сочетание произвольно заданных значений C_i, P_i, r_i представляет собой координаты вектора многомерного пространства, что позволяет использовать для математического выражения связей объекта и системы СОД элементы матричного исчисления. В результате анализа математического выражения связей возможно рациональное построение структуры системы сбора данных, зависящей от конкретных условий задачи анализа. Однако в общем случае, исходя из требования ввода информации в МП, она должна содержать: группу датчиков первичной информации, размещенных в непосредственной близости от объекта управления, контроля или измерения, блок преобразователей аналоговых сигналов датчика в цифровые коды (АЦП), цифровые и аналоговые коммутаторы, аналоговые вычислительные устройства, цифровые и аналоговые каналы связи, блок преобразователей цифровых сигналов в аналоговые (ЦАП), цифровые регистрирующие устройства.

Различные требования по стоимостным и техническим характеристикам, предъявляемые к СОД, определяют возможность вариаций структуры СОД на основе выбора метода обработки сигналов.

На рис. 1.2,а показана структура, реализующая принцип параллельной обработки аналоговых сигналов, поступающих от датчиков СОД. Данная структура позволяет обеспечить максимальную производительность аппаратуры всех каналов систем СОД (из-за независимости обработки каждого сигнала) и высокое качество преобразования сигналов вследствие возможности системы по обеспечению требуемого уровня нормализации сигнала на входе АЦП в каждом канале. Предельное количество каналов системы СОД в данном

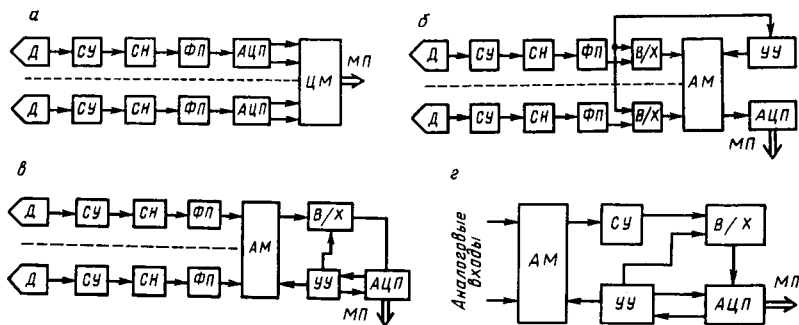


Рис. 1.2. Структура системы СОД с параллельными цифровыми (а) и аналоговыми (б) выходами; с последовательными аналоговыми (в) и с последовательными аналоговыми (г) выходами:

Д — датчик; СУ — согласующее устройство; СН — схема нормализации; ФП — функциональный преобразователь; АЦП — аналого-цифровой преобразователь; ЦМ — цифровой мультиплексор; В/Х — схема выборки — хранения; АМ — аналоговый мультиплексор; УУ — устройства управления.

случае ограничено допустимым значением динамической погрешности преобразования i -го параметра, т. е.

$$N_u \leq \Delta x_{i \text{ доп}} / [(dx_i/dt)_{\max} t_k], \quad (1.4)$$

где N_u — предельно допустимое количество цифровых каналов преобразования системы СОД; $\Delta x_{i \text{ доп}}$ — допустимое значение динамической погрешности i -го параметра; $(dx_i/dt)_{\max}$ — максимальное значение производной i -го параметра в диапазоне преобразования; t_k — период коммутации.

Независимость отдельных ветвей дает возможность просто (в структурном смысле) реализовать необходимые функциональные преобразования аналоговых сигналов. Большое преимущество такой системы заключается в устранении ошибок, возникающих при коммутации и выборке — хранении аналоговых сигналов, вносящих основной вклад в суммарную погрешность преобразования. Очень важно то, что эта структура позволяет значительно повысить помехозащищенность системы на основе использования цифровых каналов передачи данных на вход ЭВМ, если обеспечить преобразования сигналов непосредственно на выходе датчиков. Такой вариант построения системы сбора данных имеет большое будущее, однако на современном этапе развития микросхемотехники при его реализации имеет место более высокая стоимость СОД вследствие сравнительно высокой стоимости интегральных схем АЦП.

Другие варианты построения структуры системы СОД основаны на принципе последовательной обработки аналоговых сигналов и перенесении мультимплексирования из цифровой в аналоговую область, а также применения аналоговых схем выборки — хранения для фиксации аналоговых сигналов на входе АЦП. Это дает возможность использовать только один АЦП независимо от числа аналоговых входов, предельное количество которых можно определить из выражения

$$N_a \leq \Delta x_{i \text{ доп}} / [(dx_i/dt)_{\max} (t_k + t_n)], \quad (1.5)$$

где N_a — предельно допустимое количество аналоговых каналов преобразования СОД; t_n — период цикла преобразования [остальные обозначения соответствуют (1.4)].

Высокие технические характеристики свойственны структуре, приведенной на рис. 1.2, б, хорошее качество преобразования обеспечено теми же функциональными блоками, которые были включены в состав структуры, приведенной на рис. 1.2, а. Очевидно, что производительность системы сбора данных, построенной по структуре, изображенной на рис. 1.2, б, находится в прямой зависимости от быстродействия АЦП и ограничена его динамическими параметрами, поэтому в подобных системах необходимо применять АЦП с максимальным быстродействием. Применение дополнительных элементов для обработки аналоговых сигналов мультимплексоров, схем выборки — хранения ухудшает точностные характеристики системы в це-

лом. Практически идентичными техническими характеристиками по сравнению со структурой, изображенной на рис. 1.2, б, обладает система сбора данных, реализованная на основе структуры, показанной на рис. 1.2, в.

Аппаратурные затраты для реализации канала сбора данных на основе структуры, приведенной на рис. 1.2, б, будут большими по сравнению с затратами для построения канала с одной схемой выборки — хранения. Эту конфигурацию структуры наиболее рационально использовать для преобразования однократных событий.

Наиболее простой, но обеспечивающей относительно низкое качество преобразования является система, построенная по структуре, изображенной на рис. 1.2, г. Эта конфигурация структуры рациональна при условии существования сигналов высокого уровня, поступающих с датчиков системы СОД, и идентичности всех аналоговых сигналов, в этом случае можно добиться необходимого качества преобразования, сохраняя достоинства по простоте реализации.

Как правило, систему СОД в целом и систему сбора информации в частности с целью экономии средств и сокращения времени разработки наиболее рационально компоновать из отдельных блоков, реализующих функции компонентов структуры. Из анализа возможных вариантов построения структур (рис. 1.2) следует, что любой вариант реализуется на основе восьми основных функциональных устройств: датчик, схема согласования, схема нормализации, функциональный преобразователь, АЦП, цифровые мультиплексоры, аналоговые мультиплексоры, устройства управления.

Подсистема аналогового ввода. Поскольку преобразование основных физических величин в электрические обычно не относится к функциям подсистемы аналогового ввода, ограничимся рассмотрением преобразования электрических сигналов в цифровую форму. Процесс получения аналогового сигнала с какого-либо датчика, его предварительной обработки, преобразования в цифровую форму и ввода в память ЭВМ называется *сбором данных*. Рис. 1.3 иллюстрирует

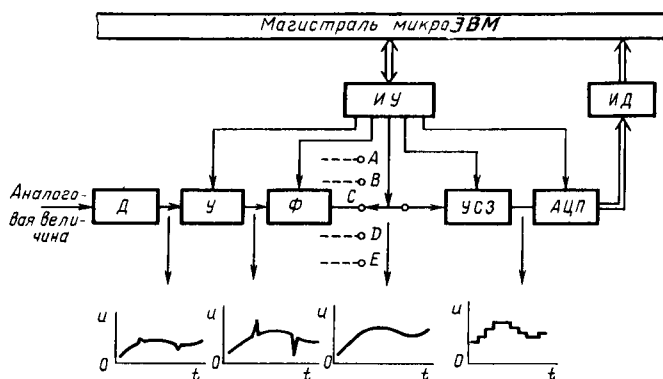


Рис. 1.3. Схема подсистемы аналогового ввода:

ИУ — интерфейс управления; ИД — интерфейс данных.

общую схему подсистемы сбора данных. Контролируемый параметр — уровень напряжения, ток, давление, температура и т. п. — с помощью соответствующего датчика Д преобразуется в электрический сигнал. Если сигнал на выходе датчика недостаточен для последующей обработки, то он усилителем У усиливается до необходимого уровня. Далее с помощью активного фильтра Ф из сигнала удаляются нежелательные низкочастотные и высокочастотные составляющие (необходимость в предварительной фильтрации будет пояснена позднее). Отфильтрованный сигнал поступает через аналоговый мультиплексор на устройство слежения — запоминания УСЗ. Применение мультиплексора позволяет в режиме разделения времени использовать единственный аналого-цифровой преобразователь АЦП для многих каналов, экономя тем самым дорогостоящее оборудование.

Преобразование аналоговой величины в цифровую всегда происходит за некоторый конечный промежуток времени. В течение этого промежутка времени сигнал на входе АЦП должен поддерживаться неизменным. Эту функцию выполняет устройство слежения — запоминания УСЗ, сигнал на выходе которого пропорционален сигналу на входе до тех пор, пока не последует команда запоминания, после которой сигнал на выходе остается постоянным в течение промежутка времени, необходимого для преобразования сигнала в цифровую форму (рис. 1.4).

АЦП преобразует уровень напряжения на входе в соответствующую цифровую величину. Далее цифровые данные через интерфейсные схемы сопряжения поступают в общую магистраль микроЭВМ.

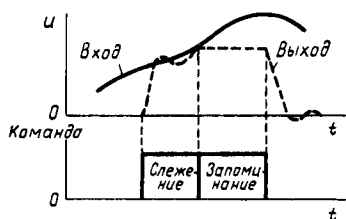


Рис. 1.4. Временная диаграмма работы устройства слежения — запоминания.

Из приведенной схемы сбора данных видно, что непрерывно изменяющийся входной сигнал заменяется совокупностью дискретных периодически измеренных значений, причем внутри каждого интервала амплитуда сигнала сохраняется неизменной. Процесс квантования искажает входной сигнал. Теорема Котельникова о дискретном представлении утверждает, что для точного восстановления первоначального сигнала из дискретного частота опроса должна по крайней мере вдвое превышать самую высокочастотную гармоническую составляющую входного сигнала. Частота квантования ограничена сверху скоростью работы АЦП. Время, необходимое для выполнения одного преобразования, называется *апертурным*. Если во входном сигнале имеются гармонические составляющие с более высокой частотой, чем частота квантования, то при восстановлении аналогового сигнала они вызовут разностные частоты более низкой частоты.

ты. Для предотвращения такого искажения следует использовать фильтр нижних частот, который бы не пропускал все составляющие сигнала, имеющие частоту выше $1/2$ частоты квантования. Только при этих условиях процесс квантования не вызовет потерь информации, содержащейся во входном сигнале.

Следует отметить, что из-за ограниченной разрешающей способности цифрового представления возникает также и амплитудная погрешность, которую необходимо учитывать.

В настоящее время аналоговые подсистемы выпускаются и в конструктивно законченном виде. Так, подсистема аналогового ввода фирмы «Data General» содержит плату, состоящую из двух 8-канальных мультиплексоров, усилителя с дифференциальным входом, усилителя слежения — запоминания и 12-разрядного АЦП последовательного приближения. Время преобразования составляет 33 мкс. Входные сигналы могут изменяться в диапазоне ± 10 В. Фирма «Analog Devices Inc» производит аналоговый интерфейс, включающий в себя плату, на которой размещены усилитель с программным усилением (1—1000) и 12-разрядный АЦП.

Отечественной промышленностью выпускается модуль аналогового ввода 15КА-60/8-010, предназначенный для преобразования напряжения постоянного тока в цифровой код и ввода его в микроЭВМ «Электроника-60». Модуль имеет в своем составе мультиплексор аналоговых сигналов на 16 входов. Входы можно использовать либо как однопроводные (для 16 источников входных сигналов), либо как дифференциальные (для 8 источников входных сигналов). Имеется возможность выбора одного из двух диапазонов входных напряжений: ± 1 или ± 10 В. Напряжение на входе с помощью АЦП поразрядного уравнивания преобразуется в 11-разрядный двоичный код (включая и знаковый разряд). Время преобразования составляет не более 100 мкс.

Упрощенная схема модуля аналогового ввода приведена на рис. 1.5, а, где Пд — передатчик; Пр — приемник; Т_{зн} — триггер знака; РгДВв — регистр данных ввода; Т — триггер «Готовность»; ДША и Упр — дешифратор адреса и управляющих сигналов; СОП — схема определения полярности; АЦП — аналого-цифровой преобразователь; СУС — схема управления и синхронизации АЦП; РгДВыв — регистр данных вывода; У — усилитель; МС — мультиплексор; АК — адрес канала. Рассмотрим принцип его работы. Вначале с помощью программного цикла «Запись» выбирается один из входных каналов. Селектор адреса и дешифратор управляющих сигналов вырабатывают разрешающий сигнал «Вывод 0», по которому разрешается запись кода адреса выбранного канала, поступающего из магистрали микроЭВМ в регистр данных вывода (биты Д₀₀ — Д₀₃). С выхода регистра код поступает на адресный вход АК мультиплексора МС. Напряжение из выбранного канала поступает на усилитель У и далее на АЦП и схему определения полярности входного сигнала СОП. Схема определения полярности позволяет упростить реализацию АЦП.

Одновременно сигнал «Вывод 0» запускает схему управления и

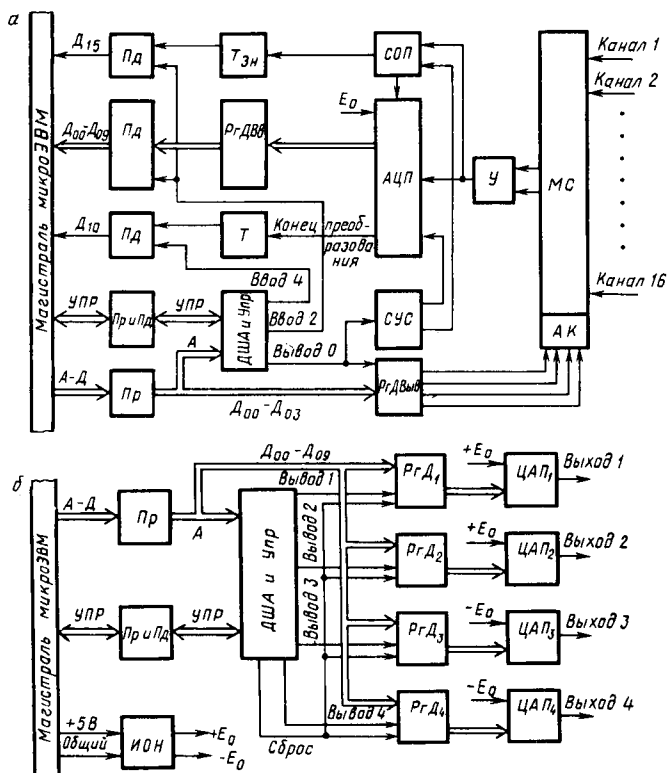


Рис. 1.5. Схема модуля аналогового ввода (а) и вывода (б).

синхронизации АЦП. После завершения цикла преобразования по сигналу «Конец преобразования» устанавливается в единицу триггер $Т$ «Готовность» (бит D_{10}). С помощью программного цикла «Чтение» состояние триггера «Готовность» может быть считано микроЭВМ (разрешающий сигнал «Ввод 4»). Прием результата преобразования в микроЭВМ осуществляется также с помощью программного цикла «Чтение», при этом вырабатывается разрешающий сигнал «Ввод 2» и через передатчики код данных ($D_{00} \dots D_{09}$) и код знака (D_{15}) поступают в магистраль микроЭВМ.

Для обеспечения гальванической развязки системы питания микроЭВМ и измерительной части модуля используется изолированный источник питания. Гальваническая развязка интерфейсной части модуля и его аналоговой части осуществляется с помощью оптоэлектронных ключей.

В последнее время ряд фирм разработали устройства ввода аналоговой информации в виде гибридных микросхем. Разработаны приборы HDAS-8 (8 дифференциальных входов) и NDAS-16 (16 обычных входов), которые, кроме обязательного набора элементов, содержат

тристабильные буферные устройства для сопряжения с МД микропроцессора. Режим непрерывного сканирования обеспечивается путем коммутации внешних выводов. Устройство позволяет без значительных затрат увеличить число каналов до 256. Быстродействующие схемы выборки и хранения дают возможность анализировать в реальном масштабе времени сигналы с частотой до 25 кГц.

Разработано устройство аналогового ввода в виде КМДП прибора АДСО816. На кристалле размером $3,7 \times 5,0$ мм расположены все элементы, необходимые для стандартной системы ввода, исключая измерительные усилители. Это устройство обладает многими полезными для ввода аналоговой информации свойствами: 16-канальный мультиплексор непосредственно воспринимает любой из 16 входных сигналов, а затем либо подает этот сигнал для прямого преобразования на входную линию, либо выдает его во внешний прибор для дополнительного преобразования. Наличие регистра хранения, дешифратора адреса и тристабильных элементов упрощает сопряжение между прибором и МП.

Рассмотренные варианты построения подсистем аналогового ввода свидетельствуют о том, что их развитие идет в направлении от специализированных плат к гибридным и монолитным микросхемам. Характерная черта всех модулей — наличие средств сопряжения с микропроцессором и возможность программного выбора каналов.

Подсистема аналогового вывода. Подсистема аналогового вывода во многом похожа на подсистему аналогового ввода. В АСК подсистема аналогового вывода используется для подачи на объект контроля тестовых испытательных сигналов в виде напряжений или токов, изменяющихся во времени по заданному закону. Она может быть также применена для вывода из ЭВМ результатов контроля в виде графиков и диаграмм с помощью самописцев и графопостроителей.

Преобразование цифровых выходных данных в аналоговый сигнал выполняется с помощью ЦАП. Принцип преобразования двоичного кода в напряжение будет рассмотрен в § 1.7.

Основные трудности при создании прецизионных ЦАП возникают при изготовлении резистивных делителей.

Существуют различные конфигурации подсистем аналогового вывода, однако наибольшее распространение получили подсистемы с использованием ЦАП в каждом выходном канале.

Рассмотрим подсистему аналогового вывода, построенную на базе серийно выпускаемого модуля 15КА-60/4-009 для микроЭВМ «Электроника-60». Модуль содержит четыре параллельных канала и предназначен для преобразования цифровых данных, выдаваемых микроЭВМ, в напряжение постоянного тока. Упрощенная схема модуля приведена на рис. 1.5, б. Модуль состоит из приемников Пр и передатчиков Пд магистрали; дешифратора адреса и управляющих сигналов ДША и Упр; регистров данных $RгД_1 — RгД_4$; 10-разрядных ЦАП₁ — ЦАП₄; источника опорного напряжения E_0 .

Код адреса А и управляющие сигналы Упр поступают из магист-

рали микроЭВМ через приемники Пр на дешифратор адреса и управляющих сигналов ДША и Упр, который вырабатывает один из сигналов обращения к выбранному регистру (*Вывод_i*). Код данных (Д₀₀—Д₀₉) поступает на информационные входы всех регистров РгД_i, однако запись произойдет только в выбранный регистр. С выходов регистра код поступает в ЦАП, где преобразуется в напряжение постоянного тока. Цифровая и аналоговая части модуля гальванически развязаны с помощью оптоэлектронных ключей. Питание аналоговой части модуля осуществляется от изолирующего источника напряжения. Источник опорного напряжения Е₀ вырабатывает два отдельных напряжения: положительное (+10,24 В) и отрицательное (—10,24 В). По выбору с помощью перемычек можно на любой из ЦАП подать соответствующее опорное напряжение Е₀ и получить на выходе ЦАП напряжение нужной полярности. Двоичные разряды, их веса и соответствующие им выходные напряжения приведены ниже:

Разряд	Д ₀	Д ₁	Д ₂	Д ₃	Д ₄	Д ₅	Д ₆	Д ₇	Д ₈	Д ₉
Двоичный вес	2 ⁰	2 ¹	2 ²	2 ³	2 ⁴	2 ⁵	2 ⁶	2 ⁷	2 ⁸	2 ⁹
Выходное напряжение										
В	5,12	2,56	1,28	0,64	0,32	0,16	0,08	0,04	0,02	0,01

Модуль обеспечивает по любому каналу ток не более 5 мА при выходном сопротивлении не более 1 Ом. Время установления выходного напряжения не более 10 мкс.

Подсистема цифрового ввода-вывода. Для АСК характерны не только аналоговые входные и выходные величины, но и величины, являющиеся по своей природе цифровыми, которые могут быть представлены двумя состояниями, например «Включено» или «Выключено», «В пределах» или «Вне пределов», «Низкий уровень» или «Высокий уровень». Кроме цифровых параметров, характеризующих состояние контролируемого объекта, существуют выходные цифровые параметры, которые задают желательное состояние объекта. Например, при контроле объекта часто возникает необходимость включения или выключения некоторых составляющих его блоков и модулей или подачи в контролируемый объект комбинаций цифровых испытательных сигналов.

Сигналы на входе, представляющие состояние контролируемого объекта, обычно имеют вид уровня напряжения, тока или состояния контактов («Замкнуто — разомкнуто»). Для сопряжения этих сигналов с ЭВМ необходимо их преобразовать в стандартные уровни. Как правило, входные цепи ЭВМ управляются ТТЛ-уровнями: «0» соответствует уровень напряжения 0—0,4 В, а «1» — уровень напряжения 2,5—5 В. В качестве преобразующих схем используют резистивные делители напряжения, диодные ограничители, компараторы, триггеры Шмитта, согласователи уровней.

Основная функция цифрового вывода — функция ключа. Обычно каждый двоичный разряд выходных цифровых данных имеет самостоятельный смысл, т. е. каждый бит выходного слова может использоваться для управления каким-либо параметром объекта независи-

мо. Выбор типа ключа определяется значением коммутируемой мощности и скорости переключения.

Электромеханические реле применяются в основном для управления сигналами средней и большой мощности при низких скоростях переключения. Они обеспечивают полную гальваническую развязку цепей.

Более надежные — полупроводниковые ключи.

Для коммутации цепей переменного тока средней и большой мощности часто используются кремниевые управляемые выпрямители.

Практически совершенными переключающими характеристиками обладают разработанные в середине 70-х годов полевые транзисторы с V-структурой (VMOS). Их отличают низкий входной управляющий ток, высокая частота переключения и значительные выходные токи. Так, мощность рассеяния прибора VN84GA составляет 90 Вт при низкой частоте переключения и 50 Вт при частоте переключения 30 МГц. Для гальванической развязки полупроводниковых ключей могут служить оптоэлектронные пары.

В качестве основы для построения подсистемы цифрового ввода-вывода удобно выпускаемое отечественной промышленностью устройство параллельного обмена И2 15КС-180-032. Оно предназначено для подсоединения к магистрали микроЭВМ «Электроника-60» внешних устройств, обменивающихся данными с ЭВМ в параллельном коде, и представляет собой стандартную плату, вставляемую в один из разъемов магистрали. На рис. 1.6 приведена упрощенная схема этого устройства. При рассмотрении принципов работы коснем-

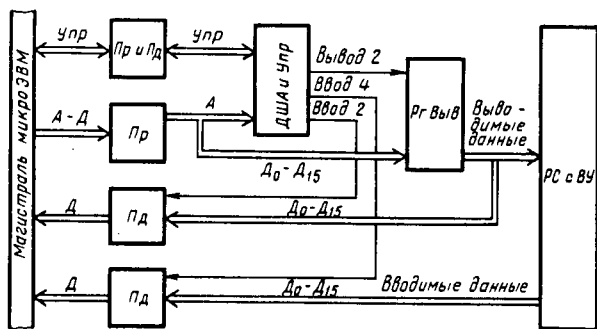


Рис. 1.6. Схема модуля цифрового ввода-вывода.

ся лишь функций, выполняемых устройством при безусловной программной передаче данных от внешнего устройства в ЭВМ и из ЭВМ во внешнее устройство. В этом случае используются два адресуемых регистра.

Для передачи данных из ЭВМ во внешнее устройство ВУ адрес регистра вывода РгВыв и управляющие сигналы Упр через приемник подаются на дешифратор адреса и управляющих сигналов ДША и Упр, который в свою очередь вырабатывает сигнал разрешения «Вывод 2». По этому сигналу происходит запись данных (Д0—Д15) в РгДВыв (данные к этому моменту присутствуют на информационных

входах регистра). С выводов регистра данные поступают на разъем связи РС с внешним устройством ВУ.

При необходимости данные, хранящиеся в регистре вывода РгДВыв, могут быть считаны программным путем. Для этой цели служат разрешающий сигнал «Ввод 2» и передатчики магистральной линии.

При вводе данных из внешнего устройства в ЭВМ используются адрес регистра ввода и цикл магистральной линии «Чтение». Следует заметить, что регистр ввода как таковой в устройстве параллельного обмена отсутствует (есть лишь его адрес). При этом вырабатывается разрешающий сигнал «Ввод 4». Вентильные схемы передатчиков открываются, и код из внешнего устройства поступает в магистраль ЭВМ. Таким образом, с помощью рассмотренного устройства можно организовать выдачу 16 управляющих сигналов в контролируемый объект, а также обеспечить прием дискретных сигналов по 16 входным каналам.

1.2. Датчики первичной информации

Типы датчиков. В основе организации автоматической системы ввода данных лежит применение датчиков входной информации с электрическим выходом и последующим преобразованием электрического сигнала в цифровую форму. Так как основные характеристики датчиков являются определяющими для системы сбора данных и систем управления, работающих в реальном масштабе времени, то они представляют интерес. В соответствии с измеряемым параметром удобно разделить на группы существующие датчики: измерения угловых и линейных величин; усилий, крутящих моментов, давлений, напряжений; параметров движения; измерения температуры.

Угловые размеры чаще всего ограничены 360 угл. град. В зависимости от требуемой точности измерения используют различные принципы построения преобразователей «угол — электрическая величина». Примером подобного типа преобразователей могут служить реостатные и дискретные на основе зубчатых и кодовых дисков с контактным, фотоэлектрическим или индуктивным съемом информации. Измерение угловых размеров реостатными преобразователями может быть реализовано с погрешностью 0,05—0,03 %. Возможность обеспечения большой выходной мощности значительно упрощает построение системы сбора данных. В качестве таких преобразователей применяют проволочные реостатные преобразователи с различным пределом измерения от долей одного оборота до десятков оборотов. Сопротивление преобразователя в зависимости от угла поворота

$$R_a = R_{\max}(\alpha/\alpha_{\max}) = R_{\max}f(\beta), \quad (1.6)$$

где $\beta = \alpha/\alpha_{\max}$ — относительное перемещение элементов преобразователя, который при включении в электрическую цепь может быть использован в качестве косвенного преобразователя угла поворота в ток или напряжение по зависимостям

$$\left. \begin{aligned} I_n &= \frac{U_0}{R_a} = \frac{U_0}{R_{\max}f(\beta)} = I_{\min}f^{-1}(\beta); \\ U_n &= I_0 R_a = I_0 R_{\max}f(\beta) = U_{\max}f(\beta). \end{aligned} \right\} \quad (1.7)$$

где U_0 , I_0 — опорные напряжения и ток преобразователя.

Разновидность высоконадежного реостатного датчика с неизменяющимся контактом изображена на рис. 1.7. Поворот датчика приводит к изменению сопротивлений $R1$, $R2$, а функциональная характеристика датчика может быть записана выражением (1.7); погрешность такого преобразователя достигает 1 %.

Очень часто для измерения параметров угла поворота используют вращающиеся трансформаторы (ВТ). На цилиндрических статоре и роторе размещаются по две обмотки, причем как обмотки статора ω_e , ω_k , так и обмотки ротора ω_s , ω_c сдвинуты относительно друг друга на 90 эл. град. В силу электрической перпендикулярности обмоток ω_e , ω_k их воздействия на вторичные обмотки ω_s , ω_c независимы и характеристика преобразования ВТ имеет вид

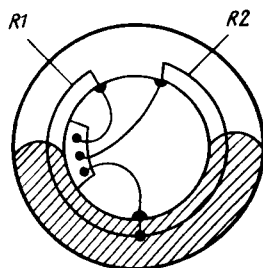


Рис. 1.7. Схема реостатного датчика.

$$\begin{aligned} U_s &= K_T (U_1 \sin \alpha - U_k \cos \alpha); \quad U_c = \\ &= K_T (U_1 \cos \alpha + U_k \sin \alpha), \end{aligned} \quad (1.8)$$

где $K_T = \omega_s / \omega_e = \omega_c / \omega_k = 0,1 - 1,0$ — коэффициент трансформации ВТ; α — угол поворота ротора ВТ.

Передаточные коэффициенты ненагруженного идеального ВТ изменяются по синусной или косинусной зависимости от угла поворота ротора:

$$K_{1s} = K_T \sin \alpha; \quad K_{1c} = K_T \cos \alpha; \quad K_{ks} = -K_T \cos \alpha; \quad K_{kc} = K_T \sin \alpha.$$

Достоинством подобных преобразователей является то, что с датчика снимаются напряжения, пропорциональные синусу и косинусу угла поворота ротора, что значительно упрощает процессы последующей обработки информации потому, что для формирования алгоритма управления чаще всего необходимы именно эти зависимости. На выходах ВТ напряжение

$$U_s = K_T U_1 \sin \alpha; \quad U_c = K_T U_1 \cos \alpha \quad \text{при} \quad U_1 = \text{const}, \quad U_k = 0.$$

В настоящее время все чаще используют дискретные преобразователи угла поворота — цифровой код, которые необходимы прежде всего для удовлетворения потребностей в точных измерениях угловых параметров. Цифровой выход датчика значительно упрощает ввод данных в МП вследствие отсутствия необходимости обработки аналоговых сигналов.

Как правило, в машиностроительной промышленности требуется измерять не полный диапазон заданного размера, а лишь его отклонения, которые обычно не превосходят долей миллиметра. Электрические микрометры широко применяют и обычно строят на основе электроконтактных, индуктивных и емкостных преобразователей. Шероховатость поверхности оценивается среднеквадратическим отклонением микронеровностей. Для их измерения используют осевые перемещения штока по испытываемой поверхности, которой шток касается своим заостренным наконечником. На рис. 1.8, а показана

конструкция профилографа. Датчик содержит шток 1, на его верхнем конце укреплен ферритовый стержень 2, который при движении штока изменяет индуктивность катушки 3. Электрическая схема включения профилографа приведена на рис. 1.8, б. Катушка

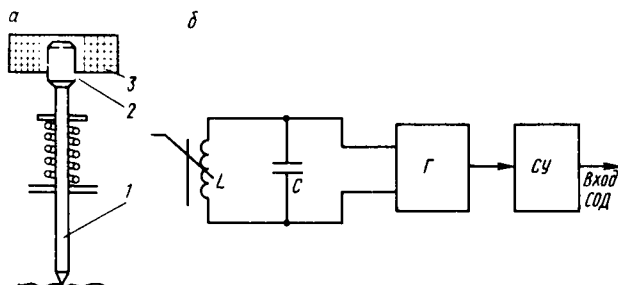


Рис. 1.8. Конструкция (а) и схема включения (б) профилографа.

профилографа L и конденсатор C образуют LC -генератор, к выходу последнего подключен частотный демодулятор. Напряжение с демодулятора поступает на вход канала сбора данных.

Примером метода измерения толщин объектов, доступных лишь с одной стороны, может служить датчик ИТП-476, в основу которого положен метод рассеянного β -излучения. Прибор (рис. 1.9) содержит три источника β -излучения 1, 4, 6, усилитель 8 и двигатель 7.

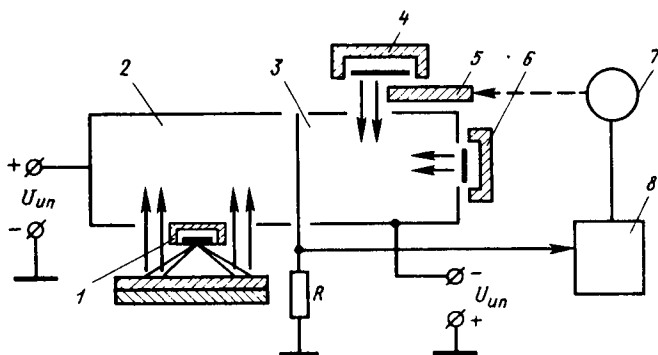


Рис. 1.9. Схема датчика для измерения толщин.

Приемник излучения представляет собой дифференциальную ионизационную камеру, состоящую из рабочей камеры 2 и компенсационной 3 с общим собирающим электродом, который через общее собирающее сопротивление R подключен к шине нулевого потенциала. Излучение радиоактивного изотопа рабочего источника 1 отражается от исследуемой поверхности; отраженное излучение содержит две составляющие: I_1 — отражение от подложки, не зависящее от толщины покрытия, I_2 — отражение от покрытия, зависящее от его

толщины. В компенсационную камеру излучение попадает от двух источников — дополнительного 6 и компенсирующего 4, которые уравнивают в ионизационной камере составляющие I_1 и I_2 рабочего источника. Напряжения питания, подаваемые на корпуса камер 2 и 3, имеют противоположные знаки, поэтому результирующий ток через резистор R равен разности ионизационных токов камер, а ток регулировкой перемещения заслонки 5 может быть, при определенной отражающей способности объекта, сведен к нулю. Следовательно, угловое положение вала двигателя 7 определяет толщину покрытия. Датчик ИТП-476 позволяет измерять толщину оловянного покрытия до 5 мм и цинкового покрытия до 0,1 мм с погрешностью порядка 2 %.

Очень часто встречается необходимость в измерениях уровней жидкости. Примером серийного датчика измерения уровня может служить серийно выпускаемый прибор УДУ-5 (рис. 1.10). Выходное напряжение датчика, пропорциональное уровню жидкости h , снимается с реостатного преобразователя. Прибор УДУ-5 при пределе измерения 12 м имеет погрешность ± 15 мм.

Широко используют для измерений уровня емкостные датчики. Датчик (рис. 1.11, а) построен на четырех коаксиаль-

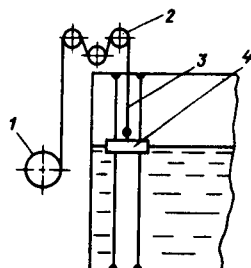


Рис. 1.10. Датчик измерения уровня жидкости:
1 — датчик; 2 — направляющие ролики; 3 — тросик; 4 — ползавок.

ных конденсаторах: трех компенсационных C_2 , C_3 , C_4 и одном рабочем C_1 . Электрическая схема датчика (рис. 1.11, б) содержит генератор G , усилитель U и трансформаторы Tr_1 , Tr_2 . Датчик реализует режим статического уравнивания, значит сумма токов, поступающих на вход усилителя через емкости $C_1—C_4$, равна нулю.

Напряжение с выхода датчика пропорционально уровню жидкости.

Измерение механических напряжений и деформаций в промышленности основано на применении тензорезисторов, струнных или

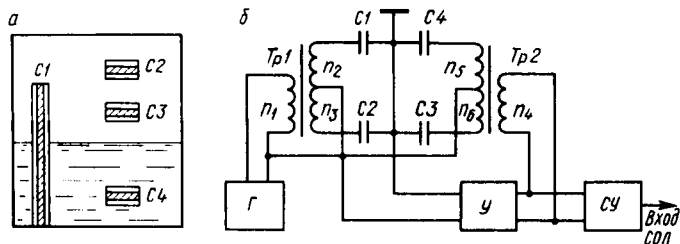


Рис. 1.11. Емкостный датчик измерения уровня жидкости (а) и его схема (б).

индуктивных датчиков. Для измерений деформаций, лежащих в диапазоне от 0,005—0,002 до 1,5—2 %, наиболее часто используют проволоочные, фольговые, пленочные тензорезисторы при измерении механических колебаний в диапазоне от 0 до 100 кГц, так как они безынерционны.

Исследования линейных напряжений с помощью тензодатчиков методически подробно разработаны, и в основе работы тензорезисторов лежит явление тензоэффекта, заключающееся в том, что активное сопротивление проводников меняется при их механической деформации. Тензоэффект материала характеризуется коэффициентом относительной тензочувствительности $K = \varepsilon_R / \varepsilon_l$, где $\varepsilon_R = \Delta R / R$ — относительное изменение сопротивления проводника; $\varepsilon_l = \Delta l / l$ — относительное изменение длины проводника.

Конструктивно тензодатчики выполняются в виде проволоочки, которая зигзагообразно наклеивается на тонкую бумагу или лаковую пленку (рис. 1.12, а), к концам проволоочки припаяны медные выводы.

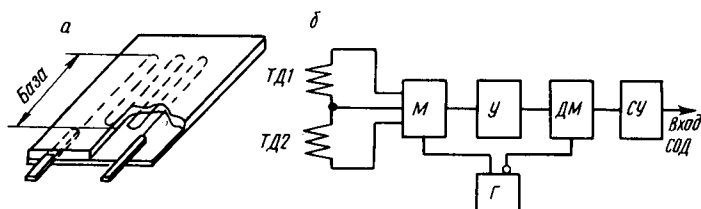


Рис. 1.12. Конструкция тензодатчика (а) и структура одного канала тензостанции (б).

Такой датчик приклеивается к испытуемой детали и воспринимает деформации ее поверхностного слоя. Выходной величиной тензопреобразователя является изменение его сопротивления, пропорциональное величине деформации. Кроме петлевого наклеиваемого тензодатчика для специальных исследований, разработаны и другие типы датчиков. Например, эластичные резиновые капилляры, заполненные ртутью, позволяют измерять деформации до 30—50 %.

Схемы измерительных цепей тензорезисторов могут быть построены весьма разнообразно. Наиболее типичная структура одного канала тензостанции приведена на рис. 1.12, б. Измерительный мост запитан переменным напряжением генератора Г, модулированный сигнал усиливается усилителем У, демодулируется ДМ и после фильтрации поступает на обработку или фиксацию. Обычно в машиностроении исследуются деформации во многих точках; следовательно, тензостанция, как правило, многоканальна.

Тензорезисторы используют также для вторичных преобразований при измерении сосредоточенных сил и давлений в датчиках с упругими элементами. Деформация упругого элемента сплошного стержня, кольца и др., возникающая от воздействия внешней силы, изменяет сопротивление тензорезистора, которое преобразуется в электрический сигнал.

Часто СОД строят на цифровых радиоизмерительных приборах, которые в подобных системах реализуют функции датчика первичной

информации. Если к системе сбора данных не предъявляют жестких требований по габаритно-весовым характеристикам, то радиоизмерительные приборы (генераторы сигналов) используют для формирования сигналов манипуляции внутренним состоянием объекта управления или измерения. Для применения в СОД радиоизмерительные приборы должны иметь цифровые выходы для считывания информации и цифровые входы управления режимами работы. Отечественная промышленность выпускает широкую номенклатуру такого типа радиоизмерительных приборов. Например, универсальный вольтметр В7-30 предназначен для измерения постоянного и синусоидального напряжений, сопротивления постоянному току, отношения постоянных напряжений, отношения синусоидального напряжения к постоянному, мгновенного значения входного напряжения в режиме выборки-хранения, работы с каналом коллективного пользования. Вольтметр обеспечивает измерение постоянного напряжения двух полярностей на пределах 0,1; 10; 100; 1000 В; измерение среднеквадратического значения синусоидального напряжения до 500 В в диапазоне частот от 20 Гц до 100 кГц. Возможность автоматического управления процессом измерения с регистрацией данных и наличие в вольтметре блока сопряжения с каналом коллективного использования обеспечивает возможность широкого применения его в СОД.

Малые постоянные напряжения измеряют вольтметром постоянного тока В2-36, который имеет диапазон измерений 0,05 мкВ — 19,999 В. Дистанционное управление запуском цикла измерения, выбор поддиапазонов измерений и цифровой выход позволяют использовать вольтметр в СОД.

Измерение медленно меняющихся напряжений, токов и малых зарядов от источников с высоким выходным сопротивлением реализует универсальный вольтметр В7-34, который обладает дистанционным управлением процесса измерения и цифровым выходом.

Цифровая информация вольтметров выдается в коде 8—4—2—1.

Широко применяют в СОД частотомеры, которые предназначены для автоматического измерения частоты электрических колебаний с высокой разрешающей способностью. Одно из основных требований к аппаратуре частотомера при включении его в состав системы сбора данных — это обеспечение возможности дистанционного управления и цифрового выхода. Подобными характеристиками обладает большинство частотомеров, выпускаемых промышленностью, например ЧЗ-55.

В системах сбора данных используют генератор импульсов Г5-79, генератор низкочастотный ГЗ-110 с программатором 321, высокочастотные генераторы Г4-141, Г4-155, Г4-156, генератор сигналов специальной формы Г6-33. Генератор импульсов Г5-79 предназначен для формирования серий прямоугольных импульсов и линейно-ступенчатых сигналов различной формы, при этом обеспечивается регулировка амплитуды, длительности, периода повторения импульсов в серии и периода повторения серий импульсов, полярности, базового смещения, временного сдвига. Схема управления генератора

обеспечивает возможность записи и хранения программы воспроизведения заданной последовательности импульсов.

Комплект генератора ГЗ-110 с программатором 321 формирует синусоидальные электрические колебания с высокой точностью и стабильностью частоты в диапазоне 0,01 Гц—2 мГц (дискретно через 0,01 Гц). Возможность программного управления позволяет применять его в системах сбора данных.

Высокочастотные генераторы сигналов Г4-14, Г4-185, Г4-156 контролируют радиоаппаратуру и другие средства связи, работающие в режимах АМ, ЧМ и, комбинированной АМ/ЧМ. Дистанционное управление работой генераторов в цифровом коде позволяет реализовать точную установку параметров генерируемого сигнала при включении их в состав СОД, причем генераторы Г4-155, 156 имеют внутреннюю оперативную память для хранения 16 наборов частот и уровней мощности.

Для исследования, настройки и испытаний радиоэлектронных устройств систем автоматического управления удобно использовать генератор Г6-33. Генератор предназначен для формирования сигналов синусоидальной, треугольной, пилообразной, пилообразно-импульсной и прямоугольной формы с переменной скважностью, высокой точностью и стабильностью по частоте. Диапазон частот 0,001—10 000 Гц, управление прибором возможно от ЭВМ, при этом программируются все параметры сигналов: частота, форма, ослабление, фазовый сдвиг.

Из краткого рассмотрения отдельных датчиков исходной информации, применяемых в СОД, видно, что параметры выхода датчиков весьма многообразны, но в самом общем виде их можно разделить на аналоговые и цифровые. Такое разделение позволяет определить основные задачи преобразования сигналов, поступающих на вход СОД, и конкретизировать схему всей системы сбора и обработки данных.

Из рассмотрения типов и конструкций датчиков следует, что характеристики выходных цепей и сигналов очень разнообразны, но с целью рационального построения системы сбора данных можно распределить параметры выхода датчиков по группам в соответствии с формой несущего сигнала табл. 1.1. Анализируя таблицу, выделяем требования по сопряжению датчика с входом канала сбора данных.

Преобразования сигналов, снимаемых с датчиков. Процесс преобразования аналоговых сигналов в цифровые, кроме процедуры преобразования, включает в себя операции обработки аналоговых устройств СОД. Операции обработки аналоговых сигналов определяются требованием обеспечения заданного качества сигнала и в отдельных случаях необходимостью проведения определенных функциональных преобразований.

Большинство датчиков СОД имеют большое выходное сопротивление и малый динамический диапазон, поэтому необходимо согласование параметров выхода датчиков с параметрами входных цепей системы обработки данных. В качестве согласующих устройств широко

Таблица 1.1

Характеристики датчика	Группы датчиков, объединенные по форме выходного сигнала					
	аналоговые			дискретные		
	Постоянное напряжение (ток)	Переменное напряжение (ток)	Цифровой код	Частота следования импульса	Относительная длительность	
1	2	3	4	5	6	
Функция преобразования	$U_{\text{вых}} = k\alpha[B]$	$u_{\text{вых}} = A \sin(\omega_0 t + \varphi_0) \times I + m\lambda$ $A = U_0[1 + m\lambda]$	$X_{\text{вых}} = k\lambda[\text{ед. кода}]$	$F_{\text{вых}} = k\alpha[\Gamma_{\text{ц}}]$	$Q_{\text{вых}} = k\alpha$	
Смещение нулевого уровня	$U_{\text{см}} = A[\text{мВ}]$		$X_{\text{см}} = x_0[\text{ед. кода}]$	$F_{\text{см}} = F_0[\Gamma_{\text{ц}}]$	$Q_{\text{см}} = Q_0$	
Чувствительность S	$\frac{dU_{\text{вых}}}{d\alpha} \left[\frac{\text{В}}{1 \text{ ф. в}} \right]$	$\frac{dA}{d\lambda} \left[\frac{\text{В}}{1 \text{ ф. в}} \right]$	$\frac{dX}{d\alpha} \left[\frac{\text{ед. кода}}{1 \text{ ф. в}} \right]$	$\frac{dF}{d\alpha} \left[\frac{\Gamma_{\text{ц}}}{1 \text{ ф. в}} \right]$	$\frac{dQ}{d\alpha} \left[\frac{1}{1 \text{ ф. в}} \right]$	
Диапазон преобразования: рабочий полный	$U_{\text{раб}}[-U_{\text{раб}}, +U_{\text{раб}}][\text{В}]$ $U_{\text{п}}[-U_{\text{п}}, +U_{\text{п}}][\text{В}]$	$u_{\text{раб}}[-u_{\text{раб}}, +u_{\text{раб}}][\text{В}]$ $u_{\text{п}}[-u_{\text{п}}, +u_{\text{п}}][\text{В}]$	$X_{\text{раб}}[-X_{\text{раб}}, +X_{\text{раб}}][\text{ед. кода}]$ $X_{\text{п}}[-X_{\text{п}}, +X_{\text{п}}][\text{ед. кода}]$	$F_{\text{раб}}[-F_{\text{раб}}, +F_{\text{раб}}][\Gamma_{\text{ц}}]$ $F_{\text{п}}[-F_{\text{п}}, +F_{\text{п}}][\Gamma_{\text{ц}}]$	$Q_{\text{раб}}[-Q_{\text{раб}}, +Q_{\text{раб}}]$ $Q_{\text{п}}[-Q_{\text{п}}, +Q_{\text{п}}]$	
Время установления τ , мкс	T	—	T	T	T	

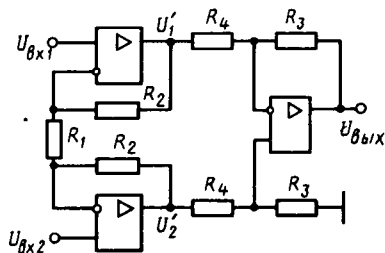
Окончание табл. 1.1

1	2	3	4	5	6
Масштаб выходного сигнала $M_{\text{вых}}$	$\frac{\alpha}{U_{\text{вых}}} \left[\frac{1 \text{ ф. в}}{B} \right]$	$\frac{\lambda}{u_{\text{вых}}} \left[\frac{1 \text{ ф. в}}{B} \right]$	$\frac{\alpha}{X_{\text{вых}}} \left[\frac{1 \text{ ф. в}}{\text{ед. кода}} \right]$	$\frac{\alpha}{F_{\text{вых}}} \left[\frac{1 \text{ ф. в}}{\Gamma_{\text{ц}}} \right]$	$\frac{\alpha}{Q_{\text{вых}}} \left[\frac{1 \text{ ф. в}}{I} \right]$
Приведенная погрешность δ , %	a	a	a	a	a
Характеристика систематических и случайных погрешностей по ГОСТ 8.009—72. Наприме- р: предел $\Delta_{\text{сд}}$ предел $\sigma_{\text{д}}(\Delta)$	b c	b c	b c	b c	b c
Выходные: емкость $C_{\text{вых}}$, пФ сопротивление $R_{\text{вых}}$, Ом	$+$ $+$	$+$ $+$	$-$ $-$	$+$ $+$	$+$ $+$
Дополнительные характеристики	$v_{\text{max}} = \frac{\Delta \alpha}{\Delta t} \left[\frac{1 \text{ ф. в}}{C} \right]$	$\omega_0 = C [\Gamma_{\text{ц}}]$	$-$	$\tau_{\text{ф}}^n, \tau_{\text{ф}}^c$	$\tau_{\text{ф}}^n, \tau_{\text{ф}}^c$

кое применение нашли операционные усилители, которые ставят на входе аналоговых устройств СОД.

Формирование схемы включения операционного усилителя для решения поставленной задачи зависит от требуемого качества сигнала. Для оценки качества усиления сигнала при условии воздействия помех хорошим критерием является определение отношения сигнал/шум. Наиболее рационально использовать схему согласующего устройства, приведенную на рис. 1.13, входное сопротивление которой $R_{вх} = R_{вх.у} (1 + K_y \beta) \approx 10^{10}$ Ом, где $\beta = R_1 / (R_1 + R_2)$.

Рис. 1.13. Схема согласующего устройства канала системы СОД.



Из выражений

$$U'_{вх1} = -U_{вх2}(R_2/R_1) + U_{вх1}(1 + R_2/R_1); \quad U'_{вх2} = -U'_{вх1}(R_2/R_1) + U_{вх2}(1 + R_2/R_1);$$

$$U'_{вх1} - U'_{вх2} = (U_{вх1} - U_{вх2}) + (U_{вх1} - U_{вх2})(2R_2/R_1);$$

$$U_{вых} = -(U'_{вх1} - U'_{вх2})(R_3/R_4),$$

а следовательно, $U_{вых} = -[(U_{вх1} - U_{вх2}) + (U_{вх1} - U_{вх2})(2R_2/R_1)] \times (R_3/R_4)$ находим коэффициент усиления дифференциального сигнала:

$$|K_{диф}| = U_{вых} / (U_{вх1} - U_{вх2}) = (1 + 2R_2/R_1)(R_3/R_4).$$

В соответствии с определением степень подавления синфазной помехи $КОСС = K_{диф} / K_{синф}$, поэтому в данной схеме можно достичь значительного ослабления синфазной составляющей сигнала-помехи. Высокое входное сопротивление, возможность получения больших коэффициентов усиления и КОСС во многих случаях делают предпочтительным выбор указанной схемы для реализации согласующего устройства системы СОД. В этом можно убедиться на следующем примере.

Пусть максимальное значение сигнала с датчика будет 10 мВ, требуемый динамический диапазон входа АЦП 10 В, синфазная помеха $U_{п.синф} = 3$ В, тогда необходимый коэффициент усиления равен 10^3 , а коэффициент ослабления синфазного сигнала

$$КОСС = \frac{1}{\delta} \frac{U_{п.синф}}{U_{вх}} = \frac{1}{0,01} \frac{3}{10 \cdot 10^{-3}}, \quad (1.9)$$

где δ — допустимая относительная приведенная погрешность от воздействия помехи (здесь 1 %).

Для реализации схемы выбираем операционный усилитель 153УД6, резисторы ОМЛТ 5 %, коэффициент передачи выходного

каскада принимаем равным единице. Для усилителя 153УД6 $\text{КОСС}_{\text{о.у.}} \approx 2 \cdot 10^4$.

Для всего согласующего устройства значение коэффициента ослабления $\text{КОСС}_{\text{СУ}} = \text{КОСС}_1 \cdot \text{КОСС}_2 \geq 3 \cdot 10^4$, где $\text{КОСС}_1 \geq 3 \cdot 10^3$; $\text{КОСС}_2 = K_{\text{диф}}/K_{\text{синф}} = 10$; $K_{\text{диф}} = 1$; $K_{\text{синф}}$ для схемы выходного каскада согласующего устройства зависит главным образом от неточности изготовления резисторов R_3, R_4 ; для выбранной схемы $K_{\text{синф}} \approx 26_7/100 = 2 \cdot 5/100 = 0,1$.

Для входного каскада $K_{\text{синф}} = 1$, $K_{\text{диф}} = 3 \cdot 10^3$, $R_1 = 1$ кОм, тогда $R_2 = 1,5$ мОм. Принимаем $R_3 = R_4 = 15$ кОм. Реальные значения коэффициентов ослабления несколько меньше:

$$\text{КОСС}_1^* = \text{КОСС}_1 \cdot \text{КОСС}_{\text{о.у.}} / (\text{КОСС}_1^* + \text{КОСС}_{\text{о.у.}}) = 2,7 \cdot 10^3;$$

$$\text{КОСС}_2^* = \text{КОСС}_2 \cdot \text{КОСС}_{\text{о.у.}} / (\text{КОСС}_2^* + \text{КОСС}_{\text{о.у.}}) = 9,9;$$

$$\text{КОСС}_{\text{с.у.}}^* = \text{КОСС}_1^* \cdot \text{КОСС}_2^* = 2,67 \cdot 10^4.$$

Функциональные преобразования аналогового сигнала проводят с целью введения в структуру датчика блоков для воспроизведения определенных математических операций и повышения его точностных параметров. Поясним сказанное на примерах. Измерение скорости потока жидкости или газа основано на применении датчиков перепада давления в двух точках потока P . В этом случае скорость потока $v = \sqrt{2g\Delta P}$.

Принцип действия расходомеров жидкости в открытых каналах основан на измерении высоты H потока в канале. Расход жидкости $V = L \sqrt{2gH} \sqrt{H}$, где L — ширина потока; $L \sqrt{2g} = \text{const}$.

Измерение расхода массы газа базируется на использовании датчиков: давления P , перепада давлений ΔP , температуры t . Расход газа $m = K \sqrt{P \Delta P / t}$.

В первом примере для получения результата в структуру датчика необходимо ввести блок воспроизведения операции извлечения корня, во втором — блок воспроизведения операции извлечения корня и умножения, в третьем — блок воспроизведения операций извлечения корня, умножения и деления. Задачи воспроизведения математических операций встречаются достаточно часто при формировании структуры различных датчиков и являются первой причиной, определяющей выбор вычислительных блоков обработки аналоговых сигналов в структуре датчика.

Вторая причина введения блоков вычислений в структурную схему датчиков вызвана отклонением его основной характеристики от идеальной. Например, термопары, мостовые датчики часто имеют характеристики, значительно отличающиеся от линейных. Это вызывает необходимость коррекции сигнала датчика, которая имеет нелинейный характер. Хорошие результаты линеаризации характеристики термопары могут быть получены на основе выражения

$$U'_{\text{вых}} = K_1 U_{\text{вых}} - K_2 U_{\text{вых}}^2,$$

где $U'_{\text{вых}}$ — линеаризованное напряжение датчика, пропорциональ-

ное измеренной температуре; $U_{\text{вых}}$ — выходное напряжение датчика; K_1, K_2 — расчетные коэффициенты.

Для реализации схемы датчика по приведенному выражению необходим блок воспроизведения операции умножения.

Техническая реализация задач, подобных приведенным в примерах, основана либо на воспроизведении математических зависимостей путем обработки исходной информации в МП или ЭВМ, либо включением в каналы обработки данных дополнительных элементов для преобразования аналоговых сигналов, например АМП или АЦМДУ. Выбор того или иного способа определяется исходя из условия удовлетворения требованиям по производительности системы обработки данных и ее точностных характеристик.

Сглаживание сигналов, снимаемых с датчиков. Дальнейшее повышение качества сигнала связано с операцией преобразования аналогового сигнала — фильтрацией, которая реализуется на основе активных фильтров с операционным усилителем. Операционный усилитель в схеме активного фильтра необходим для реализации полюсов в левой полуплоскости при использовании в качестве пассивных элементов только резисторов и конденсаторов. Операционный усилитель дает возможность применять резисторы и конденсаторы небольших номиналов даже при низких частотах (до 10^{-2} Гц).

Функциональное назначение активного фильтра СОД — подавление помех на выходе согласующего устройства; при этом предполагается, что полезный сигнал датчика и сигнал-помеха имеют разный диапазон частот. Диапазон частот, в котором полезный сигнал проходит через фильтр, называют *полосой пропускания*, а диапазон частот, в котором подавляется сигнал-помеха, — *полосой подавления*. Между полосой пропускания и полосой подавления реальных фильтров лежит область перехода, которая характеризуется частотой среза:

$$f_{\text{ср}} = \frac{1}{2} \frac{B_{\text{с.у}}(C/\text{Ш})_{\text{с.у}}}{(C/\text{Ш})_{\text{ф}}},$$

где $B_{\text{с.у}}$ — полоса пропускания согласующего устройства; $(C/\text{Ш})_{\text{ф}}$, $(C/\text{Ш})_{\text{с.у}} = (C/\text{Ш})_{\text{д}} \text{КОСС}_{\text{с.у}}^2$ — сигнал/шум на выходах фильтра и согласующего устройства [$(C/\text{Ш})_{\text{д}}$ — характеристика качества сигнала на выходе датчика, $\text{КОСС}_{\text{с.у}}$ — коэффициент ослабления синфазного сигнала, который обеспечивается согласующим устройством].

По соотношению полосы пропускания и полосы подавления фильтры разделяют на фильтры низких (ФНЧ) и высоких (ФВЧ) частот, полосно-пропускающие фильтры (ПФ), полосно-задерживающие фильтры.

Требуемые частотные характеристики являются основой синтеза фильтра. На первом этапе синтеза решают задачу аппроксимации, которая состоит в получении некоторой функции частоты $H(j\omega)$. Модуль этой функции должен удовлетворять требованиям АЧХ, а аргумент — с допустимыми отклонениями воспроизводить необходимый ФЧХ.

В общем случае относительное затухание

$$a = 20 \lg \frac{H_0}{|H(j\omega)|} = 10 \lg [1 + \varepsilon^2 \varphi(j\omega)], \quad (1.10)$$

где H_0 — коэффициент передачи фильтра в полосе пропускания; ε^2 — коэффициент, характеризующий степень постоянства затухания; $\varphi(j\omega)$ — функция фильтрации, для которой желательны значения, близкие к нулю в полосе пропускания и как можно большие в полосе подавления.

Известные из практики способы определения функции фильтрации $\varphi(j\omega)$, а следовательно, и функции передачи $H(p)$ основаны на аппроксимации полиномами Чебышева, Баттерворта и т. д.

Полиномы Чебышева в тригонометрической форме имеют вид $T_n(x) = \cos(n \arccos x)$ при $n=1$, $T_1(x) = x$ при $n=2$, $T_2(x) = 2x^2 - 1$ и т. д. Для определения полиномов с произвольными n можно воспользоваться соотношением $T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$.

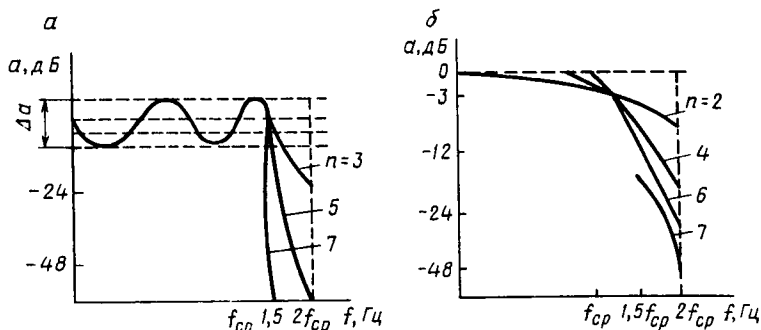


Рис. 1.14. АЧХ фильтра Чебышева (а) и фильтра Баттерворта (б).

Функция фильтрации при аппроксимации полиномами Чебышева имеет вид $a = 10 \lg [1 + \varepsilon^2 T_n^2(\Omega)]$, где $\Omega = \omega / \omega_1$ — нормированная частота; ω_1 — частота среза. Минимальное затухание a в полосе пропускания равно нулю, а максимальное $a = 10 \lg(1 + \varepsilon^2)$. В полосе пропускания неравномерность затухания $\Delta a = 10 \lg(1 + \varepsilon^2)$, откуда $\varepsilon^2 = 10^{0.1 \Delta a} - 1$.

Вид АЧХ фильтров по полиномам Чебышева приведен на рис. 1.14, а.

Полином Баттерворта имеет вид

$$B_n = \Omega^n.$$

В соответствии с (1.10) получим

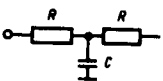
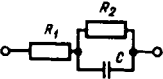
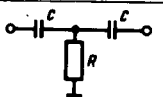
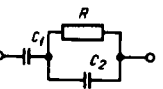
$$a = 10 \lg(1 + \varepsilon^2 \Omega^{2n}).$$

Вид характеристик относительно затухания фильтров Баттерворта для $\varepsilon^2 = 1$ и различных значений n приведен на рис. 1.14, б.

Способ реализации активных фильтров на основе операционных усилителей базируется на возможности воспроизведения различных передаточных функций усилителем, охваченным последовательной, параллельной и комбинированной отрицательной обратной связью. Наиболее часто для организации обратных связей при воспроизведении передаточных функций используют пассивные RC -четырёхполюсники для реализации параллельной обратной связи, пассивные RC -цепи, воспроизводящие последовательную связь, пассивные RC -цепи для организации параллельной многопетлевой обратной связи. Хорошими характеристиками обладают фильтры, построенные на основе интеграторов. Первый способ реализации активных фильтров достаточно прост для применения.

В табл. 1.2 представлены схемы часто применяемых пассивных цепей и формулы расчета их сопротивлений. Из таблицы подбирают

Т а б л и ц а 1.2

Схема пассивных цепей	Сопротивление z_k	Соотношения
	$A(1 + T_p)$	$A = 2R; T = \frac{RC}{2}$
	$A \left(\frac{1 + \theta T_p}{1 + T_p} \right); \theta < 1$	$A = R_1 + R_2;$ $T = R_2 C;$ $\theta = R_1 / (R_1 + R_2)$
	$\frac{1}{Bp}(1 + T_p)$	$B = C; T = RC$
	$(1/B) \frac{1 + T_p}{T_p}$	$B = C/2; T = 2RC$
	$[1/(Bp)] \frac{1 + T_p}{1 + \theta T_p}$ $\theta < 1$	$B = C_1;$ $T = R(C_1 + C_2);$ $\theta = C_2 / (C_1 + C_2)$

цепи с требуемым аналитическим выражением, включение которых на вход и в обратную связь операционного усилителя реализует требуемую передаточную функцию, причем

$$H(p) = -z(p)_{oc} / z(p)_{вх},$$

где $z(p)_{oc}$, $z(p)_{вх}$ — аналитические выражения комплексных сопротивлений в цепи обратной связи и входной цепи соответственно.

Данный способ привлекает своей простотой, но имеет ограниченную область применения при моделировании каналов сбора данных.

Лучшими эксплуатационными характеристиками обладают

фильтры на пассивных RC -четыреполюсниках, построенные на основе реализации выражения

$$\frac{U_{\text{вых}}}{U_{\text{вх}}} = - \frac{y_{21a}(y_{22b} + y_{11d}) - y_{21b}(y_{22a} + y_{11c})}{y_{12c}(y_{22b} + y_{11d}) - y_{21d}(y_{22a} + y_{11c})},$$

которому соответствует структура, изображенная на рис. 1.15, где y_{ij} с индексами a, b, c, d — проводимости соответствующих четырехполюсников.

Из всех возможных вариантов реализации выражения рассмотрим наиболее простой способ построения фильтров на операционном усилителе с параллельной обратной связью, т. е. при условии $y_{21b} = y_{12d} = 0$, тогда

$$U_{\text{вых}}/U_{\text{вх}} = -y_{21d}/y_{12c}. \quad (1.11)$$

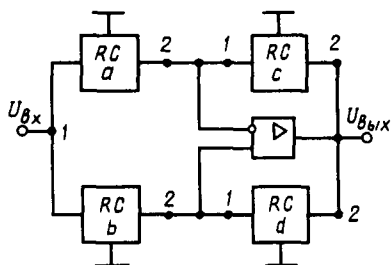


Рис. 1.15. Структура реализации активных фильтров на RC -четыреполюсниках.

На рис. 1.16, a приведена структура воспроизведения выражения (1.11), представляющая практический интерес. Результат решения матрицы проводимостей имеет вид

$$U_{\text{вых}}/U_{\text{вх}} = -y_1 y_3 / (y_1 y_5 + y_4 y_3 + y_5 y_4 + y_5 y_2 + y_5 y_3).$$

На рис. 1.16, b показана схема фильтра на усилителе с последовательной обратной связью. Для определения передаточной функции фильтра можно воспользоваться первым законом Кирхгофа. Полагая, что входной ток операционного усилителя равен нулю, запишем

$$(U_{\text{вх}} - U_a)y_1 = (U_a - U_{\text{вых}})y_2 + (U_a - U_{\text{вых}})y_3; \quad (U_a - U_{\text{вых}})y_3 = U_{\text{вых}}y_4. \quad (1.12)$$

После преобразования

$$U_a = U_{\text{вых}}(y_3 + y_4)/y_3. \quad (1.13)$$

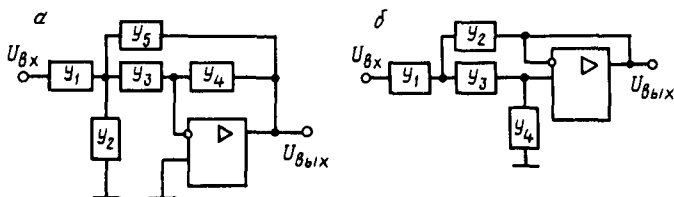


Рис. 1.16. Структура активного RC -фильтра на усилителе с параллельной (a) и последовательной (b) отрицательной обратной связью.

Решая совместно (1.12) и (1.13), получим

$$U_{\text{вых}}/U_{\text{вх}} = y_1 y_3 / (y_1 y_3 + y_1 y_4 + y_2 y_4 + y_3 y_4). \quad (1.14)$$

На рис. 1.17, а показана схема активного фильтра с многопетлевой обратной связью, реализующего передаточную функцию с парой

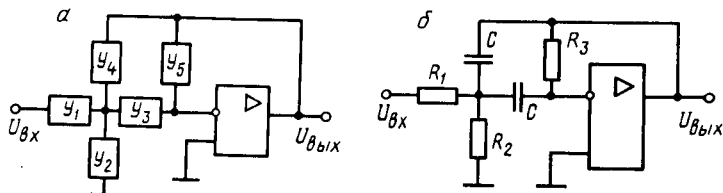


Рис. 1.17. Структура активного фильтра (а) и RC-фильтра (б) с многопетлевой отрицательной обратной связью.

комплексно-сопряженных полюсов и нулями в начале координат или на бесконечности. Передаточную функцию определяем, как в предыдущем примере. Результат решения имеет вид

$$U_{\text{вых}}/U_{\text{вх}} = y_1 y_3 / (y_1 y_3 + y_2 y_3 + y_3 y_4 + y_3 y_5). \quad (1.15)$$

Для рационального подхода к проектированию фильтров необходимо:

1. Определить способ аппроксимации идеализированных АФЧХ исходя из особенностей конкретного способа и практически допустимых значений некоторого критерия близости. Особенности аппроксимации являются следствием свойств функции фильтрации. Анализ полиномов Чебышева показывает: АЧХ фильтра всегда имеет нормируемую неравномерность в полосе пропускания и резкий спад после частоты среза, увеличение неравномерности в полосе пропускания приводит к росту затухания в области перехода. ФЧХ нелинейна. Использование фильтров Чебышева в СОД ограничены значением допустимого отклонения коэффициента передачи в полосе пропускания.

Анализ полиномов Баттерворта показывает, что они обладают максимально плоской АЧХ в полосе пропускания и умеренным спадом в переходной области. Практическое постоянство коэффициента передачи в полосе пропускания делает предпочтительным использование данного фильтра в СОД.

2. Выбрать по графикам АЧХ подходящую характеристику фильтра и определить число полюсов, требующееся для получения необходимого затухания.

3. Из схем, представленных на рис. 1.16, 1.17, выбрать соответствующую схему фильтра, причем на начальном этапе синтеза лучше рассмотреть несколько вариантов.

4. Выполнить расчет элементов фильтра, пользуясь соответствующей

щими графиками и соотношениями, рассмотренными в литературе по синтезу фильтров.

Особенностью расчета полосовых фильтров является учет добротности $Q=f_0/\Delta f$, где f_0 — средняя частота; Δf — полоса частот на уровне 3 дБ.

Полосовые фильтры достаточно просто реализуются на операционном усилителе с многопетлевой обратной связью (рис. 1.17, б). Для этой схемы выражения, определяющие параметры пассивных элементов, имеют вид: $a_c=2\pi f_0 C$; $R_1=Q/a_c$; $R_2=1/[(2Q-1Q)a_c]$; $R_3=2Q/a_c$.

Пример 1.1. Пусть необходимо рассчитать параметры фильтра с добротностью $Q=10$ на частоте $f_0=400$ Гц.

Выберем конденсаторы $K73-16 \pm 5\% - 0,01$ мкФ, тогда $a_c=6,28 \cdot 400 \text{ Гц} \times \times 0,01 \text{ мкФ} = 25 \cdot 10^{-6} \text{ См}$; $R_1=0,4 \text{ МОм}$; $R_2=1 \text{ кОм}$; $R_3=0,8 \text{ МОм}$.

Пользуясь соотношением (1.15), можно определить передаточную функцию фильтра и провести анализ его АЧХ и ФЧХ.

Способ реализации фильтров на основе интеграторов базируется на решении дифференциального уравнения, заданного передаточной функцией, методом понижения порядка производной.

Пусть необходимо реализовать фильтр с передаточной функцией вида $W(p)=K/(T_1^2 p^2 + 2T_1 p + 1)$. В этом случае можно записать

$$U_{\text{вых}}(p) = \frac{K}{T_1^2 p^2 + 2T_1 p + 1} U_{\text{вх}}(p).$$

После преобразования имеем

$$U_{\text{вых}}(p) = -\frac{1}{T_1 p} \left\{ 2U_{\text{вых}}(p) - \frac{a}{T_1 p} \left[\frac{U_{\text{вх}}(p) - U_{\text{вых}}(p)}{a} \right] \right\}.$$

Структура решения приведена на рис. 1.18, а, а принципиальная схема — на рис. 1.18, б, напряжение в точке а

$$U_a = U_{\text{вых}}/(pR_2C_1) - U_{\text{вх}}/(pR_1C_1), \text{ на выходе } U_{\text{вых}} = -U_a \times \times (R_4/R_3)/(1 + pR_4C_2).$$

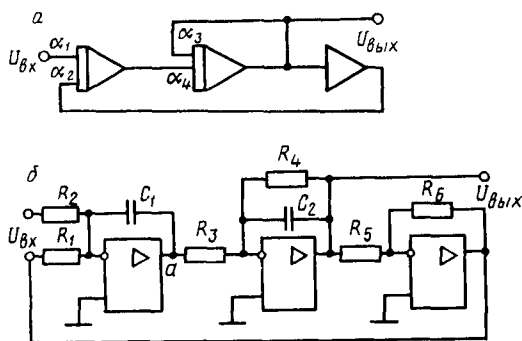


Рис. 1.18. Структура (а) и схема (б) RC-фильтра на интеграторах.

Результат решения имеет вид

$$U_{\text{вых}}/U_{\text{вх}} = (R_2/R_1) / (p^2 R_2 C_1 R_3 C_2 + p(R_3/R_4) R_2 C_1 + 1).$$

Схемы фильтров на основе интеграторов обладают малым смещением нулевого уровня, возможностью получения больших добротностей, простотой выбора интегрирующих конденсаторов.

Восстановление аналоговых сигналов и выбор частоты опроса датчиков. В многоканальных системах обработки данных возможны два способа восстановления аналоговых сигналов на выходах системы: 1) использование блока ЦАП в каждом канале, 2) применение одного блока ЦАП с включением в каждый канал схемы выборки — хранения цифрового и аналогового коммутаторов ЦК и АК. Способы реализации блоков схемы рассмотрены в § 1.3—1.7.

Точностные и функциональные характеристики параллельной схемы (рис. 1.19, а) более высокие по сравнению с последовательно-параллельным (рис. 1.19, б) вариантом построения схемы восстановления. Так как погрешности дрейфа и смещения нулевого уровня минимальны, информация на выходе может сохраняться сколь угодно долго, длительность переходных процессов при смене информации минимальна. Однако во многих случаях для удовлетворения требований по минимальности габаритно-весовой характеристики необходимо выбирать второй вариант схемы восстановления.

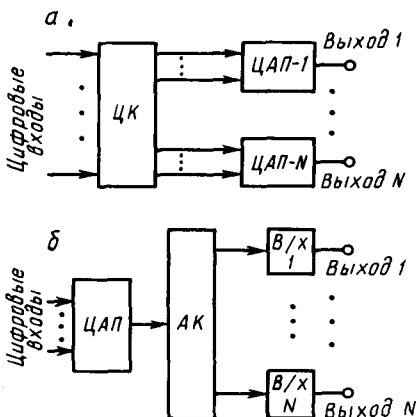


Рис. 1.19. Структуры восстановления аналоговых сигналов.

Аналоговый сигнал на выходе схемы восстановления изменяется ступенчато вследствие дискретной обработки данных в МП или ЭВМ; это вызывает появление гармоник в аналоговом сигнале на выходе, которых не было во входных сигналах. Для сглаживания ступенек в выходном сигнале (иначе — для подавления высокочастотных гармоник выходного сигнала на выходах схем восстановления) включают ФНЧ. Частота среза ФНЧ должна находиться между спектром входного сигнала и низкочастотной составляющей спектра выходной выборки. Как правило, достаточно знать высокочастотную составляющую входного аналогового сигнала, чтобы решить задачу синтеза выходного фильтра, пользуясь аналитическими выражениями (1.10) — (1.15). Отметим, что этот фильтр должен выбираться с крутым спадом $N=5-7$.

Числовое значение высокочастотной составляющей входного сиг-

нала необходимо также при определении минимально допустимой частоты опроса датчиков и частоты среза $f_{\text{ср.вх}}$ входного фильтра канала сбора данных. По теореме Котельникова, для передачи любой функции времени с максимальной частотой спектра f_{max} достаточно передавать ее отдельные значения через интервалы времени $\Delta t \leq \leq 1/(2f_{\text{max}})$. Значит, при выборе частоты опроса датчика необходимо соблюдать условия $f_{\text{опр}} \geq 2f_{\text{max}}$ и $f_{\text{ср.вх}} = f_{\text{max}}$. Значение f_{max} можно определить теоретически или практически, используя для измерений осциллограф. Если аналоговый сигнал датчика имеет вид синусоидальной функции времени, то $f_{\text{max}} = 1/T$, а если имеет вид периодической функции времени, то $f_{\text{max}} = (10-20)/T_{\text{осн}}$ (где $T_{\text{осн}}$ — основной период для медленно изменяющейся (постоянной) функции времени), $f_{\text{max}} = 1/v$ (где v — максимальная скорость изменения сигнала).

Реальный сигнал опроса датчика имеет определенный спектр частот, поэтому с целью исключения взаимодействия отдельных составляющих спектров сигнала датчика и сигнала опроса необходимо, чтобы $f_{\text{опр}} > f_{\text{ср.вх}}$. Принимают $f_{\text{опр}} = (3-10)f_{\text{ср.вх}}$; $f_{\text{опр min}}$ выбирают в зависимости от крутизны спада АЧХ входного фильтра; при $n=10$ $f_{\text{опр}} = 3f_{\text{ср.вх}}$; при $n=6$ $f_{\text{опр}} = 4f_{\text{ср.вх}}$; при $n=2$ $f_{\text{опр}} = 8f_{\text{ср.вх}}$.

1.3. Интерфейсы

Линии. Подмагистралы. Магистралы. Сопряжения узлов системы СОД предполагают наличие комплекса устройств и определенных правил, которые обеспечивают те или иные виды совместимости: по форме представления информации, электрическую, временную, пространственную, конструктивную. Это означает, что свойства сигналов датчиков первичной информации должны быть приведены в соответствие с нормированными характеристиками входа микро-ЭВМ.

Возможны два основных подхода к организации взаимодействия элементов системы СОД и построению материальных связей между ними: 1) жесткая унификация и стандартизация входных и выходных параметров элементов системы СОД; 2) использование специализированных функциональных блоков, обладающих в той или иной мере адаптивными характеристиками по входам-выходам.

В настоящее время производится и эксплуатируется много различных средств измерений с разнообразными характеристиками по входам-выходам, исключить которые из обращения практически невозможно. Если применять для построения системы СОД эти устройства, то возможен только второй подход к организации сопряжения элементов системы СОД. Однако часто сочетают и оба подхода: первый — для сопряжения вновь разрабатываемых средств измерений, а также для организации внутренних связей системы СОД, а второй — для объединения всех элементов системы СОД.

В основу разработки средств сопряжения положен принцип информационной совместимости элементов системы СОД, цель кото-

рого — создание коммуникационных соединений для единообразной передачи сигналов различного вида между функционально связанными элементами СОД.

Электрические сигналы по информативным параметрам делят на: 1) аналоговые (непрерывные), которые могут принимать любое значение на некотором интервале и являются непрерывной функцией времени; 2) дискретные, принимающие конечное множество значений. Дискретные сигналы в свою очередь могут быть двоичными, которые отображают потенциалами низкого и высокого уровней, и цифровыми: слова некоторого алфавита, символы алфавита кодированы двоичными сигналами.

Одним из многих правил, регламентирующих организацию сопряжения, являются мероприятия, направленные на реализацию передачи на расстояние электрических сигналов. В рамках этих правил для двоичных сигналов устанавливают потенциалы низкого и высокого уровней, коэффициенты нагрузки входа и выхода, динамические параметры сигналов, допустимые расстояния передачи. Для аналоговых сигналов оговаривают диапазон напряжений, полные сопротивления цепей и допустимое время установления сигналов.

На рис. 1.20, а представлена обобщенная структура канала передачи сигналов на расстояние, где Пд, Пр — передатчик и приемник сообщений, А — абонент, ИС — источник сообщений, КП — канал передачи.

Совокупность линейных, коммутирующих и других технических средств, обеспечивающих прием-передачу сигналов, образует канал передачи.

На рис. 1.20, б изображена структура элементарного канала передачи сигналов, где F' , F'' — оконечные формирователи сигналов на передающей

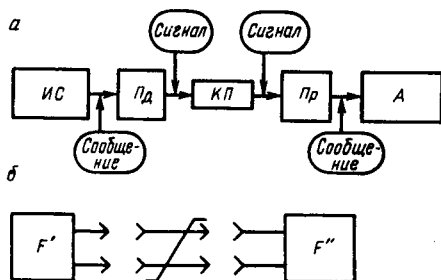


Рис. 1.20. Структура канала передачи сигналов.

и приемной сторонах соответственно. Электрические цепи, обеспечивающие передачу сигнала между двумя оконечными формирователями, образуют линию связи. Совокупность нескольких линий связи, объединенных для передачи цифровых сигналов, образует подмагистраль. Совокупность подмагистралей, обеспечивающих взаимную прием-передачу цифровых сигналов между всеми функциональными элементами системы СОД с цифровым входом-выходом, образует магистраль.

Линейные средства сопряжения (провода, экранированные провода, кабели, высокочастотные коаксиальные кабели, сигнальные кабели с витыми центральными парами) осуществляют материальную связь блоков СОД.

Выбор тех или иных линейных средств определяется длиной линии связи, параметрами передаваемого сигнала, наличием помех различного вида. Проводные линии связи, как правило, имеют максимальную длину, не превышающую 3 м, кабельные — до 300 м. Проводящие экраны кабелей значительно снижают электромагнитные наводки (на 40 дБ), экранированные кабели с витой парой проводов — воздействие емкостных и индуктивных наводок (на 40 дБ).

По способу соединения проводников нулевого потенциала (проводники «земля») линии связи можно организовать между гальванически связанными и развязанными устройствами. Линия связи первого вида состоит из общей и сигнальной цепей. Общие цепи образуются специальными проводниками и экранирующими оплетками кабелей; сигнальные цепи организуются на центральных проводниках кабелей, экранированных проводниках или отдельными проводами.

Гальваническая развязка устройств системы СОД обеспечивается входами-выходами формирователей каналов связи; позволяет исключить уравнильные токи по общим проводникам и тем самым уменьшить взаимовлияние отдельных устройств, обладающих разным энергопотреблением. Гальваническая развязка реализуется либо на одном конце линии связи, либо на обоих концах и обеспечивается трансформатором оконечного формирователя или оптронами.

Схемы оконечных формирователей сигналов на входе и выходе канала связи с гальванической развязкой на трансформаторах приведены соответственно на рис. 1.21, а, б, а на рис. 1.21, в — как на входе, так и на выходе канала.

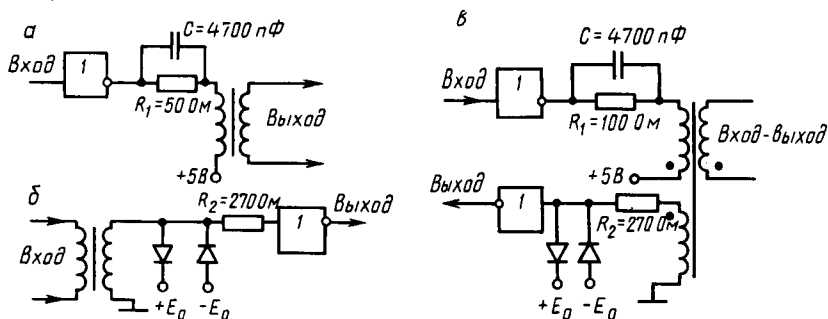


Рис. 1.21. Примеры схем оконечных формирователей сигналов в канале связи.

Сопротивления резисторов R_1 , R_2 выбираются из условия согласования волнового сопротивления кабеля и входного сопротивления оконечного формирователя на основе метода Бержерона.

Гальваническая развязка на основе оптронов иллюстрируется схемой на рис. 1.22. Сопротивление резистора R_1 выбирается из условия обеспечения номинального тока включения оптопары при выбранном источнике питания. Резистор R_2 и преобразователь уровней 564ПУ4 необходимы для согласования выхода линии с входами ТТЛ. Если использовать оптронные переключатели-инверторы

249ЛП1, то согласование выходного напряжения линии с ТТЛ реализуется без дополнительных элементов.

На рис. 1.23, а показаны подмагистраль с гальванической развязкой на основе трансформаторов (каждая линия двухпроводная и может быть витой парой) и точки «заземления» экранов витых пар и всей подмагистрали, а на рис. 1.23, б — подмагистраль

Рис. 1.22. Пример схемы реализации гальванической развязки выхода канала связи на основе оптрона.

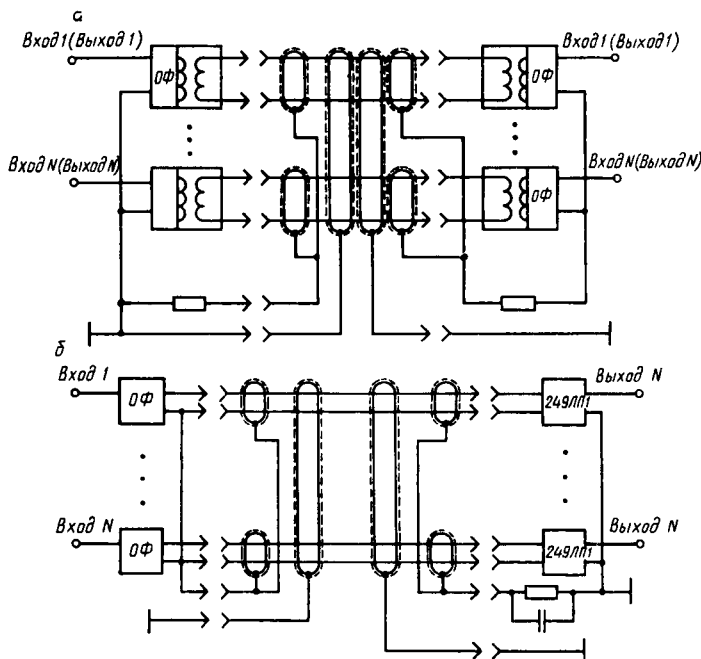
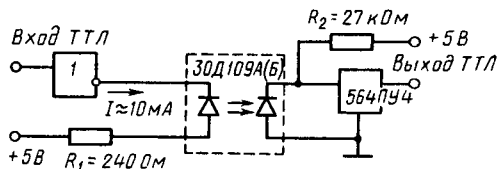


Рис. 1.23. Примеры схем подмагистрали передачи сигналов.

с гальванической развязкой на одном конце, которую обеспечивают оптронные пары и точки «заземления» экранов, где ОФ — оконечный формирователь.

Для передачи информации по подмагистралям, организующим связи в процессоре и в системе обмена информацией между процессором и элементами СОД, используют двунаправленные формирователи, например серии К589, которые представляют собой четырехканальные цифровые коммутаторы: К589АП16 — формирователь без инверсии (рис. 1.24, а); К589АП26 — формирователь с инверсией (рис. 1.24, б). Передача информации с входов на выходы осуществ-

вляется сигналами CS и DCE. Если $CS=0$ и $DCE=0$, то открыта передача сигналов с входов D на выходы DB. Если $CS=0$ и $DCE=1$, то открыта передача с входов DB на выходы DO. Если $CS=1$, то формирователь находится в выключенном состоянии (третье состояние), причем выходы имеют высокое выходное сопротивление.

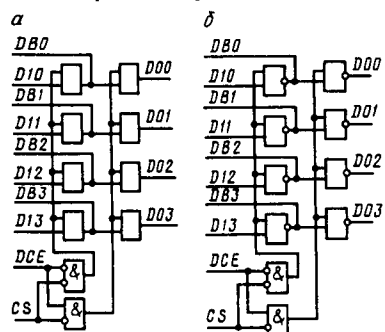


Рис. 1.24. Примеры схемы формирователей серий K589АП16 и K589АП26.

Интерфейсы. Одним из центральных, определяющих моментов в проектировании МП ИУВС является выбор совокупности унифицированных аппаратных, программных и конструктивных средств, необходимых для реализации алгоритмов взаимодействия различных функциональных устройств ИУВС; иначе говоря, выбор *интерфейса* или в общем случае совокупности интерфейсов для конкретной ИУВС. Как правило, МП ИУВС строятся с использованием нескольких интерфейсов, например интерфейса модулей МП (собственно процессора, ОЗУ, ПЗУ, УВВ), интерфейса подключения периферийных устройств (датчиков информации, объектов управления, стандартных периферийных устройств, устройств регистрации), интерфейса микропроцессоров в *т*МП ИУВС.

Стандартизации в интерфейсе обычно подлежат: форматы передаваемой информации; команды и состояния; состав и типы линий связи; алгоритм функционирования; передающие и приемные электронные схемы; параметры сигналов и требования к ним; конструктивные решения.

Уточняя требования к интерфейсу при его выборе, исходят из необходимости обеспечения заданного быстродействия ИУВС в целом; области применения (сравните внутримашинный интерфейс процессор — память МП при длине линий связи в доли и единицы метра и сетевой интерфейс микропроцессоров *т*МП ИУВС распределенного типа, отдельные МП которой могут размещаться на расстояниях в десятки и сотни метров один от другого); необходимости минимизации стоимости; обеспечения простоты схемных и конструктивных решений; стандартизации процедуры обмена информацией независимо от скорости работы устройств; учета возможности и простоты наращиваемости системы; требований диагностики системы и т. п.

К основным характеристикам интерфейса относят: функциональное назначение; тип организации связей; принцип обмена информацией; способ обмена; режим обмена; количество линий; число линий для передачи данных; количество адресов; количество команд; быстродействие; длину линий связи; число подключаемых устройств; тип линии связи.

По функциональному назначению интерфейсы можно

подразделить на магистральные (внутримашинные), внешние интерфейсы периферийных устройств, системные (интерфейсы локальных сетей).

По типу организации связей интерфейсы подразделяют на магистральные, радиальные, древовидные, радиально-магистральные.

По принципу обмена информацией — с параллельной, последовательной и параллельно-последовательной передачей информации.

По режиму обмена информацией — с симплексным, полудуплексным, дуплексным и мультиплексным режимами обмена. Для случая связи двух абонентов в *симплексном режиме* лишь один из двух абонентов может инициировать в любой момент времени передачу информации по интерфейсу (рис. 1.25, а). Для случая связи двух абонентов в *полудуплексном режиме* любой абонент

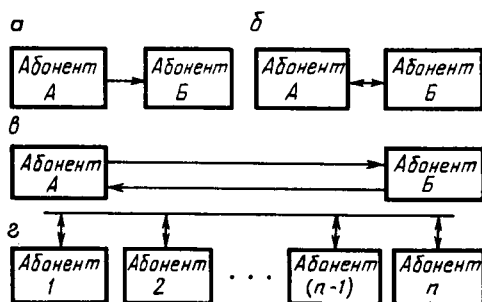


Рис. 1.25. Схемы симплексного (а), полудуплексного (б), дуплексного (в) и мультиплексного (г) режимов обмена информацией.

может начать передачу информации другому, если линия связи интерфейса при этом оказывается свободной (рис. 1.25, б). Для случая связи двух абонентов в *дуплексном режиме* каждый абонент может начать передачу информации другому в произвольный момент времени (рис. 1.25, в). В случае связи нескольких абонентов в *мультиплексном режиме* в каждый момент времени связь может быть осуществлена между парой абонентов в любом, но единственном направлении от одного из абонентов к другому (рис. 1.25, г).

По способу передачи информации во времени различают интерфейсы с синхронной передачей данных (с постоянной временной привязкой в цикле сбора информации) и с асинхронной (без постоянной временной привязки к определенному временному интервалу цикла сбора).

При синхронной передаче данных синхронизирующие сигналы МП задают определенный временной интервал, в течение которого считывается информация с одного датчика первичной информации. Временной интервал в данном случае определяется наибольшими временными задержками в системе передачи данных и максимальным временем преобразования измеренного сигнала в цифровой сигнал.

Таблица 1.3

Технические характеристики	Интерфейс					
	Microbus	Z-bus	Unibus	Q-bus	Multibus	Versabus
Число линий:						
общее	37	—	56	43	86*	260*
информационных	8	8 совмещенные	16	16 совмещенные	16	8, 16, 32
адресных	16	—	18	—	20	35
управления	13	9	20	—	11	—
Скорость передачи информации, Кбод	10^3	—	$(1-2)10^3$	800	$(5-10)10^3$	До $20 \cdot 10^4$
Длина линий, м	В пределах печатной платы	—	15	15	—	—
Число подключаемых абонентов	≤ 10	—	20	15	—	—

* Это общее число контактов в разъеме (разъемах), принятом для интерфейса.

Асинхронная передача данных характеризуется наличием управляющих сигналов: ГОТОВНОСТЬ к ОБМЕНУ, вырабатываемых датчиком исходной информации, и НАЧАЛО ОБМЕНА, КОНЕЦ ОБМЕНА, КОНТРОЛЬ ОБМЕНА, вырабатываемых микропроцессором при необходимости сбора информации. При такой организации обмена автоматически устанавливается рациональное соотношение между скоростью передачи данных и временем задержки сигналов в каналах связи. Сигнал КОНТРОЛЬ ОБМЕНА является результатом решения задачи обнаружения сбоя в процессе передачи и обеспечивает распознавание сигнала помехи в канале передачи.

Магистральные интерфейсы. Для обеспечения высоких скоростей обмена информацией в качестве магистрального (внутри-машинного) интерфейса в зарубежных и отечественных МП ИУВС распространены асинхронные мультиплексные интерфейсы с параллельным способом передачи информации: 8-разрядные интерфейсы Microbus, шина iSBX, Z-bus; 16-разрядные интерфейсы ОШ НМ МПК по ВТ 34-80 (Unibus) [3, 4], Multibus (IEEE-P796-bus), интерфейс микроЭВМ «Электроника-60» (Q-bus); 32-разрядный интерфейс Versabus [5].

Основные технические характеристики некоторых магистральных параллельных асинхронных интерфейсов с мультиплексным режимом обмена приведены в табл. 1.3.

Наиболее широко в отечественных разработках среди 16-разрядных интерфейсов применяют интерфейсы типа Unibus (ОШ НМ МПК по ВТ 34—80) и Q-bus (МПШ ОСТ 11.305.903—80). В табл. 1.4 и 1.5 приведены перечень и назначение информационных линий, временные диаграммы, поясняющие циклы чтения, записи, захвата магистрали и передачи вектора прерывания этих интерфейсов, подробное описание которых дано в [3,4].

Таблица 1.4

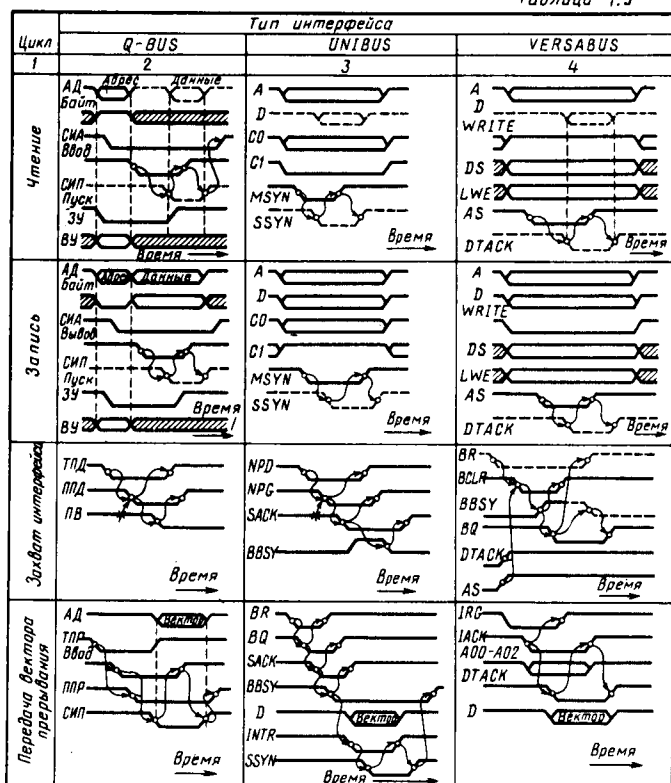
Линии	Интерфейс		
	Q-bus	Unibus	Versabus
Адреса	Совмещенные ДА00-ДА15	A00—A17	A00—A24 основные A25—A31 дополни- тельные DSO, DSI — выбор байта LWE — выбор двой- ного слова 2 линии четности
Данных	ВУ — выбор внеш- него устройства	D00—D15	D00—D15 основные D16—D31 дополни- тельные 4 линии четности
Стробирова- ния адреса и данных	СИА — синхрониза- ция активного СИП — синхрониза- ция пассивного ВВОД — ввод ин- формации ВЫВОД — вывод информации БАЙТ — запись/байт ПУСК ЗУ — пуск ЗУ	MSYN (схз) — син- хронизация задатчика SSYN (схи) — син- хронизация исполните- ля C1, C0 — код опе- рации	IOEN — деблоки- ровка AS — адресный строб DTACK — подтвер- ждение WRITE — тип цикла EXTDADR — расши- рение адреса RAMDIS — блоки- ровка ОЗУ ROMDIS — блоки- ровка ПЗУ
Захвата ма- гистрала (пря- мой доступ)	ТПД — требование прямого доступа ППД — предостав- ление прямого доступа ПВ — подтвержде- ние выбора	NPR — запрос пря- мого доступа NPG — разрешение прямого доступа SACK — подтверж- дение выборки BBSY — занято	BR0—BR4 — за- прос прямого доступа BQ0—BQ4 — раз- решение прямого до- ступа BBSY — занято BCLR — очистить шину
Прерывания	ТПР — требование ППР — предостав- ление ПРТ — прерывание от внешнего устрой- ства	BR4—BR7 — запро- сы BG4—BG7 — раз- решения INTR — передача вектора прерывания	IRQ1—IRQ7 — за- прос JACK — разреше- ние
Дополнитель- ные	ПОСТ — постоянное питание ПИТ — сетевое пи- тание нормальное СБРОС — началь- ный сброс ОСТ — останов РЕГ — регенерация	ACLO — авария се- тевого питания DCLO — авария ис- точника питания INIT — начальный сброс РА, РВ — указатели ошибки устройства	FC0—FC3 — инди- кация состояния про- цессора ACFAIT — авария сетевое питания PDOWN — авария источника SYSRESET — на- чальный сброс BERR — ошибка шины BREAL — очистить шину за 16 тактов

Пример 1.2. Особенности магистральных интерфейсов рассмотрим на примере перспективного интерфейса с наращиваемой разрядностью Versabus, данные о котором приведены в табл. 1.4 и 1.5.

Интерфейс Versabus разработан как универсальный для создания быстродействующих (с частотой передачи данных до 5 МГц) МП средств вычислительной техники с переменной разрядностью (от 8 до 32) для систем управления производственными процессами, связанных систем и *m*МП ИУВС. Для этого в интерфейсе Versabus предусматриваются следующие возможности: малая длительность циклов передачи информации; асинхронная двунаправленная передача данных; передача 1, 2 или 4 байт данных с указанием адресов с точностью до одного байта; прямая адресация до 4 млрд слов памяти (32-разрядные адресные линии); непосредственный доступ к памяти и обеспечение многопроцессорной работы; управление прерываниями с семиуровневым приоритетом; приоритетное представление (захват) интерфейса с пятиуровневой очередью; до 50 линий периферийного ввода-вывода с соответствующими обратными (земляными) линиями; линии стандартного питания ± 5 , ± 12 В для логических схем и линии стандартного питания ± 15 В для схем управления процессами; линия резервного источника питания; линии последовательной передачи данных; линии отдельных средств отображения для ввода и вывода; линии сигналов интерфейса и повторной попытки обмена; линии отдельной аналоговой земли.

В интерфейсе Versabus применен асинхронный принцип обмена типа «master-slave», что облегчает взаимодействие различных по быстродействию устройств и упрощает организацию многопроцессорных систем. Условные временные диаграммы цикла чтения и записи, поясняющие суть этого принципа, приведены в табл. 1.5.

Таблица 1.5



Сплошные линии временных диаграмм соответствуют сигналам ведущего устройства «master», пунктирные — ведомого устройства «slave», заштрихованные участки — безразличным состояниям. Адресация до байта в 8, 16 и 32-разрядных словах осуществляется в соответствии с табл. 1.6 по сигналам выбора байта DSO и DS1 и сигнала выбора двойного слова LWE.

Таблица 1.6

Адресуемая информация	Сигналы		
	DS0	DS1	LWE
Младший байт младшего полуслова	0	1	1
Старший « « «	1	0	1
Младший « старшего «	0	1	0
Старший « « «	1	0	0
Младшее полуслово	0	0	1
Старшее «	1	1	0
Двойное (32-разрядное) слово	0	0	0

Если в течение установленного времени сигнал подтверждения DTACK не принимает значения «1», то такая ситуация в работе интерфейса рассматривается как аварийная и автоматически осуществляется переход к подпрограмме обслуживания аварийной ситуации.

При работе с памятью возможен специальный цикл «чтение — модификация — запись». В этом цикле в период двух последовательных обращений к памяти (чтение, затем запись) сигналы на адресных линиях и строб адреса остаются неизменными, что предотвращает захват интерфейса другими ведущими устройствами. Тем самым достигается реализация операций типа «проверить и установить», необходимых для построения МП систем. Декодирование адресов ОЗУ и ПЗУ осуществляется по линиям RAMDIS и ROMDIS соответственно. При низком уровне на линии IOEM на младших шестнадцати адресных линиях выставляется адрес устройства ввода-вывода. Сигналы остальных адресных линий при этом игнорируются. При низком уровне на линии расширения адреса EXTADDR используется 24-разрядный адресный код, при высоком — 32-разрядный.

Для организации прерываний интерфейс Versabus включает семь линий запроса векторных прерываний: от IRQ1 (низший приоритет) до IRQ7 (высший приоритет) (рис. 1.26). Любое устройство, подключенное к интерфейсу, может выставить запрос на векторное прерывание по схеме «монтажного ИЛИ» на соответствующую линию; например, устройства Y_1 , Y_3 и Y_{n-1} — на линию IRQ2. Получив запрос на прерывание по крайней мере по одной из этих семи линий, контроллер прерываний КП сравнивает уровень запроса с текущим приоритетом ведущего устройства, и если прерывание

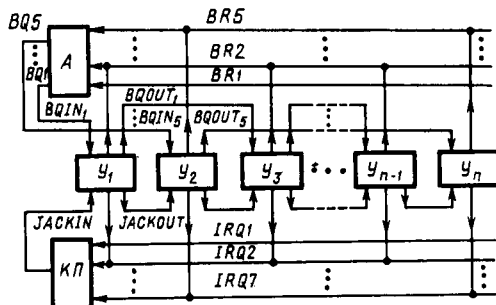


Рис. 1.26. Схема прерываний и захвата интерфейса Versabus.

допустимо, то по окончании выполняемой ведущим устройством команды запрос обрабатывается. На линию JACKIN КП выставляет разрешение прерывания, а на трех младших адресных шинах АОО—АО2 — код уровня принятого прерывания. Сигнал JACK распространяется по цепочке устройств последовательно до тех пор, пока не встречается устройство, запрашивающее прерывание и уровень приоритета которого совпадает с уровнем, выставляемым КП на шинах АОО—АО2. Декодированное таким образом устройство выставляет затем в младшем полуслове данных адрес своего вектора прерывания и становится ведущим на магистрали интерфейса.

Для захвата ведущего положения любым устройством интерфейс Versabus включает пять линий запроса: от BR1 (низший приоритет) до BR5 (высший приоритет). Любое устройство, подключенное к интерфейсу, выставляет запрос на захват ведущего положения по схеме «монтажного ИЛИ» по соответствующей линии BR1 — BR5. Арбитр А интерфейса, получив по этим линиям запрос (или несколько запросов), сравнивает приоритет полученного запроса с приоритетом устройства, занимающего ведущее положение в текущий момент (текущий ведущий). Если полученный запрос имеет более высокий уровень приоритета, то А выдает сигнал BCLR, означающий, что текущему ведущему необходимо освободить интерфейс, и он, начиная последний цикл обмена, снимает сигнал занятости интерфейса.

После этого арбитр выдает сигнал разрешения захвата на одну из линий разрешения BQ1—BQ5. Сигнал BQ, аналогично сигналу JACK в системе прерываний, распространяется по цепочке устройств с той лишь разницей, что на каждом уровне приоритета захвата интерфейса эта цепочка своя, например на рис. 1.26 сигнал разрешения захвата низшего уровня BQ1 распространяется через устройства У₁, У₃, ..., У_{л-1}. Устройство, выставившее перед этим запрос на захват и получившее в ответ сигнал BQ, выставляет сигнал занятости интерфейса BBSY и снимает свой сигнал запроса захвата. В ответ на сигнал BBSY арбитр снимает сигнал BQ. Новое ведущее устройство занимает в интерфейсе ведущее положение, но оно сможет начать свой цикл обращения (чтения, записи) только после того, как будут сняты предыдущим ведущим сигналы подтверждения DTACK и адресный строб AS.

Хотя текущий ведущий должен освободить интерфейс в ответ на сигнал BCLR, он может удерживать управление интерфейсом достаточно долго, для того чтобы завершить передачу блока данных.

В особых ситуациях (например, пропадание питания) в интерфейсе Versabus используется вторая линия очистки интерфейса. По сигналу на этой линии требуется немедленное (в течение 16 циклов) освобождение интерфейса.

Интерфейс Versabus содержит ряд линий специального назначения: функциональные; ошибка/восстановление интерфейса; тест/ошибка системы; системного и вторичного сброса; тактовой частоты; системной паузы; системного уведомления; пропадания питания. *Функциональные линии* FC0—FC3 — это выходные линии процессора, индуцирующие его состояние. Назначение этих линий индивидуально для различных устройств.

По *линии ошибка/восстановление* интерфейса BERR передается сигнал на завершение текущего цикла интерфейса при возникновении неисправимой ошибки. *Линии тест/ошибка* переводят устройства на выполнение тестовых процедур. Любое устройство, обнаружившее при этом ошибку, выставляет сигнал SYS FALL (системная ошибка). По *линии системного сброса* сигнал SYSRESET переводит в исходное состояние все устройства, а по линии вторичного сброса — только выбранные устройства. По *линии тактовой частоты* на все устройства подается опорная частота 16 МГц (обычно от главного процессора). Сигнал на *линии системной паузы* останавливает все процессы и каналы прямого доступа. Сообщение о достижении условия останова передается по *линии системного уведомления* SYSPACK. *Линия пропадания питания* ACFAIT сигнализирует о выходе за допустимые пределы напряжения переменного тока на входах источника питания. *Линия* PDOWN сигнализирует о выходе за допустимые пределы напряжения постоянного тока.

Внешние интерфейсы периферийных устройств. Для связи датчиков информации, исполнительных органов, территориально удаленных от процессора на десятки и сотни метров, в МП ИУВС применяют интерфейсы периферийных устройств. В таких интерфейсах исполь-

зуются как *параллельный*, так и *последовательный* способы обмена информацией, причем последний по причине существенного упрощения собственной линии связи, а следовательно, и снижения стоимости наиболее предпочтителен, если при этом обеспечивается необходимая скорость передачи информации. Основные технические характеристики широко применяемых интерфейсов периферийных устройств приведены в табл. 1.7.

Таблица 1.7

Технические характеристики	Интерфейс				
	КОП (ГОСТ 26.003—80)	RS-232	RS-423	RS-422	MIL-STD-1553
Скорость передачи информации, Кбод	$8 \cdot 10^3$	20	300	10^4	125
Длина магистрали, м	20	15	600	1200	100
Число линий интерфейса	16	1	1	2	2
Способ обмена информацией	Параллельный	Последовательный	Последовательный	Последовательный	Последовательный
Режим обмена	Полудуплексный	Симплексный	Симплексный	Полудуплексный, симплексный	Симплексный, полудуплексный, мультиплексный
Число подключаемых периферийных устройств	15	1	1	1(10)	31

Пример 1.3. В соответствии с ГОСТ 26.003—80 соединение программируемых и непрограммируемых электронных измерительных устройств, применяемых в лабораторных или цеховых условиях, должно осуществляться через многопроводный магистральный канал общего пользования (КОП), схема подключения и структура которого приведены на рис. 1.27, где А — устройство, способное управлять передачей, передавать и принимать информацию (например, ИУВС); В — устройство, способное передавать и принимать информацию (например, цифровой вольтметр); С — устройство, способное только передавать (например, считывающее устройство) либо только принимать (например, генератор) информацию; МД, МС, МУ — соответственно магистрали данных (8-разрядные), синхронизации и управления.

Общая длина КОП не должна превышать 20 м. Число устройств, присоединяемых к КОП, не должно превышать 15 шт. Общее число адресов приемников и источников информации в системе не должно превышать 31 при однобайтовой и 961 при двухбайтовой адресации. Максимальная скорость передачи информации КОП 10^6 байт/с.

Пример 1.4. Стандартный интерфейс BS4421 (Великобритания) предназначен

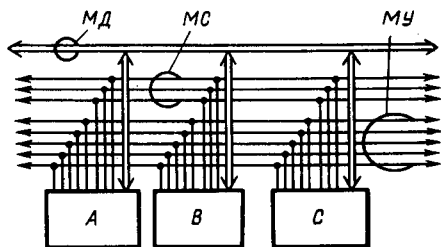


Рис. 1.27. Структура магистрального канала общего пользования.

для организации связей элементов измерительной и вычислительной техники, образующих радиальные соединения. Для функционирования интерфейса, кроме информационных связей между источником и приемником сообщений, необходимо организовать вспомогательные связи (рис. 1.28). Описание функций соответствующих сигналов приведено ниже:

$$\begin{aligned}
 AO &= \begin{cases} 1 - \text{приемник готов к работе;} \\ 0 - \text{приемник не готов к работе;} \end{cases} & SC &= \begin{cases} 1 - \text{переданное слово не содержит ошибок;} \\ 0 - \text{переданное слово содержит ошибки;} \end{cases} \\
 SO &= \begin{cases} 1 - \text{передатчик готов к работе;} \\ 0 - \text{передатчик не готов к работе;} \end{cases} & AE &= \begin{cases} 1 - \text{приемник обнаружил ошибку;} \\ 0 - \text{приемник не обнаружил ошибку;} \end{cases} \\
 AC &= \begin{cases} 1 - \text{приемник готов к работе;} \\ 0 - \text{приемник не готов к работе;} \end{cases} & ST &= \begin{cases} 1 - \text{передаваемое слово последнее;} \\ 0 - \text{передаваемое слово не последнее.} \end{cases}
 \end{aligned}$$

Следовательно, существует два вида управляющих сигналов: сигналы приемника АС и сигналы источника сообщений SC. Цикл работы устройства повторяется до тех пор, пока либо не будет передан последний знак блока данных (при этом сигнал принимает единичное значение), либо пока приемник не распознает ошибку (сигнал АЕ примет единичное значение). Скорость передачи данных зависит от характеристик функциональных элементов системы и задержки сигнала в кабельных соединениях.

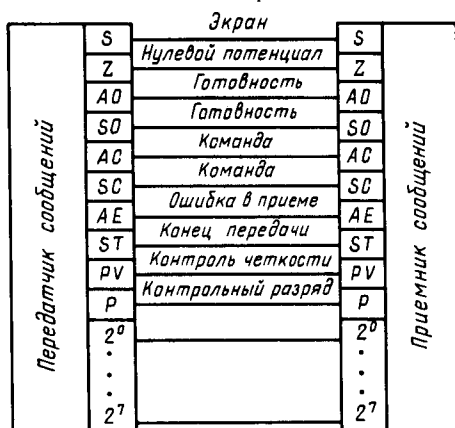


Рис. 1.28. Организация связи интерфейса BS4421.

Стандартом на интерфейс предусмотрена возможность использования управляющих (командных и оповещения), информационных и программных сигналов.

Автоматическое функционирование элементов и обмен информацией между ними обеспечиваются формированием внутренних сигналов управления и воздействием внешних сигналов, которые воспринимаются и обрабатываются этими элементами.

Управляющие сигналы, передаваемые элементом во внешнюю линию связи, разделяются на сигналы оповещения М, характеризующие состояние элемента, и командные В сигналы воздействия на состояние абонента. Сигналы оповещения М1 и М2 формируются в зависимости от готовности к приему или передаче информации функциональными элементами. Для реализации приема информации элементом n необходимо, чтобы $(n - 1)$ -й элемент завершил цикл функционирования, при этом на его выходах устанавливаются информационные I или программные Р сигналы. О готовности к пе-

передаче ($n-1$)-й элемент сообщает сигналом $M2$, который принимает значения: $M2=1$ (сигналы I и P не сформированы); $M2=10$ (сигналы I и P сформированы); $M2=0$ (готовность к передаче).

В момент приема информации элементом n элемент $n+1$ не должен нуждаться в выходных сигналах I и P элемента n , об этом ($n+1$)-й элемент сообщает сигналом $M1$. Сигналы $M1=1$, I и P необходимы на входе элемента, так как не завершён очередной цикл. Сигнал $M1=0$ смысла не имеет, сигнал $M10=10$ означает, что входные сигналы можно менять. Сигналы $B1$ и $B2$ — командные. Сигналы оповещения, которые воздействуют на элемент n со стороны элементов $n-1$ и $n+1$, являются для элемента n командными. Функционирование элемента n возможно при условии, что элемент $n-1$ сформирует сигнал $M2=10$, а элемент $n+1$ — сигналы $M1=10$, которые для элемента n являются командными сигналами $B2=10$ и $B1=10$. В элементе n воздействие сигналов $B2=10$, $B1=10$ приводит к формированию сигналов $M2=1$, $M1=1$.

На рис. 1.29 приведена структура устройства, содержащая три функциональных элемента и поясняющая формирование сигналов B и M . Очередной цикл работы устройства начинается с переключения мультиплексора для обеспечения связи цифро-

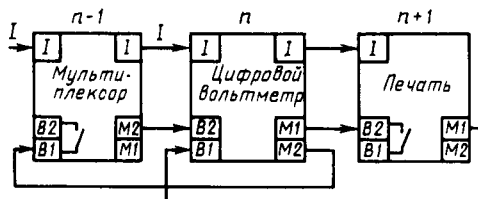


Рис. 1.29. Структура связи интерфейса.

вого вольтметра и требуемого канала измерения. После окончания переходных процессов в мультиплексоре первый функциональный элемент формирует сигнал $M2$, а третий элемент — печатающее устройство — формирует сигнал $M1$ о том, что предыдущее измерение зафиксировано. С момента появления сигналов $M1$ и $M2$ производится измерение напряжения сигнала подключенного канала. Кроме перечисленных сигналов $B1$ и $B2$, для автоматической установки в исходное состояние всех элементов предусмотрен командный сигнал $B0$ — центральный сброс. По этому сигналу все функциональные элементы переводятся в исходное состояние и сигналы $M1=0$, $M2=1$. Сигнал $B0$ формируется вручную соответствующим формирователем, который управляется от клавиш; если клавиша нажата, то $B0=0$, если отжата, то $B0=1$.

Функциональный элемент, включаемый в цепь первым и инициирующий начало цикла функционирования, должен содержать кнопку и соответствующий формирователь сигнала запуска, условие запуска, изменение сигналов $B1$ и $B2$ с единичных значений на нулевые. Для всех остальных элементов системы установки запуска $B2=10$, так как $B1$ подготовлен воздействием $B0=0$.

Изложенные логические условия функционирования интерфейса реализуются схемой управления, показанной на рис. 1.30. При воздействии сигнала $B0$ триггеры $T1$ и

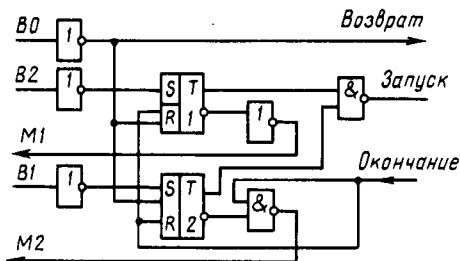


Рис. 1.30. Пример схемы выработки управляющих сигналов.

T2 переключаются в исходное состояние ($M1=0$; $M2=1$). С появлением сигнала $B2=10$ триггер T1 формирует сигнал «Запуск» — начало функционирования элемента ($M1=1$). По окончании цикла на схему управления поступает сигнал «Окончание», при воздействии которого триггеры T1 и T2 переключаются в исходное состояние и появляются сигналы $M1=10$ и $M2=10$. Каждый последующий запуск начинается при сигналах $B1=10$ и $B2=10$.

Интерфейс IMS-1 может использоваться для подключения функциональных элементов к ЭВМ, образуя каналы ввода-вывода данных. Распределение управляющих сигналов в подобных устройствах иллюстрирует рис. 1.30. Запуск функционирования каскада элементов выполняется либо вручную (клавиша запуска первого элемента), либо управляющим сигналом MS, формируемым ЭВМ. Сигнал B0 формируется также ЭВМ.

Примерами наиболее распространенных *последовательных интерфейсов периферийных устройств могут служить интерфейсы RS-232C, RS-423, RS-422*. В основе интерфейсов RS-232C, RS-422 лежит однопроводная несогласованная линия (рис. 1.31, а), по которой информация передается двуполярными посылками со скоростью до 20 Кбод (RS-232C) и 300 Кбод (RS-423) при длине линии не более 15 и 600 м

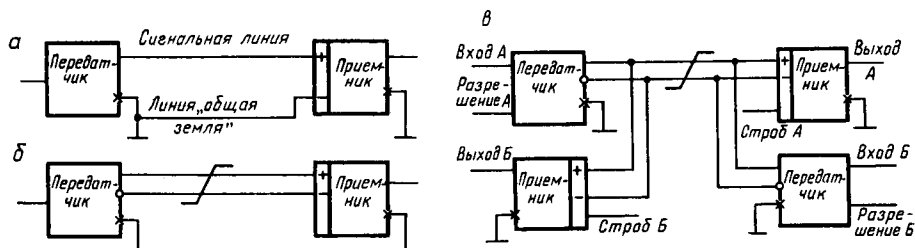


Рис. 1.31. Схемы однопроводной несогласованной (а) и симметричных дифференциальных последовательных линий связи с симплексным (б) и полудуплексным (в) режимами передачи информации.

соответственно. В однопроводной линии для передачи сигнала используется один провод, напряжение на котором сравнивается в приемнике с напряжением линии «общая земля», общей для всех прочих сигнальных проводников. Этот способ построения линий наиболее прост, но имеет существенный недостаток: на информационный сигнал накладываются помехи в линии. Так как помехи в линии пропорциональны длине линии связи и ширине полосы рабочих частот, то в интерфейсах RS-232C и RS-423 наложены ограничения на оба параметра (табл. 1.8). Например, в интерфейсе RS-423 при скорости передачи информации 3 Кбод длина линии может достигать 1200 м, а при скорости 300 Кбод — всего лишь 12 м. Кроме того, для уменьшения взаимных помех скорость нарастания фронтов передаваемых сигналов ограничивается до 30 В/мкс.

Интерфейс RS-422 распространяется на симметричные дифференциальные линии (витая пара, радиочастотный кабель), обладающие более высокими характеристиками, чем однопроводные линии. В частности, по линии интерфейса RS-422 возможна передача информации со скоростью до 100 Кбод на расстояние до 1200 м и со скоростью 10 Мбод на расстояние до 12 м. Дифференциальный режим (рис.

Таблица 1.8

Технические характеристики	Интерфейс		
	RS-232C	RS-423	RS-422
Линия	Однопроводная, несогласованная	Однопроводная, несогласованная	Дифференциальная, согласованная
Максимальная длина линии, м	15	600	1200
Максимальная скорость передачи информации, Кбод	20	100 (при 12 м)	10 ⁴ (при 12 м)
Выходное напряжение передатчика при работе на нагрузку, В	От ± 5 до ± 15 при $R_n = 3-7$ кОм	$\pm 3,6$	2,0
Скорость нарастания сигнала на выходе передатчика, В/мкс	<30	Зависит от длины кабеля и частоты переключения	Не ограничена
Входное сопротивление приемника, кОм	3—7	≥ 4	≥ 4
Максимальное значение порогового напряжения приемника, В	± 3	$\pm 0,2$	$\pm 0,2$
Максимально допустимое входное напряжение приемника, В	± 25	± 12	± 12

1.31, б) достигается применением дифференциального передатчика, согласованной линии связи (в виде витой пары или радиочастотного кабеля) и дифференциального приемника. Сигнал передатчика появляется на входе приемника в виде разностного напряжения, тогда как помехи в линии остаются синфазными. Благодаря этому дифференциальный приемник с достаточным диапазоном подавления синфазной составляющей может отличать сигнал от помехи. К тому же интерфейс RS-422 позволяет наряду с симплексным режимом передачи организовать полудуплексный режим (рис. 1.31, в) и мульт

типлексный режим последовательной передачи информации. В последнем случае, применяемом для связи между периферийными устройствами m МП ИУВС, для передачи информации в одном направлении используются две дифференциальные симметричные линии связи, например витые пары. В исходящем направлении включены один передатчик и n приемников. Такой режим позволяет обслуживать до 12 абонентов. Основные трудности заключаются в обеспечении хорошего заземления системы. Основные параметры последовательных интерфейсов периферийных устройств приведены в табл. 1.8.

Говоря о внешнем интерфейсе периферийных устройств, в частности последовательном интерфейсе в широком смысле слова, обычно под термином «интерфейс» понимают не только функциональное назначение линии связи и электрические параметры передаваемых сигналов, но и состав аппаратуры, необходимой для реализации связи между рядом периферийных устройств с помощью последовательной линии связи, принципы организации обмена информацией, язык обмена информацией, организацию контроля передачи информации. При этом в общем случае различают несколько уровней интерфейса, часть которых реализуется аппаратурно, а часть — программно.

Аппаратура для физической реализации внешнего последовательного интерфейса периферийных устройств включает в себя (рис. 1.32)

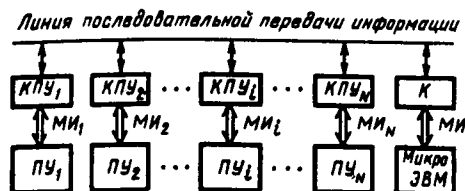


Рис. 1.32. Структура внешнего последовательного интерфейса.

собственно линию последовательной передачи информации (одиночный проводник, витая пара, радиочастотный кабель, волоконно-оптический кабель) и контроллеры устройств, которые предназначены для управления обменом информацией, осуществления сопряжения с линией передачи информации, осуществления контроля за передачей информации, состоянием устройств, участвующих в связи, самоконтроля. На рисунке $ПУ_i$ и $КПУ_i$ периферийное устройство и его контроллер для связи с внешним последовательным интерфейсом; микроЭВМ и $К$ — центральная микроЭВМ ИУВС и ее контроллер для связи с внешним интерфейсом; $МИ_i$ и $МИ$ — магистральный (внутримашинный) интерфейс i периферийного устройства и микроЭВМ соответственно.

Обмен информацией осуществляется в большинстве случаев передачей сообщений, состоящих из командных, информационных и ответных слов. При этом используется определенный язык передачи этих слов. Типичное слово синхронного протокола с битовой организацией содержит [6] начальный флаг, адресное, управляющее и информационные поля, проверочную последовательность слова и завершающий флаг.

Для передачи слов с битовой организацией широко применяют код без возвращения к нулю, код без возвращения к нулю с инверсией, код Манчестер II.

Код без возвращения к нулю (рис. 1.33, а) отображает последовательность двоичных битов последовательностью уровней напряжения, постоянных на интервале передаваемого двоичного разряда.

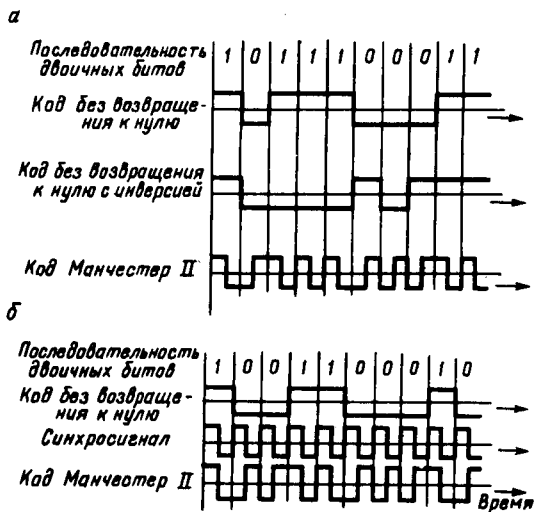


Рис. 1.33. Примеры кодов передачи информации.

В коде без возвращения к нулю с инверсией «1» передается отсутствием изменения уровня предшествующего бита, а «0» — инверсией этого уровня (рис. 1.33, а). Таким образом, «1» и «0» в этом коде могут быть представлены в одной последовательности как низким, так и высоким уровнями напряжения, причем длинная последовательность «1» образует постоянный уровень, а длинная последовательность «0» — биполярный периодический сигнал.

Код Манчестер II отображает каждый бит двоичной последовательности переходом уровней: если низкий уровень сменяется высоким, то передается «0», если высокий уровень сменяется низким — «1». Переходы имеют место в середине временного интервала, отведенного каждому двоичному биту (рис. 1.33, а). Код Манчестер II легко получается из кода без возвращения к нулю, если последний подать на один вход схемы логической равнозначности, на второй вход которой поступают синхросигнал в виде меандра с периодом, равным периоду кода без возвращения к нулю, и синфазный с ним (рис. 1.33, б). Код Манчестер II, в частности, применен в интерфейсе стандарта MIL — STD-1553 [7], используемом при создании бортовой самолетной локальной сети. Он обладает большей помехозащищенностью по сравнению с кодом без возвращения к нулю, поскольку, во-первых, занимает полосу частот от половины до полного значения тактовой частоты, в то время как код без возвращения к нулю —

от постоянной составляющей до половины тактовой частоты (отсутствие постоянной составляющей при кодировании по коду Манчестер II позволяет использовать более простые полосовые усилители переменного тока, применять подавление низкочастотных шумов); во-вторых, применение кода Манчестер II дает возможность обнаружения ошибки в каждом передаваемом разряде, если помнить ту особенность, что информационный переход имеет место в середине интервала, соответствующего биту. Поскольку присутствие высокого уровня или низкого уровня в течение всего интервала бита недопустимо, появление таких ситуаций может свидетельствовать о наличии ошибок. Декодирование «0» («1») при передаче «1» («0») в коде Манчестер II может произойти только тогда, когда из-за помехи инверсия логических уровней будет иметь место в каждой половине интервала одного бита, однако вероятность одновременной инверсии фазы в пределах одного бита невелика.

Применение кода Манчестер II, наконец, позволяет уменьшить стоимость линии за счет исключения провода для передачи тактовых импульсов в интерфейсе, поскольку информационная последовательность в коде Манчестер II содержит в себе как тактовые, так и информационные сигналы. Более того, при применении кода без возвращения к нулю не исключено возникновение фазовых сдвигов между информационными и синхронизирующими последовательностями. Обладая свойством самосинхронизации, код Манчестер II позволяет обойти эти проблемы.

Системные интерфейсы. В последнее время в связи с распространением МП и *m*МП ИУВС, отдельные микропроцессоры или устройства ввода-вывода которых могут отстоять друг от друга территориально на сотни метров (например, заводская или цеховая ИУВС), все большее применение находят системные интерфейсы или интерфейсы локальных сетей. Известны такие локальные сети, как DECnet фирмы «Digital Equipment Corp.», Net /One фирмы «Ungermann Bass Inc.», Z-net фирмы «Zilog Inc.», Cluster/One фирмы «Nester Systen Inc.», сеть фирмы «Motorola», сеть фирмы IBM, Omminet фирмы «Corvus Inc.», Ethernet фирмы «Intel Corp». Некоторые технические характеристики зарубежных локальных сетей приведены в табл. 1.9.

Системный интерфейс (интерфейс локальных сетей), как правило, многоуровневая архитектура (совокупность) аппаратурных и программных средств.

Рассмотрим задачи, решаемые отдельными уровнями системного интерфейса на примере архитектуры локальной сети *i*LNA фирмы «Intel». Физические средства интерфейса (первый уровень, рис. 1.34, а) в сети *i*LNA являются аппаратурными средствами и включают в себя последовательную линию связи, приемопередатчики, передающие информацию в коде Манчестер II. Второй уровень интерфейса (уровень звена данных) реализуется как аппаратурными, так и программными средствами и позволяет организовать передачу пакетов сообщений от одного устройства сети к другому. На этом уровне выполняются такие функции, как кадровая синхронизация,

Таблица 1.9

Технические характеристики	Локальная сеть		
	Cluster/One	PLAN	Ethernet
Максимальная скорость передачи данных, Мбод	0,24	2,5	10
Число подключаемых устройств	64	225	1024
Шина интерфейса	16-проводный плоский кабель	Коаксиальный кабель	Коаксиальный кабель
Длина связей сети, м	300	6400	2300
Топология сети	Произвольная	Многозвездная	Линейная шина
Метод доступа к шине локальной сети	Множественный доступ с контролем несущей и исключением столкновений	Эстафетная передача маркера управления	Множественный доступ с контролем несущей и обнаружением столкновений

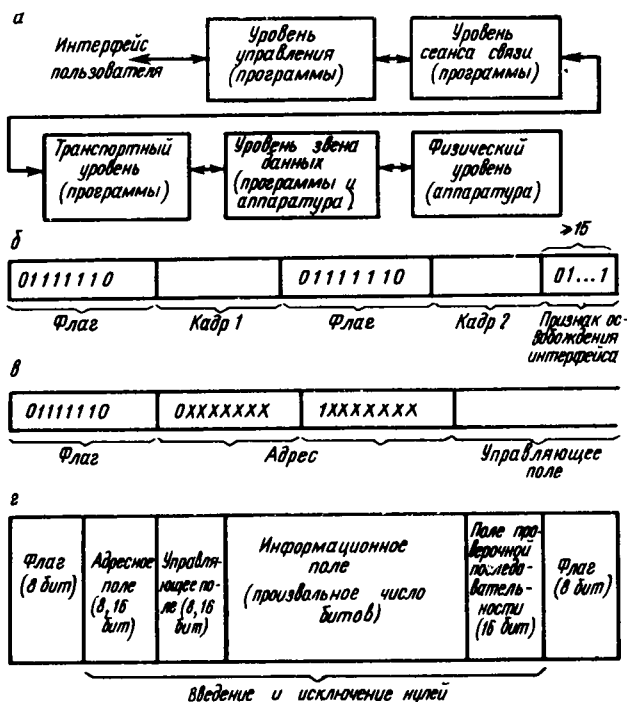


Рис. 1.34. Системный интерфейс:

а — уровни системного интерфейса, б — последовательность кадров, в — формат адреса кадра, г — формат кадра.

т. е. определение начальных и конечных точек сообщения; адресация, т. е. определение устройства, принимающего сообщение; обнаружение ошибок; управление подключением многочисленных устройств сети к физическому адресу. Ниже подробнее рассмотрим типичное образование кадра передаваемой информации при последовательной передаче данных с битовой организацией [8] и те особенности построения кадра, которые обеспечивают выполнение функций, возлагаемых на второй уровень интерфейса.

Информация передается последовательно кадрами. В протоколах последовательной передачи информации с битовой организацией для определения различных частей сообщения используется положение этих частей в последовательности битов. Кадр с обеих сторон ограничивается флаговыми последовательностями 01111110 (рис. 1.34, б). Если кадр оканчивается не флагом, а последовательностью единиц числом ≥ 15 , то это означает, что передающее устройство закончило передачу и другие устройства в сети могут начать передачу информации. Передающее устройство может закончить кадр посылкой комбинации от 7 до 14 единиц, что означает режим преждевременного прерывания кадра.

Приемное устройство в этом случае не принимает последний кадр и не может выходить на передачу, пока не получит команды от передающего устройства. Кадр содержит адресное, управляющее и информационные поля. Адресное поле следует за первым флагом кадра и определяет адрес отправителя или получателя кадра. Возможны два режима адресации: режим основной адресации, при котором используется один адресный байт, и режим расширенной адресации, при котором длина адресного поля равна 1 байт или более. В последнем случае, если младший бит первого адресного байта есть «0», следующий байт есть продолжение адресного поля. Это поле заканчивается первым встречаемым байтом, в котором младший бит равен «1». Таким образом, адресное поле может быть расширено до любого числа байтов. Пример двухбайтового адресного поля приведен на рис. 1.34, в.

За адресным полем следует управляющее поле длиной 1 или 2 байт, предназначенное для кодирования команд и ответов, необходимых для управления линией передачи данных. Возможны три типа управляющего поля: супервизорный (контрольный); передачи информации; с неупорядоченной последовательностью (нумерованных) кадров. Супервизорный тип управляющего поля используется для передачи состояния готовности, занятости и для оповещения об ошибках в последовательности кадров. Второй тип управляющего поля применяется собственно для передачи упорядоченной последовательности информационных кадров. Тип управляющего поля с неупорядоченной последовательностью кадров необходим для управления линией передачи данных, т. е. для включения в работу, управления режимом ответа приемных устройств.

Перед флагом в конце кадра размещается 16-битовое поле проверочной последовательности, обеспечивающей обнаружение ошибок (рис. 1.34, г). Обнаружение ошибок ведется по всему кадру (между

начальной и замыкающей флаговыми последовательностями) с использованием циклического кода вида $X^{16} + X^{12} + X^5 + 1$ и передачей инвертированного остатка в качестве проверочной последовательности. В передающем устройстве для этого сначала все 16 бит последовательности проверочного кода (ППК) устанавливаются в «1». Затем двоичное значение подлежащего передаче отрезка сообщения умножается и делится на соответствующие полиномы. Целая часть частного отбрасывается, а дополнение остатка передается в качестве ППК. В приемнике проводятся аналогичные вычисления по всем кадрам, включая и ППК. Если ошибок нет, то остаток равен FOB8. Информационное поле может быть любой длины, в том числе и нулевой (минимальная длина кадра, не считая флаговых последовательностей, равна 32). Информационное поле включает в себя собственно пересылаемые данные и информацию о распределении данных в этом поле. Важной особенностью протокола последовательной передачи данных с битовой организацией является введение и исключение нулей, чем обеспечивается прозрачность (выделение) данных в кадре. В пределах битовой последовательности кадра между двумя флагами нуль автоматически вставляется в процессе передачи после пяти подряд идущих единиц и исключается на приеме. Поэтому ни одна битовая последовательность в кадре не может быть ошибочно расшифрована как флаговая, как комбинация освобождения интерфейса.

Последовательная передача данных по линии интерфейса может вестись либо в коде БВНИ [8], либо в коде Манчестер II по линии связи, соответствующей интерфейсам RS-232C, RS-422, RS-423.

Функции формирования кадра в интерфейсе выполняются на аппаратном уровне. Программные же средства второго уровня интерфейса решают функции управления передачей пакета данных (доступа устройства к линии последовательной передачи данных). При этом используется принцип приоритетного представления линии под наименованием «коллективный доступ с контролем несущей и исключением столкновений». Суть принципа заключается в следующем: устройства, подключенные к сети, асинхронно пытаются передать по общей линии каждый свой пакет данных, одновременно «прислушиваясь» (следа за несущей) к работе других устройств. Если в момент передачи происходит «столкновение» передач нескольких устройств, то все они прекращают передачу на случайный по длительности интервал времени, затем делают попытку снова передать данные. Другой способ управления передачей данных основан на принципе «передачи эстафетной палочки», при котором каждое устройство последовательно получает возможность для обмена и либо пользуется этой возможностью, либо нет — в зависимости от необходимости.

Транспортный уровень интерфейса, реализуемый программно, обеспечивает надежную передачу сообщений переменной длины, несмотря на то что, скажем, второй уровень передачи может терять отдельные пакеты из-за ошибок, столкновений; дублировать их и передавать в непоследовательном порядке. Одновременно транспорт-

ный уровень следит за тем, чтобы одно более «скоростное» устройство «не завалило» данными менее «скоростное», короче говоря, транспортный уровень интерфейса управляет потоками данных с учетом объема буферной памяти устройств.

Преобразователи уровней. Кроме частей управляющей системы, которые хорошо реализуются на основе типовых комплектов БИС микропроцессора, имеется большое количество средств сопряжения с объектом управления, индикации, документирования и т. д.

В подавляющем большинстве управляющих систем широко используется вся номенклатура радиоэлектронных элементов, как дискретных (транзисторы, диоды, резисторы, конденсаторы, индуктивности), так и интегральных (ИС, СИС, БИС, наборы компонентов), а также конструктивные установочные детали (клавиатура, кнопки, индикаторы, тумблеры, разъемы).

Преобразователи уровней (адаптеры, драйверы, трансляторы) — специальные элементы цифровых устройств, предназначенные для обеспечения совместимости уровней различных серий цифровых элементов. В настоящее время уровни «0» и «1» элементов ТТЛ-типа и их нагрузочные характеристики стали фактически стандартными для цифровых устройств, МП, микроЭВМ и т. д. вне зависимости от технологии и схемотехники базового элемента.

Кроме обеспечения совместимости уровней сигналов, преобразователи уровней должны удовлетворять специальным требованиям, например таким, как сохранение преобразователем порогового уровня управляющего элемента, уровней токов, способа кодирования двоичных переменных (или изменения способа кодирования); обеспечение заданных требований по нагрузочной способности и параметрам быстродействия; необходимость выполнения логических операций; обеспечение парафазных выходов и т. п.

Большинство схем с высоким уровнем интеграции выполнено на основе p -, n - или КМДП-технологии, в то время как схемы малого и среднего уровней интеграции — на основе ТТЛ-, ЭСЛ- и КМДП-технологии. Имеется большое число других базовых схем, поэтому рассмотреть все возможные варианты преобразователей уровня не представляется возможным. Тем не менее можно сформулировать **общие правила** построения преобразователей уровня, пригодные для большинства случаев:

преобразователи уровней проектируются для конкретных схем с обязательным учетом выходных характеристик и параметров управляющего элемента и входных характеристик и параметров управляемого элемента;

перепад логических уровней управляющего элемента должен быть достаточным для надежного функционирования преобразователя уровней;

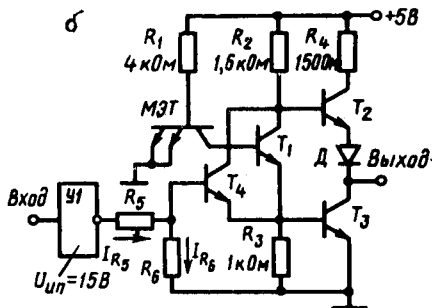
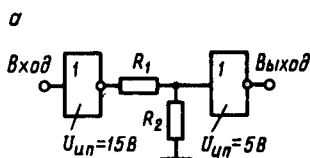
преобразователь уровней должен обеспечивать необходимые динамические параметры с учетом емкостных и активных нагрузок.

В составе схем малой и средней степеней интеграции ТТЛ-, ЭСЛ-, КМДП-типа имеются специально разработанные преобразователи уровней. Среди них можно выделить преобразователи ЭСЛ→ТТЛ

(К500ПУ125); ТТЛ→ЭСЛ (К500ПУ124); КМДП→ТТЛ (176ПУ1, 176ПУ2, 176ПУ3, 564ПУ4, 564ЛН1, 564ЛН2); ТТЛ→КМДП (К155ЛН3, К155ЛН5) и др. В тех случаях, когда необходимо разработать специальный преобразователь уровней, можно воспользоваться одной из следующих схем: 1) делители напряжения; 2) фиксаторы уровней; 3) сдвигатели уровней; 4) ключевые транзисторные схемы; 5) схемы, работающие на принципе переключения тока; 6) оптронные переключатели; 7) трансформаторные схемы.

Делители напряжения применяют для преобразования высоких уровней напряжения в низкие. На рис. 1.35, а показан пример стыковки схем КМДП-типа, работающих при высоком уровне напряжения

Рис. 1.35. Примеры схемы преобразования высокого уровня в низкий для схемы КМДП-типа (а) и ТТЛ-типа (б).



источника питания, со схемами КМДП-типа, работающими с низким уровнем напряжения источника питания. Для схемы можно рекомендовать $R_1 = 20$ кОм, $R_2 = 10$ кОм. Для улучшения динамических свойств преобразователя уровней можно использовать компенсированные делители.

Более сложная схема делителя для случая, когда схемы КМДП-типа запитываются напряжениями, большими +5 В, показана на рис. 1.35, б. В качестве активного элемента преобразователя уровней в данной схеме используется любая схема ТТЛ-типа, имеющая расширительные входы по ИЛИ. Информационные входы схемы ТТЛ-типа заземляются, в результате чего транзистор T_1 всегда закрыт. Внешний транзистор T_4 присоединяется к расширительным входам схемы ТТЛ-типа. Управление транзистором T_4 осуществляется делителем R_5, R_6 , подключенным к выходу $Y1$ управляющей схемы КМДП-типа. Когда на выходе $Y1$ низкий потенциал, транзистор T_4 закрыт и на выходе формируется уровень «1» схем ТТЛ-типа. Когда на выходе $Y1$ высокий потенциал, транзисторы T_4 и T_3 насыщены и на выходе формируется уровень «0» схем ТТЛ-типа. Рассмотрим порядок расчета величин R_5 и R_6 для типовой схемы ТТЛ-типа серии К155, обеспечивающей коэффициент разветвления по выходу, равный десяти:

$$I_{\text{к. нас } T_4} = (U_{\text{ип}} - U_{\text{кз. нас. } T_4} - U_{\text{бз. нас } T_d}) / R_2 \approx 2,5 \text{ мА.} \quad (1.16)$$

Приняв для $T_4 \beta = 30$ и $k_{\text{нас}} = 1,5$, определим

$$I_{б. \text{ нас } T_4} = I_{к. \text{ нас } T_4} k_{\text{нас}} / \beta \approx 0,13 \text{ мА.} \quad (1.17)$$

Через резистор R_6 проходит ток

$$I_{R_6} = (U_{бэ. \text{ нас } T_4} + U_{бэ. \text{ нас } T_3}) / R_6. \quad (1.18)$$

Приняв $I_{R_6} \leq I_{б. \text{ нас } T_4}$, определим R_6 . При $I_{R_6} = I_{б. \text{ нас } T_4}$ получим $R_6 = 10 \text{ кОм}$. Через резистор R_5 проходит ток

$$I_{R_5} = I_{R_6} + I_{б. \text{ нас } T_4} \approx 0,26 \text{ мА.} \quad (1.19)$$

По выходным характеристикам схемы КМДП-типа У1 определяем $U_{\text{вых}}^1$ при токе I_{R_5} . Сопротивление

$$R_5 = (U_{\text{вых}}^1 - U_{бэ. \text{ нас } T_4} - U_{бэ. \text{ нас } T_3}) / I_{R_5} \approx 51 \text{ кОм.} \quad (1.20)$$

Для улучшения динамических свойств рассмотренного преобразователя уровней необходимо либо уменьшить R_5 , R_6 , либо применить компенсированный делитель.

На рис. 1.36, а показан пример схемы преобразователя уровней на основе схемы *фиксатора потенциалов*. Требуется преобразовать высокий потенциал $27 \text{ В} \pm \frac{2}{5}$ в уровни, необходимые для работы схе-

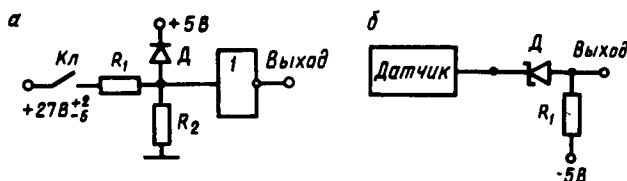


Рис. 1.36. Примеры схемы преобразователя уровней на основе схемы фиксатора потенциалов (а) и сдвигателя уровня (б).

мы КМДП-типа У1, запитываемой напряжением $+5 \text{ В}$. При разомкнутом ключе уровень U^0 обеспечивается резистором R_2 . При замкнутом ключе на выходе схемы У1 будет действовать потенциал, равный $U_{\text{ин}} + U_{\text{д}}$. Произведем числовой расчет. Выберем $R_2 = 10 \text{ кОм}$, тогда

$$I_{R_2} = (U_{\text{ин}} = U_{\text{д}}) / R_2 = 0,57 \text{ мА.} \quad (1.21)$$

Зададимся током, проходящим через диод Д, равным 1 мА , при минимальном входном напряжении 22 В , тогда

$$I_{R_1} = I_{R_2} + I_{\text{д}} = 1,57 \text{ мА;} \quad (1.22)$$

$$R_1 = [U_{\text{вх min}} - (U_{\text{ин}} + U_{\text{д}})] / I_{R_1} \approx 10 \text{ кОм.} \quad (1.23)$$

На рис. 1.36, б приведена схема *сдвигателя уровней*, преобразующего сигнал с низкоомного датчика с уровнями $+5$ и $+4,1 \text{ В}$ в уровни стандартного элемента ЭСЛ-типа $-0,7$, $-1,6 \text{ В}$. Очевидно, что напряжение пробоя стабилитрона Д, используемого в схеме, должно

быть равно 5,7 В. Задавшись током, проходящим через стабилитрон, равным 5 мА (при условии, что такой ток обеспечивает выходная цепь датчика), определим

$$R_1 = [4,1 - U_{\text{д}} - (-5)] / 5 = 0,68 \text{ кОм.} \quad (1.24)$$

В качестве сдвигателей уровней в различных схемах применяют эмиттерные повторители, резисторы, диоды, опорные элементы и т. п.

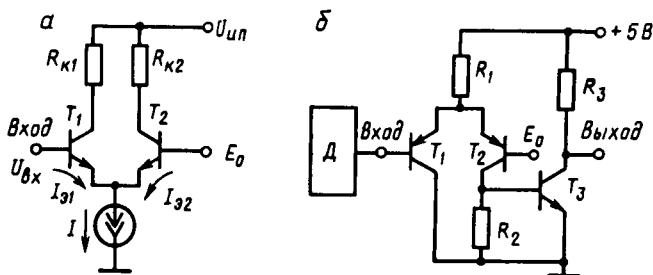


Рис. 1.37. Схемы переключателя тока (а) и преобразователя уровней на его основе (б).

Ключевые транзисторные схемы в качестве преобразователей уровней применяют в том случае, когда требуется согласовать схемы, запитываемые напряжениями разного знака при большом значении логического перепада в каждой из схем [9].

Схемы, работающие на *принципе переключения тока*, используют в качестве преобразователей уровней в случаях, когда логический период может составлять доли вольт. Принцип переключения тока иллюстрируется схемой, показанной на рис. 1.37. Для данной схемы можно записать

$$-U_{\text{вх}} + U_{\text{б}31} - U_{\text{б}32} + E_0 = 0. \quad (1.25)$$

Через ток эмиттера выразим

$$U_{\text{б}3} = U_{\text{б}3}^* + m\varphi_T \ln(I_3/I_3^*), \quad (1.26)$$

где $U_{\text{б}3}^*$ и I_3^* — параметры характерной точки на зависимости $U_{\text{б}3} = f(I_3)$, определенные экспериментально или из характеристик транзистора [9], тогда

$$U_{\text{вх}} = E_0 + m\varphi_T \ln(I_{31}/I_{32}) = E_0 + m\varphi_T \ln[I_{31}/(I - I_{31})]. \quad (1.27)$$

Будем считать, что транзистор, через который проходит ток $0,01 I$, закрыт, а транзистор, через который проходит ток $0,99 I$, открыт. Активный режим для транзисторов T_1 и T_2 обеспечивается соответствующим выбором $R_{\text{к}1}$, $R_{\text{к}2}$ при заданных E_0 , $U_{\text{нп}}$. Подставляя $I_{31} = 0,01 I$, $I_{32} = 0,99 I$ и $I_{31} = 0,99 I$, $I_{32} = 0,01 I$, определим при $m\varphi_T = 30 \text{ мВ}$, что соответствующие $U_{\text{вх}} \approx E_0 - 140 \text{ мВ}$ и $U_{\text{вх}} \approx E_0 + 140 \text{ мВ}$.

Итак, изменение входного напряжения по отношению к опорному на ± 140 мВ приводит к переключению тока из коллектора цепи транзистора T_1 в цепь коллектора транзистора T_2 и наоборот, что может быть использовано при разработке преобразователей уровней. При этом возможны две ситуации:

выход управляющей схемы имеет одну цепь, в которой сигнал изменяется с малым логическим перепадом. Тогда

$$E_0 = (U^0 + U^1)/2; \quad (1.28)$$

выход управляющей схемы имеет дифференциальную структуру. В этом случае одна цепь выхода подключается к базе транзистора T_1 , а другая — к базе транзистора T_2 .

На рис. 1.37,б приведена схема преобразователя уровней $+1,5$, $+1,0$ В в уровни схем ТТЛ-типа. Порядок расчета схемы следующий. Обеспечиваем $E_0 = +1,25$ В. Пусть схема нагружена на 10 схем ТТЛ-типа с параметрами $I_{вх}^0 = -1,6$ мА; $I_{вх}^1 = 0,04$ мА; $U^0 \leq 0,4$ В; $U^1 \geq 2,4$ В.

Допустим, выбран транзистор T_3 с $I_{k\max} = 20$ мА; $\beta = 30$, тогда $I_{R3\max}$ определится как

$$I_{R3\max} = I_{k\max} - k_{раз} |I_{вх}^0| = 4 \text{ мА} \quad (1.29)$$

и, следовательно,

$$R_{3\min} = (U_{ин} - U_{кз.насT_3}) / (k_{раз} I_{вх}) = 1,2 \text{ кОм}. \quad (1.30)$$

Максимальное значение R_3 определим из соотношения

$$R_{3\max} = (U_{ин} - U_{мин}^1) / (k_{раз} I_{вх}^1) = 6,2 \text{ кОм}. \quad (1.31)$$

Ток базы насыщенного транзистора T_3 при $R_3 = 1,2$ кОм

$$I_{б.насT_3} = (I_{к.насT_3} k_{нас}) / \beta = 1 \text{ мА}. \quad (1.32)$$

Примем $I_{R_2} = 1$ мА, тогда

$$R_2 = U_{бэ.насT_3} / I_{R_2} = 0,68 \text{ кОм}. \quad (1.33)$$

Переключаемый ток коллектора транзистора T_2

$$I_{кT_2} = I_{R_2} + I_{б.насT_3} = 2 \text{ мА}. \quad (1.34)$$

Определяем

$$R_1 = (U_{ин} - U_{бэT_2} - E_0) / \{(1 + \beta) / \beta\} I_{кT_2} = 1,5 \text{ кОм}. \quad (1.35)$$

Оптронные переключатели используются для гальванической развязки электрических цепей при одновременном преобразовании

уровней. На рис. 1.38 показана схема преобразователя уровней тока в уровни схем ТТЛ-типа.

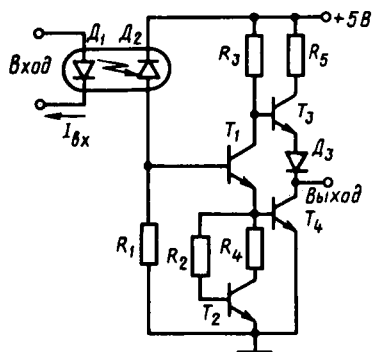
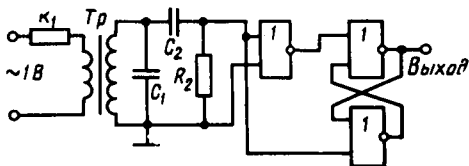


Рис. 1.38. Схема преобразователя тока в уровни схем ТТЛ-типа.

Рис. 1.39. Схема трансформаторного преобразователя уровней.



На рис. 1.39 приведен пример использования схемы *трансформаторного преобразователя* уровней.

Примечание. Различные варианты преобразователей уровней приведены в [9].

1.4. Схемотехника цифровых и аналоговых интегральных схем малой, средней и большой степени интеграции

Элементная база цифровых больших интегральных схем и микропроцессоров. Технические характеристики и режимы эксплуатации цифровых БИС и микропроцессоров в значительной степени определяются структурой и параметрами их базовых элементов. С точки зрения технической реализации наибольшей простотой изготовления характеризуются БИС, схемотехника которых выполнена на транзисторах с индуцированным каналом *p*-типа. Эти БИС обладают невысоким быстродействием, и поэтому их доля в общем выпуске БИС мала и непрерывно сокращается. По *p*-МДП-технологии отечественная промышленность выпускает микропроцессорный комплект серии К536.

Поскольку подвижность электронов в транзисторах с каналом *n*-типа более чем в два раза превышает подвижность дырок в транзисторах с каналом *p*-типа, *n*-канальные БИС обладают рядом неоспоримых преимуществ перед аналогичными БИС на транзисторах с каналом *p*-типа, однако технология их изготовления сложнее. Например, большие трудности вызывает обеспечение воспроизводимых значений пороговых напряжений транзисторов с каналами *n*-типа. В процессе изготовления *n*-МДП БИС на поверхности кремния возникает инверсионный слой и образуются транзисторы со встроенным каналом. Эффективным методом предотвращения возникновения в МДП-транзисторах *n*-типа встроенного канала является подача на подложку запирающего отрицательного напряжения 3—5 В, что обеспечивает сдвиг порогового напряжения до требуемого значе-

ния. БИС на МДП-транзисторах n -типа требуют для своей работы трех источников питания (например, К580ИК80): +5 В для питания выходных каскадов и обеспечения совместимости уровней выходных сигналов с уровнями схем ТТЛ-типа; +12 В для питания внутренних каскадов БИС, например для запитки затворов нагрузочных транзисторов для реализации квазилинейной нагрузки [1]; -5 В для смещения подложки. Для нормальной работы n -МДП БИС необходимо в первую очередь подавать запирающее напряжение -5 В на подложку, а затем +5 и +12 В. Порядок выключения источников питания обратный. По n -МДП-технологии выпускаются БИС серий К580, К581 и К586.

Наиболее сложной является технология, по которой создают КМДП БИС (комплементарные БИС). Достоинством таких схем является то, что в статике они не потребляют ток от источника питания, сохраняют работоспособность в широком диапазоне изменения напряжения питания. Помехоустойчивость КМДП схем достигает 40 % от напряжения питания. По КМДП-технологии изготавливаются БИС серий К587, К588.

Основное преимущество биполярных схем — высокое быстродействие, оптимизировать которое для какого-либо микропроцессорного приложения можно только в том случае, если разработчик определяет структуру магистралей, размер слова и набор команд машины. Необходимость предоставления разработчику подобных возможностей привела к тому, что в области биполярных БИС развивается направление процессорных секций, которое в значительной степени отличается от более универсальной байтовой ориентации менее быстродействующих МДП-микропроцессоров. Примерами являются комплект БИС серий К589, КР1804 и расширяющий их функциональные возможности комплект КР1802. Все комплекты выполнены по ТТЛШ-технологии.

Самыми быстродействующими являются логические схемы с эмиттерными связями (ЭСЛ), на основе которых выполняется БИС серии КР1800. Высокое быстродействие этой серии обеспечивается главным образом благодаря новым схемным решениям, а не каким-либо коренным усовершенствованиям технологии ИС. Примером является внутренняя низкоуровневая логика, работающая от источника — 2 В, более подходящего, чем источник — 5 В, для работы мультиплексоров, регистров и других широко применяемых логических узлов. Тем самым обеспечивается легкая интеграция этих элементов с такими схемными узлами, как сумматоры, которые гораздо легче строить на структурах ЭСЛ с последовательными вентилями, запиываемыми от стандартного источника — 5 В.

Перспективным направлением в области развития биполярных БИС являются *интегральные инжекционные логические схемы* (I^2L), иногда называемые совмещенной транзисторной логикой или схемами с инжекционным питанием. Отличие схем I^2L от других биполярных схем заключается в том, что в них используется совмещение электрически связанных однородных областей полупроводника в одном кристалле, а традиционный способ питания цепей

базы и коллектора транзисторов через резисторы заменен непосредственным введением избыточных подвижных носителей в базу переключаемых транзисторов. Большинство кристаллов БИС, выполненных по И²Л-технологии, имеют по периферии схемы согласования уровней, обеспечивающие их стыковку со схемами ТТЛ. Микросхемы МПК серии К584 выполнены по И²Л-технологии, совместимы по электрическим параметрам с серийными интегральными схемами ТТЛ, ТТЛШ и обладают средним быстродействием между БИС *n*-МДП и ТТЛШ-технологией. По И²Л-технологии выполнены серии К582 и К583.

Принципиальные схемы базовых логических элементов, выполненных по рассмотренным выше технологиям, рассмотрены в [9].

Базовые схемы и статические характеристики малой (ИС) и средней (СИС) степени интеграции. Под *цифровыми элементами* в дальнейшем подразумеваются изделия или части изделий, реализующие функцию или систему функций алгебры логики (например, интегральная микросхема, микросборка, набор компонентов или их совокупности, а также генераторы, усилители, коммутаторы и т. п.).

При разработке и создании микроЭВМ необходимо выполнить специальный расчет характеристик цифровых элементов и их совместимости, особенно если в микроЭВМ используются разные типы логических элементов. В большинстве микроЭВМ все еще широко применяют интегральные схемы малой и средней степеней интеграции, среди которых наиболее распространены схемы типов ТТЛ (ТТЛШ), ЭСЛ и КМДП. Рассмотрим подробнее базовые схемы этих элементов и их статические характеристики.

Элементы ТТЛ-типа. Базовая схема элемента ТТЛ-типа серии 155 приведена на рис. 1.40, а. В схемах ТТЛШ-типа все транзисторы выполняются с диодами Шотки (рис. 1.40, б), благодаря чему устраняются этапы накопления и рассасывания избыточных зарядов и существенно уменьшаются задержки распространения сигналов. При

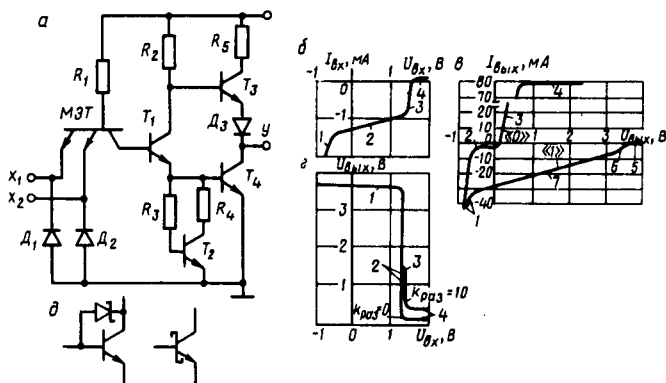


Рис. 1.40. Схема базового логического элемента ТТЛ-типа (а), статические характеристики (б — входная, в — выходная, г — передаточная), условное графическое обозначение транзистора Шотки (д).

той же мощности схемы ТТЛШ-типа имеют задержки в 5—7 раз меньше, чем схемы ТТЛ-типа. Элемент ТТЛ-типа реализует функцию И — НЕ, т. е. $y = \overline{x_1 x_2}$. Резистор R_1 и многоэмиттерный транзистор МЭТ образуют логический элемент И, антизвонные диоды D_1 и D_2 улучшают переходные процессы в линиях связи, а остальные компоненты являются сложным инвертором, способным отдать в нагрузку и принять от нагрузки большие токи.

На входной характеристике (рис. 1.40, б) можно выделить четыре участка. Участок 1 определяется вольт-амперной характеристикой антизвонного диода. Участок 2, соответствующий диапазону $-0,5 \text{ В} < U_{\text{вх}} < +1,2 \text{ В}$, определяет входной ток элемента ТТЛ-типа при низком входном напряжении ($U_{\text{вх}} < U_{\text{пор}} \approx 1,2 - 1,3 \text{ В}$).

При $U_{\text{вх}} = U_{\text{вх}}^0$

$$I_{\text{вх}}^0 = [(U_{\text{нп}} - U_{\text{бэм}} - U_{\text{вх}}^0) / R_1] [1 + (R_{\text{об}} - 1)\beta_i]. \quad (1.36)$$

Если прямую, соответствующую участку 2, продлить до пересечения с осью $U_{\text{вх}}$, то она пересечет ее при числовом значении $U_{\text{вх}}$, равном $U_{\text{нп}} - U_{\text{бэм}}$, где $U_{\text{бэм}} \approx 0,6 - 0,7 \text{ В}$. Динамическое входное сопротивление на участке 2 равно R_1 . Участок 3 соответствует переключению тока базы многоэмиттерного транзистора между его эмиттером и коллектором. Динамическое входное сопротивление на участке 3 составляет десятки ом. Входное напряжение, соответствующее середине участка 3, называется *входным пороговым напряжением* $U_{\text{пор. вх.}}$. При анализе некоторых схем, выполненных на элементах ТТЛ-типа, например разностных преобразователей, элементов задержки и т. п. [9], считают, что $U_{\text{вх}} < U_{\text{пор. вх}}$ воспринимается как «0», а $U_{\text{вх}} > U_{\text{пор. вх}}$ — как «1». На участке 4 МЭТ работает в инверсном активном режиме. Этот участок определяет $I_{\text{вх}}^1$:

$$I_{\text{вх}}^1 = \beta_i I_{R_1 \text{ max}}, \quad (1.37)$$

где

$$I_{R_1 \text{ max}} = (U_{\text{нп}} - U_{\text{бэм}} - U_{\text{вх}}^0) / R_1 \quad (1.38)$$

($U_{\text{вх}}^0$ подано на любой эмиттер МЭТ).

Передаточная характеристика элемента ТТЛ-типа приведена на рис. 1.40, г. На ней можно выделить также четыре участка. Участок 1 соответствует формированию на выходе напряжения «1»:

$$U_{\text{вых}}^1 = U_{\text{нп}} - [R_2 / (1 + \beta)] I_{\text{н}} - U_{\text{бэТз}} - U_{\text{Дз}}, \quad (1.39)$$

где $I_{\text{н}}$ — ток нагрузки.

Неизменяемость $U_{\text{вых}}^1$ при $U_{\text{вх}} < +1,2 \text{ В}$ объясняется использованием в цепи эмиттера T_1 схемы, состоящей из резисторов R_3 , R_4 и транзистора T_2 . Участок 2 соответствует активному режиму работы транзисторов $T_1 - T_4$. На этом участке элемент ТТЛ-типа потребляет от источника $U_{\text{нп}}$ ток до 30 мА. По этой причине во избежание больших помех в аппаратуре, построенной на схемах ТТЛ-типа, необходимо создавать цепи питания с малой индуктивностью проводников, предусматривать развязку между соседними узлами и организовывать

вать время нарастания и спада входных импульсов до 150 нс (не больше). Участок 3 проявляется на передаточной характеристике только для нагруженной аналогичными элементами схемы ТТЛ-типа. Наличие этого участка определяется низким динамическим сопротивлением на участке 3 входной характеристики, так как выходное напряжение управляющей схемы является входным для нагрузочных схем. Участок 4 на передаточной характеристике соответствует насыщенному режиму транзистора T_4 . На этом участке выходное напряжение

$$U_{\text{вых}}^0 = U_{\text{ост}} + [r_k + 4m\varphi_T / (\beta I_{64})] I_n, \quad (1.40)$$

где $U_{\text{ост}}$ — остаточное напряжение на промежутке коллектор — эмиттер при $I_k = 0$; r_k — объемное сопротивление тела коллектора; $4m\varphi_T / (\beta I_{64})$ — динамическое сопротивление промежутка коллектор — эмиттер для насыщенного режима [9].

Так как элемент ТТЛ-типа может иметь два состояния выхода, надо рассмотреть две соответствующие выходные характеристики.

Рассмотрим выходную характеристику (рис. 1.40, в) для случая, когда транзистор T_3 закрыт (схема ТТЛ-типа воспринимает ток от нагрузки). Участок 1 определяется вольт-амперной характеристикой диода «коллектор T_4 — подложка». Участок 2 характеризует инверсный активный режим работы транзистора T_4 (рис. 1.40, а), а участок 3 — насыщенный режим его работы. Для участка 3 справедливо соотношение (1.40), так как $I_{\text{вых}} = I_n$. Участок 4 соответствует активному режиму работы транзистора T_4 . В данном случае $I_{\text{вых}} = I_{k4} \approx \beta I_{64}$, причем для серии 155/64 $\approx 2 - 3$ мА. В этом режиме транзистор кратковременно находится во время формирования спада t^{10} . В статике этот режим — аварийный и недопустимый.

Рассмотрим теперь выходную характеристику для случая, когда T_4 закрыт (схема ТТЛ-типа отдает ток в нагрузку). Эта характеристика содержит участок 1, рассмотренный выше. На участке 7 (рис. 1.40, в) транзистор T_3 (рис. 1.40, а) работает в насыщенном режиме ($I_n > 6 - 7$ мА). Выходное сопротивление на участке 7 приблизительно будет $R_2 || R_5 \approx 130$ Ом. Участок 6 соответствует активному режиму работы транзистора T_3 ($I_n < 6 - 7$ мА). Выходное сопротивление на участке 6 приблизительно равно $R_2 / (1 + \beta)$ (около 50 Ом). На участке 5 при $3,6 \text{ В} < U_{\text{вых}} < 5 \text{ В}$ транзистор T_3 работает в активном режиме при малых токах, а при $U_{\text{вых}} > 5 \text{ В}$ (точнее, $> U_{\text{ин}}$) — в режиме отсечки. При $U_{\text{вых}} > 7 - 10 \text{ В}$ пробивается коллекторный переход транзистора T_4 . Это аварийный режим. Как видно из выходной характеристики, схема элемента ТТЛ-типа способна отдавать в нагрузку ток до 25—30 мА, однако при $I_n = 10 - 15$ мА к выходу элемента ТТЛ-типа нельзя подключать другие элементы ТТЛ-типа, так как при таких токах нагрузки $U_{\text{вых}}$ становится меньше $U_{\text{мин}} = 2,4 \text{ В}$.

Элементы ЭСЛ-типа. Элементы ЭСЛ-типа — самые быстродействующие современные логические элементы благодаря: активному режиму работы транзисторов в обоих логических состояниях; низкому логическому перепаду, из-за чего уменьшается время заряда и разряда собственных емкостных составляющих схем; использованию

на выходах элемента эмиттерных повторителей, обеспечивающих значительные токи для перезаряда емкостной составляющей нагрузки. Основное назначение эмиттерных повторителей — смещение выходных напряжений для обеспечения совместимости входных и выходных уровней «0» и «1». Важное достоинство элементов ЭСЛ-типа — наличие инверсных выходов, реализующих логическую функцию и ее отрицание (рис. 1.41). Для выполнения равенства $U_{\text{вых},y}^0 = U_{\text{вых},\bar{y}}^0$ обеспечивают соотношение $R_{K1} = 0,86 R_{K2}$.

Входная характеристика элемента ЭСЛ-типа приведена на рис. 1.41, в. На ней можно выделить четыре участка. На участке 1 входной транзистор T_2 (рис. 1.41, а) работает в режиме отсечки при $U_{\text{вх}} < -1,9$ В или в активном режиме при малых токах, когда $-1,9 \text{ В} < U_{\text{вх}} < -1,3$ В. На участке 2 при $-1,3 \text{ В} < U_{\text{вх}} < -1,1$ В происходит переключение тока, проходящего через R_3 , между входным транзистором T_2 и транзистором T_3 . Этот участок характеризуется динамическим входным сопротивлением ≈ 1 кОм. На участке 3 входной транзистор T_2 работает в активном режиме в составе эмиттерного повторителя T_2 ; R_3 . Динамическое входное сопротивление в этом случае равно $(1 + \beta) R$ и составляет ≈ 30 кОм. Участок 4

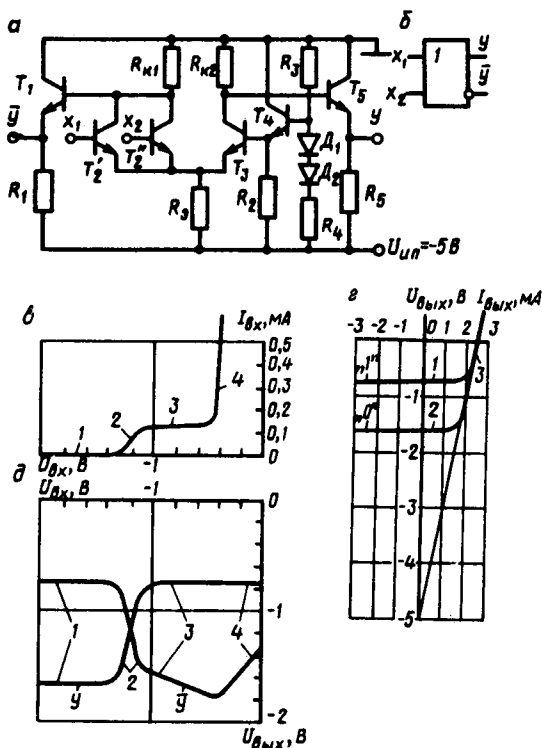


Рис. 1.41. Схема базового логического элемента ЭСЛ-типа (а), его условное графическое обозначение (б) и статические характеристики ЭСЛ-типа (в — входная; г — выходная; д — передаточная).

характеризуется насыщенным режимом транзистора T_2 . Динамическое входное сопротивление на этом участке равно $R_{к1} \parallel R_3$. В нормальном режиме рабочая точка никогда не попадает на участок 4.

Передаточная характеристика элемента ЭСЛ-типа приведена на рис. 1.41, д для прямого и инверсного выходов. На этих характеристиках также можно выделить четыре участка. Для участка 1 характерно следующее: транзистор T_2 — в режиме отсечки или в активном режиме при малых токах, транзистор T_3 — в активном режиме, причем через него протекает практически весь переключаемый ток. На участке 2 транзисторы T_2 и T_3 работают в активном режиме и происходит переключение тока между ними. На участке 3 транзистор T_2 работает в активном режиме, причем через него проходит практически весь переключаемый ток, а транзистор T_3 — в активном режиме при малых токах. Участок 4 соответствует насыщенному режиму работы транзистора T_2 и режиму отсечки для транзистора T_3 .

Выходные характеристики для двух состояний выходов элемента ЭСЛ-типа приведены на рис. 1.41, г. Участок 1 или 2 соответствует активному режиму работы выходного транзистора при формировании уровня «1» или «0» на выходе, участок 3 — режиму отсечки выходных транзисторов. Этот участок определяется вольт-амперной характеристикой резисторов R_1 или R_5 , подключенных к источнику питания $U_{нп} = -5$ В. Для участков 1 и 2 $R_{вых} = R_{к1(2)}/(1 + \beta) \parallel R_1$ (или R_5), а для участка 3 — R_1 или R_5 .

Элементы КМДП-типа. Основой всех логических элементов КМДП-типа является инвертор со схемой защиты (рис. 1.42, а). Структура КМДП инвертора (без схемы защиты) приведена на рис. 1.42, б. Статистические характеристики инвертора КМДП-типа можно построить, используя уравнения Хофстайна для униполярных транзисторов:

$$I_c = K[(U_{зи} - U_{пор})U_{си} - U_{си}^2/2]; \quad (1.41)$$

$$U_{си \text{ нас}} = U_{зи} - U_{пор}; \quad (1.42)$$

$$I_c = (K/2)(U_{зи} - U_{пор})^2, \quad (1.43)$$

где I_c — ток стока; $U_{зи}$ — напряжение на затворе относительно истока; $U_{пор}$ — пороговое напряжение на затворе относительно истока; $U_{си}$ — напряжение на стоке относительно истока; K — удельная крутизна транзистора.

Уравнение (1.41) определяет зависимость I_c от $U_{си}$ в крутой области характеристики. Напряжение $U_{си \text{ нас}}$ — граница между крутой и пологой областями характеристики.

Уравнение (1.43) определяет пологую область выходной характеристики $I_c = f(U_{си})$, в которой I_c не зависит от $U_{си}$.

Входная характеристика инвертора КМДП-типа определяется цепью защиты по входу и имеет три участка (рис. 1.42, г). На участке 1 входная характеристика является вольт-амперной характеристикой последовательной цепи R_1, D_3 . Участок 2 ($-0,5 \text{ В} < U_{вх} < U_{нп} + 0,5 \text{ В}$) определяется обратными токами диодов D_1 — D_3 и токами утечки. На участке 3 входная характеристика является

вольт-амперной характеристикой диода D_1 , начало координат которой смещено в точку $U_{вх} = U_{ин}$.

Передаточные характеристики приведены на рис. 1.42, в при двух значениях $U_{ин}$ для схемы с транзисторами, имеющими параметры $U_{пор n} = +2$ В, $K_n = 0,2$ мА/В², $U_{пор p} = -1,5$ В, $K_p = 0,1$ мА/В². На характеристике можно выделить пять участков.

На участке 1 транзистор n -типа закрыт, а транзистор p -типа работает в пологой области. На участке 2 транзистор n -типа работает в крутой области, а транзистор p -типа — в пологой. На участке 3 оба транзистора работают в пологой области, поэтому выходное напряжение здесь не определено, а соответствующее входное пороговое напряжение можно определить из уравнений Хофштайна:

$$U_{вх. пор} = [\sqrt{K_n} U_{пор n} + \sqrt{K_n}(U_{ин} + U_{пор p})] / (\sqrt{K_n} + \sqrt{K_p}). \quad (1.44)$$

При указанных выше параметрах получим $U_{вх. пор 2} = 4,7$ В, $U_{вх. пор 1} = 2,6$ В. На участке 4 транзистор n -типа работает в пологой области, а транзистор p -типа — в крутой. На участке 5 транзистор n -типа работает в пологой области, а транзистор p -типа закрыт.

Для улучшения передаточной характеристики в некоторых элементах КМДП-типа применяют на выходе буферный усилитель,

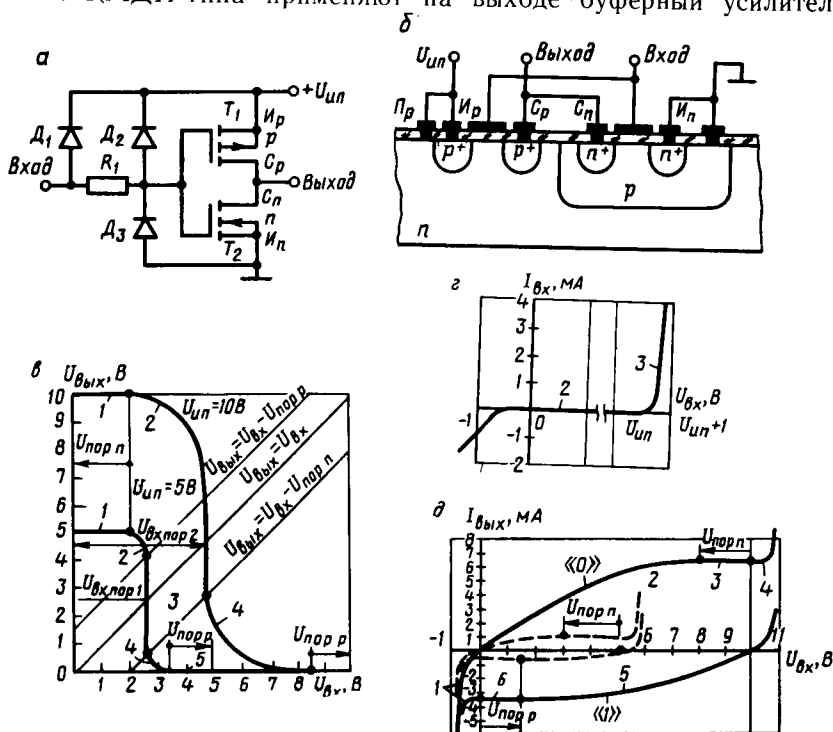


Рис. 1.42. Схема (а), структура (без схемы защиты по входу) (б), статические характеристики [в — передаточная ($U_{ин} = 5-10$ В), г — входная, д — выходная ($U_{ин} = 5-10$ В)] инвертора КМДП-типа.

представляющий собой два последовательно включенных инвертора.

Выходные характеристики инвертора КМДП-типа приведены на рис. 1.42, *б* для двух напряжений $U_{ин}$ для транзисторов, параметры которых указаны выше, и для двух состояний выхода инвертора. Выходные характеристики имеют четыре участка. Участок 1 определяется вольт-амперной характеристикой диода структуры $C_n P_p$. Участок 2 является крутой областью вольт-амперной характеристики транзистора n -типа при $U_{зи} = U_{ин}$ и занимает область $-0,5 \text{ В} < U_{вых} < U_{ин} - U_{пор\ n}$. Участок 3 является пологой областью транзистора n -типа при $U_{зи} = U_{ин}$ и занимает область $U_{ин} - U_{пор\ n} < U_{вых} < U_{ин} + 0,5 \text{ В}$. Участок 4 определяется вольт-амперной характеристикой диода структуры $C_p P_n$. Участок 5 аналогичен участку 2, а участок 6 — участку 3.

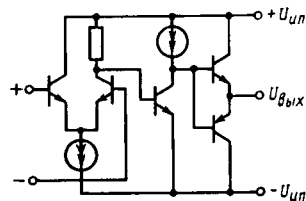
Для элементов КМДП-типа недопустима перемена полярности источника питания, так как к нему подключается прямосмещенный диод структуры $P_n P_p$. К достоинствам элементов КМДП-типа можно отнести нечувствительность короткого замыкания их выхода к шинам «Земля» и «Питание».

1.5. Элементная база аналоговых устройств средств сопряжения

Операционные усилители. Дифференциальные операционные усилители (ОУ) служат основой для построения большинства устройств, предназначенных для преобразования аналоговых сигналов. Они обеспечивают усиление постоянного и переменного напряжений положительной и отрицательной полярности.

Как правило, дифференциальный усилитель komponуется из трех каскадов: входного дифференциального, промежуточного с большим коэффициентом усиления по напряжению и выходного повышенной мощности. Упрощенная схема дифференциального усилителя приведена на рис. 1.43.

Рис. 1.43. Схема дифференциального операционного усилителя.



Схемотехнические варианты построения каскадов дифференциальных усилителей и погрешности, вносимые отдельными каскадами, подробно изложены в монографиях.

С точки зрения использования усилителей в системах сбора информации большой интерес представляют характеристики усилителей и основные схемы их включения.

Классифицировать дифференциальные операционные усилители можно по схемотехнической реализации дифференциального каскада, а также по использованию устройств, в которые включают схему усилителя с целью получения улучшенных рабочих параметров:

На биполярных транзисторах

На полевых транзисторах

МДП

С каналом МДМ-типа

Широкий диапазон применения (например, 140УД1, 5, 6, 9, 11, 12, 14, 20; 153УД1, 2, 3, 5, 6; 154УД1, 2, 3)

Высокое входное сопротивление (например, 140УД8, 574УД1, 544УД1, 2), что является основным фактором применения в системе сбора для согласования с датчиками первичной информации

Высокое входное сопротивление, малый входной ток, малое смещение нуля, небольшой коэффициент усиления (например, 140УД13)

Минимальное напряжение смещения нулевого уровня (например, 140УД13)

Основные характеристики дифференциального операционного усилителя, являющиеся данными для определения возможности применения его в различных устройствах, характеризуют степень отличия реальных усилителей от его идеализированной модели. Идеализированный дифференциальный операционный усилитель (рис. 1.44) функционирует в соответствии с выражением $U_{\text{вых}} = K_y(U_{\text{вх}}^+ - U_{\text{вх}}^-)$, где $U_{\text{вх}}^+$, $U_{\text{вх}}^-$ — значения напряжения на неинвертирующем и инвертирующем входах, K_y — коэффициент усиления.

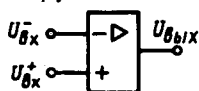


Рис. 1.44. Условное графическое обозначение идеализированного дифференциального операционного усилителя.

Для идеального усилителя напряжение на выходе равно нулю при $U_{\text{вх}}^+ - U_{\text{вх}}^- = 0$, входное сопротивление бесконечно велико, выходное сопротивление равно нулю, коэффициент усиления не зависит от частоты входных сигналов и $K_y \rightarrow \infty$.

Основные статические параметры дифференциальных усилителей следующие: $I_{\text{вх}}$ — входной ток, $\Delta I_{\text{вх}}$ — разность входных токов, $U_{\text{см}}$ — напряжение смещения нулевого уровня, $I_{\text{вых max}}$, $U_{\text{вых max}}$ — максимальные выходные ток и напряжения, K_y — коэффициент усиления напряжения на нулевой частоте, КОСС — коэффициент ослабления синфазного сигнала, $U_{\text{пл}}$ — напряжение питания, $I_{\text{ип}}$ — потребляемый ток.

Основные динамические параметры: $f_{\text{ср}}$ — частота среза; f_1 — частота, при которой коэффициент усиления $K_y = 1$; $V_{U_{\text{вых max}}}$ — максимальная скорость нарастания напряжения на выходе.

Свойства дифференциального ОУ, охваченного обратной связью, определяются как цепями обратных связей, так и свойствами усилителя. Принцип обратной связи состоит в том, чтобы выделить на выходе какого-либо устройства часть получаемого напряжения и ввести его вновь на вход усилителя. Если на вход усилителя приложено напряжение, то реализация обратной связи сводится к включению между источником сигнала и входом усилителя идеального источника, напряжение которого равно части напряжения на зажимах нагрузки, в соответствии с данным определением:

$$U_{\text{вх}} = \alpha U_c + \beta U_{\text{н}}, \quad (1.45)$$

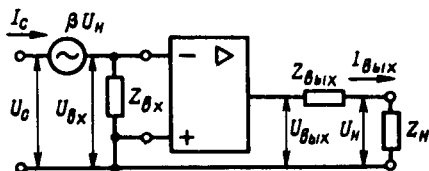
где U_c — напряжение источника сигнала; U_n — напряжение на зажимах нагрузки; α — коэффициент прямой передачи напряжения U_c ; β — коэффициент обратной передачи напряжения U_n .

Обобщенная схема ОУ с обратной связью приведена на рис. 1.45, откуда

$$U_n = K_y(\alpha U_c + \beta U_n) - R_{\text{вых}} I_n; \quad (1.46)$$

$$U_n = U_c \alpha K_y / (1 - \beta K_y) - [R_{\text{вых}} / (1 - \beta K_y)] I_n. \quad (1.47)$$

Рис. 1.45. Структура дифференциального операционного усилителя с обратной связью.



Обозначим $K_{\text{ос}} = \alpha K_y / (1 - \beta K_y)$, $R_{\text{вых.ос}} = R_{\text{вых}} / (1 - \beta K_y)$, где $K_{\text{ос}}$, $R_{\text{вых.ос}}$ — коэффициент передачи и выходное сопротивление усилителя, охваченного обратной связью. При

$$\beta K_y \gg 1; K_{\text{ос}} \approx \alpha / \beta \quad (1.48)$$

входное сопротивление усилителя с обратной связью $R_{\text{вх.ос}} = U_c / I_c$ или $I_c = I_{\text{вх}} = U_{\text{вх}} / R_{\text{вх}}$, откуда

$$R_{\text{ос}} = (U_c / U_{\text{вх}}) R_{\text{вх}}; U_c = (U_{\text{вх}} - \beta U_n) / \alpha;$$

$$U_n = U_{\text{вых}} - R_{\text{вых}} I_{\text{вых}}; U_{\text{вых}} = K_y U_{\text{вх}};$$

$$R_{\text{вх.ос}} = [(1 - \beta K_y) / \alpha + (\beta / \alpha) (R_{\text{вых}} I_{\text{вых}}) / U_{\text{вх}}] R_{\text{вх}}.$$

Так как $I_{\text{вх}} = K_y U_{\text{вх}} / (R_{\text{вых}} + R'_n)$, где R'_n — эквивалентная нагрузка ОУ, учитывающая как R_n , так и элементы для реализации β , то

$$R_{\text{вх.ос}} = \left(\frac{1 - \beta K_y}{\alpha} + \frac{\beta K_y}{\alpha} \frac{R_{\text{вых}}}{R_{\text{вых}} + R_n} \right) R_{\text{вх}}. \quad (1.49)$$

Выходное сопротивление усилителя с обратной связью

$$R_{\text{вых.ос}} = R_{\text{вых}} / (1 + K_y \beta). \quad (1.50)$$

Анализируя выражения для $K_{\text{ос}}$, $R_{\text{вх.ос}}$, $R_{\text{вых.ос}}$, можно отметить следующие положительные стороны введения отрицательной обратной связи по напряжению:

независимость коэффициента $K_{\text{ос}}$ от коэффициента K_y ;

всегда $R_{\text{вых.ос}} \ll R_{\text{вых}}$;

при определенных условиях $R_{\text{вх.ос}} \gg R_{\text{вх}}$;

улучшена линейность коэффициента передачи, так как $dK_{\text{ос}} / K_{\text{ос}} = (1 - [1 / (1 + K_y \beta)]) (dK_y / K_y) < dK_y / K_y$; расширен диапазон входных напряжений $U_c = U_{\text{вх}} (1 + K_y \beta) > U_{\text{вх}}$.

На основе (1.48) возможно построить три основные конфигурации отрицательных обратных связей и соответствующие схемы усилителей.

Последовательная отрицательная обратная связь (в неинвертирующем усилителе, рис. 1.46, а) — при $\alpha=1$, $\beta=R_1/(R_1+R_0)$; $K_{oc}=1/\beta$ напряжение

$$U_{вых} = -K_{oc} U_{вх}. \quad (1.51)$$

Из (1.51), если $R_1=\infty$, $K_{oc}=1$ и не зависит от R_0 , формируется схема повторителя $U_{вых}=U_{вх}$ (рис. 1.46, б).

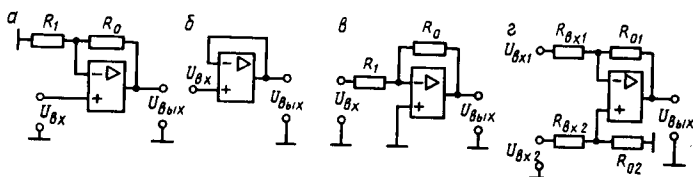


Рис. 1.46. Схемы реализации отрицательных обратных связей (ООС): а — последовательная ООС — неинвертирующий усилитель; б — последовательная ООС — повторитель; в — параллельная ООС — инвертирующий усилитель; г — комбинированная ООС — вычитатель.

Параллельная отрицательная обратная связь (в инвертирующем усилителе, рис. 1.46, в) — при $\alpha=R_0/(R_0+R_1)$; $\beta=R_1/(R_0+R_1)$; $K_{oc}=-\alpha/\beta$ напряжение

$$U_{вых} = -K_{oc} U_{вх}. \quad (1.52)$$

Комбинированная отрицательная обратная связь $U_{вых} = U_{вых1} K_{oc1} + U_{вых2} K_{oc2}$ (в разностном усилителе, рис. 1.46, г) — при $K_{oc1}=\alpha_1/\beta_1$; $\alpha_1=R_{02}/(R_{02}+R_{вх2})$; $\beta_1=R_{вх1}/(R_{01}+R_{вх1})$; $K_{oc2}=-\alpha_2/\beta_2$; $\alpha_2=R_{01}/(R_{01}+R_{вх1})$; $\beta_2=R_{вх1}/(R_{01}+R_{вх1})$ напряжение

$$U_{вых} = U_{вх1} (1 + R_{01}/R_{вх1}) [R_{02}/(R_{02} + R_{вх2})] - U_{вх2} R_{01}/R_{вх1}.$$

При $R_{01}=R_{вх1}$ и $R_{02}=R_{вх2}$ напряжение

$$U_{вых} = U_{вх1} - U_{вх2}.$$

Анализ схем ОУ с обратными связями позволяет определить основные параметры для выбора варианта построения преобразователя аналоговых сигналов систем сбора информации.

Неинвертирующие усилители характеризуются тем, что $R_{вх} \rightarrow \infty$; $R_{вых} \rightarrow 0$, и по этой причине используются для развязки источников сигналов малой мощности от входных цепей устройств сбора информации.

При анализе неинвертирующих схем надо учитывать следующие основные положения: а) разность потенциалов между входами дифференциального усилителя, работающего в линейном режиме, равна нулю; б) токи через входные зажимы операционного усилителя отсутствуют.

Входное сопротивление инвертирующих ОУ равно $R_{вх1}$, т. е. сопротивлению резистора на входе. Подобные схемы включения ОУ реализуют необходимый коэффициент усиления по каналу обработки аналогового сигнала.

При анализе инвертирующих схем надо учитывать следующие

основные положения: а) потенциал суммирующей точки равен нулю; б) токи через входные зажимы усилителя отсутствуют; в) ток, попадающий в суммирующую точку из цепей входа, равен току, протекающему в цепи обратной связи.

Важным параметром ОУ, определяющим составляющую систематической погрешности преобразования аналоговых сигналов, является ЭДС смещения нулевого уровня $U_{см}$. Для схем с отрицательными обратными связями ЭДС смещения на выходе

$$U_{вых. см} = U_{см} / \beta.$$

Это выражение справедливо как для инвертирующих, так и для неинвертирующих ОУ.

Эффективным средством уменьшения ЭДС $U_{вых. см}$ при построении высокоточных преобразователей аналоговых сигналов является использование ОУ с каналом МДМ-типа (рис. 1.47).

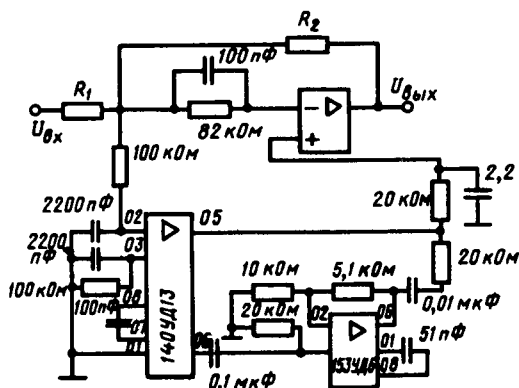


Рис. 1.47. Пример схемы дифференциального операционного усилителя с каналом МДМ-типа.

Входные токи ОУ, проходя по цепям входа и цепи обратной связи, приводят к появлению на выходе составляющей общей погрешности. Например, для схемы операционного усилителя в инвертирующем режиме (см. рис. 1.46, в) можно записать

$$U_{вых/вх} = I_{вх1} R_0 - I_{вх2} R_2 (1 + R_0 / R_1).$$

Так как для операционных усилителей $|I_{вх1}| \approx |I_{вх2}| \approx I_{вх}$, то при $R_2 = R_0 R_1 / (R_0 + R_1)$ получим

$$U_{вых \Delta / вх} = \Delta I_{вх} R_0,$$

где $\Delta I_{вх}$ — разность входных токов (нормируемый параметр).

Важным свойством дифференциального усилителя является возможность подавления синфазной помехи, которая характеризуется коэффициентом ослабления синфазного сигнала (КОСС). Идеальный дифференциальный усилитель усиливает только разностное напряжение $U_{вх1} - U_{вх2}$, и напряжение на выходе не зависит от синфазного

напряжения на входе. Однако в реальных усилителях под действием синфазного напряжения на входе $U_{с. вх}$ возникает выходное напряжение $U_{с. вых}$. Так как коэффициент усиления синфазного сигнала

$$K_c = U_{с. вых} / U_{с. вх}$$

то коэффициент его ослабления

$$KOCC = K_{диф} / K_c$$

где $K_{диф}$ — коэффициент усиления дифференциального сигнала.

Напряжение на выходе ОУ

$$U_{вых} = K_{диф} E_{диф} + K_c E_c = K_{диф} E_{диф} [1 + (1/KOCC)(E_c/E_{диф})],$$

где $(1/KOCC)(E_c/E_{диф})$ — синфазная ошибка.

Динамические характеристики дифференциальных усилителей характеризуют в первую очередь АЧХ. Типичная АЧХ усилителя имеет три излома (рис. 1.48). Первый излом находится на частоте ~ 10 Гц и определяется частотными характеристиками промежуточного каскада. Второй излом находится на частоте ~ 1 МГц

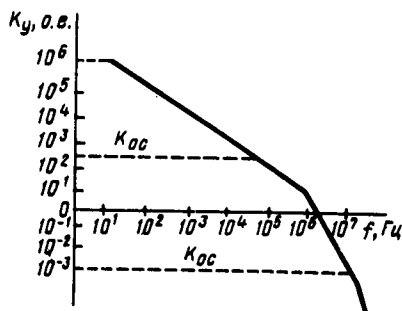


Рис. 1.48. Типичная АЧХ дифференциального усилителя.

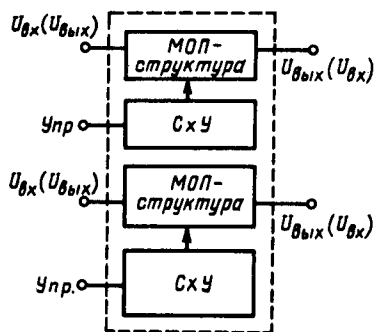


Рис. 1.49. Структура интегрального переключателя серии 143КТ1.

и обусловлен частотными параметрами дифференциального каскада. Третий излом связан с АЧХ выходного каскада и находится примерно на частоте 50 МГц. На этой характеристике легко выделить возможность расширения частотного диапазона при введении обратных связей. В паспорте на усилитель указывают частоту среза $f_{ср}$ на уровне — 3 дБ. Отношение частоты f_1 , при которой $K_y = 1$, к частоте $f_{ср}$ выражается соотношением $f_{ср} = f_1/\beta$. Поэтому ширина полосы при замкнутой обратной связи пропорциональна β . Скорость нарастания выходного напряжения $V_{вых}$ будет максимальной при передаче входного сигнала максимальной амплитуды.

Аналоговые ключи и коммутаторы. Аналоговые ключи и коммутаторы используются в системах сбора информации для обработки аналоговых сигналов, поступающих по параллельным каналам

для преобразования в цифровые сигналы. Число каналов может достигать нескольких сотен, но максимальное значение определяется необходимой частотой опроса датчиков первичной информации, которая выбирается исходя из максимальной частотной составляющей, содержащейся в аналоговом сигнале. Методы организации переключателей аналоговых сигналов с большим числом каналов связаны с реализацией многоступенчатых переключателей.

На рис. 1.49, где СхУ — схема управления, показана структура ключа серии 143КТ1, выполненного на основе МДП-структур с каналом p -типа. Напряжение источников питания $U_{\text{ип1}} = +5 \text{ В}$, $U_{\text{ип2}} = -24 \text{ В}$. Входы схем управления СхУ 11, 12 согласованы с уровнями ТТЛ «1» и «0». Максимальное переключаемое напряжение $\pm 4 \text{ В}$, переходное сопротивление открытого ключа $\sim 100 \text{ Ом}$. Чтобы избежать падения части переключаемого напряжения на переходном сопротивлении, выходы ключей нагружают на нагрузку с высоким входным сопротивлением, например повторитель на операционном усилителе.

Компараторы. Компараторы — схемы сравнения аналоговых сигналов — по принципу функционирования относят к аналого-цифровым элементам, так как входные сигналы — непрерывные функции времени, а выходные — дискретные, принимающие одно из двух возможных состояний: «0» или «1». Для двухвходового компаратора $U_{\text{вых}} = f(U_{\text{вх1}}, U_{\text{вх2}})$, где $U_{\text{вых}} = \text{«1»}$ при $U_{\text{вх1}} < U_{\text{вх2}}$; $U_{\text{вых}} = \text{«0»}$ при $U_{\text{вх1}} > U_{\text{вх2}}$.

Компараторы в системах сбора и обработки информации используются при формировании схем АЦП.

Наиболее просто выражение реализуется на основе дифференциального операционного усилителя (рис. 1.50, а), где диод включен

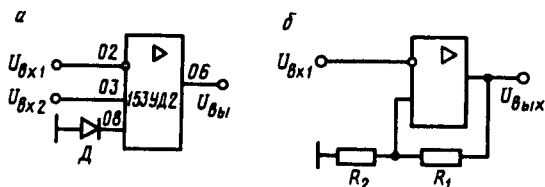


Рис. 1.50. Схема компаратора напряжений (а), схема компаратора напряжения с обратной связью (б).

в схему компаратора для согласования уровней «0» и «1» с входами ТТЛ.

Основными параметрами компараторов являются: а) пороговый ток или напряжение срабатывания; б) ширина зоны нечувствительности; в) время переключения из нулевого состояния в единичное; г) напряжение смещения нулевого уровня. Например, для компараторов 521СА3, 521СА4 соответственно $I_{\text{вх.ср}} = 100 \text{ нА}$; $t^{0,1} = 300 \text{ нс}$ и $I_{\text{вх.ср}} = 2 \text{ мкА}$; $t^{0,1} = 50 \text{ нс}$; $U_{\text{см}} = 4 \text{ мВ}$.

Практически во всех случаях использования компараторов необходимо исключать многократные срабатывания, т. е. переключения

$0 \rightarrow 1$, $1 \rightarrow 0$, вызванные наложением высокочастотных помех на аналоговые сигналы $U_{вх1}$, $U_{вх2}$ (рис. 1.51). Из рисунка видно, что, расширив зону нечувствительности компаратора на значение $U_{ср}$, можно исключить многократные изменения выходного состояния компаратора в окрестности точки равенства $U_{вх1} = U_{вх2}$.

Наиболее просто эта задача решается введением обратной связи (рис. 1.50, б): при $U_{вх1} < U_{пор. в}$ имеем $U_{вых} = +E_{нас}$; при $U_{вх1} > U_{пор. н}$ имеем $U_{вых} = -E_{нас}$, где $U_{пор. в} = R_2 E_{нас} / (R_1 + R_2)$ — напряжение

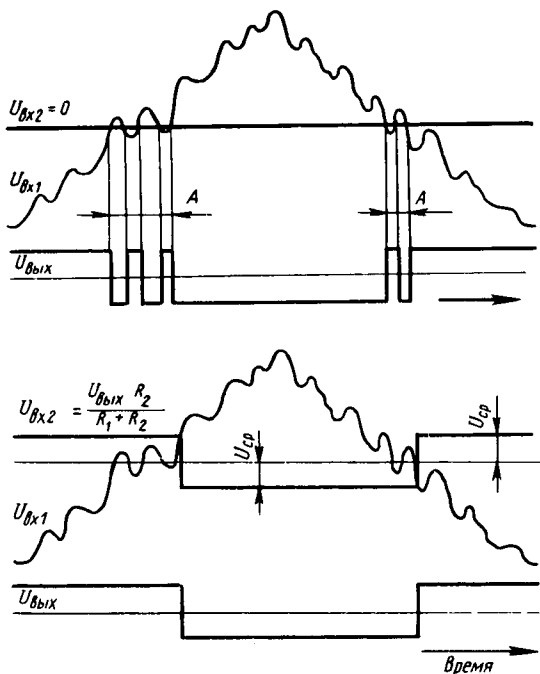


Рис. 1.51. Эпюра напряжения на выходе компаратора.

верхнего порога срабатывания компаратора; $U_{пор. н} = -R_2 E_{нас} / (R_1 + R_2)$ — напряжение нижнего порога срабатывания компаратора; $\pm E_{нас}$ — напряжение насыщения операционного усилителя.

Если $|U_{пор. в}| + |U_{пор. н}|$ превышает область А совмещения шумов $U_{вх1}$ и $U_{вх2}$, то введение положительной обратной связи (R_1 ; R_2) исключает ложные срабатывания компаратора и, следовательно, устраняет неопределенное состояние $U_{вых}$.

1.6. Узлы цифровых устройств средств сопряжения

Система синхронизации микроЭВМ. Синхронизация основана на обеспечении совпадения сигналов управления с периодическим сигналом, называемым тактовой последовательностью.

оператора или с блока наладки. В качестве ГСС используются любые самовозбуждающиеся генераторы, часто с кварцевой стабилизацией частоты. В последнее время широко распространены генераторы, построенные на логических вентилях и линиях задержки с отводами. Важным достоинством таких генераторов является слабая зависимость относительных временных соотношений от внешних факторов.

Частота следования тактовых сигналов определяется быстродействием, при котором отдельные части МП способны надежно выполнять заданные функции. Иногда надо принимать во внимание и временные параметры запоминающего устройства. Внутренний цикл процессора во избежание ненужных временных потерь может быть переменным (изменяется как частота, так и длительность тактовых сигналов). Для этой цели применяется управление модификацией синхросигналами СС.

Синхросигналы с ГСС поступают на блок разрешения БР ГСС, управляемый соответствующими разрядами микрокоманды. Выход-

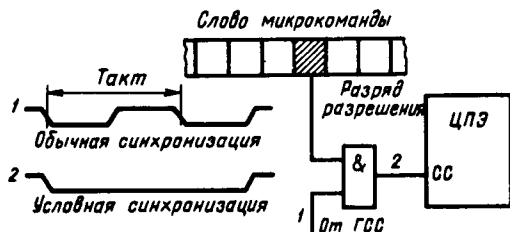


Рис. 1.53. Реализация условной синхронизации ЦПЭ K589ИК02.

ные сигналы генератора используются либо непосредственно, либо при наличии разрешающих сигналов, поступающих с соответствующих разрядов микрокоманды.

Примером управления синхросигналом является так называемая условная синхронизация, применяемая для центрального процессорного элемента серии K589. В этом случае центральный процессор синхронизируется не непосредственно от генератора, а через элемент И, на второй вход которого подается специальный дополнительный разряд микрокоманды (рис. 1.53). Кратковременная приостановка последовательности тактовых импульсов позволяет выполнить микрооперацию, но результаты микрооперации не передаются в аккумулятор или регистры общего назначения из-за отсутствия разрешения по входу синхросигнала. В то же время информационные сигналы, возникающие при операции и распространяющиеся по цепям межразрядных связей в комбинационном АЛУ, могут быть прочитаны блоком микропрограммного управления в качестве признаков условного перехода. Благодаря этому информация, хранимая в любом из регистров ЦПЭ, может быть подвергнута различным операциям проверки без ее перемещений в схеме ЦП.

В целом логика управления вырабатывает сигналы, которые

заставляют логические элементы, выполняющие перемещение данных, образовывать соответствующие связи внутри процессора.

На рис. 1.54, *а* показан пример однофазной последовательности синхросигналов с активным положительным перепадом, а на рис. 1.54, *б* — однофазной последовательности, которая, по существу, двухфазная, так как у нее активными являются и положительный и отрицательный перепады. Например, в ИС К589ИК01 по отрицательному фронту синхросигнала результат операции АЛУ либо помещается в аккумулятор, либо записывается в выбранный регистр СОЗУ. С положительным фронтом синхросигнала подается новая микрокоманда. На рис. 1.54, *в, г* соответственно показаны

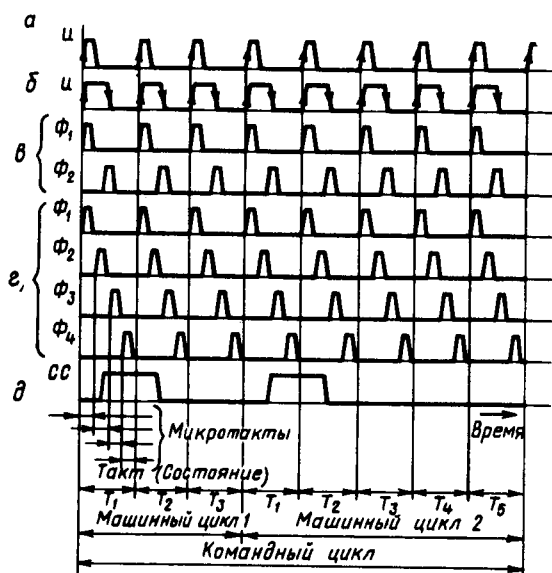


Рис. 1.54. Синхросигналы системы синхронизации.

двух- и четырехфазные последовательности СС. В микропроцессорах обычно предусматривается выход синхросигнала (СС) (рис. 1.54, *д*) для синхронизации работы периферийных устройств с работой МП, дающего информацию о начале нового машинного цикла.

Машинный такт — интервал времени, в течение которого выполняется элементарная часть операции, т. е. время, необходимое для передачи информации от одного регистрового узла к другому, а также (при необходимости) для выполнения некоторой операции над этими данными в процессе передачи. В многофазной системе синхронизации такты разбиваются на несколько микротактов.

Машинный цикл — интервал времени, в течение которого выполняется операция в целом. Машинный цикл содержит от одного до нескольких тактов.

Командный цикл — интервал времени, в течение которого выполняется команда. Он содержит от одного до нескольких машинных циклов.

Требования к параметрам синхросигналов зависят от конкретной конфигурации системы, построенной на основе комплекта БИС микропроцессорного набора. Длительности циклов выполнения операций рассчитываются при наихудших динамических параметрах микросхем в требуемом диапазоне рабочих температур, при заданном допуске на напряжение источника питания и с учетом применения микросхем малой и средней степеней интеграции, не входящих в комплект БИС микропроцессорного набора.

На рис. 1.55 приведен один из возможных вариантов построения двух синхросигналов для К580.

В микроЭВМ периферийные устройства часто работают от отдельного генератора тактовых сигналов, поэтому необходима син-

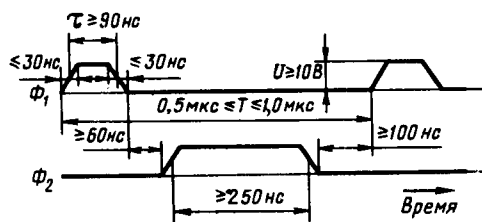


Рис. 1.55. Вариант двухфазной синхронизации К580.

хронизация сигналов, которыми обмениваются эти устройства. Синхронизация осуществляется синхронизаторами для импульсных и потенциальных входных сигналов. На рис. 1.56 приведены варианты принципиальных схем импульсного (d) и потенциальных (a , $в$) синхронизаторов и временные диаграммы их работы ($б$, $г$, $е$).

Методы организации асинхронного обмена между периферийными устройствами и микроЭВМ рассмотрены в § 1.3.

Для обеспечения правильной работы микроЭВМ необходимо после включения питания произвести ее начальную установку. Для этой цели в систему синхронизации встраивают соответствующие специальные схемы предварительной установки. Большинство микроЭВМ содержит также источник реального времени — часы и таймер. Последний предназначен для деления машинного времени на временные интервалы для более эффективного использования процессора при работе с периферийными устройствами. Таймер обеспечивает совместную работу МП и периферийных устройств в реальном масштабе времени, осуществляет разбиение машинного времени на равные интервалы времени, например длительностью 20 мс, соответствующей $\sim 50 \text{ Гц}$. В состав таймера входят схемы, необходимые для запроса и получения канала, формирования вектора прерывания, регистра состояния, благодаря которому анализируется режим работы таймера (с прерыванием или без прерывания,

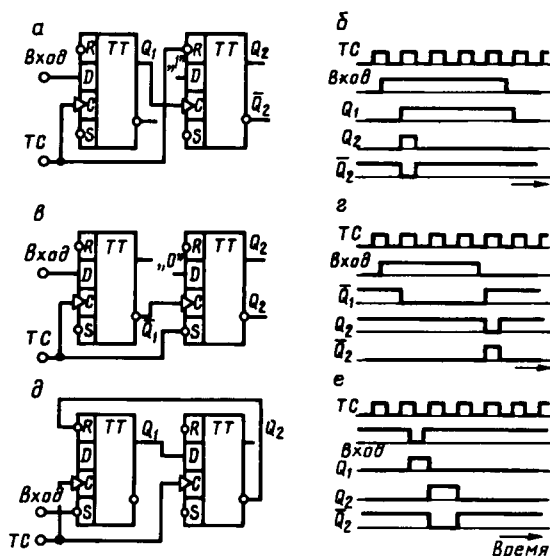


Рис. 1.56. Схемы синхронизаторов асинхронных сигналов (а, в, д) и временные диаграммы их работы (б, з, е).

причем в последнем случае таймер может выполнять функции программного переключателя), и т. п.

Узлы системы синхронизации выполняются либо в виде отдельных микросхем, либо в виде составных частей БИС управления или процессоров.

В серии К536 БИС К536ГГ1 формирует синхросигналы для микропроцессов, устройств ввода-вывода, запоминающих.

Устройство ввода-вывода (таймер) К536ИК5 предназначено для формирования различных временных сигналов и может быть использовано в функции делителя, распределителя, задержки, широтно-импульсного модулятора, интерполятора, преобразующего 8-разрядный параллельный двоичный код в последовательный число-импульсный код, в котором заданное число импульсов равномерно распределено в интервале, составляющем 256 периодов входной частоты.

В серии К1800 имеется схема синхронизации К1800ВБ2. Многофункциональное синхронизирующее устройство К589ХЛ4 предназначено для деления частоты, задержки сигналов, формирования пачки сигналов и сигналов заданной длительности.

Счетчики. Регистры. Сдвигатели. Счетчик — операционный узел ЭВМ, обеспечивающий хранение слова информации и выполнение над ним микрооперации счета. Микрооперация счета состоит в изменении значения слова или состояния счетчика на 1. Счетчик, на котором реализуется микрооперация счета вида $C := C + 1$, называется *суммирующим*. Если на счетчике реализуется микроопе-

рация $C := C - 1$, то счетчик называется *вычитающим*. Счетчик, на котором реализуются обе указанные микрооперации, называется *реверсивным*.

Счетчики применяются для подсчета шагов программы, циклов сложения и вычитания при выполнении операций сложения, вычитания, умножения, деления; преобразования кодов; построения распределителей импульсов, накапливающих сумматоров, пересчетных устройств и т. д. *Пересчетным устройством* называется операционный узел ЭВМ, обеспечивающий хранение слова информации и выполнение над ним микрооперации произвольной смены состояния, благодаря чему определения «суммирующий», «вычитающий», «реверсивный» неприменимы к пересчетным устройствам.

Простейшим одноразрядным счетчиком является счетный триггер, в качестве которого можно применить универсальные D- или JK-триггеры, выпускаемые в виде микросхем.

На рис. 1.57 приведены схемы D-триггера типа K155TM2 (а) и JK-триггера типа K155TB1 (г, д) и временные диаграммы их работы соответственно (б — для D-триггера, в — для JK-триггера).

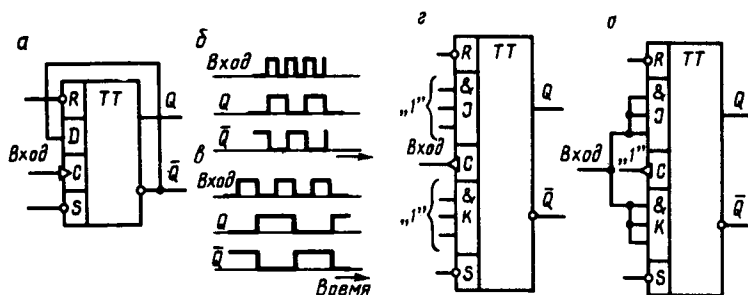


Рис. 1.57. Схемы использования в счетном режиме универсальных D- и JK-триггеров и временные диаграммы их работы.

На рис. 1.58, а показана схема пересчетного устройства, реализованного на микросхеме K155IE7 и принимающего в цикле состояния $0-1-2-3-1-2-3-0-1-\dots$.

На рис. 1.58, б даны временные диаграммы, поясняющие различные режимы работы 4-разрядного двоичного реверсивного счетчика K155IE7 (а).

В распоряжении разработчика имеется большая номенклатура ИС, позволяющая реализовать требуемую структуру счетчика.

Регистр — операционный узел ЭВМ, представляющий собой регулярную совокупность элементов памяти и комбинационных схем и предназначенный для ввода, хранения, преобразования и выдачи числа, а также для выполнения простейших поразрядных операций и выработки осведомительных сигналов о хранящемся в регистре числе.

В отличие от запоминающих устройств в регистрах осуществляется кратковременное хранение информации, например на период

выполнения одного или нескольких циклов работы всего устройства. Несколько регистров общего назначения (РОН) образуют сверх-оперативное запоминающее устройство (СОЗУ).

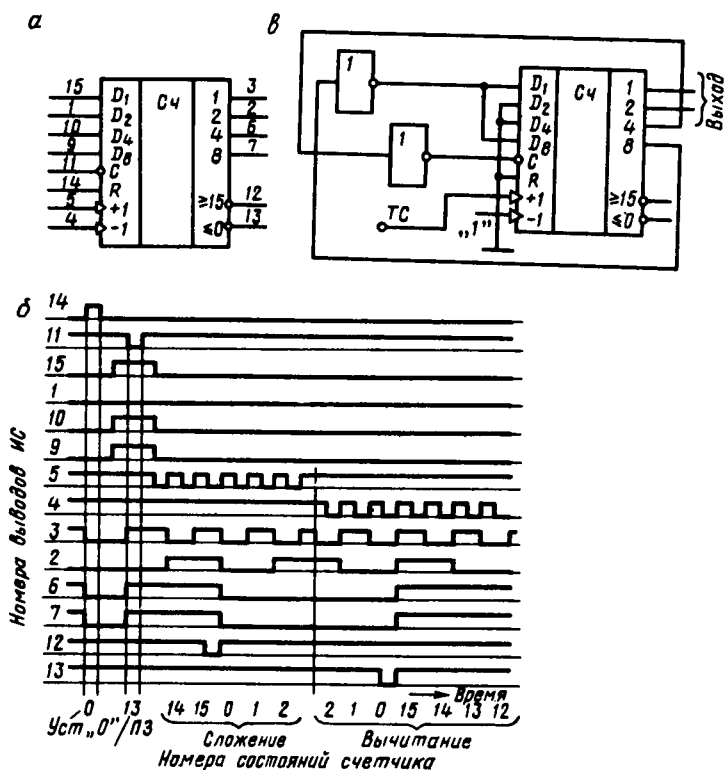


Рис. 1.58. Схема пересчетного устройства на микросхеме К155ИЕ7 и временные диаграммы работы этой микросхемы.

В качестве запоминающих элементов в регистрах используются триггеры, построенные на логических элементах. Число триггеров, предназначенных для хранения многоразрядного двоичного числа, равно количеству разрядов в этом числе. Вспомогательные элементы используются для операций ввода и вывода из регистра хранимого числа, преобразования кода числа, хранящегося в регистре, сдвига числа влево или вправо на определенное число разрядов и др. Вспомогательные элементы обычно строятся на основе комбинационных схем. Различные структуры регистров можно синтезировать на основе универсальных D- и JK-триггеров.

На рис. 1.59 приведены схема (б) и временные диаграммы работы (в) сдвигового регистра на D-триггерах, а на рис. 1.59, а — схема сдвигового регистра на основе JK-триггеров.

В виде схем средней степени интеграции выпускаются много-

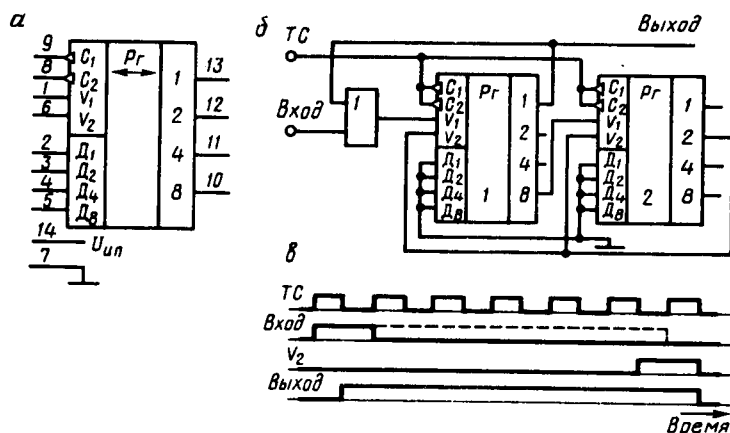


Рис. 1.60. Условное графическое обозначение универсального регистра К155ИР1 (а), схема цифрового одновибратора (б) и временные диаграммы его работы (в).

Т а б л и ц а 1.10

Режим работы		C_1	C_2	V_1	V_2	Примечание	
Запись параллельного кода		$\emptyset$		$\emptyset$	1	Записываемый код подается на D_i	
Сдвиг вправо			$\emptyset$	Вход	0	Допускаются указанные ниже соединения	
Сдвиг влево		$\emptyset$		$\emptyset$	1	Соединить: 2—12, 3—11, 4—10 Вход-вывод 5 (D_8)	
Делитель на	2	ТС	$\emptyset$	y_2		$y_2 = \overline{I3}$	y_i — логические функции; 10, 11, 12, 13 — логические переменные на соответствующих выводах микросхемы
	3			y_3		$y_3 = \overline{I2} \cdot \overline{I3}$	
	4			y_4		$y_4 = \overline{I2}$	
	5			y_5		$y_5 = \overline{II} \cdot \overline{I2}$	
	6			y_6		$y_6 = \overline{II}$	
	7			y_7		$y_7 = \overline{I0} \cdot \overline{II}$	
	8			y_8		$y_8 = \overline{I0}$	

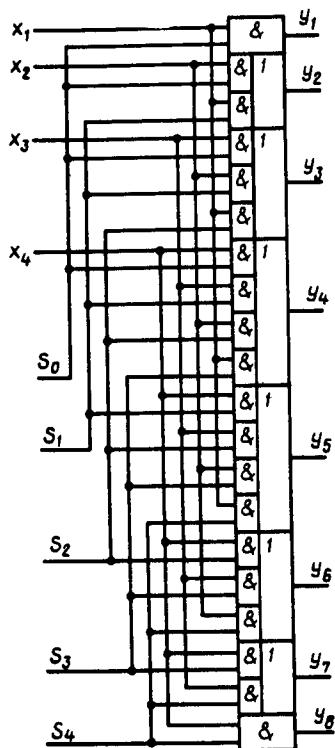


Рис. 1.61. Схема 4-разрядного сдвигателя.

один такт на 0—4 разряда. Уравнения, описывающие структуру и работу этого сдвигателя, имеют следующий вид:

$$\begin{aligned}
 y_1 &= S_0 x_1; \\
 y_2 &= S_0 x_2 + S_1 x_1; \\
 y_3 &= S_0 x_3 + S_1 x_2 + S_2 x_1; \\
 y_4 &= S_0 x_4 + S_1 x_3 + S_2 x_2 + S_3 x_1; \\
 y_5 &= \quad \quad \quad + S_1 x_4 + S_2 x_3 + S_3 x_2 + S_4 x_1; \\
 y_6 &= \quad \quad \quad \quad \quad + S_2 x_4 + S_3 x_3 + S_4 x_2; \\
 y_7 &= \quad \quad \quad \quad \quad \quad + S_3 x_4 + S_4 x_3; \\
 y_8 &= \quad \quad \quad \quad \quad \quad \quad S_4 x_4, \quad (1.53)
 \end{aligned}$$

где x_i — i -й разряд входного слова; y_j — j -й разряд выходного слова; S_k — управляющий сигнал, сдвигающий входное слово на k разрядов.

Наиболее просто сдвигатели реализуются на мультиплексорах. Перечисленные узлы ЭВМ широко используются для построения различных цифровых схем.

Дешифраторы, шифраторы. Дешифратор — операционный узел ЭВМ, выполняющий микрооперацию преобразования позиционного двоичного кода в унитарный цифровой код. На каждой выходной шине дешифратора вырабатывается информационный сигнал только при определенной комбинации входных сигналов. На рис. 1.62 показаны условное графическое

обозначение и схема дешифратора К155ИД3. На выходах данного дешифратора реализуются функции

$$\left. \begin{aligned}
 y_0 &= \overline{V_1} \overline{V_2} x_1 x_2 x_3 x_4; & y_2 &= \overline{V_1} \overline{V_2} x_1 x_2 x_3 x_4; & y_{14} &= \overline{V_1} \overline{V_2} x_1 x_2 x_3 x_4; \\
 y_1 &= \overline{V_1} \overline{V_2} x_1 x_2 x_3 x_4; & \dots & & y_{15} &= \overline{V_1} \overline{V_2} x_1 x_2 x_3 x_4.
 \end{aligned} \right\} \quad (1.54)$$

При $V_1 = V_2 = 0$ дешифратор реализует инверсии всех конъюнкций входных переменных, а при всех остальных наборах V_1 и V_2 на выходах дешифратора будут уровни «1».

Входы V_1 и V_2 служат для стробирования дешифратора и для расширения числа его информационных входов и выходов.

На рис. 1.63 приведена схема дешифратора на 32 выхода, построенная на ИС К155ИД3. Так как дешифратор реализует все конъюнкции (или их инверсии), он может быть использован для реализации одной или нескольких функций алгебры логики от переменных, подаваемых на информационные входы дешифратора.

На рис. 1.64 показан пример реализации четырехфазной систе-

мы синхронизации с неперекрывающимися импульсами. Входные переменные поступают с 4-разрядного двоичного счетчика.

На рис. 1.65 дана схема реализации с помощью дешифратора функции $y = x_2x_3 + x_1x_2x_4 + x_1x_2x_3x_4$.

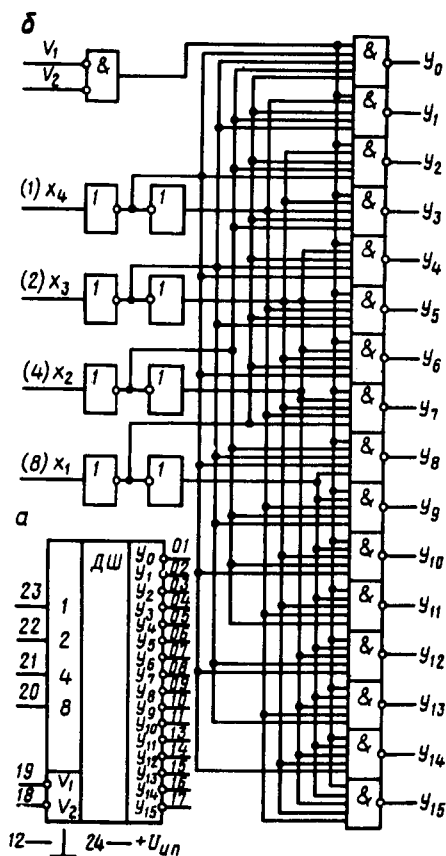


Рис. 1.62. Условное графическое обозначение (а) и схема (б) дешифратора К155ИД3.

Развернув данное выражение до суммы конstituент единицы и применив правило де-Моргана, получим

$$y = x_1x_2\bar{x}_3\bar{x}_4 + x_1x_2\bar{x}_3x_4 + \bar{x}_1x_2\bar{x}_3x_4 + \bar{x}_1x_2\bar{x}_3\bar{x}_4 + x_1x_2x_3x_4 + \bar{x}_1\bar{x}_2x_3x_4.$$

Использование дешифратора в режиме селектора рассмотрено ниже.

Шифратор — комбинационная схема, предназначенная для преобразования унитарного (однопозиционного) кода в двоичный позиционный. Типичным примером является шифратор клавиатуры для ввода в цифровое устройство шестнадцатиричного кода (рис. 1.66).

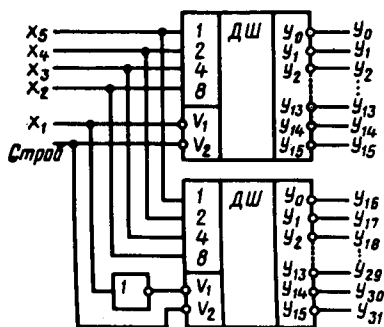
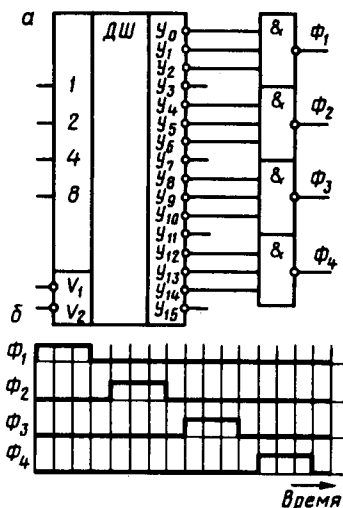


Рис. 1.63. Схема дешифратора 1 из 32.

Рис. 1.64. Схема четырехфазного генератора синхросигналов (а) и временная диаграмма его работы (б).



При ненажатых клавишах на всех входах шифратора обеспечивается «0», а нажатая клавиша подает на соответствующий вход шифратора «1». Шифратор вырабатывает также осведомительный сигнал Z при нажатии любой из клавиш. Уравнения, описывающие структуру шифратора, имеют следующий вид:

$$\left. \begin{aligned} y_1 &= 1 + 3 + 5 + 7 + 9 + B + D + F; \\ y_2 &= 2 + 3 + 6 + 7 + A + B + E + F; \\ y_4 &= 4 + 5 + 6 + 7 + C + D + E + F; \\ y_8 &= 8 + 9 + A + B + C + D + E + F; \\ z &= 0 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + A + B + C + D + E + F. \end{aligned} \right\} (1.55)$$

Мультиплексоры. Селекторы. Мультиплексор — операционный узел ЭВМ, осуществляющий микрооперацию передачи сигнала с одного из своих входов на один выход. На рис. 1.67 показаны схема и функциональное обозначение стробируемого мультиплексора К155КП7. Структура этого прибора описывается следующими уравнениями:

$$\begin{aligned} y_1 &= \bar{V}\bar{a}_4\bar{a}_2\bar{a}_1D_0 + \bar{V}\bar{a}_4\bar{a}_2a_1D_1 + \bar{V}\bar{a}_4a_2\bar{a}_1D_2 + \bar{V}\bar{a}_4a_2a_1D_3 + \\ &+ \bar{V}a_4\bar{a}_2\bar{a}_1D_4 + \bar{V}a_4a_2\bar{a}_1D_5 + \bar{V}a_4a_2a_1D_6 + \bar{V}a_4a_2a_1D_7; \\ y_2 &= \hat{y}_1. \end{aligned} \quad (1.56)$$

Вход V мультиплексора используется для стробирования и наращивания числа входов мультиплексора. Входы a_4 , a_2 и a_1 — адресные. Двоичный код на этих входах определяет, какой сигнал D_i будет передан на выход. Например, при $a_4=1$, $a_2=0$, $a_1=1$, $V=0$ сигнал D_5 передается на выход, так как все остальные члены в (1.56) при этом наборе будут равны 0, т. е. $y_1=D_5$, а $y_2=\bar{D}_5$.

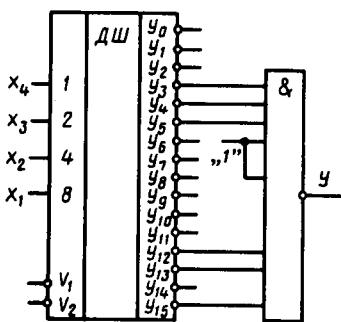


Рис. 1.65. Схема реализации произвольной функции алгебры логики.

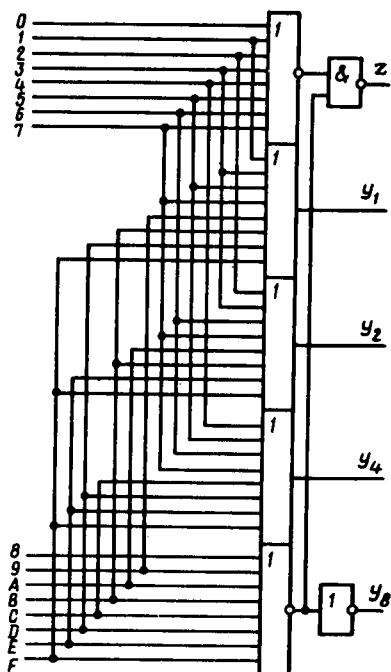


Рис. 1.66. Схема шифратора клавиатуры.

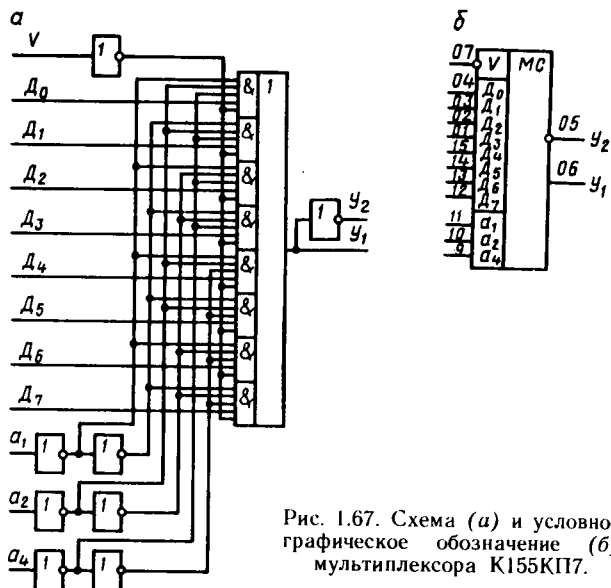


Рис. 1.67. Схема (а) и условное графическое обозначение (б) мультимплексора К155КП7.

На рис. 1.68 показана схема мультимплексора $16 \rightarrow 1$, построенная на двух мультимплексорах К155КП7.

Из (1.56) при $V=0$ следует, что это совершенная дизъюнктивная

нормальная форма функции y_1 переменных a_4, a_2 и a_1 , если D_i рассматривать как конкретное значение y_1 на соответствующем наборе переменных a_4, a_2, a_1 . В табл. 1.11 приведены функции: y_1 — функция трех переменных (x_1, x_2, x_3) и y_2 — функция четырех переменных (x_1, x_2, x_3, x_4).

На рис. 1.69, а показана реализация на мультиплексоре МС функции y_1 . В этом случае D_i равно «0» или «1» для соответствующего набора в табл. 1.11. На рис. 1.69, б показана реализация функции y_2 . При этом D_i равно 0, 1, $x_4, \bar{x}_4$. Если реализовать D_i как функцию

Таблица 1.11

	x_1	x_2	x_3		y_1	y_2
	x_1	x_2	x_3	x_4		
0	0	0	0	0	0	0
0	0	0	0	1	0	0
0	0	0	1	0	0	0
0	0	0	1	1	1	1
0	0	1	0	0	0	1
0	0	1	0	1	1	1
0	0	1	1	0	1	1
0	0	1	1	1	1	0
1	1	0	0	0	0	0
1	1	0	0	1	1	1
1	1	0	1	0	0	1
1	1	0	1	1	1	0
1	1	1	0	0	0	0
1	1	1	0	1	1	0
1	1	1	1	0	0	1
1	1	1	1	1	1	0

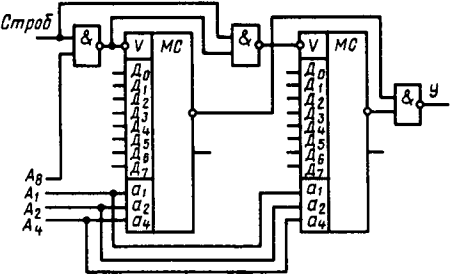


Рис. 1.68. Схема мультиплексора 16→1.

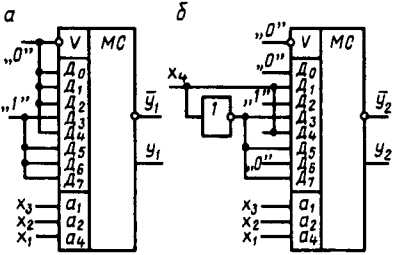


Рис. 1.69. Схемы реализации произвольных функций алгебры логики на мультиплексоре.

двух переменных x_4, x_5 , то мультиплексор 8→1 можно использовать для реализации функции пяти переменных и т. д.

На рис. 1.70 показан 4-разрядный фрагмент сдвигателя со сдвигом от 0 до 3 разрядов. При a_2a_1 , равном 00, сдвига нет, а при a_2a_1 , равном 11, осуществляется сдвиг на три разряда. Таким образом, количество разрядов, на которое надо сдвинуть число, определяется дво-

ичным кодом, подаваемым на адресные входы мультиплексора.

Селектор (демультиплексор) — операционный узел ЭВМ, осуществляющий микрооперацию передачи сигнала с одного входа на один из нескольких выходов. Селекторы в виде самостоятельных приборов не изготавливаются, так как их функции может выполнять дешифратор. Обратимся к рис. 1.62 и уравнениям (1.54). Если $V_1 = x_{\text{вх}}$, $V_2 = 0$, то очевидно, что $x_{\text{вх}}$ появится на том выходе дешифратора,

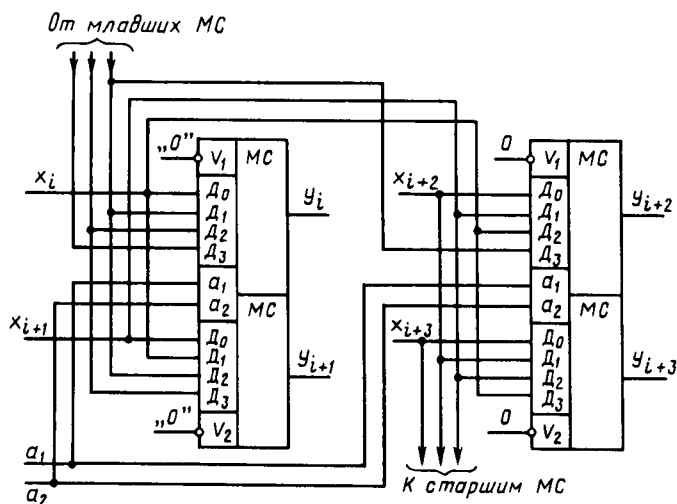


Рис. 1.70. Схема реализации сдвигателя на мультиплексорах.

которому соответствует конкретный набор переменных x_1, x_2, x_3, x_4 , являющийся в данном случае адресом селектора. Если дешифратор нестробируемый, то входом селектора будет один из входов дешифратора, а остальные входы можно использовать как адресные. В этом случае на одной половине выходов дешифратора получается прямой выходной сигнал, а на другой — обратный.

Широкие возможности предоставляет разработчику *мультиплексор-селектор*, выполненный на КМДП-структурах. На рис. 1.71 показаны схема и функциональное обозначение микросхемы 564КП2. Один из восьми выходов дешифратора, задаваемый адресом $a_4a_2a_1$ при $V=0$, открывает соответствующий двунаправленный ключ Кл, обеспечивая связь цепей y и D_i . При $V=1$ все ключи закрыты. Данная схема может применяться в качестве дешифратора с активным уровнем выходов «1» или «0», мультиплексора и селектора.

При использовании схемы в качестве:

дешифратора с активным уровнем выходов «1» необходимо присоединить цепь y к источнику питания, а цепи D_i через резисторы не менее 20 кОм — к земле;

дешифратора с активным уровнем выходов «0» необходимо присоединить цепь y к земле, а цепи D_i через резисторы не менее 20 кОм —

к источнику питания (входами дешифратора будут входы a_4, a_2, a_1 , а выходами — D_i);

мультиплексора адресными входами являются входы a_4, a_2, a_1 ; информационными входами — D_i , а выходом — y ;

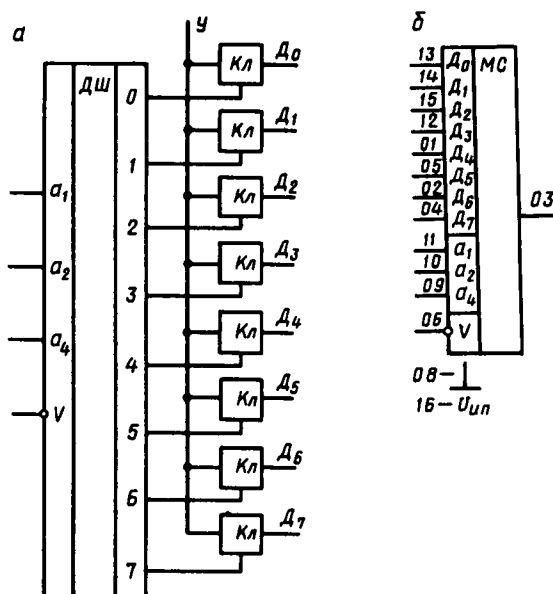


Рис. 1.71. Схема (а) и условное графическое обозначение (б) селектора-мультиплексора 564КП2.

селектора адресными входами являются входы a_4, a_2, a_1 , информационным входом — y , а выходами — D_i .

Из приведенных примеров следует, что мультиплексоры и селекторы являются многофункциональными узлами ЭВМ.

Сумматоры. Арифметическо-логические устройства. Сумматор — операционный узел ЭВМ, выполняющий арифметическое суммирование кодов чисел. Сумматор является одним из основных узлов арифметического устройства. Правила для поразрядного сложения двух чисел представлены в табл. 1.12, где a_i, b_i — одноименные разряды слагаемых; S_i — сумма; C_{i+1} — перенос в старший разряд.

Т а б л и ц а 1.12

a_i	b_i	S_i	C_{i+1}
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Уравнения для суммы и переноса имеют вид

$$S_i = \bar{a}_i b_i + a_i \bar{b}_i = a_i \oplus b_i; C_{i+1} = a_i b_i. \quad (1.57)$$

Схема, реализующая эти уравнения, называется *полусумматором*. Условное графическое обозначение (а) полусумматора и его схема (б) приведены соответственно на рис. 1.72.

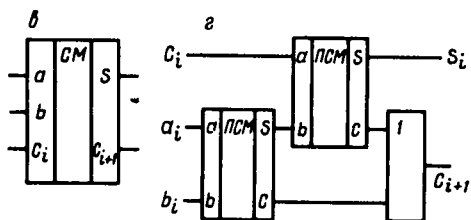


Рис. 1.72. Условное графическое обозначение и схема полусумматора (а, б) и сумматора (в, г).

Таблица 1.13

a_i	b_i	C_i	S_i	C_{i+1}	a_i	b_i	C_i	S_i	C_{i+1}
0	0	0	0	0	1	0	0	1	0
0	0	1	1	0	1	0	1	0	1
0	1	0	1	0	1	1	0	0	1
0	1	1	0	1	1	1	1	1	1

Полный двоичный одноразрядный сумматор можно рассматривать как логическое устройство, имеющее три входа (две цифры слагаемых и цифра переноса из соседнего младшего разряда) и два выхода (сумма и перенос в старший разряд). Работа сумматора отражена в табл. 1.13.

Уравнения для суммы и переноса имеют вид

$$\left. \begin{aligned} S_i &= \bar{a}_i \bar{b}_i C_i + \bar{a}_i b_i \bar{C}_i + a_i \bar{b}_i \bar{C}_i + a_i b_i C_i; \\ C_{i+1} &= \bar{a}_i b_i C_i + a_i \bar{b}_i C_i + a_i b_i \bar{C}_i + a_i b_i C_i. \end{aligned} \right\} \quad (1.58)$$

На рис. 1.72 приведены схема (в) реализации сумматора на двух полусумматорах и элементе ИЛИ и условное графическое обозначение сумматора (г).

Для практической реализации сумматора уравнения (1.58) могут быть преобразованы к виду, удобному для реализации на заданных логических элементах и удовлетворяющему предъявляемым к сумматору требованиям по быстродействию.

Один из возможных вариантов быстродействующего сумматора в виде интегральной схемы К155ИМ1 (рис. 1.73) реализует уравнения

$$\left. \begin{aligned} S_i &= \overline{a_i b_i C_i + \bar{C}_{i+1} (a_i + b_i + C_i)}; \\ S_i &= \overline{a_i b_i C_i + \bar{C}_{i+1} (a_i + b_i + C_i)}; \\ \bar{C}_{i+1} &= \overline{a_i b_i + a_i C_i + b_i C_i}, \end{aligned} \right\} \quad (1.59)$$

где $a_i = a_1 a_2 + \bar{a}_3 + \bar{a}_4$; $b_i = b_1 b_2 + \bar{b}_3 + \bar{b}_4$.

Схема сумматора включает в себя входную логику, схему формирования суммы и ее инверсии и инверсии переноса. С учетом того, что S_i является самодвойственной функцией (инверсия функции

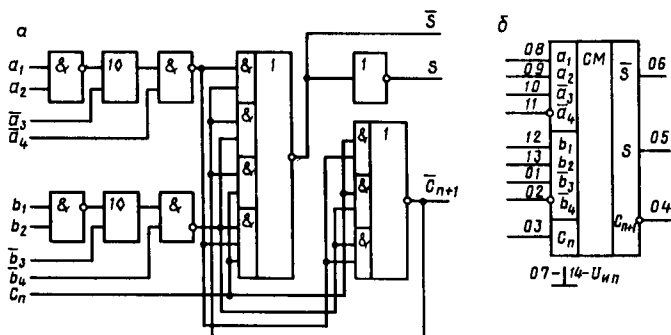


Рис. 1.73. Схема (а) и условное графическое обозначение (б) сумматора К155ИМ1.

получается при инвертировании значений всех переменных), сумматор К155ИМ1 обладает широкими логическими возможностями. Входные цепи сумматора осуществляют операции суммирования с прямыми и инверсными значениями переменных, а с использованием входов $\bar{a}_3, \bar{b}_3$ реализуют схему «монтажное ИЛИ», при этом управление по входам 01 и 10 микросхемы осуществляется от схем с открытым коллектором. Указанные особенности сумматора позволяют на его основе строить многоразрядные сумматоры, работающие в режиме сложения или вычитания. При построении многоразрядных сумматоров вход C_i первого разряда подключается к уровню «0», перенос из первого разряда $\bar{C}_{i+1}$ подключается на вход C_i второго разряда, цифры a_i, b_i второго разряда подаются на входы a_4, b_4 , при этом обеспечивается $a_1 a_2 = 0, b_1 b_2 = 0$, сумма слагаемых с нечетных разрядов снимается с выхода S , а с четных — с выхода $\bar{S}$.

Для реализации *многоразрядных вычитателей* вход C_i первого разряда подключается к источнику «1», а слагаемое B подается на инвертированные входы a_4 , благодаря чему операция вычитания заменяется операцией сложения с отрицательным числом $A - B = A + (-B)$.

По аналогичной схеме выполнены 2-разрядный сумматор К155ИМ2 и 4-разрядный сумматор К155ИМ3 с организацией последовательного переноса между разрядами. Входная логика, благодаря которой из микросхем К155ИМ1 можно осуществлять построение различных арифметических устройств без применения других схем,

в микросхемах К155ИМ2, К155ИМ3 не нужна, так как соединение между разрядами постоянно.

Арифметическо-логическое устройство (АЛУ) — операционный узел ЭВМ, выполняющий арифметические и логические операции над двумя многоразрядными словами в зависимости от управляющего слова. АЛУ должно удовлетворять следующим требованиям: 1) выполнению заданного набора операций, 2) обеспечению поразрядного переноса и возможности его блокирования, 3) обеспечению наращиваемости разрядности обрабатываемых слов.

На рис. 1.74 приведено условное графическое обозначение АЛУ типа К155ИП3. Схема имеет восемь информационных входов: $A_0, B_0, \dots, A_3, B_3$. На эти входы подаются четыре разряда чисел A и B , над которыми производятся арифметические или логарифмические

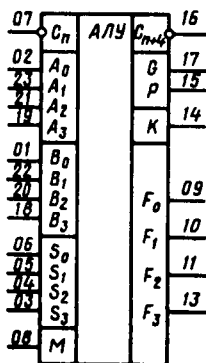


Рис. 1.74. Условное графическое обозначение арифметическо-логического устройства К155ИП3.

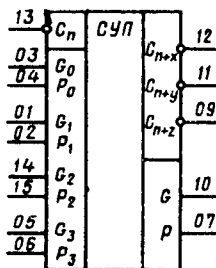


Рис. 1.75. Условное графическое обозначение схемы ускоренного переноса К155ИП4.

операции, определяемые управляющими входами $S_0, \dots, S_3$. Вход модификатора операций M обеспечивает выбор между арифметическими и логическими операциями. При $M=1$ блок ускоренного переноса внутри АЛУ, осуществляющий перенос из разряда в разряд, блокируется и схема выполняет логические операции. При $M=0$ блок ускоренного переноса оказывается включенным и схема выполняет арифметические операции.

Вход C_n является входом переноса из предыдущего разряда. Этот вход необходим при разрядности обрабатываемых чисел, превышающей 4. На каждом из четырех информационных выходов $F_0, \dots, F_3$ получается результат арифметической или логической операции над соответствующими разрядами чисел A и B . В схеме АЛУ имеется специальный выход K , называемый выходом сравнения ($A=B$). Для организации ускоренного переноса в АЛУ предусмотрены три выхода: G — выход образования переноса 4-разрядного каскада, P — выход распространения переноса 4-разрядного каскада, C_{n+1} — выход переноса 4-разрядного каскада.

Таблица 1.14

Выбор функций $S_3 S_2 S_1 S_0$	Положительная логика				Отрицательная логика			
	$M=1$ Логические функции	$M=0$ Арифметические операции		$M=1$ Логические функции	$M=0$ Арифметические операции		$M=0$ Арифметические операции	
		$C_n=1$ (нет входного переноса)	$C_n=0$ (есть входной перенос)		$C_n=1$ (нет входного переноса)	$C_n=0$ (есть входной перенос)		
0000	$F=A$	$F=A$	$F=A$ плюс 1	$F=\bar{A}$	$F=A$	$F=A$ минус 1		
0001	$F=\bar{A} + \bar{B}$	$F=A+B$	$F=(A+B)$ плюс 1	$F=\bar{A} \cdot \bar{B}$	$F=A \cdot B$	$F=A \cdot B$ минус 1		
0010	$F=A \cdot B$	$F=A + \bar{B}$	$F=(A+B)$ плюс 1	$F=\bar{A} + B$	$F=A \cdot B$	$F=A \cdot B$ минус 1		
0011	$F=0$	$F=\text{минус } 1$ (дополнение до 2)	$F=0$	$F=1$	$F=0$	$F=\text{минус } 1$ (дополнение до 2)		
0100	$F=\bar{A} \cdot \bar{B}$	$F=A$ плюс $A \cdot B$	$F=A$ плюс 1	$F=\bar{A} + B$	$F=A$ плюс $(A+B)$ плюс 1	$F=A$ плюс $(A+B)$		
0101	$F=\bar{B}$	$F=(A+B)$ плюс $A \cdot \bar{B}$	$F=(A+B)$ плюс 1	$F=\bar{B}$	$F=A \cdot B$ плюс $(A+B)$ плюс 1	$F=A \cdot B$ плюс $(A+B)$		
0110	$F=A \oplus B$	$F=A$ минус B минус 1	$F=A$ минус B	$F=\bar{A} \oplus \bar{B}$	$F=A$ минус B	$F=A$ минус B минус 1		
0111	$F=A \cdot \bar{B}$	$F=A \cdot \bar{B}$ минус 1	$F=A \cdot \bar{B}$	$F=A + \bar{B}$	$F=(A + \bar{B})$ плюс 1	$F=A + \bar{B}$		
1000	$F=\bar{A} + B$	$F=A$ плюс $A \cdot B$	$F=A$ плюс 1	$F=\bar{A} \cdot B$	$F=A$ плюс $(A+B)$ плюс 1	$F=A$ плюс $(A+B)$		
1001	$F=\bar{A} \oplus \bar{B}$	$F=A$ плюс B	$F=A$ плюс B плюс 1	$F=A \oplus B$	$F=A$ плюс B	$F=A$ плюс B		
1010	$F=B$	$F=(A+B)$ плюс $A \cdot B$	$F=(A+B)$ плюс 1	$F=B$	$F=A \cdot B$ плюс $(A+B)$ плюс 1	$F=A \cdot B$ плюс $(A+B)$		
1011	$F=A \cdot B$	$F=A \cdot B$ минус 1	$F=A \cdot B$	$F=A + B$	$F=(A+B)$ плюс 1	$F=A + B$		
1100	$F=1$	$F=A$ плюс A	$F=A$ плюс 1	$F=0$	$F=A$ плюс A	$F=A$ (сдвиг)		
1101	$F=A + \bar{B}$	$F=(A+B)$ плюс A	$F=(A+B)$ плюс 1	$F=A \cdot B$	$F=A \cdot B$ плюс 1	$F=A \cdot B$ плюс A		
1110	$F=A + B$	$F=(A+B)$ плюс A	$F=(A+B)$ плюс 1	$F=A \cdot B$	$F=A \cdot B$ плюс 1	$F=A \cdot B$ плюс A		
1111	$F=A$	$F=A$ минус 1	$F=A$	$F=A$	$F=A$ плюс 1	$F=A$		

Аналитические выражения для выходных сигналов АЛУ имеют следующий вид:

$$F_i = (a_i b_i S_3 + a_i \bar{b}_i S_2) \oplus (a_i + \bar{b}_i S_1 + b_i S_0) \oplus \bar{M} C_n; \quad (1.60)$$

$$K = F_0 F_1 F_2 F_3; \quad (1.61)$$

$$P = \overline{P_3 P_2 P_1 P_0}; \quad (1.62)$$

$$G = \overline{G_3 + G_2 P_3 + G_1 P_3 P_2 + G_0 P_3 P_2 P_1}; \quad (1.63)$$

$$C_{n+4} = \bar{G} + \bar{P} C_n, \quad (1.64)$$

где

$$P_i = \overline{a_i b_i S_3 + a_i \bar{b}_i S_2}; \quad (1.65)$$

$$G_i = \overline{a_i + \bar{b}_i S_i + b_i S_0}. \quad (1.66)$$

Полный набор операций, выполняемых АЛУ, приведен в табл. 1.14. Кроме 32 арифметических (16 при наличии переноса и 16 при отсутствии переноса) и 16 логических операций, АЛУ может осуществлять операцию сравнения двух чисел. Выход K используется для указания $A=B$, когда АЛУ находится в режиме, позволяющем осуществлять операцию вычитания. Для этого необходимо подать комбинацию 0110 на входы S_0-S_3 , $M=0$, $C_n=1$, тогда АЛУ будет выполнять операцию вычитания и на каждом выходе F_i появится результат произведения операции $F_i=A_i$ минус B_i минус единица.

При $A=B$ на всех выходах f_i будет единица, т. е. на выходе K будет также единица. Выход сравнения K имеет открытый коллектор и может быть соединен («монтажное И») с другими выходами K аналогичных схем для сравнения более чем 4-разрядных чисел.

Выходы образования и распространения переноса G и P используются при объединении нескольких схем АЛУ с помощью блока ускоренного переноса К155ИП4 (рис. 1.75). С помощью этого блока можно сформировать ускоренный перенос между группами АЛУ в пределах 16, 32, 64 разрядов (рис. 1.76, $a-z$).

По аналогичной структуре выполнены АЛУ: 134ИПЗ, 530ИПЗ, 531ИПЗ, 564ИПЗ.

АЛУ явились основой при разработке центральных процессорных элементов микропроцессорных комплектов БИС.

Элементы памяти. Под *элементами памяти* подразумеваются простейшие схемы или физические среды, предназначенные для приема, хранения и выдачи одного бита информации.

Память ЭВМ — функциональная часть ЭВМ, используемая для хранения обрабатываемых данных, запоминания результатов выполнения операций или отдельных команд, сохранения информации о состоянии процессора и т. п.

Запоминающие устройства представляют собой комплекс технических средств, предназначенный для приема, хранения и выдачи

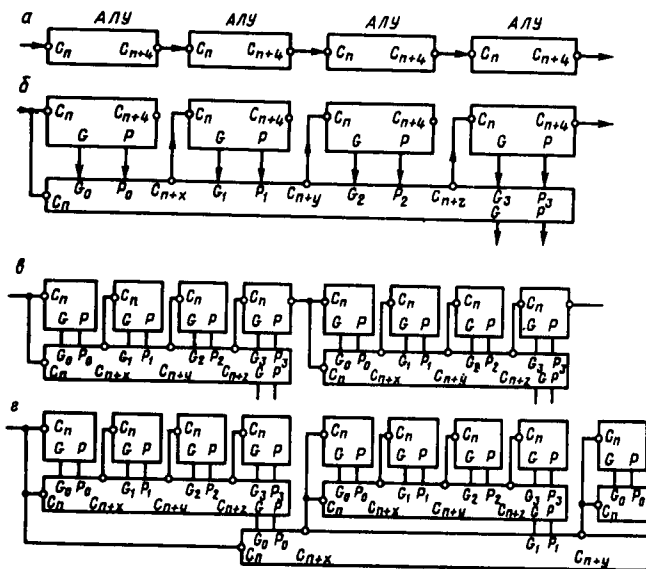


Рис. 1.76. Схемы реализации многоразрядного АЛУ.

информации, представленной двоичным словом. Этот комплекс включает в себя средства адресации информации, накопитель информации и устройство управления, осуществляющее запись, считывание, синхронизацию, регенерацию информации (если это необходимо) и блокирование всего запоминающего устройства. В микроЭВМ используется исключительно полупроводниковая память, стоимость которой составляет более 70 % всей стоимости микроЭВМ.

По принципу построения накопителя ОЗУ могут строиться со *словарной (линейной) организацией (однокоординатной выборкой)* или *матричной организацией (двухкоординатной выборкой)*, показанных на рис. 1.77.

На рис. 1.77, в приведен пример статического шеститранзисторного запоминающего элемента, являющегося RS-триггером с принудительной установкой его состояния парафазными разрядными линиями. Этим запоминающий элемент отличается от RS-триггера, выполненного на логических элементах.

В режиме записи на выходах усилителей записи, подключенных к разрядным линиям РЛ₀ и РЛ₁, появляются парафазные сигналы. При появлении сигнала на выбранной линии адреса ЛА открываются транзисторы Т₁ и Т₆ и запоминающий элемент, выполненный на транзисторах Т₂—Т₅, устанавливается в требуемое состояние. После снятия сигнала с ЛА запоминающий элемент хранит информацию сколь угодно долго при наличии напряжения питания.

В режиме чтения на выходах усилителей записи устанавливается третье состояние и при подаче сигнала на ЛА разрядная линия РЛ₁ передает информацию на выход через выходной усилитель. Управление чтением/записью осуществляется сигналом Чт/Зп.

Сигнал выбора кристалла ВК устанавливает в третье состояние и входные и выходные усилители. Это позволяет непосредственно соединять выходы микросхем при наращивании объема памяти.

Метод словарной организации выборки применим только при небольшом количестве запоминающих ячеек. При использовании этого метода для памяти объемом 1 К бит необходимо иметь дешифратор с 1024 выходными линиями, что просто нерационально. В ОЗУ с матричной организацией запоминающие элементы располагают в виде матрицы и используют два дешифратора: один — для выбора столбца (Y-линии), а другой — для выбора строки (X-линии).

На рис. 1.77, а показана схема ОЗУ с организацией $2^x \times 2^y$ накопителя, а на рис. 1.77, б — пример статического восьмитранзисторного запоминающего элемента. Принцип работы этой схемы аналогичен рассмотренной выше. Последовательность подачи сигналов адреса, данных, Чт/Зп и ВК, а также их временные параметры указываются в технических характеристиках микросхем.

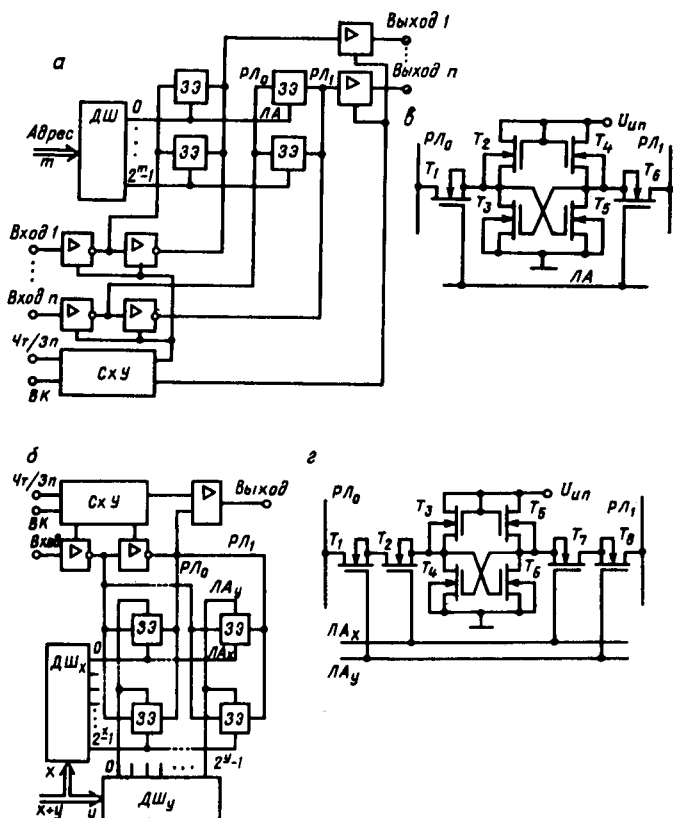


Рис. 1.77. Схемы ОЗУ с линейной (а) и матричной (б) организацией и 6-транзисторного (в) и 8-транзисторного (г) запоминающих элементов.

Кроме статических ЗУ, широко распространены динамические, основным запоминающим элементом которых является емкость затвор — исток МДП-транзистора. Каждый запоминающий элемент ЗЭ содержит как минимум один транзистор, и поэтому плотность элементов на кристалле больше, чем в статических ОЗУ. Наличие заряда на емкости определяет одно логическое состояние, а его отсутствие — другое. Благодаря утечке заряда хранимая информация разрушается, поэтому каждые несколько миллисекунд должна выполняться регенерация хранимого на емкости заряда путем выполнения операций чтения и записи на кристалле памяти. Схемы регенерации, являющиеся внешними по отношению к кристаллам памяти, повышают стоимость динамической памяти по сравнению со статической, из-за чего применение динамической памяти не оправдывает себя в системах с небольшой памятью.

При объемах 4—8 К байт и более затраты на схемы регенерации для динамической памяти компенсируются более низкой стоимостью, меньшей энергоемкостью и размерами этой памяти. Разработаны псевдостатические ЗУ, у которых схемы регенерации реализованы на кристалле, благодаря чему пользователь может рассматривать их как статические.

По методу доступа к информации различают ЗУ с произвольной и последовательной выборкой. Время доступа к произвольной ячейке в ЗУ первого типа не зависит от ее адреса, а в ЗУ второго типа (рис. 1.78) сдвиговый регистр РгСд может

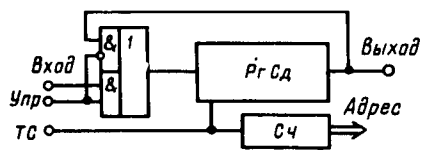


Рис. 1.78. Схема ОЗУ с последовательной выборкой.

накапливать и выдавать биты данных только последовательно по адресу, задаваемому счетчиком Сч.

Запоминающие устройства, для которых возможны операции как чтения, так и записи, называются *оперативными запоминающими устройствами* (ОЗУ). Для некоторых применений необходимо, чтобы записанная информация сохранялась постоянно, например программы инициализации операционной системы, программы-мониторы или таблицы часто используемых данных. В таких случаях желательно, чтобы МП, имея доступ к хранимой информации, не мог изменить ее. Такое ЗУ называется памятью только для чтения или *постоянным запоминающим устройством* (ПЗУ). Важнейшим достоинством ПЗУ перед ОЗУ является то, что оно энергонезависимо, так как отключение питания не может изменить содержимого памяти. Существуют большое количество различных типов ПЗУ [9]. На рис. 1.79 показаны условное графическое обозначение (а) электрически программируемого ПЗУ емкостью 256 бит с организацией 32×8 (K155PE3), фрагмент схемы (б) этого ПЗУ и временные диаграммы (в) режима программирования. Рассмотрим этот режим. Перво-

начально во всех ячейках ППЗУ записан «0». Для записи «1» необходимо: подать на адресные входы требуемый адрес, установить $U_{ип1}=12,5$ В, на вход выбора кристалла подать «0», а на резистор $R=390$ Ом — напряжение $U_{ип2}=12,5$ В. В этом случае пробивается стабилитрон Д, открывается транзистор T_1 и пережигается нихромовая перемычка у соответствующего эмиттера много-

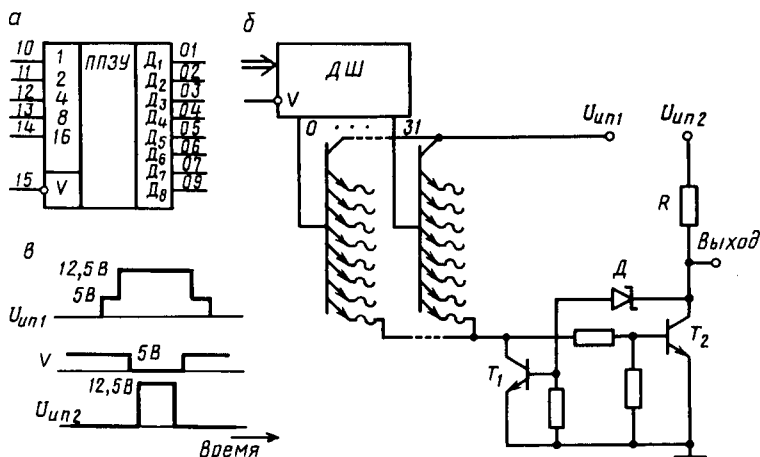


Рис. 1.79. Условное графическое обозначение (а), фрагмент схемы (б) и временные диаграммы режима программирования (в) микросхемы К155РЕ3.

эмиттерного транзистора. Выход из режима программирования осуществляется в обратной последовательности. В одном цикле допускается программирование только одного разряда. Выпускаются ППЗУ с организацией 256×4 ; 512×8 ; 2048×8 , для программирования которых разработаны специальные программаторы.

В ПЗУ адресные дешифраторы используются для обращения к одному из n слов памяти. Если ПЗУ имеет 11-битовый адресный вход, оно может содержать 2048 слов. В ряде случаев нет необходимости осуществлять дешифрацию всех возможных комбинаций, поступающих по адресным входам. Типичным примером сказанного является реализация управляющей памяти, для которой необходимо несколько десятков или сотен слов. В таких случаях широко применяют *программируемые логические матрицы* (ПЛМ), которые делятся на две части (рис. 1.80). В первой части входные переменные и их инверсии, получаемые в самой матрице, селективно подсоединяются ко входам логических элементов И, образующих матрицу произведений. Сигналы с выходов элементов вводятся во вторую часть матрицы, где с помощью других селективных соединений могут быть поданы на входы элементов ИЛИ, образующих матрицу сумм. В ПЛМ в отличие от ПЗУ программируются не только данные, но и адреса, благодаря чему она может справляться с определенными специальными случаями адресации легче, чем ПЗУ.

Эти специальные случаи объединяют незапрограммированные адреса, а также наличие одного адреса для многих слов и многих адресов для одного слова.

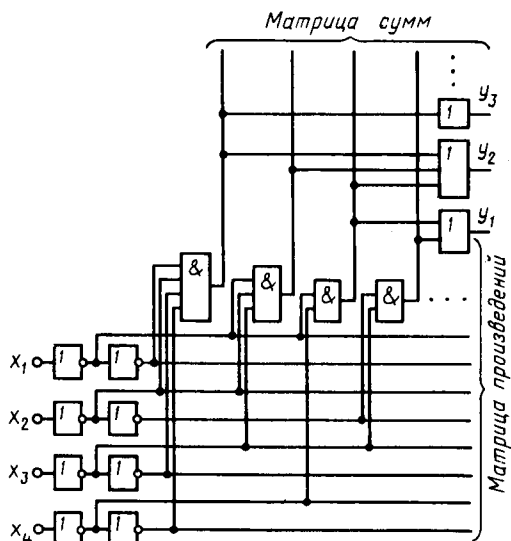


Рис. 1.80. Схема программируемой логической матрицы.

Схемой ПЛМ, представленной на рис. 1.80, реализуются следующие функции:

$$\left. \begin{aligned} y_1 &= x_2 \bar{x}_3 + \bar{x}_1 x_4; \\ y_2 &= \bar{x}_1 x_4 + x_1 \bar{x}_2 x_3 + x_1 x_2 x_3 x_4; \\ y_3 &= x_1 \bar{x}_2 x_3 x_4. \end{aligned} \right\} \quad (1.67)$$

Подробнее со структурой и элементами ЗУ можно познакомиться в [9].

Заказные БИС на основе некоммутированных логических матриц. Большие интегральные схемы в составе микропроцессорных наборов обеспечивают реализацию основных системных блоков на малом числе микросхем. Однако пока еще невозможно получить функционально законченную конструкцию цифрового устройства без некоторых дискретных логических приборов, с помощью которых эти блоки объединяются в общей структуре. Сложность и нестандартность системных архитектур или требования по производительности приводят к тому, что совокупность дискретных логических приборов представляет собой нетривиальные блоки произвольной логики, выполняемые с использованием большого количества малых и средних ИС, функции и параметры которых не всегда полностью соответствуют поставленным задачам. Один из путей решения указанной проблемы заключается в применении *некоммутированных*

логических матриц (НЛМ), достоинствами которых являются: универсальность, возможность реализации как цифровых, так и линейных функций; повышение надежности реализуемых схем; возможность реализации на их основе большого количества различных относительно дешевых интегральных схем; возможность реализации полностью законченной цифровой системы обработки информации.

Между дешевыми стандартными интегральными схемами и дорогими заказными приборами в настоящее время находится множество некоммутированных логических матриц.

Можно выделить три основных варианта применения некоммутированных логических матриц: «сборные ИС», «системные ИС» и «ИС-полуфабрикаты». «Сборные ИС», как показывает само их название, дают возможность избавиться от многочисленных самостоятельных логических схем путем замены различных ИС малой степени интеграции одним прибором. В свою очередь «системная ИС» выполняет одну законченную функцию. «ИС-полуфабрикаты» обеспечивают конкретного пользователя «стандартной» деталью, которая служит в качестве неперменного компонента, типичного для ряда разработок. В любом случае выбор определенного матричного изделия производится по четырем ключевым параметрам: числу интерфейсных сигналов и логических элементов, нагрузочной способности по выходу и быстродействию логических схем.

Некоммутированные логические матрицы представляют собой кристалл кремния со сторонами 4—7 мм, на котором располагаются контактные площадки и схемы стыковки, обеспечивающие стандартные логические уровни (например, ТТЛ-схем), возможность организации выходов с тремя состояниями, активных уровней «1» или «0» и т. п., а также места для трассировки соответствующих связей; большая часть кристалла отводится под ячейки библиотечных элементов, в зависимости от функциональной сложности содержащие инверторы, логические элементы И, ИЛИ, И-НЕ, ИЛИ-НЕ с различным числом входов, двунаправленные ключи, RS-триггеры, схемы «исключающее ИЛИ», отдельные транзисторы, управляемые буферы прямого или обратного кода, триггерные элементы ЗУ и пр.

По специальной инструкции разработчик кодирует требуемую схему на основе библиотечных элементов. Информация вводится в систему автоматического проектирования, которая проверяет правильность исходных данных пользователя, преобразует физические базы данных в физические координаты кристалла, а затем разрабатывает рисунок соединительных связей. Последнее в цикле проектирования — проверка работоспособности спроектированной схемы с помощью тестов, разработанных пользователем.

В настоящее время ведутся работы по созданию автоматических генераторов тест-векторов и языков высокого уровня для описания схем и тестов. Последнее обстоятельство является одним из двух факторов, препятствующих широкому применению схем на базе НЛМ. Другим фактором является то, что практически любое изменение схемы требует изготовления новых фотошаблонов и опыт-

ных образцов приборов, что влечет за собой большие затраты средств и времени.

Примером крупного успеха в области автоматического проектирования на основе НЛМ служит разработка центрального процессора крупного компьютера «Система 370» в виде одного кристалла с размером стороны 7 мм, на котором размещено примерно 5 тыс. логических вентилях, содержащих свыше 45 тыс. компонентов, с необходимой трассировкой связей.

Устройства индикации состояния регистровых структур. Цифровые индикаторы часто являются единственными периферийными устройствами микроЭВМ. В них широко используются двоичные индикаторы для представления информации на магистралях адреса, данных, аккумулятора и т. п. и буквенно-цифровые индикаторы, например при вводе данных с клавиатуры.

Широкое распространение получили двоичные индикаторы с применением миниатюрных ламп накаливания и светодиодов, так как они совместимы со схемами ТТЛ-типа и имеют высокую надежность. На рис. 1.81, а приведена схема индикации при работе микросхемы К155ЛА7 на лампу накаливания типа НСМ-6,3-20.

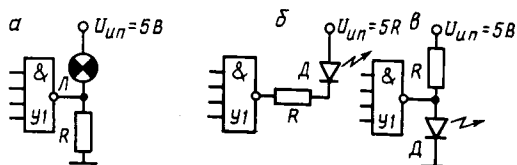


Рис. 1.81. Схемы двоичных индикаторов.

Резистор $R=680$ Ом при закрытом транзисторе вентиля Y1 обеспечивает на лампе напряжение около 15 % от номинального значения, при котором лампочка еще не светится, но подогревается, благодаря чему повышается ее сопротивление. Это существенно снижает ток через открывающийся транзистор вентиля Y1.

Светодиоды излучают световые волны при прохождении в прямом направлении электрического тока 5—20 мА, обеспечивающего хорошую видимость при нормальном освещении. На рис. 1.81, б приведена схема управления светодиодом низким уровнем на выходе схемы ТТЛ-типа. В данной схеме сопротивление

$$R = (U_{нп} - U_d - U_{вых}^0) / I_d. \quad (1.68)$$

Например, при $U_{нп} = 5В$, $U_d = 1,9В$, $U_{вых}^0 = 0,1В$, $I_d = 10$ мА сопротивление $R = 300$ Ом.

На рис. 1.81, в приведена схема управления светодиодом высоким уровнем на выходе схемы ТТЛ-типа с открытым коллектором. Для данной схемы

$$R = (U_{нп} - U_d) / I_d. \quad (1.69)$$

Поскольку $U_d \gg U_{вых}^0$, рекомендуется вместо резистора R применять активный генератор тока. Это устраняет броски тока при

включении и выключении светодиода и уменьшает помехи по цепям питания.

Отечественная промышленность серийно выпускает *цифровые и буквенные индикаторные лампы тлеющего разряда* (серии ИН), которые широко применяют в аппаратуре отображения информации, вычислительной технике и различных измерительных приборах. Для управления ими разработаны специальные микросхемы. На рис. 1.82, а приведена схема управления цифровым индикатором с помощью микросхемы К155ИД1.

В данной схеме сопротивление

$$R = (U_{\text{ип}} - U_{\text{л}} - U_{\text{вых}}^0) / I_{\text{л}}, \quad (1.70)$$

где $U_{\text{ип}} \geq 200$ В; $U_{\text{л}}$ — напряжение поддержания заряда на лампе; $U_{\text{вых}}^0$ — уровень напряжения на активном выходе микросхемы; $I_{\text{л}}$ — ток индикации.

Например, при $U_{\text{ип}} = 200$ В, $U_{\text{л}} = 150$ В, $U_{\text{вых}}^0 = 2,5$ В, $I_{\text{л}} = 2$ мА сопротивление $R = 24$ кОм.

Семисегментные индикаторы содержат семь светодиодов, расположенных так, как показано на рис. 1.82, б. С помощью такого индикатора можно индцировать цифры от 0 до 9 и некоторые буквы русского и латинского алфавита. Индикатор может иметь дополнительный светодиод для индикации десятичной точки. Аноды или катоды всех светодиодов-сегментов индикатора обычно объединены. Для представления десятичных цифр от 0 до 9 используются соответствующие комбинации сегментов, свечением которых управляет специально разработанная микросхема — семисегментный дешифратор. На рис. 1.82, б показана схема управления индикатором АЛ305 с помощью дешифратора К514ИД2.

Применение многоразрядных семисегментных индикаторов, показанных в последней схеме, имеет следующий недостаток: для каждого разряда необходим свой дешифратор, что приводит к большому количеству соединений с индикатором. Для устранения этого не-

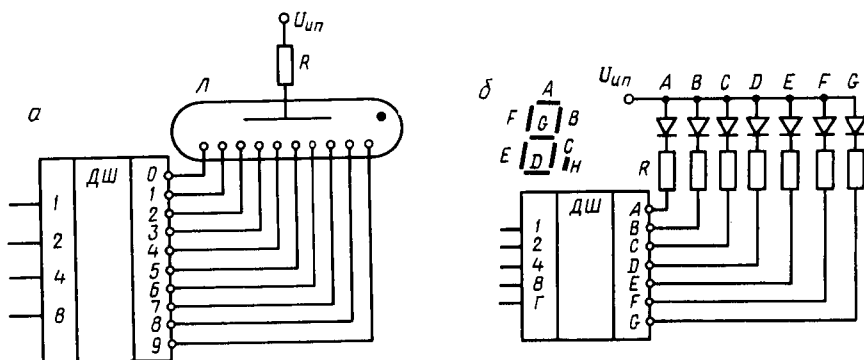


Рис. 1.82. Схемы десятичного индикатора на газоразрядной лампе (а) и буквенно-цифрового семисегментного индикатора (б).

достатка применяют *мультиплексное управление* многоразрядным индикатором, пример которого приведен на рис. 1.83. Схема содержит генератор, подключенный ко входу 3-разрядного двоичного счетчика СЧ (для 8-разрядного цифрового индикатора), который циклически устанавливает адреса четырех мультиплексоров МС 8→1 и дешифратора ДШ 1 из 8. Последний через буферный усилитель

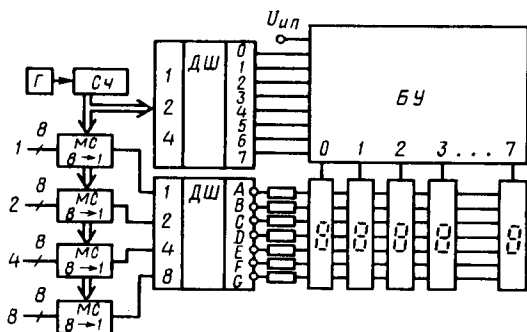


Рис. 1.83. Схема мультиплексорного управления 8-разрядным семисегментным индикатором.

БУ подключает источник питания к одному из восьми индикаторов (в точку соединения всех анодов сегментов). Все одноименные сегменты восьми индикаторов соединяются вместе и подключаются к соответствующим выходам семисегментного дешифратора. На входы семисегментного индикатора через четыре мультиплексора МС подается один из восьми 4-разрядных двоичных кодов индицируемых восьми десятичных разрядов.

Так как каждый десятичный разряд включен только в течение одного из восьми периодов тактового сигнала генератора, необходимо так выбрать резистор на выходе семисегментного дешифратора, чтобы ток сегмента был в восемь раз больше необходимого среднего значения.

Кроме рассмотренных типов индикаторов, широко применяются точечные матрицы светодиодов 5×7 , линейчатые светодиодные индикаторы, электронно-лучевые трубки, электролюминесцентные индикаторы, вакуумные катодно-люминесцентные индикаторы, жидкокристаллические индикаторы и др.

1.7. Узлы цифроаналоговых устройств средств сопряжения

Аналоговый мультиплексор. В отличие от цифровых мультиплексоров, в которых не происходит потерь информации при передаче цифровых сигналов, аналоговый мультиплексор — более сложное устройство. При разработке мультиплексора основная цель состоит в том, чтобы в измерение аналогового сигнала не вносилась ошибка. Это требование может быть выполнено при использовании идеального ключа. В разомкнутом состоянии идеальный ключ имеет бесконечное, а в замкнутом — нулевое сопротивление. Электромеха-

нические переключатели наиболее близки по своим характеристикам к идеальным, однако из-за низкой скорости переключения и малого срока службы не широко распространены.

Среди полупроводниковых переключателей наилучшими характеристиками обладают аналоговые ключи на полевых транзисторах с p — n -переходом и МДП-транзисторах. Сопротивление закрытого ключа достигает тысяч мегаом, и в этом отношении он становится близким к идеальному, находящемуся в разомкнутом состоянии. По сравнению с другими полупроводниковыми ключами как полевой транзистор с p — n -переходом, так и МДП-транзистор характеризуются полным отсутствием напряжения смещения в открытом состоянии. Открытый ключ в этом случае подобен линейному омическому резистору от нескольких десятков до нескольких сотен ом.

Динамические характеристики аналоговых коммутаторов определяются временем переключения. Это время для ключей на МДП-транзисторах составляет сотни наносекунд. Под влиянием паразитных емкостей C_0 ключей время установления коммутатора может существенно возрасти. Если суммарное сопротивление относительно входа открытого канала равно R_0 , то даже при отсутствии задержек в самих ключах время установления коммутатора $t_{уст.к}$ с погрешностью δ_k будет $t_{уст.к} = R_0 C_0 \ln 1/\delta_k$. Например, при $\delta_k = 0,001$, $R_0 = 1$ кОм и $C_0 = 50$ пФ время $t_{уст.к} = 350$ нс.

Т а б л и ц а 1.15

Микро- схемы	Назна- чение	Параметры						
		$U_{нп},$ В	$R_{отк},$ Ом	$U_{вх}^0,$ В	$U_{вх}^1,$ В	$\tau_{вкл},$ мкс	$I_{ут. вх.},$ нА	$I_{ут. вых.},$ нА
590КН1	Комму- татор	+15	200	0—0,8	4—15	1	50	50
590КН2	»	+15	300	0—0,8	4—15	0,3	50	70
590КН4	Ключ	+15	75	0—0,8	4—15	0,15	70	70
590КН5	»	+15	70	0—0,8	4—15	0,3	70	70
590КН6	Комму- татор	+15	75	0—0,8	4—15	0,3	50	70
590КН7	Ключ	+15	30	0—0,8	4—15	0,3	70	70

Следовательно, если при включении канала коммутатора сигнал на его входе изменится, то после окончания интервала $t_{уст.к}$ сигнал на выходе коммутатора будет повторять входной сигнал с некоторым запаздыванием. При дальнейшей обработке сигнала (например, аналого-цифровом преобразовании) это запаздывание необходимо учитывать. В табл. 1.15 приведены технические характеристики некоторых аналоговых ключей и коммутаторов в микроисполнении, выпускаемых отечественной промышленностью.

Основные способы подключения источников входного сигнала к аналоговому мультиплексору следующие:

1. Все аналоговые сигнала $U_{вхi}$ имеют общую точку («землю»), относительно которой отсчитывается и выходной сигнал мультиплек-

сора (рис. 1.84, а). Недостатком этого способа является незащищенность от синфазной помехи.

2. Мультиплексор имеет дифференциальные входы и соответственно два выхода (рис. 1.84, б). Следующее за подобным мультиплексором устройство должно также иметь дифференциальный вход. Достоинством усилителя с дифференциальным входом является его свойство подавлять синфазную помеху. Поэтому в тех случаях, когда измеряемые сигналы невелики, а требования к точности измерения достаточно высокие, следует применять дифференциальные схемы.

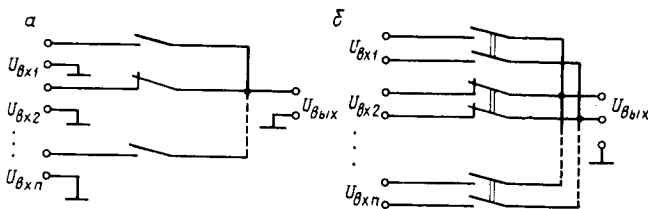


Рис. 1.84. Схема аналогового мультиплексора с однопроводными (а) и дифференциальными (б) входами.

Структура мультиплексора на основе ключей Кл 143КТ1 приведена на рис. 1.85. Предельная частота переключения определяется динамическими параметрами операционного усилителя и ключа.

Для построения мультиплексоров рационально использовать не отдельные ключи, а многоходовые интегральные коммутаторы, например серии 590, которые состоят из микросхем, выполненных на основе полупроводниковой КМДП-технологии. Параметры основных коммутаторов напряжения серии 590 приведены в табл. 1.15. Условное графическое обозначение мультиплексоров изображено на рис. 1.86. Преимуществом подобных схем по сравнению с другими схемами построения мультиплексоров являются возможность коммутации сигналов, уровень которых достигает напряжения питания (до 15 В), и стабильность переходного сопротивления в открытом состоянии при изменении входного сигнала.

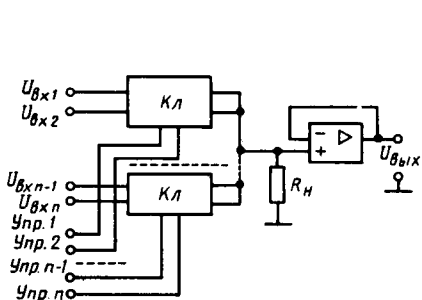


Рис. 1.85. Структура коммутатора аналоговых сигналов.

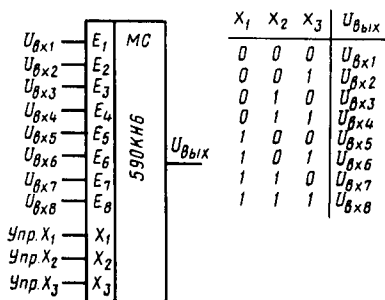


Рис. 1.86. Условное графическое обозначение интегрального коммутатора аналоговых сигналов 590КН6.

Аналоговые мультиплексоры вносят погрешности передачи аналоговых сигналов, причиной возникновения которых являются токи утечки через закрытый канал, паразитные емкости каналов, сопротивление открытого ключа.

Токи утечки проходят как в выключенном, так и во включенном каналах и являются суммой токов соответствующих утечек, указанных в нормативных документах (см. табл. 1.18), а также входного тока операционного усилителя (если он используется).

Входная емкость мультиплексора небольшая (~ 30 пФ), поэтому практически не вносит динамических погрешностей переключения, так как источником переключаемого напряжения являются операционные усилители, охваченные отрицательной обратной связью, с выходным сопротивлением ~ 100 Ом. Паразитные емкости закрытого канала (~ 70 пФ) и переходное сопротивление открытого канала образуют RC -цепь, постоянная которой для многоходовых мультиплексоров может быть достаточно большой. Например, 32-канальный мультиплексор, для которого $C_{\Sigma} = 1170$ пФ и $R_0 = 100$ Ом, образует RC -цепь с постоянной $\tau = 0,12$ мкс. Для установления напряжения выхода в пределах $0,01\%$ от переключаемого напряжения необходимо время, равное примерно 10τ , или $\sim 1,2$ мкс. В быстродействующих системах сбора это вносит ощутимый вклад в общую задержку передаваемого аналогового сигнала.

Токи утечки закрытого канала, примерно равные 50 нА, вызывают в 32-канальном мультиплексоре падение напряжения на сопротивлении открытого канала ~ 155 мкВ и практически не вносят погрешностей передачи сигнала, однако они неприемлемы для коммутации сигналов милливольтового диапазона. При необходимости построения мультиплексоров с большим числом каналов используют различные конфигурации многоступенчатых аналоговых коммутаторов АК с целью уменьшения указанных погрешностей переключений (рис. 1.87).

Схемы выборки — хранения. Для запоминания изменяющихся аналоговых сигналов на время преобразования, коммутации и других операций в системах сбора информации используют схемы

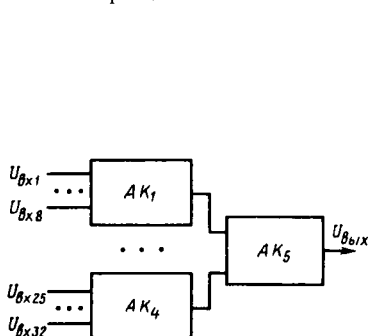


Рис. 1.87. Структура многоканального коммутатора аналоговых сигналов.

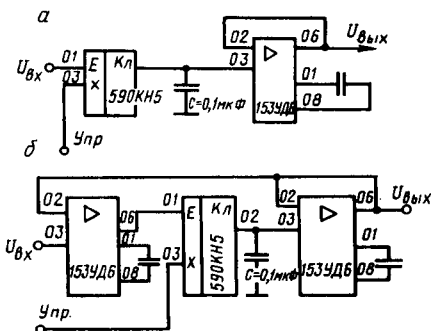


Рис. 1.88. Схемы устройства выборки — хранения.

выборки — хранения (рис. 1.88, а, на котором Кл — ключ). Элементом памяти схемы выборки — хранения является конденсатор С, подключаемый на время выборки к источнику сигнала. Основные параметры этого устройства: *время выборки*, *апертурное время* и *максимальное время хранения*.

Время выборки ограничено снизу постоянной времени цепи заряда $\tau_3 = R_3 C$ и требуемой точностью запоминания, а сверху — скоростью изменения запоминаемого аналогового сигнала. Сопротивление цепи заряда $R_3 = R_{\text{вых}} + R_{\text{о. кл}}$, где $R_{\text{вых}}$ — выходное сопротивление источника сигнала; $R_{\text{о. кл}}$ — сопротивление открытого ключа.

На практике время выборки несколько меньше расчетного за счет нелинейного режима источника сигнала в начальный момент заряда.

Когда устройство переходит в режим хранения, некоторая часть заряда утекает с конденсатора С через межэлектродную емкость ключа. Связанное с этим зарядом изменение напряжения на конденсаторе называют *погрешностью сдвига уровня* при переходе в режим хранения. С момента подачи команды «Хранение» до замыкания ключа проходит некоторое время, называемое *апертурным*. В течение апертурного времени амплитуда запоминаемого напряжения не определена, поэтому в зависимости от требуемой точности запоминания и максимальной частоты переключаемого сигнала на апертурное время накладываются ограничения. Например, $f_{\text{max}} = 10$ Гц, $\delta = 0,4\%$, $t_a \leq 100$ мкс.

В режиме хранения максимальная длительность времени хранения сигнала ограничивается сверху разрядом конденсатора С, обусловленным входным током операционного усилителя, дрейфом нуля усилителя, конечным значением входного сопротивления операционного усилителя, токами утечки ключа. Предельное изменение напряжения устройства выборки — хранения

$$\Delta U_{\text{в-х}} = \Delta U_{\text{в-х}}(i_{\text{вх}}) + \Delta U_{\text{в-х}}(e_{\text{др}}) + \Delta U_{\text{в-х}}(R_{\text{вх}}) + \Delta U_{\text{в-х}}(i_y),$$

где $\Delta U_{\text{в-х}}(i_{\text{вх}}) = (i_{\text{вх}}/C)t$; $\Delta U_{\text{в-х}}(e_{\text{др}}) = [e_{\text{др}}/(R_{\text{вх}}C)]t$; $\Delta U_{\text{в-х}}(R_{\text{вх}}) = [U_{\text{вых}}/(K_y R_{\text{вх}}C)]t$; $\Delta U_{\text{в-х}}(i_y) = (i_y/C)t$. (1.71)

Здесь $e_{\text{др}}$ — напряжение дрейфа.

Выражение (1.71) определяет максимальное время хранения при заданной точности операции запоминания.

При необходимости развязки источника сигнала от входа схемы выборки — хранения можно воспользоваться схемой с повышенной точностью, представленной на рис. 1.88, б, которая обладает высоким входным сопротивлением и большим коэффициентом подавления синфазных и сдвиговых погрешностей.

К важнейшим характеристикам устройства выборки — хранения относятся: *время выборки* — время от момента подачи сигнала на выборку до момента установления выходного сигнала; *апертурное время* — интервал времени между моментом подачи сигнала на хранение и моментом завершения аналого-цифрового преобразования; *скорость разрушения информации* — изменение выходного напряже-

ния вследствие разряда запоминающей емкости; *сквозная передача* — часть входного сигнала вследствие конечного сопротивления и разомкнутого ключа передается на выход.

Цифроаналоговые и аналого-цифровые преобразователи. *Цифроаналоговый преобразователь* (ЦАП) — функциональный узел, однозначно преобразующий кодовые комбинации цифрового сигнала в значения аналогового сигнала. Основой для нахождения однозначного соответствия может служить напряжение на выходе ЦАП

$$U_{\text{вых}} = E_0(x_1 2^{-1} + x_2 2^{-2} + \dots + x_n 2^{-n}), \quad (1.72)$$

где E_0 — опорное напряжение; $X\{x_1, x_2, \dots, x_n\}$ — цифровой код; x_i принимают значения 0 или 1.

При определенном E_0 каждому X_i на выходе устройства соответствует напряжение $U_{\text{вых}i}$.

В цифроаналоговых преобразователях используют три основных двоичных кода: прямой, смещенный, дополнительный. Графики соответствия цифровых кодов $X_{\text{вх}}$ и аналогового напряжения $U_{\text{вых}}$ при прямом (а), смещенном (б) и дополнительном (в) кодах показаны на рис. 1.89.

Наиболее просто определяется соответствие цифровых и аналоговых величин при *прямом коде*. Этот код удобен при преобразовании сигналов следящих систем, так как при переходе через нуль не ме-

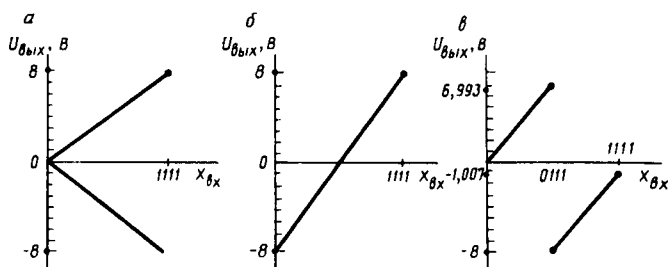


Рис. 1.89. Графики соответствия цифровых кодов и аналоговых напряжений.

няются старшие разряды кода, что позволяет реализовать линейный переход от малых положительных к малым отрицательным выходным напряжениям. Для преобразования как положительных, так и отрицательных кодов используют знаковый разряд, который управляет переключением выходного напряжения ЦАП.

Для исключения коммутирующих элементов из схемы ЦАП используют *смещенный код*, являющийся наиболее простым для реализации в схеме преобразователя.

При применении *дополнительного кода* положительные числа преобразуются так же, как для прямого кода, а отрицательные — двоичным дополнением соответствующего положительного числа (инверсия всех разрядов с последующим добавлением единицы в младший разряд).

Из табл. 1.16, соответствующей рис. 1.89, видно, что прямой код дает возможность использовать в два раза большее разрешение по сравнению со смещенным и дополнительным кодами.

Базовая схема ЦАП, реализующего выражение (1.72), показанная на рис. 1.90, а, содержит источник опорного напряжения E_0 , матрицы двоично-весовых резисторов, набор ключей и дифференциальный операционный усилитель. Основные недостатки этой схемы определяются необходимостью применения резисторов с большим диапазоном номиналов, например $1R-1024R$ для 10-разрядного ЦАП.

Т а б л и ц а 1.16

Прямой код		Смещенный код		Дополнительный код	
$X_{вх}$	$\pm U_{вых}, В$	$X_{вх}$	$\pm U_{вых}, В$	$X_{вх}$	$\pm U_{вых}, В$
0000	0,00	0000	-8,00	0111	7
0001	0,50	0001	-6,93	0110	6
0010	1,00	0010	-5,87	0101	5
0011	1,50	0011	-4,80	0100	4
0100	2,00	0100	-3,73	0011	3
0101	2,50	0101	-2,67	0010	2
0110	3,00	0110	-1,60	0001	1
0111	3,50	0111	-0,53	0000	0
1000	4,00	1000	0,53	1111	-1
1001	4,50	1010	1,60	1110	-2
1010	5,00	1010	2,67	1101	-3
1011	5,50	1011	3,73	1100	-4
1100	6,00	1100	4,80	1011	-5
1101	6,50	1101	5,87	1010	-6
1110	7,00	1110	6,93	1001	-7
1111	7,50	1111	8,00	1000	-8

Рациональным способом уменьшения количества номиналов резисторов является использование резистивной (лестничной) матрицы $R-2R$, изображенной на рис. 1.90, б. Выражение (1.72) реализуется схемой ЦАП (рис. 1.90, а) непосредственно, так как соотношение

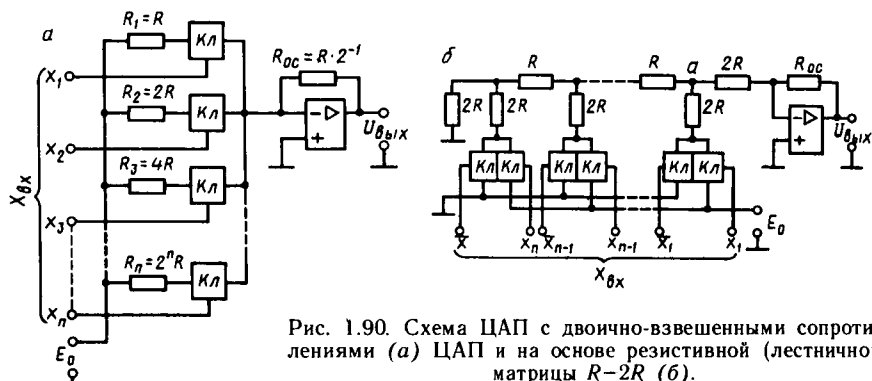


Рис. 1.90. Схема ЦАП с двоично-взвешенными сопротивлениями (а) ЦАП и на основе резистивной (лестничной) матрицы $R-2R$ (б).

R_{oc}/R_i равно весу соответствующего x_i . Схема ЦАП на основе матрицы $R-2R$ также реализует выражение (1.72). Так как потенциал суммирующей точки операционного усилителя равен нулю, то, анализируя эквивалентную схему, изображенную на рис. 1.91, а, можно записать для точки a

$$U_a = E_0 x_1 K_1 + E_0 x_2 K_2 + \dots + E_0 x_n K_n, \quad (1.73)$$

где K_i — коэффициент передачи E_0 в точку a от разряда цифрового кода с соответствующим индексом.

Значения коэффициентов можно определить, изменяя конфигурацию эквивалентной схемы, при условии равенства единице только одного разряда цифрового кода. На рис. 1.91, б приведены эти из-

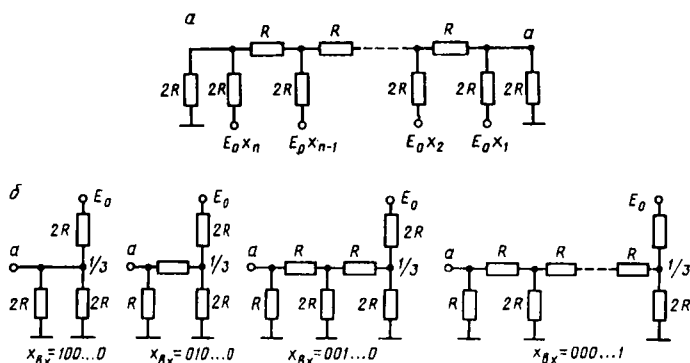


Рис. 1.91. Схемы резистивной (лестничной) матрицы $R-2R$.

менения для четырех разрядов: трех старших и одного младшего. Из рисунка следует, что

$$K_1 = 1/3; K_2 = (1/3)2^{-1}; K_3 = (1/3)2^{-2}; \dots; K_n = (1/3)2^{-n}. \quad (1.74)$$

Из (1.73) и (1.74) получим

$$U_a = (1/3)E_0(x_1 2^{-1} + x_2 2^{-2} + x_3 2^{-3} + \dots + x_n 2^{-n}).$$

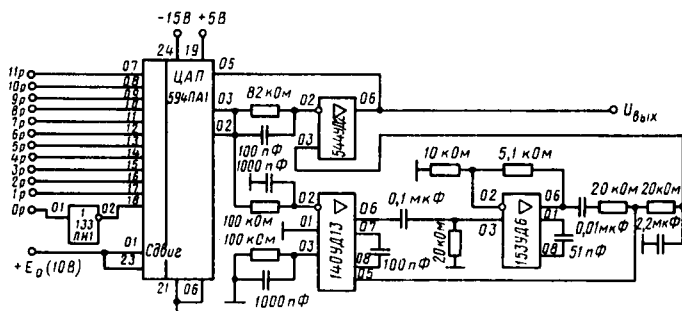


Рис. 1.92. Пример схемы ЦАП.

При $R_{oc}=3R$ (см. рис. 1.90, б) схема ЦАП реализует (1.72).

Практическую реализацию многоразрядных схем ЦАП рационально проводить на основе микросхем, содержащих основные блоки ЦАП в одном корпусе. На рис. 1.92 приведена принципиальная схема 12-разрядного ЦАП на микросхеме 594ПА1, содержащей резистивную матрицу и токовые ключи. Для уменьшения погрешностей от входных токов операционного усилителя введен усилитель 544УД2 с большим входным сопротивлением; для компенсации смещения нулевого уровня усилителя 544УД2 введен канал МДП-типа. Операционный усилитель 140УД13 является модулятором, а усилитель 153УД6 — демодулятором. ЦАП работает в дополнительном коде в диапазоне напряжений $\pm 10\text{В}$. Источники опорного напряжения E_0 , как правило, строятся на основе использования полупроводниковых стабилитронов. На рис. 1.93 приведены схемы

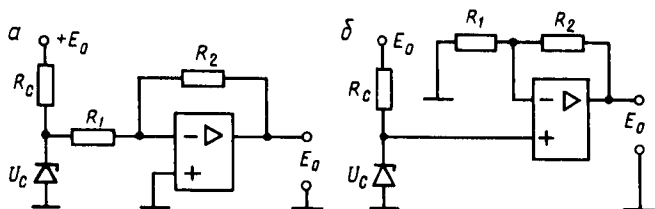


Рис. 1.93. Схемы источников опорного напряжения.

источников опорного напряжения: на основе операционного усилителя с параллельной обратной связью (а) и с последовательной обратной связью (б). Отношение сопротивлений R_2/R_1 определяется требуемыми значениями E_0 и U_c .

При использовании подобных схем необходимо выполнить условие $E_0 \leq (U_{нас} - \Delta U_{с\max})$, где $U_{нас}$ — граница динамического диапазона операционного усилителя; $\Delta U_{с\max}$ — максимальное температурное и временное изменение напряжения стабилизации. Температурная стабильность E_0 в данных схемах определена стабильностью U_c и выходного напряжения операционного усилителя.

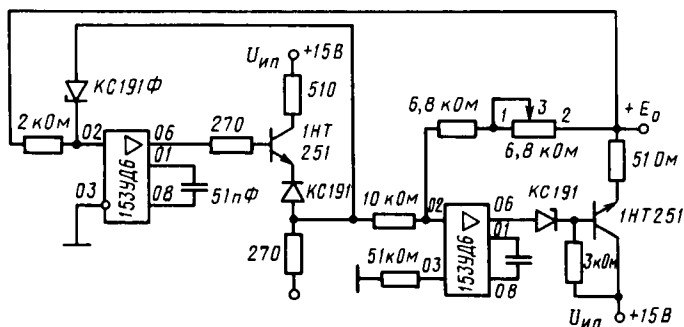


Рис. 1.94. Пример схемы прецизионного источника опорного напряжения.

Построение высокоточных ($\delta \leq 0,1 \%$) источников опорного напряжения связано с введением в схему реализации E_0 элементов, обеспечивающих температурную и временную стабилизацию тока опорного стабилитрона.

Пример практической реализации рассмотренного решения приведен на рис. 1.94. Здесь в качестве опорного выбран температурно-стабилизированный стабилитрон КС191Ф; транзисторы 1НТ251 введены для обеспечения необходимого тока стабилизации при максимальных изменениях температуры окружающей среды $\pm 50^\circ\text{C}$ и повышенной выходной мощности.

Аналого-цифровые преобразователи (АЦП) решают задачу поиска однозначного соответствия аналоговому сигналу цифрового кода. На вход АЦП поступает аналоговый сигнал, и после определенного конечного времени преобразования на его выходе появляется цифровой код.

В настоящее время разработано много различных методов аналого-цифрового преобразования, например, методы последовательного счета, поразрядного уравнивания, двойного интегрирования; с преобразованием напряжения в частоту, параллельного преобразования. Схемы преобразователей, построенных на основе перечисленных методов, могут содержать или не содержать ЦАП.

Схема и график работы АЦП последовательного счета приведены на рис. 1.95. Как видно из графика, время преобразования этого типа переменное и зависит от входного аналогового сигнала,

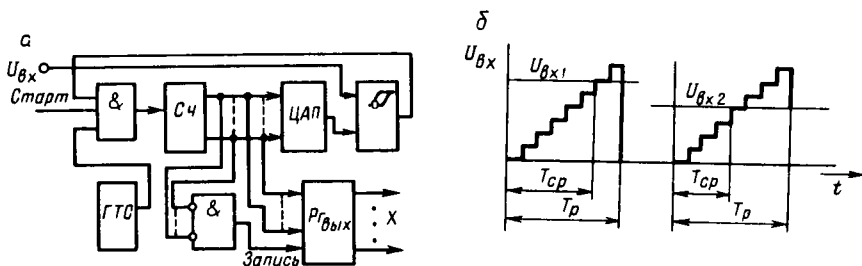


Рис. 1.95. Структура (а) АЦП последовательного счета и временная диаграмма его работы (б).

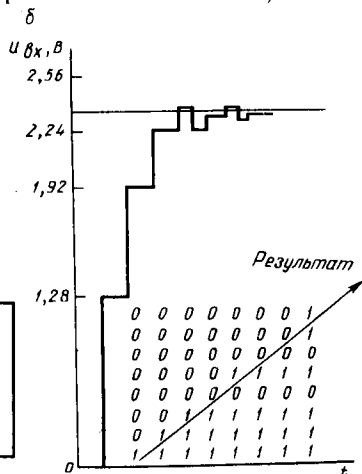
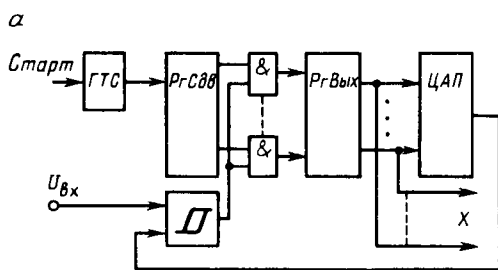
однако такт работы всего устройства постоянен и равен $T_p = T_0 2^n$, где T_0 — период генератора опорных импульсов, n — разрядность счетчика и собственно АЦП. Работа такого АЦП не требует синхронизации, что значительно упрощает построение схемы управления. С момента поступления сигнала «Старт» на выходе АЦП с частотой $1/T_p$ изменяются цифровые коды результата преобразования (частота $1/T_p$ — параметр, определяющий максимально допустимую частоту отслеживания входного сигнала, например $T_0 = 1 \text{ мкс}$, $n = 8$, $f_{отс} = 4 \text{ кГц}$).

Некоторым усложнением схемы управления, заменой суммирующего счетчика на реверсивный и введением элементов, обеспечивающих его работу, реализуется схема следящего АЦП. Этим дости-

гается значительное сокращение времени преобразования, однако подобные методы рационально использовать в системах с небольшим числом каналов преобразования, так как один АЦП работает на один канал.

В многоканальных системах сбора и обработки данных широко применяют АЦП *поразрядного уравнивания* (рис. 1.96, а, где $RrCдв$ и $Rr_{вых}$ — соответственно регистры сдвига и выхода). Схема

Рис. 1.96. Структура АЦП поразрядного уравнивания (а) и временная диаграмма его работы (б).



управления этого АЦП более сложная по сравнению с АЦП последовательного счета, но время преобразования значительно меньше. Например, так как $T_p = T_{0n}$, то при $T_0 = 1$ мкс $n = 8$, $T_p = 8$ мкс и $f_{отс} \approx 100$ кГц.

Временная диаграмма работы АЦП этого типа приведена на рис. 1.96, б. С момента поступления сигнала «Старт» генератор тактовых сигналов ГТС перемещает единичный сигнал на регистре сдвига $RrCдв$, с регистра сдвига через схему И единица записывается в регистр выхода, если $U_{вх} > U_{вых}$ ЦАП. Если $U_{вх} < U_{вых}$ ЦАП, то в соответствующий разряд регистра выхода записывается ноль, таким образом формируется цифровой код на выходе АЦП начиная со старшего разряда.

В АЦП *двойного интегрирования* используется метод предварительного преобразования напряжения во временной интервал, а затем измерения временного интервала. Преобразователь «напряжение — временной интервал» строится на основе интегратора, так как при постоянном входном напряжении выходное напряжение интегратора — линейная функция времени:

$$u_{вых}(t) = -[1/(RC)] \int_0^t u_{вх}(t) dt = -[1/(RC)] u_{вх} t.$$

На рис. 1.97 приведены схема и график работы АЦП двойного интегрирования. На первом этапе преобразования на вход интегратора через входной коммутатор подается напряжение $U_{вх}$ и в течение фиксированного времени T_1 реализуется первое интегрирование, в ре-

зультате которого на выходе интегратора устанавливается напряжение, пропорциональное $U_{вх}$, т. е. $U_{вых} = -U_{вх} T_1 / (RC)$.

На втором этапе преобразования на вход интегратора подключается E_0 , знак которого противоположен знаку $U_{вх}$, напряжение на выходе интегратора изменяется от $U_{вых\max}$ до 0 в течение времени

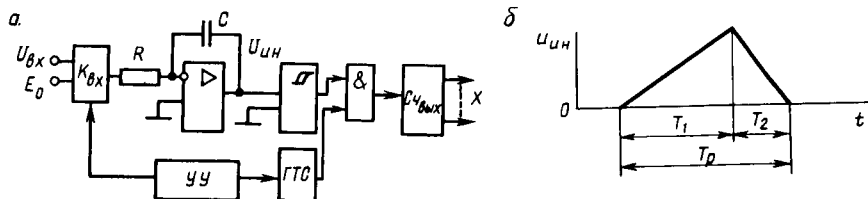


Рис. 1.97. Структура (а) АЦП двойного интегрирования и временная диаграмма его работы (б).

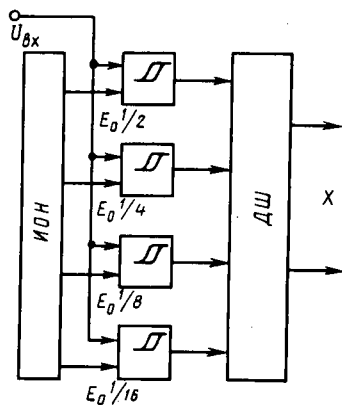
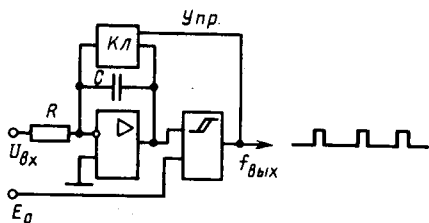
$T_2 = U_{вх} T_1 / E_0$. При достижении на выходе интегратора нулевого уровня «нулькомпаратор» прекращает поступление тактовых импульсов на вход счетчика результата.

В АЦП с преобразованием напряжения в частоту используется метод измерения частоты следования прямоугольных импульсов, основанный на заполнении счетчика импульсов в течение фиксированного времени. Эти преобразователи используют при медленно изменяющемся входном напряжении. Основное достоинство этих АЦП — простота схемотехнического решения. Небольшие габаритные размеры преобразователя «напряжение — частота» позволяют разместить его в непосредственной близости от датчиков исходной информации и тем самым реализовать АЦП с высоким соотношением сигнал/шум. Преобразователи (рис. 1.98, Кл — ключ) с операционными усилителями серий 154, 574 с малыми входными токами позволяют решить задачу аналого-цифрового преобразования за время $T = 10$ мкс при 8-разрядном выходном коде.

АЦП параллельного преобразования построены на основе двоич-

Рис. 1.98. Структура преобразователя «напряжение — частота».

Рис. 1.99. Структура АЦП параллельного преобразования.



но-взвешенного источника E_0 (рис. 1.99, где ИОН — источник опорного напряжения, ДШ — дешифратор). Достоинство таких преобразователей — высокое быстродействие, определяемое временем переключения компараторов. Частота преобразования может достигать 10—20 МГц при 3-разрядном выходном коде. Число компараторов равно числу разрядов выходного кода. Для получения двоичного выходного кода необходим на выходе дешифратор ДШ.

Подобные преобразователи рациональны при $n=2-4$ и необходимости быстрых аналого-цифровых преобразований.

Схема АЦП на основе микросхемы 572ПВ1, содержащей все блоки преобразователя поразрядного уравнивания, кроме выходного операционного усилителя и компаратора, приведена на рис. 1.100.

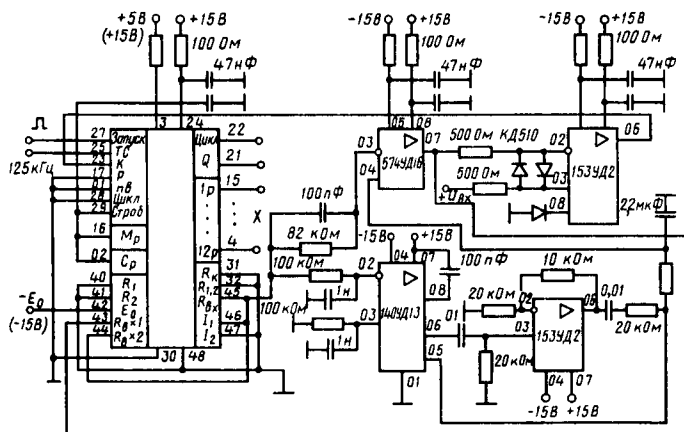


Рис. 1.100. Пример схемы АЦП.

Для получения максимального быстродействия выходной усилитель реализуется на основе микросхемы 574УД1Б, а компаратор — на основе операционного усилителя 153УД2.

Для компенсации смещения нулевого уровня 574УД1Б (50—100 мВ) в схему введен канал МДМ-типа (модулятор ОУ 140УД13 и демодулятор ОУ 153УД2). Приведенная на рис. 1.100 схема реализует точность преобразования до 10-го двоичного разряда в широком диапазоне изменения температуры окружающей среды при частоте преобразования $f \approx 300$ кГц.

В табл. 1.17 приведены основные параметры микросхем, на основе которых реализуются аналого-цифровые и цифроаналоговые преобразователи (n — число двоичных разрядов, $t_{пр}$ — время преобразования, $U_{пит}$ — напряжение питания).

Аналого-цифровые множительно-делительные устройства. При разработке системы обработки данных, связывающей измерительные датчики с входом ЭВМ, для получения рациональных результатов необходимо совместное выполнение ряда условий. Обычно в рам-

Таблица 1.17

Микросхемы	Назначение	Параметры			Микросхема (зарубежный аналог)
		n	$t_{\text{пр}}, \text{ мкс}$	$U_{\text{ин}}, \text{ В}$	
572ПВ1А, Б	АЦП — ЦАП (по- разрядного урав- новешивания)	12	130	+5,0; +15,0 —10—10 (E_0)	АД7570
572ПВ2	АЦП (двойного интегрирования)	13—14		+5,0; —5,0 —5—5 (E_0)	ICL7107
1113ПВ1	АЦП (последова- тельных прибли- жений)	10	30	+5,0; —15,0	АД571
1107ПВ1	АЦП (параллель- ные секции с вы- ходным дешифра- тором)	6	0,1	+5,0; —6,0 —0,075—1 (E_0) —1,9—2,1 (E_0)	ТДС1007
572ПА1А, В	ЦАП	10	5	+15,0 —22,5—22,5 (E_0)	АД7520
572ПА2А, В	ЦАП	12	10	+5,0; $\pm 15,0$ —22,5—22,5 (E_0)	—
594ПА1	ЦАП (секцион- ные переключате- ли взвешенных токов)	12	3,5	+5,0; —15,0 10 (9—11) (E_0)	АД562
1108ПА1А, Б	ЦАП	12 (А) 10 (Б)	0,4	+5,0; —15,0 10 (2—10,5) (E_0)	Н1562

ках некоторых ограничений на габаритно-весовую характеристику и стоимость стараются получить требуемые параметры системы по производительности (причем прямое увеличение быстродействия процессора не всегда возможно). Существует возможность решения поставленной задачи на основе аналого-цифровых множительно-делительных устройств (АЦМДУ).

Преобразователи аналоговых сигналов в цифровые и цифровых в аналоговые, кроме собственно преобразований, способны воспроизводить операции умножения и деления. Используя эти возможности, можно расширить функции устройства ввода данных в микропроцессор, проводя определенные функциональные преобразования сигналов, поступающих с датчиков до ввода их в процессор, и тем самым увеличить производительность системы обработки данных (СОД) в целом.

Рассмотрим возможности воспроизведения операций умножения на основе ЦАП. Операция умножения воспроизводится схемой ЦАП непосредственно, так как напряжение на выходе ЦАП

$$U_{\text{вых}} = E_0(x_1 2^{-1} + x_2 2^{-2} + \dots + x_n 2^{-n}).$$

Если $E_0 = U_{\text{вх1}}$, а $X\{x_1 x_2 \dots x_n\} = X_{\text{вх2}}$, то

$$U_{\text{вых}} = U_{\text{вх1}} X_{\text{вх2}} \quad (1.75)$$

Выражение (1.75) позволяет реализовать схему воспроизведения операции умножения аналогового сигнала на цифровой. Схемы

аналоговых и цифровых коммутаторов и схемы выборки — хранения (рис. 1.101, а) расширяют функции устройства.

Если кодовая комбинация на входах управления аналоговых АК и цифрового ЦК коммутаторов соответствует состоянию определенного открытого аналогового и цифрового каналов, т. е. Упр

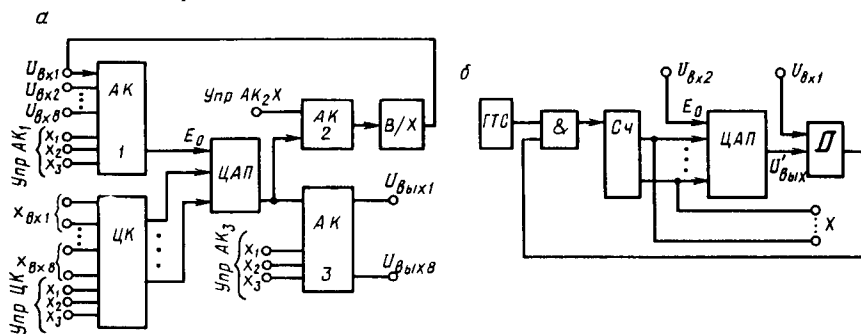


Рис. 1.101. Структура цифроаналогового умножителя (а) и делителя (б).

$AK_1 = 001$; Упр $AK_2 = 0$; Упр $AK_3 = 000$; Упр $ЦК = 000$, то напряжение на выходе устройства

$$U_{вых1} = U_{вх2} X_{вх1}.$$

Изменяя кодовые комбинации на входах Упр AK_1 , Упр AK_3 , Упр $ЦК$, можно получать любые произведения вида

$$U_{выхi} = U_{вхj} X_{вхk}; i = 1, 2, \dots, 8; j = 2, 3, \dots, 8; K = 1, 2, \dots, 8.$$

Введем временное разделение работы устройства на два такта t_1 и t_2 для записи результатов в блоке выборки — хранения В/Х. В этом случае: при t_1 и кодовых комбинациях на входах управления Упр $AK_1 = 001$; Упр $AK_2 = 1$; Упр $ЦК = 000$

$$U_{вх1} = U_{вх2} X_{вх1},$$

при t_2 и кодовых комбинациях на входах управления Упр $AK_1 = 000$; Упр $AK_2 = 0$; Упр $AK_3 = 000$; Упр $ЦК = 000$

$$U_{вых1} = U_{вх2} X_{вх1} X_{вх1}.$$

Изменяя кодовые комбинации на входах управления, можно получать произведения вида

$$U_{выхi} = U_{вхj} X_{вхl} X_{вхm}; i = 1, 2, \dots, 8; j = 2, 3, \dots, 8; m = 1, 2, \dots, 8.$$

Разделяя во времени работу устройства на три (или более) такта, можно получать произведения аналоговых сигналов на любое число цифровых кодов (от 1 до 8).

Воспроизведение операций деления реализуется на основе схемы АЦП. АЦП последовательного приближения показан на рис. 1.101, б. Напряжение на выходе ЦАП в любой момент времени равно $U'_{вых} = U_{вх2} X_{вых}$. Кодовые комбинации счетчика изменяются в соответствии с натуральным рядом чисел от 0 до N . В момент времени, когда $U'_{вых} =$

$= U_{\text{вх1}}$, состояние компаратора изменяется из «1» в «0» и запрещается поступление сигналов с генератора тактовых сигналов ГТС на суммирующий вход счетчика. В этом случае

$$U_{\text{вх2}} X_{\text{вых}} = U_{\text{вх1}} \text{ или } X_{\text{вых}} = U_{\text{вх1}} / U_{\text{вх2}}.$$

Введение в схему на рис. 1.101, б элементов для коммутации аналоговых сигналов, подобной коммутации аналоговых сигналов в схеме, изображенной на рис. 1.101, а, позволяет реализовать зависимости вида

$$X_{\text{вых}} = U_{\text{вхi}} / U_{\text{вхj}}; i = 1, 2, \dots, 8; j = 1, 2, \dots, 8.$$

Разделение работы устройства на n тактов выявляет возможность воспроизведения более сложных зависимостей:

$$U_{\text{вых}} = U_{\text{вхi}} / (U_{\text{вхj}} U_{\text{вхk}} \dots U_{\text{вхn}}), \\ i = 1, 2, \dots, 8; j = 1, 2, \dots, 8; k = 1, 2, \dots, 8; i \neq j; i \neq k; i \neq n.$$

Совмещая функции множительного и делительного устройств в рамках единого аналого-цифрового множительно-делительного устройства, можно сформировать вычислительный блок, функциональная характеристика которого имеет вид

$$z = x_1 x_2 \dots x_n / (y_1 y_2 \dots y_n),$$

где x, y, z — переменные функции времени, которые могут быть представлены как аналоговыми, так и цифровыми сигналами.

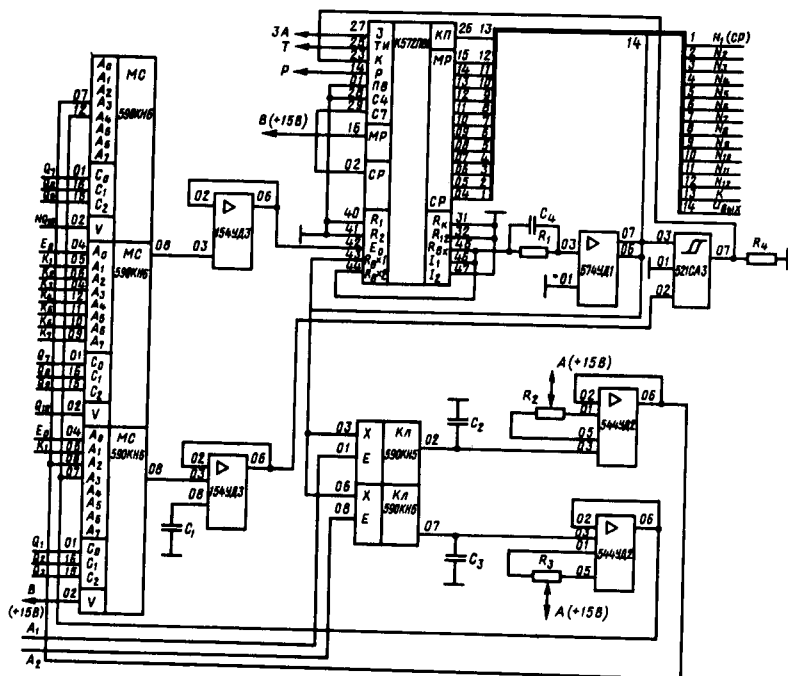


Рис. 1.102. Пример схемы АЦМДУ.

На рис. 1.102 приведена схема АЦМДУ на основе микросхемы 572ПВ1. Устройство реализует функции ЦАП, АЦП и АЦМДУ в зависимости от состояния управляющих кодовых комбинаций.

На рисунке микросхема 572ПВ1 имеет входы управления: ЗА — запуск, Т — вход ГТС, Р — режим (при $P=0$ реализуется режим АЦП, при $P=1$ — режим ЦАП). Режимы работы АЦМДУ зависят от кодовых комбинаций на входах управления (см. рис. 1.101).

Аналоговые микропроцессоры. Решение задачи рационального построения системы обработки данных может базироваться на использовании аналоговых микропроцессоров (АМП), структура которых формируется на основе следующей цепи преобразования и обработки сигналов:

$$АСВх \rightarrow АЦП \rightarrow ЦОС \rightarrow ЦАП \rightarrow АСВ_{вых},$$

где АСВх — входные аналоговые сигналы; АЦП — аналого-цифровое преобразование сигналов; ЦОС — цифровая обработка сигналов; ЦАП — цифроаналоговое преобразование; АСВ_{вых} — аналоговые сигналы выхода. Так как в основу АМП положена цифровая обработка сигналов, то системы обработки данных, построенные с применением АМП, обладают высокими эксплуатационными характеристиками. Примером АМП данного класса является программируемый микропроцессор 2920, разработанный специалистами фирмы «Intel» США.

Структура АМП 2920 приведена на рис. 1.103 и содержит следующие блоки: Г — генератор тактовой частоты (до 6,67 МГц), предусмотрена возможность работ АМП от внешней тактовой частоты; ППЗУ — программируемое постоянное запоминающее устройство хранения команд с информационным объемом 4528 бит (192 24-разрядных числа); ОЗУ — оперативное запоминающее устройство с информационным объемом 1000 бит (40 25-разрядных чисел),

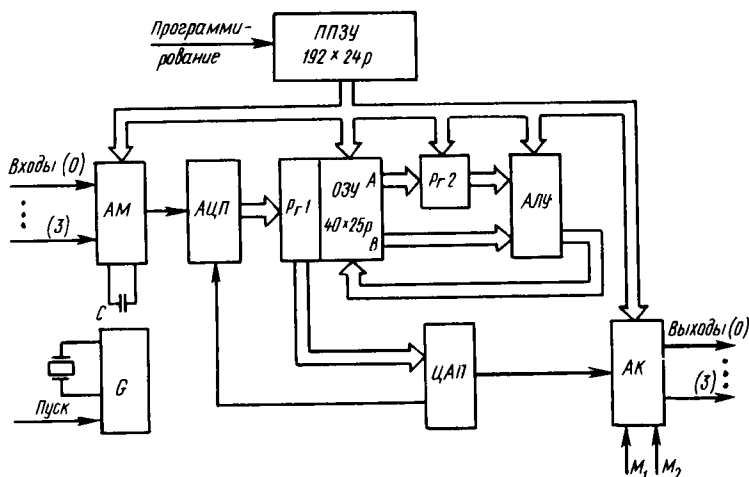


Рис. 1.103. Структура аналогового микропроцессора 2920.

имеет два выхода А и В и один вход В; Rg1 — многофункциональный регистр, реализует функции регистра аналого-цифрового преобразователя (АЦП) последовательного счета и регистра цифро-аналогового преобразователя (ЦАП). Обращение к Rg1 для записи или считывания информации идентично обращению к ОЗУ, т. е. адрес Rg1 содержится в соответствующих командах; Rg2 — сдвиговый регистр от 2 разрядов влево и до 13 разрядов вправо; АЛУ — арифметическо-логическое устройство, выполняет арифметические и логические операции, операции АЛУ реализуются с фиксированной запятой, количество разрядов 25, один — знаковый. При переполнении АЛУ вырабатывается максимальное число соответствующего знака; АМ — аналоговый мультиплексор входных аналоговых сигналов (0), ..., (3), реализует последовательную передачу аналоговых сигналов на вход АЦП, предусмотрена передача последовательного цифрового сигнала со входа (3) в ОЗУ; АК — аналоговый коммутатор выходов АМП (0), ..., (7), предусмотрена возможность вывода последовательных цифровых сигналов, управление выводом аналоговым или цифровым реализуется в зависимости от состояния M_1M_2 ; ЦАП — цифроаналоговый преобразователь 9-разрядных цифровых сигналов в соответствующие аналоговые сигналы.

Команды АМП 24-разрядные, двухадресные, содержат два адреса: А и В. Формат команды приведен на рис. 1.104.

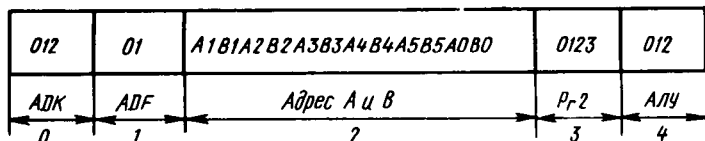


Рис. 1.104. Формат команды АМП2920.

Содержимое команды условно разделено на пять зон: зону 4 — код операции АЛУ — 3 разряда, зону 3 — управление регистром сдвига Rg2 — 4 разряда, зону 2 адреса А и адреса В — 12 разрядов, зону 1 — код аналоговых операций ADF — 2 разряда, зону 0 — код аналоговых операций ADK — 3 разряда. Перечень операций АЛУ приведен в табл. 1.18. Операции ADD, SUB, LDA, ABA, XOR выполняются различно в зависимости от результата выполнения аналоговой операции CND (K). Здесь же приведен перечень операций управления Rg2 (зона 3). В табл. 1.19 перечислены аналоговые операции АМП 2920.

Каждая команда АМП выполняется за четыре такта, минимальная длительность исполнения команды ~600 нс.

Конструктивно АМП размещен в 28-выводном корпусе (рис. 1.105). Обозначения и функции выводов следующие: 1, 2, 3, 5, 6, 26, 27, 28 — аналоговые выходы АВых (0), ..., АВых (7) соответственно; 4 — общий аналоговый; 7, 9 — внешний конденсатор схемы выборки — хранения; 8 — напряжение опоры 1—2 В; 10, 14, 13, 11 — аналоговые входы АВх(0), ..., АВх(3) соответственно; 12, 18 —

Таблица 1.18

Зоны команд АМП	Мнемонический код операций	Содержимое операции
Зона 4 (АЛУ)		
0 1 2		
0 1 1	ADD	$(A \times 2^N) + B \rightarrow B$
1 0 1	SUB	$B - (A \times 2^N) \rightarrow B$
1 1 1	LDA	$(A \times 2^N) + 0 \rightarrow B$
0 0 0	XOR	$(A \times 2^N) \oplus B \rightarrow B$
1 0 0	AND	$(A \times 2^N) \cdot B \rightarrow B$
1 1 0	ABS	$ (A \times 2^N) \rightarrow B$
0 0 1	ABA	$ (A \times 2^N) + B \rightarrow B$
0 1 0	LIM	$\text{sign}(A) \pm F.S. \rightarrow B$
0 1 1	ADD CND	$(A \times 2^N) + B \rightarrow B$ $B \rightarrow B$
1 0 1	SUB CND (K)	$B - (A \times 2^N) \rightarrow B$ $B + (A \times 2^N) \rightarrow B$
1 1 1	LDA CND (K)	$(A \times 2^N) \rightarrow B$ $B \rightarrow$
0 0 1	ABA CND (K)	$(A \times 2^N) + B \rightarrow B$
0 0 0	XOR CND (K)	$(A \times 2^N) \cdot B \rightarrow B$
Зона 3		
3 2 0 1		
1 1 0 0	R13	Сдвиг вправо на 13 разрядов
1 0 1 1	R12	То же, на 12 разрядов
...	...	...
0 0 0 0	R01	Сдвиг вправо на 1 разряд
1 1 1 1	R00	Отсутствие сдвига
1 1 1 0	L01	Сдвиг влево на 1 разряд
1 1 0 1	L02	То же, на 2 разряда

Таблица 1.19

Зона 1 команд АМП	Зона 0 команд АМП	Мнемонический код операций	Содержимое операций
0 1	0 1 2		
0 0	0 1 1	IN (K)	Ввод с K-входа АМ аналогового сигнала
1 0	1 1 1	OUT (K)	Вывод на K-выход АК аналогового сигнала
0 0	0 1 0	CVTS	Формирование знакового разряда при АЦП
0 1	1 1 1	CVT (K)	Формирование K-го разряда при ЦАП
0 0	1 0 1	EOP	Возврат к началу программы
0 0	1 0 0	NOP	Пустая операция
1 1		CND (K)	Анализ K-го разряда в Pr1
0 0	1 1 1	CNDS	Анализ знакового разряда в Pr1

напряжения питания — 5 В и 5 В соответственно; 15, 16 — внешний кварцевый резонатор; 17 — общий цифровой; 19 — внешняя тактовая частота; 21 — сброс к началу программы; 22 — сигнал перепол-

нения АМП; 23 — управление программированием АМП; 24, 25 — управление аналоговым или цифровым выходом. Для пояснения процедуры программирования АМП рассмотрим реализацию зависимости между аналоговыми переменными, которая задана выражениями:

$$U_{y0} = a_0 + a_1 U_{x0} + a_2 U_{x0}^2;$$

$$U_{y1} = 0,5(U_{x0} + U_{x1}),$$

где U_{x0} , U_{x1} — напряжения на входах АМП; U_{y0} , U_{y1} — напряжения на выходах АМП; a_0 , a_1 , a_2 — постоянные коэффициенты.

Подпрограмма ввода аналогового сигнала с входа АМП имеет такую последовательность команд:

- | | | |
|----------|----------|----------|
| 1) IN(K) | 13) CVT7 | 25) CVT3 |
| 2) IN(K) | 14) NOP | 26) NOP |
| 3) IN(K) | 15) NOP | 27) NOP |
| 4) IN(K) | 16) CVT6 | 28) CVT2 |
| 5) IN(K) | 17) NOP | 29) NOP |
| 6) IN(K) | 18) NOP | 30) NOP |
| 7) IN(K) | 19) CVT5 | 31) CVT1 |
| 8) IN(K) | 20) NOP | 32) NOP |
| 9) NOP | 21) NOP | 33) NOP |
| 10) CVTS | 22) CVT4 | 34) CVT0 |
| 11) NOP | 23) NOP | 35) NOP |
| 12) NOP | 24) NOP | 36) NOP |

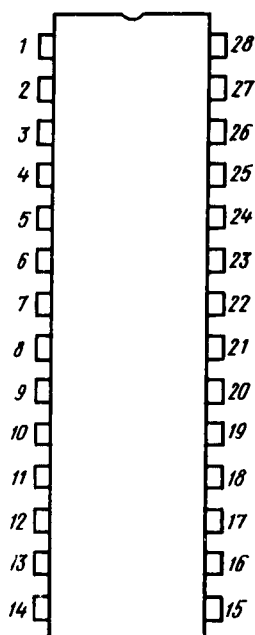


Рис. 1.105. Схема корпуса АМП2920.

Назовем ее «Вход с входа $K U_{xk}$ ».

Подпрограмма вывода аналогового сигнала с выхода АМП имеет такую последовательность команд:

- | | | |
|--------|------------|------------|
| 1) NOP | 4) OUT (K) | 7) OUT (K) |
| 2) NOP | 5) OUT (K) | 8) OUT (K) |
| 3) NOP | 6) OUT (K) | 9) OUT (K) |

Назовем ее «Вывод с выхода $K U_{yk}$ ».

Вычисление значения функции U_{y0} выполняется такого вида последовательностью команд:

- | | |
|-----------|--------|
| 1) LDA 03 | 01 R00 |
| 2) LDA 04 | 48 R00 |
| 3) LDA 05 | 03 R00 |
| 4) AND 05 | 49 R00 |
| 5) ADD 04 | 05 R00 |
| 6) AND 03 | 03 R00 |
| 7) AND 03 | 50 R00 |
| 8) ADD 04 | 03 R00 |

Эту последовательность команд назовем «Вычисление U_{y0} ».

Полный текст программы реализации заданной функциональной зависимости имеет следующий вид:

- 1—36) «Ввод с входа 0 U_{x0} »
 37) LDA 01 DAR R00 NOP

38—46) «Вычисление U_{y0} »; 38—73) «Ввод с входа 1 U_{x1} »
 74) LDA 02 ДАР R00 NOP
 75) LDA 07 01 R01 NOP
 76) АДД 07 02 R01 NOP
 77) LDA ДАР04 R00 NOP
 78—86) «Вывод с выхода 0 U_{y0} »
 87) LDA ДАР 07 R00 NOP
 88—96) «Вывод с выхода 1 U_{y1} »
 97) EOP

По командам 37, 74 выполняется запись в ОЗУ числовых значений U_{x0} , U_{x1} соответственно. Команды 75, 76 реализуют вычисление функции U_{y1} . Команды 77, 87 обеспечивают запись числовых значений U_{y0} , U_{y1} в регистр R01 с целью последующего цифроаналогового преобразования.

II. КОНТРОЛИРУЮЩИЕ И ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

В разделе приводится анализ и классификация объектов управления и контроля, даются примеры задач, решаемых ИУВС; определяются требования к ИУВС; рассматриваются классификация и примеры архитектуры однопроцессорных и мультипроцессорных ИУВС; приводятся краткая характеристика языков программирования и их выбор для конкретной ИУВС; излагаются принципы использования языков высокого уровня.

2. МИКРОПРОЦЕССОРНЫЕ КОНТРОЛИРУЮЩИЕ И ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

2.1. Особенности проектирования контролирующих систем на микроЭВМ с универсальной системой команд

*Основные направления развития. Автоматизированные системы контроля (АСК) можно рассматривать как подкласс средств информационной техники — информационно-измерительных вычислительных систем (ИИВС). В них необходимая информация об объекте контроля (ОК) получается с помощью системы измерительных датчиков, установленных на нем. На начальной стадии развития автоматизированных систем для получения количественных характеристик при контроле использовались специализированные вычислительные устройства, работающие по жесткой программе. В качестве элементной базы в них применялась дискретная полупроводниковая техника. Эта ступень развития средств сбора и обработки информации соответствовала *первому поколению АСК.**

По мере развития цифровой вычислительной техники в АСК происходила унификация ряда измерительных и вычислительных процедур: после аналого-цифрового преобразования результаты измерений вводились в ЭВМ, где программным путем выполнялись операции обработки, необходимые для получения результатов контроля состояния объектов. Наиболее широкое развитие получили АСК с включенной в их состав мини-ЭВМ, появившиеся в конце 60-х годов. Наличие мини-ЭВМ позволило перейти от выполнения ряда операций измерения и обработки информации аппаратным путем по жесткой программе к их выполнению по изменяющимся программам, к варьированию в процессе работы программным путем структуры и состава работающих функциональных устройств системы. Объедини-

нение измерительной части и ЭВМ привело к появлению *второго поколения АСК*.

Автоматизированные системы контроля *второго поколения* характеризуются адресным сбором информации, преимущественно программной обработкой информации на ЭВМ, использованием микроэлектронных устройств малой и средней степени интеграции, блочно-модульным принципом построения. Для автоматизации управления и контроля технологических процессов были разработаны управляющие вычислительные машины (УВМ) «Днепр», «Электроника-100», М-6000, М-40, М-400 и др. Создание автоматизированных систем контроля на базе УВМ предусматривает разработку блоков связи машины с объектом контроля, датчиков и измерительных цепей, интерфейсных устройств к ним, а также специфических алгоритмов и программ управления сбором и обработкой информации (*централизованные АСК*).

Широкое применение микропроцессорных средств привело к появлению *третьего поколения АСК* — микропроцессорных автоматизированных систем контроля. В этих системах с помощью микропроцессорных средств выполняются обработка и промежуточное хранение информации в местах, максимально приближенных к ее получению. Из-за распределенного характера процессов сбора информации, ее обработки и хранения эти системы называют *децентрализованными*.

Децентрализованные АСК отличаются от централизованных АСК не столько рассредоточенным расположением аппаратуры, сколько распределением функций. В этих системах на конструктивно автономных процессорах выполняется обработка данных, поэтому центральный процессор берет на себя только наиболее сложные и срочные задачи, что позволяет одновременно осуществлять операции обработки в нескольких местах контролируемого объекта, существенно уменьшать потоки информации, сделать более рациональным обмен информацией между функциональными устройствами и центральным процессором, а также получать более быструю реакцию на запросы ОК.

Дальнейшее развитие микропроцессорной техники, в особенности появление микроЭВМ с достаточно высокой производительностью и большим объемом оперативной памяти, позволило перейти к созданию полностью автономных систем сбора и обработки измерительной информации. При этом сама система стала значительно проще и надежнее, так как полностью исключились расходы, затрачиваемые на программные и аппаратурные средства связи с центральным обрабатывающим устройством.

Обоснование выбора типа микроЭВМ. Промышленностью выпускаются модули, с помощью которых можно собрать автоматизированные системы самого различного назначения. Для АСК наиболее часто применяют модули: ввода аналоговых сигналов, осуществляющие прием и преобразование аналогового сигнала в двоичный код и ввод его в микроЭВМ; ввода-вывода дискретных сигналов; вывода аналоговых сигналов, преобразующих выходной двоичный код

микроЭВМ в аналоговый сигнал; таймера для отсчета временных интервалов и т. п.

Основные принципы построения и проектирования микропроцессорных систем контроля в настоящей книге будут рассмотрены на примерах использования микроЭВМ «Электроника-60», «Электроника МС 1201.01» и выпускаемых к ним периферийных устройств и модулей.

В настоящее время отечественная промышленность выпускает микроЭВМ, которые с успехом работают в автоматизированных системах контроля: «Электроника С-5», СМ-1800, «Электроника-60», «Электроника МС 1201.01». Две последние модели микроЭВМ программно-совместимы с рядом микро- и мини-ЭВМ, получивших широкое распространение как в СССР, так и за рубежом. К семейству отечественных мини-ЭВМ данного типа относятся «Электроника 100-25», «Электроника-79», СМ-3, СМ-4, а зарубежных — мини-ЭВМ PDP-11 и микроЭВМ LSI-11. Характерной особенностью указанных ЭВМ является наличие *общей магистрали*, служащей для связи всех устройств, входящих в систему. Подобно тому, как для 60-х годов стала образцовой архитектура ЭВМ IBM/360, для 70-х и 80-х годов для мини- и микроЭВМ стала образцовой архитектура PDP-11. Популярности ЭВМ PDP-11 способствует также широкий диапазон разработанного и продолжающегося развиваться математического обеспечения.

Принцип объединения компонентов ЭВМ и периферийных устройств с помощью общей магистрали позволяет легко модифицировать систему в соответствии с конкретными потребностями. Форма связи каждого устройства, подключенного к общей магистрали, — стандартная.

В зависимости от конкретных условий применения следует также учитывать и такие факторы, как быстродействие, объем оперативной памяти, мощность потребления, габаритные размеры, надежность и т. п. Наиболее полно вышеприведенным требованиям отвечает одноплатная микроЭВМ «Электроника МС 1201.01», воплощающая в себе значительные достижения отечественной технологии и к настоящему времени являющаяся наиболее совершенной среди аналогичного класса моделей. Основа ее — микропроцессорный комплект серии К 1801, выполненный по *n*-канальной МОП-технологии. МикроЭВМ «Электроника МС 1201.01» предназначена для использования в АСУТП, контрольно-измерительной аппаратуре, в системах обработки цифровой информации общего назначения. Широкая область применения обеспечивается встраиванием ее в аппаратуру потребителя.

МикроЭВМ представляет собой систему функциональных устройств, связь между которыми осуществляется через общую магистраль (единый системный канал обмена информацией) (рис. 2.1, где ЦП — процессор; ОЗУ — оперативное запоминающее устройство емкостью 28К (К-1024) 16-разрядных слов; СПЗУ — системное постоянное запоминающее устройство емкостью 4К слов; УПВВ — устройство последовательного ввода-вывода информа-

ции; УБПИ — устройство байтового параллельного интерфейса; УИГМД — устройство интерфейса на гибком магнитном диске).

Имеется возможность установки дополнительной микросхемы ПЗУ объемом 4К 16-разрядных слов в контактирующее устройство КУ, расположенное на плате.

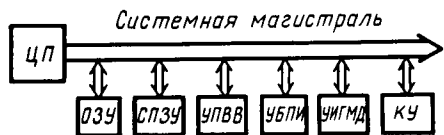


Рис. 2.1. Схема микроЭВМ «Электроника МС 1201.01 (02)».

МикроЭВМ обеспечивает значительную гибкость, модульную простоту и упрощение разработки АСК. Выбор типа микроЭВМ целесообразно проводить, руководствуясь возможностью использования стандартного программного обеспечения существующих микро- и мини-ЭВМ; наличием развитого периферийного оборудования; наращиванием технических возможностей микроЭВМ.

Расположение основных устройств на плате микроЭВМ «Электроника МС 1201.01» показано на рис. 2.2.

Процессор ЦП выполнен на основе БИСК 1801 ВМ1, содержащей около 50 тыс. транзисторов. Кристалл имеет размеры 5×5 мм и помещен в 42-выводной керамический корпус. Процессор является основным устройством микроЭВМ, который выполняет все необходимые

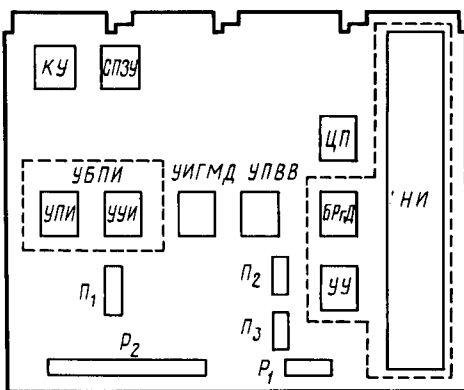


Рис. 2.2. Схема расположения основных устройств на плате микроЭВМ «Электроника МС 1201.01 (02)».

операции по приему команд, их исполнению, обработке внешних и внутренних прерываний программы, а также осуществляет управление общей магистралью.

Оперативное запоминающее устройство ОЗУ состоит из накопителя информации НИ; устройства управления ОЗУ УУ; буферного регистра данных БРГД; переключателя банков Пз.

Накопитель информации НИ выполнен на основе 32 микросхем типа K565PY6 с динамическим хранением информации. Полная емкость накопителя составляет 32К 16-разрядных слов, или 8 банков памяти по 4К слов каждый. Для пользователя доступны лишь 7 банков памяти, т. е. 28К слов.

Устройство управления УУ выполнено на основе микросхемы К 1801ВП1-030, в его функции входит управление обменом информа-

ции НИ с общей магистралью и регенерации (освежения) информации в НИ.

Буферный регистр данных БРгД выполнен на микросхеме К1801ВП1-034 и предназначен для временного хранения данных после завершения их выборки из НИ.

Переключатель банков П₃ предназначен для отключения отдельных банков ОЗУ из области адресации микроЭВМ. Необходимость такой операции возникает, например, в случае установки пользовательского ПЗУ в контактирующее устройство КУ.

Устройство байтового параллельного интерфейса УБПИ предназначено для связи микроЭВМ с внешними устройствами по асинхронному параллельному каналу ввода-вывода. Оно состоит из устройства управления интерфейсом УУИ на основе микросхемы К1801ВП1-033 и устройства передачи информации УПИ на основе микросхемы К1801ВП1-034. Базовым устройством для этого канала принято мозаичное печатающее устройство типа DZM-180.

Устройство последовательного ввода-вывода информации УПВВ предназначено для связи микроЭВМ с внешними устройствами по асинхронному последовательному каналу ввода-вывода. Оно выполнено на основе микросхемы К1801ВП1-035. Базовым устройством для него является алфавитно-цифровой дисплей 15ИЭ-00-013.

Устройство интерфейса накопителя на гибком магнитном диске УИГМД предназначено для связи с накопителем на магнитном диске ГМД-70 и выполнено на основе микросхемы К1801ВП1-033.

Переключатели П₁ и П₂ служат для задания режимов начального пуска, адресов регистров и векторов прерываний внешних устройств.

Разъем Р₁ служит для подключения внешнего устройства к последовательному каналу ввода-вывода, а разъем Р₂ — для подключения устройств к байтовому параллельному каналу ввода-вывода, а также для подключения накопителя ГМД-70.

МикроЭВМ «Электроника МС 1201.01» унифицирована по конструктивному исполнению, системе команд, организации и интерфейсу с микроЭВМ типа «Электроника-60», что позволяет наращивать технические возможности микроЭВМ «Электроника МС 1201.01» за счет подключения через общую магистраль дополнительных типовых функциональных модулей, а также модулей, разработанных пользователем.

Приведем основные технические характеристики микроЭВМ «Электроника МС 1201.01»:

1. Габаритные размеры 252×296×12 мм.
2. Масса не более 0,8 кг.
3. Мощность потребления не более 12,6 Вт (ток не более 2,4 А от источника +5 В).
4. Быстродействие при выполнении команд типа «Сложение»: а) при регистровом методе адресации — (400 ± 100) тыс. операций/с; б) при косвенно-регистровом методе адресации — (180 ± 40) тыс. операций/с.
5. Нарботка на отказ не менее 10 000 ч.
6. Нарботка на сбой не менее 500 ч.

7. Срок службы не менее 10 лет.

Как видно, слово «микро» можно отнести лишь к массогабаритным характеристикам ЭВМ, но никак не к быстродействию и широким функциональным возможностям.

Более подробно о принципах организации микроЭВМ и системе команд будет сказано ниже.

Программное обеспечение. Эффективность АСК в значительной мере определяется качеством ее программного обеспечения. Чтобы АСК выполняла заданные ей функции, в памяти ЭВМ должна находиться последовательность команд, реализующая эти функции, т. е. программа. *Программа* определяет порядок, в котором машина извлекает из памяти или из регистров внешних устройств ту или иную часть информации, коды операций (сложение, вычитание, сравнение с нулем, пересылки и т. д.), которые должны выполняться, адреса хранения результатов, адреса, откуда должны выбираться следующие команды при выполнении некоторых условий в исходных данных или в результатах вычислений.

Единственный язык, который «понимает» ЭВМ,— это язык, основанный на двоичной системе счисления. Хотя разработчик АСК и может использовать этот машинный язык, но затраты труда на разработку даже сравнительно небольшой программы при этом были бы огромны, что связано с самой природой двоичного счисления. Поэтому созданы специальные программы, использующие ЭВМ для перевода программ с языков более высокого уровня на язык машины. Эти программы составляют часть *системы программирования*, поставляемой изготовителем вычислительной машины.

Программная совместимость микроЭВМ «Электроника МС 1201.01» с рядом мини-ЭВМ (СМ-3, СМ-4, «Электроника 100-25») позволяет использовать весь широкий ассортимент программного обеспечения, разработанный для этих мини-машин. Разработчик программного обеспечения АСК может создавать свои программы непосредственно на мини-ЭВМ. С незначительной модификацией или даже вообще без изменений многие из этих программ можно прямо загружать в микроЭВМ для проверки и окончательной отладки. Таким образом, программная совместимость не только облегчает процесс написания и отладки программ для АСК, но и сокращает сроки проектирования системы в целом, поскольку разработка аппаратурной и программной частей системы может вестись параллельно.

Рассмотрим программные средства, необходимые разработчику программного обеспечения АСК. В вычислительной машине команды и данные хранятся в памяти в двоичном коде. В принципе, можно вводить программу в вычислительную машину по одной команде с пульта управления, располагая командой в нужном месте памяти. Неудобство этого метода станет очевидным каждому, кто попытается воспользоваться им. Человеку трудно читать и распознавать длинные последовательности двоичных цифр. Значительно удобнее использовать символические адреса и мнемонические коды команд. Поэтому программы сначала записывают на языке ассемблера, а затем с помощью специальной программы АССЕМБЛЕР выполняется автома-

тический перевод (трансляция) в последовательность машинных кодов.

Язык ассемблера — это отображение машинного языка. Иначе говоря, имеет место соответствие «один к одному» между машинными и символическими командами. Написание программы на языке ассемблера представляет собой трудоемкую задачу, так как от программиста требуется доскональное знание организации ЭВМ и ее системы команд. Поэтому большинство программ записывают на языках высокого уровня, которые позволяют выполнять наиболее распространенные операции, такие, как вычисление выражений, присвоение значений, условные действия, циклы и т. п., используя в каждом случае только один оператор этого языка. Для трансляции программы, написанной на языке высокого уровня, в последовательность машинных команд, с помощью которых решается поставленная задача, необходим Компилятор. Разница между ассемблером и Компилятором заключается в том, что последний переводит один оператор языка высокого уровня в ряд машинных команд. По сравнению с языком ассемблера использование языка высокого уровня для составления программ приводит к сокращению времени и стоимости разработки. В отличие от языка ассемблера языки высокого уровня являются машинно-независимыми или проблемно-ориентированными. В настоящее время широко распространены языки высокого уровня ФОРТРАН, КОБОЛ, ПЛ/1, ПАСКАЛЬ и др.

Однако имеются и некоторые аргументы в пользу программирования на языке ассемблера [6]. Теоретически всегда есть возможность написать такую программу на языке ассемблера, которая была бы не менее компактной и эффективной, чем программа на машинном языке, полученная с помощью Компилятора. Компилированная программа в зависимости от типа Компилятора, вычислительной машины и назначения может быть на 0—300 % длиннее и медленнее оптимальной программы на машинном языке. Из приведенных рассуждений следует, что если создаваемая программа будет эксплуатироваться интенсивно и не будет требовать частых изменений, то ее целесообразно писать на языке ассемблера. Примером тому является системное программное обеспечение многих вычислительных машин, написанное на языках ассемблера.

Другим средством для обработки программ, написанных на языке высокого уровня, является Интерпретатор. В отличие от Компилятора Интерпретатор реально выполняет исходную программу, написанную на языке высокого уровня, по одному оператору, а не транслирует каждый ее оператор в последовательность машинных команд, реализуемых позднее при прогоне программы. Благодаря такой особенности Интерпретатора существенно облегчается отладка программы, в нее легко вносить изменения и дополнения. Язык интерпретирующего типа прост в изучении и эксплуатации. Успехи в разработке таких языков делают доступным программирование большому числу людей, не имеющих специальных знаний в области вычислительной техники.

Другими средствами, необходимыми для разработки программ, являются редактор текста, компоновщик и отладчик. *Редактор текста*

используется для ввода текста исходных программ в ЭВМ и последующего редактирования. *Программа-компоновщик* служит для связи объектных модулей, являющихся результатом трансляции программ, написанных на языках высокого уровня, и их последующего исполнения на ЭВМ. *Программа-отладчик* выполняет программу по одной или нескольку команд за один шаг, что позволяет программисту проверять результат выполнения отдельных частей программы и обнаруживать ошибки.

Перечисленные выше средства создания и отладки программ выполняются под управлением операционных систем. Для рассматриваемых программно-совместимых микро- и мини-ЭВМ наибольшее распространение получили операционные системы ОС ДВК, РАФОС, ОС РВ, RSX-11, RSX-11М и др.

Структура микроЭВМ с общей магистралью. В основу организации микроЭВМ «Электроника МС 1201.01» положен принцип связи между устройствами с помощью одной *общей магистрали* (рис. 2.3).

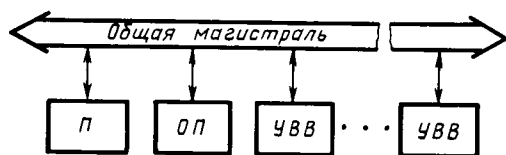


Рис. 2.3. Структура микроЭВМ с общей магистралью.

Для каждого устройства, подсоединенного к общей магистрали, вид связи одинаков. Процессор П использует один и тот же набор сигналов как для связи с ячейками оперативной памяти ОП, так и для связи с периферийными устройствами. Каждой ячейке памяти, регистрам процессора и регистрам периферийных устройств присваивается определенный адрес магистрали. Благодаря такой структуре все команды для данных, хранящихся в оперативной памяти ОП, в равной мере могут использоваться и для данных в регистрах периферийных устройств. Этот принцип — весьма существенная особенность микроЭВМ с общей магистралью, так как одно и то же множество команд можно применять как для вычислений, так и для управления вводом-выводом. Специальные команды ввода-вывода становятся ненужными, и ввод-вывод информации может быть совмещен с ее обработкой. Благодаря двунаправленным и асинхронным передачам устройства могут посылать, принимать и обмениваться данными между собой. Организация работы общей магистрали по асинхронному принципу «запрос — ответ» позволяет согласовать работу устройств, работающих в самом широком диапазоне частот.

Связь между устройствами, соединенными общей магистралью, осуществляется по принципу «задатчик — исполнитель». В любой момент времени имеется только одно устройство, которое управляет магистралью и носит название «задатчик». Это устройство управляет работой магистрали при осуществлении связи с другим устройством, соединенным с магистралью и именуемым «приемником». Типичным примером этого соотношения является процессор П, служащий в качестве задающего устройства и выбирающий команду из оперативной

памяти ОП (память всегда исполняющее устройство). Задающим устройством, например, может быть накопитель на магнитном диске (НМД), передающий данные в ОП. Таким образом, связь между устройствами — динамическая.

Общая магистраль используется процессором П и всеми устройствами ввода-вывода УВВ. Какое устройство принимает на себя управление магистралью, определяет система приоритета. Таким образом, каждое устройство, подсоединенное к магистрали и способное стать задатчиком, имеет присвоенный ему приоритет.

В рассматриваемой микроЭВМ имеется одна линия приоритетных прерываний. В том случае, когда два (или более) устройства, которые способны стать управляющими устройствами для магистрали, одновременно посылают запросы на использование магистрали, управление передается тому устройству, электрическое соединение которого находится ближе к процессору.

Обмен данными между устройствами, подключенными к магистрали, может осуществляться в режимах: 1) программном; 2) по прерыванию; 3) прямого доступа к памяти (ПДП).

Программный режим — наиболее универсальный. По инициативе и под управлением программы между задающим и приемным устройствами могут пересылаться полные 16-разрядные слова или 8-разрядные байты информации. Информацией могут быть команды, адреса или данные. Обычно процессор как задающее устройство выбирает команды из памяти и операнды из памяти или регистров, а после выполнения команд засылает результаты в память или регистры.

В режиме работы по прерыванию обмен данными происходит по требованию периферийного устройства. При этом процессор приостанавливает выполнение текущей программы, чтобы обслужить запрашивающее устройство. После завершения выполнения программы обслуживания процессор возобновляет выполнение прерванной программы с того места, где она была прервана. Поскольку в общем случае процессор способен выполнить от десятка до тысячи команд в течение промежутка времени между двумя последовательными передачами данных от устройства ввода-вывода, экономически нецелесообразно вынуждать простаивать его в течение этого времени. Передача данных по прерыванию позволяет процессору работать одновременно с процессом ввода-вывода и получать информацию о моменте его завершения.

В режиме прямого доступа к памяти (ПДП) обмен данными осуществляется без программного управления со стороны процессора и является самым быстрым способом передачи данных между памятью и внешним устройством. Адресацию и управление размерами передаваемого массива должно обеспечить устройство, получившее прямой доступ к памяти. Обычно режим ПДП используется при обмене массивами данных между накопителями на магнитных дисках или лентах (НМД или НМЛ) и оперативной памятью.

Рабочими циклами магистрали при взаимодействии с процессором являются: «Чтение» — перенос слова данных от внешнего устройства

в процессор; «Запись» — перенос слова или байта данных из процессора во внешнее устройство; «Чтение — модификация — запись» — перенос слова данных от внешнего устройства в процессор, за которым следует запись слова (байта) данных из процессора во внешнее устройство.

Для каждой команды процессора требуется один или несколько рабочих циклов магистрали. В первую очередь выполняется цикл «Чтение», при котором команда выбирается из ячейки памяти (адрес ячейки задан счетчиком команд). Если больше нет операндов, для которых необходимо обращение к памяти или регистрам устройств ввода-вывода, то для выполнения команды никаких дополнительных рабочих циклов не требуется. Однако если необходимо обращение к памяти или к внешнему устройству, то требуется один (или больше) дополнительный рабочий цикл.

Следует обратить внимание на разницу между прерываниями и операциями прямого доступа к памяти. *Прерывания* изменяют состояние процессора и поэтому могут иметь место только в промежутках между командами процессора. *Операции прямого доступа к памяти* могут выполняться внутри исполнения команды в перерывах между отдельными рабочими циклами магистрали, поскольку эти операции не изменяют состояния процессора.

Цикл магистрали «Чтение — модификация — запись» — примечательная особенность ЭВМ. По этому циклу происходят считывание данных из адресуемой ячейки памяти, модификация этих данных (т. е. выполнение операции арифметическо-логическим устройством) и запись в ту же ячейку памяти без повторения адреса ячейки. Исключение повторной выдачи адреса ячейки памяти на магистраль приводит к экономии времени и, как следствие, к увеличению производительности ЭВМ.

Организация памяти. В микроЭВМ «Электроника МС1201.01» каждое слово содержит 16 бит. Наименьшей адресуемой частью памяти является байт, состоящий из 8 бит. Два байта составляют слово. Память можно рассматривать как последовательность ячеек с номером (адресом), присвоенным каждой ячейке (рис. 2.4).

Если оперативная память содержит N байт, то последовательность адресов байтов будет $0, 1, 2, \dots, N - 1$. Адреса слов совпадают с адресами четных байтов, и их последовательность выглядит как $0, 2, 4, 6, \dots, N - 2$.

Адреса памяти удобно записывать в восьмеричной системе счисления. На рис. 2.4 показана структура оперативной памяти максимального объема для рассматриваемой микроЭВМ. Она содержит 56К байт, или 28К слов.

С помощью 16-разрядного слова процессор способен прямо адресовать 64К байт, или 32К слов, однако последние 8К байт, или 4К слов, выделяются для адресации регистров внешних устройств и регистров процессора. Таким образом, адреса слов $160000_8 - 177776_8$ относятся не к ячейкам оперативной памяти, а к различным регистрам. Следует иметь также в виду, что первые 200_8 слов (адреса $0 - 376_8$) резервируются системой для векторов внутренних и внеш-

них прерываний. По этой причине обычно в практике программирования начиная с адреса 400_{16} отводят некоторое число ячеек памяти для использования аппаратным указателем стека, а далее располагаются программы вплоть до самой последней ячейки памяти с адресом 15776_{16} .

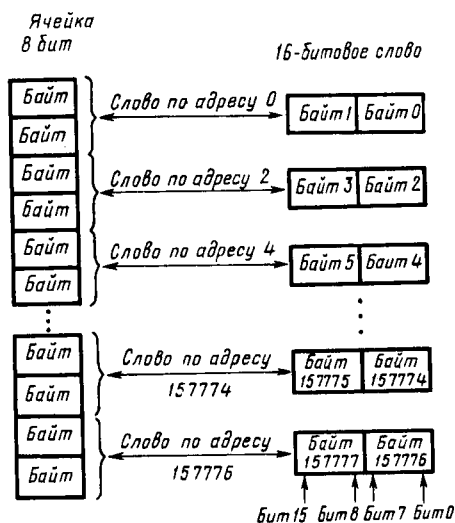
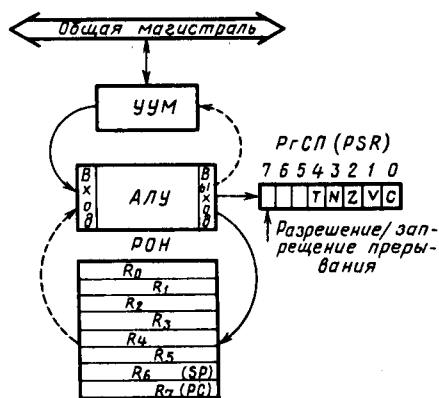


Рис. 2.4. Структура адресного пространства микроЭВМ.

Рис. 2.5. Логическая схема процессора микроЭВМ.



Логическая организация процессора. Структура процессора, состоящего из регистров общего назначения РОН, арифметическо-логического устройства АЛУ, регистра состояния процессора РПС и устройства управления магистралью УУМ, показана на рис. 2.5.

Процессор, подключаемый к общей магистрали как подсистема, управляет временным распределением общей магистрали для устройств ввода-вывода, декодирует команды и выполняет арифметические и логические операции. Он содержит восемь (R_0 — R_7) 16-разрядных регистров общего назначения РОН, которые могут использоваться как накапливающие, индексные, с автоувеличением и автоуменьшением или как указатели стеков для временного запоминания данных в оперативной памяти, для выборки операндов и записи результатов при выполнении арифметических и логических операций аналогично ячейкам памяти и регистрам внешних устройств. Операции по выполнению команд с регистровым методом адресации не требуют рабочих циклов общей магистрали. Обмен же данными с оперативной памятью и регистрами внешних устройств выполняется с участием общей магистрали и занимает более длительное время. По этой причине применение регистров общего назначения для хранения операндов и промежуточных результатов увеличивает производительность ЭВМ. Регистры R_6 и R_7 , помимо указанного выше, выполняют еще специальные функции. Регистр R_6 (SP) используется как аппаратный указатель стека и содержит адрес последней за-

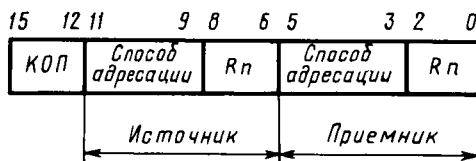
полненной ячейки стека. Система автоматически изменяет значение его содержимого при обработке внутренних и внешних прерываний, а также при работе с подпрограммами. Регистр R7 (PC) служит в качестве счетчика команд, который после выборки очередной команды всегда содержит адрес следующей выбираемой команды. Обычно он служит только для адресации команд, а не как накапливающий регистр.

Возможны следующие виды передачи данных: регистр — регистр, регистр — память, память — регистр, память — память. Кроме того, в процессоре имеется отдельный 8-разрядный регистр состояния процессора PrCII (PSR). Разряд 7 служит для управления разрешением или запрещением прерываний от внешних устройств. Разряды 6 и 5 в данной модели микроЭВМ для работы не нужны. Разряд 4 (обозначен T) используется системным математическим обеспечением при отладке программ (позволяет осуществить покомандное исполнение программы). Разряды 3—0 (обозначены как N, Z, V, S) содержат информацию о результатах выполнения предыдущих операций. Эти разряды имеют следующее значение: N=1 — если результат был отрицательным, Z=1 — если результат был равен 0, V=1 — если было переполнение при выполнении арифметических операций, C=1 — если был перенос из 15-го разряда.

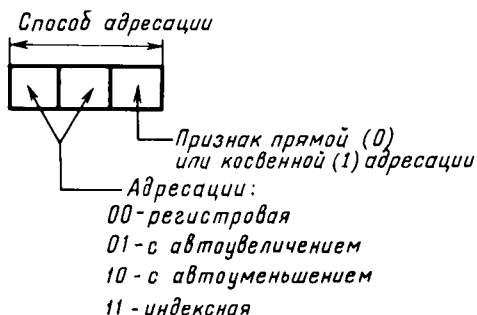
Структура одноадресной команды имеет вид

Поле адреса приемника занимает разряды 0—5, код операции (КОП) — разряды 6—15.

Структура двухадресной команды имеет вид



Способ адресации кодируется тремя двоичными разрядами:



Таким образом, существует всего восемь способов адресаций (4 прямых и 4 косвенных).

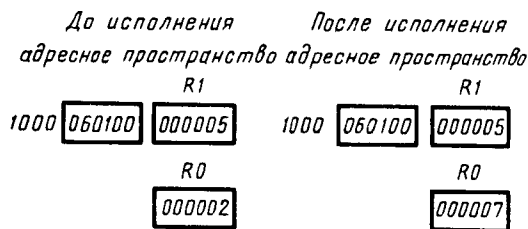
Учитывая важность адресации для понимания основных принципов организации и работы микроЭВМ, рассмотрим способы адресации подробнее.

Регистровая адресация (код 0₈). В качестве операнда используется содержимое адресуемого регистра. Это самый простой и наиболее быстрый способ извлечения операнда, не требующий обращения к оперативной памяти. Например, по команде

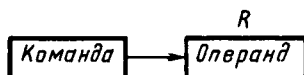
Счетчик команд	Машинный код	Мнемническое обозначение
1000	06 01 00*	ADD R1, R0

* На этом рисунке и на последующих рисунках адреса источника и приемника подчеркнуты. Первые две цифры машинного кода представляют собой код операции (в данном примере — сложить).

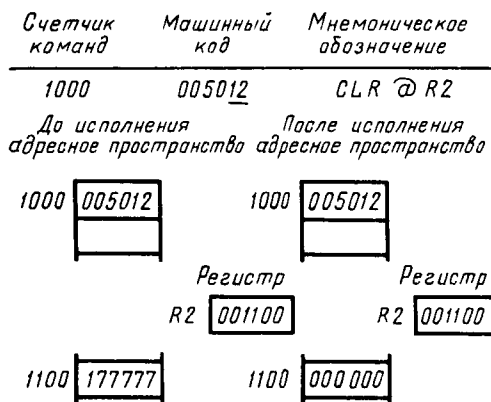
содержимое регистра R1 складывается с содержимым регистра R0 и результат направляется в регистр R0:



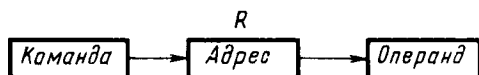
С помощью диаграммы регистровую адресацию можно изобразить так:



Регистровая косвенная адресация (код 1₈). Содержимое адресуемого регистра используется как адрес операнда. Например, команда

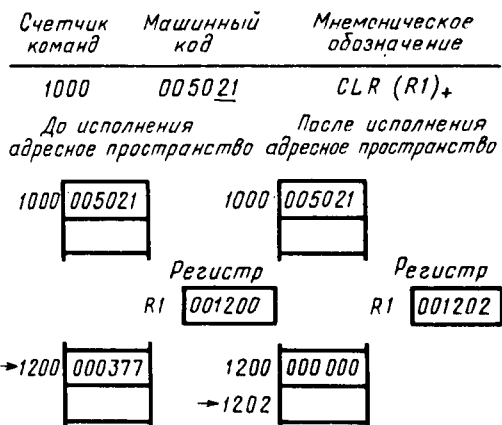


очищает ячейку памяти, адрес которой представлен содержимым регистра R₂. В приведенном примере это ячейка с адресом 1100. Условно этот тип адресации выглядит так:

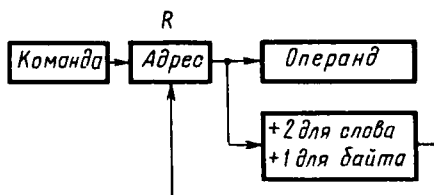


Адресация с автоувеличением (код 2₈). Содержимое адресуемого регистра рассматривается как адрес операнда. После получения адреса содержимое регистра автоматически увеличивается на 1 при байтовых операциях, а при операциях со словами -- на 2.

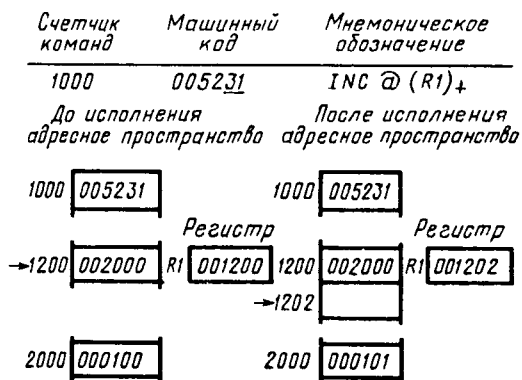
Содержимое регистров R6 и R7 (SP и PC) всегда увеличивается на 2. Этот способ адресации очень удобен для обработки таблиц (последовательно расположенных в памяти данных). Например, команда



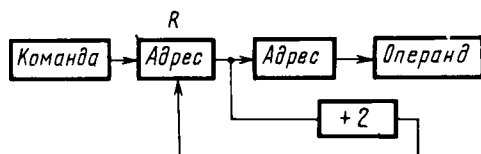
интерпретируется так: очистить ячейку памяти, адрес которой указан регистром R1, а затем сдвинуть указатель на следующее слово. Условное изображение этого вида адресации следующее:



Косвенная адресация с автоувеличением (код 3₈). Содержимое регистра используется как указатель ячейки памяти, в которой находится адрес операнда, затем содержимое регистра увеличивается на 2 (в том числе и для байтовых команд). Например, команда



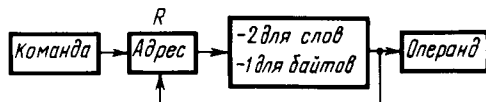
увеличивает на 1 содержимое ячейки памяти, адрес адреса которой указан содержимым регистра R1, затем содержимое регистра R1 увеличивается на 2. Этот способ адресации удобен для обработки таблиц, состоящих из адресов. Ниже приведено его условное изображение:



Адресация с автоуменьшением (код 4₈). Содержимое регистра сначала уменьшается на 1 для байтовых операций и на 2 для операций со словами и затем используется как адрес операнда. Содержимое регистров R6 и R7 в любом случае уменьшается на 2. Например, команда

Счетчик команд	Машинный код	Мнемоническое обозначение
1000	005044	CLR -(R4)
		До исполнения После исполнения
		адресное пространство адресное пространство
1000	005044	1000 005044
		Регистр Регистр
	R4 001200	R4 001176
1176	174000	→ 1176 000000
→ 1200		1200

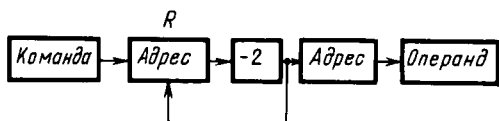
уменьшает содержимое регистра R4 на 2, затем очищает ячейку памяти, адрес которой определяет новое значение содержимого регистра R4. Нетрудно видеть, что при адресации с автоуменьшением адресация к ячейкам памяти идет в противоположном направлении, чем при адресации с автоувеличением. Этот способ адресации также удобен при обработке таблиц. Его условное изображение имеет вид



Косвенная адресация с автоуменьшением (код 5₈). Содержимое регистра всегда уменьшается на 2 и затем применяется в качестве указателя слова, содержащего адрес операнда. Например, команда

Счетчик команд	Машинный код	Мнемоническое обозначение
1000	005055	CLR @-(R5)
До исполнения адресное пространство		После исполнения адресное пространство
1000	005055	1000 005055
1304 →1306	R5 001306	→1304 001500 R5 001304 1306
1500	010101	1500 000000

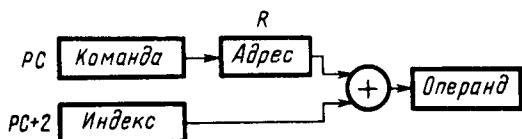
уменьшает содержимое регистра R5 на 2, затем очищается ячейка памяти, адрес которой определяется новым содержимым регистра R5. Этот способ адресации можно представить так:



Индексная адресация (код 6₈). Содержимое адресуемого регистра складывается с индексным словом, следующим за командой, и используется как указатель ячейки памяти, где находится адрес операнда. Ни содержимое индексного слова, ни содержимое регистра после исполнения команды не изменяются. Например, команда

Счетчик команд	Машинный код	Мнемоническое обозначение
1000	005064	CLR 200 (R4)
1002	000200	
До исполнения адресное пространство		После исполнения адресное пространство
1000	005064 000200	1000 005064 000200
	Регистр R4 002000	Регистр R4 002000
2200	070707	2200 000000

очищает ячейку памяти, адрес которой определяется как сумма 200 и содержимого регистра R4. Индексную адресацию условно можно представить так:

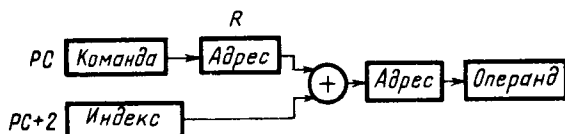


Косвенная индексная адресация (код 7₈). Содержимое индексного регистра складывается с индексным словом, следующим за командой, и используется как указатель ячейки памяти, где находится адрес операнда. Например, по команде

Счетчик команд	Машинный код	Мнемоническое обозначение
1000	005070	CLR @500(R0)
1002	000500	
До исполнения адресное пространство		После исполнения адресное пространство
1000	005070	1000 005070
1002	000500	1002 000500
Регистр R0 002000		Регистр R0 002000
2500	003000	2500 003000
3000	050505	3000 000000

произойдет очистка содержимого ячейки с адресом 3000, так как регистр R0 содержит значение 2000, а индекс равен 500.

Условно этот способ адресации можно представить так:



Если из восьми рассмотренных способов адресации отнести некоторые способы адресации к содержимому счетчика команд, то можно определить еще четыре удобных и эффективных способа адресации, получивших специальные названия: непосредственная, абсолютная, относительная и косвенная относительная адресации.

Непосредственная адресация (код адреса операнда 27₈). Этот способ адресации образуется применением адресации с автоувеличением к содержимому счетчика команд. Слово, следующее за командой, используется как непосредственный операнд. Таким образом, операнд-константа включается в саму команду. Например, команда

Счетчик команд	Машинный код	Мнемоническое обозначение
1000	062700	ADD #20, R0
1002	000020	
До исполнения После исполнения		
1000	062700	Регистр 1000 062700
1002	000020	R0 005000 1002 000020 R0 005020

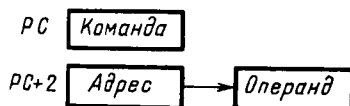
обеспечит сложение константы 20 с содержимым регистра R0. Условно непосредственную адресацию можно представить так:



Абсолютная адресация (код адреса операнда 37₈). Этот режим адресации образуется применением косвенной адресации с автоувеличением к содержимому счетчика команд. Слово, следующее за командой, является адресом операнда. Например, команда

Счетчик команд	Машинный код	Мнемоническое обозначение
1000	005037	CLR @ #2000
1002	002000	
До исполнения После исполнения		
1000	005037	1000 005037
1002	002000	1002 002000
2000	161616	2000 000000

выполняет очистку ячейки памяти с адресом 2000. Условно этот способ адресации можно обозначить так:



Относительная адресация (код адреса операнда 67₈). Этот способ адресации образуется применением индексной адресации к содержимому счетчика команд. Адрес операнда опреде-

ляется суммированием индексного слова, следующего за командой, и содержимого счетчика команд. Фактически индексное слово указывает, насколько далеко вперед или назад операнд отстоит от команды. Например.

Счетчик команд	Машинный код	Мнемоническое обозначение
α 1000	005067	CLR 1200
$\alpha+2$ 1002	000174	
$\alpha+4$ 1004	(следующая команда)	
До исполнения адресное пространство		После исполнения адресное пространство
1000	005067	1000 005067
1002	000174	1002 000174
1200	010101	1200 000000

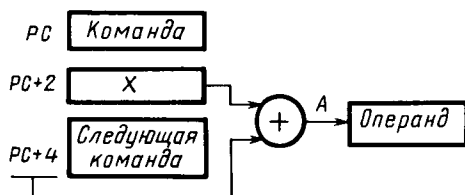
Исполнительный адрес

$$A = \alpha + 4 + X,$$

где α — адрес команды; $X = A - \alpha - 4 = 1200_8 - 1000_8 - 4_8 = 174_8$ — смещение (индекс).

Если при обращениях к памяти используется этот способ адресации, то программа загружается в любое место памяти и правильно выполняется, поскольку, несмотря на изменение абсолютных адресов необходимых операндов, расстояние между операндами и командами остается неизменным. Такая адресация называется *позиционно-независимой*.

Относительную адресацию условно можно представить так:

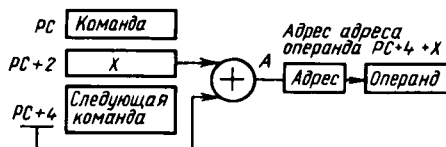


Относительная косвенная адресация (код адреса операнда 77_8). Этот способ адресации образуется применением косвенной индексной адресации к содержимому счетчика команд. Адрес адреса операнда определяется суммированием индексного слова, следующего за командой, и содержимого счетчика команд. Например, команда

Счетчик команд	Машинный код	Мнемоническое обозначение
1000	005077	CL R @ 1070
1002	000064	
1004	„следующая команда“	
...	...	
1070	002000	
...	...	
2000	177400	

До исполнения	После исполнения
адресное пространство	адресное пространство
1000 005077	1000 005077
1002 000064	1002 000064
1004 	1004
1070 002000	1070 002000
2000 177400	2000 000000

очищает ячейку памяти по адресу 2000. Условно этот способ адресации можно представить так:



Представляет интерес анализ четырех оставшихся способов адресации с использованием счетчика команд. Анализ проведем на примерах.

Команда с регистровой адресацией

Счетчик команд	Машинный код	Мнемоническое обозначение
1000	010701	MOV R7, R1

вполне допустима. Она запишет содержимое счетчика команд в регистр R1:

До исполнения	После исполнения
R1 177777	R1 001002

т. е. в R1 окажется адрес следующей команды. Пусть регистр R7 находится по месту адреса приемника, тогда по команде

<i>Счетчик команд</i>	<i>Машинный код</i>	<i>Мнемоническое обозначение</i>
1000	06 27 07	ADD #2, R7
1002	00 00 02	
1004	„следующая команда”	

текущее значение счетчика команд увеличится на 2, т. е. произойдет пропуск следующей команды. Однако проще и эффективнее выполнять безусловную передачу управления с помощью команды BR, так как она занимает только одну ячейку памяти.

Команда с косвенной регистровой адресацией

<i>Счетчик команд</i>	<i>Машинный код</i>	<i>Мнемоническое обозначение</i>
1000	01 17 02	MOV @R7, R2
1002	„следующая команда”	

запишет копию команды, находящейся в ячейке 1002, в регистр R2. Затем будет выполнена команда, содержащаяся по адресу 1002.

Применение кода адресации 17₈ по адресу приемника нецелесообразно, так как приведет к модификации (порче) следующей команды. Например, пусть содержимое регистра R2 равно нулю и исполняются две команды:

<i>Счетчик команд</i>	<i>Машинный код</i>	<i>Мнемоническое обозначение</i>
1000	01 02 17	MOV R2, @R7
1002	00 52 01	INCR1

После выполнения команды по адресу 1000 команда INC R1, находящаяся по адресу 1002, будет заменена командой HALT (останов):

<i>Счетчик команд</i>	<i>Машинный код</i>	<i>Мнемоническое обозначение</i>
1002	00 00 00	HALT

Практика программирования не рекомендует писать программы, вызывающие модификацию исходных команд, так как при этом существенно затрудняются отладка программ и поиск ошибок. Способы адресации с кодами 47₈ и 57₈ применять не рекомендуется, так как они приводят к заиклииванию. Действительно, команды

<i>Счетчик команд</i>	<i>Машинный код</i>	<i>Мнемоническое обозначение</i>
1000	01 47 00	MOV -(R7), R0
2000	01 57 01	MOV @-(R7), R1

при выборке операнда по адресу источника уменьшают содержимое счетчика команд на 2 и таким образом возвращают управление той же самой команде.

Таким образом, поскольку способы адресации операнда-источника и операнда-приемника информации могут быть заданы независимо друг от друга, каждый код операции может обеспечить большое число вариантов одной и той же команды. Благодаря исключительной гибкости системы адресации возможности рассматриваемой микро-ЭВМ чрезвычайно широки, в том числе и при автоматизации контроля.

Система команд. Как уже отмечалось, команды могут быть трех типов: безадресные, одноадресные и двухадресные. В безадресной команде все слово занимает код операции. Одноадресная команда имеет два поля: для кода операции отводится 10 разрядов и для адреса операнда-приемника — 6 разрядов. Двухадресная команда содержит три поля: 4 разряда отводится для кода операции, 6 разрядов — для адреса операнда-источника и 6 разрядов — для адреса операнда-приемника.

В зависимости от способа адресации одноадресная команда может иметь одно продолжение, т. е. занимать два последовательных слова. Во втором слове может находиться (в зависимости от способа адресации) либо непосредственный операнд, либо абсолютный адрес, либо индекс (смещение). Соответственно двухадресная команда может иметь одно или два продолжения. Таким образом, в зависимости от типа команды и способа адресации каждая команда может занимать от одной до трех последовательных ячеек памяти:



Рассмотрим состав команд и отметим их некоторые характерные особенности.

Примечание. С детальным описанием каждой команды при необходимости можно ознакомиться по руководству к микроЭВМ.

При описании команд приняты следующие обозначения: DD — адрес приемника; SS — адрес источника; R — регистр общего назначения; * — звездочка: перед кодом операции принимает значение 0 для слов и 1 — для байтов. Буква В в круглых скобках в мнемонических обозначениях команды показывает, что команда может быть использована как для слов, так и для байтов.

В табл. 2.1 приведены соответственно группы одноадресных команд, двухадресных команд и команд управления программой.

Рассмотрим особенности выполнения *команд условного перехода*. Четыре разряда регистра состояния N, Z, V, C процессора представляют собой признаки результата выполнения предыдущей команды

Таблица 2.1

Мнемоника	Команда	Машинный код или базовый код
1	2	3
Одноадресные		
<i>Основные</i>		
CLR (B)	Очистка	*050DD
COM (B)	Вычисление обратного кода	*051DD
INC (B)	Приращение положительное	*052DD
DEC (B)	Приращение отрицательное	*053DD
NEG (B)	Вычисление дополнительного кода	*054DD
TST (B)	Проверка	*057DD
<i>Сдвиг и циклический сдвиг</i>		
ASR (B)	Арифметический сдвиг вправо на 1 бит	*062DD
ASL (B)	Арифметический сдвиг влево на 1 бит	*063DD
ROR (B)	Циклический сдвиг вправо на 1 бит	*060DD
ROL (B)	Циклический сдвиг влево на 1 бит	*061DD
SWAB	Обмен байтами	0003DD
<i>Увеличенной точности</i>		
ADC (B)	Прибавить перенос	*055DD
SBC (B)	Вычесть перенос	*056DD
<i>Прочие</i>		
SXT	Расширение знака	0067DD
MFPS	Передача байта из регистра состояния процессора	1067DD
MTPS	Передача байта в регистр состояния процессора	1064DD
Двухадресные		
<i>Основные</i>		
MOV (B)	Пересылка	*1SSDD
CMP (B)	Сравнение	*2SSDD
ADD	Сложение	06SSDD
SUB	Вычитание	16SSDD
<i>Логические</i>		
BIT (B)	Проверка битами	*3SSDD
BIC (B)	Очистка битами	*4SSDD
BIS (B)	Установка битами	*5SSDD
XOR	Исключающее ИЛИ	074RDD
Управление программой		
<i>Ветвление</i>		
BR	Переход	000400
BNE	Переход, если не равно (нулю)	001000
BEQ	Переход, если равно (нулю)	001400
BPL	Переход, если плюс	100000
BMI	Переход, если минус	100400
BVC	Переход, если нет переполнения	102000
BVS	Переход, если есть переполнение	102400
BCC	Переход, если нет переноса	103000
BCS	Переход, если есть перенос	103400

1	2	3
	<i>Ветвление по знаку</i>	
BGE	Переход, если больше или равно (нулю)	002000
BLT	Переход, если меньше (нуля)	002400
BGT	Переход, если больше (нуля)	003000
BLE	Переход, если меньше или равно (нулю)	003400
	<i>Беззнаковое ветвление</i>	
BHI	Переход, если больше	101000
BLOS	Переход, если меньше или равно	101400
BHIS	Переход, если больше или равно	103000
BLO	Переход, если меньше	103400
	<i>Переходы и подпрограммы</i>	
JMP	Переход	0001DD
JSR	Переход к подпрограмме	004RDD
RTS	Возврат из подпрограммы	00020R
MARK	Очистка стека	006400
SOB	Вычитание единицы и переход	077R00
	<i>Ловушки и прерывания</i>	
EMT	Ловушка при эмуляции	104000—104377
TRAP	Ловушка	104400—104777
BPT	Ловушка отладки	000003
IOT	Ловушка ввода-вывода	000004
RTI	Возврат из прерывания	000002
RTT	Возврат из прерывания	000006
	<i>Установка кодов условий</i>	
CLC, CLV, CLZ, CLN, CCC SEC, SEV, SEZ, SEN, SCC	Очистка	000240
	Установка	000260
	<i>Специальные</i>	
HALT	Остановка	000000
WAIT	Ожидание прерывания	000001
RESET	Общий сброс	000005
NOP	Холостая команда	000240

и используются командами условного перехода. Передача управления происходит при выполнении условия, определяемого кодом операции.

Характерная особенность команд условного перехода — передача управления относительно счетчиков команд, что обеспечивает свойство позиционной независимости. Команды условных переходов имеют следующий формат:

КОП	Смещение
-----	----------

Младший байт команды рассматривается как знаковое смещение (в диапазоне от -128 до $+127$ слов) относительно счетчика команд. При написании программ следует учитывать ограниченную возможность передачи управления вперед и назад и в случае необходимости применять команду передачи управления JMP. Команда JMP не имеет ограничений на длину передачи, однако занимает в памяти два слова.

Адрес команды, на которую передается управление в случае выполнения условия, вычисляется по формуле

$$A = \alpha + 2 + 2 \times \text{смещение},$$

где α — адрес команды условного перехода.

При передаче управления назад смещение задается в дополнительном коде.

В рассматриваемой системе команд эффективно решен вопрос вызова *подпрограмм* и возвращения в вызывавшую программу. Вызов подпрограммы осуществляется с помощью команды вида

JSR R_i , SUB,

где R_i — один из регистров общего назначения; SUB — имя некоторой подпрограммы.

Действие команды обращения к подпрограмме можно условно представить в виде эквивалентной системы команд:

$$\text{JSR } R_i, \text{ SUB} \equiv \begin{cases} \text{MOV } R_i, -(\text{SP}); & \text{запоминание содержимого } R_i \\ & \text{в стеке} \\ \text{MOV PC, } R_i & \text{запоминание адреса возврата,} \\ \text{JMP SUB} & \text{переход на подпрограмму} \end{cases}$$

При обращении к подпрограмме прежнее содержимое регистра связи R_i запоминается в стеке, затем в регистр связи заносится адрес возврата (адрес команды, следующей за командой вызова) и происходит переход на подпрограмму с именем SUB.

Возврат из подпрограммы происходит по команде RTS R_i , действие которой можно также условно представить в виде системы эквивалентных команд:

$$\text{RTS } R_i \equiv \begin{cases} \text{MOV } R_i, \text{ PC} & \text{, передача управления в вызывающую} \\ & \text{; программу} \\ \text{MOV } (\text{SP}) +, R_i & \text{; восстановление прежнего содержимого} \\ & \text{; регистра связи} \end{cases}$$

По команде RTS R_i содержимое регистра связи загружается в счетчик команд PC (этим самым обеспечивается возврат в вызывавшую программу) и затем прежнее содержимое регистра связи восстанавливается путем извлечения его из стека.

В качестве регистра связи возможно использование и счетчика команд. В этом случае экономится один регистр общего назначения (его можно применять для других целей).

В виде эквивалентной системы команд обращение и возврат из подпрограммы будут выглядеть следующим образом:

$$\text{JSR, PC, SUB} \equiv \begin{cases} \text{MOV PC, — (SP)} & ; \text{запоминание адреса возврата} \\ & ; \text{в стеке} \\ \text{JMP SUB} & ; \text{переход к подпрограмме} \end{cases}$$

$$\text{RTS PC} \equiv \begin{cases} \text{MOV (SP) +, PC} & ; \text{извлечение адреса возврата} \\ & ; \text{из стека и возврат в вызва-} \\ & ; \text{вшую программу} \end{cases}$$

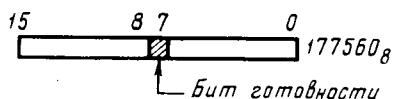
Такой способ связи с подпрограммами практически не накладывает ограничений на глубину вложенности подпрограмм (она зависит лишь от числа ячеек памяти, отведенных под стек) и способы обращений. Не существует ограничений и для обращения подпрограммы самой к себе — *рекурсивное обращение*. Использование рекурсивных обращений позволяет создавать компактные и эффективные программы обработки данных.

Обмен данными с внешними устройствами. Обмен данными между процессором и внешними устройствами осуществляется с помощью стандартных циклов магистрали. Каждое внешнее устройство должно иметь как минимум два регистра: регистр команд и состояний и буферный регистр данных. За каждым регистром закрепляется один из четных адресов на общей магистрали, принадлежащих области последних 4К слов адресного пространства.

Обмен данными рассмотрим на примере взаимодействия процессора и дисплея, являющегося стандартным системным устройством. Дисплей рассматривается как прибор, состоящий из двух совершенно независимых устройств (клавиатуры и устройства отображения на экране). Каждое устройство имеет свои два регистра.

Рассмотрим *программный способ передачи данных*.

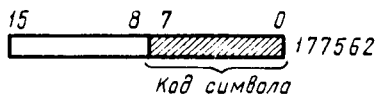
Формат регистра команд и состояний клавиатуры имеет вид



Седьмой бит регистра команд и состояний, называемый *битом готовности*, имеет следующий смысл: БИТ [7] = 1 показывает, что была нажата клавиша, БИТ [7] = 0 показывает, что клавиша не нажималась.

В случае нажатия одной из клавиш клавиатуры дисплея бит готовности устанавливается в 1 и код символа, соответствующий нажатой клавише, пересылается в буферный регистр ввода клавиатуры.

Формат буферного регистра ввода клавиатуры имеет вид



Справа от регистров указаны их восьмеричные адреса на общей магистрали.

На языке ассемблера процесс ввода одного символа с клавиатуры дисплея в регистр R0 описывается следующим образом:

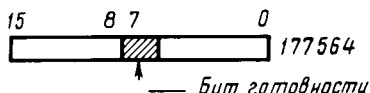
```

      TKS = 177560      ; регистр команд и состояний клавиатуры
      TKB = 177562      ; буферный регистр клавиатуры
LOOP.  TSTB TKS          ; опрос бита готовности
      BPL LOOP:         ; если готовности нет, то вернуться на
                        ; команду с меткой LOOP
      MOVB TKB, R0      ; передать символ из буферного регистра
                        ; клавиатуры в регистр R

```

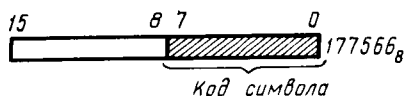
Аналогично происходит и процесс вывода символа на экран.

Формат регистра команд и состояний устройства отображения на экране имеет вид



Бит готовности принимает значение 1 в том случае, когда устройство готово принять символ для вывода на экран. В процессе вывода символа на экран бит готовности сбрасывается устройством в 0 и по окончании вывода снова устанавливается им в 1.

Формат буферного регистра вывода на экран имеет вид



Если необходимо вывести на экран дисплея младший байт регистра R0, то программа на языке ассемблера будет выглядеть следующим образом:

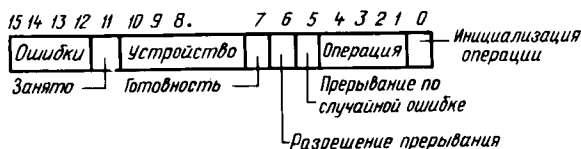
```

      TPS = 177564      ; регистр команд и состояний экрана
      TPB = 177566      ; буферный регистр данных экрана
OUT:   TSTB TPS          ; опрос готовности
      BPL OUT           ; если не готово, вернуться к опросу
      MOVB R0, TPB      ; вывод символа на экран

```

Форматы регистров данных нестандартной аппаратуры в основном определяются разработчиком. Для регистров команд и состояний рекомендуется общий формат, которым следует руководствоваться при разработке аппаратуры.

Общий формат регистра команд и состояний таков:



Разряды 15—12 используются для сигнализации о наличии ошибки (разряд 15) и о характере ошибки (14—12). Они устанавливаются внешним устройством. Разряд 11 сигнализирует о том, что внешне

устройство занято выполнением внутренних операций и не может принять участия в обмене данными. Разряды 10—8 применяются для выбора устройств, подключенных к общему контроллеру. Разряды 7—5 являются разрядами условий и служат для программного определения состояний устройства и управления прерыванием. Для определения функции, выполняемой внешним устройством, служат разряды 4—1. Разряд 0 служит для инициирования (запуска) операции.

Рассмотрим программу вывода одного символа на строочное печатающее устройство с анализом бита ошибки:

	LPS = 177514	; регистр команд и состояний
		; строочного печатающего устройства
	LPB = 177516	; буферный регистр этого устройства
PRINT:	BIT $\neq$ 100200, LPS	; опрос на ошибку и готовность
	BEQ PRINT	; вернуться к опросу, если нет ошибки
		; и готовности
	BMI ERROR	; перейти на обработку ошибки, если
		; БИТ [15] = 1
	MOVB R0, LPB	; вывести символ на печать
	.	
	.	
ERROR:	.	, начало программы обработки ошибки

Система прерывания. Прерывание — непредусмотренное обращение к подпрограмме. МикроЭВМ имеет два вида прерываний: внутренние и внешние. *Внутренние прерывания* происходят по инициативе процессора при возникновении в нем необычных ситуаций, а *внешние прерывания* — по требованию устройств ввода-вывода, подсоединенных к общей магистрали.

Внешние прерывания. Разрешением запроса на внешнее прерывание от конкретного устройства управляет БИТ [6] регистра команд и состояний этого устройства.

Сигнал запроса на прерывание формируется в двух случаях: 1) бит разрешения прерывания установлен (БИТ [6] = 1), а бит готовности (БИТ [7]) переходит из состояния 0 в состояние 1; 2) бит готовности установлен (БИТ [7] = 1), а бит разрешения прерывания (БИТ [6]) переходит из состояния 0 в состояние 1.

Установка и сброс бита разрешения прерывания осуществляются программным способом. При начальном пуске машины, а также при выполнении команды RESET вырабатывается на общей магистрали импульс INIT, который сбрасывает в нуль биты разрешения прерывания в регистрах команд и состояний всех устройств, подсоединенных к общей магистрали. Запросы на прерывание процессора от внешних устройств могут быть удовлетворены только в том случае, если седьмой бит регистра состояния процессора равен нулю (PSR [7] = 0). В противном случае (PSR [7] = 1) все запросы на прерывание игнорируются (замаскированы).

Линия разрешения прерывания от процессора проходит последовательно через все внешние устройства. Таким образом, приоритет устройств понижается по мере удаления от процессора. Любое устройство, которое может прервать процессор, может также пре-

рвать выполнение программы обслуживания устройства с более низким приоритетом, если PSR [7] = 0. При такой структуре возможно вложение прерываний любого уровня (число вложений ограничивается лишь числом ячеек памяти, отведенных под стек).

Каждому внешнему устройству, подсоединенному к общей магистрали, ставится в соответствие своя пара ячеек оперативной памяти, называемая *вектором прерывания*. Первое слово вектора прерывания содержит адрес начала подпрограммы обработки прерывания для данного устройства, а второе слово — содержимое регистра состояния процессора для этой подпрограммы. Обычно второе слово в этом случае хранит информацию о значении седьмого бита — бита разрешения прерывания. Если этот бит равен единице, то прерывание от устройства с более высоким приоритетом будет запрещено на время работы программы, обработки прерывания. Каждое внешнее устройство, способное работать по прерыванию, имеет свой «запаянный» вектор, числовое значение которого соответствует адресу первого слова вектора прерывания в оперативной памяти. В случае удовлетворения запроса на прерывание устройство посылает свой вектор прерывания в процессор, тем самым сообщая процессору адрес начала подпрограммы обслуживания прерывания для этого устройства и состояние процессора.

Действия, которые выполняет процессор после получения вектора прерывания, можно описать с помощью эквивалентной системы команд:

```
MOV PSR,— (SP)    ; запомнить в стеке текущее состояние
                  ; процессора
MOV PC,— (SP)     ; запомнить в стеке адрес возврата
MOV α, PC         ; перейти на подпрограмму обработки пре-
                  ; рывания
MOV α+2, PSR      ; записать новое значение регистра состоя-
                  ; ния процессора
```

где α — адрес вектора прерывания.

Возврат из прерывания осуществляется с помощью команды RTI, действие которой можно представить в виде системы команд:

$$RTI = \begin{cases} \text{MOV (SP) +, PC} & ; \text{извлечь из стека адрес возврата} \\ & ; \text{и загрузить его в счетчик команд} \\ \text{MOV (SP) +, PSR} & ; \text{извлечь из стека старое значение} \\ & ; \text{регистра состояния процессора и загрузить его в PSR} \end{cases}$$

Схема, приведенная на рис. 2.6, поясняет принцип работы микроЭВМ по прерыванию.

Адреса векторов внешних прерываний и адреса регистров основных системных устройств приведены в табл. 2.2.

В качестве примера использования системы прерывания приведем программу, обеспечивающую работу системного терминала в режиме электрифицированной пишущей машинки во время выполнения некоторой фоновой программы:

```
TKS = 177560      ; регистр команд и состояний клавиатуры
                  ; ры
TKB = 177562      ; буферный регистр клавиатуры
TPS = 177564      ; регистр команд и состояний печати
```

```

TPB = 177566      ; буферный регистр печати
SP = % 6          ; указатель стека
.= 60             ; установка счетчика команд
INTHND            ; вектор прерывания клавиатуры
200              ; слово состояния процессора (преры-
                ; вания запрещены)
                ; установка счетчика команд
START:  MOV #., SP ; установка указателя стека (SP = 500)
        BIS #100, TKS ; установка бита разрешения прерыва-
                ; ния от клавиатуры
                ; фоновая программа
        :
INTHND:  TSTB TPS   ; начало подпрограммы обработки пре-
                ; рывания
        BPL INTHND ; опрос готовности печати, вернуться
                ; к опросу, если не готово
        MOV B TKB, TPB ; вывод символа на печать
        RTI          ; возврат из прерывания
        .END START   ; конец программы

```

Всякий раз при нажатии клавиши выполнение фоновой программы приостанавливается и происходит передача управления по вектору прерывания, расположенному в ячейке памяти с адресом 60.

После печати символа по команде RTI происходит возврат к прерванному месту фоновой программы и продолжение ее выполнения.

Внутренние прерывания. Возникают в случае возникновения необычных ситуаций в процессоре (ошибка магистрали, резервная или запрещенная команда, установлен бит слежения T в регистре состояния процессора, сбой питания) или в случае появления в программе команд IOT, EMT, TRAP. Механизм внутренних прерываний аналогичен механизму внешних прерываний. Например, выход на прерывания или частоты питающей сети за допустимые пределы вызывает внутреннее прерывание, действие которого можно описать с помощью системы команд:

```

MOV PSR,— (SP) ; запомнить в стеке состояния процессора
MOV PS,— (SP)  ; запомнить в стеке адрес возврата
MOV 24, PC     ; перейти на подпрограмму обработки
MOV 26, PSR    ; записать новое состояние процессора

```

Причины, вызывающие прерывания и адреса векторов внутренних прерываний, приведены ниже:

Ошибка магистрали	04
Резервная или запрещенная команда	10
Прерывание по T-разряду	14
Командное прерывание для ввода-вывода	20
Сбой питания	24
Командное прерывание для системных программ	30
Командное прерывание для программ пользователя	34

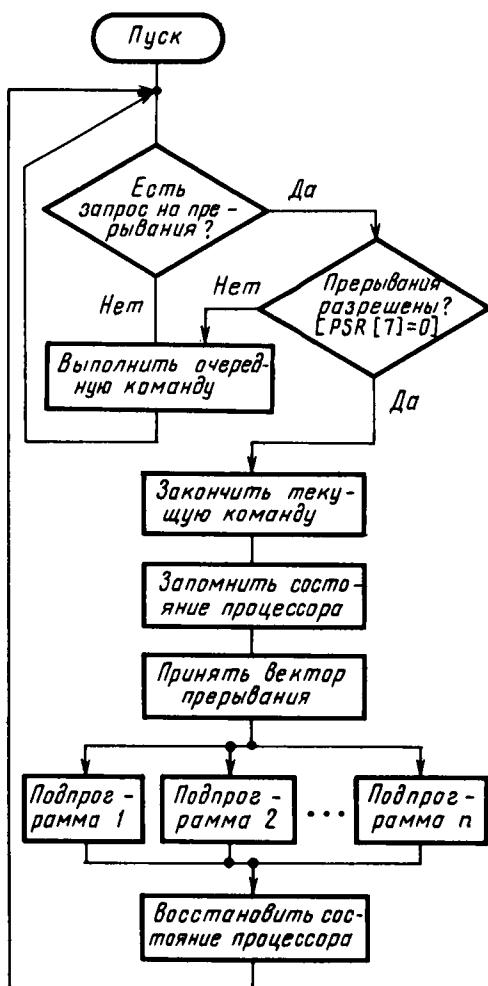


Рис. 2.6. Схема системы прерывания микроЭВМ.

Т а б л и ц а 2.2

Устройство	Адрес вектора	Адреса регистров
Устройство терминала	60	177560; 177562
Экран терминала	64	177564; 177566
Считыватель с перфоленты	70	177550; 177552
Перфоратор ленточный	74	177554; 177556
Построчная печать	200	177514; 177516

Самый простой случай использования внутренних прерываний — организация ловушек останова. Например, чтобы организовать останов процессора при возникновении ошибки магистрали или появлении в программе резервной или запрещенной команды, достаточно включить в программу строки

```

      . = 4
      . = . + 2
      HALT
      . = . + 2
      HALT

```

Результатом трансляции этих строк будет следующая информация в ячейках памяти:

<i>Счетчик команд</i>	<i>Машинный код</i>
000004	000006
000006	000000
000010	000012
000012	000000

Как видно, каждая подпрограмма обработки прерывания состоит из единственной команды останова HALT.

ЕМТ- и TRAP-команды используются в качестве эффективного средства организации и вызова подпрограмм. Любая из 256_{10} ЕМТ-команд с машинными кодами 104000_8 — 104377_8 вызывает внутреннее прерывание по вектору, расположенному по адресу 30_8 . Соответствующим образом написанная подпрограмма обработки прерывания (подпрограмма-диспетчер) обеспечивает вызов одной из 256_{10} подпрограмм согласно коду младшего байта ЕМТ-команды.

Обычный вызов подпрограммы с помощью команды JSR R_i , SUB требует двух ячеек памяти, вызов же подпрограммы с помощью ЕМТ-команды занимает одну ячейку памяти. В сложных программах, построенных по модульному принципу, вызовы подпрограмм занимают значительное место, поэтому применение ЕМТ- и TRAP-команд позволяет получить существенную экономию ячеек памяти.

Сетевой таймер (часы реального времени). Для организации измерения временных интервалов в микроЭВМ реализовано внутреннее прерывание по таймеру с вектором, находящимся по адресу 100_8 . Если таймер включен, то на линию прерывания по таймеру поступают импульсы с частотой питающей сети 50 Гц или с периодом 0,02 с. Подпрограмма обслуживания прерывания по таймеру обеспечивает накопление и подсчет импульсов («тиков») в некоторой выбранной ячейке памяти. Для обеспечения достаточной емкости таймера вместо одной ячейки памяти применяют две. Для реализации арифметической операции с удвоенной разрядностью используется команда ADC — прибавление переноса.

Программа обслуживания таймера на языке ассемблера может быть написана так:

$\cdot = 100$	; вектор прерывания таймера
INTCLK	; слово состояния процессора
200	; программа обслуживания прерывания по таймеру
INTCLK: ADD #1, CLK	; прибавить единицу
ADC CLK +2	; прибавить перенос
RTI	; возврат из прерывания
CLK: . WORD 0, 0	; рабочие ячейки таймера

Емкость таймера в этом случае составит более 20 000 ч. Действительно, максимальное число, которое можно записать в 32 двоичных разрядах, $N = 2^{32} - 1 = 4\,294\,967\,295$. Отсюда максимальная емкость таймера $T = 4\,294\,967\,295 / (50 \cdot 3600) = 23\,859$ ч.

2.2. Основные требования, предъявляемые к микропроцессорным информационно-управляющим вычислительным системам

Задачи, решаемые МП ИУВС. По назначению информационно-управляющие системы делят на ИУВС для управления технологическими процессами и объектами и ИУВС для управления отраслями производства, предприятиями, городским хозяйством и т. п.

Ниже будут рассмотрены ИУВС, предназначенные для управления технологическими процессами и объектами. Их отличительные признаки: а) наличие ограниченного набора четко сформулированных задач; б) требования оптимизации структуры ИУВС для конкретного применения; в) работа в реальном масштабе времени, т. е. обеспечение минимального времени реакции на изменение внешних условий; г) наличие развитой системы внешних устройств, их большое разнообразие; д) существенное различие функциональных задач; е) высокие требования по надежности с учетом большой продолжительности непрерывной работы; ж) сложные условия эксплуатации; з) обеспечение автоматического режима работы или режима с участием оператора как элемента системы.

В общем виде структура микропроцессорной ИУВС (рис. 2.7, где ОУ — объект управления) содержит датчики входной информации ДВИ; устройство сопряжения с объектом УСО; микроЭВМ; пульт

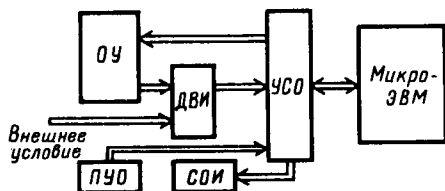


Рис. 2.7. Структура МП ИУВС.

управления оператора ПУО; систему отображения информации СОИ; устройства передачи информации между элементами системы.

Ядром ИУВС, реализующим процесс переработки информации и формирования воздействий на объект управления ОУ, является микроЭВМ. Применение микроЭВМ в ИУВС способствует повышению надежности; снижению стоимости системы; сокращению сроков раз-

работки аппаратных средств и программного обеспечения; быстрой адаптации к изменениям требований заказчика в процессе разработки системы и в период ее модернизации; более простому осуществлению функций распределенного, децентрализованного управления; обеспечению модульного принципа построения системы.

С учетом приведенной структуры задачи, решаемые ИУВС, классифицируются следующим образом: 1) ввод и предварительная обработка входной информации в УСО; 2) обработка входной информации в микроЭВМ; 3) решение основных функциональных задач и выработка управляющих воздействий в микроЭВМ; 4) диагностирование ИУВС; 5) вывод информации.

Ввод и предварительная обработка информации в устройстве сопряжения с объектом. При решении задач этого класса основными целями обработки входной информации в УСО являются предварительная фильтрация входных сигналов и приведение входной информации к стандартной для микроЭВМ ИУВС форме.

Преобразование входной информации и сигналов, заданных в виде напряжений, токов, углов поворотов валов и т. д., осуществляется в большинстве случаев стандартными средствами [10].

Обработка входной информации в микроЭВМ. Этот класс задач, решаемых ИУВС, представляет собой этап обработки информации, связанный с контролем правильности приема входной информации; сглаживанием (фильтрацией) входной информации для выявления полезного сигнала на фоне помех.

Для сглаживания входная информация, описывающая состояние объекта управления, поступает в микроЭВМ в виде дискретных или решетчатых функций. *Дискретная функция* — функция $f(t)$, получившаяся в результате выборки в моменты времени $t = nT_0$ (где T_0 — шаг квантования по времени; n — натуральный ряд чисел) данных с датчика входной информации. Таким образом, в микроЭВМ ИУВС должны быть реализованы алгоритмы сглаживания дискретных функций, искаженных помехами, которые обусловлены случайными ошибками измерения. Реализация алгоритмов сглаживания сводится к нахождению такой функции $F(t)$, которая дает наилучшее приближение к заданной. Вид функции $F(t)$ зависит от характера изменения во времени параметров состояния объекта управления.

Для широкого класса типов объектов управления в качестве сглаживающей функции выбирается полином s -й степени:

$$F(\theta, t) = \tilde{\theta}_0 + \tilde{\theta}_1 t + \tilde{\theta}_2 t^2 + \dots + \tilde{\theta}_s t^s = \sum_{i=0}^s \tilde{\theta}_i t^i,$$

где $\tilde{\theta}_i$ — случайные коэффициенты.

Если $x_i = x(t_j)$ — полученное из УСО значение параметра в дискретный момент времени t_j ; $\sum_{i=0}^s \theta_i^* t^i$ — сглаживающая функция (θ_i^* — математическое ожидание величины $\tilde{\theta}_i$), то использование *среднеквадратичного критерия* приводит к решению следующей системы уравнений:

[illegible]

где N — количество полученных из УСО значений $\tilde{x}_j$ (выборка).

В частности, при выборе в качестве $F(\theta^*, t)$ полинома первой степени $F(\theta^*, t) = \theta_0^* + \theta_1^* t$ коэффициенты определяются следующим образом:

$$\mathfrak{d}_0^* = \sum_{j=1}^N P_{\mathfrak{d}_0}(j) \tilde{x}_j, \quad \mathfrak{d}_1^* = \sum_{j=1}^N P_{\mathfrak{d}_1}(j) \tilde{x}_j,$$

$$P_{\theta 0}(j) = -\frac{-6j-4N-4}{N(N+1)}; \quad P_{\theta 1}(j) = \frac{1}{T_0} \frac{12j-6N-6}{N(N^2-1)} -$$

функції веса при сглаживанні параметра.

Критерий среднеквадратичного приближения не является единственно возможным. Широко используется сглаживание входной информации *сплайнами* [11]. При этом минимизируется функционал:

$$\int_a^b |f''(t)|^2 dt + \sum_i (1/\varepsilon_i)(f_i - \tilde{x}_i)^2,$$

где ε_i — параметр сглаживания; $\tilde{x}_i$ — полученные из УСО параметры состояния объекта; $f(t)$ — сглаживающая функция.

В частности, при выборе в качестве сглаживающей функции полиномов третьей степени

$$Q_i(t) = \sum_{\lambda=0}^3 a_{\lambda}^{(i)}(t-t_i)^{\lambda}, \quad i=0, 1, 2, \dots, N,$$

определение коэффициентов $\alpha^{(i)}$ сводится к решению следующей матричной системы уравнений:

$$KA = D,$$

где

$$A = \begin{bmatrix} a_2^{(0)} \\ a_2^{(1)} \\ \dots \\ a_2^{(N)} \end{bmatrix} \text{ — матрица неизвестных коэффициентов } a_2,$$

$$\mathbf{D} = \begin{bmatrix} d_0 \\ d_1 \\ \dots \\ d_N \end{bmatrix} \text{ — матрица свободных членов } d_i, \text{ причем}$$

$$d_0 = (3/T_0^3) [\tilde{x}_1 - \tilde{x}_0 - x'_0 T_0], \quad d_N = (3/T_0^3) [\tilde{x}_{N-1} - \tilde{x}_N + \tilde{x}'_N T_0],$$

$$d_i = 6\tilde{x}(t_{i-1}, t_i, t_{i+1}), i=1, 2, \dots, N-1;$$

здесь $\tilde{x}(t_{i-1}, t_i, t_{i+1}) = (\tilde{x}_{i-1} - 2\tilde{x}_i + \tilde{x}_{i+1}) / (2T_0^2)$; x'_0, x'_N — заданные граничные условия; K — матрица коэффициентов.

Матрица коэффициентов K является пятидиагональной и для заданного типа граничных условий имеет вид

$$K = \begin{bmatrix} (2k_2 + 10k_3) & (k_2 + k_3) & k_3 & 0 & 0 & 0 & \dots & 0 \\ (k_2 + k_3) & k_1 & k_2 & k_3 & 0 & 0 & \dots & 0 \\ k_3 & k_2 & k_1 & k_2 & k_3 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & k_3 & k_2 & k_1 & k_2 & k_3 \\ 0 & \dots & 0 & 0 & k_3 & k_2 & k_1 & (k_2 + k_3) \\ 0 & \dots & 0 & 0 & 0 & k_3 & (k_2 + k_3) & (2k_2 + 10k_3) \end{bmatrix}$$

где $k_1 = 4(1 + 9\varepsilon/T_0^3)$; $k_2 = 1 - 24\varepsilon/T_0^3$; $k_3 = 6\varepsilon/T_0^3$.

При известных характеристиках датчиков входной информации и параметрах УСО можно заранее, на этапе проектирования ИУВС, определить T_0 и ε , а соответственно и элементы обратной матрицы $R = K^{-1}$, такой, что

$$R = \begin{bmatrix} r_{00} & r_{01} & r_{02} & \dots & r_{0N} \\ r_{10} & r_{11} & r_{12} & \dots & r_{1N} \\ \dots & \dots & \dots & \dots & \dots \\ r_{N0} & r_{N1} & r_{N2} & \dots & r_{NN} \end{bmatrix}$$

Значения коэффициентов сглаживающей функции определяются как $a_0^{(i)} = \sum_{j=0}^N P^{(i)}(j) \tilde{x}_j$, где для выбранных граничных условий весовая

функция определяется следующими ординатами:

для $i=1, 2, \dots, N-1$

$$P^{(i)}(j) = \delta_{ij} - 6(\varepsilon/T_0^3)(\Delta^2 r_{i,j-1} - 2\Delta^2 r_{i,j} + \Delta^2 r_{i,j+1} + 1), \quad j=1, 2, \dots, N-1;$$

$$P^{(i)}(0) = -6(\varepsilon/T_0^3)(\Delta^2 r_{i,1} - \Delta^2 r_{i,0});$$

$$P^{(i)}(N) = -6(\varepsilon/T_0^3)(\Delta^2 r_{i,N-1} - \Delta^2 r_{i,N}),$$

где $\Delta^2 r_{i,j} = (r_{i-1,j} - 2r_{i,j} + r_{i+1,j})$;

для $i=0$

$$P^{(0)}(j) = -6(\varepsilon/T_0^3)(\Delta r_{1,j-1} - 2\Delta r_{1,j} + \Delta r_{1,j+1}), \quad j=1, 2, \dots, N-1;$$

$$P^{(0)}(0) = 1 - 6(\varepsilon/T_0^3)(\Delta r_{1,1} - \Delta r_{1,0});$$

$$P^{(0)}(N) = -6(\varepsilon/T_0^3)(\Delta r_{1,N-1} - \Delta r_{1,N}),$$

где $\Delta r_{i,j} = r_{1,j} - r_{0,j}$;

для $i=N$

$$P^{(N)}(j) = 6(\varepsilon/T_0^3)(\Delta r_{N,j-1} - 2\Delta r_{N,j} + \Delta r_{N,j+1}), \quad j = 1, 2, \dots, N-1;$$

$$P^{(N)}(0) = 6(\varepsilon/T_0^3)(\Delta r_{N,1} - \Delta r_{N,0});$$

$$P^{(N)}(N) = 1 + 6(\varepsilon/T_0^3)(\Delta r_{N,N-1} - \Delta r_{N,N}),$$

где $\Delta r_{N,j} = r_{N,j} - r_{N-1,j}$.

Аналогично могут быть определены и коэффициенты $a_1^{(j)}$ и $a_2^{(j)}$.

Значение количества фиксированных выборок N связано с динамикой изменения состояния объекта управления. При большом N увеличивается наблюдательное время $T_{\text{н}} = NT_0$, при малом N увеличивается ошибка сглаживания. Таким образом, при выборе алгоритма сглаживания ищется компромисс между допустимой ошибкой сглаживания и требуемой реакцией ИУВС на изменение состояния объекта.

Решение основных функциональных задач и выработка управляющих воздействий в микроЭВМ. Этот класс задач, решаемых ИУВС, наиболее объемный. Он включает в себя задачи: распределения ресурсов ИУВС, выбора источников и потребителей информации; прогнозирования изменения состояния объекта управления по его характеристикам и измеренным параметрам и выработки управляющих воздействий; информационные и взаимодействия с оператором; автоматического слежения и адаптации системы к изменяющимся условиям.

Рассмотрим наиболее характерные типы функциональных задач.

Тип I. К этому типу задач относится задача преобразования координат системы измерений в систему координат исполнительных элементов [10], так как в силу конструктивных и технологических ограничений пространственное положение датчиков входной информации и исполнительных элементов, оказывающих непосредственное воздействие на объект управления, различно.

Необходимость в пересчете, преобразовании входных данных вызывается также невозможностью непосредственного измерения какого-либо параметра γ . В этом случае значение γ можно оценить косвенно по измеряемым значениям $\xi_1, \xi_2, \dots, \xi_r$, при условии существования возможности однозначного выражения в виде $\gamma = f(\xi_1, \xi_2, \dots, \xi_r)$. Например, для определения дозировки топлива с целью получения максимальной мощности при минимальных расходах топлива в двигателях внутреннего сгорания необходимо знать количество воздуха, всасываемого в каждый цилиндр за один ход поршня. Количество поступающего в цилиндры воздуха $V = f(n, P)$, где n — частота вращения коленчатого вала; P — давление во впускном трубопроводе.

Тип II. Этот тип задач наиболее представительен и включает в себя задачи прогнозирования, функционального преобразования, логического управления. Приведенное разделение является условным, но охватывает значительный объем встречающихся на практике задач.

Характерными задачами прогнозирования являются задачи управления технологическими процессами. При этом весьма часто решается система уравнений

$$\begin{cases} z_k^{(i)} = z_0^{(i)} + a_1^{(i)} \dot{z}^{(i)} t + a_2^{(i)} \ddot{z}^{(i)} t^2, & i = 1, 2, \dots, n; \\ t = f(z_k^{(1)}, z_k^{(2)}, \dots, z_k^{(n)}), \end{cases}$$

где $z_0^{(i)}$ и $\dot{z}^{(i)}$ — начальное и конечное значения изменяющегося во времени i -го параметра технологического процесса; $\dot{z}^{(i)}$ и $\ddot{z}^{(i)}$ — первая и вторая производные по времени i -го параметра; $a_1^{(i)}$ и $a_2^{(i)}$ — коэффициенты; t — время протекания процесса.

Для решения систем уравнений подобного типа используют различные итерационные методы. Однако при построении специализированных ИУВС целесообразным является применение не традиционных, а применительно к данной задаче конкретных способов решения.

В ИУВС особое место занимают задачи функционального преобразования. Необходимость реализации этих задач вызывается: 1) сложностью выражений, полученных аналитическим путем, и, как следствие, трудностями при их реализации в микроЭВМ ИУВС; 2) экспериментальным характером получения функциональных зависимостей.

Как в первом, так и во втором случаях основными способами представления функциональных зависимостей являются табличный способ с интерполяцией и способ аппроксимирующих многочленов. Выбор способа интерполяции обусловлен характером функциональной зависимости $y = f(x)$ и допустимой ошибкой получения функции. Наиболее часто используются линейные и квадратичные формулы с постоянным шагом интерполяции.

В экспериментально полученных и заданных в виде таблиц или графиков функциональных зависимостях опорные значения $f(x_i)$, как правило, представлены с ошибками измерений и обработки. Поэтому нецелесообразно требовать точного совпадения приближаемой и приближающей функций в отдельных точках, а значительно лучше представить реальную функцию в виде многочлена, полученного по методу среднеквадратичного приближения.

Степень аппроксимирующего многочлена определяется на этапе проектирования ИУВС экспериментально. Другим способом получения аппроксимирующих многочленов является применение критерия равномерного приближения и методов аппроксимации сплайнами [11].

При выборе способа реализации функциональных зависимостей в микроЭВМ руководствуются следующими обстоятельствами:

методическая ошибка воспроизведения функций должна находиться в пределах допустимой для согласованного критерия оценки; необходимое число ячеек памяти для реализации и количество времени (рабочих тактов) должны быть минимальными;

избранные методы должны обеспечивать воспроизведение максимально возможного количества из всей совокупности функций, подлежащих реализации в ИУВС.

Решение большого количества задач в микроЭВМ ИУВС связано с логическим управлением состояниями объекта. Наиболее часто встречаются ИУВС, относящиеся к категории систем управления с

памятью, в которых набор выходных сигналов, вырабатываемый в некоторый отрезок времени, зависит не только от входных сигналов, поступивших в тот же момент, но и от сигналов, принятых ранее.

Иначе: совокупность управляющих сигналов, вырабатываемых ИУВС, определяется не только входными сигналами $\{x_i(t)\}$, но и внутренними состояниями автомата $\{z_j(t)\}$. Управляющие воздействия могут быть выработаны в соответствии с системой функций выходов и переходов автомата:

$$y_1(t+1) = f_1[x_1(t), x_2(t), \dots, x_n(t); z_1(t), \dots, z_k(t)];$$

$$y_2(t+1) = f_2[x_1(t), x_2(t), \dots, x_n(t); z_1(t), \dots, z_k(t)];$$

$$y_l(t+1) = f_l[x_1(t), x_2(t), \dots, x_n(t); z_1(t), \dots, z_k(t)];$$

$$z_1(t+1) = \varphi_1[x_1(t), x_2(t), \dots, x_n(t); z_1(t), \dots, z_k(t)];$$

$$z_k(t+1) = \varphi_k[x_1(t), x_2(t), \dots, x_n(t); z_1(t), \dots, z_k(t)],$$

где $x_i(t)$ — переменные, кодирующие входные сигналы; $z_j(t)$ — переменные, кодирующие состояния автомата.

Таким образом, в микроЭВМ ИУВС должна быть программно реализована адекватная модель реального объекта управления.

Существуют различные способы задания конечных автоматов, однако наиболее употребительной для реализации в ЭВМ является схема алгоритма.

Основные требования при программной реализации логического управления в микроЭВМ следующие: управление переходами объекта управления из состояния в момент времени t в следующее состояние в момент $t+1$; выработка управляющих воздействий с временем запаздывания Δt не более допустимого; управление переходами с режима R_i на режим R_j .

Задачи информационные и взаимодействия оператора с ИУВС разделяются на задачи: выработки информационных сообщений о состоянии объекта управления, ИУВС и о режиме работы; взаимодействия с оператором как элементом системы автоматического регулирования.

В первом случае вырабатываются информационные сообщения, непосредственно не влияющие на взаимодействие ИУВС с объектом управления, но дающие оператору дополнительные данные о характеристиках процесса управления.

Например, при выводе на рабочий режим диффузионной печи осуществляется автоматическое изменение температуры нагревательных элементов, однако при этом оператору могут выдаваться сообщения о времени окончания процесса, о температуре в данный момент времени.

Используемые в ИУВС алгоритмы для формирования информационных сообщений не требуют, как правило, высокой точности реализации, и допустимое время их выработки не является определяющим для системы в целом.

В отличие от информационных задач при реализации взаимодей-

ствия ИУВС с оператором как элементом системы автоматического регулирования динамика оператора в ряде случаев может быть описана в виде передаточной функции

$$W_0(s) = k_0 \frac{e^{-\tau_0 s}}{(T_0 s + 1)},$$

где k_0 , τ_0 , T_0 — соответствующие параметры оператора, полученные экспериментальным путем.

Диагностирование ИУВС. Задачи этого класса излагаются в [3,12]

Вывод информации. Задачи вывода информации включают в себя выполнение следующих основных процедур: 1) упорядочение, сортировка в микроЭВМ ИУВС информации, подлежащей передаче на объект управления; 2) кодирование информации; 3) программное формирование посылок информации в соответствии с протоколом обмена между ИУВС и объектом управления; 4) формирование сигналов по форме и мощности в соответствии с требованиями исполнительных элементов объекта управления.

Параметры входной информации. Описание функционирования ИУВС при управлении некоторым объектом на практике задается в виде: словесного описания ее функционирования; системы аналитических зависимостей, решаемых ИУВС при управлении объектом; системы дифференциальных уравнений, описывающих законы функционирования объекта управления; табличных значений, в соответствии с которыми должна функционировать ИУВС; схемы, в которой отдельные части алгоритма работы ИУВС задаются передаточными звеньями, совокупностью перечисленных форм.

Задание технических требований на ИУВС *в виде системы дифференциальных уравнений*, описывающих закономерности функционирования объекта управления, проиллюстрируем на примере [13] задания алгоритмов движения самолета:

$$\left. \begin{aligned} m\dot{V} &= P \cos \alpha + F_{vx} - X - G \sin \theta; \\ mV\dot{\theta} &= Y + P \sin \alpha + F_{vy} - G \cos \theta; \\ J_z \ddot{\theta} &= M_z, \end{aligned} \right\} \quad (2.1)$$

где P — сила тяги двигателя; X — сила лобового сопротивления; Y — подъемная сила; G — вес самолета; F_{vx} , F_{vy} — возмущающие силы, обусловленные действием ветра; J_z — момент инерции самолета относительно оси z ; θ — угол наклона траектории самолета; α — угол атаки; $\vartheta = \theta + \alpha$ — угол тангажа.

Задание технических требований на ИУВС *в виде схемы* проиллюстрируем на примере задания алгоритма управления при автоматическом движении самолета по запрограммированному курсу. На рис. 2.8 представлен пример схемы упрощенного алгоритма управления самолетом по курсу, если известны в дискретные моменты времени курсовой угол φ_3 и скорость V_3 , полученная в [13] из анализа системы (2.1).

Формы представления входной информации. Многообразие форм задания алгоритмов работы ИУВС, многообразие

задач, решаемых ИУВС, широта номенклатуры датчиков входной (первичной) информации приводят к тому, что входная информация может быть представлена в следующих формах: цифровой (параллельный n -разрядный двоичный код, последовательный двоично-десятичный код и т. д.); аналоговой (амплитуда напряжения по-

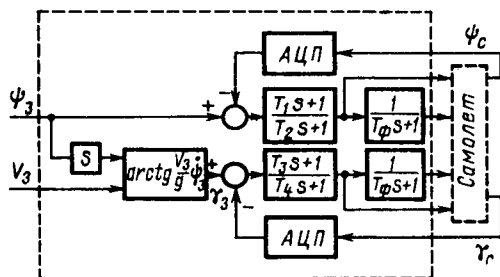


Рис. 2.8. Схема упрощенного алгоритма управления самолетом.

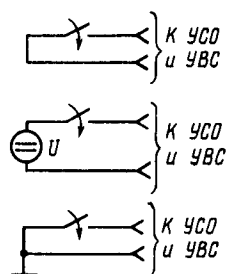


Рис. 2.9. Примеры схем датчиков релейных сигналов.

стоянного или переменного тока, фаза сигнала, значение тока и т. д.); в виде импульсных и изменяющихся скачкообразно (релейных) сигналов.

Информация в цифровой форме вводится в ИУВС, как правило, по какому-либо стандартному последовательному или параллельному интерфейсу (см. § 1.3). Поэтому вопросы подключения датчиков цифровой информации к средствам вычислительной техники решаются унифицированно. Однако разработчику ИУВС при ее проектировании необходимо уточнить следующие данные об информации, вводимой в систему в цифровой форме: 1) разрядность r_i входной информации по каждому сигналу; 2) цену младшего разряда для каждого входного сигнала ЦМР; 3) диапазон представления сигналов $x_{\max}$ и $x_{\min}$; 4) первую, а иногда и вторую производные сигналов; 5) количество достоверных разрядов $r_{\text{д}}$; 6) закон изменения шумов и его параметры; 7) скорость передачи информации; 8) территориальную удаленность датчика входной информации от ИУВС.

Информация в аналоговой форме, вводимая с датчиков и задатчиков, как правило, представляется в виде медленно меняющихся напряжений постоянного тока, модулированного по амплитуде и фазе синусоидального напряжения, импульсов напряжения или их последовательности. Информация в последних кодируется длительностью, амплитудой, площадью импульса или периодом следования импульсов.

При проектировании ИУВС в случае аналогового представления информации необходимо знать: 1) минимальное $x_{\min}$ и максимальное $x_{\max}$ значения величины, кодируемой аналоговым сигналом A_x , коэффициент пропорциональности $K_x = A_x/x$; 2) вид аналогового сигнала (медленно меняющееся напряжение, синусоидальное напряжение, импульсная последовательность); 3) полосу частот входного сигнала

ΔF_x и полосу частот аддитивных шумов $\Delta F_{ш}$, маскирующих информационный сигнал; 4) закон изменения шумов и его параметры (максимальное значение шумов $A_{ш}$, среднеквадратичное отклонение $СКО_{ш}$); 5) точность представления входной информации, характеризующейся либо абсолютной $\Delta x = x - x_a$, либо относительной $\delta_0 x = |\Delta x/x|$, либо приведенной ошибкой $\delta x = |\Delta x/x_{\max}|$; 6) максимальную скорость изменения аналогового сигнала $\dot{x}_{\max}$; 7) параметры, определяющие особенности подключения датчика к ИУВС, на пример выходной импеданс датчика или требуемый входной импеданс канала ввода информации в ИУВС.

Входная информация в виде релейных сигналов поступает в ИУВС, как правило, по схемам, примеры которых приведены на рис. 2.9. Для подобного типа датчиков о входной информации необходимо знать: 1) принадлежность каждого датчика к какой-либо функциональной группе (к системе стабилизации объекта управления, к системе управления манипулятором и т. д.); 2) максимальные и минимальные частоты включения каждого датчика и время нахождения датчика во включенном (выключенном) состоянии; 3) максимально допустимое время «дребезга» контактов каждого датчика.

Формулируя требования на разработку ИУВС, необходимо уточнить вышеуказанные параметры всех входных сигналов и для систематизации свести их в таблицу.

Выходные сигналы ИУВС могут быть также представлены либо в цифровой, либо в аналоговой форме, либо в виде релейных сигналов.

Примечание. Методы и способы преобразования информации из цифровой формы в аналоговую рассмотрены в § 1.7.

Кратко отметим особенности структурной реализации устройства сопряжения с объектом (УСО) для ввода информации в аналоговой форме.

В зависимости от структуры УСО делятся на параллельные, последовательные, последовательно-параллельные.

Параллельные УСО (рис. 2.10, а, где МД1 — входная магистраль данных) преобразуют информацию датчиков в структурно идентичных каналах. В состав каждого канала входят: МУ_і — масштабный усилитель с фиксированным или изменяемым по сигналу М_і коэффициентом передачи; УВЗ_і — устройство, предназначенное для выборки и запоминания информации в моменты поступления управляющих сигналов Т_і; Ф_і — фильтр (иногда с перестраиваемой структурой или изменяемыми значениями постоянных времени под воздействием управляющих сигналов У_і; АЦП_і — аналого-цифровой преобразователь; мультиплексор МС — общий для каналов передачи цифровых кодов на выходную магистраль данных МД2 микроЭВМ в зависимости от адреса на магистрали МА. В сложных ИУВС в состав УСО входит контроллер К, связанный с центральной микроЭВМ. В зависимости от характера входной информации микроЭВМ устанавливает режим работы контроллера К, который в свою очередь задает коэффициенты передачи усилителей, определяет моменты выборки и запо-

минания входной информации и устанавливает постоянные времени фильтров. В простейшем случае контроллер не имеет связи с микро-ЭВМ и работает по «жесткому» алгоритму с обеспечением синхронизации от внешних сигналов (например, от генератора, запитываю-

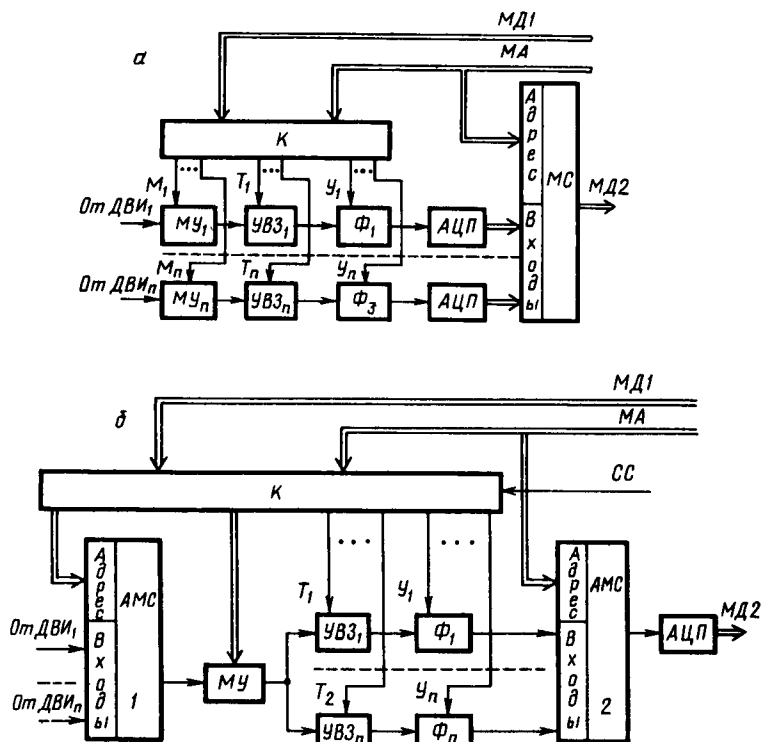


Рис. 2.10. Структуры параллельного (а) и последовательного (б) устройств сопряжения с объектом.

щего первичные датчики входной информации). Время преобразования входной информации в подобных УСО не зависит от числа каналов, а определяется лишь характеристиками блоков, входящих в их состав, т. е.

$$t_{\text{пар}} = t_{\text{к}} + t_{\text{АЦП}}.$$

В последовательных УСО объем аппаратных средств меньше, чем в параллельных (рис. 2.10, б, где МД1 и МД2 — соответственно входная и выходная магистрали данных; МА — магистраль адреса). В них сигналы от датчиков входной информации ДВИ_і поступают через аналоговый мультиплексор АМС1 на вход масштабного усилителя МУ. Выходы фильтров Ф_і через аналоговый мультиплексор АМС2 подключаются к входу АЦП. Последовательность коммутации входов мультиплексоров АМС1 и АМС2 задается контроллером К.

Как и в предыдущем случае, контроллер К может работать автономно и под управлением микроЭВМ. Сокращение оборудования достигается уменьшением в n раз количества масштабных усилителей и АЦП. Однако время преобразования возрастает с увеличением числа каналов:

$$t_{\text{посл}} = t_{\text{к}} + nt_{\text{АЦП}}$$

Последовательно-параллельные УСО оптимальны для специализированной, ориентированной на конкретное применение ИУВС. В таких УСО сигналы с датчиков входной информации объединены в группы с одинаковыми допустимыми временами и точностью преобразования входных данных. Внутри группы реализуется принцип последовательной обработки, а между группами — параллельной. Время преобразования при этом больше, чем в параллельных УСО, но меньше, чем в последовательных УСО:

$$t_{\text{посл-пар}} = t_{\text{к}} + mt_{\text{АЦП}},$$

где $m = n/k$ — количество каналов в каждой группе при равномерном распределении: k — количество групп.

Примечание. Методы построения узлов УСО изложены в § 1.1.

При обработке информации, представленной в цифровой форме, обмен осуществляется, как правило, унифицированными способами [14], а задача сглаживания реализуется программными средствами в микроЭВМ.

Для ввода релейных сигналов в УСО используется, как правило, структура, приведенная на рис. 2.11, где ДВИ_{*i*} — датчики входной информации; МА — магистраль адреса; МД — магистраль данных. Здесь входные сигналы объединены в группы по признаку функциональной принадлежности. В состав групп входят устройства сопряжения УС для приведения сигналов к требуемому уровню и фильтры Ф для устранения помех, в основном из-за «дребезга» контактов. Сигналы в каждой группе кодируются устройством кодирования УК и представляются в виде, удобном для передачи в микроЭВМ.

При разработке УСО ИУВС определяющими требованиями технического задания являются требования к разрядной сетке и частоте

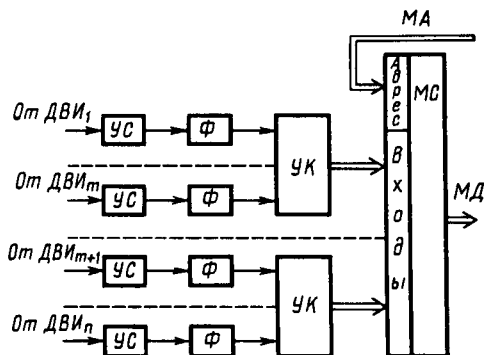


Рис. 2.11. Структура устройств сопряжения с объектом для ввода релейных сигналов.

опроса входных данных, т. е. требования к квантованию входных данных по уровню и времени.

Отметим, что требования обеспечения устойчивости и точности решения алгоритмов управления ИУВС в динамике, т. е. с учетом динамических погрешностей конкретных алгоритмов управления, могут привести к более жестким ограничениям при квантовании входных сигналов по времени, чем это вытекает из условия обеспечения точности представления входных данных. Ниже на примерах рассматривается методика анализа динамических погрешностей алгоритмов управления. Имея данные о динамических погрешностях как функции периода решения задачи управления, а следовательно, и частоты поступления входных данных (периода квантования), разработчик может решить вопрос о выборе периода квантования входных сигналов по времени с учетом допустимых динамических погрешностей алгоритма управления.

Анализ динамических погрешностей алгоритма управления. Динамические погрешности определяют из анализа устойчивости вычислительных алгоритмов, анализа амплитудно-частотных (АЧХ) и фазочастотных (ФЧХ) характеристик алгоритмов, выполняемых ИУВС. АЧХ и ФЧХ алгоритмов в свою очередь определяются видом решаемых зависимостей, тактом обмена информацией (тактом вычисления одной итерации, т. е. быстроедействие ИУВС), численным методом решения зависимости. Вид АЧХ и ФЧХ алгоритмов может быть найден с помощью аппарата преобразования Лапласа для импульсных переменных (z -преобразования).

Если аналоговая входная переменная (рис. 2.12, а), представленная импульсной последовательностью $x^*(t)$ (рис. 2.12, б), у которой длительность импульса $T_{\text{и}}$ много меньше периода T , поступает на линейное устройство с передаточной функцией $\omega(t)$, то для описания таких устройств (в частности, и ЭВМ) можно использовать аппарат z -преобразования.

Передаточная функция такой импульсной системы (рис. 2.13, а) имеет вид

$$W^*(z) = Y^*(z)/X^*(z) = L[\omega(t)\delta_T(t)] = \\ = \left\{ \sum_{n=-\infty}^N [A(s_n)/B'(s_n)] [1 - \exp(-T(s - s_n))]^{-1} \right\}_{z=\exp(sT)},$$

где $\delta_T(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$ — последовательность единичных импульсов; $A(s)/B(s) = W(s) = L[\omega(t)]$ — преобразование Лапласа передаточной функции устройства; $B'(s_n) = dB(s)/ds_{s \rightarrow s_n}$; $s_1, \dots, s_N$ корни выражения $B(s) = 0$.

Алгебра z -преобразований выходной величины для рассмотренных на рисунке случаев имеет следующий вид:

рис. 2.13, а $Y^*(z) = W^*(z)X^*(z)$;

рис. 2.13, б $Y^*(z) = X^*(z)L[W_1(s)W_2(s)]|_{z=\exp(sT)}$;

рис. 2.13, в $Y^*(z) = W_1^*(z)W_2^*(z)X^*(z)$;

рис. 2.13, $Y^*(z) = X^*(z) \frac{W^*(z)}{1 + L[W_1(s)W_2(s)]|_{z=\exp(Ts)}}$.

Важнейшие для практических применений z -преобразования последовательностей входных данных приведены в табл. 2.3.

Амплитудно-частотные и фазочастотные характеристики алгоритмов вычисляются путем перехода от дифференциальных уравнений,

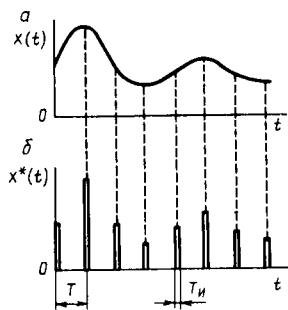


Рис. 2.12. Графическое представление аналоговой входной переменной (а) и ее импульсной последовательности (б).

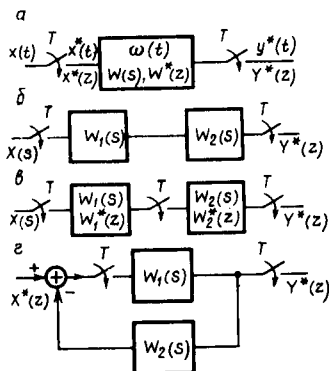


Рис. 2.13. Примеры z -преобразования.

связывающих выходные и входные переменные, к соответствующим конечно-разностным уравнениям с использованием численных методов решения и последующим применением к конечно-разностным уравнениям z -преобразования.

Пример 2.1. Рассмотрим применение z -преобразования для анализа динамических погрешностей вычислительного алгоритма дифференцирования входной величины. Другими словами, найдем вычислительный алгоритм, реализующий операцию дифференцирования с заданной динамической погрешностью $\Delta_{\text{дин}}$ для диапазона частот от 0 до f_{max} входных сигналов.

Учитывая необходимость вычислений в реальном масштабе времени, т. е. используя значения входной функции лишь в предшествующие моменты времени и второй полином Ньютона, можем записать формулы численного дифференцирования в виде

$$y'_k(t) = \frac{1}{2T} (3y_k - 4y_{k-1} + y_{k-2}) \quad \text{для } n=2; \quad (2.2)$$

$$y'_k(t) = \frac{1}{6T} (11y_k - 18y_{k-1} + 9y_{k-2} - 2y_{k-3}) \quad \text{для } n=3; \quad (2.3)$$

$$y'_k(t) = \frac{1}{12T} (25y_k - 48y_{k-1} + 36y_{k-2} - 16y_{k-3} + 3y_{k-4}) \quad (2.4)$$

для $n=4$,

где T — период квантования входных данных; $y_k, y_{k-1}, y_{k-2}, \dots, y_{k-4}$ — значения входной величины в моменты времени $k, k-1, k-2, \dots, k-4$; n — степень интерполирующего полинома.

Применяя z -преобразование к выражению (2.2), в частности используя свойство z -преобразования для последовательности запаздывающих входных данных табл. 2.3, получим

$$y'_k(z) = \frac{1}{2T} y_k(z) [z^{-2} - 4z^{-1} + 3] \quad \text{для } n=2.$$

Т а б л и ц а 2.3

Операция	Последовательность (оригинал)	Z-преобразование (изображение)
Линейность ($\alpha, \beta = \text{const}$)	$\alpha f_k + \beta g_k$	$\alpha F_z(z) + \beta G_z(z)$
Опережение	$f_{k+1} \equiv E f_k$ $f_{k+r} \equiv E^r f_k$	$z F_z(z) - f_0 z^{-1}$ $z^r F_z(z) - \sum_{i=0}^{r-1} f_i z^{r-i-1}$
Запаздывание	$f_{k-r} \equiv F^{-r} f_k$ ($r = 1, 2, \dots$)	$z^{-r} F_z(z)$
Конечные разности: нисходящие восходящие	$f_{k+1} - f_k \equiv \Delta f_k$ $f_k - f_{k-j} \equiv \nabla f_k$	$(z-1)F_z(z) - f_0(z)$ $\frac{z-1}{z} F_z(z)$
Суммирование последовательностей	$\sum_{i=0}^k f_i$	$\frac{z}{z-1} F_z(z)$
Свертка последовательностей	$\sum_{i=0}^{\infty} f_i g_{k-i}$	$F_z(z) G_z(z)$
Непрерывность (α не зависит от k и z)	$\lim_{\alpha \rightarrow a} f_k(\alpha)$	$\lim_{\alpha \rightarrow a} F_z(z, \alpha)$
Дифференцирование изображения	$k f_k$ $k^r f_k (r = 1, 2, \dots)$	$-z \frac{d}{dz} F_z(z)$ $-z \frac{d}{dz}; Z[k^{n-1} f_k; z]$

Учитывая, что $z^{\pm k} = \exp(\pm kTs) = \exp(\pm jkT\omega) = \cos k\omega T \pm j \sin k\omega T$, вычислим передаточную характеристику программы дифференцирования:

$$W_{\text{пр}}(z) = \frac{y'_k(z)}{y_k(z)} = \frac{1}{2T} [z^{-2} - 4z^{-1} + 3]; \quad W_{\text{пр}}(j\omega) = \frac{1}{2T} [M(\omega) + jN(\omega)].$$

Тогда амплитудно-частотная характеристика программы

$$A_{\text{пр}}(\omega) = \sqrt{M^2(\omega) + N^2(\omega)}; \quad (2.5)$$

фазочастотная характеристика

$$\varphi_{\text{пр}}(\omega) = \arctg \frac{N(\omega)}{M(\omega)}, \quad (2.6)$$

где $M(\omega) = 3 + \cos 2\omega T - 4 \cos \omega T$; $N(\omega) = -\sin 2\omega T + 4 \sin \omega T$.

Аналогично получим передаточную характеристику, а следовательно, АЧХ (2.5) и ФЧХ (2.6) для программ дифференцирования по формуле (2.3):

$$W_{np}(z) = \frac{1}{6T} [11 - 18z^{-1} + 9z^{-2} - 2z^{-3}] = \frac{1}{6T} [M(\omega) + jN(\omega)],$$

где $M(\omega) = 11 - 18 \cos \omega T + 9 \cos 2\omega T - 2 \cos 3\omega T$; $N(\omega) = 18 \sin \omega T - 9 \sin 2\omega T + 2 \sin 3\omega T$,

и по формуле (2.4):

$$W_{np}(z) = \frac{1}{12T} [25 - 48z^{-1} + 36z^{-2} - 16z^{-3} + 3z^{-4}] = \frac{1}{12T} [M(\omega) + jN(\omega)],$$

где $M(\omega) = 25 - 48 \cos \omega T + 36 \cos 2\omega T - 16 \cos 3\omega T + 3 \cos 4\omega T$; $N(\omega) = 48 \sin \omega T - 36 \sin 2\omega T + 16 \sin 3\omega T - 3 \sin 4\omega T$.

Вычислив зависимости АЧХ и ФЧХ программ дифференцирования $A_{np}(\omega)$, $\varphi_{np}(\omega)$ для ряда значений периода квантования T в интервале ω , соответствующем полосе частот входных сигналов ($f_{\min} - f_{\max}$), можно решить, какой из рассмотренных алгоритмов численного дифференцирования (2.2) — (2.4) удовлетворяет условию обеспечения максимально допустимой динамической погрешности (амплитудной $\Delta_{\text{дин}}^A$ и фазовой $\Delta_{\text{дин}}^\varphi$ при минимальной сложности вычислений и (или) максимальном периоде квантования). Другими словами, находим, какой из рассмотренных в примере алгоритмов численного дифференцирования обеспечивает (допускает) максимальное значение периода квантования входных данных $T_{\max}$ и приемлемой сложности. Последнюю можно, например, оценивать объемом требуемой памяти программы при выполнении условий $|A_{np}(\omega) - A(\omega)| \leq \Delta_{\text{дин}}^A$; $|\varphi_{np}(\omega) - \varphi(\omega)| \leq \Delta_{\text{дин}}^\varphi$, где $2\pi f_{\min} \leq \omega = 2\pi f = 2\pi f_{\max}$; $A(\omega)$ и $\varphi(\omega)$ — АЧХ и ФЧХ функции дифференцирования соответственно; $\Delta_{\text{дин}}^A$ и $\Delta_{\text{дин}}^\varphi$ — максимально допустимые (заданные) значения динамических погрешностей АЧХ и ФЧХ программ численных методов дифференцирования. При этом должно обеспечиваться условие устойчивости (сходимости) алгоритма. Последнее проверяется моделированием работы выбранных числовых алгоритмов на универсальной ЭВМ или путем оценки погрешности интерполяционного метода по разности точного решения и полученной вычислительной формулы.

Из рис. 2.14, на котором приведены зависимости $A_{np}(\omega)$ и $\varphi_{np}(\omega)$ для формулы

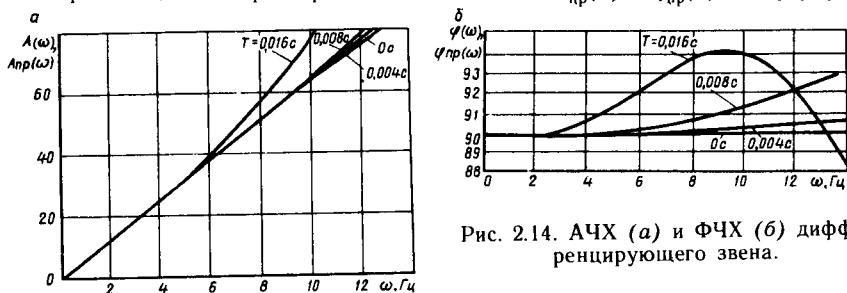


Рис. 2.14. АЧХ (а) и ФЧХ (б) дифференцирующего звена.

(2.3) в диапазоне частот от 0 до 14 Гц и зависимости $A(\omega)$ и $\varphi(\omega)$ (случай $T=0$ с), видно, что динамические погрешности $\Delta_{\text{дин}}^A = 5$ дБ и $\Delta_{\text{дин}}^\varphi = 1^\circ$ обеспечиваются вычислительным алгоритмом (2.3) при $T \leq 0,004$ с. Следовательно, и период обновления входных данных, т. е. период квантования входных данных, должен выбираться меньшим или равным 0,004 с.

Отметим, что для ИУВС, работающих в реальном масштабе времени, анализ динамических погрешностей алгоритмов позволяет определить ряд важнейших технических характеристик ИУВС: требуемое быстродействие, информационные объемы оперативной и постоянной памяти, систему команд, вычислительные алгоритмы, циклограммы обработки входных данных.

Обоснование основных требований, предъявляемых к МП ИУВС.

Техническое задание (ТЗ) на разработку ИУВС должно включать в себя: цель разработки и назначение ИУВС; технические требования; требования к стандартизации и унификации; технико-экономические показатели; эргономические требования; порядок разработки и испытаний ИУВС.

В свою очередь, технические требования включают в себя требования: функциональные; технические, по составу ИУВС; к конструкции и технологичности; к сырью, исходным материалам и покупным изделиям; к надежности; эксплуатационные и ремонтные; к упаковке, хранению и транспортировке. К функциональным требованиям относятся, например, порядок и способы взаимодействия ИУВС с сопрягаемыми объектами; показатели, определяющие эффективность ИУВС (точность вычислений, время готовности к работе и др.); электромагнитную совместимость; расход электроэнергии; специфические эксплуатационные и т. д.

Основными техническими параметрами при разработке ИУВС, как правило, служат: разрядность и форма представления информации (фиксированная или (и) плавающая точка); разрядность и форматы команд; состав и число команд (либо требование унификации системы команд); время выполнения отдельных команд или производительность на смеси команд; информационный объем ЗУ (ОЗУ, ПЗУ, ППЗУ в отдельности); возможности по наращиванию информационной емкости ЗУ; номенклатура датчиков входной информации и параметры входных сигналов; требования к устройствам вывода и индикации информации; архитектура (структура аппаратных средств, формат и система команд); число уровней прерывания, система прерываний; особенности системного программного обеспечения; требования к встроенному контролю; уровень реконфигурации аппаратных средств; организация интерфейсов (магистральная, радиальная, древовидная, радиально-магистральная) и физические длины линии связи между устройствами ИУВС; скорость передачи информации и протоколы обмена информацией по внешним интерфейсам; схемотехнические особенности интерфейсов (типы линий связи, логические уровни передаваемых сигналов, входные и выходные сопротивления, гальваническая развязка и т. п.); параметры первичных источников питания (номиналы напряжений, допуски на изменения, допуски на уровень помех).

Технические требования должны включать в себя следующие *конструктивные и технологические требования:* к габаритным, установочным и присоединительным размерам, способам крепления элементов ИУВС на объекте; ограничения по массе; ограничения по виду исполнения (блочное, моноблочное); доступ ко всем элементам системы в рабочем и нерабочем состояниях; ограничения по количеству транспортируемых единиц; по средствам защиты от климатических, механических и других факторов воздействия; по обеспечению безопасности при эксплуатации; по применению прогрессивных технологических процессов; по обеспечению взаимозаменяемости в усло-

виях производства, эксплуатации и ремонта; к виду и составу ЗИПа и поверочных средств.

Надежные показатели в зависимости от особенностей использования ИУВС чаще всего задаются: вероятностью безотказной работы в течение некоторого интервала времени; наработкой на отказ; функцией и коэффициентом готовности; средним временем между двумя отказами, средним временем восстановления [15].

Эксплуатационные требования, как правило, характеризуются комплексом климатических, механических и других факторов воздействия; устойчивостью к влиянию внешних физических полей.

Кроме того, технические требования могут включать в себя специальные требования, например к патентной чистоте, к контролю ИУВС (ручной, автоматизированный), КПА и средствам обслуживания, к эксплуатации в аварийных ситуациях, к математическому обеспечению.

На практике при разработке ИУВС разработчик зачастую сталкивается с проблемой формулировки ТЗ в выше описанном виде по исходным данным, представленным в неявной форме. Рассмотрим следующие инженерные методики: 1) оценки разрядности и формата команд, выбора адресности; 2) оценки информационного объема памяти; 3) определения длины разрядной сетки; 4) выбора системы команд; 5) оценки характеристик системы прерывания.

Оценка разрядности и формата команд, выбор адресности. Основным фактором, определяющим форматы и адресность, является принятая к реализации в микроЭВМ ИУВС система команд. При этом возможны: 1) использование регламентированной, не изменяемой в процессе проектирования ИУВС системы команд; 2) разработка специальной системы команд для конкретного применения ИУВС.

В первом случае система команд может быть директивной, определенной требованиями ТЗ или обусловленной выбором типа микропроцессора.

В настоящее время к разрабатываемым микроЭВМ ИУВС часто предъявляют требование использования системы команд машин серии СМ ЭВМ. Это требование позволяет сохранить преемственность ИУВС и использовать для отладки программного обеспечения средства, серийно выпускаемые промышленностью (например, СМ-4, «Электроника-60»). Один из основных форматов команды показан на рис. 2.15, а. Десять разрядов служат для указания кода операции, три — для кодирования номера регистра и три — для указания способа адресации [14]. В данной системе команд широко применяются прямая, косвенная и относительная адресации, а также более эффективная форма последней — индексная адресация [16, 14, 4].

В некоторых типах микропроцессоров, используемых для построения микроЭВМ ИУВС, существует единственный способ программирования — на языке команд микропроцессоров. Многие из них не имеют микропрограммного управления, однако ряд микропроцессоров выполнен на одном кристалле с ПЗУ, что делает практически

невозможным изменение содержимого последнего. Например, в МП серии К580ИК80 принята байтовая структура, причем в зависимости от типа команд требует от 1 до 3 байт информации [17]. На рис. 2.15, б представлена запись команды MVI загрузки второго байта в аккумулятор А.

В обоснованных случаях процесс проектирования микроЭВМ ИУВС включает в себя разработку специальной системы команд.

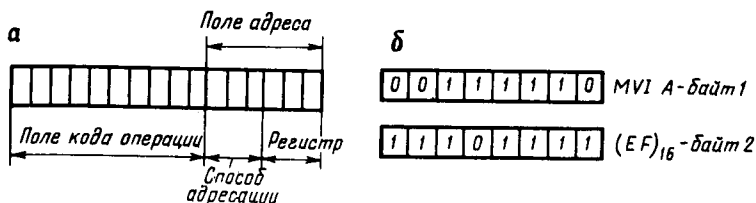


Рис. 2.15. Пример формата команды (а) и запись команды MVI микропроцессора К580 (б).

При этом предполагается, что в качестве аппаратных средств микроЭВМ используются микропроцессоры с микропрограммным управлением, позволяющие создать требуемый набор команд (микропроцессоры серий К589, К1802, К1804). В формате команды число разрядов, отводимых для представления кода операции, определяется в основном типом микропроцессора, а число адресов и их разрядность — емкостью памяти для хранения программ и временем их решения.

Существующие методики [3] позволяют сделать выбор адресности по двум критериям: минимума объема памяти для хранения программ; минимума времени решения задачи.

Выбор адресности по критерию минимума объема памяти показывает, что программы для одноадресных машин в большинстве случаев экономичнее, чем для двух- и трехадресных.

Выбор адресности по критерию минимума времени решения задачи базируется на оценке выигрыша (проигрыша) времени решения конкретной задачи на машинах с различной адресностью.

Оценка типовых алгоритмов показывает, что: 1) в случае последовательного алгоритма, когда результат предыдущей операции непосредственно используется в следующей, одноадресная ЭВМ обладает преимуществом; 2) в случае параллельного алгоритма, когда результат предыдущей операции не используется в следующей, а отсылается в оперативную память, трехадресная ЭВМ предпочтительнее, так как при этом для одноадресной ЭВМ на каждую арифметическую операцию приходится две вспомогательных (вызов из памяти в АЛУ и загрузка в память); 3) в случае комбинированного алгоритма, когда его составляющие имеют последовательный и параллельный характер, выигрыш можно представить как $\Delta t = 2\tau_0(N_{A2} - N_{A1} - N_y)$, где N_{A2} — число арифметических операций, выполняемых в виде параллельного алгоритма; N_{A1} — количество арифметических операций, выполняемых последовательно; N_y —

число управляющих команд; τ_0 — время цикла обращения к памяти.

Следовательно, если $N_{A2} > (N_{A1} + N_y)$, то $\Delta t > 0$ и преимущество имеет трехадресная ЭВМ; при $N_{A2} < (N_{A1} + N_y)$ $\Delta t < 0$ и целесообразно построение одноадресной ЭВМ.

Разрядность адреса определяется количеством ячеек системы памяти, количеством источников и приемников информации, а также способами адресации.

В специализированных микроЭВМ система памяти, как правило, включает в себя: ОЗУ для хранения вводимой информации и промежуточных результатов вычислений; ППЗУ — перепрограммируемое (чаще всего электрически) постоянное запоминающее устройство; ПЗУ — постоянное запоминающее устройство для хранения рабочих программ, тестов и т. д.

При использовании в микроЭВМ *прямой адресации* разрядность может быть определена как

$$R_a = \lceil \log_2(N_{\text{ОЗУ}} + N_{\text{ППЗУ}} + N_{\text{ПЗУ}} + N_n + N_p) \rceil,$$

где $N_{\text{ОЗУ}}$, $N_{\text{ППЗУ}}$, $N_{\text{ПЗУ}}$ — число ячеек оперативной, постоянной и перепрограммируемой памяти соответственно; N_n , N_p — число источников и приемников информации; $\lceil \rceil$ — знак, обозначающий округление до ближайшего большего целого числа.

При *относительной адресации* обеспечивается сокращение адресной части команды, так как она является смещением относительно определенного регистра процессора. Чаще всего в качестве регистра используется счетчик команд. При $R_a = 8$ можно охватить до 256 адресов относительно программного счетчика.

При *индексной адресации* адресная часть команды рассматривается как смещение относительно индексных регистров процессора. Этим также достигается уменьшение разрядности адресной части команды. Исполнительный адрес формируется как сумма содержимого индексного регистра и R_a .

В микроЭВМ ИУВС широко распространена также *непосредственная адресация*, когда в разрядах адресной части команды содержится операнд. Непосредственная адресация удобна для занесения постоянных величин в регистр или их использования в операциях АЛУ.

Основное требование при выборе способа адресации — обеспечение удобства программирования на микроЭВМ.

Оценка информационного объема памяти. Вычислительные возможности микроЭВМ ИУВС определяются не только типом используемых микропроцессоров и системой команд, но в значительной степени зависят и от характеристик системы памяти. В состав системы памяти, как правило, входят ОЗУ, ПЗУ и ППЗУ.

Исходными данными для оценки числа слов ОЗУ являются: число входных a и выходных b величин, применяемых в алгоритмах; число одновременно хранимых на каждом i -м этапе вычислений промежуточных величин q_i^k для каждого k -го алгоритма; число уровней прерывания n .

Так как каждый входной параметр заносится в одну ячейку ОЗУ и после обработки вводится в другую, то объем памяти, необходимый для хранения входных и выходных величин, $N_{b/b} = 2a + b$.

Объем ОЗУ для хранения промежуточных величин

$$N_{\text{пр}} = \sum_{k=1}^n (\max_i \{q_i^k\}) + nq_n,$$

где q_n — количество ячеек памяти для хранения рабочих регистров микропроцессора, необходимых для запоминания по прерванной программе.

С учетом средств программной защиты от помех и программ предварительной обработки информации [13] объем ОЗУ

$$N_{\text{ОЗУ}}^0 \geq 1,4[2a + b + \sum_{k=1}^n (\max_i \{q_i^k\}) + nq_n].$$

Кроме того, необходимо учесть количество рабочих ячеек, отводимых под тестовые $N_{\text{т}}$ и стандартные $N_{\text{сн}}$ программы. Тогда

$$N_{\text{ОЗУ}} = N_{\text{ОЗУ}}^0 + N_{\text{т}} + N_{\text{сн}}.$$

Объем ПЗУ складывается в основном из объемов: рабочих программ $N_{\text{рп}i}$ для реализации всех k алгоритмов управления; управляющей программы-диспетчера $N_{\text{д}}$; программ прерываний $N_{\text{п}}$; тестовых программ $N_{\text{т}}$; стандартных подпрограмм $N_{\text{ст}}$; ячеек для хранения констант $N_{\text{с}}$. Таким образом,

$$N_{\text{ПЗУ}} = \sum_{i=1}^k N_{\text{рп}i} + N_{\text{д}} + N_{\text{п}} + N_{\text{т}} + N_{\text{ст}} + N_{\text{с}}.$$

Значение $N_{\text{ПЗУ}}$ может быть окончательно определено после составления и отладки программ с учетом выбранной системы команд. С большой степенью достоверности необходимые объемы ОЗУ и ПЗУ можно оценить с помощью отладки программ на аппаратурно-программных комплексах проектирования, прототипных микропроцессорных комплектах и универсальных ЭВМ с использованием кросс-средств.

Объем $N_{\text{ПЗУ}}$ определяется системными требованиями, предъявляемыми к ИУВС. В частности, пропадание питающих напряжений в микроЭВМ приводит к потере набранной и хранимой в ОЗУ информации, если не приняты специальные меры.

В специализированных ИУВС алгоритмы построены так, что при восстановлении в момент времени t_i питания происходит опрос датчиков входной информации и набор потерянной информации. Однако существуют входные величины, восстановление которых в момент времени t_{i+1} принципиально невозможно (например, время наработки ИУВС, параметры, введенные оператором в момент включения системы, координаты о положении объекта управления в моменты времени t_{i-1} , t_{i-2} , ..., t_{i-k} и т. д.). Количество таких величин может быть определено достаточно точно непосредственно из анализа алгоритмов.

В зависимости от технических возможностей устранение потери информации может быть реализовано: 1) перезаписью в момент времени t_i невосстанавливаемых величин в ППЗУ; 2) постоянным хранением всех невосстанавливаемых величин в ППЗУ.

Требуемый объем ППЗУ:

в первом случае $N_{\text{ППЗУ}} = \max\{N_{\text{нв}i}\} + n_{\text{нво}}$;

во втором случае $N_{\text{ППЗУ}} = \sum_{i=1}^n N_{\text{нв}i} + N_{\text{нво}}$,

где $N_{\text{нв}i}$ — число невосстанавливаемых величин, которые используются только в i -м алгоритме; $N_{\text{нво}}$ — число невосстанавливаемых величин, которые являются общими для нескольких алгоритмов.

Определение длины разрядной сетки. Основными требованиями, предъявляемыми к ИУВС, являются выработка управляющих воздействий y_i на объект управления с точностью не хуже заданной $\sigma_{i \text{ доп}}$ и выработка y_i с временем запаздывания не более допустимого $\Delta t_{i \text{ доп}}$.

Таким образом, при реализации алгоритмов управления имеют место ограничения

$$\sigma_i \leq \sigma_{i \text{ доп}}; \Delta t_i \leq \Delta t_{i \text{ доп}},$$

где σ_i , Δt_i — ошибка и время запаздывания выработки i -го воздействия.

Ошибка выработки i -го воздействия может быть представлена в виде

$$\sigma_i^2 = \sigma_{\text{м}i}^2 + \sigma_{\text{т}i}^2 + \sigma_{\text{н}i}^2,$$

где $\sigma_{\text{м}i}$, $\sigma_{\text{т}i}$, $\sigma_{\text{н}i}$ — среднеквадратичные значения методической, трансформируемой и инструментальной ошибок.

Причиной возникновения методических ошибок является в ряде случаев замена требуемой функции управления $y_i^* = f_i^*\{x_1, x_2, \dots, x_n\}$ более простой в реализации, аппроксимирующей функцией $y_i = f_i\{x_1, x_2, \dots, x_n\}$.

Ошибки $\sigma_{\text{т}i}$ обусловлены трансформацией входных ошибок σ_{xj} аргументов x_j . В свою очередь ошибки σ_{xj} в ИУВС складываются из инструментальных и методических ошибок измерения датчиков входной информации и ошибок преобразования в УСО аналоговых величин в код [10]. Третья составляющая $\sigma_{\text{н}i}$ вызвана ограниченностью длины l разрядной сетки, используемой для представления величин в микроЭВМ.

Распределение ошибок по составляющим $\sigma_{\text{м}i}$, $\sigma_{\text{т}i}$, $\sigma_{\text{н}i}$ зависит в основном от технических возможностей построения датчиков входной информации, УСО, выбора численных методов и длины l разрядной сетки микроЭВМ.

При разработке микроЭВМ ИУВС наиболее часто ставится задача определения длины разрядной сетки с учетом предварительно выбранных численных методов и известных ошибок σ_{xj} . Выбором

длины разрядной сетки требуется обеспечить выполнение соотношения $\sigma_{ni}^2 \leq \sigma_{i \text{ доп}}^2 - \sigma_{ti}^2 - \sigma_{mi}^2$ при наименьшем количестве разрядов.

Существует два основных метода определения разрядности микроЭВМ: 1) аналитического расчета; 2) статистического моделирования.

При использовании метода аналитического расчета последовательно определяются значения σ_{ti} , σ_{mi} , а затем σ_{ni} . Этот метод является весьма приближенным, что вызывается рядом причин [3]: а) отсутствием учета трансформации ошибок округления, появляющихся в процессе вычислений; б) допущением нормального закона распределения ошибок без специального анализа; в) оценкой ошибок по общему количеству операций с округлением без учета ветвления алгоритмов выработки параметров y_i .

Для более точной оценки ошибок выработки управляющих воздействий y_i целесообразно использование методов статистического моделирования путем многократного решения задач при различных наборах входных переменных [3]. Для реализации статистического моделирования при проектировании ЭВМ на основе микропроцессоров широко применяют аппаратно-программные средства, к которым относятся оценочные модули, аппаратно-программные отладочные комплексы. Оценочные модули представляют собой одноплатные микроЭВМ с минимальным набором технических средств и программой-монитором, размещенной в ПЗУ. С учетом небольшого объема памяти (256 байт — 2К байт) они позволяют оценить ошибки вычисления по наиболее критическому параметру ИУВС. Для более полной оценки (а практически точного расчета по согласованному с потребителями тестам) ошибок выработки параметров и окончательного выбора разрядности микроЭВМ нужны аппаратно-программные отладочные комплексы, которые имеют внутрисхемную эмуляцию отлаживаемых программ, компиляторы с языков высокого уровня и большой набор устройств ввода-вывода.

Основное достоинство применения отладочных средств — высокая достоверность результатов, снижение трудоемкости, одновременная отладка технических средств и программного обеспечения микроЭВМ. Кроме того, использование секционированных микропроцессорных комплектов допускает оценку длины разрядной сетки микроЭВМ с точностью до n , где n — разрядность одной секции. Для большинства микропроцессорных комплектов с наращиваемой разрядностью $n = 2, 4, 8, 16$ (серии К589, К584, К588)

Микропроцессоры с фиксированной разрядностью, не наращиваемые аппаратно (например, микропроцессоры серий К580ИК80), требуют оценки кратности разрядности обрабатываемых слов («удвоенная», «утроенная», n -кратная). При этом наиболее критичными становятся вопросы временных затрат Δt_i , так как время выполнения отдельных операций увеличивается в $\sim (1,1 - 1,3) n$. Возможной является ситуация, когда для n -кратной обработки $\sigma_i \leq \sigma_{i \text{ доп}}$, но $\Delta t_i > \Delta t_{i \text{ доп}}$. В этом случае целесообразно введение $(n + 1)$ -

кратной обработки только в «узких» местах наибольшей потери точности. Эффективным методом оценки является статистическое моделирование с помощью аппаратурно-программных отладочных средств.

Выбор системы команд. Одним из ответственных этапов при проектировании МП ИУВС является выбор системы и форматов команд. При выборе системы команд ИУВС следует учитывать два важных момента.

Во-первых, с учетом тенденций совершенствования архитектуры МП ИУВС в направлении сближения последней с архитектурой универсальных ЭВМ необходимо во всех допустимых случаях в качестве системы команд ИУВС выбирать систему команд (в целом или ее подмножество) универсальных вычислительных машин с сохранением форматов команд. Этим достигается программная совместимость ИУВС с универсальной ЭВМ, что в свою очередь позволяет использовать универсальную ЭВМ как технологическую, на которой может быть выполнено программирование рабочих программ ИУВС с использованием системного программного обеспечения универсальной ЭВМ и полное или частичное моделирование алгоритмов работы ИУВС. Применение в ИУВС системы команд универсальной ЭВМ в несколько раз уменьшает затраты на создание математического обеспечения ИУВС, которые могут составлять большую часть всех затрат на разработку ИУВС; сокращает сроки разработки; позволяет параллельно вести разработку аппаратурных средств и рабочих программ. Поэтому при разработке МП ИУВС, в первую очередь однопроцессорных, рекомендуется для ИУВС с обработкой 32-разрядной информации выбирать систему и форматы команд ЕС ЭВМ, а в качестве технологической ЭВМ — одну из ЭВМ, входящих в ряд ЕС ЭВМ; для ИУВС с обработкой в основном 16-разрядной информации выбирать систему команд СМ ЭВМ и соответственно в качестве технологической ЭВМ — одну из мини-ЭВМ ряда СМ ЭВМ.

Во-вторых, в ряде случаев, когда при построении ИУВС определяющим моментом является построение ее на базе конкретного микропроцессорного набора, система и формат команд полностью определяются свойствами этого микропроцессорного набора, как, например, для микропроцессорных наборов с постоянной системой команд: К580, КР588, К1801 (см. табл. В.1).

В общем случае, когда системы команд универсальных ЭВМ (СМ ЭВМ и ЕС ЭВМ) не могут быть применены в первую очередь из-за недостаточного быстродействия или при необходимости построения ИУВС с точки зрения быстродействия, надежности, территориальной распределенности как мультимикропроцессорной, для разрабатываемой ИУВС выбирается специализированная система команд, оптимизирующая архитектуру ИУВС на выполнение конкретных задач. При этом следует помнить, что выбор специализированной системы команд приводит в общем случае к разработке вновь в полном объеме системного программного обеспечения (микроассемблера, кроссассемблера, эмулирующих средств, операционных

систем, прикладного программного обеспечения), что представляет собой огромнейшую по объему и трудоемкости техническую задачу.

Поэтому переход к специализированной системе команд должен быть всесторонне и тщательнейшим образом обоснован.

Примечание. Особенности разработки системного программного обеспечения микроЭВМ изложены в [16].

Единой инженерной методики выбора системы команд в настоящее время не существует [3], поскольку отдельные этапы формального выбора оптимальной системы команд нуждаются в данных, которые не могут быть получены аналитически, и эти данные, как правило, получают путем перебора большого числа вариантов. Статистика, собираемая разработчиком в процессе перебора вариантов, весьма разнообразна; отметим лишь основные характеристики, необходимые для решения задачи выбора системы команд: 1) частота встречи отдельных команд или групп команд в процессе решения типовых задач; 2) отношение времени выполнения этих команд к общему времени решения задач; 3) количество команд рабочих программ, использующих результаты предыдущих команд и не использующих их; 4) число команд с модификацией адресной части и отношение их к общему числу выполняемых команд; 5) оценка точности выполнения арифметических команд и т. п.

Напомним, что *команда* — это упорядоченная совокупность битов информации, представляющая наименование операции, инициируемой командой, и адреса операндов, участвующих в выполнении операции. Длина команды в битах, количество полей, требуемых для кодирования элементов команды (операции, адресов, признаков), и расположение полей в команде определяют формат команды. Выбор формата команды, в частности адресности, зависит от большого числа факторов, важнейшими из которых являются: тип машинных элементов информации (слова или байты); длина машинного слова; наличие внутренней памяти микропроцессора; класс алгоритмов, для реализации которого предназначена ИУВС; способы адресации, необходимые для эффективной обработки информации.

Инженерная методика выбора системы команд заключается в определении списка операций в предположении выбранных форм представления информации (фиксированной или плавающей точкой) и длины разрядной сетки, после чего решается задача выбора системы адресации. В [3] предложен следующий алгоритм определения списка команд. Пусть арифметическое устройство ИУВС выполняет множество $\{k_i\} = k_1, k_2, \dots, k_i, \dots, k_n$ типов команд, частота встреч которых соответственно $m_1, m_2, \dots, m_i, \dots, m_n$ (причем $\sum_{i=1}^n m_i = M_0$ — операционная длина алгоритма). Пусть каждая операция характеризуется временем выполнения $\tau_1, \tau_2, \dots, \tau_i, \dots, \tau_n$ при аппаратном и $t_1, t_2, \dots, t_i, \dots, t_n$ при программном способах реализации.

Предполагаем, что каждый тип команды характеризуется мно-

жеством затрат оборудования $Q_1, Q_2, \dots, Q_n$ при аппаратурной и множеством затрат оборудования $R_1, R_2, \dots, R_n$ при программной реализациях команд.

При принятых допущениях задача выбора оптимальной системы команд заключается в выборе множества таких команд, реализуемых арифметическим устройством, чтобы суммарные аппаратурные затраты были минимальными при выполнении временных ограничений на реализацию алгоритма T_0 , т. е.

$$\sum_{i=1}^n \lambda_i Q_i + \sum_{i=1}^n \bar{\lambda}_i R_i = \min \quad \text{при} \quad \sum_{i=1}^n \lambda_i m_i \tau_i + \sum_{i=1}^n \bar{\lambda}_i m_i t_i \leq T_0,$$

где $\lambda_i = 1$, $\bar{\lambda}_i = 0$, если i -я операция выполняется аппаратурно, и $\lambda_i = 0$, $\bar{\lambda}_i = 1$, если i -я операция выполняется программно.

Примечание. Примеры широко применяемых систем и форматов команд приведены в [4].

Отметим, что для ИУВС, как правило, характерно расширение списка стандартных команд универсальных ЭВМ (арифметических с фиксированной и плавающей точкой, логических, управления, пересылок, сдвигов, сравнения, преобразования, обращения к памяти, условных и безусловных переходов) специализированными командами, например командами вычисления квадратного корня, $\sin$, $\cos$, tg , $\arcsin$, $\arccos$, $\arctg$, $\lg$, $\ln$, $\exp$ и др.

Для адаптации аппаратурных средств ИУВС к решению конкретных алгоритмов можно применить переменную систему команд. В этом случае микропрограммное устройство управления ИУВС вместо ПЗУ микропрограмм фиксированного списка команд содержит ОЗУ микропрограмм переменного списка команд. Конкретный набор микропрограмм, реализующих некоторый список команд, загружается из ПЗУ банка микропрограмм в ОЗУ микропрограмм перед работой ИУВС, тем самым достигается оперативная смена системы команд, оптимизирующая характеристики ИУВС. Аппаратурные средства ИУВС при этом остаются неизменными.

Примечание. Пример специализированной системы команд приведен в § 2.4.

Оценка характеристик системы прерываний. Система прерывания — совокупность аппаратурных и программных средств, которые при наличии запросов обеспечивают прекращение выполнения текущей программы, работу по вызванной программе и возврат к прерванной. Система предназначена для обеспечения эффективной мультипрограммной работы и организации взаимодействия в многопроцессорных ИУВС.

Источниками прерываний в ИУВС могут быть сигналы счетчика реального времени, аварийные сигналы от схем контроля, инициативные запросы датчиков входной информации, запросы от других процессоров и т. д.

Основными параметрами системы прерываний являются [14]: время реакции t_p — время между появлением запроса на прерывание и началом выполнения первой полезной команды вызываемой

программы; время запоминания t_z ; время восстановления t_v ; структура прерываний — средства, позволяющие микроЭВМ обслуживать запросы в соответствии с присвоенными им приоритетами.

Наиболее часто в ИУВС, работающих в реальном масштабе времени, используются относительные, абсолютные и абсолютно-относительные системы организации приоритетов, причем в системах с относительными приоритетами начатое обслуживание заявки любого приоритета продолжается до полного завершения, а в системах с абсолютными приоритетами исполнение заявки j -го приоритета прерывается при поступлении заявки приоритета $i > j$.

При организации вычислительных процессов в микроЭВМ ИУВС возникает задача рационального назначения приоритетов. Для решения поставленной задачи необходимо знание *характеристик потоков заявок*: количество потоков заявок различных типов n ; интенсивность потоков заявок каждого типа λ_i ; взаимная относительная важность отдельных типов заявок α_i ; распределение интервалов времени между последовательными заявками каждого типа $A_i(t)$.

Примечание. В [18] выводятся основные соотношения для оценки параметров систем приоритетов.

В зависимости от характера решаемых задач и природы возникновения заявок используются различные *оценки эффективности организации системы приоритетов*. Для заявок, ценность которых снижается пропорционально времени, критерием служит *функционал*

$$C = \sum_{i=1}^n \alpha_i \lambda_i W_i, \quad (2.7)$$

где W_i — среднее время ожидания в очереди заявок i -го потока.

Правильное назначение приоритетов заявкам обеспечивает минимизацию функционала C .

Для систем с относительной организацией приоритетов назначение осуществляется в *порядке убывания отношений*:

$$(\alpha_k/T_k) > (\alpha_{k+1}/T_{k+1}), \quad (2.8)$$

где T_k — время обслуживания заявки k -го потока.

Соотношение (2.8), полученное для произвольного количества пуассоновских потоков заявок и широкого класса функций распределения времени обслуживания $B_i(t)$, справедливо при назначении абсолютных приоритетов в случае экспоненциального распределения времени обслуживания заявок.

Для достижения наибольшей эффективности микроЭВМ с учетом экономии аппаратурных средств в системах реального счета времени часто используют *смешанные системы организации приоритетов*, например в случае, когда n заявок разбивается на k групп с абсолютными приоритетами между группами и относительными — внутри каждой группы.

Приведенные выше соотношения позволяют оценить основные

характеристики систем приоритетов и правильно построить вычислительный процесс ИУВС.

Пример 2.2. В качестве примера формулировки технического задания на МП ИУВС рассмотрим формулировку ТЗ на устройство числового программного управления (УЧПУ) «Электроника НЦ-31», предназначенного для применения в оперативной системе управления универсальным токарным станком, снабженным следящими приводами и фотоэлектрическими импульсными измерительными преобразователями (ИП).

1. Технические требования:

1.1. В УЧПУ должны входить: два процессора (П); ПЗУ (для исполнительных программ); ОЗУ (для программ обработки и промежуточных результатов); ППЗУ (для констант станка и устройства управления); пульт оператора (ПО); программируемый таймер (ПТ); адаптер канала (АК); контроллеры приводов (КП); контроллеры измерительных преобразователей (КИП); контроллеры электроавтоматики (КЭ); блоки питания.

1.2. Структура УПЧУ должна обеспечивать возможность построения многопроцессорных (до четырех процессоров) систем с локальной и общей памятью и обеспечивать возможность подключения отладочной системы и кассетного накопителя информационной емкостью не менее 1К слов.

1.3. Технические характеристики процессора:

Система команд: команды микроЭВМ «Электроника НЦ-03Т» с добавлением операции умножения 16-битных операндов в РОН, арифметическо-логических операций с 16- и 32-битными операндами в РОН.

Число 16-битных РОН-16.

Производительность (тыс. операций/с): 120 — сложение 16-битных операндов в РОН; 70 — пересылки РОН-память; 40 — сложение 32-битных операндов в РОН; 6 — умножение 16-битных операндов в РОН.

1.4. Технические характеристики памяти: информационная емкость ОЗУ $\geq 2К$, ПЗУ ≥ 2 слова.

1.5. Технические характеристики пульта оператора (ПО):

Лицевая панель ПО должна содержать 14 цифровых индикаторов, 29 светодиодов, 46 клавиш.

Органы управления и индикации ПО должны обеспечивать ввод, редактирование, исполнение программ и управление станком в различных режимах работы.

1.6. Технические характеристики программируемого таймера (ПТ):

Программируемый таймер (каждый на свой процессор) предназначен для автономного отсчета временных установок, задаваемых по командам процессора.

Диапазон временных установок до 6,5 с с дискретностью 0,1 мс; нестабильность временных интервалов $\leq 5\%$.

Сообщение в процессор об истечении заказанной установки — по прерыванию или по адресной команде опроса программируемого таймера.

1.7. Технические характеристики адаптера канала (АК):

Адаптер канала предназначен для связи процессора УЧПУ с блоками сопряжения (БС) станка.

Состав сигнальных линий канала приведен в табл. 2.4.

Т а б л и ц а 2.4

Название линии или группы линий	Число линий в группе	Направление передачи
Выбор блока (ВБ)	16	П → БС
Под адрес (ПА)	8	П → БС
Запись (ЗП)	1	П → БС
Сброс (СБ)	1	П → БС
Прерывание принято (ПРП)	1	П → БС
Данные (Д)	16	П → БС
Готовность (ГТ)	1	БС → П
Прерывание — запрос (ПРЗ)	4	БС → П

Взаимодействие П с БС по командам (запись из П в БС, чтение в П из БС) и по прерыванию от БС.

Сигнал СБ — низкий уровень на линии СБ в течение 10 мкс.

Параметры сигналов канала должны соответствовать схемам ТТЛ-типа.

1.8. Технические характеристики контроллера измерительных преобразователей (КИП приводов, шпинделя, штурвала):

Сквозимость информационных сигналов $2 \pm 0,2$; длительность фронтов ≤ 1 мкс; максимальная частота следования импульсов (при максимальной скорости перемещения) 63 кГц; импульс «Начало отсчета» появляется один раз за оборот шпинделя.

Уровень сигналов на входе КИП (при нагрузке 150 Ом) не более 1,5 В («0») и не менее 11,5 В («1»).

На входе КИП сигнальные цепи должны быть гальванически развязаны от остальной части УЧПУ.

1.9. Технические характеристики контроллера приводов (КП):

Код установки (11 бит и знак) преобразуется в аналоговый управляющий сигнал, поступающий на силовую часть привода.

$U_{\text{вых}} = -10 \text{ В} - +10 \text{ В}$ при $R_n \geq 2 \text{ кОм}$. Максимально допустимая нестабильность 2,5 мВ, максимально допустимая пульсация 2,5 мВ.

ЦАП по входам и его блоки питания должны быть гальванически развязаны от остальной части УЧПУ.

1.10. Технические характеристики контроллера электроавтоматики станка (КЭ): КЭ предназначен для переключения исполнительных реле станка и приема сигналов состояния контактов реле и кнопок.

КЭ обеспечивает 16 выходных и 16 входных линий.

$I_{\text{вых}} = 0,2 \text{ А}$ при $U_{\text{нп}} = 24 \text{ В}$ по каждому выходу и $I_{\text{вх}} = 0,1 \text{ А}$ при $U_{\text{нп}} = 24 \text{ В}$ по каждому входу.

КЭ гальванически должен быть развязан со станком.

1.11. Технические характеристики блока питания:

Источник первичного питания — однофазная сеть переменного тока 220 В $\pm 10\%$, (50 ± 1) Гц. Предусмотреть защиту, отключающую УЧПУ от сети при перегрузках.

Предусмотреть несетевое питание, обеспечивающее сохранение информации в памяти в течение не менее 100 ч.

На сетевом входе блоков питания необходимо установить фильтры для защиты от импульсных помех амплитудой до 200 % амплитудного значения напряжения длительностью до 1 мс и времени нарастания от 0,5 до 500 мкс.

2. Требования к конструкции УЧПУ.

2.1. Конструктивно УЧПУ должен состоять из блока первичного питания и основного блока, которые соединяются между собой и с внешними устройствами кабелями с разъемами.

2.2. Блок первичного питания должен размещаться в вентилируемом отсеке электрошкафа, защищенном от попадания стружек, эмульсии, масла. Ориентировочные габариты блока 400 $\times$ 200 $\times$ 200 мм, рассеиваемая мощность 100 Вт.

2.3. Основной блок должен размещаться в специальном вентилируемом отсеке фартука, защищенном от попадания стружек, эмульсии, масла, пыли. Ориентировочные габариты блока 400 $\times$ 260 $\times$ 325 мм, масса 40 кг, рассеиваемая мощность 100 Вт.

2.4. Конструкция лицевой панели основного блока и способ его применения должны обеспечивать защиту клавиш и индикаторов панели и отсека фартука от попадания эмульсии, масла, пыли.

2.5. Требования к органам управления и индикации по ГОСТ 12.2.007.0—75.

3. Требования к надежности.

3.1. Нарботка на отказ $T_0 \geq 2000$ ч.

3.2. Нарботка на индицируемый сбой ≥ 50 ч.

3.3. Среднее время восстановления $t_n \leq 2$ ч.

3.4. Средний срок службы ≥ 10 лет.

4. Условия эксплуатации.

4.1. Температура окружающего воздуха от 5 до 45 °С.

4.2. Относительная влажность $\leq 80\%$ при 30 °С.

4.3. Вибрация частот до 25 Гц с двойной амплитудой до 0,1 мм.

5. Требования к транспортировке и хранению.

5.1. Допускается перевозка всеми видами транспорта при условии защиты по 2-й группе ГОСТ 15150—69.

5.2. Условия хранения по группе Л (ГОСТ 15150—69).

6. Токарный станок с системой ЧПУ должен обеспечивать: рост производительности токарной обработки мелких серий; улучшение условий труда; уменьшение нагрузки на службы подготовки программы ЧПУ.

7. Ориентировочная цена УЧПУ на второй год освоения не более 10 тыс. руб.

2.3. Архитектура микропроцессорных информационно-управляющих вычислительных систем

Архитектура аднопроцессорных ИУВС. В настоящее время аднопроцессорные ИУВС составляют наиболее многочисленный класс систем. В общем случае архитектура аднопроцессорной ИУВС имеет вид, представленный на рис. 2.16, где СУО — сигналы управления обменом. В ее состав входят: микроЭВМ; устройство сопряжения с объектом УСО; пульта управления ПУ₁, ..., ПУ_к с системами отображения

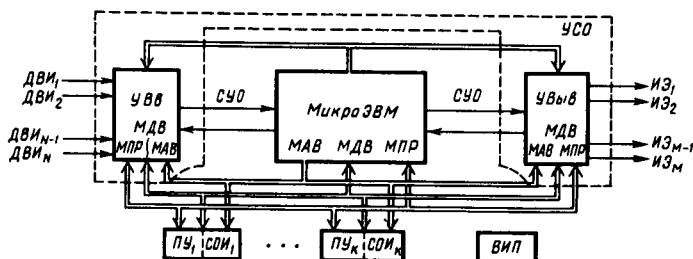


Рис. 2.16. Структура аднопроцессорной ИУВС.

информации СОИ₁, ..., СОИ_к; вторичный источник питания ВИП. Иногда принадлежностью системы являются также датчики входной информации ДВИ_и и исполнительные элементы ИЭ₁. В составе УСО имеются устройства ввода УВВ и вывода УВыв информации.

МикроЭВМ выполняет заданные программой преобразования информации и управляет работой всех составных частей ИУВС. В процессе работы микроЭВМ опрашивает состояние органов управления и задания режимов (тумблеры, переключатели и т. д.) на пультах управления ПУ_и и в зависимости от их положения реализует тот или иной режим функционирования системы. Взаимодействие микроЭВМ с УСО_и и ПУ_и осуществляется по внешним магистралям адреса МАВ, данных МДВ, прерывания МПР. Для опроса требуемой в данный момент времени информации на магистрали МА устанавливается адрес (номер) источника информации. Такими источниками могут быть тумблер ПУ, канал ввода и вывода УСО, СОИ и т. д. В источнике информации определяется соответствие его номера адресу и обеспечивается выдача информации по магистрали МДВ. В этом случае инициатором обмена является микроЭВМ. С целью уменьшения времени реакции системы на изменения состояния объекта управления УСО_и, ПУ_и и СОИ_и могут формировать запросы на прерывание работы в текущем режиме и переход к обработке новой информации. Для этого в ИУВС предусматривается магистраль прерывания МПР.

Устройство ввода УВв выполняет функцию приведения информации, полученной от ДВИ_и, к виду, удобному для использования в микроЭВМ. Как правило, микроЭВМ использует информацию, представленную в двоичном коде: параллельном; последовательном; параллельно-последовательном.

Информация от ДВИ_и имеет весьма разнообразный характер и может содержать ошибки измерений и быть искаженной помехами в линиях связи. Поэтому УВв составляют в большинстве ИУВС значительную не только по объему, но и по ее роли в системе часть. Аналогичные задачи преобразования, но в обратной последовательности выполняет устройство вывода информации УВыв. В ряде случаев оно преобразует информацию, полученную от микроЭВМ, не только к виду, удобному для исполнительных элементов ИЭ_и, но и обеспечивает усиление полученных сигналов.

С точки зрения пользователя ИУВС представляет собой совокупность тем или иным способом конструктивно выполненных и установленных на объекте блоков обработки информации, которые всегда выполняют ожидаемую и объявленную в технических условиях функцию. Единственным средством общения пользователя с системой и посредством ее с объектом управления являются ПУ_и и СОИ_и.

Пульт управления обеспечивает перевод ИУВС из одного режима работы в другой, ручной ввод числовой информации, покомандное исполнение отдельных операций ИУВС и т. д. Необходимые сведения о состоянии системы и объекта управления выводятся на СОИ_и.

В настоящее время *системы отображения информации* весьма развиты и имеют в своем составе электронно-лучевые трубки, газоразрядные индикационные панели, цифровые и буквенно-цифровые индикаторы, транспаранты с подсветом и т. д. Состав СОИ определяется задачами ИУВС, объемом и характером выводимой одновременно для отображения информации.

Вторичный источник питания ВИП, кроме главной функциональной задачи — обеспечения потребителей (микроЭВМ, ПУ_и, СОИ_и, УСО_и) необходимым питанием (по номенклатуре, величине и нагрузке), — выполняет не менее важную роль фильтра помех, возникающих в цепях первичной сети. Корректная разработка ВИП и правильная разводка цепей питания ИУВС обеспечивают стабильную работу в широком диапазоне температур и в условиях помех в первичных источниках питания.

Архитектура мультимикропроцессорных ИУВС. Для современного этапа развития ИУВС характерен переход от ИУВС с одним микропроцессором к *мультимикропроцессорным* ИУВС (*тМП ИУВС*), в которых сложнейшие задачи управления выполняются одновременно множеством микропроцессоров. Интерес к созданию управляющих и вычислительных систем, получаемых соединением в сложную структуру большого числа микропроцессоров, запоминающих устройств, устройств ввода-вывода информации, т. е. к *тМП ИУВС*, обусловлен [2, 19, 20]:

1) необходимостью создания высокопроизводительных ИУВС с эквивалентным быстроедействию в десятки и сотни миллионов

операций в секунду, например для анализа и прогнозирования погоды, распознавания образов, систем автоматизированного проектирования, решения задач обработки сигналов и сопровождения целей в военной технике и т. п.;

2) требованием обеспечения живучести ИУВС, работающих в режиме реального времени, т. е. необходимостью создания ИУВС, способных продолжить работу при выходе из строя отдельных их устройств;

3) территориальной удаленностью на десятки, сотни и тысячи метров элементов систем управления и сбора информации, включающих вычислительные средства;

4) необходимостью в большинстве случаев коммутации вычислительной мощности, ее динамическим перераспределением в зависимости от динамики решаемых задач;

5) необходимостью адаптации ИУВС к внешним условиям, а не наоборот;

6) требованием достижения гибкости вычислительных средств, т. е. возможности изменения конфигурации, расширяемости и модульности структуры;

7) потребностью в повышении производительности самих микропроцессоров, поскольку они не могут конкурировать со сверхбыстродействующими ЭВМ;

8) низкой стоимостью элементной базы (БИС микропроцессорных наборов, ОЗУ, ПЗУ, ППЗУ, УВВ), обладающей достаточно большими вычислительными возможностями.

Успех широкого применения *тМП* ИУВС связан с разработкой архитектуры (структурной организации) и системного программного обеспечения; с созданием параллельных алгоритмов решения задачи и соответствующего прикладного программного обеспечения.

Архитектуру параллельных вычислительных систем вообще можно классифицировать, например, по типу внутрисистемных связей, связанности процессоров, по тому, состоит ли система из однородных или неоднородных вычислительных средств, и т. п.

Одна из наиболее распространенных классификаций вычислительных систем в зависимости от взаимодействия потоков команд и данных в них приведена в табл. 2.5.

Рассмотрим архитектурные особенности систем типа MIMD. Системы типа MIMD, в которых отдельные ЭВМ, работая в основном под управлением своих операционных систем, могут одновременно выполнять различные части одной задачи и имеют собственные УВВ, называют *многомашинными*. Системы типа MIMD, в которых управление процессорами осуществляется одной операционной системой и используется общее ЗУ, независимо от наличия в процессорах местных ЗУ и устройств управления называют *мультипроцессорными*.

Архитектура *тМП* ИУВС в первую очередь должна удовлетворять трем основным свойствам: 1) параллельности; 2) однородности (модульности); 3) программной изменяемости (реконфигурации) структуры.

Параллельность структуры означает, что *тМП* ИУВС в целях

Параметры обрабатываемых вычислительной системой потоков		Условное обозначение	Особенности архитектуры вычислительной системы
команд	данных		
Один	Один	SISD	Однопроцессорная
Множество	»	MISD	Однопроцессорная с конвейерной обработкой (магистральная)
Один	Множество	SIMD	Многoproцессорная матричная (векторная, ассоциативная, ортого- нальная)
Множество	»	MIMD	Мультипроцессорная, многома- шинная

повышения производительности должна обеспечивать одновременное параллельное решение большого числа частей общей программы.

В соответствии со свойством *однородности структуры* и в целях обеспечения ее реконфигурации, ремонтпригодности, унификации *mMII* ИУВС должна строиться из набора идентичных базовых модулей, идентичным образом соединенных между собой.

Программная изменяемость структуры позволяет, задав программным способом структуру, организовывать из отдельных модулей системы, наиболее соответствующие требуемым вычислительным алгоритмам и требованиям живучести.

Перечисленным свойствам в различной степени отвечают известные и применяемые структуры мультимикропроцессорных ИУВС [23]: 1) с общим ЗУ (рис. 2.17, а); 2) с местными ЗУ (рис. 2.17, б); 3) с соединением модулей по полному графу (рис. 2.17, в); 4) радиальная (рис. 2.17, г); 5) шинная (рис. 2.17, д); 6) кольцевая (рис. 2.17, е); 7) с пирамидальной иерархической архитектурой (рис. 2.17, ж); 8) с регулярным объединением через коммутаторы (рис. 2.17, з); 9) с регулярным объединением через память (рис. 2.17, и). На этих рисунках MII_i — i -й микропроцессор, в общем случае включающий операционную часть, запоминающее устройство, устройство управления; $УВВ_j$ — j -е устройство ввода-вывода; $ЗУ_k$ — k -е запоминающее устройство; ЦК — центральный коммутатор; К — коммутатор; А — адаптер общей шины. Примеры структур *mMII* ИУВС не равнозначны и характеризуются конкретными достоинствами и недостатками. Наличие свойства у структуры отмечено «+», а его отсутствие — «—»:

Свойство <i>mMII</i> ИУВС	а	б	в	г	д	е	ж	з	и
Параллельность	+	+	+	+	+	+	+	+	+
Однородность	—	+	—	—	—	+	—	+	+
Программная изменяемость	+	+	+	+	+	—	+	+	+

Из таблицы видно, что свойствам параллельности, однородности, программной изменяемости удовлетворяют матричные структуры (см. рис. 2.7, в, з, и).

Примечание. Анализ особенностей таких структур посвящены работы Э. В. Евреинова, В. Г. Хорошевского, М. И. Соболевского [2].

Из-за большой сложности матричных структур и, как правило, территориальной удаленности элементов системы управления в настоящее время широко распространены *тМП ИУВС распределен-*

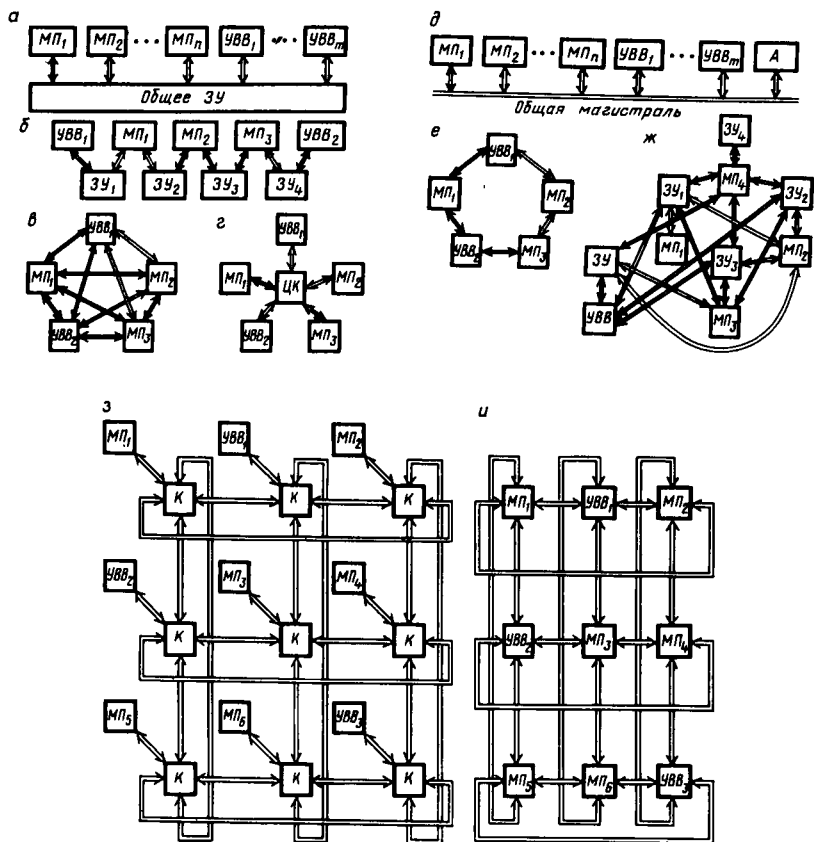


Рис. 2.17. Примеры структур мультимикропроцессорной ИУВС.

ного типа, к которым можно отнести, например, варианты систем с архитектурой, изображенной на рис. 2.17, а, б, г — ж, которые включают в себя в общем случае множество микропроцессоров МП, модулей запоминающих устройств ЗУ, модулей ввода-вывода информации УВВ. Каждый модуль выполняет специализированные функции в соответствии со своей индивидуальной программой, являющейся частью общей задачи системы. Здесь под микропроцессором понимается вычислительное устройство, выполняющее функции обработки, хранения и обмена информации и реализованное на одной БИС или в виде модуля, содержащего несколько БИС.

Связи между модулями мультимикропроцессорной ИУВС распределенного типа обычно реализуются через общую магистраль; общую память, организованную по принципу «почтового ящика»; многопортовую память; с помощью матричных переключателей П или через дельта-сеть С.

В случае связи модулей мультимикропроцессорной ИУВС *через общую магистраль* (рис. 2.17, *д*) последняя является распределенным ресурсом. Управление обменом информацией осуществляется в соответствии с принципами построения интерфейса и схем приема/передачи информации в каждом вычислительном модуле. Перед началом передачи информации активный модуль определяет готовность общей магистрали и далее готовность выбранного пассивного модуля к приему информации. Пассивный модуль опознает свой адрес и отвечает на сигнал управления активного модуля. Наличие общего ресурса — общей магистрали — приводит к конфликтам при обращении к ней нескольких активных модулей. Для разрешения этих конфликтов применяют следующие методы: синхронное разделение во времени и асинхронное разделение с приоритетами.

В случае связи *через общую память*, организованную по принципу «почтового ящика» (рис. 2.18, *а*), эта память представляет собой распределенные ресурсы данных. В одном из вариантов для каждого микропроцессорного модуля и (или) модуля ввода-вывода мультимикропроцессорной ИУВС в общей памяти отводятся фиксированные области: «почтовый ящик состояний» 2 и «почтовый

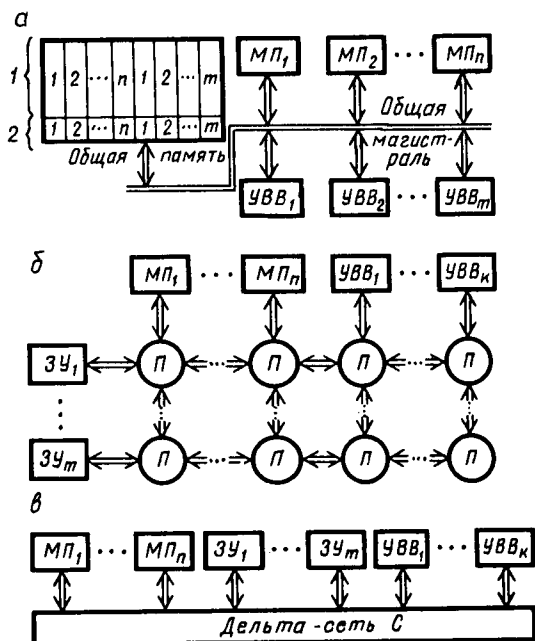


Рис. 2.18. Примеры связей между модулями мультимикропроцессорной ИУВС распределенного типа.

ящик сообщений» 1. Активный (передающий) модуль формирует пакет информации и сопровождающую ее квитанцию, помещает их соответственно в «почтовый ящик сообщений» 1 и «почтовый ящик состояний» 2. Далее пассивный (принимающий) модуль по мере готовности обращается к своему «почтовому ящику состояний» 2 и при обнаружении квитанции о наличии данных для него в соответствующем «почтовом ящике сообщений» 1 считывает пакет информации.

В другом варианте активный модуль помещает в «почтовый ящик состояний» 2 пассивного модуля (адресата) вместе с квитанцией также адрес начальной ячейки информационного массива общей памяти, в котором он размещает сообщение, и длину передаваемого массива. Если передаваемые активным модулем данные предназначены для нескольких пассивных модулей, то все пассивные модули должны быть извещены о том, где они могут получить данные, либо путем повторного помещения информации в «почтовые ящики сообщений» каждого пассивного модуля, либо путем указания в «почтовом ящике состояний» начального адреса и длины информационного сообщения. После завершения передачи информационного сообщения в общую память активный модуль генерирует сигнал прерывания всех остальных модулей *m*МП ИУВС на определенный фиксированный промежуток времени и освобождает общую магистраль *m*МП ИУВС. Все пассивные до этого момента времени модули, получив доступ к общей магистрали, например, последовательно опрашивают свои «почтовые ящики состояний» и, если последний содержит квитанцию об обмене информацией, либо обращаются к соответствующему «почтовому ящику сообщений», либо считывают из общей памяти информационный массив заданной длины начиная с заданного начального адреса в свою местную память. После этого стирают квитанцию в «почтовом ящике состояний».

Преимущество *m*МП ИУВС с общей памятью, организованной по принципу «почтового ящика», состоит в том, что пассивные модули (адресаты) получают данные без участия активного модуля (источника), т. е. обмен данными является в большей степени асинхронным, чем в системах с общей магистралью без общей памяти.

Устранение конфликтов в *m*МП ИУВС со связями между микропроцессорными модулями и модулями ввода — вывода через *многопортовую память* (см. рис. 2.17, а) достигается ценой усложнения общего ЗУ, обеспечивающего аппаратурный доступ к любой его ячейке со множества направлений. На практике построение ЗУ с числом портов более четырех представляет сложную техническую задачу.

При соединении модулей в *m*МП ИУВС с помощью *матричных переключателей* (рис. 2.18, б) связь между активным и пассивным модулями осуществляется с помощью матрицы соединительных шин и совокупности переключателей П, т. е. реализуется пространственное разделение сигналов. Конфликты в такой системе возникают лишь в тех случаях, когда несколько активных модулей одновремен-

но обращаются к одному пассивному модулю. Однако построение такой системы требует больших аппаратных затрат.

С точки зрения аппаратных затрат наиболее экономичным является применение специального класса коммутационных сетей, называемых *дельта-сетями* (рис. 2.18, в). Дельта-сеть $a^n \times b^n$ — сеть коммутации a^n k -разрядных входов в b^n k -разрядных выходов, построенная с использованием модулей $a \times b$. Каждый модуль $a \times b$ по сигналам управления может соединить каждый k -разрядный вход a с каждым k -разрядным выходом b . На рис. 2.19, а приведена упрощенная структура модуля 2×2 дельта-сети, где a_1, a_2 — входные k -разрядные двунаправленные магистрали; b_1, b_2 — выходные k -разрядные двунаправленные магистрали; Y — сигнал управления. Модуль 2×2 дельта-сети функционирует так, что при $Y=1$, например, магистраль a_1 может быть соединена с магистралью b_2 , при этом магистраль a_2 соединена с магистралью b_1 (рис. 2.19, б). Если сигнал управления $Y=0$, то магистраль a_1 соединена с магистралью b_1 ,

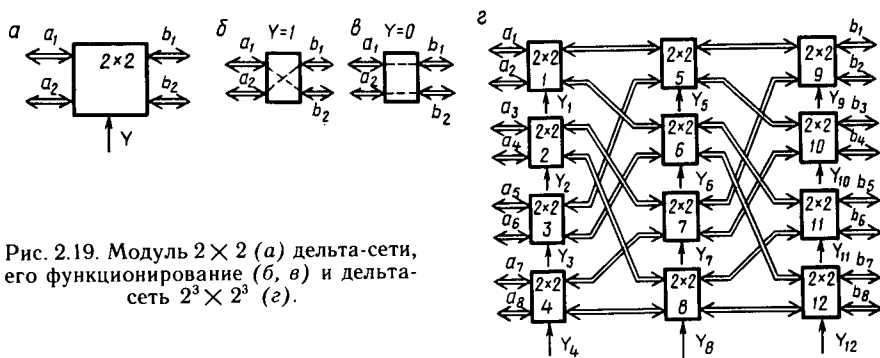


Рис. 2.19. Модуль 2×2 (а) дельта-сети, его функционирование (б, в) и дельта-сеть $2^3 \times 2^3$ (з).

а магистраль a_2 — с магистралью b_2 (рис. 2.19, в). Дельта-сеть $a^n \times b^n$ содержит n уровней модулей $a \times b$, причем связи между уровнями в дельта-сети организованы так, что от каждого входа a есть единственный путь к любому выходу b и длина всех путей одинакова. При этом возможна реализация дельта-сети $a^n \times b^n$ множеством способов. Каждый модуль $a \times b$, участвующий в соединении, управляется определенным разрядом кода управления, соответствующим коду связи определенной пары вход — выход. На практике модуль дельта-сети выполняет не только функции коммутатора, но и функции арбитра. Если по нескольким входам имеется запрос на подключение к одному и тому же выходу, то модуль обслуживает лишь один запрос в соответствии с системой приоритетов. В качестве примера на рис. 2.19, з приведена упрощенная структура дельта-сети $2^3 \times 2^3$ (8×8), построенная с использованием 12 модулей 2×2 , функционирующих в соответствии с изложенными выше принципами. Показано, что общее число модулей N_m типа $a \times b$ в дельта-сети $a^n \times b^n$ будет

$$N_m = \sum_{i=1}^{i=n} a^{n-i} b^{i-1} = \begin{cases} (a^n - b^n)/(a - b), & \text{если } a \neq b; \\ nb^{n-1}, & \text{если } a = b. \end{cases}$$

При построении конкретной m МП ИУВС распределенного типа разработчику приходится отвечать на следующие вопросы: система состоит из однородных или неоднородных вычислительных средств? многомерна или одномерна? с изменяемой или неизменяемой структурой? с централизованным или территориальным размещением? с совмещенными или несовмещенными обменами? с последовательной и (или) параллельной передачей информации?

Формулировка архитектурных особенностей m МП ИУВС и МП, служащего основой ее построения, в общем случае затруднительна в силу существенной зависимости последних от специфики решаемых задач и особенностей эксплуатации системы.

В частном случае, когда алгоритм, решаемый m МП ИУВС распределенного типа, допускает разбиение на относительно независимые части и к ИУВС предъявляется требование обеспечения высокой производительности (порядка нескольких миллионов операций в секунду) при обработке информации с высокой точностью, а также требование надежной работы в масштабе реального времени, архитектура возможного варианта процессорной части m МП ИУВС распределенного типа может иметь вид, приведенный на рис. 2.20, а. Система содержит $n + m$ одинаковых МП, из которых m резервных служат для повышения живучести системы, работающих в масштабе реального времени. В случае ненагруженного скользящего резервирования наработка на отказ $T_0 = \frac{m+1}{n} \cdot \frac{1}{\lambda}$, а зависимость вероятности безотказной работы от времени t выражается функцией

$$p(t) = [\exp(-n\lambda t)] \sum_{i=0}^{i=m} \frac{(n\lambda t)^i}{i!},$$

где λ — интенсивность отказов одного МП.

В системе для повышения скорости обмена между МП и уменьшения конфликтных ситуаций применены два параллельных асинхронных интерфейса. Один служит для передачи информации между МП массивами произвольной длины, другой — для передачи управляющей информации. К архитектуре МП высокопроизводительной резервированной m МП-системы распределенного типа, работающей в реальном масштабе времени, предъявляется ряд специфических требований: 1) обмен информацией между МП должен происходить независимо от выполнения вычислительных операций, другими словами, каждый МП должен включать в себя процессор обмена, работающий асинхронно с арифметическим процессором; 2) единство адрес-

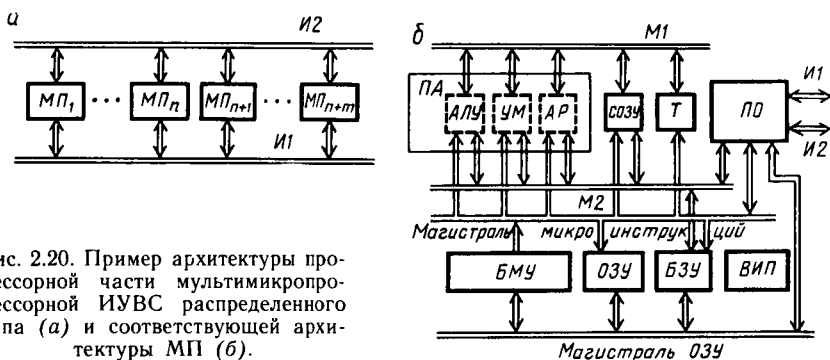


Рис. 2.20. Пример архитектуры процессорной части мультимикропроцессорной ИУВС распределенного типа (а) и соответствующей архитектуры МП (б).

ного пространства для всей m МП ИУВС и соответствующая разрядность команды; 3) наличие оперативной памяти в МП; 4) наращиваемость вычислительной мощности системы подключением дополнительных МП к m МП-системе; 5) возможность адресации по физическим и математическим адресам; 6) дистанционное управление МП; 7) наличие развитой системы прерываний; 8) возможность расширения системы команд; 9) наличие системы отсчета времени; 10) требование развитой системы адресации (прямой, индексно-базовой); 11) малая интенсивность отказов на уровне $10^{-3} - 10^{-5}$ ч $^{-1}$; 12) стоимость на уровне нескольких тысяч рублей; 13) модульность конструктивного исполнения.

На рис. 2.20, б приведен вариант архитектуры МП на базе микропроцессорного набора серии КР1802 и БИС ЗУ серии К565РУЗ, обеспечивающий построение m МП ИУВС распределенного типа с эквивалентным быстродействием в несколько миллионов операций в секунду при обработке 32-разрядной информации.

Микропроцессор включает в себя процессор арифметический (ПА), процессор обмена информации (ПО), сверхоперативное (СОЗУ), буферное (БЗУ) и оперативное (ОЗУ) запоминающие устройства, блок микропрограммного управления (БМУ), таймер, вторичный источник питания (ВИП). Арифметический процессор в свою очередь включает 32-разрядное арифметическо-логическое устройство (АЛУ) на БИС КР1802ВС1, 32-разрядный арифметический расширитель (АР) для выполнения сдвигов на БИС КР1802 ВР1, 32-разрядный умножитель-делитель (УМ) на БИС КР1802 ВР2. СОЗУ информационной емкостью тридцать два 32-разрядных слова, выполняющие роль регистров общего и специального назначения, и БЗУ информационной емкостью шестнадцать 32-разрядных слова, обеспечивающее конвейерность обработки информации, реализуется на БИС КР1802 ИР1. Информационная емкость ОЗУ для хранения программ и данных на БИС К565РУЗ составляет 16К 32-разрядных слов. Память микрокоманд БМУ 1К 72-разрядных слова на БИС К565РЕ5.

Передача информации между МП по интерфейсам группового и быстрого обмена осуществляется процессорами со скоростью 16—32 Мбод асинхронно с работой арифметических процессоров. По мере готовности арифметический процессор i -го МП организует множество блоков данных БД $_i$ в своем ОЗУ и одновременно заносит множество признаков P_{ij} в свой процессор обмена, где $j=1, \dots, n$; $i \neq j$. Один из n микропроцессоров, выполняющий роль диспетчера в системе, периодически читает содержимое регистров признаков P_{ij} всех $(n-1)$ ПО по интерфейсу быстрого обмена. Он организует график обменов и назначает в любой текущий момент связь по интерфейсу группового обмена конкретной пары ОЗУ через соответствующие ПО k -го и l -го МП.

Примечание. В § 4.1 рассмотрен еще один пример архитектуры МП ИУВС, в частности архитектуры m МП ИУВС распределенного типа, предназначенной для аэрокосмических применений (см. рис. 4.6).

Последовательность разработки ИУВС может быть представлена в виде схемы, показанной на рис. 2.21.

Анализ ТЗ на разработку позволяет выявить и обосновать основные требования, предъявляемые к ИУВС, и ограничения, накладываемые на основные технические характеристики.

К требованиям, предъявляемым к ИУВС, относятся: количество и типы входных и выходных каналов; точность обработки информации и выработки выходных величин; максимально допустимое время запаздывания выработки информации по каждому каналу.

Перечень ограничений, которые необходимо учитывать при разработке ИУВС, связан с принципиально существующими пределами изменений технических характеристик ИУВС (если входная информация представлена в виде амплитудного значения напряжения переменного тока частотой 400 Гц, то принципиальная граница времени обновления данных составляет 2,5 мс).

Ограничения эксплуатационные, связанные с невозможностью применения тех или иных технических решений (например, невоз-

возможность использования перфорированных корпусов; требования установки внешних разъемов по условиям подключения ИУВС к конкретному объекту и т. д.).

Технологические ограничения, связанные с тем, что существующие уровни промышленного производства предприятий — поставщиков комплектующих элементов и предприятия — изготовителя ИУВС — не всегда позволяют использовать желаемые, но принципиально возможные элементы, достижимые технические решения и технологические приемы.

Анализ ТЗ позволяет обосновать требования к каждой составной части системы: микроЭВМ, устройству сопряжения с объектом, пульту управления, системам отображения информации и программному обеспечению.

В силу взаимовлияния параметров устройства сопряжения с объектом и микроЭВМ на окончательные характеристики ИУВС в целом на схеме (рис. 2.21) представлена только одна, но наиболее важная часть алгоритма их совместной разработки.

Особенностью разработки микропроцессорных ИУВС является

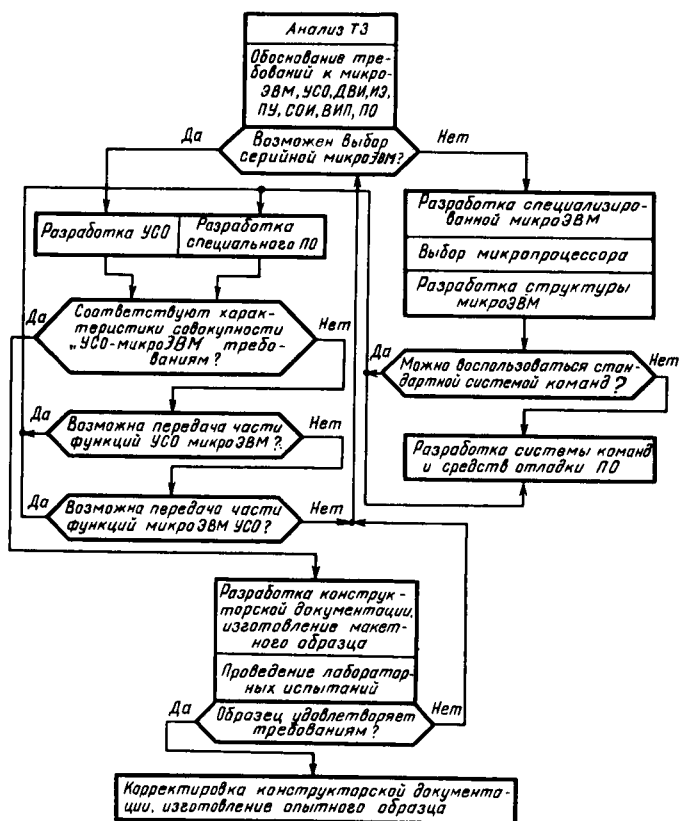


Рис. 2.21. Схема алгоритма разработки ИУВС.

возможность применения микроЭВМ, серийно выпускаемых промышленностью. Выбор микроЭВМ зависит от результата анализа на соответствие ее параметров выработанным требованиям в условиях заданных ограничений. При известных характеристиках микроЭВМ и с учетом предъявленных требований разрабатывается устройство сопряжения с объектом (УСО) и одновременно — специальное программное обеспечение. Составляются рабочие программы по наиболее ответственному и критическому с точки зрения времени и точности реализации алгоритмам.

Оценочные характеристики устройства сопряжения с объектом и микроЭВМ позволяют сделать вывод о возможности их совместного использования в ИУВС. Если полученный вариант не обеспечивает выполнения ТЗ, то анализируются возможности передачи части функций УСО микроЭВМ и наоборот. Здесь возможны не аппаратурные (в УСО), а программные методы преобразования входных величин; организация программного контроля состояния УСО; использование аппаратурных методов сглаживания входной информации.

На каждом этапе взаимной передачи функций осуществляется переработка УСО и специального программного обеспечения с последующей оценкой результатов. Если результат оказывается отрицательным, то выбирают другую микроЭВМ из перечня серийно выпускаемых и все действия повторяют.

При невозможности выбора серийно выпускаемой микроЭВМ возникает проблема разработки специализированной микроЭВМ для данной ИУВС. При этом должны быть решены такие основные задачи, как выбор типа микропроцессора, выбор и разработка структуры микроЭВМ.

Проблема выбора типа микропроцессора достаточно хорошо освещена в [21, 22].

Решение задачи выбора и разработки структуры микроЭВМ ставит вопрос и о системе команд. Наиболее рациональным, а для систем с преемственностью программного обеспечения ранее разработанных ИУВС обязательным является применение *стандартной системы команд* — системы команд серийно выпускаемых ЭВМ, принятых в качестве базовых (например, СМ-3, СМ-4, «Электроника-100/25» и т. д.), или системы команд разработанных ранее специализированных микроЭВМ для данного класса ИУВС. Выбор стандартной системы команд (см. § 2.2) позволяет использовать типовые средства отладки программного обеспечения и приступить к разработке УСО.

Если выбрана специальная система команд, ориентированная на конкретный класс задач, то далее разрабатываются микропрограммы выполнения команд и средства отладки программного обеспечения (различные кросс-средства, оценочные модули и аппаратурно-программные отладочные комплексы).

Положительное решение всех указанных вопросов приводит к разработке конструкторской документации, изготовлению макетного образца и проведению лабораторных испытаний с целью опреде-

ления его соответствия требованиям ТЗ. Если образец не удовлетворяет требованиям, то решается вопрос о глубине его переработки. В случае больших объемов работ осуществляется возврат к предыдущим этапам, а при малом объеме доработок или при соответствии требованиям ТЗ корректируется конструкторская документация и изготавливается опытный образец.

Аналогичные этапы существуют и при разработке пульта управления (ПУ) и систем отображения информации, однако в силу их большей независимости этот процесс, как правило, протекает параллельно разработке УСО и микроЭВМ.

2.4. Особенности проектирования информационно-управляющих вычислительных систем на микроЭВМ со специализированной системой команд*

Структура МП ИУВС. Рассмотрим пример однопроцессорной ИУВС, представляющий собой один из возможных вариантов ее реализации. В состав системы входят (рис. 2.22): микроЭВМ; адаптированные датчики входной информации ДВИ₁, ..., ДВИ_М; первичные

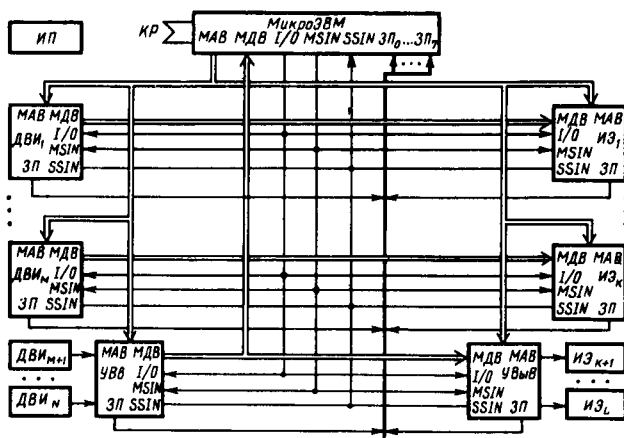


Рис. 2.22. Структура МП ИУВС.

датчики входной информации ДВИ_{м+1}, ..., ДВИ_Н; устройство ввода информации УВВ; адаптированные исполнительные элементы ИЭ₁, ..., ИЭ_К; исполнительные элементы ИЭ_{к+1}, ..., ИЭ_Л, требующие согласования с микроЭВМ; устройство вывода информации УВВ; источник питания ИП.

МикроЭВМ на микропроцессорном комплексе повышенного быстродействия серии К589 [17, 21] является основой ИУВС. Требование выполнения микроЭВМ функций управления обуславливает

* § 2.4 написан авторами по материалам разработки, выполненной совместно с В. В. Пысиным и Ю. А. Новокрещеновым.

необходимость разработки развитой и стандартизованной, хотя бы в рамках одной или нескольких ИУВС, системы обмена информацией. В данной микроЭВМ обмен происходит по 16-разрядной двунаправленной внешней магистрали данных МДВ с указанием внешнего устройства по 16-разрядной однонаправленной магистрали адреса МАВ. Направление передачи информации (ввод — вывод) определяется состоянием выхода I/O. При I/O = «1» осуществляется ввод информации в микроЭВМ, при I/O = «0» — вывод.

Для обеспечения асинхронного режима обмена информацией с учетом различного времени реакции внешних устройств микроЭВМ вырабатывается через 0,5 мкс после установки адреса сигнал I/O и через 0,7 мкс — сигнал начала обмена MSIN. Наличие этого сигнала соответствует состоянию готовности ЭВМ к вводу — выводу информации (рис. 2.23). В течение 3,2 мкс ЭВМ находится

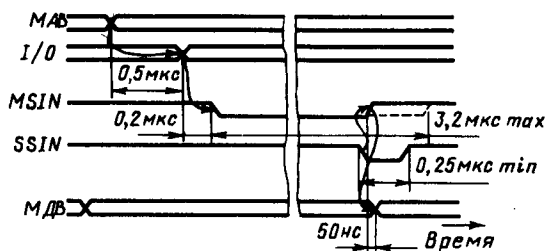


Рис. 2.23. Временная диаграмма обмена информацией микроЭВМ с внешними устройствами.

в состоянии ожидания ответного сигнала SSIN от внешнего устройства о приеме (передаче) информации. При поступлении сигнала SSIN в течение 3,2 мкс ЭВМ снимает сигнал MSIN и переходит к программе обработки информации от внешнего устройства. Отсутствие сигнала SSIN или его появление в более поздний момент времени оценивается ЭВМ как неготовность внешнего устройства к обмену.

Кроме обмена информацией по инициативе ЭВМ в системе, существует возможность ввода — вывода информации и перехода к той или иной программе обработки по запросам внешних устройств. Для этого в ЭВМ имеется восемь уровней прерывания, на входы которых поступают запросы на прерывание ЗП₀, ..., ЗП₇ от датчиков входной информации, исполнительных элементов, устройств ввода и вывода.

Использование в ИУВС адаптированных датчиков входной информации и исполнительных элементов, непосредственно связанных с системой обмена микроЭВМ, т. е. ДВИ₁, ..., ДВИ_М и ИЭ₁, ..., ИЭ_К, наиболее целесообразно. Этим достигается гибкость системы по отношению к условиям ее применения, относительная простота замены существующих и введения новых типов датчиков и исполнительных элементов.

В случае принципиальной конструктивной невозможности (по условиям размещения, больших механических нагрузок, низкой помехоустойчивости и т. д.) использования адаптированных внешних устройств применяются *первичные датчики входной информации ДВИ_{м+1}, ..., ДВИ_н и исполнительные элементы ИЭ_{к+1} и ИЭ_л*.

Для взаимодействия с микроЭВМ разработаны специализированные (для конкретных типов ДВИ и ИЭ) устройства ввода и вывода, снабженные адаптерами принятой системы обмена. К такого типа датчикам относятся вращающиеся трансформаторы, контактные переключатели, датчики временных интервалов и т. д., соответственно исполнительные элементы — электромагниты, электродвигатели, гидронасосы и т. д. [10].

При разработке ИУВС к микроЭВМ предъявлялись следующие основные требования (полученные в соответствии с методиками, изложенными в гл. 1):

- 1) разрядность операционной части — 16 двоичных разрядов;
- 2) представление чисел: двоичные без знака — для адресов ОЗУ, ПЗУ и внешних устройств; двоичные целые со знаком — для представления порядков; числа и типов сдвигов; двоичные с фиксированной точкой; двоичные с плавающей точкой;
- 3) система команд: команды обмена информацией с внешними устройствами; команды условных и безусловных переходов; команды арифметических и логических операций; команды обмена между регистрами процессора, ячейками оперативной и постоянной памяти;
- 4) время выполнения операций: типа сложения ≤ 1 мкс; типа умножения ≤ 25 мкс;
- 5) число уровней прерывания — восемь;
- 6) емкость ОЗУ — 2048 слов по 16 двоичных разрядов;
- 7) емкость ПЗУ — не менее 16К слов;
- 8) емкость электрически перепрограммируемого ЗУ (ЭППЗУ) с сохранением информации при отключении питания — 256 слов по 16 двоичных разрядов;
- 9) таймер с частотой заполнения 1 кГц, позволяющий вести счет отрезков времени от 1 до 65 535 мс.

Структура микроЭВМ МП ИУВС. Структура микроЭВМ ИУВС, изображенная на рис. 2.24, реализует требования по функциональному составу микроЭВМ с учетом ее построения на основе микропроцессорного комплекта серии К589. Здесь ядром является *центральный процессор ЦП*, который выполняет арифметические и логические операции над 16-разрядными словами. Входная информация поступает по магистрали М, связанной с запоминающими устройствами, и по магистрали В, связанной со схемами обмена. Управление работой ЦП осуществляется заданием кода по магистрали F, констант по магистрали К и сигнала ФВ по входу С₁ о значении переноса и сдвига вправо. На выходе ЦП формируются 16-разрядные адрес А и данные Д.

МикроЭВМ содержит *запоминающие устройства* трех типов: ПЗУ, ОЗУ и ЭППЗУ. Постоянное запоминающее устройство ПЗУ предназначено для хранения команд и констант. Структурно разряды

слов, считываемых из ПЗУ, делятся на код операции и адресную часть команды (или константу) (рис. 2.25). Постоянное запоминающее устройство ПЗУ по магистралям МА и М, а оперативное ОЗУ — и по магистрали МД непосредственно связаны с ЦП (см. рис. 2.24).

В отличие от ОЗУ и ПЗУ электрически перепрограммируемое ЗУ ЭППЗУ является внешним устройством по отношению к магистралям ЦП. Обмен информацией происходит по внешней двунаправ-

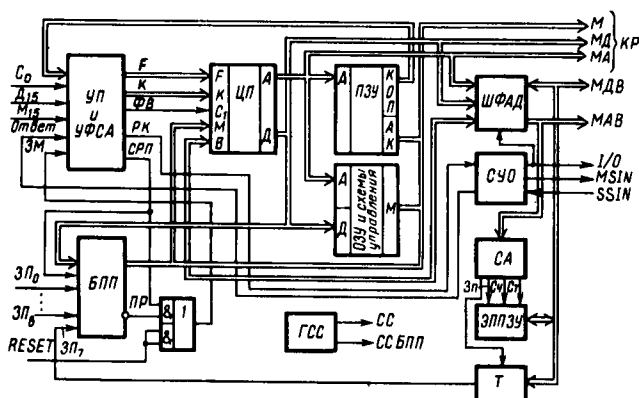


Рис. 2.24. Структура микроЭВМ МП ИУВС.

ленной магистрали данных МДВ через шинные формирователи адреса и данных ШФАД. Запоминающее устройство ЭППЗУ работает в трех режимах: записи Зп, считывания Сч, стирания Ст. Режим работы задается селектором адреса СА по значению кода на внешней адресной магистрали МАВ. Кроме того, селектор СА вырабатывает сигнал записи в таймер Т кода временного интервала, установ-



Рис. 2.25. Формат команды.

ленного на внешней магистрали МДВ. При считывании кода до нулевого значения таймер Т формирует по седьмому уровню запрос на прерывание.

Обмен информацией по внешним магистралям сопровождается выработкой *схемы управления обменом* СУО сигналов I/O, MSIN, SSIN. В зависимости от значения сигнала I/O шинные формирователи передают информацию с магистрали МД на магистраль МДВ при I/O = «0» и с магистрали МДВ на магистраль В при I/O = «1». Запуск схемы СУО осуществляется сигналом РК. Временная диаг-

рамма работы схемы показана на рис. 2.23; по завершении цикла обмена формируется сигнал «Ответ».

Блок приоритетных прерываний БПП по стробу разрешения прерывания СРП сравнивает значение текущего приоритета с запросами ЗП₀, ..., ЗП₇ и формирует сигнал прерывания ПР. Запись текущего приоритета происходит по магистрали М центрального процессора. При наличии сигнала прерывания процессор ЦП переходит на программу обработки прерывания, начальный адрес которой задается блоком БПП по магистрали М.

Режимы работы всех устройств микроЭВМ, тип выполняемых ими операций в каждый момент времени (такт) определяются состоянием разрядов микрокоманд, формируемых *устройством микропрограммного управления (УМУ)*. В состав УМУ входят управляющая память УП и устройство формирования следующего адреса микрокоманды УФСА. Управляющая память УП хранит микропрограммы системы команд микроЭВМ. При появлении сигнала ЗМ — занесения кода операции следующей команды в УМУ *устройство формирования следующего адреса УФСА* по значению кода операции команды, считанной из ПЗУ, вырабатывает начальный адрес микропрограммы. Последовательность микрокоманд выполняется до тех пор, пока не будет выполнена данная команда. По окончании каждой микропрограммы формируется сигнал СРП, по которому опрашивается состояние БПП и при отсутствии прерывания вырабатывается сигнал ЗМ. При наличии сигнала прерывания происходит автоматический переход к микропрограмме входа в прерывание. При выполнении данной микропрограммы из БПП в ЦП считывается и выдается на магистраль МА начальный адрес подпрограммы обработки прерывания.

Разряды считываемых микрокоманд фиксированы и закреплены за конкретными управляющими входами всех устройств микроЭВМ. Среди основных можно выделить разряды и группы разрядов: F, K, ФВ, РК и т. д. Ветвление микропрограмм при выполнении команд условных и безусловных переходов осуществляется на основе анализа состояния отдельных разрядов магистралей микроЭВМ, «флажков» условий, некоторых внешних сигналов и т. д. В данной микроЭВМ наиболее часто используются сигналы С₀ — выход переноса ЦП; D₁₅, M₁₅ — старшие разряды магистралей данных и памяти; сигнал внешних устройств «Ответ» о завершении цикла обмена информацией.

Для пуска ЭВМ в начальный момент и периодического пуска в рабочем режиме, для устранения накапливающихся сбойных ситуаций и зацикливания используется сигнал RESET. По этому сигналу ОЗУ и ПЗУ переводятся в невыбранное состояние, осуществляется занесение нулевого кода операции и переход к нулевой, стартовой ячейке программной памяти. Работа устройств синхронизируется двумя последовательностями сигналов СС и СС БПП, которые вырабатывают *генератор синхросигналов ГСС*.

Состояние машины контролируется по внутренним шинам, выведенным на контрольный разъем КР.

Реализация отдельных устройств микроЭВМ МП ИУВС. Генератор синхросигналов (ГСС). Основу ГСС (рис. 2.26) составляет кварцевый генератор сигналов ГС. При значении сигнала останова $S = \langle 1 \rangle$ импульсы генератора поступают на вход C_1 регист-

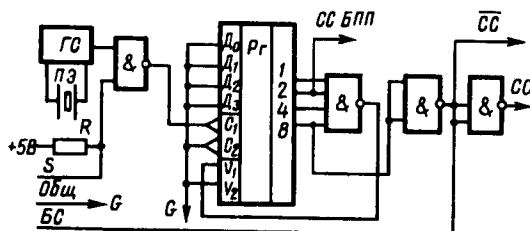
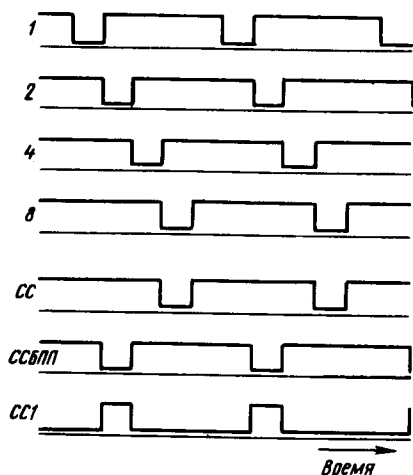


Рис. 2.26. Структура генератора синхросигналов.

ра R_g (например, K155ИР1). На выходах регистра формируется временная диаграмма, представленная на рис. 2.27. На выходах 1, 2, 4, 8 последовательно появляются сигналы нулевого потенциала (см. рис. 2.26). При этом на входе V_1 присутствует «1». После установ-



тановки всех разрядов регистра R_g в состояние «1» на выходе элемента 4И—НЕ формируется сигнал «0», по отрицательному перепаду импульса на выходе элемента 2И—НЕ выход 1 регистра R_g устанавливается в состояние «0» и цикл повторяется.

Последовательности сигналов CC , CC , CC БПП, используемые для синхронизации работы микроЭВМ, формируются на элементах

Рис. 2.27. Временная диаграмма работы генератора синхросигналов.

4И—НЕ и 2И—НЕ. Частота следования сигналов равна $f_r/5$, где f_r — рабочая частота кварцевого генератора. При $f_r = 20$ МГц частота следования синхросигналов 4 МГц.

Сигнал БС используется для блокировки синхросигналов ЦП при организации обмена с внешними устройствами. До завершения цикла обмена ЦП находится в режиме ожидания.

Центральный процессор. Центральный процессор (рис. 2.28) представляет собой совокупность восьми БИС центральных процессорных элементов ЦПЭ (K589ИК02), объединенных схемой ускоренного переноса СУП (K589ИК03). Входная информация

поступает в ЦП по двум 16-разрядным магистралям В и М. Выходная информация формируется на магистралях данных МД и адреса МА. Входы синхронизации объединены между собой и подключены к цепи генератора тактовых сигналов СС. Тип выполняемой ЦЭП операции задается по 7-разрядной магистрали управления ($F_0—F_6$), непосредственно связанный с управляющей памятью УП.

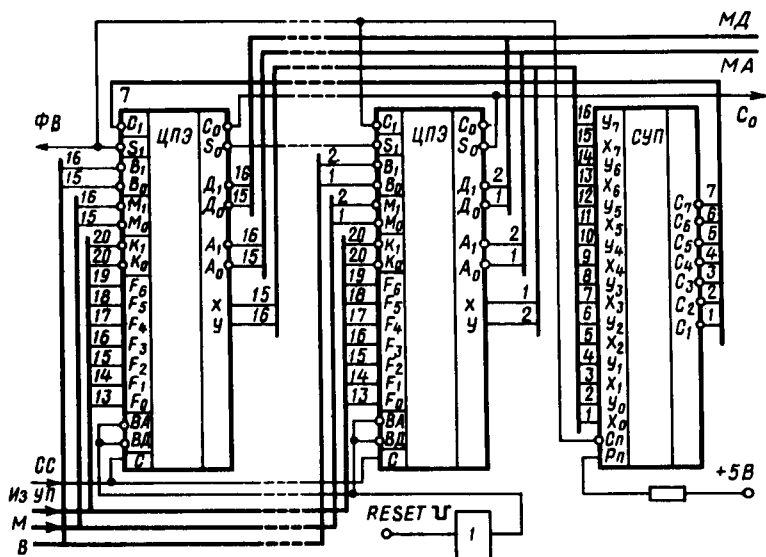


Рис. 2.28. Структура центрального процессора.

Особенностью использования ЦПЭ в данной микроЭВМ является одновременное задание во всех БИС только двух значений входов K_0 , K_1 маскирующей магистрали в виде «0» или «1». Таким образом, ЦП выполняет операции, имеющие mnemonic обозначения, указанные в [17]. К таким операциям относятся LMI, ALR и т. д.

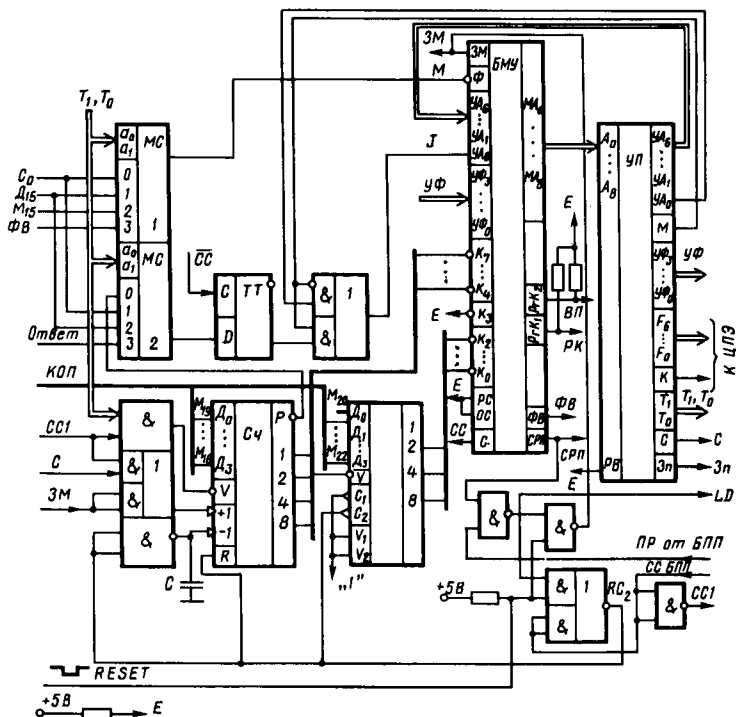
В силу того что арифметические операции и операция сдвига вправо выполняются в различные моменты времени, значение входного переноса и вход сдвига вправо задаются по одной цепи ФВ. Аналогично при организации условных переходов и сдвига вправо анализируется сигнал в объединенной цепи C_0 .

При организации начальной загрузки по сигналу RESET магистраль адреса МА переводится в третье состояние, обеспечивая выборку первой команды по нулевому адресу ПЗУ команд.

Устройство микропрограммного управления. Устройство микропрограммного управления (УМУ) предназначено для формирования последовательностей микрокоманд, которые реализуют микропрограммы выполнения команд, загружаемых из ПЗУ.

Основой УМУ является (рис. 2.29): блок микропрограммного

Наиболее характерные режимы работы УМУ следующие: загрузка кода операций, последовательная выборка микрокоманд; условная передача управления; организация перехода к обработке прерываний.



анной микроЭВМ для загрузки КОП, как правило, выполняется последовательность микрокоманд (рис. 2.30): Т — загрузка в регистр адреса РгА центрального процессора и установка в него значения адреса МА следующей команды $R_n \rightarrow \text{РгА}$; увеличение содержимого регистра R_n , используемого в качестве счетчика команд;

T+2 — загрузка К→Сч младших разрядов $K_4, \dots, K_7$ КОП в счетчик Сч (см. рис. 2.29). Загрузка регистра команд RgK_i , условный переход JPX по данным на входах $K_4, \dots, K_7$ БМУ [17].

214

серии СС (в предпоследней микрокоманде каждой микропрограммы) через время t_A в регистре адреса процессора, соответственно и на магистрали МА, фиксируется адрес следующей команды (рис. 2.30, 2.31). По истечении времени считывания t_n из ПЗУ программ на магистрали КОП, соответствующей разрядам $M_{16}, \dots, M_{22}$ (см. рис. 2.29) магистрали М, устанавливается код операции команды. Таким образом, код операции подготавливается для использования в УМУ в качестве начального адреса микропрограммы выполнения команд.

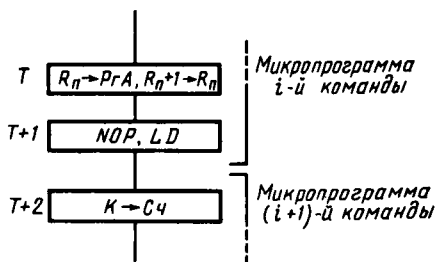


Рис. 2.30. Микропрограмма загрузки кода операции.

ды. Однако особенностью рассматриваемого УМУ является двух-этапный вход в микропрограмму. На первом этапе вход в микропрограмму определяется старшими разрядами КОП $M_{20}, \dots, M_{22}$ ($K_2, \dots, K_0$), а на втором доопределяется младшими разрядами $M_{19}, \dots, M_{16}$ ($K_7, \dots, K_4$). При этом по переходу 01 синхросерии СС через время установки t_m на выходах $MA_0, \dots, MA_8$ БМУ формируется адрес последней микрокоманды в микропрограмме. Из управляющей памяти через время $t_{уп}$ считывается микрокоманда, указывающая по входам управления $YA_6, \dots, YA_0$ переход JZR на колонку 15 двумерного массива адресов [17]. На выходе СРП БМУ формируется строб разрешения прерывания. При отсутствии сигнала прерывания $PR = \langle 1 \rangle$ от блока приоритетных прерываний БПП формируется сигнал ЗМ разрешения занесения кода операции в БМУ и сигнал RC_2 установки счетчика Сч и регистра Рг в исходное для данной команды состояние. По значению сигнала $RC_2 = 1$ через время t_R происходит установка счетчика Сч в состояние $\langle 0 \rangle$, а по переходу 10 — через время t_F в состояние $\langle 1 \rangle$. При этом на регистре

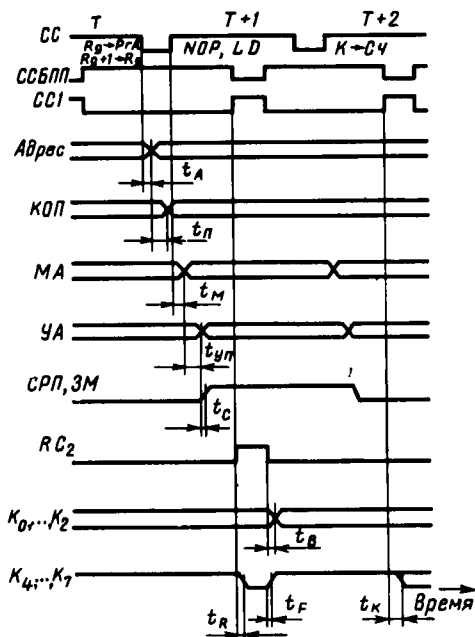


Рис. 2.31. Временная диаграмма загрузки кода операции.

адрес последней микрокоманды в микропрограмме. Из управляющей памяти через время $t_{уп}$ считывается микрокоманда, указывающая по входам управления $YA_6, \dots, YA_0$ переход JZR на колонку 15 двумерного массива адресов [17]. На выходе СРП БМУ формируется строб разрешения прерывания. При отсутствии сигнала прерывания $PR = \langle 1 \rangle$ от блока приоритетных прерываний БПП формируется сигнал ЗМ разрешения занесения кода операции в БМУ и сигнал RC_2 установки счетчика Сч и регистра Рг в исходное для данной команды состояние. По значению сигнала $RC_2 = 1$ через время t_R происходит установка счетчика Сч в состояние $\langle 0 \rangle$, а по переходу 10 — через время t_F в состояние $\langle 1 \rangle$. При этом на регистре

Рг фиксируются старшие разряды КОП. Следовательно, значение КОП, установленное на входах $K_0, \dots, K_7$ БМУ, определяет первые 8 строк двумерного массива адресов памяти УП [17]. Так как сигнал $ЗМ = \langle 1 \rangle$, то по переходу 01 синхросерии СС данные с выходов $K_0, \dots, K_7$ заносятся в регистр адреса БМУ и устанавливаются в качестве начального адреса микропрограммы на выходах $MA_0, \dots, MA_8$. Считывание следующей микрокоманды приводит к изменению значения управляющих входов $УА_6, \dots, УА_0$ и установке сигналов СРП и $ЗМ = \langle 0 \rangle$.

Для доопределения адреса микропрограммы в микрокоманде $K \rightarrow Cч$ разрядами T_1, T_0 разрешается занесение младшей части КОП в счетчик Сч при наличии $СС_1 = \langle 1 \rangle$. Дальнейшая работа сводится к выборке микрокоманд из УП при непосредственном формировании адреса следующей микрокоманды по входам $УА_6, \dots, УА_0$.

Условная передача управления реализуется с помощью входа признаков Ф блока БМУ и схемы дополнительного условного перехода (СДУП). В первом случае широко применяются типовые для БМУ переходы JFL, JCF, JZF. Во втором случае — любые переходы БМУ, причем в зависимости от состояния разряда М микрокоманды (выход М памяти УП) данными на входе $УА_0$ блока БМУ являются: при $M = \langle 0 \rangle$ — значение разряда $УА_0$, считанное из УП; при $M = \langle 1 \rangle$ — состояние триггера ТТ схемы СДУП.

В качестве признаков переходов используются: выходы переноса и сдвига вправо процессора C_0 ; значения старших разрядов магистралей данных и памяти D_{15} и M_{15} ; выход признаков ФВ; сигнал «Ответ» схемы управления обменом; выход переноса счетчика Сч.

Счетчик циклов применяется не только для хранения разрядов $K_4, \dots, K_7$ КОП, но и для организации счета в циклических микропрограммах. Для этого в микрокоманде выделен разряд С добавления единицы в счетчик циклов. Выбор конкретного признака определяется состоянием разрядов T_1, T_0 микрокоманды.

При сигнале прерывания $ПР = \langle 0 \rangle$ сигнал $ЗМ$ не вырабатывается, загрузка кода операции не происходит и при выполнении микрокоманды $T+1$ (рис. 2.30 и 2.31) переход JZR завершается входом в микропрограмму обработки прерываний.

По сигналу RESET начальной установки вырабатываются сигналы $ЗМ$ и RC_2 для занесения КОП команды, считанной из содержимого ячейки ПЗУ с нулевым адресом. В остальном УМУ не отличается от типовых УМУ, приведенных в [17].

Блок приоритетных прерываний (БПП). На рис. 2.32 представлен один из возможных вариантов использования БПП (серии К589ИК14) [17].

При этом входы уровня приоритета $П_0, \dots, П_2$ связаны со старшими разрядами магистрали процессора данных $D_{13}, \dots, D_{15}$, а выходы кода прерывания $КП_0, \dots, КП_2$ — с разрядами магистрали памяти $M_{13}, \dots, M_{15}$.

В случае если запрос на прерывание $ЗП_i$ выше уровня текущего приоритета, хранимого в БПП, при наличии сигнала СРП по переходу 01 сигнала СС БПП вырабатывается сигнал прерывания $ПР =$

= «0». Этим обеспечивается вход в микропрограмму обработки прерывания, основной задачей которой является переход к подпрограмме обработки прерывания по адресу кода прерывания КР₀, ..., КР₂. Для этого разряд микрокоманды, управляющий режимом

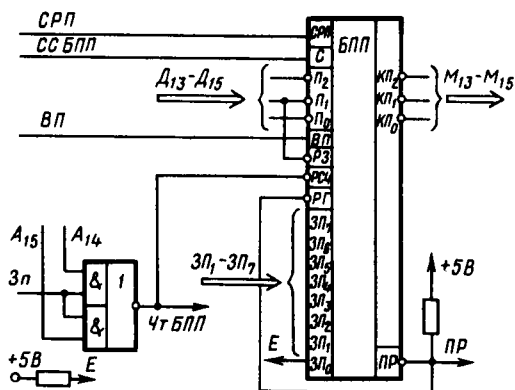


Рис. 2.32. Структура блока приоритетных прерываний.

«Запись/считывание» ОЗУ, устанавливается в состояние «Запись», а адрес на магистрали МА не принадлежит ОЗУ. Таким образом, разряды A_{15} , A_{14} имеют одно из возможных сочетаний 01, 10, 11. И при сигнале $3П = «1»$ формируется сигнал разрешения чтения блока прерываний Чт БПП. По этому сигналу на выходах КР₀, ..., КР₂ устанавливается код прерывания, который и используется процессором как адрес подпрограммы обработки прерывания.

Блок приоритетных прерываний БПП не реагирует на любые новые запросы и удерживает информацию в памяти запросов на прерывание до засылки уровня текущего приоритета. Для этого все подпрограммы обработки прерываний начинаются с команды входа в прерывание INP, в конце которой вырабатывается переход БМУ типа JCE и сигнал ВП (рис. 2.32).

Оперативное запоминающее устройство (ОЗУ). Оно содержит 32 микросхемы K134PY6, организованных в накопитель емкостью 2048 16-разрядных слов (рис. 2.33). Во всех микросхемах соответствующим образом объединены адресные входы А, входы данных D, цепи выборки кристалла CS, цепи записи W и информационные выходы M_{15} , ..., M_0 . Адресные входы для согласования по нагрузке через шинные формирователи ШФ подключены к магистрали адреса МА (A_0 , ..., A_9), а информационные входы — непосредственно к магистрали данных (D_0 — D_{15}) центрального процессора.

Обращение к ОЗУ обусловлено заданием адресов $(F8...)_{16}$, $(FC...)_{16}$ с учетом инверсии разрядов адреса в центральном процессоре. Например, должно быть задано:

а) $A_{15} A_{14} A_{13} A_{12} A_{11} A_{10} A_9 A_8 \dots$
 0 0 0 0 0 0 1 1

с учетом инверсии

1	1	1	1
---	---	---	---

1	1	0	0
---	---	---	---

(F)₁₆ (C)₁₆

б) $A_{15} A_{14} A_{13} A_{12} A_{11} A_{10} A_9 A_8 \dots$
 0 0 0 0 0 1 1 1

с учетом инверсии

1	1	1	1
---	---	---	---

1	0	0	0
---	---	---	---

(F)₁₆ (8)₁₆

При считывании информации установка адреса ОЗУ приводит к выборке соответствующей группы микросхем. Информация может быть считана назначением нужного адреса ячейки в младших разрядах $A_0, \dots, A_9$.

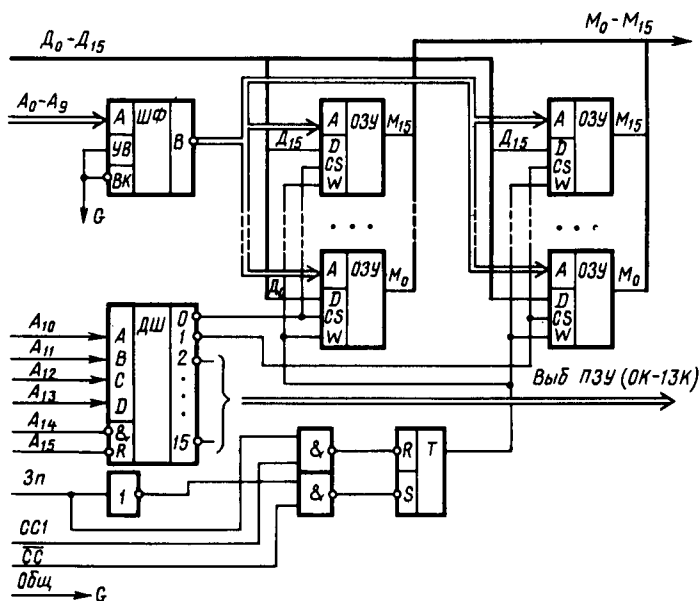


Рис. 2.33. Структура оперативного запоминающего устройства.

Режим записи иллюстрируется рис. 2.34. В отличие от считывания после установки адреса ОЗУ по переходу сигнала 01 серии СС считывается микрокоманда со значением $3n = \langle 1 \rangle$. По $3n \wedge CC1 = \langle 1 \rangle$ формируется начало сигнала записи W на триггере T , а по $3n \wedge \overline{CC} = \langle 1 \rangle$ — окончание.

Постоянное запоминающее устройство (ПЗУ). Имеет емкость 16 384 24-разрядных слов (рис. 2.35). Основу устройства составляют микросхемы БИС ПЗУ серии К556РТ5 с органи-

зацией 512 слов $\times$ 8 разрядов. Накопитель представляет собой 32 линейки по 3 БИС ПЗУ общей емкостью 0,5К слов. Выбор ячейки в пределах одной линейки обеспечивается заданием младших разрядов адреса $A_0, \dots, A_8$, выбор линейки — заданием пяти разрядов $A_9, \dots, A_{13}$. Особенность данного блока — динамическая выборка линеек ПЗУ не только по входам P_3, P_4 , БИС, но и по цепям

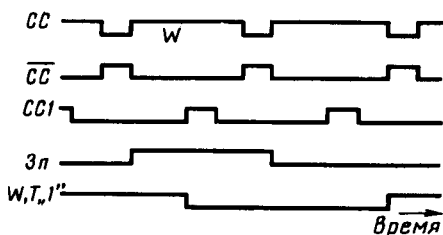


Рис. 2.34. Временная диаграмма работы ОЗУ.

питания U . Этим достигается уменьшение в 32 раза общей мощности потребления накопителя. С этой целью используются группа из 32 ключей K и дополнительная цепь питания $+5,5$ В.

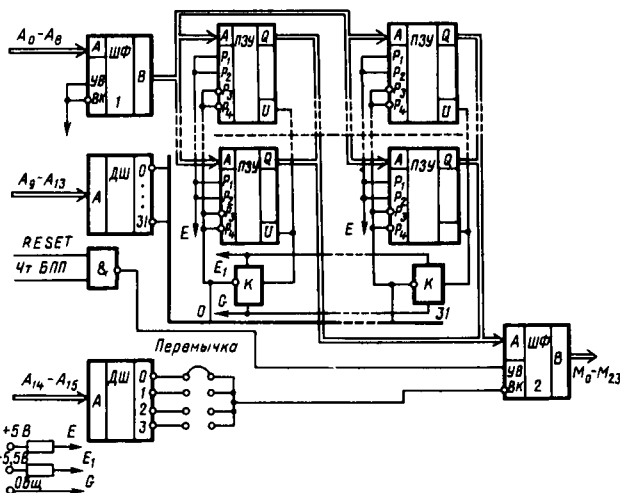
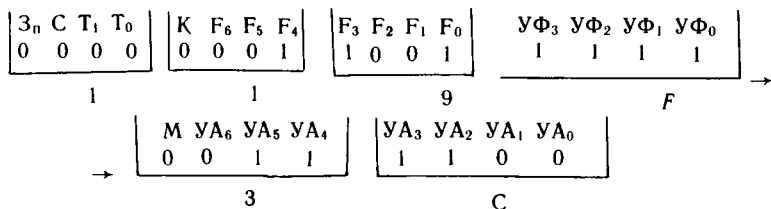


Рис. 2.35. Структура постоянного запоминающего устройства.

По значению старших разрядов A_{14}, A_{15} определяется соответствие адреса обращения номеру блока ПЗУ, установленного переключкой. При несоответствии адреса и номера выхода B формирователя ШФ₂ переводятся в третье состояние. Аналогично выходы B имеют высокое выходное сопротивление при начальной загрузке по сигналу RESET и по сигналу чтения блока прерываний Чт БПП.

Система команд микроЭВМ МП ИУВС. Система команд микроЭВМ ориентирована на решение типовых задач ИУВС и содержит 98 кодов операций, включая команды обмена информацией между регистрами процессора, ячейками ОЗУ и ПЗУ, внешними устройствами, команды арифметических и логических операций, условных

УМУ; разряды 12, ..., 18 ($F_0, \dots, F_6$) задают тип операции, источники и приемник результата центрального процессора; разряд 19 (K) — состояние магистрали K центрального процессора; разряды 20, 21 (T_0, T_1) управляют выбором признаков условий мультиплексорами $MC1$ и $MC2$ УМУ; разряд 22 (C) добавляет единицу в счетчик циклов $Cч$ УМУ; разряд 23 ($Зп$) управляет режимом работы ОЗУ (запись/считывание). Все микрокоманды записываются в виде 6 шестнадцатиричных чисел. Например, запись 019F3C расшифровывается так:



Состояние разрядов микрокоманды позволяет сделать следующие выводы:

устройство микропрограммного управления должно обеспечить немодифицированный ($M=0$) переход в текущей строке JCR [17] (так как $УA_6 = \langle 0 \rangle$; $УA_5 = \langle 1 \rangle$; $УA_4 = \langle 1 \rangle$) в колонку с номером $(12)_{10} = (C)_{16}$;

состояние $УФ_0, \dots, УФ_3 = \langle 1 \rangle$ соответствует хранению БМУ флажков C и Z и выдаче на выход ФВ $\langle 1 \rangle$, т. е. подаче на вход сдвига и переноса центрального процессора единицы;

значения разрядов K и $F_0, \dots, F_6$ указывают на требования выполнения центральным процессором операции LMI с регистром R_9 , т. е. $R_9 \rightarrow PгA$; $R_9 + 1 \rightarrow R_9$;

$T_1 = T_0 = 0$ — выбираются признаки по нулевым адресам мультиплексоров MC_1 и MC_2 , но они не используются;

$C = \langle 0 \rangle$ — единица в счетчик циклов $Cч$ УМУ не добавляется;

$Зп = \langle 0 \rangle$ — ОЗУ находится в режиме считывания.

Организация обмена информацией микроЭВМ МП ИУВС с внешними устройствами. Для организации обмена с внешними устройствами используются программно-аппаратурные средства (см. рис. 2.29 и 2.38). Рассмотрим работу схемы (рис. 2.38) при выполнении команд вывода и ввода по адресу ADDR. Микропрограмма вывода содержит микрокоманды в ячейках: 050, 051, 055 и 045 (рис. 2.39).

В y в о.д. 1. При занесении кода операции команды вывода, т. е. при выполнении микрокоманды NOP, LD предыдущей команды (рис. 2.39), счетчик $Cч$ устанавливается в нулевое состояние и по значению младшего разряда M_{16} КОП формирует сигнал управления выводом $УВ = 0$.

2. При выполнении микрокоманды, записанной в ячейке 050, ЦП реализует операцию LMM для регистра T , т. е. $M \rightarrow PгA$, T . Таким образом, адрес внешнего устройства ($ВУ$) устанавливается на магистрали MAB. Устройство микропрограммного управления выполняет переход JCR в текущей строке в колонку с номером 1.

3. В ячейке 051 записана микрокоманда, по исполнению которой ЦП выполняет SDR $A \rightarrow R_0$, а УМУ осуществляет переход JPX в текущей строке в колонку с номером 5.

4. В ячейке 055 записана микрокоманда, по исполнении которой ЦП выполняет пустую операцию, УМУ — переход JCE с выдачей на

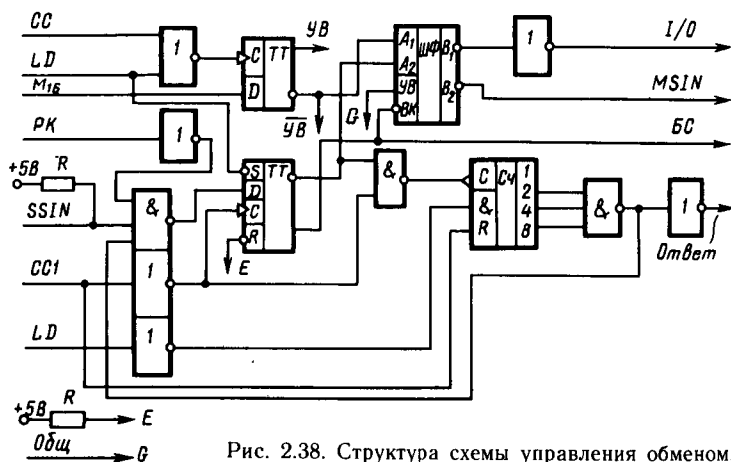


Рис. 2.38. Структура схемы управления обменом.

ВП и РК содержимого регистра команд БМУ (см. рис. 2.29); переход осуществляется в ячейку с номером 045. При этом на выходе РК появляется «0», а по переходу 10 CC1 формируются сигналы $BC = I/O = MSIN = \text{«0»}$ (см. рис. 2.38).

5. Установка сигнала блокировки синхросигналов (BC) в нуль переводит ЦП в режим ожидания. Внешнему приемнику информации сообщается о режиме вывода $I/O = \text{«0»}$ и вырабатывается сигнал синхронизации MSIN. До получения ответного сигнала SSIN об окончании приема информации счетчик С4 схемы обмена осуществ-

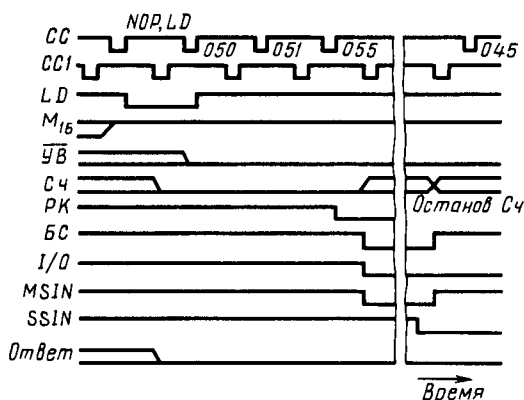


Рис. 2.39. Временная диаграмма работы схемы управления обменом.

ляет счет числа сигналов СС1. При поступлении от приемника сигнала SSIN триггер ТТ по переходу 10 СС1 устанавливается в единичное состояние, сигналы синхронизации MSIN и блокировки БС снимаются, а счет числа сигналов на счетчике Сч прекращается, причем на выходе сигнал «Ответ» равен «0» и оценивается УМУ как указание ($J = \langle 0 \rangle$) к выполнению следующей команды.

В случае отсутствия ответного сигнала SSIN счетчик досчитывает до кода 1111, формирует сигнал об окончании процесса передачи информации (выход элемента ЗИ — HE = «0») и снимает сигналы синхронизации и блокировки БС. При этом сигнал «Ответ» остается в единичном состоянии ($J = \langle 1 \rangle$), что обуславливает пропуск одной и переход к выполнению следующей за ней микрокоманды. Если $J = \langle 0 \rangle$, то последовательно выполняются микрокоманды в ячейках 093, 00B и 00C.

В о д. При выполнении команды ввода в ИУВС обмен и синхронизация с внешними устройствами осуществляются аналогично команде вывода, за исключением установки сигнала I/O = «1» и выполнения на третьем такте микрокоманды занесения информации, установленной на магистрали внешних устройств В в аккумулятор А (см. рис. 2.28).

3. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННО-УПРАВЛЯЮЩИХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

3.1. Обоснование выбора языка программирования

Возросшие требования к скорости, точности обработки информации и значительно увеличившийся ее объем привели к необходимости широкого применения средств вычислительной техники. Стало практически невозможно качественно и в короткий срок проверить вручную техническое состояние сложного объекта, аппаратуры. Решению этой задачи способствует активное внедрение микроэлектронной техники в системы контроля. Попытки использования для этих целей больших ЭВМ и мини-ЭВМ, работающих в режиме разделения времени, встретили такие трудности, как недостаточно высокий уровень надежности связи с удаленными объектами, неэффективность использования ЭВМ по стоимости, трудно контролируемое время ответа ЭВМ. Успехи микроэлектроники и особенно появление доступных, надежных и сравнительно недорогих микроЭВМ привели к возможности реализации идеи децентрализованного контроля.

Децентрализованный контроль позволяет разместить микроЭВМ рядом с проверяемым объектом, а зачастую и встраивать прямо в объект. Программные средства, предназначенные для обеспечения режима разделения времени между многими объектами, требуют больших накладных расходов в добавление к значительным затратам на аппаратурные средства. Такого рода затраты полностью отсутствуют в децентрализованной системе. Следует также отметить еще одно важное достоинство децентрализованных систем — их высокую

надежность, так как выход из строя одной из систем не оказывает никакого влияния на другие системы.

Рассмотрим функции микроЭВМ в системе контроля. Вычислительная машина должна помогать оператору проводить установку и настройку аппаратуры контроля, задавать системе контроля исходные параметры, выдавать информацию о ходе контроля в реальном масштабе времени, фиксировать сбойные ситуации и ошибки, выдавать в виде печатного документа протокол контроля объекта. Кроме того, должна иметься возможность оперативно вносить исправления и добавления в программу контроля как в процессе отладки, так и в процессе ее прохождения на ЭВМ. Для этой цели разрабатываются и используются диалоговые языки программирования.

Среди диалоговых языков в последнее время большое распространение получили языки БЭЙСИК и ФОКАЛ, входящие в программное обеспечение ряда мини- и микроЭВМ. Оба языка обладают всеми свойствами, которые присущи диалоговым языкам такого рода, т. е. простотой и легкостью для изучения; возможностью использования обозначений, подобных обычным математическим; способностью работы как в диалоговом, так и в программном режимах; наличием библиотеки стандартных математических функций и возможностью ее расширения, средств для редактирования и отладки программ, команд для работы со стандартным периферийным оборудованием; диагностикой ошибок.

Языки БЭЙСИК и ФОКАЛ применяются для решения научно-технических задач, экономических расчетов, в области автоматизации научного эксперимента, сфере обучения. Результат их сравнения показывает, что языки обладают примерно одинаковыми вычислительными возможностями, однако язык ФОКАЛ имеет ряд преимуществ в случае его использования в АСК. Так, он в отличие от языка БЭЙСИК обладает: большими возможностями для работы с периферийным оборудованием (наличие функции управления общей магистралью позволяет подключать к ЭВМ нестандартные внешние устройства); расширенными возможностями отладки и редактирования (в нем предусмотрен оператор, с помощью которого можно заменять символы, уничтожать часть строки, вносить новую информацию); возможностью трассировки (управление распечаткой частей программы во время ее выполнения). На уровне языка ФОКАЛ также возможно обслуживание прерываний от различных устройств (при прерывании от устройства выполнение программы приостанавливается и управление передается специальной подпрограмме, обслуживающей прерывания, по окончании работы этой подпрограммы управление будет передано к прерванному оператору); применение сокращенной мнемоники — все операторы языка ФОКАЛ можно обозначать одной начальной буквой, что позволяет, во-первых, уменьшить объем занимаемой оперативной памяти и, во-вторых, до некоторой степени ускорить время составления и выполнения программы. Для микроЭВМ, имеющей ограниченный объем оперативной памяти (28К слов для рассматриваемых типов микроЭВМ), экономия памяти — существенный фактор.

Наконец, к особым достоинствам языка ФОКАЛ следует отнести компактность программы-интерпретатора языка, объем которой не превышает 4К слов. Это дает возможность записать интерпретатор языка ФОКАЛ в одну микросхему программируемого (K573РФЗ) или масочного (K1801PE1) постоянного запоминающего устройства. В одноплатной микроЭВМ «Электроника МС1201.01» предусмотрено место, куда вставляется эта микросхема (контактное устройство для пользовательского ПЗУ). Эксплуатация системы с встроенным интерпретатором существенно облегчается, так как отпадает необходимость ввода в память ЭВМ программы-интерпретатора с каких-либо внешних носителей информации (перфолист, магнитных лент и т. п.). Система готова к работе сразу при включении питания.

3.2. Общие сведения о языке ФОКАЛ

Язык программирования ФОКАЛ относится к диалоговым языкам высокого уровня. Он предусматривает два режима работы: программный (автоматический) и диалоговый (непосредственный).

Для работы в *программном режиме* необходимо, чтобы каждый оператор или строка с несколькими операторами имели свой номер. Последовательность таких пронумерованных операторов и составляет программу, которая может быть выполнена в любое время.

При работе в *диалоговом режиме* оператор (без номера) набирается непосредственно с пульта терминала и выполняется сразу. Для повторного вычисления необходимо с пульта оператор набрать снова. Диалоговый режим работы применяется для выполнения простых вычислений и для отладки программ.

Язык ФОКАЛ имеет 15 операторов [23]. Названия операторов несут смысловую нагрузку. В тексте программ возможно указание лишь одной начальной буквы в названии оператора. Ниже приводятся перечень операторов ФОКАЛа и примеры их использования (табл. 3.1).

Положительные или отрицательные десятичные числа в языке ФОКАЛ представляются в диапазоне $10^{-38} — 10^{+38}$. При выводе чисел на печать применяются форматы с фиксированной точкой и экспоненциальный. Существенными являются шесть знаков в числе, причем пять знаков — точные, а шестой может быть неточным из-за округления. Имена переменных могут состоять из одного или нескольких символов. Первый символ должен быть буквой (кроме буквы F, которая используется для обозначения функций), второй и последующие могут быть как буквами, так и цифрами.

Интерпретатор языка ФОКАЛ использует только два первых символа для идентификации переменной. Переменные, не начинающиеся с символов A и F, могут применяться вместо номера строки в операторах управления.

В языке ФОКАЛ допускаются одно- и двумерные массивы. Индекс может быть числом, переменной или выражением. Пределы применения индекса следующие: для переменной с двумя индексами (—128, +127) и для переменной с одним индексом (—32768, +32767).

Таблица 3.1

Оператор	Пример	Действие
ASK COMMENT	ASK M COMMENT ЗАДАЧА 2	Ввод переменной M Комментарий. При выполнении игнорируется
DO	DO 4.1 DO 4	Выполняется строка с номером 4.1. Затем происходит возвращение к оператору, следующему за DO Выполняется группа операторов с номером 4. Затем происходит возвращение к оператору, следующему за DO
ERASE	ERASE ERASE 2 ERASE 3.1 ERASE ALL ERASE TEXT	Уничтожает переменные Уничтожает группу строк с номером 2 Уничтожает строку с номером 3.1 Уничтожает программу и все переменные Уничтожает только текст программы, переменные остаются
FOR	FOR I=X, Y, Z	Оператор цикла, где X — начальное значение I, Y — шаг приращения I, Z — конечное значение I. Если Y не указан, то предполагается шаг приращения равным 1
GO	GO GO 3.4	Передача управления оператору с наименьшим номером Передача управления оператору с номером 3.4
IF	IF (X) n1, n2, n3 IF (X) n1, n2 IF (X) n1	Оператор условного перехода, где X — переменная, число или выражение При $X < 0$ происходит передача управления к оператору с номером n1; при $X = 0$ — к n2; при $X > 0$ — к n3
MODIFY OPERATE	MODIFY 1.15 OPERATE TK OPERATE P OPERATE R OPERATE L	Редактируется строка с номером 1.15 Выбор в качестве устройства ввода — вывода терминала Выбор ленточного перфоратора Выбор считывателя с перфоленты Выбор широкоформатного печатающего устройства
QUIT	QUIT	Останов программы и переход к диалоговому режиму
RETURN SET TYRE	RETURN SET A=5/B*C TYPE A+B-C	Возвращение в основную программу Оператор присваивания Вычисляет выражения и результат выводит на печать
WRITE	WRITE ALL WRITE 3 WRITE 3.1	Выводит на печать всю программу Печать группы строк с номером 3 Печать строки с номером 3.1
XECUTE	XECUTE FSBR (5,X)	Исполнить подпрограмму-функцию

Арифметические выражения представляют собой комбинацию элементов, связанных знаками десятичной арифметики. Под элементами понимаются переменная, число, функция, элемент массива. Арифметические выражения вычисляются в соответствии с приоритетом операций и наличием скобок.

В случае программного режима номера строк имеют формат

mm.nn,

где *mm* — номер группы; *nn* — номер в группе.

В программе можно использовать любые номера строк от 1.01 до 99.99, за исключением тех, которые оканчиваются на 00. Номера 1.00, 2.00, 3.00 и т. д. используются для указания всей группы строк.

В качестве примера приведем программу для вычисления суммы чисел от 1 до 100:

```
* 1.01 COMMENT вычисление суммы чисел от 1 до 100 <BK>
* 1.03 SET S=0 <BK>
* 1.05 FOR I=1, 100; SET S=S+I <BK>
* 1.07 TYPE «СУММА=», S, ! <BK>
* 1.09 QUIT <BK>
* GO <BK>
```

Каждая строка, введенная с терминала, заканчивается символом <BK> (возврат каретки). Когда ЭВМ готова к приему новой строки, она выводит на терминал символ приглашения *. Для запуска программы на счет вводятся оператор GO и символ <BK>. После выполнения программы ЭВМ выводит на экран терминала строку

СУММА = 5050

Первая строка является комментарием и на выполнение программы никакого действия не оказывает. В строке с номером 1.07 в качестве заключены строковые данные, а символ «!» вызывает перевод строки и возврат каретки печатающего устройства.

Оператор QUIT в строке с номером 1.09 обеспечивает возврат в диалоговый режим.

Нумерацию строк целесообразно вести таким образом, чтобы при необходимости можно было в программу вставить дополнительные строки. В приведенном выше примере программы пропущены для этой цели строки с четными номерами.

Язык ФОКАЛ содержит в своем составе 12 встроенных функций. Аргументом функции может быть арифметическое выражение. Результат вычисления функции есть единственное число, которое присваивается имени функции. Список стандартных функций языка ФОКАЛ и их названия приведены ниже:

FSIN	(аргумент)	Синус (аргумент в радианах)
FCOS	(аргумент)	Косинус (аргумент в радианах)
FX	(функция, адрес, данные)	Функция доступа к общей магистральной
FCHR	(аргумент)	Печать и прием кодов КОИ-7
FRAN	()	Генерация случайных чисел
FADC	(усиление, канал)	Преобразование аналог-цифра
FCLK	()	Часы (датчик времени)
FABS	(аргумент)	Абсолютная величина
FSGN	(аргумент)	Знаковая часть числа
FITR	(аргумент)	Целая часть числа
FSBR	(группа, аргумент)	Функция, программируемая пользователем

Примечание. Функцией FADC можно пользоваться только при наличии соответствующей аппаратуры.

Язык ФОКАЛ содержит удобные средства ввода — вывода числовых данных и текста программ. Строки комментариев начинаются с буквы С и могут содержать, кроме английских букв, любые буквы русского алфавита. Текст, предназначенный для вывода на печать, включается в оператор TYPE и заключается в кавычки. В нем также могут использоваться русские буквы. Назначение устройств ввода — вывода осуществляется с помощью оператора OPERATE.

Данные и текст программ могут быть введены в ЭВМ с помощью клавиатуры терминала или считывателя с перфоленты.

Вывод результатов и текста программ может быть выполнен на экран терминала, электрифицированную пишущую машинку, широкоформатное печатающее устройство или ленточный перфоратор.

Средства отладки программ включают в себя редактирование исходного текста и внесение в него изменений и дополнений, автоматическую выдачу кода ошибок пользователя, трассировку (управление распечаткой текста программы при ее исполнении).

Оператор MODIFY служит для корректировки и уничтожения знаков в строке. Он позволяет отыскать нужный символ, включить новые или уничтожить ненужные знаки, закончить строку в нужном месте.

Диагностическая система ФОКАЛа позволяет обнаружить большинство ошибок, допущенных пользователем. Каждая ошибка идентифицируется своим кодом. Сообщения об ошибках и их код даны ниже:

Повторный запуск системы	00
Неправильный номер строки	01
Неправильное имя функции или имя переменной	02
Непарные скобки	03
Неправильный оператор	04
Несуществующий номер строки	05
Несуществующий номер группы или номер строки в операторе DO	06
Неправильный формат SET или FOR	07
Двойной или отсутствующий оператор в выражении	08
Переполнение стека или несуществующее устройство	09
Переполнение памяти текстом программ	10
Нет места для переменных	11
Порядок больше $E \pm 38$	12
Запрещенный адрес шины в операторе FX	13
Попытка деления на нуль	14
Попытка возведения в отрицательную или слишком большую степень	15
Слишком много символов во входных данных	16
Корень квадратный из отрицательного числа	17
Переполнение входного буфера	18

Язык ФОКАЛ входит в состав стандартного математического обеспечения мини-ЭВМ СМ-3 и СМ-4. Перфоленточный вариант языка получил название ДС СМ ЭВМ (ДС — диалоговая система), а расширенный вариант, использующийся совместно с дисковой операционной системой, — название ДИАСП — диалоговая система программирования.

3.3. Общие сведения об интерпретаторе языка ФОКАЛ

Реализация версии языка ФОКАЛ для АСК учитывает требования преемственности и совместимости со стандартной версией языка ФОКАЛ. При разработке версии учитывалось также размещение программы-интерпретатора языка в ПЗУ и ограничение на объем.

Интерпретатор занимает адресное пространство со 140000₈ по 157776₈ (4К слов). Все свои функции интерпретатор выполняет, непосредственно используя ПЗУ и ограниченное число ячеек оперативной памяти.

Примечание. Разработка версии интерпретатора языка ФОКАЛ для АСК выполнена в Московском институте электронной техники Г. И. Фроловым и Т. А. Ку-права.

Интерпретатор языка ФОКАЛ представляет собой набор подпрограмм, таблиц, флагов, указателей, переменных и комментариев, в целом обеспечивающий простой диалог и все необходимые средства для загрузки, отладки и выполнения программ системы контроля.

Подпрограммы интерпретатора можно разделить в зависимости от их функционального назначения на пять основных групп:

1) подпрограммы начальной загрузки, обеспечивающие формирование в оперативной памяти области векторов прерываний, ловушек останова, начальных значений переменных, указателей, флагов и нулевой строки, а также автоматическое определение непрерывной области исправных ячеек оперативной памяти;

2) обслуживающие подпрограммы, вызываемые с помощью команд TRAP и выполняющие функции обработки текста программ, установку флагов, указателей и переменных для дальнейшего их использования в приведенных ниже подпрограммах;

3) подпрограммы, интерпретирующие операторы языка ФОКАЛ;

4) подпрограммы, интерпретирующие стандартные функции языка ФОКАЛ и функции, обслуживающие устройства системы контроля;

5) пакет подпрограмм, обеспечивающий работу с числами, представленными в формате с плавающей точкой.

Интерпретатор содержит таблицы: 1) обслуживающих подпрограмм; 2) флагов, указателей и переменных; 3) векторов прерываний; 4) операторов языка ФОКАЛ; 5) функций; 6) комментариев; 7) вспомогательные.

Запуск интерпретатора выполняет *подпрограмма начальной загрузки* INIT с пусковым адресом 140000. После запуска интерпретатора языка ФОКАЛ адресное пространство оперативной памяти будет распределено так, как показано на рис. 3.1. На этом же рисунке дано адресное пространство ПЗУ интерпретатора.

При правильном запуске интерпретатора языка ФОКАЛ на экране терминала появляется сообщение:

??00AT0.00

ГОТОВНОСТЬ К РАБОТЕ

*

Наличие * указывает на готовность системы к выполнению любых действий пользователя. Пользователь может теперь либо вводить и выполнять свои программы, либо выполнять отдельные операторы с пульта терминала.

Обслуживающие подпрограммы выполняют организацию диалога человека с машиной, размещение программ пользователя в памяти,

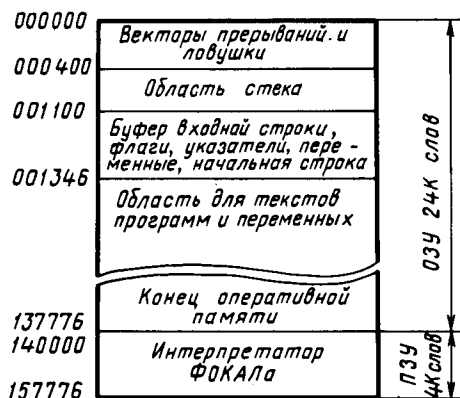


Рис. 3.1. Распределение адресного пространства оперативной памяти.

вычислительные действия, работу с переменными; организуют ввод — вывод информации в коде КОИ-7, поиск элементов в таблицах, выдачу сообщений об ошибках и ответы на запросы пользователя. Большая часть обслуживающих подпрограмм вызывается с помощью команд TRAP, так как на вызов вида

JSR <регистр>, <точка входа>

требуется два слова памяти, а на вызов вида

TRAP + код,

где TRAP = 104400, код = 000—377 — только одно слово памяти.

Вызов вида TRAP работает медленнее, чем вызов вида JSP, но вызов вида TRAP существенно экономит память и обеспечивает более гибкое взаимодействие подпрограмм интерпретатора. При обнаружении TRAP-команды процессор осуществляет внутреннее прерывание и через вектор прерывания (34, 36) выходит на программу «Диспетчер TRAP-команд». В зависимости от младшего байта кода TRAP управление передается на одну из подпрограмм.

Организацию работы подпрограмм, интерпретирующих операторы языка ФОКАЛ, осуществляет диспетчер операторов входного языка. Диспетчер операторов проверяет имя оператора по таблице имен операторов, определяет адрес входа и передает управление на подпрограмму, интерпретирующую данный оператор. Операторы распознаются диспетчером по первой букве.

Примечание. Список операторов языка ФОКАЛ см. в табл. 3.1.

Пакет подпрограмм языка ФОКАЛ включает в себя стандартные функции обслуживания устройств системы контроля. Имена всех функций начинаются с буквы F, которая зарезервирована для указания и вызова функции. Вызов подпрограммы, интерпретирующей конкретную функцию, выполняет обслуживающая подпрограмма EVAL.X через таблицу функций. В таблице функций для каждой функции отводятся два слова, которые содержат: а) адрес входа в подпрограмму-функцию; б) смешанный код функции.

Примечание. Методика вычисления смешанного кода функции будет приведена при рассмотрении построения подпрограммы обслуживания аналого-цифрового преобразователя напряжения.

Для обработки чисел с плавающей точкой в языке ФОКАЛ используются специальные операции и функции. Вызов подпрограмм, выполняющих эти операции и функции, осуществляется при обнаружении процессором кодов 007000₈—007377₈. При этом происходит внутреннее прерывание по вектору (10, 12).

Операции, применяемые в языке ФОКАЛ для работы с числами с плавающей точкой, представлены в табл. 3.2.

Таблица 3.2.

Операция	Код	Назначение
FGET	007000	Загрузка операнда во FLAC
FADD	007010	Сложение с FLAC
FSUB	007020	Вычитание из FLAC
FDIV	007030	Деление FLAC
FMUL	007040	Умножение FLAC
FPOW	007050	Возведение FLAC в степень
FPUT	007060	Размещение FLAC по определенному адресу

Примечание. FLAC — накопитель чисел с плавающей точкой, размещенный в трех последовательных ячейках системной области оперативной памяти.

Команды для чисел с плавающей точкой записываются в виде

код операции + тип адресации

Пакет подпрограмм для чисел с плавающей точкой реализует семь способов адресации (табл. 3.3). Для работы с числами в форме с плавающей точкой интерпретатор языка ФОКАЛ содержит функции, представленные в табл. 3.4.

Использование операций и функций пакета с плавающей точкой облегчает написание подпрограмм-функций для обслуживания устройств АСК.

Т а б л и ц а 3.3

Способ адресации	Код	Назначение
DIRECT	0	Следующее слово — адрес операнда
IPTR	1	Адрес операнда находится в R2
XPTR	2	$((R2) + 4)$ есть адрес операнда
INTO + STACK	3	Данные находятся в стеке
FROM + STACK	3	То же
THROUGH + STACK	4	Стек содержит адрес операнда
IMMED	5	Следующие два слова являются операндом
REL	6	$(R0)$ есть адрес операнда

П р и м е ч а н и е. Адресации INTO + STACK и FROM + STACK одинаковы и введены для удобства программиста.

Т а б л и ц а 3.4

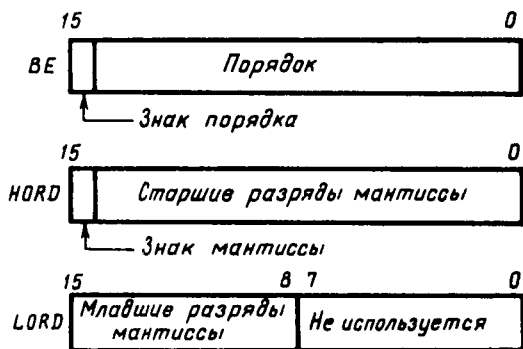
Функция	Код	Назначение
FNOR	007070	Нормализация числа во FLAC
FINT	007071	Берется целая часть числа от FLAC
FSGN	007072	Знак числа
FABS	007073	Абсолютная часть числа
FNEG	007074	Изменение знака числа во FLAC
FREAD	007075	Чтение чисел во FLAC
FPRINT	007076	Вывод числа с плавающей точкой
FZER	007077	Обнуление FLAC
FCODE	007200	Выполнение операции, код которой содержится в R1

3.4. Разработка подпрограмм-функций обслуживания внешних устройств

Для каждого нестандартного устройства, входящего в состав АСК, пользователь должен разработать и включить новую функцию в интерпретатор языка ФОКАЛ. Эта функция должна учитывать особенности того устройства, которое она обслуживает, и быть написана на языке ассемблера. В общем случае написание программ на языке ассемблера — достаточно трудная и кропотливая задача, так как пользователь должен хорошо представлять себе организацию ЭВМ, знать систему команд, иметь практические навыки работы с операционной системой. Однако эти трудности не столь велики при написании новых функций для АСК, что объясняется двумя причинами: во-первых, сами функции обычно достаточно просты, и, во-вторых, задача существенно облегчается, если использовать готовые операции и функции пакета подпрограмм, входящего в состав интерпретатора.

Для обработки чисел с плавающей точкой в языке ФОКАЛ используются специальные операции и функции. Вызов программ,

интерпретирующих эти функции, осуществляется с помощью внутреннего прерывания по кодам резервных команд 007000₈—007377₈. Адрес программной обработки этих кодов (вектор прерывания) находится в ячейке 10₈, а состояние процессора — в ячейке 12₈. Один из операндов и результат всех операций помещаются в накопитель с плавающей точкой, названный FLAC. FLAC состоит из трех последовательных слов оперативной памяти, обозначенных BE, HORD, LORD, содержащих соответственно порядок, старшие разряды мантиссы и младшие разряды мантиссы:



Порядок представляется в дополнительном коде, величина его ограничена и находится в следующем диапазоне: $-200_8 \leq \text{порядок} \leq +177_8$.

Это связано с тем, что числа с плавающей точкой хранятся в памяти в двухсловном формате и для порядка отводится 1 байт. Таким образом, наибольший положительный порядок в слове равен 000177₈, а наибольший по модулю отрицательный порядок равен -177600_8 (дополнительный код числа -200_8).

Мантисса числа с плавающей точкой представляется в дополнительном коде в нормализованном виде и занимает 3 байта: старший и младший байты слова HORD и старший байт слова LORD. Младший байт слова LORD не используется. Знак мантиссы представлен 15-м разрядом слова HORD.

Существует простой способ, чтобы практически «посмотреть» представление различных чисел с плавающей точкой во FLAC. Для этого нужно исполнить оператор

*EXECUTE <число> <БК>

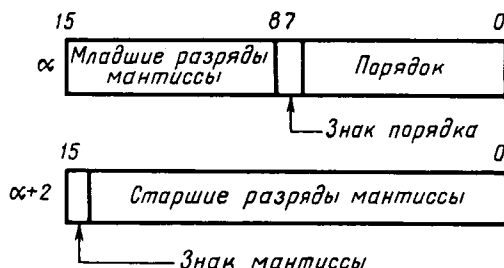
Затем остановить микроЭВМ и в пультовом режиме посмотреть содержимое ячеек памяти, соответствующее BE, HORD и LORD. Например,

$$+1 \equiv \begin{cases} 000001 \\ 040000; \\ 000200 \end{cases} \quad +1/7 \equiv \begin{cases} 177776 \\ 044444; \\ 111200 \end{cases}$$

$$-1 \equiv \begin{cases} 000001 \\ 140000; \\ 000200 \end{cases} \quad -1/7 \equiv \begin{cases} 177776 \\ 133333. \\ 067200 \end{cases}$$

Примечание. Код 200_8 в младшем байте LORD интерпретатором игнорируется.

В памяти микроЭВМ числа с плавающей точкой хранятся в двух последовательных словах в формате



При выполнении операций и функций над числами с плавающей точкой двухсловный формат (внешний) преобразуется в трехсловный (внутренний). Например, дробь $+1/7$ во внешнем формате представляется так:

$$\begin{array}{ll} \alpha & 111376 \\ \alpha + 2 & 044444 \end{array}$$

Здесь младший байт слова, находящегося по адресу α , представляет порядок (-2), записанный в дополнительном коде. Старший байт этого слова содержит младшие разряды мантииссы. В следующем слове расположены знак и старшие разряды мантииссы.

Диапазон чисел, которые можно представить в форме с плавающей точкой, практически зависит только от числа разрядов, отводимых под порядок, и равен

$$10^{-38} < |X| < 10^{+38},$$

а точность определяется числом разрядов, отводимых под мантииссу. Мантиисса с 23 двоичными разрядами дает 6—7 верных десятичных цифр в числе.

Этапы разработки подпрограммы-функции рассмотрим на примере построения подпрограммы обслуживания преобразователя напряжения в цифровой код. В качестве преобразователя использован выпускаемый промышленностью модуль аналогового ввода 15KA-60/8-10, предназначенный для применения в составе микроЭВМ «Электроника-60».

Схема аналого-цифрового преобразователя и его сопряжение с общей магистралью показаны на рис. 3.2. Аналого-цифровой преобразователь (АЦП) состоит из мультиплексора МС, выбирающего один из 16 аналоговых входных сигналов, и собственно преобразователя П, вырабатывающего цифровой эквивалент аналогового входа,

передатчиков Пд, приемников Пр, дешифратора адреса и управляющих сигналов ДША и Упр, триггера «Готовность» Т. Схема сопряжения АЦП с общей магистралью микроЭВМ, называемая *интерфейсом*, содержит три адресуемых регистра: регистр номера канала РгНк ($RN=160070$); буферный регистр данных ($RD=160072$); регистр состояния ($RS=160074$). В скобках указаны символические

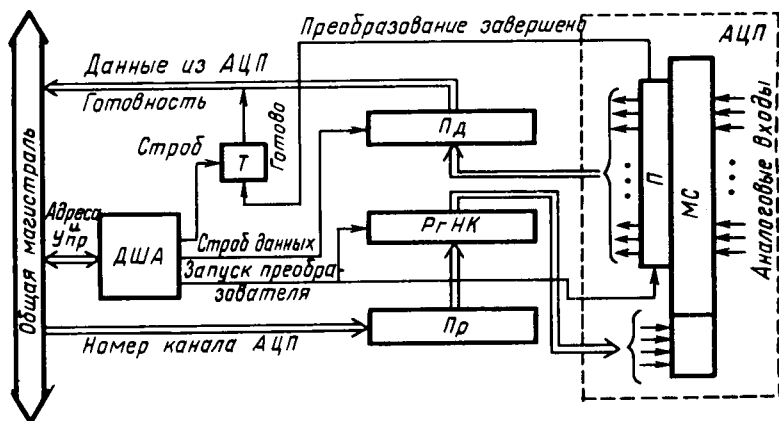
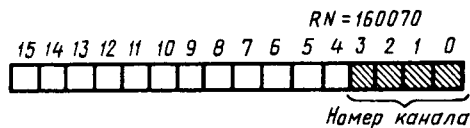


Рис. 3.2. Схема модуля аналогового ввода микроЭВМ «Электроника-60».

имена и адреса регистров на общей магистрали. По желанию пользователя эти адреса можно изменить, установив соответствующие переключки на плате модуля.

Рассмотрим форматы регистров.

В регистре номера канала РгНк АЦП активными являются четыре младших бита, с помощью которых можно выбрать любой из 16 каналов, имеющих адреса $00_8—17_8$:



АЦП начинает свой цикл преобразования с момента отправки номера соответствующего канала в регистр номера канала.

Буферный регистр данных содержит 10-разрядный выходной код АЦП и его знак:

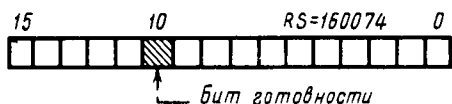


Весовые эквиваленты соответствующих разрядов кода приведены ниже

Номер бита	БИТ [09]	БИТ [08]	БИТ [07]	БИТ [06]	БИТ [05]
Весовой эквивалент, В	0,01	0,02	0,04	0,08	0,16
Номер бита	БИТ [04]	БИТ [03]	БИТ [02]	БИТ [01]	БИТ [00]
Весовой эквивалент, В	0,32	0,64	1,28	2,56	5,12

Нулевым значениям всех битов соответствует напряжение 0, а всем единицам (код 1777) напряжение 10,23 В. Знаковый бит БИТ [15] = 0 кодирует положительные напряжения, а БИТ [15] = 1 — отрицательные. Следовательно, диапазон измеряемых входных напряжений рассматриваемого АЦП ограничен пределами $-10,23$ и $+10,23$ В.

Регистр состояния имеет лишь один активный бит — БИТ [10]:



Значение БИТ [10] = 1 свидетельствует о завершении цикла преобразования, после чего возможно считывание содержимого буферного регистра данных в микроЭВМ.

Для измерения входного напряжения, например, по седьмому каналу и передачи соответствующего измеренному значению кода в регистр R0 можно написать такую последовательность команд:

```

MOV #7, RN      ; засылка номера канала и инициирование АЦП
LOOP: BIT #2000, RS ; проверка завершения цикла преобразования
      BEQ LOOP    ; вернуться к проверке, если преобразование не
                  ; завершено
MOV RB, R0      ; передача кода в R0

```

Для дальнейшей обработки в ЭВМ кода, принятого из АЦП, необходимо привести его к форме числа с плавающей точкой, принятой в интерпретаторе. Сначала код, находящийся в буферном регистре АЦП, преобразуем так, чтобы биту RD [00] соответствовал наименьший весовой коэффициент 0,01 В, биту RD [01] — весовой коэффициент 0,02 В, биту RD [02] — весовой коэффициент 0,04 В и т. д. до самого старшего значащего бита RD [09], которому соответствовал бы весовой коэффициент 5,12 В. Указанные преобразования легко выполняются использованием команд правого и левого циклического сдвигов с участием бита переноса С регистра состояния процессора. Логiku работы приводимой ниже программы поясняют комментарии к командам:

```

CLR HORD      ; подготовка ячейки памяти для преобразо-
              ; ванного кода
MOV RD, WORK  ; пересылка содержимого буферного регистра
              ; АЦП в рабочую ячейку

```


	MOV $\neq$ 10., CNT	;	подготовка счетчика сдвигов
AGAIN:	ROR WORK	;	правый циклический сдвиг
	ROL HORD	;	левый циклический сдвиг
	DEC CNT	;	уменьшение счетчика сдвигов
	BNE AGAIN	;	вернуться к метке AGAIN, если содержимое счетчика сдвигов не 0

После выполнения этой программы содержимое ячеек HORD и WORK будет следующим:

HORD:															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	RD	RD	RD	RD	RD	RD	RD	RD	RD	RD
						[00]	[01]	[02]	[03]	[04]	[05]	[06]	[07]	[08]	[09]

WORK:															
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	RD	0	0	0	0	0
										[15]					

БИТ [5] рабочей ячейки WORK содержит знак кода, принятого из АЦП. Преобразованный код АЦП запишем в виде целого числа в накопитель с плавающей точкой FLAC с учетом его знака:

CLR LORD	;	очистка младших разрядов
	;	мантиссы
MOV $\neq$ 17, BE	;	подготовка порядка
TST WORK	;	опрос знака кода
BEQ M	;	пропустить следующую ко-
	;	манду, если код положитель-
	;	ный
NEG HORD	;	взять дополнение кода
M: FNOR	;	нормализовать число во
		FLAC

Для получения результата напряжения, выраженного в вольтах, необходимо содержимое накопителя с плавающей точкой умножить на константу 0,01:

FMUL + IMMED	;	умножить содержимое на число,
	;	расположенное в двух следую-
	;	щих за этой командой словых
104372	;	константа 0,01 в форме с пла-
050753	;	вающей точкой

Пример полностью законченной подпрограммы — функции обслуживания АЦП — приводится ниже:

AC = %1	;	определение регистра
RN = 160070	;	регистр номера канала АЦП
	;	
RD = 160072	;	буферный регистр данных АЦП
	;	
RS = 160074	;	регистр состояния АЦП
	;	
FMUL = 7040	;	подпрограмма умножения чисел с плавающей точкой
	;	
	;	

	IMMED=5	:	непосредственный тип
	FNOR = 7070	:	адресации
	FINT= 7071	:	подпрограмма норма-
		:	лизации
		:	подпрограмма выделе-
		:	ния целой части и за-
		:	сылки в AC
XADC:	FINT	:	
	BIT≠177760, AC	:	засылка номера кана-
	BEQ .+4	:	ла АЦП в AC
	ERROR+251	:	проверка на допусти-
	MOV AC, RN	:	мость номера канала
		:	пропустить следующую
		:	команду, если номер
		:	канала допустим
		:	ошибка: недопустимый
		:	номер канала
LOOP:	BIT≠2000, RS	:	передача номера ка-
	BEQ LOOP	:	нала и запуск АЦП
	MOV RD, —(SP)	:	
	MOV≠17, BE	:	преобразование завер-
	CLR HORD	:	шено?
	CLR LORD	:	вернуться к проверке,
		:	если нет
		:	запомнить код АЦП в
		:	стеке
		:	занести порядок
		:	обнулить старшие и
		:	младшие разряды ман-
		:	тиссы
		:	подготовить счетчик
AGAIN:	ROR 2(SP)	:	сдвигов
	ROL HORD	:	
	DEC (SP)	:	правый циклический
	BNE AGAIN	:	сдвиг
	TST 2(SP)	:	левый циклический
	BEQ M1	:	сдвиг
		:	уменьшить содержимое
		:	счетчика сдвигов
		:	вернуться, если счет-
		:	чик сдвигов не 0
		:	анализ знака кода
		:	АЦП
		:	пропустить следую-
		:	щую команду, если код
M1:	NEG HORD	:	АЦП положительный
	FNOR	:	взять дополнение
	FMUL+IMMED	:	
	104372	:	нормализовать число
	050753	:	во FLAC
	CMP (SP)+, (SP)+	:	умножить FLAC на
	RTS PC	:	константу
		:	0,01
		:	освободить стек
		:	выход из подпрограм-
		:	мы

Прежде чем ввести новую функцию в состав интерпретатора, необходимо определить ее смешанный код. Пользователь выбирает имя функции с учетом того, что она должна начинаться с буквы F. Же-

лательно, чтобы имя функции содержало возможно меньшее число символов, так как при этом экономится память ЭВМ и меньше расходуется машинного времени на расшифровку функции. Интерпретатор использует имя функции в специальном 16-битовом смешанном коде. Чтобы получить этот код, необходимо выполнить следующие операции:

1) найти для каждого символа имени функции соответствующее значение кода КОИ-7. Пусть имя функции — FADC. Тогда F=106; A=100; D=104; C=103;

2) умножить каждый код на число, представляющее номер его позиции в степени 4, т. е. самый правый символ умножается на 4^0 , следующий символ слева — на 4^1 , следующий символ слева — на 4^2 и т. д. Все операции выполняются в восьмеричной системе счисления, и результат берется по модулю 200 000;

$$\begin{aligned} 103 \times 4^0 &= 103 \times 1 = 103; \\ 104 \times 4^1 &= 104 \times 4 = 420; \\ 101 \times 4^2 &= 101 \times 20 = 2020; \\ 106 \times 4^3 &= 106 \times 100 = 10600; \end{aligned}$$

3) результаты суммировать для получения смешанного кода $103 + 420 + 2020 + 10600 = 013343$.

Проверить, чтобы полученный смешанный код не совпадал со смешанными кодами функций, входящими в состав интерпретатора. В противном случае подобрать другое имя функции.

Когда в программе пользователя встречается имя функции, то интерпретатор сначала вычисляет ее смешанный код, а затем осуществляет поиск в таблице функций. Таблица функций имеет следующую структуру:

000000	;	признак окончания
000000	;	таблицы функций
XADC	;	имя точки входа новой функции
013343	;	смешанный код новой функции
XSGN	;	имя стандартной функции интерпретатора
	;	FSGN
014032	;	смешанный код этой функции
.....	;	
XCLK	;	имя стандартной функции FCLK
013453	;	смешанный код функции FCLK

Поиск в таблице начинается снизу, при совпадении смешанных кодов управление передается на подпрограмму, точка входа которой определяется именем, расположенным в следующем слове таблицы над смешанным кодом. Признаком конца таблицы служат два нуля в слове. Если ни один смешанный код не найден, то выдается диагностическое сообщение об ошибке — несуществующая функция. Следует иметь в виду, что чем ближе к началу таблицы будут расположены точка входа и смешанный код новой функции, тем быстрее эта функция будет найдена.

После того как написан текст новой функции и найден ее смешанный код, с помощью системной программы-редактора эта информа-

ция вносится в исходный текст программы-интерпретатора. Затем выполняются трансляция и получение рабочей версии интерпретатора с включенной в его состав новой функцией.

Теперь, чтобы измерить напряжение по любому каналу и вывести его значение на экран дисплея, достаточно ввести с клавиатуры оператор

*T FADC (3) <BK>
5.364*

Измеренное значение напряжения по третьему каналу АЦП выведено в следующей строке на экран дисплея. Следует учитывать, что рассматриваемый модуль аналогового ввода значение измеренных напряжений по каналам 8—15 выдает с обратным знаком.

3.5. Программирование устройств выдачи и приема релейных сигналов

Для реализации возможности выдачи управляющих релейных сигналов в объект контроля, а также для приема релейных сигналов из объекта контроля в ЭВМ целесообразно применить стандартное устройство параллельного обмена И2 (15КС-180-032). Это устройство представляет собой плату, вставляемую в одну из розеток общей магистрали микроЭВМ «Электроника-60». Функционально устройство И2 обеспечивает прием и передачу 16-битовых данных в режимах программном и по прерыванию. Для решения задачи выдачи и приема релейных сигналов достаточно использовать лишь часть возможностей, предоставляемых устройством параллельного обмена.

Упрощенная схема выдачи и приема релейных сигналов показана на рис. 3.3. Устройство содержит в своем составе дешифратор адресов и управляющих сигналов ДША и Упр, выходной регистр РгВых, приемники Пр, передатчики Пд и управляющие цепи.

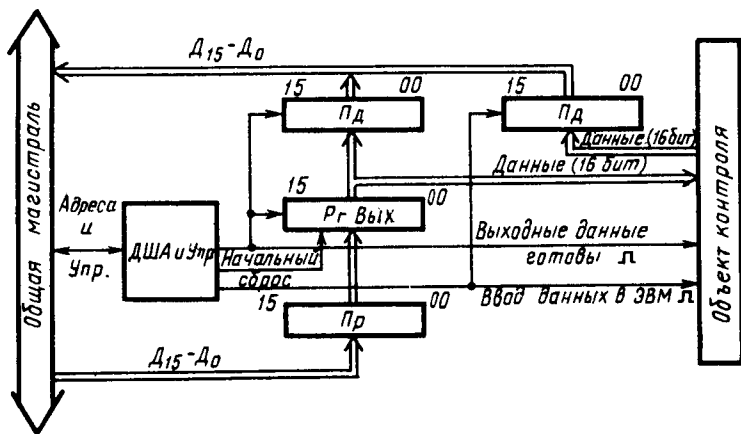


Рис. 3.3. Схема устройства параллельного обмена.

С точки зрения программиста, устройство содержит два адресуемых регистра: выходной 16-разрядный регистр RBO, предназначенный как для записи, так и для считывания, и входной 16-разрядный регистр RBI, предназначенный только для считывания. Фактически входной регистр отсутствует — при обращении к регистру ввода RBI открываются лишь вентили, управляющие передачей данных из объекта контроля через передатчики Пд в общую магистраль.

При начальном запуске интерпретатора языка ФОКАЛ происходит автоматическое обнуление выходного регистра RBO с помощью сигнала «Начальный сброс».

Для управления установкой или сбросом любого из 16 управляющих релейных сигналов введем в язык ФОКАЛ новые функции FSET (Nk) и FCLR(Nk). Функция FSET(Nk) устанавливает в «1» сигнал по выбранному каналу с номером Nk, а функция FCLR(Nk) сбрасывает его в «0». Номер канала Nk находится в диапазоне 0—15.

Рассмотрим пример реализации подпрограммы-функции FSET (Nk) на языке ассемблера. Для выдачи управляющих сигналов в объект контроля используем выходной регистр RBO устройства параллельного обмена. Для управления установкой или сбросом любого из битов выходного регистра в состав интерпретатора включим нижеприведенную табл. 3.5 констант TCNST:

Т а б л и ц а 3.5

Восьмеричный код константы (маска)	Номер управляемого бита регистра	Восьмеричный код константы (маска)	Номер управляемого бита регистра
000001	0	000400	8
000002	1	001000	9
000004	2	002000	10
000010	3	004000	11
000020	4	010000	12
000040	5	020000	13
000100	6	040000	14
000200	7	100000	15

Каждая строка таблицы содержит маску и номер соответствующего бита. Чтобы установить, например, бит 10 регистра RBO в «1», необходимо исполнить команду

BIS # 2000, RBO.

Общая структура подпрограммы-функции FSET (Nk) и ее характер весьма схожи со структурой, рассмотренной в предыдущем разделе подпрограммы-функции обслуживания аналого-цифрового преобразователя напряжения. Существо программы FSET (Nk) поясняют комментарии к каждой команде:

AC = % 1	;	общий регистр
RBO = 167762	;	выходной регистр устройства парал-
	;	лельного обмена
ERROR = 104400	;	код подпрограммы обработки ошибок
FINT = 007071	;	подпрограмма преобразования со-
	;	держимого FLAC в целое число и
	;	засылки его в аккумулятор
XSET: FINT	;	запись номера канала в аккумулятор
BIT #177740, AC	;	проверка на допустимость номера
	;	канала
BEQ .+4	;	пропустить следующую команду, ес-
	;	ли номер канала допустим
ERROR + 247	;	ошибка 19: недопустимый номер ка-
	;	нала
ASL AC	;	удвоить номер канала (для получе-
	;	ния четного адреса)
ADD #TCNST, AC	;	сформировать адрес соответствую-
	;	щей маски в таблице констант
BIS (AC), RBO	;	установить в «1» соответствующий
	;	бит регистра
RTS PS	;	выход из подпрограммы
	;	далее следует таблица констант

TCNST: WORD 1, 2, 4, 10, 20, 40, 100, 200, 400, 1000
WORD 20000, 40000, 100000

Подпрограмма-функция сброса в «0» сигнала в соответствующем канале FCLR(Nk) аналогична и отличается лишь другим именем точки входа в подпрограмму и единственной командой (для нее написан комментарий):

```

AC = % 1
RBO = 167762
ERROR = 104400
FINT = 007071
XCLR: FINT
      BIT #177740, AC
      BEQ .+4
      ERROR + 247
      ASL AC
      ADD #TCNST, AC
      BIC (AC), RBO ; сбросить в «0» соответствующий канал
      RTS PC

```

Примечание. Таблица констант TCNST для этой программы та же, что и для предыдущей.

С помощью устройства параллельного обмена можно проанализировать состояние (уровень) любого из 16 сигналов, поступающих из объекта контроля в ЭВМ. Для этой цели введем язык ФОКАЛ в функцию FTST(Nk).

Функция FTST(Nk) принимает значение «0» или «1» в зависимости от уровня сигнала в выбранном канале (соответственно низкий или высокий).

В приводимой ниже программе на языке ассемблера в зависимости от состояния проверяемого бита во входном регистре RBI в аккумулятор FLAC заносится либо «1», либо «0»:

AC = % 1	;	общий регистр
RBI = 167764	;	входной регистр устройства параллельного обмена

	ERROR=104400 ;	код подпрограммы обработки ошибок
	FINT=007071 ;	целая часть от содержимого FLAC заносится в аккумулятор
XTST:	FINT ;	запись номера канала в аккумулятор AC
	BIT #177740, AC;	проверка на допустимость номера канала
	BEQ .+4 ;	пропустить следующую команду, если номер канала допустим
	ERROR+247 ;	ошибка 19; недопустимый номер канала
	ASL AC ;	удвоить номер канала (для получения четного адреса)
	ADD #TCNST, ;	сформировать в аккумуляторе адрес соответствующей маски
	AC ;	
	BIT (AC), RBI ;	опросить следующий бит
	BNE XONE ;	если соответствующий бит «1», то перейти к метке XONE
	CLR BE ;	очистить порядок
	CLR HORD ;	очистить старшие разряды мантиссы
	BR EXIT ;	безусловный переход
XONE:	MOV #1, BE ;	занести «1» во FLAC
	MOV #40000, HORD ;	
EXIT:	CLR LORD ;	очистить младшие разряды мантиссы
	RTS PS ;	выход из подпрограммы

Примечание. Программа использует ранее описанную таблицу констант TCNST.

Для включения функций FSET(Nk), FCLR(Nk) и FTST(Nk) в состав интерпретатора ФОКАЛА необходимо вычислить их смешанные коды. Методика вычисления смешанных кодов была описана ранее.

Ниже приводятся примеры использования названных функций в программах на языке ФОКАЛ:

* 1.10 X FSET(4); X FSET(8); X FCLR(12)

Выполнение этой строчки приведет к установке «1» в каналах 4 и 8 и установке «0» в канале 12

* 2.15 S K1=FTST(1); S K7=FTST(7)

После выполнения строки операторов с номером 2.15 переменные K1 и K7 получают значение либо «0» либо «1» в зависимости от состояния (уровня) сигнала в соответствующих каналах.

3.6. Общие принципы построения программного обеспечения автоматизированных систем контроля

Программная совместимость микроЭВМ «Электроника MC1201.01» с микроЭВМ «Электроника-60», мини-ЭВМ CM-3, CM-4 и «Электроника-100-25» позволяет применять широкий ассортимент системного программного обеспечения, разработанного для этого семейства ЭВМ. Кроме того, программная совместимость дает возможность вести параллельно разработку как аппаратурной части АСК, так и программной. При этом отпадает необходимость в имитато-

рах и кросс-средствах. Написание новых подпрограмм-функций, их редактирование, трансляция и отладка могут быть выполнены на мини-ЭВМ, после чего полученный машинный код загружается в микроЭВМ для проверки и окончательной отладки.

Программа-интерпретатор языка ФОКАЛ (в дальнейшем просто интерпретатор) написана на языке ассемблера в виде абсолютной секции. Листинг интерпретатора содержит три файла с именами FOCAL1.MAC, FOCAL2.MAC и FOCAL3.MAC. Файлы записаны на магнитном диске мини-ЭВМ СМ-4 и занимают область памяти объемом 154 блока (каждый блок содержит 256 16-разрядных слов). Для получения абсолютного модуля интерпретатора, пригодного для загрузки и выполнения в микроЭВМ, исходные файлы транслируются с помощью системной программы MACRO операционной системы РАФОС.

С помощью директивы

```
RUN MACRO<BK>
```

вызывается транслятор с языка МАКРОАССЕМБЛЕР и затем вводится командная строка

```
* FOCAL.LDA/E:ABS= FOCAL1, FOCAL2,  
FOCAL3<BK>
```

Результатом трансляции является абсолютный модуль с именем

```
FOCAL.LDA
```

Вывод полученного модуля на перфоленту осуществляется с помощью системной программы COPY (копирование):

```
.COPY FOCAL.LDA PC:<BK>
```

В приведенной строке через PC: обозначено системное устройство вывода на перфоленту.

В адресном пространстве микроЭВМ интерпретатор занимает ячейки памяти начиная с адреса 140000₈. В минимальном объеме интерпретатор занимает примерно 2,5 К слов и обеспечивает организацию диалога пользователя с микроЭВМ, все необходимые вычислительные средства — средства управления программой, вводом — выводом, редактированием, а также обеспечивает вызов и исполнение встроенных функций (см. § 3.2).

Для использования интерпретатора языка ФОКАЛ в конкретной АСК он дополняется необходимыми специальными функциями.

Примечание. Методика составления таких функций была описана ранее на примерах АЦП и устройства выдачи и приема релейных сигналов.

Общий объем интерпретатора не должен превышать 4К слов. Это ограничение накладывается размером адресного пространства микроЭВМ «Электроника» МС1201.01», выделенного для пользовательского ПЗУ. Таким образом, для специальных функций остается

около 1,5 К слов. Учитывая, что специальные функции для внешних устройств достаточно просты и объем каждой не превышает нескольких десятков слов, это ограничение не является жестким.

Схема алгоритма получения новой версии интерпретатора для АСК показана на рис. 3.4.

Обсудим несколько подробнее оператор схемы с номером 6. Опробование новой версии интерпретатора осуществляется путем

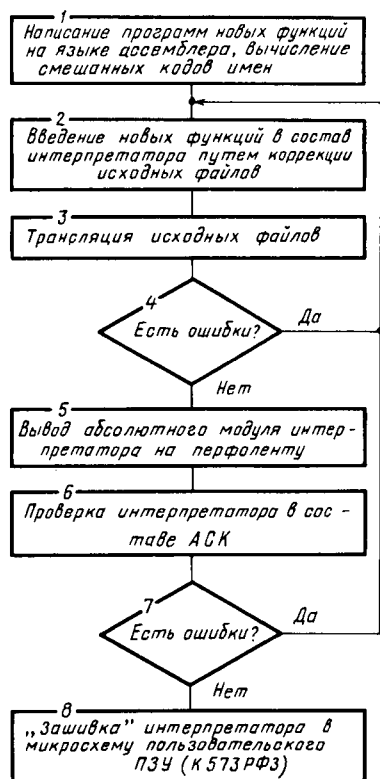
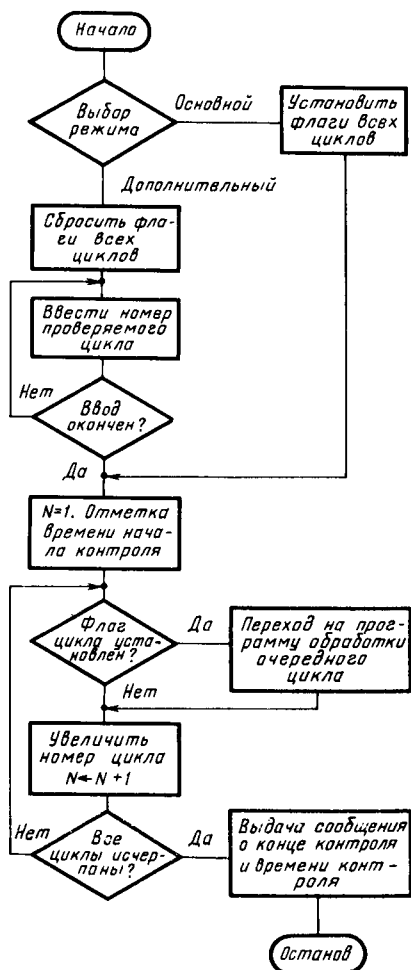


Рис. 3.4. Схема алгоритма получения версии интерпретатора.

Рис. 3.5. Схема организации программы диспетчера.



его загрузки в оперативную память микроЭВМ с адреса 140000₈. С этой целью используются программа абсолютной загрузки и считыватель с перфоленды. Затем в составе АСК тщательно проверяются на функционирование вновь введенные компоненты программного обеспечения.

После устранения имеющихся ошибок интерпретатор может

быть «зашит» в перепрограммируемое ультрафиолетовое ЗУ на микросхеме К573РФ3 с помощью программатора. Операция записи предварительно «стертого» ультрафиолетового ПЗУ занимает несколько минут. Микросхема К573РФ3 с записанным в ней интерпретатором вставляется затем в гнездо пользовательского ПЗУ, расположенное на плате микроЭВМ. Далее с помощью переключателя отключается шестой банк оперативной памяти микроЭВМ, соответствующий адресам 140000₈—157776₈.

П р и м е ч а н и е. Вновь вводимые специальные функции должны быть написаны таким образом, чтобы обеспечить их функционирование именно в постоянной памяти, а не в оперативной.

После того как в состав интерпретатора языка ФОКАЛ включены необходимые дополнительные функции, можно приступить к составлению программ контроля. Запись программ на языке высокого уровня не представляет особых затруднений, ее структура и объем определяются конкретно поставленной задачей контроля. Целесообразно при написании программы руководствоваться модульным принципом ее построения. Ядро программного обеспечения АСК составляет программа-диспетчер. Она выполняет функции управления системой и организует работу подпрограмм контроля. Каждая подпрограмма обеспечивает один из необходимых циклов контроля, например выдает управляющие сигналы, проводит необходимые измерения, контролирует состояние выходных сигналов объекта, проверяет, укладывается ли данный параметр в допуск, и т. д.

В качестве примера рассмотрим организацию простой программы-диспетчера, схема которой изображена на рис. 3.5. Эта программа в зависимости от выбранного режима позволяет последовательно выполнить либо все десять циклов (подпрограмм) контроля, либо только выбранные из них. Выбор соответствующих циклов осуществляется с помощью флагов, записанных в массив ячеек $A(N)$. Например, если массив $A(N)$ заполнен так, как показано ниже:

0	1	1	0	0	0	1	1	0	1
---	---	---	---	---	---	---	---	---	---

 $A(N)$

то будут выполнены циклы с номерами 2, 3, 7, 8 и 10.

Ниже на языке ФОКАЛ приведена программа-диспетчер с необходимыми комментариями:

```

1.01 Т «ВВЕДИТЕ РЕЖИМ: ОСНОВНОЙ (0), ДОПОЛНИТЕЛЬНЫЙ (1)»
1.03 А А; С ОЖИДАНИЕ ВВОДА КОДА РЕЖИМА
1.05 I (А) 1.07, 1.09, 1.11
1.07 Т «ПОВТОРИТЕ ВВОД»; G 1.01
1.09 F I=1.10; S A(I)=1; G 1.23
1.11 F I=1.10; S A(I)=0; С ОБНУЛЕНИЕ ФЛАГОВ ЦИКЛОВ
1.13 Т «ВВЕДИТЕ НОМЕРА ПРОВЕРЯЕМЫХ ЦИКЛОВ»
1.15 А N; I (N) 1.17, 1.19, 1.21
1.17 Т «ПОВТОРИТЕ ВВОД!» G 1.15
1.19 Т «ВВОД КОНЧЕН»; G 1.23
1.21 S A(N)=I; I (N—10) 1.15, 1.15, 1.17

```

1.23 S TO=FCLK (); S N=1
 1.25 I (A(N)) 1.27, 1.29, 1.27; С ОПРОС ФЛАГА ЦИКЛА
 1.27 S NC=N+10.01; G NC; С ПЕРЕХОД НА П/П цикла
 1.29 S N=N+1; I (N-10) 1.25, 1.25, 1.31
 1.31 T «КОНЕЦ КОНТРОЛЯ»! «ВРЕМЯ КОНТРОЛЯ»
 1.33 T (FCLK ()-TO)/3000, «МИНУТ»
 1.35 Q; С КОНЕЦ ПРОГРАММЫ И ВОЗВРАТ В ДИАЛОГОВЫЙ РЕЖИМ

После исполнения оператора ASK в строке 1.03 ЭВМ выводит на экран символ «:» и ожидает ввода кода режима. После ввода 1 и 0 нажимается клавиша «Возврат каретки» (признак окончания ввода информации). Строки операторов с номерами 1.13, 1.15, 1.17 и 1.19 служат для установки флагов контролируемых циклов. Признаком окончания ввода номеров циклов является ввод символа «Возврат каретки» (ему соответствует нулевое значение, присваиваемое переменной N). Подпрограммы циклов должны начинаться с номеров строк 11.01, 12.01, 13.01, ..., 19.01, 20.01. Переход на соответствующий цикл выполняется с помощью операторов строки с номером 1.27. Оператор

S NC=N+10.01

вычисляет метку перехода на соответствующий цикл, а оператор

G NC

передает управление на эту метку. Каждый цикл должен заканчиваться оператором

G 1.29

возвращающим управление программе-диспетчеру.

В качестве примера рассмотрим один из возможных вариантов подпрограммы на языке ФОКАЛ, определяющей, находится ли измеренное напряжение в допуске, и устанавливающей соответствующее значение флага ошибки

6.01 C ПОДПРОГРАММА ОПРЕДЕЛЕНИЯ ГОДНОСТИ ПАРАМЕТРА
 6.03 S ER=0; С УСТАНОВИТЬ ФЛАГ ОШИБКИ В 0
 6.05 S U=FADC (I); С ИЗМЕРИТЬ НАПРЯЖЕНИЕ ПО КАНАЛУ I
 6.07 I (U-TU-MA) 6.09, 6.09, 6.11
 6.09 I (U-TU-MI) 6.11, 6.13, 6.13
 6.11 S ER=1; С УСТАНОВИТЬ ФЛАГ ОШИБКИ В 1
 6.13 R ; С ВОЗВРАТ ИЗ ПОДПРОГРАММЫ

Здесь ER — флаг ошибки (параметр не укладывается в допуск); U — измеренное значение напряжения; I — номер канала АЦП; TU — значение параметра по ТУ; МА — положительное отклонение параметра; МІ — отрицательное отклонение параметра.

Допустим, что необходимо проверить, укладывается ли в необходимые допуски напряжение, измеренное по каналу 6 АЦП. Для этой цели в подпрограмму обслуживания одного из циклов должны быть вставлены следующие строки операторов:

14.10 S I=6; S TU=5; S MA=0,25; S MI=-0,25; D 6
 14.12 I (ER) 14.14, 14.16, 14.14
 14.14 T «ПАРАМЕТР U (»,I,«) НЕ В ДОПУСКЕ»
 14.16 C ПРОДОЛЖЕНИЕ ПРОГРАММЫ КОНТРОЛЯ

Интерпретатор языка ФОКАЛ с необходимыми встроенными функциями предоставляет широкие возможности пользователю в деле создания программного обеспечения АСК.

Особенности выполнения оператора цикла в языке ФОКАЛ. Оператор цикла FOR имеет две формы:

FOR A=B, C, D; оператор; оператор; ...; оператор или
или

FOR A=B, D ; оператор; оператор; ...; оператор

Переменной A, называемой *параметром цикла*, присваивается значение B, затем выполняется один оператор или несколько операторов, находящихся на этой же строке. Затем переменная A увеличивается на величину C и сравнивается с величиной D. Если $A \leq D$, то операторы выполняются снова. Этот процесс повторяется до тех пор, пока величина A не станет больше величины D, после чего начинают исполняться операторы, находящиеся на следующей строке.

Если величина C пропущена, то предполагается, что шаг увеличения равен единице. Величины B, C, D могут быть положительными или отрицательными. В любом случае операторы, входящие в цикл, будут исполнены по крайней мере один раз. Следует заметить, что при отрицательном значении величины C цикл будет повторяться бесконечно.

Если величины C и D пропущены, то оператор FOR будет выполняться как оператор SET.

Особо необходимо отметить случай, когда в цикл включен оператор IF. При этом происходит передача управления на строку с номером, указанным в операторе IF, оператор с этим номером выполняется и вновь происходит возврат в цикл, т. е. оператор FOR всегда выполняет заданное количество циклов.

Поясним сказанное на примере. Пусть задан одномерный массив со следующими значениями переменных:

A (1) = 0;
A (2) = 2;
A (3) = 3;
A (4) = 4;
A (5) = 0;
A (6) = 6;
A (7) = 0;
A (8) = 8

и программа

1.1 FI = 1, 8; I (A(I)) 1.5, 1.5
1.3 T «КОНЕЦ1»; Q
1.5 T «A (« I, »)=», A (I), !
1.7 T «КОНЕЦ2»

Результаты выполнения приведенной программы:

A (1) = 0
A (5) = 0
A (7) = 0
КОНЕЦ1 *

Строка с номером 1.7 не выполнится ни разу. Параметр цикла после окончания программы будет иметь значение $I=9$. В этом можно убедиться, если исполнить оператор

TI

В результате ЭВМ отпечатает:

9 *

Если необходимо прервать цикл до его естественного окончания, то можно воспользоваться следующим приемом. При выходе из цикла по оператору IF присваиваем параметру цикла его конечное значение плюс единица. В рассмотренную выше программу добавляем оператор присваивания в строку с номером 1.5:

1.1 F I = 1, 8; I (A(I)) 1.5, 1.5

1.3 T «КОНЕЦ1»; Q

1.5 T «A («, I, ») = », A(I), !; S I = 9

1.7 T «КОНЕЦ2»

Результат выполнения этой программы будет такой:

A(1) = 0
КОНЕЦ1 *

Чтобы при выходе из цикла исполнилась строка с номером 1.7, нужно этой строке передать управление с помощью оператора GO:

1.1 F I = 1, 8; I (A(I)) 1.5, 1.5

1.3 T «КОНЕЦ1»; Q

1.5 T «A («, I, ») = », A(I), !; S I = 9; G 1.7.

1.7 T «КОНЕЦ2»

Ниже приведен результат выполнения программы:

A(1) = 0
КОНЕЦ2 КОНЕЦ *

В языке ФОКАЛ допускаются также и вложенные циклы. Глубина вложения зависит от количества ячеек памяти, отведенных под стек.

Принцип распределения ячеек оперативной памяти для переменных в языке ФОКАЛ. Каждая переменная занимает в памяти четыре последовательно расположенных ячейки. В первой ячейке находится имя переменной — одна или две буквы; во второй — индекс (один или два в зависимости от размерности переменной); в третьей и четвертой ячейках — значение переменной в форме с плавающей точкой.

Для переменных в памяти резервируется блок от первой свободной ячейки до ячейки с именем ВОТТОМ (последняя ячейка оперативной памяти). Размеры блока динамически изменяются: чем больше места занимает текст программы, тем меньше становится блок ячеек памяти для переменных.

Переменные распределяются в отведенной области по *методу хеширования (метод рассеянной памяти)*. Суть его состоит в том, что переменные хранятся в памяти не в последовательных ячейках, а рассеиваются случайным образом по всему блоку. Каждая но-

вая переменная помещается таким образом, что ее присутствие или отсутствие в блоке может быть установлено позднее без поиска по всему блоку. Для размещения переменной вычисляется ее хеш-адрес (смешанный код). Смешанный код находят следующим образом.

Если x_1 — первая буква имени переменной; x_2 — вторая буква имени переменной; y_1 — первый индекс; y_2 — второй индекс, то вычисляется выражение $x_2 \cdot 2^8 + x_1 + y_2 \cdot 2^8 + y_1$, переставляются байты в полученном слове (команда SWAB). После этого вычисляется результат по модулю размера блока памяти, отведенного для переменных, и по модулю 10_8 , так как каждая переменная занимает четыре ячейки.

Рассмотрим алгоритм размещения переменной в памяти. При загрузке системы блок переменных обнуляется. Встретившаяся в программе переменная (с индексами) размещается в стеке и вычисляется ее хеш-адрес. Если по этому адресу ячейка свободна (содержит нуль), то происходит запись переменной; если же ячейка занята, то происходит сдвиг указателя на четыре слова и снова проверяется, свободна ли ячейка. Таким образом, коллизии разрешаются методом линейного поиска свободного места от вычисленного хеш-адреса. Если свободного места нет, то жертвуют переменной с нулевым значением. Если и этого нет, то выдается диагностическое сообщение: «Переполнение буфера переменных».

В случае большого количества переменных в программе или слишком малого места для них из-за большого текста программы по причинам, изложенным выше, может произойти значительное снижение скорости вычислений. Это обстоятельство нужно учитывать на практике. В качестве одной из мер может быть рекомендовано применение оператора вычеркивания переменных ERASE, если это допустимо по смыслу программы.

III. КОНСТРУИРОВАНИЕ И НАДЕЖНОСТЬ ИНФОРМАЦИОННО-УПРАВЛЯЮЩИХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

В разделе рассматриваются критерии надежности, методы оценки показателей надежности и пути повышения надежности ИУВС; формулируются требования к конструкции; рассматриваются конструкционные системы ИУВС, особенности элементной базы, особенности конструирования ячеек, блоков ИУВС, вопросы обеспечения теплового режима, устойчивости к воздействию механических факторов, рекомендации по обеспечению помехоустойчивости ИУВС.

4. ОБЕСПЕЧЕНИЕ НАДЕЖНОСТИ И ОСОБЕННОСТИ КОНСТРУИРОВАНИЯ МИКРОПРОЦЕССОРНЫХ ИНФОРМАЦИОННО-УПРАВЛЯЮЩИХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

4.1. Обеспечение надежности микропроцессорных информационно-управляющих вычислительных систем

Под *надежностью* изделия (элемента, узла, устройства, системы) понимается свойство последнего сохранять свое качество при определенных условиях эксплуатации в течение заданного промежутка времени, т. е. надежность — качество, развернутое во времени. Количественно надежность характеризуется рядом интервальных, интегральных и точечных показателей.

Невосстанавливаемые изделия — изделия, поведение которых существенно лишь до первого отказа, — характеризуются следующими количественными показателями надежности: интенсивностью отказов $\lambda(t)$; частотой отказов $f(t)$; вероятностью безотказной работы $P(t)$; вероятностью отказа $Q(t)$; наработкой до отказа T_o .

Восстанавливаемые изделия — изделия, эксплуатация которых допускает их многократный ремонт, — характеризуются следующими количественными показателями надежности: параметрами потока отказов $\omega(t)$ и потока восстановлений $\mu(t)$; функцией готовности $K_r(t)$; коэффициентом готовности K_r ; средним временем работы между двумя отказами t_{cp} ; средним временем восстановления t_v .

Если в процессе функционирования невосстанавливаемого изделия возможен ремонт отдельных его элементов при сохранении работоспособности изделия в целом за счет резерва или если надежность функционирования восстанавливаемого изделия оцени-

вается в интервале времени до первого отказа восстанавливаемого изделия в целом, то такие изделия характеризуются следующими количественными показателями надежности: вероятностью безотказной работы $P(t)$; вероятностью отказа $Q(t)$; наработкой до отказа T_0 ; параметрами потока отказов элементов изделия $\omega(t)$ и потока восстановлений элементов изделий $\mu(t)$.

Количественные показатели надежности невосстанавливаемых изделий. Интервальные показатели надежности — *вероятность безотказной работы* $P(t)$ и *вероятность отказа* $Q(t)$ — определяются как вероятности событий $P(t) = P\{\tau > t\}$ и $Q(t) = P\{\tau \leq t\}$ соответственно, где τ — случайный момент времени, в который происходит отказ. При этом $P(t) + Q(t) = 1$, $P(0) = 1$, $Q(0) = 0$, $P(\infty) \rightarrow 0$, $Q(\infty) \rightarrow 1$.

Точечный (локальный) показатель надежности — *интенсивность отказов* $\lambda(t)$ — определяется как вероятность невосстанавливаемого отказа изделия в единицу времени после некоторого момента времени при условии, что до этого момента времени отказ не возникал, т. е.

$$\lambda(t) = Q(t, t + \Delta t) / \Delta t. \quad (4.1)$$

При $\Delta t \rightarrow 0$ имеем $Q(t, t + \Delta t) = 1 - P(t, t + \Delta t) = 1 - P(t + \Delta t) / P(t) \approx -[P'(t) / P(t)] \Delta t + o(\Delta t^2)$. С учетом (4.1) интенсивность отказов

$$\lambda(t) = -P'(t) / P(t). \quad (4.2)$$

Интегрируя левую и правую части выражения (4.2) в пределах от 0 до t , нетрудно получить другую форму связи между вероятностью безотказной работы и интенсивностью отказов изделия:

$$\int_0^t \lambda(t) dt = - \int_0^t [P'(t) / P(t)] dt = - \int_0^t [\ln P(t)]' dt = - \ln P(t);$$

$$P(t) = \exp \left[- \int_0^t \lambda(t) dt \right]. \quad (4.3)$$

Интенсивность отказов $\lambda(t)$ — один из наиболее удобных количественных показателей надежности изделий электроники: интегральных схем, радиоэлектронных изделий (транзисторов, диодов, резисторов, конденсаторов и т. п.). Изменение интенсивности отказов $\lambda(t)$ во времени большинства изделий электронной техники имеет существенно нелинейный характер (рис. 4.1), тем не менее на большом по времени участке работы интенсивность отказов изделия обычно мало изменяется и принимается в практических расчетах постоянной.

Ниже приведены значения интенсивности отказов $\lambda(1/ч)$ наиболее часто применяемых изделий электронной техники [15]:

Резисторы:

постоянные композиционные	$5 \cdot 10^{-8}$
угольные пленочные	$5 \cdot 10^{-8}$
металлизированные пленочные	$5 \cdot 10^{-7}$
пленочные	$2 \cdot 10^{-8}$

проволочные прецизионные	$1 \cdot 10^{-6}$
переменные композиционные	$2 \cdot 10^{-6}$
с ведущей червячной передачей	$5 \cdot 10^{-6}$
Конденсаторы:	
постоянной емкости керамические	10^{-7}
электролитические, с алюминиевой фольгой	$2 \cdot 10^{-6}$
электролитические, танталовые, твердые	$4 \cdot 10^{-7}$
Диоды:	
кремниевые	$5 \cdot 10^{-8}$
германиевые	$8 \cdot 10^{-7}$
Зенера	$1 \cdot 10^{-7}$
Транзисторы:	
германиевые	$1 \cdot 10^{-7}$
кремниевые	$8 \cdot 10^{-8}$
Интегральные микросхемы:	
кремниевые цифровые	$1 \cdot 10^{-7} - 5 \cdot 10^{-7}$
линейные	$3 \cdot 10^{-7} - 6 \cdot 10^{-7}$
Реле герметичное с одним переключающим	
контактом	$3 \cdot 10^{-8}$
Геркон	$3 \cdot 10^{-9}$
Переключатель на контакт	$3 \cdot 10^{-9}$
Ручная пайка	$3 \cdot 10^{-10} - 6 \cdot 10^{-9}$
Пайка погружением	10^{-10}
Роликовая сварка	$2 \cdot 10^{-8}$
Сварные соединения	$2 \cdot 10^{-9}$
Монтаж накруткой	$4 \cdot 10^{-12}$
Соединители коаксиальные концевые	
(на штырь)	$1 \cdot 10^{-8}$
Штепсельный соединитель	$3 \cdot 10^{-9}$
Лампы:	
неоновые	$2 \cdot 10^{-7}$
накаливания	$1 \cdot 10^{-6}$
Электромоторы малой мощности	$5 \cdot 10^{-6}$
Двигатели шаговые	$3 \cdot 10^{-6}$

Следует помнить, что $\lambda(t)$, оставаясь постоянной во времени на основном участке работы, существенно зависит от условий эксплуатации изделия (климатических, механических и радиационных воздействий, электрической нагрузки и т. п.), т. е.

$$\lambda(t) = \lambda_0 \prod_{i=1}^{i=n} a_i,$$

где λ_0 — интенсивность отказов изделия при номинальных (лабораторных) условиях эксплуатации; a_i — поправочный коэффициент i -го эксплуатационного фактора.

Частота отказов $f(t)$ — плотность вероятности времени работы изделия до первого отказа:

$$f(t) = Q'(t). \quad (4.4)$$

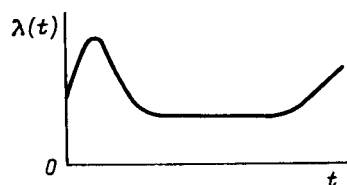


Рис. 4.1. Зависимость интенсивности отказов изделий от времени.

С вероятностью безотказной работы частота отказов связана соотношением

$$P(t) = 1 - \int_0^t f(t) dt. \quad (4.5)$$

Одной из часто используемых на практике интегральных характеристик надежности является *наработка до отказа* T_0 — математическое ожидание случайного момента времени τ , в который происходит отказ, т. е.

$$\begin{aligned} T_0 &= \int_0^\infty t Q'(t) dt = \int_0^\infty t [1 - P(t)]' dt = - \int_0^\infty t P'(t) dt = \\ &= -t P(t) \Big|_0^\infty + \int_0^\infty P(t) dt = \int_0^\infty P(t) dt. \end{aligned} \quad (4.6)$$

Учитывая свойство преобразования Лапласа, заметим, что если известно изображение вероятности безотказной работы $P(s)$, то

$$T_0 = P(s) \Big|_{s=0}. \quad (4.7)$$

Для экспоненциального закона вероятности безотказной работы изделия, т. е. при $\lambda(t) = \text{const}$ и $P(t) = \exp(-\lambda t)$, наработка на отказ равна величине, обратной интенсивности отказов:

$$T_0 = \int_0^\infty P(t) dt = \int_0^\infty \exp(-\lambda t) dt = 1/\lambda.$$

Количественные показатели надежности восстанавливаемых изделий. Точечный (локальный) показатель надежности — параметр *потока отказов* $\omega(t)$ — удельная вероятность появления хотя бы одного отказа в единицу времени, т. е.

$$\omega(t) = \lim_{\Delta t \rightarrow 0} P_0(t, t + \Delta t) / \Delta t,$$

где $P_0(t)$ — поток отказов — последовательность отказов, наступающих в случайные моменты времени.

Точечный (локальный) показатель надежности — *параметр потока восстановлений* $\mu(t)$ — удельная вероятность хотя бы одного восстановления в единицу времени, т. е.

$$\mu(t) = \lim_{\Delta t \rightarrow 0} P_{\text{в}}(t, t + \Delta t) / \Delta t,$$

где $P_{\text{в}}(t)$ — *поток восстановлений* — последовательность восстановлений, наступающих в случайные моменты времени.

Среди множества различных потоков отказов (восстановлений) в теории надежности особое место занимает *простейший поток отказов (восстановлений)*, поскольку наиболее важные для практики результаты получены в теории надежности именно для случая

простейших потоков. Это объясняется тем, что поведение изделия как системы массового обслуживания при простейших потоках отказов и восстановлений описывается системой дифференциальных уравнений с постоянными коэффициентами, методы решения которой хорошо разработаны.

Примечание. *Простейший* поток — поток, при котором события потока удовлетворяют одновременно условиям стационарности, ординарности и отсутствия последствий.

Поток называется *стационарным*, если вероятность возникновения k событий за время Δt зависит лишь от k и значения Δt , но не зависит от того, где на оси времени расположен участок Δt . *Ординарность* потока означает невозможность появления более одного события в данный момент времени. *Отсутствие последствий* означает, что вероятность наступления k событий в течение промежутка времени Δt не зависит от того, сколько было событий и как они распределялись до начала этого промежутка.

Простейший поток обладает следующими важными свойствами: 1) сумма простейших потоков образует также простейший поток с параметром $\Pi = \sum_{i=1}^n \Pi_i$, где Π_i — параметр i -го потока, n — число потоков; 2) закон распределения промежутков времени между соседними событиями является показательным, т. е. $a(t) = \Pi \exp(-\Pi t)$; 3) в течение времени t вероятность возникновения k событий подчиняется пуассоновскому закону $P_k(t) = \frac{(\Pi t)^k}{k!} \exp(-\Pi t)$, где Π — параметр потока событий; 4) для простейших потоков математическое ожидание времени между двумя событиями $T = 1 / \lim_{t \rightarrow \infty} \Pi(t)$.

Для практики расчетов надежностных показателей изделий очень важна связь между параметром потока отказов $\omega(t)$ восстанавливаемого изделия и интенсивностью отказов $\lambda(t)$ того же изделия, рассматриваемого как невозстанавливаемое, т. е. функционирующее до первого отказа.

Известно, что

$$\omega(t) = f(t) + \int_0^t f(t - \tau) \omega(\tau) d\tau, \quad (4.8)$$

где $f(t) = \lambda(t)P(t)$ — частота отказов невозстанавливаемого изделия.

Решение дифференциального уравнения (4.8) в предположении, что поток отказов соответствующего восстанавливаемого изделия простейший, дает $\omega(t) = \lambda(t)$.

Если на практике в большинстве случаев предполагается, что $\lambda(t) = \lambda = \text{const}$, то $\omega(t) = \lambda$, т. е. численно параметр потока отказов восстанавливаемого изделия равен интенсивности отказов соответствующего невозстанавливаемого изделия.

В предположении, что поток восстановлений изделия простейший, на практике параметр потока восстановлений изделия находят как $\mu(t) = 1/T_v = \text{const}$, где T_v — эмпирическое (опытное) значение среднего времени восстановления (ремонта) изделия.

Точечный (локальный) показатель восстанавливаемого изделия — *функция готовности* $K_r(t)$ — определяется как вероятность того, что в любой момент времени t изделие оказывается в работоспособном состоянии, т. е.

$$K_r(t) = \begin{cases} \sum_{i=0}^{k-1} P_i(t), & \text{если } k \leq \frac{N+1}{2}; \\ 1 - \sum_{j=1}^{N-k+1} P_j(t), & \text{если } k > \frac{N+1}{2}, \end{cases} \quad (4.9)$$

где $P_i(t)$ и $P_j(t)$ — вероятности нахождения системы в момент времени t в i -м исправном и j -м отказовом состояниях соответственно; $N+1$ — общее число; k — число исправных состояний изделия.

Предел функции готовности $K_r(t)$ при $t \rightarrow \infty$ называется коэффициентом готовности K_r и служит интегральным показателем надежности восстанавливаемого изделия:

$$K_r = \lim_{t \rightarrow \infty} K_r(t). \quad (4.10)$$

Так как коэффициент готовности является финальной вероятностью пребывания системы в исправном состоянии, то его можно вычислить, используя изображения Лапласа соответствующих вероятностей:

$$K_r = \lim_{t \rightarrow \infty} K_r(t) = \lim_{s \rightarrow 0} s K_r(s) = \lim_{s \rightarrow 0} s \sum_{i=0}^{k-1} P_i(s). \quad (4.11)$$

Обычно изображение функции готовности

$$K_r(s) = \sum_{i=0}^{k-1} P_i(s) = \frac{B_0 s^m + B_1 s^{m-1} + \dots + B_m}{s(A_0 s^n + A_1 s^{n-1} + \dots + A_n)},$$

причем $n \geq m$, поэтому

$$K_r = \lim_{s \rightarrow 0} s K_r(s) = B_m / A_n. \quad (4.12)$$

Интегральные показатели надежности — среднее время работы между двумя отказами t_{cp} и среднее время восстановления t_b , т. е. математическое ожидание времени между соседними отказами и восстановлениями соответственно. Показатели t_{cp} и t_b можно определить, если известны финальные вероятности пребывания изделия во всех возможных состояниях и интенсивности переходов из отказовых в предотказовые состояния:

$$t_{cp} = \frac{\sum_{l=1}^{l=L} P_l^\Phi}{\sum_{i=1}^{i=N} \left(\sum_{l=0}^{l=L} \mu_{L+i,l} \right) P_{L+i}^\Phi}; \quad (4.13)$$

$$t_b = \frac{\sum_{i=1}^{i=N} P_{L+i}^\Phi}{\sum_{i=1}^{i=N} \left(\sum_{l=0}^{l=L} \mu_{L+i,l} \right) P_{L+i}^\Phi}, \quad (4.14)$$

где $P_i^\Phi = \lim_{t \rightarrow \infty} P_i(t) = \lim_{s \rightarrow 0} sP_i(s)$ — финальная вероятность нахождения изделия в i -м рабочем состоянии, $i=0 \dots L$; $P_{L+i}^\Phi = \lim_{t \rightarrow \infty} P_{L+i}(t) = \lim_{s \rightarrow 0} sP_{L+i}(s)$ — финальная вероятность нахождения изделия в $(L+i)$ -м отказовом состоянии, $L+i=L+1, \dots, L+N$; $\mu_{L+i,i}$ — интенсивность перехода из $(L+i)$ -го отказового состояния в i -е предотказовое состояние; N — число отказовых состояний.

Иными словами, среднее время между двумя отказами t_{cp} определяется как отношение суммы финальных вероятностей нахождения системы в рабочих состояниях к сумме финальных вероятностей нахождения системы в отказовых состояниях, непосредственно связанных с рабочими состояниями и умноженных на соответствующую эквивалентную интенсивность восстановления. Последняя есть сумма интенсивностей восстановления, с которыми возможен переход из данного нерабочего состояния $L+i$, где $i=1-N$, во все связанные с ним рабочие состояния. При вычислении среднего времени восстановления t_b в числителе отношения берется сумма финальных вероятностей нахождения системы во всех отказовых состояниях, в знаменателе — сумма, аналогичная выражению t_{cp} . Заметим, что

$$\frac{t_{cp}}{t_{cp} + t_b} = \frac{\sum_{i=0}^{L=L} P_i^\Phi}{\sum_{i=0}^{L=L} P_i^\Phi + \sum_{i=1}^{i=N} P_{L+i}^\Phi} = \frac{\sum_{i=0}^{L=L} P_i^\Phi}{\sum_{i=0}^{i=N} P_i^\Phi} = K_r.$$

Классификация методов резервирования. При расчете надежности сложного изделия (узла, устройства, системы) полезно составить расчетную надежностную схему.

Если изделие состоит из N элементов и отказ изделия в целом наступает при отказе любого одного из его элементов, то говорят об основном (последовательном) соединении этих элементов, условное изображение расчетной надежностной схемы которого приведено на рис. 4.2, а. Так как каждый i элемент характеризуется в общем случае интенсивностью отказов $\lambda_i(t)$ и вероятностью безотказной работы $P_i(t) = \exp \left[- \int_0^t \lambda_i(t) dt \right]$, то вероятность безотказной работы изделия в целом

$$P(t) = \prod_{i=1}^{i=N} P_i(t) = \exp \left[- \sum_{i=1}^{i=N} \int_0^t \lambda_i(t) dt \right]. \quad (4.15)$$

Для экспоненциального закона вероятности безотказной работы отдельных элементов, т. е. при $\lambda_i = \text{const}$, вероятность безотказной работы изделия в целом

$$P(t) = \prod_{i=1}^{i=N} \exp [-\lambda_i t] = \exp \left[- \sum_{i=1}^{i=N} \lambda_i t \right] = \exp (-\lambda_{\Sigma} t)$$

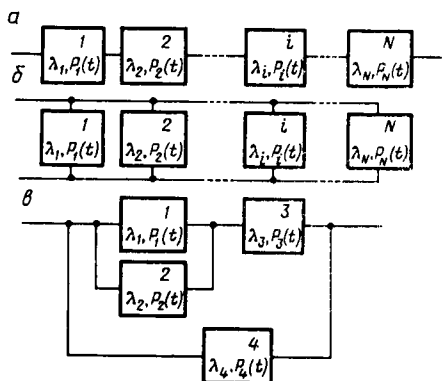


Рис. 4.2. Условное изображение в надежности схем последовательного (а), параллельного (б) и параллельно-последовательного (в) соединений изделий.

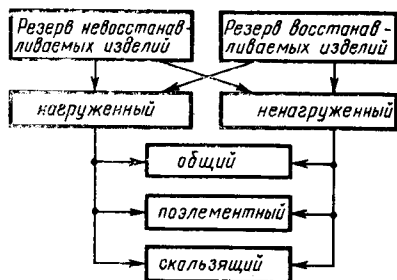


Рис. 4.3. Классификация способов резервирования изделий.

и наработка на отказ $T_0 = 1/\lambda_\Sigma$, где $\lambda_\Sigma = \sum_{i=1}^{i=N} \lambda_i$.

Если изделие состоит из N элементов и отказ изделия в целом наступает лишь в случае, когда откажут все N входящих в него элементов, то говорят о *параллельном соединении* этих элементов, расчетная надежность схема которого приведена на рис. 4.2, б. В этом случае вероятность безотказной работы

$$\begin{aligned}
 P(t) &= 1 - Q(t) = 1 - \prod_{i=1}^{i=N} Q_i(t) = 1 - \prod_{i=1}^{i=N} [1 - P_i(t)] = \\
 &= 1 - \prod_{i=1}^{i=N} \left\{ 1 - \exp \left[-\int_0^t \lambda_i(t) dt \right] \right\},
 \end{aligned} \quad (4.16)$$

а наработка на отказ T_0 и интенсивность отказов изделия вычисляются по (4.6) и (4.2) соответственно.

В общем случае изделие с точки зрения надежности может быть представлено *параллельно-последовательной* рабочей надежностной схемой, в которой последовательное соединение элементов отражает поведение элементов, отказ которых приводит к отказу изделия в целом, а параллельное соединение элементов отражает поведение элементов, отказ которых приводит к отказу изделия в целом, если откажут все элементы параллельного соединения. На рис. 4.2, в приведен пример параллельно-последовательной надежностной схемы.

Если надежностная схема изделия содержит параллельное соединение, т. е. если в изделии повышение надежности обеспечивается использованием функционально избыточных элементов, то в изделии имеет место *резерв*. Различают поэлементный, общий и скользящий резерв.

Поэлементный резерв — резерв, при котором функционально

избыточные элементы предусматриваются на случай отказа отдельных элементов или групп элементов изделия.

Общий резерв — резерв, при котором функционально избыточные элементы предусматриваются на случай отказа изделия в целом.

Скольльзящий резерв — резерв, при котором функции элемента неизбыточного изделия передаются резервному элементу только после отказа основного элемента, причем основные элементы резервируются одним или несколькими резервными элементами, каждый из которых может заменить любой отказавший основной элемент.

Общий, поэлементный и скольльзящий резерв в зависимости от того, в каком режиме (включенном или выключенном) используются резервные элементы до момента начала их функционирования вместо отказавших основных элементов, подразделяют на нагруженный (горячий) и ненагруженный (холодный) резерв.

В случае *нагруженного (горячего) резерва* резервные элементы находятся в том же рабочем режиме, что и основные.

В случае *ненагруженного (холодного) резерва* резервные элементы до момента их использования вместо основных элементов практически не несут нагрузок, находятся в выключенном состоянии.

Классификация способов резервирования невосстанавливаемых и восстанавливаемых изделий (узлов, устройств, систем ЭВМ) в зависимости от режима работы и способа включения резервных элементов приведена на рис. 4.3.

Методы расчета количественных показателей надежности изделий. Расчет показателей надежности невосстанавливаемых изделий с нагруженным общим или поэлементным резервом в предположении внезапных отказов элементов с постоянными во времени интенсивностями отказов элементов проводится с использованием соотношений (4.6), (4.15), (4.16).

Пример 4.1. Для расчетной надежностной схемы, показанной на рис. 4.2, а,

$$\begin{aligned} P(t) &= 1 - [1 - P_4(t)] \{1 - [1 - (1 - P_1(t))(1 - P_2(t))] P_3(t)\} = \\ &= 1 - [1 - \exp(-\lambda_4 t)] \{1 - [1 - (1 - \exp(-\lambda_1 t))(1 - \exp(-\lambda_2 t))] \times \\ &\times \exp(-\lambda_3 t)\} = \exp(-\lambda_4 t) + \exp[-(\lambda_1 + \lambda_3)t] + \exp[-(\lambda_2 + \lambda_3)t] + \\ &+ \exp[-(\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4)t] - \exp[-(\lambda_1 + \lambda_2 + \lambda_3)t] - \\ &- \exp[-(\lambda_1 + \lambda_3 + \lambda_4)t] - \exp[-(\lambda_2 + \lambda_3 + \lambda_4)t]; \end{aligned}$$

$$\begin{aligned} T_0 &= \int_0^{\infty} P(t) dt = \frac{1}{\lambda_4} + \frac{1}{\lambda_1 + \lambda_3} + \frac{1}{\lambda_2 + \lambda_3} + \frac{1}{\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4} - \\ &- \frac{1}{\lambda_1 + \lambda_2 + \lambda_3} - \frac{1}{\lambda_1 + \lambda_3 + \lambda_4} - \frac{1}{\lambda_1 + \lambda_2 + \lambda_4}. \end{aligned}$$

Расчетная надежностная схема для невосстанавливаемых изделий с нагруженным скольльзящим резервом (рис. 4.4, а) содержит n основных элементов и m резервных. В предположении, что вероятности безотказной работы всех элементов (основных и резервных) одинаковы и равны $p(t)$, вероятность безотказной работы изделий

в целом $P(t)$ определяется как вероятность события, что за время t в изделии произойдет не более m отказов, т. е.

$$P(t) = \sum_{k=0}^m C_{m+n}^k p^{m+n-k}(t) [1-p(t)]^k, \quad (4.17)$$

где $p(t)$ в случае внезапных отказов с постоянной во времени интенсивностью λ равна $\exp(-\lambda t)$.

На практике с целью повышения надежности сложных вычислительных устройств широкое распространение получило мажори-

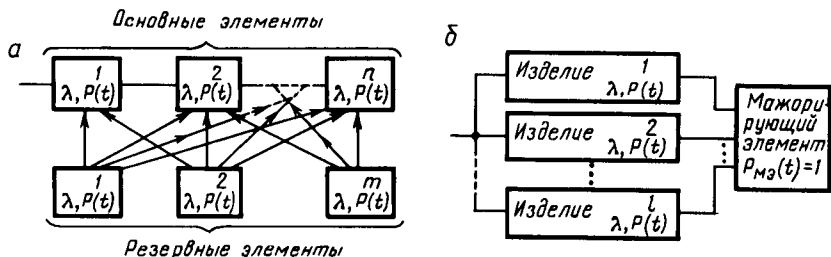


Рис. 4.4. Расчетные надежность схемы для случаев невозстанавливаемых изделий с нагруженным скользящим резервом (а) и мажорирования (б).

рование, которое можно рассматривать как частный случай скользящего нагруженного резервирования.

При *мажорировании* изделие l -кратно резервируется; причем l нечетно. Результат работы всех изделий сравнивается в специальном устройстве — мажорирующем элементе (рис. 4.4, б) — и за истинное значение принимается такое, которое имеет место на выходе большинства изделий, т. е. на выходе $(l-1)/2 + 1$ изделий. Вероятность безотказной работы l -кратно мажорированного изделия в предположении, что мажорирующий элемент абсолютно надежен, можно оценить, используя соотношение (4.17), если в нем положить $m+n=l$, $m=(l-1)/2$, т. е.

$$P_l(t) = \sum_{k=0}^{\frac{l-1}{2}} C_l^k p^{l-k}(t) [1-p(t)]^k.$$

При $l=3, 5$ и 7 соответственно получим $P_3(t)=p^2(t)[3-2p(t)]$; $P_5(t)=p^3(t)[10-6p^2(t)-15p(t)]$; $P_7(t)=p^4(t)[35-84p(t)+70p^2(t)-20p^3(t)]$.

Показатели надежности невозстанавливаемых изделий при ненагруженном общем, поэлементном и скользящем резерве либо восстанавливаемых изделий при ненагруженном или нагруженном общем, поэлементном и скользящем резерве можно вычислить, описывая «старение» таких изделий случайным марковским процессом с дискретными состояниями.

Случайный процесс называется *марковским случайным процес-*

сом (процессом без последствия), если дальнейшее поведение процесса определяется его состоянием в данный момент времени и не зависит от его предыстории. Случайный марковский процесс называется *процессом с дискретными состояниями*, если возможные состояния изделия $S_1, S_2, S_3, \dots$ можно перечислить (перенумеровать) одно за другим, а сам процесс состоит в том, что время от времени изделие S скачком (мгновенно) переходит из одного состояния в другое под действием простейших потоков отказов и восстановлений отдельных элементов изделия.

При анализе поведения изделия во времени в процессе износа (старения) удобно пользоваться *графом состояний*, содержащим столько вершин, сколько различных состояний возможно у изделия. *Ребра графа состояний* отражают возможные переходы из некоторого состояния во все остальные в соответствии с параметрами потоков отказов или восстановлений. Если для каждого состояния изделия, другими словами, для каждой вершины графа вычислить вероятность нахождения изделия именно в этом состоянии в любой произвольный момент времени $P_i(t)$, то, зная эти вероятности, можно оценить интересующие на практике показатели надежности, используя соотношения (4.2), (4.6), (4.9) — (4.14).

Связь между вероятностями нахождения изделия во всех его возможных состояниях в свою очередь выражается *системой дифференциальных уравнений Колмогорова*. Структура уравнений Колмогорова построена по вполне определенному правилу: *в левой части каждого уравнения Колмогорова записывается производная вероятности нахождения изделия в рассматриваемом состоянии (вершине графа), а правая часть содержит столько членов, сколько ребер графа состояний связано с данной вершиной графа. Если ребро направлено из данной вершины, соответствующий член имеет знак минус, если в данную вершину — знак плюс. Каждый член равен произведению параметра потока отказа (восстановления), связанного с данным ребром, на вероятность нахождения в той вершине графа, из которой исходит ребро*. Система уравнений Колмогорова включает столько уравнений, сколько вершин в графе состояний изделия. Решение системы уравнений Колмогорова при конкретных начальных условиях, определяемых спецификой эксплуатации изделия, дает значения искомых вероятностей $P_i(t)$.

В общем случае применение теории случайных марковских процессов к решению задач оценки показателей надежности восстанавливаемых и невосстанавливаемых изделий включает: 1) составление списка всех возможных состояний изделия; 2) вычисление параметров потоков отказов и восстановлений для каждого состояния; 3) составление графа состояний; 4) запись системы дифференциальных уравнений Колмогорова; 5) решение системы уравнений Колмогорова и определение количественных показателей надежности по соотношениям (4.2), (4.6), (4.9) — (4.14).

Пример 4.2. Применение теории марковских случайных процессов для расчета показателей надежности восстанавливаемых и невосстанавливаемых изделий рассмотрим на примере вычислительной системы, состоящей из двух микроЭВМ, рабо-

тающих одновременно, и одной резервной, используемой в режиме ненагруженного скользящего резерва (рис. 4.5, а). Предположим, что система обслуживается одной ремонтной бригадой в порядке прямого приоритета.

Перед составлением графа состояний системы полезно перечислить возможные состояния системы в период ее эксплуатации, сведя их в таблицу (например, табл. 4.1 для системы, изображаемой на рис. 4.5, б).

Граф состояний, описывающий износ (старение) такой системы, как восстанавливаемой в предположении, что поток отказов и восстановлений простейший

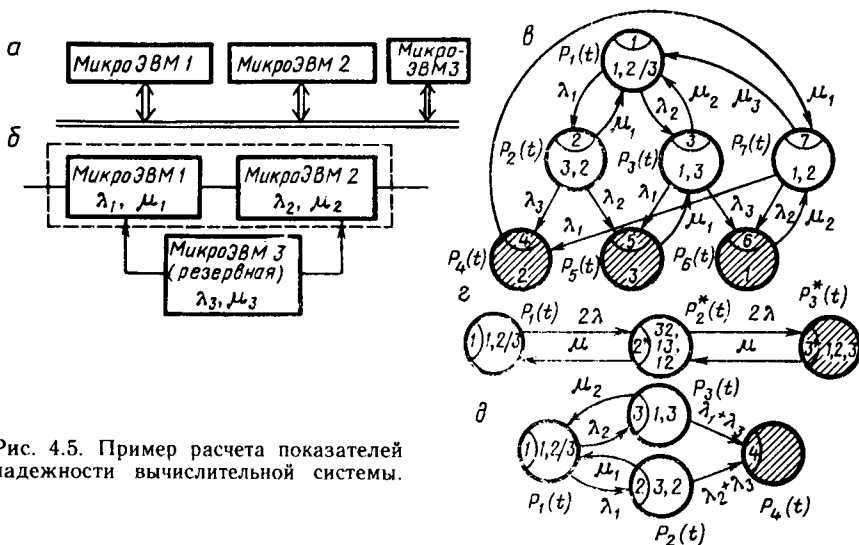


Рис. 4.5. Пример расчета показателей надежности вычислительной системы.

и одна ремонтная бригада ремонтирует в любой момент времени только одну микроЭВМ в порядке прямого приоритета, имеет вид, приведенный на рис. 4.5, в.

На рисунке вершины графа состояний, соответствующие отказовым состояниям, заштрихованы. В изображении вершин указываются номер состояния, работающие и резервные микроЭВМ в этом состоянии. Например, вершина 1 соответствует

Таблица 4.1

Номер состояния (вершины графа)	МикроЭВМ			Состояние системы в целом
	работающие	резервные	отказавшие	
1	1,2	3	Рабочее	Рабочее
2	3,2	—	1	«
3	1,3	—	2	«
4	2	—	1,3	Отказовое
5	3	—	1,2	«
6	1	—	3,2	«
7	1,2	—	3	Рабочее

состоянию системы, когда работают первая и вторая микроЭВМ, а третья находится в ненагруженном резерве; вершина 7 — когда работают первая и вторая микроЭВМ, а третья отказала и восстанавливается ремонтной бригадой.

Система дифференциальных уравнений Колмогорова, связывающая вероятности нахождения системы в любом из возможных ее состояний, в этом случае имеет вид

$$\left. \begin{aligned} P_1'(t) &= -(\lambda_1 + \lambda_2)P_1(t) + \mu_1 P_2(t) + \mu_2 P_3(t) + \mu_3 P_7(t); \\ P_2'(t) &= -(\mu_1 + \lambda_2 + \lambda_3)P_2(t) + \lambda_1 P_1(t); \\ P_3'(t) &= -(\mu_2 + \lambda_1 + \lambda_3)P_3(t) + \lambda_2 P_1(t) + \mu_1 P_5(t); \\ P_4'(t) &= -\mu_1 P_4(t) + \lambda_3 P_2(t) + \lambda_1 P_7(t); \\ P_5'(t) &= -\mu_1 P_5(t) + \lambda_2 P_2(t) + \lambda_1 P_3(t); \\ P_6'(t) &= -\mu_2 P_6(t) + \lambda_3 P_3(t) + \lambda_2 P_7(t); \\ P_7'(t) &= -(\mu_3 + \lambda_2 + \lambda_1)P_7(t) + \mu_2 P_6(t) + \mu_1 P_4(t). \end{aligned} \right\} \quad (4.18)$$

Решая систему (4.18) при начальных условиях $P_1(0)=1$, $P_i(0)=0$, $i=2, \dots, 7$, можно вычислить вероятности нахождения системы в любом возможном состоянии $P_i(t)$ и такие показатели надежности восстанавливаемой системы, как функция готовности $K_r(t)$, коэффициент готовности K_r , среднее время работы между двумя отказами t_{cp} , среднее время восстановления t_b .

В предположении, что все микроЭВМ одинаковы и, следовательно, $\lambda_1=\lambda_2=\lambda_3=\lambda$ и $\mu_1=\mu_2=\mu_3=\mu$, система (4.18) примет вид

$$\left. \begin{aligned} P_1'(t) &= -2\lambda P_1(t) + \mu P_2^*(t); \\ [P_2^*(t)]' &= -(\mu + 2\lambda)P_2^*(t) + 2\lambda P_1(t) + \mu P_3^*(t); \\ [P_3^*(t)]' &= -\mu P_3^*(t) + 2\lambda P_2^*(t), \end{aligned} \right\} \quad (4.19)$$

где $P_2^*(t)=P_2(t)+P_3(t)+P_7(t)$; $P_3^*(t)=P_4(t)+P_5(t)+P_6(t)$.

Такому частному случаю соответствует граф, приведенный на рис. 4.5, г. Здесь вершине 2^* соответствует состояние, когда работают две микроЭВМ, резерва нет, причем в состояние, когда работают первая и вторая микроЭВМ, система переходит лишь из отказового состояния.

Применяя преобразование Лапласа к системе дифференциальных уравнений (4.19) с учетом начальных условий $P_1(0)=1$, $P_2^*(0)=P_3^*(0)=0$, получим систему алгебраических уравнений

$$\left. \begin{aligned} sP_1(s) - 1 &= -2\lambda P_1(s) + \mu P_2^*(s); \\ sP_2^*(s) &= 2\lambda P_1(s) - (\mu + 2\lambda)P_2^*(s) + \mu P_3^*(s); \\ sP_3^*(s) &= 2\lambda P_2^*(s) - \mu P_3^*(s). \end{aligned} \right\}$$

Решение ее дает изображение вероятностей:

$$P_1(s) = [s^2 + 2s(\mu + \lambda) + \mu^2] / \{s[s^2 + 2s(\mu + 2\lambda) + 4\lambda^2 + \mu^2 + 2\lambda\mu]\}; \quad (4.20)$$

$$P_2^*(s) = [2\lambda(s + \mu)] / \{s[s^2 + 2s(\mu + 2\lambda) + 4\lambda^2 + \mu^2 + 2\lambda\mu]\}; \quad (4.21)$$

$$P_3^*(s) = 4\lambda^2 / \{s[s^2 + 2s(\mu + 2\lambda) + 4\lambda^2 + \mu^2 + 2\lambda\mu]\}. \quad (4.22)$$

В соответствии с определением функции готовности (4.9) значение функции готовности для данной системы

$$K_r(t) = 1 - P_3^*(t),$$

где

$$P_3^*(t) = 4\lambda^2 \left[\frac{1}{4\lambda^2 + \mu^2 + 2\lambda\mu} + \frac{1}{(A+B)2B} \exp[(A+B)t] \right] - \\ - \frac{1}{(A+B)2B} \exp[(A-B)t] - \text{оригинал изображения } P_3^*(s); \quad A = -(\mu + 2\lambda); \quad B = \\ = \sqrt{2\lambda\mu}.$$

В соответствии с определением (4.10) коэффициент готовности

$$K_r = \lim_{t \rightarrow \infty} [1 - P_3^*(t)] = (\mu + 2\lambda)\mu / (4\lambda^2 + \mu^2 + 2\lambda\mu).$$

Среднее время работы между двумя отказами t_{cp} и среднее время восстановления t_b вычисляем, используя выражение (4.13) и (4.14). В частности, из выражений (4.20) — (4.22) находим финальные вероятности:

$$P_1^\Phi = \mu^2 / (4\lambda^2 + \mu^2 + 2\lambda\mu); P_2^{*\Phi} = 2\lambda\mu / (4\lambda^2 + \mu^2 + 2\lambda\mu); \\ P_3^{*\Phi} = 4\lambda^2 / (4\lambda^2 + \mu^2 + 2\lambda\mu).$$

Тогда, поскольку для рассматриваемой системы $L=2$, $M=N=1$, находим

$$t_{cp} = \frac{\sum_{l=1}^{l=2} P_l^\Phi}{\sum_{i=1} \left(\sum_{l=1} \mu_{2+i,l} \right) P_{2+i}^\Phi} = \frac{P_1^\Phi + P_2^{*\Phi}}{\mu P_3^{*\Phi}} = \frac{\mu}{2\lambda} \left(1 - \frac{\mu}{2\lambda} \right); \\ ; \\ t_B = \frac{\sum_{i=1} P_{2+i}^\Phi}{\sum_{i=1} \left(\sum_{l=1} \mu_{2+i,l} \right) P_{2+i}^\Phi} = \frac{P_3^{*\Phi}}{\mu P_3^{*\Phi}} = \frac{1}{\mu}.$$

Если поведение рассматриваемой системы нас интересует лишь до первого отказа системы в целом, т. е. система рассматривается как невосстанавливаемая, то граф состояний системы, очевидно, изменится. Данная система никогда не попадает в состояние, когда работают первая и вторая микроЭВМ, а третья находится в ремонте, поэтому все различные отказовые состояния можно заменить одним (рис. 4.5, δ). Система дифференциальных уравнений Колмогорова при этом имеет вид

$$\left. \begin{aligned} P_1'(t) &= -(\lambda_1 + \lambda_2)P_1(t) + \mu_1 P_2(t) + \mu_2 P_3(t); \\ P_2'(t) &= -(\mu_1 + \lambda_2 + \lambda_3)P_2(t) + \lambda_1 P_1(t); \\ P_3'(t) &= -(\mu_2 + \lambda_1 + \lambda_3)P_3(t) + \lambda_2 P_1(t); \\ P_4'(t) &= (\lambda_2 + \lambda_3)P_2(t) + (\lambda_1 + \lambda_3)P_3(t). \end{aligned} \right\} \quad (4.23)$$

Решая систему (4.23) операторным методом в предположении, что $\lambda_1 = \lambda_2 = \lambda$, $\mu_1 = \mu_2 = \mu$ и начальные условия $P_1(0)=1$, $P_2(0)=P_3(0)=P_4(0)=0$, получаем изображение вероятности нахождения системы в рабочем состоянии в любой момент времени:

$$P(s) = P_1(s) + P_2(s) + P_3(s) = (s + 4\lambda + \mu) / [s^2 + s(\mu + 4\lambda) + 4\lambda^2].$$

Переходя к оригиналу, определим

$$P(t) = -\frac{8\lambda^2}{(A+B)B} \exp \left[-\frac{(A+B)t}{2} \right] + \frac{8\lambda^2}{B(A-B)} \exp \left[-\frac{(A-B)t}{2} \right],$$

где $A = \mu + 4\lambda$; $B = \sqrt{\mu^2 + 8\lambda\mu}$.

Используя соотношение (4.7), вычислим наработку на отказ системы:

$$T_o = P(s)|_{s=0} = \frac{4\lambda + \mu}{4\lambda^2} = \frac{1}{\lambda} \left(1 + \frac{\mu}{4\lambda} \right). \quad (4.24)$$

Из (4.24) нетрудно видеть, что наработка на отказ системы в целом с восстановлением отдельных микроЭВМ в процессе эксплуатации в $\mu/(4\lambda)$ раз больше наработки на отказ системы без восстановления отдельных микроЭВМ (случай $\mu=0$). Так, если принять наработку на отказ отдельной микроЭВМ 100 ч, а время восстановления 0,5 ч, то наработка на отказ системы возрастет в 51 раз. Это очень важное замечание, если учесть, что наработка на отказ изделия для случая скользящего ненагруженного резерва без восстановления отказавших элементов

$$T_o = \frac{m+1}{n} \frac{1}{\lambda}, \quad (4.25)$$

где λ — параметр потока отказов одного элемента изделия; n и m — число основных и резервных элементов изделия; весьма медленно растет с увеличением числа резервных элементов в изделии. Применение восстановления отказавших элементов (где это допустимо по условиям эксплуатации) является одним из эффективных путей повышения надежности вычислительных систем.

Пути повышения надежности МП ИУВС. Надежность ИУВС растет по мере совершенствования элементной базы. В частности, применение микропроцессорных наборов БИС, БИС ЗУ, матричных кристаллов ведет к уменьшению числа элементов и межсоединений (паек, сварок) в средствах вычислительной техники. Однако из-за тенденции постоянного увеличения функциональных возможностей средств вычислительной техники число элементов в системах остается достаточно большим.

Если резерв в ИУВС отсутствует, то практически невозможно достичь приемлемых показателей надежности. Так как в инженерной практике считают, что вычислительное средство надежно, если в течение некоторого интервала времени Δt вероятность безотказной работы $P(\Delta t) \geq 0,997$, то при $\lambda = 10^{-4} - 10^{-7}$ 1/ч и числе элементов в системе $n = 10^4 - 10^5$ время безотказной работы в указанном выше смысле составляет лишь единицы часов:

$$P(\Delta t) = \exp[-\lambda n \Delta t] \approx 1 - \lambda n \Delta t \geq 0,997; \Delta t \leq 0,003/(\lambda n). \quad (4.26)$$

Например, при $\lambda = 10^{-7}$ и $n = 10^4$ $\Delta t \leq 3$ ч. Так как существенно уменьшить n и λ нельзя, то и увеличить Δt без применения резерва практически не удастся. Опыт эксплуатации электронной техники показывает, что интенсивность отказов элементов при ее хранении примерно на порядок ниже, чем при работе в реальных условиях эксплуатации, т. е. $\lambda_{xp} \approx 0,1\lambda$ [15]. Это означает, что применение ненагруженного резерва без восстановления может снизить $n\lambda\Delta t$ не более чем в 10 раз. Такой подход дает возможность создавать средства вычислительной техники, работающие безотказно практически лишь в течение десятков часов, что не решает проблемы резкого повышения надежности ИУВС.

Теоретически введением избыточности в структуру ИУВС можно создать сколь угодно надежную вычислительную систему. Но не всегда это практически выполнимо. Для подтверждения этого тезиса сравним количественные показатели надежности: 1) нерезервированной ИУВС, характеризуемой параметром потока отказов $\omega = \lambda$ и параметром потока восстановлений μ ; 2) дублированной ИУВС (общее резервирование) с восстановлением отказавших ЭВМ; 3) дублированной ИУВС (поэлементное резервирование) с восстановлением отказавших элементов; 4) ИУВС, состоящей из n основных и m резервных равнонадежных ЭВМ (предполагается, что параметр потока отказов каждой ЭВМ равен λ , восстановление отказавших ЭВМ с параметром μ возможно в процессе работы системы).

Для нерезервированной ИУВС

$$P(t) = \exp(-\lambda t); K_r = \mu/(\mu + \lambda); t_{cp} = 1/\lambda;$$

для ИУВС с общим нагруженным дублированием и восстановлением

$$P_{\text{общ}}(t) = \frac{4\lambda^2}{B(A-B)} \exp \left[-\frac{(B-A)t}{2} \right] - \frac{4\lambda^2}{B(A+B)} \times \\ \times \exp \left[-\frac{(B+A)t}{2} \right]; K_{\text{г общ}} = \frac{\mu^2 + 2\lambda\mu}{\mu^2 + 2\lambda\mu + 2\lambda^2}; \\ t_{\text{ср общ}} = \frac{1}{\lambda} \left(1 + \frac{\mu}{2\lambda} \right),$$

где $A = \mu + 3\lambda$; $B = \sqrt{\mu^2 + \lambda^2 + 6\mu\lambda}$.

Несмотря на то что общее нагруженное дублирование с восстановлением отказавших ЭВМ ведет к росту $K_{\text{г}}$, $t_{\text{ср}}$, тем не менее вероятность безотказной работы порядка 0,997 обеспечивается лишь в течение десятка часов. Действительно, так как наработка на отказ современных ЭВМ не превышает нескольких сотен часов, а среднее время восстановления не меньше 0,5 ч при $\lambda = 0,01 - 0,004$ 1/ч и $\lambda/\mu = 0,005 - 0,002$, то

$$t_{\text{ср общ}} = \frac{1}{\lambda} \left(1 + \frac{\mu}{2\lambda} \right) = (1 - 6,25) 10^4 \text{ ч.}$$

При этом интервал времени Δt , в течение которого $P_{\text{общ}}(\Delta t) \geq 0,997$, составляет лишь 10—25 ч, что по крайней мере на два порядка ниже, чем требуется в ряде прикладных задач управления техническими объектами. Чтобы достичь $\Delta t = 1000$ ч, нужно либо уменьшить $t_{\text{в}}$ до 10 мин, либо при $t_{\text{в}} = 0,5$ ч и $t_{\text{ср}} = 100$ ч применять четырехкратный общий резерв. Такая высокая кратность резервирования вряд ли реализуема на практике по экономическим соображениям. Из приведенного анализа следует важный *практический вывод: обеспечить высокую надежность ИУВС путем общего нагруженного резерва не представляется возможным по экономическим соображениям.*

Известно, что наибольший эффект дает поэлементное резервирование. Для средств вычислительной техники с поэлементным нагруженным дублированием и восстановлением отказавших элементов

$$t_{\text{ср эл}} \approx n\mu/(2\lambda^2), \quad (4.27)$$

где n — число элементов в ЭВМ.

Как следует из (4.27), наработка на отказ такой системы в n раз больше по сравнению с наработкой на отказ в случае общего нагруженного дублирования. Так как число элементов n современных ИУВС велико, то наработка на отказ восстанавливаемых ИУВС с поэлементным дублированием соизмерима с долговечностью элементов. Расчеты показывают, что вероятность безотказной работы такой ЭВМ настолько велика, что ИУВС можно считать идеально надежной в течение тысяч часов. Использование поэлементного резервирования для обеспечения надежности ИУВС, несмотря на высокую эффективность, крайне затруднительно. Это объясняется тем, что его техническая реализация практически невозможна для современной элементной базы.

Построение ИУВС как вычислительных систем, состоящих из n основных и m ненагруженных резервных равнонадежных ЭВМ

с параметром потока отказов λ , восстановление которых с параметром μ допускается в процессе работы системы, является *практически осуществимым путем создания высоконадежных ИУВС*. Уже при $m=2-3$ и $\mu/(n\lambda) \geq 10^2$ наработка на отказ $t_{cp} \approx 10^4 - 10^6$ ч, так как при $\mu/\lambda \gg 1$

$$t_{cp} \approx [\mu/(n\lambda)]^m (1/\lambda).$$

Вытекающими из выше приведенных рассуждений *практическими рекомендациями по построению надежных ИУВС следует считать: введение структурной избыточности путем построения узлов, устройств и вычислительных систем с изменяющейся архитектурой при возникновении отказов отдельных изделий; ремонт отказавших изделий без нарушения функционирования ИУВС в целом; создание бессбойного математического обеспечения.*

Примеры высоконадежных МП ИУВС. К ИУВС, используемым для исследования космоса, предъявляются особо жесткие требования по надежности.

В аэрокосмических ИУВС, разрабатываемых в начале 80-х годов без применения микропроцессоров, жесткие ограничения по габаритным размерам, массе и мощности потребления не позволяли использовать многопроцессорную архитектуру с заменой отказавших процессоров. Повышение надежности ИУВС достигалось, например, в аэрокосмической ЭВМ STAR [24] применением: резервных модулей с заменой отказавшего модуля путем надлежащей коммутации лишь цепей питания; специального процессора диагностики и устранения неисправностей; внутреннего самоконтроля процессора диагностики путем трехкратной избыточности; относительно сложных методов контроля вычислений (арифметика в остаточных классах); корректирующих кодов; повторных просчетов для обнаружения и коррекции сбоев.

Успехи электронной промышленности и схемотехники, в первую очередь микропроцессоры и КМДП БИС, позволили разделять ИУВС на модули на уровне микропроцессоров. Это привело к использованию методов обнаружения и устранения отказов и сбоев: 1) попарная синхронная работа микропроцессоров, которые автоматически заменяются новой парой микропроцессоров при обнаружении несовпадения результатов работы первой пары; 2) работа микропроцессоров тройками с использованием голосования для локализации и коррекции ошибок (трехкратное мажорирование); 3) применение микропроцессоров со встроенными средствами самопроверки и увеличение числа микропроцессоров в системе (мультимикропроцессорные ИУВС) с целью замены отказавших процессоров резервными.

Пример 4.3. Пример архитектуры, устойчивой к отказам аэрокосмической ЭВМ (МП ИУВС распределенного типа) [18], приведен на рис. 4.6, а, где 1 — командный МП; 2 — МП преобразования форматов; 3 — резерв МП; 4 — МП телекамеры; 5 — МП научных инструментов; 6 — МП управления инженерными службами; 7 — МП управления положением космического корабля; 8 — ЗУ; 9 — МП связи; 10 — троированная магистраль системы.

Архитектура микропроцессора со встроенными средствами проверки, используемого для построения всех МП системы, приведена на рис. 4.6, б, где 1 — контроллер магистрали; 2 — резервированный адаптер магистрали; 3 — корректируемое по Хеммингу ЗУ; 4 — управляющая часть микропроцессора; 5 — резервированный процессорный блок; 6 — резервированное УВВ; 7 — троированная магистраль системы; 8 — внутренняя магистраль МП; 9 — магистраль ввода-вывода.

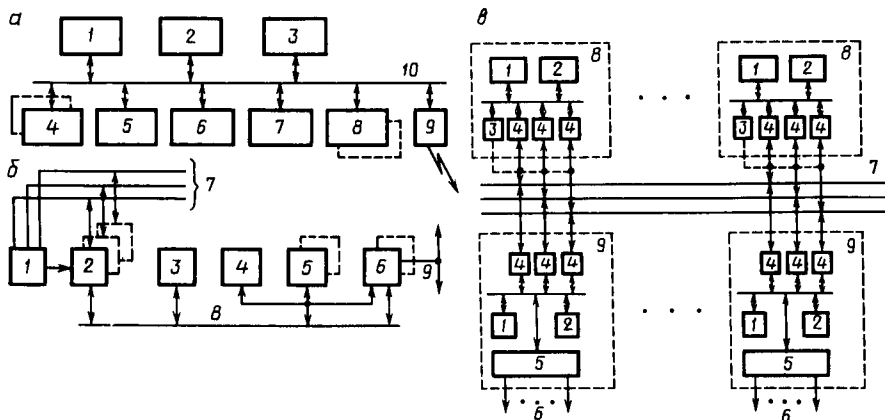


Рис. 4.6. Пример архитектуры, устойчивой к отказам аэрокосмической МП ИУВС.

Ответ самопроверяемого модуля на внутренний отказ заключается в выставлении флажка — индикатора отказа. Схема всей системы с использованием вышеописанных микропроцессоров приведена на рис. 4.6, в, где 1 — МП; 2 — ЗУ; 3 — контроллер магистрали; 4 — адаптер магистрали; 5 — УВВ; 6 — резервированные магистрали ввода-вывода; 7 — троированная магистраль системы; 8 — модуль высокого уровня; 9 — терминальный модуль.

Из рассмотренных примеров видно, что генеральная стратегия обеспечения надежности ИУВС — применение ИС большой и сверхбольшой степени интеграции, автоматический поиск неисправностей и использование резервных элементов путем реконфигурации архитектуры системы.

4.2. Особенности конструирования микропроцессорных информационно-управляющих вычислительных систем

Требования к конструкции и конструкционные системы ИУВС. Так как ИУВС — одно из звеньев системы управления, работающей, как правило, в масштабе реального времени в цехе, на буровой, корабле, самолете, космическом корабле, то при конструировании ИУВС необходимо учитывать следующие особенности их использования: 1) широкий спектр механических (удары, вибрации, ускорения), климатических (температура, влажность, давление окружающей среды), а иногда и радиационных факторов воздействия; 2) повышенная надежность (безотказность, ремонтпригодность); 3) встраиваемость ИУВС в аппаратуру, в ряде случаев учет формы контейнера носителя; 4) жесткие требования к габаритным размерам и массе; 5) территориальная распределенность элементов

ИУВС на десятки, сотни и даже тысячи метров; 6) работа в условиях высокого уровня внешних электромагнитных помех; 7) малая стоимость; 8) модульность построения ИУВС, позволяющая подключать различные модули в любых местах единого интерфейса.

Конструирование ИУВС при выполнении комплекса подобных требований представляет собой сложнейшую инженерную задачу. Конкретное конструктивное решение существенно зависит от требований технического задания (ТЗ).

Устойчивость к механическим воздействиям обеспечивается уменьшением габаритных размеров и массы; подбором конструктивных материалов, элементов, наиболее устойчивых к механическим воздействиям; увеличением собственных резонансных частот элементов конструкции путем введения дополнительных конструктивных элементов, увеличивающих жесткость; выбором способа крепления радиоэлектронных изделий (РЭИ); введением дополнительных точек крепления; применением амортизаторов и амортизационных оснований.

Жесткий тепловой режим в конструкции из-за широкого диапазона изменения температуры окружающей среды, высокой плотности компоновки и, как следствие, из-за высоких мощностей рассеяния требует принятия специальных мер: использования РЭИ, рассчитанных на работу в широком температурном диапазоне; применения теплоотводящих и теплопроводящих конструктивных элементов, обеспечивающих малое тепловое сопротивление для всех участков передачи теплоты, принудительного охлаждения (вентиляция, жидкостное охлаждение, тепловые трубки), принудительного подогрева.

Ремонтопригодность повышается обеспечением доступа ко всем элементам и межсоединениям, сохранением неразрывности электрических цепей при профилактических и ремонтных работах.

Надежность обеспечивается с помощью интегральных схем высокой степени интеграции, подбором соответствующих элементов, уменьшением числа разъемных соединений (блоки книжной, планшетной конструкции).

Малые габаритные размеры и масса обеспечиваются интегральными схемами средней и большой степени интеграции, бескорпусными гибридными микросборками, микроминиатюрными РЭИ и конструктивными элементами, современными методами монтажа с высокой плотностью компоновки.

Необходимая помехозащищенность достигается тщательной проработкой трассировки печатных плат, выбором конструкции цепей питания, экранированием элементов и линий связи, применением согласованных радиочастотных линий связи и волоконно-оптических линий связи, введением информационной избыточности при передаче и хранении цифровой информации.

Особенности конструкционной системы (КС) микроЭВМ — совокупности уровней базовых конструкций, организованных в определенной соподчиненности на основе единого размерного модуля

(2,54 мм), оптимальной технологии производства с учетом функциональных, механических и тепловых факторов, а также требований технической эстетики — подробно рассмотрены в [25].

Конструкционная система микроЭВМ широкого применения включает в себя следующие уровни базовых конструкций: плату, корпуса частичный, комплектный и комплексный.

Конструкционная система специализированной ИУВС, рассчитанной на работу в жестких условиях эксплуатации, в зависимости от конструктивно-технологического исполнения включает в себя следующие уровни:

для корпусных элементов: 1) ИС, СИС, БИС, РЭИ; 2) ячейку; 3) блок;

для бескорпусных элементов: 1) бескорпусные РЭИ, ИС, БИС; 2) микросборку; 3) ячейку; 4) блок.

Анализируя многообразие конструкций ИУВС, можно выделить следующие характерные варианты конструкций блоков: 1) разъемную; 2) книжную; 3) кассетную; 4) цилиндрической формы и др.

Пример 4.4. Блок разъемной конструкции с принудительным воздушным охлаждением, предназначенный для работы в сложных условиях окружающей среды, приведен на рис. 4.7, а. Каждая ячейка 3, представляющая собой функционально

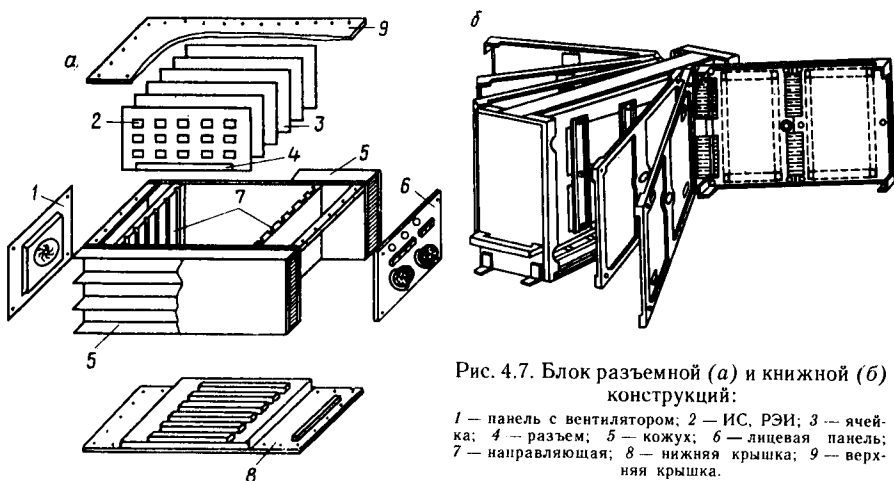


Рис. 4.7. Блок разъемной (а) и книжной (б) конструкций:

1 — панель с вентилятором; 2 — ИС, РЭИ; 3 — ячейка; 4 — разъем; 5 — кожух; 6 — лицевая панель; 7 — направляющая; 8 — нижняя крышка; 9 — верхняя крышка.

законченное устройство, имеет разъем 4, что обеспечивает ремонтпригодность блока на уровне ячейки, удобство доступа ко всем элементам в процессе регулировки.

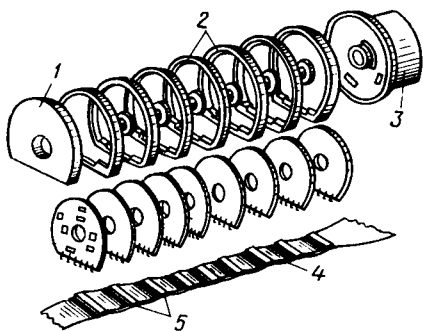
Пример 4.5. Увеличение надежности в совокупности с большей плотностью компоновки, большей механической прочностью, неразрывностью электрических цепей при доступе к элементам достигается в книжном варианте блока, пример которого приведен на рис. 4.7, б.

Пример 4.6. Пример конструкции ИУВС, учитывающей форму носителя (контейнера), приведен на рис. 4.8, где изображен блок цилиндрической формы, разработанный для размещения на ракете.

Рекомендуемые конструктивные решения при создании ИУВС, нормы и требования по конструированию элементов разъемной и

Рис. 4.8. Блок цилиндрической формы:

1 — верхняя крышка; 2 — рамки; 3 — нижняя крышка; 4 — соединительная плата; 5 — ответные части разъемов.



книжной конструкции блоков с применением полупроводниковых и гибридных пленочных микросхем устанавливаются отраслевым стандартом ОСТ 4ГО. 010.009, ред. 3—76.

Конструктивные особенности элементной базы ИУВС. Применение в МП ИУВС в качестве элементной базы комплектов БИС микропроцессорных наборов, БИС ЗУ (ОЗУ, ПЗУ, ППЗУ), специальных ИС средней и большой степени интеграции приводит к использованию ИС с корпусами на 24, 48 выводов и более. Распространенные в настоящее время корпуса типа 1 и 2 с шагом выводов 2,5 мм и типа 4 с шагом выводов 1,25 мм при числе выводов $n \geq 48$ имеют большие размеры, тем самым ограничивают плотность компоновки, увеличивают емкость, индуктивность и сопротивление внутрикорпусных соединений. В связи с этим в настоящее время получают применение *микрокорпуса (керамические кристаллодержатели)* — маленькие безвыводные корпуса из оксида алюминия, по периметру которых располагают выводные площадки с шагом 1 мм (рис. 4.9, а, где крышка микрокорпуса не показана). Соединение микрокорпуса с подложкой из оксида алюминия или любой другой с аналогичным термическим коэффициентом расширения осу-

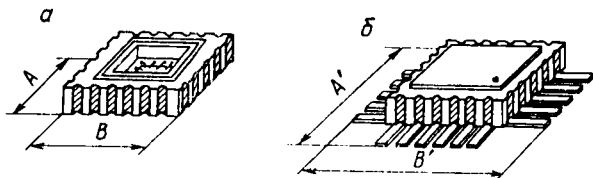


Рис. 4.9. Варианты конструкции микрокорпуса.

ществляется пайкой расплавлением дозированного припоя. Возможна модификация микрокорпуса с металлическими выводами (рис. 4.9, б).

Как видно из табл. 4.2 [25], где приведены конструктивные параметры некоторых микрокорпусов, микрокорпус с числом выводов, например 64, в безвыводном исполнении занимает площадь платы в пять раз меньшую, чем обычный корпус типа 2 с тем же числом выводов (18,2×18,2 мм против 81,3×20,3 мм).

Особенности конструирования ячеек ИУВС. При конструировании ячеек ИУВС применяют печатный тонкопленочный, толстопле-

ночный и объемный монтаж. В табл. 4.3 приведены сравнительные характеристики компоновки ячеек с печатным монтажом и перспективных вариантов тонкопленочного и толстопленочного монтажа. Из таблицы видно, что наибольшую плотность компоновки обеспечивают бескорпусные кристаллы, монтируемые на многослойную полиимидную пленку.

Таблица 4.2

Шифр типо- размера микро- корпуса	Количество выводов микро- корпуса	Размеры, мм, в плане микро- корпуса в исполнении				Размеры монтажной площадки под полупровод- никовый кри- сталл, мм
		без выводов		с выводами		
		А	В	А'	В'	
Н03	16	6,5	6,5	13,5	13,5	2,5×2,3
Н06	24	9,2	7,6	16,2	14,6	5,3×3,7
Н15	42	12,0	12,0	19,0	19,0	6,7×6,7
Н18	48	13,6	13,6	20,6	20,6	7,3×7,3
Н20	64	18,2	18,2	25,2	25,2	9,1×9,1
Н21	84	≤23,76	≤23,76	—	—	—

Таблица 4.3

Корпус ИС	Подложка	Максимальный размер подложки, см	Плотность компоновки, кристалл, см ²
Корпус с двухрядным расположением выводов	Двусторонняя или многослойная печатная плата	60×40	0,3—0,4
Плоский корпус	Многослойная печатная плата	60×40	0,5—0,7
Керамический кристаллодержатель	Многослойная керамика	5×5	1,25
Бескорпусный кристалл	Многослойная полиимидная пленка	5×5	2,4—4
Бескорпусный кристалл на ленте-носителе	Многослойная полиимидная плетка	5×5	2,4—4

При этом возникают большие трудности с пооперационным контролем элементов и герметизацией блока в целом. Высокую плотность компоновки при пооперационном контроле элементов обеспечивают микрокорпуса, монтируемые на многослойную керамическую подложку.

Примечание. Основные рекомендации по проектированию печатных плат изложены в ГОСТ 23752—79 «Платы печатные. Общие технические условия» и ГОСТ 23751—79 «Платы печатные. Требования и методы конструирования».

По плотности компоновки печатные платы разделяются на три класса. Ниже приведены основные параметры для этих классов:

Класс печатной платы	1	2	3
Минимальная ширина проводников и минимальное расстояние между ними, мм	0,5	0,25	0,15
Ширина проводников и расстояние между ними в свободных местах, мм	0,7	0,4	0,25
Минимальная ширина контактной площадки вокруг отверстия, мм	0,05	0,035	0,025
Отношение минимального диаметра металлизированного отверстия к толщине платы	0,4	0,33	0,33

Шаг координатной сетки печатных плат в соответствии с ГОСТ 10317—79 рекомендуется выбирать 2,5 и 1,25 мм. Допускается применять вспомогательную координатную сетку с шагом 1,25 и 0,625 мм соответственно. Линейные размеры печатной платы ячейки зависят от технологических ограничений, функциональной сложности устройства, размещаемого на печатной плате, и ориентировочно могут быть оценены из эмпирического соотношения [26]

$$B_n = \sqrt[3]{\frac{58,6(m+1)(c+t)\sqrt[3]{n^5}}{m \ln(1,72+m)}},$$

где A_n и B_n — ширина и длина печатной платы; $m = A_n/B_n$; $n = 0,7 W$; W — число использованных выводов ИС; c и t — ширина и зазор между проводниками.

Примечание. Рекомендации по конструированию ячеек на корпусных и бескорпусных микросхемах изложены в ОСТ 4ГО.010.009, ряд 3—76.

Особенности теплового конструирования ИУВС. Практический опыт учит, что надежность ИУВС весьма чувствительна к рабочей температуре ее элементов, в частности интенсивность отказов ИС увеличивается вдвое при изменении температуры последних на каждые 18°C . Как отмечалось, основные трудности по обеспечению требуемого теплового режима элементов ИУВС связаны с отводом теплоты. В процессе распространения тепловой энергии участвуют три физических явления: излучение, теплопроводность, конвекция.

Передача теплоты излучением происходит в виде электромагнитных волн главным образом в диапазоне длин волн 0,3—10 мкм. Мощность теплового потока $P_{\text{из}}$, излучаемого телом с поверхности излучения $S_{\text{из}}$, оценивают по закону Стефана — Больцмана:

$$P_{\text{из}} = \varepsilon C_0 S_{\text{из}} [(T_1/100)^4 - (T_2/100)^4], \quad (4.28)$$

где ε — степень черноты поверхности излучения (значения ε для некоторых материалов, применяемых при конструировании, приведены в табл. 4.4); $C_0 = 5,67 \text{ Вт}/(\text{м}^2 \cdot \text{K}^4)$ — коэффициент излучения абсолютно черного тела; $S_{\text{из}}$ — поверхность излучения; T_1 и T_2 — температуры поверхности тела и окружающей среды соответственно.

Передача теплоты теплопроводностью обуславливается, как известно, обменом энергией между молекулами вещества и диффузией электронов. Оба эти явления имеют место при разнице температур в различных точках конструкции. Количество теплоты Q , про-

Т а б л и ц а 4.4

Материал	Температура, °С	Степень черноты поверхности излучения ϵ
Алюминий полированный	50—500	0,04—0,06
Дюралюминий Д-16	50—350	0,37—0,41
Лак черный матовый	40—100	0,96—0,98
Краски эмаливые	20—100	0,92
Краска защитно-зеленая	20	0,9

ходящей через однородное тело в единицу времени, связано законом Фурье с градиентом температур dT/dx так:

$$dQ/dt = -\lambda S_{\text{конд}}(dT/dx), \quad (4.29)$$

где $S_{\text{конд}}$ — поверхность поперечного сечения тела, нормального к направлению теплового потока; λ — теплопроводность материала (значения λ для некоторых материалов, применяемых при конструировании, приведены в табл. 4.5).

Т а б л и ц а 4.5

Материал	Теплопроводность λ , Вт/(м · °С)	Материал	Теплопроводность λ , Вт/(м · °С)
Серебро	416	Керамика 22ХС	8,0
Медь	390	Ситалл	1,5
Золото	297	Паста КПТ-8	0,8
Бериллиевая керамика	220	Кремний	84
Алюминий А0, А5, А7	210	Стекло С48-2, С52-1	1,2
Алюминий АЛ-6	196	Клей ВК-9	0,8
Алюминиевый сплав АМД	188	Слюда	0,58
Олово	67	Текстолит	0,33
Сталь низкоуглеродистая	46	Стеклотекстолит	0,231
Оксид алюминия	27,5	Полиимидная пленка ПМ-1	0,16
Керамика «Поликор»	22	Эпоксидная смола	0,1
Ковар	18	Полихлорвиниловая пленка	0,11
Сталь нержавеющая 12Х18Н10Т	14	Воздух в малых зазорах	0,05
Керамика М-7	9,7	Пенопласт	0,0575

Для стационарных процессов, когда $dQ/dt = P_{\text{конд}} = \text{const}$, из (4.29) получим

$$P_{\text{конд}} = (\lambda S_{\text{конд}}/b)(T_1 - T_2),$$

где b — толщина материала между точками с температурой T_1 и T_2 .

Явление конвекции состоит в переносе теплоты перемещающимися частицами вещества (жидкости или газа). Поток теплоты в единицу времени, отводимый конвекцией $P_{\text{конв}}$ от нагретого тела, пропорционален теплоемкости вещества окружающей среды c , расходу этого вещества W и средней разности температур между нагретым телом T_1 и окружающей средой T_2 :

$$P_{\text{конв}} = cW(T_1 - T_2). \quad (4.30)$$

На практике в конструкциях ИУВС распространены естественное воздушное и принудительное воздушное охлаждения. В связи с ростом удельной мощности, рассеиваемой в блоках ИУВС, начинают применять и такие виды конвективного охлаждения, как естественное испарительное и принудительное жидкостное охлаждения.

При анализе теплопроводящих свойств конструкций блоков ИУВС можно использовать *метод конечных разностей*, который заключается в рассмотрении тепловых соотношений как аналогов электрических соотношений. Поток теплоты P в единицу времени ставится в соответствие ток в электрической цепи, разности температур в двух точках конструкции — разность потенциалов электрической цепи. Вводится понятие теплового сопротивления

$$R_T = \Delta T / P,$$

аналогичное понятию электрического сопротивления. В зависимости от способа передачи теплоты различают:

тепловое сопротивление излучения

$$RT_{\text{изл}} = \frac{T_1 - T_2}{\varepsilon C_0 S_{\text{изл}} [(T_1/100)^4 - (T_2/100)^4]}; \quad (4.31)$$

сопротивление теплопроводности (кондуктивное тепловое сопротивление)

$$R_{T_{\text{конд}}} = b / (\lambda S_{\text{конд}}); \quad (4.32)$$

конвективное тепловое сопротивление

$$R_{T_{\text{конв}}} = 1 / (cW). \quad (4.33)$$

Конструкцию разделяют на ряд частей, предполагая, что каждая часть имеет некоторую среднюю температуру T_i , одинаковую во всех ее точках. С целью анализа распространения тепловых потоков в конструкции составляется расчетная тепловая схема, в которой каждой части конструкции ставится в соответствие узел с температурой T_i . Узлы соединяют соответствующими тепловыми сопротивлениями. Достоинство метода конечных разностей в том, что он дает разработчику математический аппарат анализа зависимости тепловых режимов в точках конструкции от теплового потока и значений тепловых сопротивлений, т. е. от конструктивных параметров (размеры, материал, покрытие) и эксплуатационных параметров (температура, влажность, давление окружающей среды, расход охлаждающего вещества).

Так как конвективное тепловое сопротивление зависит от теплоемкости окружающей среды, а последняя нелинейно зависит от ряда трудно контролируемых параметров (влажность, температура, давление, аэро- или гидродинамические характеристики) среды, то расчет конвективного охлаждения, как правило, ведут с использованием эмпирических данных, полученных для элементов конструкций правильной геометрической формы.

При выборе способа охлаждения учитывают, что диапазон конвективного теплового сопротивления при естественном воздушном охлаждении лежит в пределах 10—0,2 °C/Вт, при принудительном

воздушном охлаждении — $1-0,05\text{ }^{\circ}\text{C/Вт}$, при принудительном жидкостном охлаждении $0,1-0,02\text{ }^{\circ}\text{C/Вт}$.

Теплопроводность конструкции повышается за счет применения: 1) медных или алюминиевых теплопроводящих шин, на которые непосредственно устанавливаются корпуса ИС; 2) металлических (алюминиевых, титановых) рамок в ячейках; 3) металлических плат (сталь, покрытая эмалью, платы из инвара, анодированный алюминий); 4) многослойных керамических плат, изготовленных по толсто пленочной технологии; 5) тепловых разъемов и установочных клиньев для ячеек; 6) теплопроводящих паст, клеев.

Интенсивность теплообмена при естественном воздушном охлаждении увеличивается при: 1) оптимальном расположении элементов; 2) перфорации кожуха; 3) применении тепловых экранов; 4) оребрении отдельных поверхностей; 5) использовании теплопроводящих шин; 6) соответствующей окраске.

При принудительном воздушном охлаждении широко распространены схемы: перемешивание воздуха внутри блока; наружный обдув; продувка.

Продолжение примера 4.4. Пример конструкции блока ИУВС с наружным обдувом приведен на рис. 4.7, а. Здесь теплота от ИС 2 по теплопроводящим шинам передается рамкам ячеек 3; последние установочными клиньями зажимаются в металлических направляющих 7, имеющих низкое тепловое сопротивление до наружного кожуха 5, по которому прогоняется принудительно наружный воздух. Пример варианта охлаждения продувкой поясняет рис. 4.10, где изображена двухплатная пустотелая ячейка. Совокупность таких ячеек образует блок.

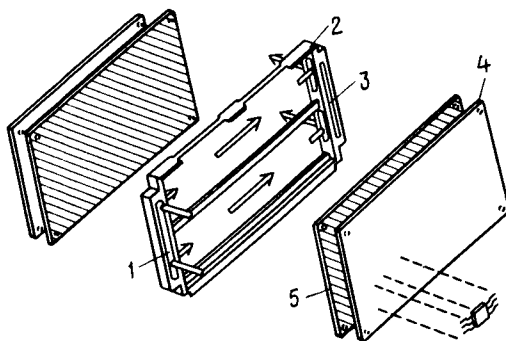


Рис. 4.10. Конструкция ячейки бортового вычислителя с охлаждением продувкой:

1 — вход воздуха; 2 — рамка с фланцем; 3 — выход воздуха; 4 — печатная плата; 5 — плата, отводящая теплоту.

Теплопроводящим элементом конструкции с весьма малым тепловым сопротивлением может служить *тепловая трубка*, работающая на принципе естественного испарительного охлаждения. Типичная тепловая трубка состоит из герметичного металлического цилиндра, внутри которого имеется фитиль или структура, обладающая капиллярным действием. Фитиль насыщается рабочей жидкостью. Если один конец трубки касается нагретой части конструкции, то рабочая жидкость испаряется и перемещается по трубке до тех пор, пока не сконденсируется в более холодной части и не перейдет в жидкое состояние. Затем жидкость возвращается к нагретой части благодаря капиллярному действию фитиля, в результате чего теп-

лота непрерывно передается от одной части трубки к другой с помощью почти изотермического процесса испарения и конденсации.

Методики оценки вибропрочности элементов конструкции ИУВС.

При конструировании МП ИУВС, устанавливаемых на подвижных объектах, необходимо учитывать воздействие на них механических факторов: вибрации, ударов, линейных ускорений, невесомости.

Одним из наиболее разрушительных факторов является вибрация, особенно на частоте, близкой к резонансным частотам узлов и элементов конструкции. В обобщенной конструкции блока ИУВС (рис. 4.11) можно выделить следующие резонансные контуры: резонанса блока в целом, установленного на систему упругих опор (амортизаторов); резонанса ячейки, закрепленной в направляющих блока; резонанса РЭИ, ИС, закрепленных на ячейке. В конструкциях с несколькими степенями свободы наиболее опасные совместные резонансы удается исключить, если по крайней мере удвоить собственную резонансную частоту каждого элемента, вносящего дополнительную степень свободы, т. е. если обеспечить условие (правило «октавы»)

$$f_0^{\text{РЭИ}} \geq 2f_0^{\text{яч}}; f_0^{\text{яч}} \geq 2f_0^{\text{б}}$$

где $f_0^{\text{РЭИ}}, f_0^{\text{яч}}, f_0^{\text{б}}$ — резонансные частоты РЭИ, ячейки, блока соответственно.

Кроме того, собственная резонансная частота ячейки, исходя из условия обеспечения усталостной долговечности, должна удовлетворять условию

$$f_0^{\text{яч}} \geq f_{\min}^{\text{яч}} = \sqrt[3]{Y g j_{\max} / (0,003 b_{\text{яч}})^2}, \quad (4.34)$$

где $j_{\max}$ — вибрационные перегрузки в единицах g , действующие

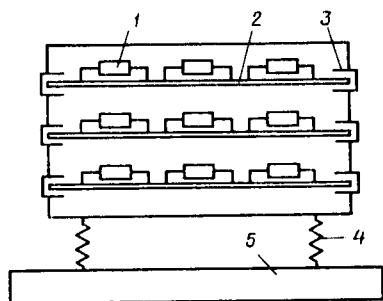


Рис. 4.11. Расчетная схема для анализа вибропрочности блока:

1 — ИС, РЭИ; 2 — ячейка; 3 — корпус блока; 4 — амортизатор; 5 — носитель.

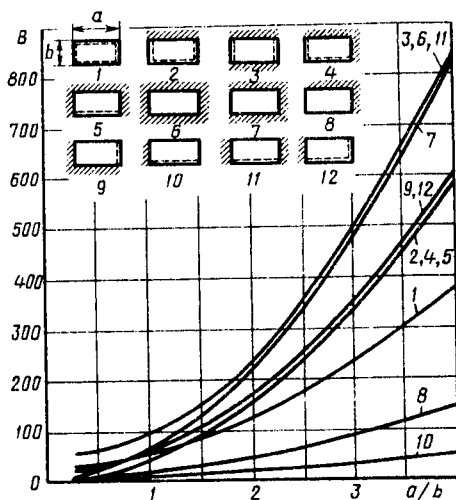


Рис. 4.12. Зависимость коэффициента B от способа закрепления платы.

на ячейку в рабочем диапазоне частот внешних колебаний; $b_{яч}$ — длина короткой стороны ячейки, мм; Y — некоторая эмпирическая постоянная, зависящая от граничной частоты $f_{мин}^{яч}$ и действующих ускорений; при $j_{max} = (3-10)g$

$$Y = \begin{cases} 17,5 & \text{для } f_{мин}^{яч} = 50-100 \text{ Гц;} \\ 25 & \text{для } f_{мин}^{яч} = 100-400 \text{ Гц;} \\ 35 & \text{для } f_{мин}^{яч} = 400-700 \text{ Гц.} \end{cases}$$

Ориентировочно собственная резонансная частота ячейки

$$f_0^{яч} = 10^4 k_{мат} k_{вес} B h_{пл} / a_{пл}^2, \quad (4.35)$$

где $a_{пл}$ и $h_{пл}$ — длина и толщина платы ячейки, см; $k_{мат} = \sqrt{E_{\rho c} / (E_{c\rho})}$ — поправочный коэффициент, учитывающий материал платы (E и ρ — модуль упругости и плотности материала платы, E_c и ρ_c — стали); $k_{вес} = (\sqrt{1 + G_э / G_{пл}})^{-1}$ — поправочный коэффициент, учитывающий влияние распределенной весовой нагрузки ($G_э$ — вес РЭИ, равномерно размещенных по плате, $G_{пл}$ — вес платы); B — коэффициент, зависящий от способа закрепления платы, числовые значения которого можно определить по графику, показанному на рис. 4.12.

Резонансные частоты радиоэлектронных изделий, определенным образом закрепленных на ячейке, можно рассчитать, используя соотношения, приведенные в [27].

Методика выбора системы амортизации. При монтаже блока непосредственно на носитель («жесткий монтаж»), как правило, не удастся обеспечить условие $F_0^6 > f_{max}$, где f_{max} — максимальная частота внешних вибрационных воздействий, и поэтому для существенного ослабления разрушающего влияния внешних вибрационных воздействий блок ИУВС монтируют на носитель использованием амортизаторов или амортизационных оснований.

Анализ колебаний блока, установленного на носитель с использованием амортизаторов («мягкий монтаж»), при кинематическом возбуждении показывает, что виброизолирующие свойства амортизаторов проявляются лишь в тех случаях, когда резонансная частота блока F_0^6 , установленного на систему упругих опор, в $\sqrt{2}$ раз меньше минимальной частоты $f_{мин}$ внешних вибрационных воздействий, т. е.

$$F_0^6 \leq f_{мин} / \sqrt{2}. \quad (4.36)$$

С увеличением расстройки частот f / F_0^6 амплитуда вынужденных колебаний блока A^6 уменьшается по сравнению с амплитудой внешних вибрационных возмущений $A^в$. При отсутствии демпфирования в амортизаторах

$$A^6 / A^в = (|1 - (f / F_0^6)^2|)^{-1}. \quad (4.37)$$

Резонансная частота системы «блок — амортизаторы»

$$F_0^6 = \frac{1}{2\pi} \sqrt{k_{экг} g / P}, \quad (4.38)$$

где $k_{\text{эKB}}$ — эквивалентная жесткость системы упругих опор ($k_{\text{эKB}} = \sum_{i=1}^{i=n} k_i$ при «параллельном включении», $k_{\text{эKB}}^{-1} = \sum_{i=1}^{i=n} k_i^{-1}$ при «последовательном включении» амортизаторов); k_i — жесткость i -го амортизатора; P — масса амортизируемого блока; $g = 9810$ мм/с².

При выборе варианта монтажа блока на носителе с использованием амортизаторов необходимо соблюдать *принцип рационального монтажа*, в соответствии с которым координаты центра тяжести блока должны совпадать с координатами центра жесткости систем амортизации и общая грузоподъемность всех амортизаторов должна быть не менее массы блока, т. е.

$$\begin{aligned} \sum_{i=1}^{i=n} P_i &= G; & \sum_{i=1}^{i=n} P_i x_i y_i &= 0; & \sum_{i=1}^{i=n} P_i x_i &= 0; \\ \sum_{i=1}^{i=n} P_i x_i z_i &= 0; & \sum_{i=1}^{i=n} P_i y_i &= 0; & \sum_{i=1}^{i=n} P_i y_i z_i &= 0; \\ \sum_{i=1}^{i=n} P_i z_i &= 0, \end{aligned} \quad (4.39)$$

где x_i , y_i , z_i , P_i — координаты расположения и весовая нагрузка амортизатора.

При n амортизаторах, исходя из особенностей установки блока на носитель, соображений симметрии и т. п., доопределяют $4n-7$ величин в уравнениях (4.39).

Примечание. Инженерная методика выбора и расчета системы амортизации приведена в [27].

Если основным фактором механических воздействий являются удары, то, жертвуя в разумных пределах виброизоляционными свойствами, следует выбирать либо более жесткие (с большим k_i) амортизаторы, либо использовать амортизаторы с нелинейными характеристиками коэффициента упругости.

При больших линейных перегрузках «мягкий монтаж» блока нецелесообразен, так как ускорения, действующие на него в этом случае, изменяются по закону $\ddot{x}(t) = j_n(1 - \cos 2\pi F_n^0 t)$ и могут вдвое превысить ускорения, действующие на носитель: $\ddot{x}_{\text{max}} = 2j_n$, где j_n — линейные ускорения в единицах g . Поэтому в условиях преобладающих линейных перегрузок применяют «жесткий монтаж».

Рекомендации по обеспечению помехозащищенности аппаратурных средств ИУВС. В процессе создания ИУВС, работающих в условиях высокого уровня помех, разработчик встречается, как правило, с большими трудностями по обеспечению надежного, устойчивого функционирования системы, если им не приняты меры по обеспечению помехозащищенности. Эти трудности возрастают при построении ИУВС с использованием быстродействующих микропроцессорных наборов, например серий К1800, КР1802, К589 и др.

От правильного решения задачи обеспечения помехоустойчиво-

сти элементов и устройств аппаратурных средств ИУВС зависят как сроки разработки, изготовления и наладки последней, так и нормальное ее функционирование в процессе эксплуатации. Наиболее успешная борьба с помехами возможна лишь в том случае, когда разработки электрических схем и конструкции элементов и узлов неразрывно связаны.

Для вычислительного устройства *помеха* — внешнее или внутреннее воздействие, приводящее к искажению дискретной информации во время ее хранения, преобразования и передачи.

По источнику создания помехи целесообразно разделять на *помехи внешние*, как правило, создаваемые внешними по отношению к рассматриваемому устройству аппаратами, устройствами, условиями эксплуатации (двигатели, электромеханические механизмы, передатчики электромагнитной энергии, удары молнии и т. п.), и *помехи внутренние*, обусловливаемые конкретным выполнением линий связи в рассматриваемом устройстве.

По месту проявления помехи могут быть подразделены на *помехи в сигнальных линиях связи* и *помехи в цепях питания*. Помехи в сигнальных линиях связи в свою очередь подразделяют на *помехи в электрически коротких* и *помехи в электрически длинных линиях связи*.

Линия связи считается *электрически короткой линией*, если $\min\{t^{1,0}, t^{0,1}\} \geq 2l/v_p$, где $t^{1,0}$ и $t^{0,1}$ — соответственно время спада и нарастания передаваемого сигнала; l — длина линии связи; v_p — скорость распространения сигнала в линии связи. На практике принимают $\min\{t^{1,0}, t^{0,1}\} \geq (8-10)l\sqrt{\epsilon}/c_0$, где ϵ — диэлектрическая постоянная среды; $c_0 = 300\,000$ км/с. Свойства электрически коротких линий связи можно описать электрическими цепями с сосредоточенными параметрами.

Линия связи считается *электрически длинной линией*, если $\max\{t^{1,0}, t^{0,1}\} \leq 2l\sqrt{\epsilon}/c_0$. Электрически длинные линии связи в расчетах представляют распределенными структурами; для их анализа используют математический аппарат длинных линий. Доля электрически длинных линий связи в конструкциях ИУВС с ростом быстродействия увеличивается. Из табл. 4.6 видно, что, например, линии связи между элементами ТТЛ-типа нужно рассматривать как электрически длинные уже при физической длине более 0,30 м.

По характеру воздействия на дискретную инфор-

Т а б л и ц а 4.6

Тип логических схем	Логический перепад ΔU_d , В	Время спада или нарастания сигнала $t^{1,0}$, $t^{0,1}$, нс	Скорость изменения напряжения в сигнальной линии, В/нс	Скорость изменения тока в цепи питания, А/нс	Длина электрически короткой линии связи, м
КМДП	15	65	0,23	0,037	3,9
Маломощные ТТЛ/Ш	3,5	5	0,7	0,8	0,3
ЭСЛ	0,9	2	0,45	0	0,12

мацию помехи в устройствах ИУВС, выполняемых на ИС, могут быть разделены на виды. Видами проявления *внутренних помех в электрически коротких линиях связи* являются задержки сигналов из-за емкостного и индуктивного характера линии связи, емкостные и индуктивные взаимные помехи между сигнальными проводниками, а *внутренних помех в электрически длинных линиях связи* — задержки передачи сигналов, искажения формы передаваемых сигналов из-за отражений, взаимные помехи между линиями связи, затухания сигналов.

К проявлениям помех в цепях питания относят: постоянные смещения уровня шины «Земля», обусловливаемые активным сопротивлением шины «Земля»; импульсные ЭДС, вызываемые динамическими токами потребления ИС, протекающими по индуктивности шин «Земля» и «Питание»; периодические колебания напряжения питания, вызываемые реактивным характером цепей питания.

Помехи в электрически коротких линиях связи. Собственные емкость или индуктивность линии связи могут вносить дополнительные задержки в распространение сигналов, если не учтены особенности логических элементов и не приняты соответствующие меры при конструировании.

Задержка сигнала, вносимая линией связи:

для линии связи с большой собственной емкостью относительно общей шины «Земля» и в предположении, что собственная индуктивность линии и взаимные индуктивности и емкость связи с другими сигнальными линиями пренебрежительно малы,

$$t_{эд} = C_{эвб} \frac{R_{вх.эвб} R_{вых}}{R_{вх.эвб} + R_{вых}} \ln \frac{1}{1 - U_{пор} (R_{вх.эвб} + R_{вых}) / \Delta U_{л} R_{вх.эвб}}; \quad (4.40)$$

для линии связи с большой собственной индуктивностью и в предположении, что собственная емкость линии и взаимные индуктивности и емкость связи с другими сигнальными линиями пренебрежительно малы,

$$t_{эд} = \frac{L_{л}}{R_{вх.эвб} + R_{вых}} \ln \times \frac{1}{1 - U_{пор} (R_{вх.эвб} + R_{вых}) / \Delta U_{л} R_{вх.эвб}}. \quad (4.41)$$

Здесь $C_{эвб} = C_{вых} + C'_л + k_n C_{вх}$; $R_{вх.эвб} = R_{вх} / k_n$; $R_{вх}$, $C_{вх}$ и k_n — входные сопротивление, емкость и число нагрузок (логических элементов), подключенных к выходу линии связи; $R_{вых}$, $C_{вых}$ — выходные сопротивление и емкость управляющего логического элемента, подключенного к входу линии; $\Delta U_{л}$ и $U_{пор}$ — логический перепад и пороговое напряжение логического элемента; $C'_л$ и l — удельная собственная емкость и длина линии связи; $L_{л} = lL'_л$ и $L'_л$ — удельная собственная индуктивность линии связи (формулы для расчета удельных собственных емкостей $C'_л$ и индуктивностей $L'_л$ при различных вариантах выполнения линии связи приведены, например, в [9]).

Для ориентировочного определения параметров объемного монтажа можно воспользоваться следующими значениями удельных соб-

ственных емкостей: одиночный проводник, расположенный вплотную к печатной плате, — $(30—50) \cdot 10^{-12}$ Ф/м; проводник в объемном жгуте с плотностью 10—40 проводников на 1 см^2 поперечного сечения — $(40—70) \cdot 10^{-12}$ Ф/м; экранированный провод — $(100—200) \cdot 10^{-12}$ Ф/м; витая пара проводников — $(35—50) \cdot 10^{-12}$ Ф/м; витая пара проводников в экране — $(50—100) \cdot 10^{-12}$ Ф/м; плоский жгут из проводов МГШВ — $(0,2—33) \cdot 10^{-12}$ Ф/м.

Для ориентировочного определения параметров печатного монтажа в табл. 4.7 приводятся значения удельных собственных и взаимных емкостей печатных проводников для печатных плат: а) с сигнальными проводниками с двух сторон, удаленных от плоскости «Земля» на расстояние не менее 0,02 м (вариант I); б) с сигнальными проводниками с одной стороны и плоскостью «Земля» с другой стороны (вариант II). В обоих вариантах печатная плата толщиной 0,001 м, диэлектрическая постоянная материала платы $\epsilon = 6—7$.

Т а б л и ц а 4.7

Вариант	Погодная емкость печатного проводника, Ф/м	
	собственная $C_0 \cdot 10^{-12}$	взаимная $C_{св} \cdot 10^{-12}$
I	10—15	40—55* 25—35**
II	45—60* 60—90**	35—20* 10—5**

* При ширине печатных проводников и расстоянии между ними $(0,2—0,4) \cdot 10^{-3}$ м.

** При ширине печатных проводников $(0,2—0,8) \times 10^{-3}$ м и расстоянии между ними $(0,8—1,0) \cdot 10^{-3}$ м.

Анализ (4.40) и (4.41) показывает, что с целью уменьшения задержек, вносимых линиями связи, необходимо использовать логические элементы с малым выходным $R_{вых}$ и большим входным $R_{вх}$ сопротивлениями. Элементы КМПД-типа имеют относительно большое выходное сопротивление $R_{вых} = 0,5—2$ кОм, поэтому при конструировании ИУВС на ИС КМПД-типа необходимо применять варианты линий связи с малым значением удельной собственной емкости, например объемные и печатные проводники, удаленные от заземленной плоскости. Поскольку логические элементы ТТЛ-типа имеют относительно малое входное сопротивление $R_{вх} = 4—40$ кОм, для соединения последних необходимо применять варианты линий связи с малыми собственными индуктивностями, например монтаж объемными и печатными проводниками над заземленной плоскостью, симметричные и несимметричные микрополосковые линии связи в многослойных печатных платах.

Взаимная индуктивная и емкостная связь — причина возникновения взаимных помех в «пассивных» проводниках при передаче по «активным» проводникам импульсных сигналов. В течение време-

ни спада и нарастания сигнала в «активной» линии и «пассивной» линии возникают помехи, проявляющиеся в изменении уровней напряжения передаваемой информации. Взаимные помехи проявляются в виде:

ЭДС взаимной индукции

$$U_{\text{пом}} = \left| -M_{\text{св}} \frac{di_A}{dt} \right| \approx M_{\text{св}} \frac{\Delta U_{\text{л}}}{t_{\text{нар}} (R_{\text{вых}} + R_{\text{вх}})},$$

заряда помехи

$$Q_{\text{пом}} = C'_{\text{св}} \int_0^{t_{\text{нар}}} dU_{\text{а}}(t) = C'_{\text{св}} l \Delta U_{\text{л}}.$$

Здесь di_A/dt — скорость изменения тока в «активной» линии связи; $M_{\text{св}}$ — взаимная индуктивность «активной» и «пассивной» линий; $t_{\text{нар}}$ — длительность фронта сигнала, распространяющегося в «активной» линии; $C'_{\text{св}}$ — удельная взаимная емкость между «активным» и «пассивным» проводниками.

Поскольку $U_{\text{пом}}$ и $Q_{\text{пом}}$ зависят от $M_{\text{св}}$ и $C_{\text{св}}$, а последние есть функции конструктивного варианта линии связи и ее длины, нетрудно оценить допустимую длину $l_{\text{доп}}$ близко расположенных сигнальных линий, при которой не происходит ложного срабатывания логических элементов, подключенных к «пассивной» линии, в момент перепада импульсного сигнала по «активной» линии.

Для случаев емкостной связи

$$l_{\text{доп}} \leq \begin{cases} \frac{Q_{\text{пор}} t_{\text{нар}}}{2C'_{\text{св}} \Delta U_{\text{л}} t_{\text{зд.р.ср}}}, & \text{если } t_{\text{нар}} \geq 2t_{\text{зд.р.ср}}; \\ \frac{Q_{\text{пор}}}{C'_{\text{св}} \Delta U_{\text{л}}}, & \text{если } t_{\text{нар}} < 2t_{\text{зд.р.ср}}, \end{cases}$$

где $Q_{\text{пор}}$ — пороговый заряд «пассивной» линии; $t_{\text{зд.р.ср}}$ — среднее время задержки распространения сигнала в логическом элементе.

Для случая индуктивной связи

$$M_{\text{св}}(l_{\text{доп}}) \leq \begin{cases} \frac{2U_{\text{пор}}(R_{\text{вх}} + R_{\text{вых}})^2 t_{\text{зд.р.ср}}}{R_{\text{вх}} \Delta U_{\text{л}}}, & \text{если } t_{\text{нар}} \leq 2t_{\text{зд.р.ср}}; \\ \frac{U_{\text{пор}}(R_{\text{вх}} + R_{\text{вых}})^2 t_{\text{нар}}}{R_{\text{вх}} \Delta U_{\text{л}}}, & \text{если } t_{\text{нар}} > 2t_{\text{зд.р.ср}}. \end{cases}$$

В руководящих технических материалах по применению ИС, как правило, приводятся значения допустимых длин линий связи, при которых взаимные помехи не существенны, например в табл. 4.8 приводятся значения допустимой длины печатных параллельных проводников в устройствах на ИС ТТЛ-типа при ширине последних $(0,5—1,5) \cdot 10^{-3}$ м.

Учитывая особенности элементной базы (см. табл. 4.6), нетрудно понять, что для линий связи между логическими элементами КМДП-типа доминирующей оказывается емкостная взаимная связь (токи

Таблица 4.8

Число парал- лельных проводников	Допустимая длина, м, печатных параллельных проводников при интервале между ними, м				
	0,0005	0,001	0,0015	0,003	0,005
2	0,1	0,120	0,130	0,150	0,170
3	0,06	0,07	0,075	0,090	0,100
4	0,05	0,06	0,065	0,070	0,80
5	0,04	0,05	0,060	0,065	0,070

в сигнальных линиях малы), для линий связи элементов ТТЛ-типа доминирует по влиянию индуктивная помеха (зарядовая составляющая помехи оказывает малое влияние из-за относительно больших значений собственной емкости «пассивной» линии). Поэтому с точки зрения уменьшения взаимных емкостных помех в линиях связи элементов КМДП-типа монтаж рекомендуется выполнять с использованием двусторонних печатных плат с перпендикулярным расположением проводников на их сторонах.

Для уменьшения взаимных индуктивных помех в линиях связи элементов ТТЛ-типа объемный и печатный монтаж рекомендуется выполнять над заземленной плоскостью или применяя многослойные печатные платы, в которых сигнальные слои чередуются со слоями «Питание» и «Земля».

Помехи в электрически длинных линиях связи. Если между логическими элементами линия связи электрически длинная, то ее эквивалентная схема представляется распределенной однородной структурой (рис. 4.13) и связь между токами и напряжениями в лю-

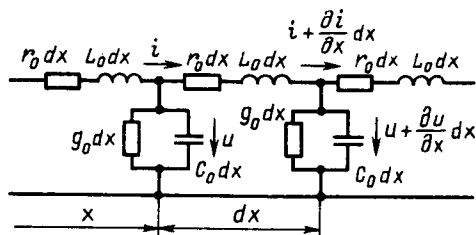


Рис. 4.13. Эквивалентная схема электрически длинной линии связи.

бой момент времени t в точке x описывается следующей системой дифференциальных («телеграфных») уравнений:

$$-\frac{\partial u}{\partial x} = ir_0 + L_0 \frac{\partial i}{\partial t}; \quad -\frac{\partial i}{\partial x} = g_0 u + C_0 \frac{\partial u}{\partial t}, \quad (4.42)$$

где C_0 , L_0 , r_0 , g_0 — соответственно удельные емкость, индуктивность, сопротивление и проводимость длинной линии связи.

При неоднородностях в линиях связи или несогласованности сопротивлений нагрузки на концах электрически длинной линии с волновым сопротивлением линии в последней наряду с задержкой пере-

дачи сигнала $t_{зд} = l\sqrt{L_0 C_0}$ происходит искажение передаваемой информации. Характер возникающих искажений нетрудно предсказать, если процессы, происходящие в электрически длинной линии, представлять как совокупность падающей и отраженной волн. В линии без потерь ($r_0 \rightarrow 0, g_0 \rightarrow 0$) решение системы (4.42) дает связь между напряжениями и токами в любой точке линии в любой момент времени t с напряжениями и токами падающей и отраженной волн в линии:

$$u = f_1(x - vt) + f_2(x + vt) = U_{\text{пад}} + U_{\text{отр}};$$

$$i = z_0^{-1} [f_1(x - vt) - f_2(x + vt)] = i_{\text{пад}} - i_{\text{отр}},$$

где $v = 1/\sqrt{L_0 C_0}$ — скорость распространения прямой и обратной волн; $z_0 = \sqrt{L_0/C_0}$ — характеристическое (волновое) сопротивление электрической длинной линии связи.

При этом $U_{\text{отр}}(x, t)$ вычисляется через $U_{\text{пад}}(x, t)$ и коэффициент отражения

$$K_{\text{отр}}(s) = \frac{U_{\text{отр}}(s)}{U_{\text{пад}}(s)} = \frac{z_n(s) - z_0}{z_n(s) + z_0},$$

где $z_n(s)$ — сопротивление нагрузки.

Если линия без потерь рассогласована на концах, то в ней теоретически бесконечно долго продолжают колебания напряжения, приводящие к ложным срабатываниям нагрузочных логических элементов.

Расчет переходных процессов при передаче сигналов по электрически длинной линии выполняют, используя понятие коэффициента отражения в случае линейных нагрузок и графический метод Берже-рона в случае нелинейных нагрузок на ее концах [9].

На практике электрически длинные линии согласуют, обеспечивая условие $R = z_0$ либо на одном, либо на обоих концах линии. В частности, магистральные линии, в любой промежуточной точке которых могут быть подключены и передатчик и приемник, согласуют на обоих концах, включая делители напряжений из сопротивлений $R_1 - R_2$ между шиной питания и «общей» шиной, такие, что

$$R_1 R_2 / (R_1 + R_2) = z_0; [U_{\text{ин}} R_2 / (R_1 + R_2)] \geq U_{\text{мин}}^{\text{л}}.$$

В таком случае отражения, возникающие в любой промежуточной точке линии, достигнув ее конца, прекращаются.

При большой физической длине линии необходимо учитывать затухание, вносимое ею. Для кабельных линий связи затухание составляет 20—30 дБ/км.

Наряду с искажениями сигналов, вызываемыми отражениями, в электрически длинных линиях связи, как и в электрически коротких линиях, возникают взаимные помехи, амплитуда которых даже в согласованных микрополосковых линиях связи может составлять до 30 % от амплитуды логического перепада при длительности импульса помехи $2l\sqrt{L_0 C_0}$ [9].

На практике линии связи элементов ЭСЛ-типа уже в пределах ячеек (печатных плат) необходимо рассматривать как электрически

длинные, применяя согласованные симметричные и несимметричные микрополосковые линии. Электрически длинными также являются межъячеечные и межблочные линии связи элементов ТТЛ-типа, и их выполняют с использованием радиочастотных кабелей, плоских кабелей, гибких печатных плат, бифиляров.

Перспективным направлением в создании высокоскоростных и высокопомехоустойчивых линий связи является применение волоконно-оптических линий связи (ВОЛС) (рис. 4.14.). Оптический сигнал

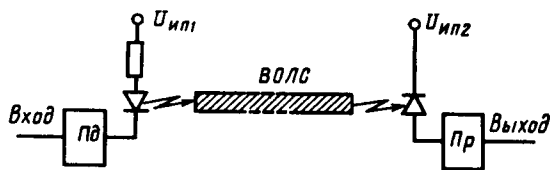


Рис. 4.14. Схема волоконно-оптической линии связи.

в таких линиях распространяется по волоконно-оптическому кабелю (ВОК) с полосой пропускания до 3 ГГц при весьма малом затухании 5 дБ/км. В качестве передатчика Пд может быть использован лазер или светодиод, в качестве приемника Пр рpn-диод или лавинный фотодиод.

Применение ВОЛС наряду с максимальными скоростями передачи информации при умеренном затухании уменьшает габаритно-весовые показатели, как правило, на порядок и обеспечивает устойчивость к внешним электромагнитным помехам. Эти линии связи широко применяют при построении линий связи, обеспечивающих последовательную передачу информации между территориально удаленными средствами ИУВС.

Помехи в цепях питания. Одним из важнейших моментов по обеспечению помехозащищенности проектируемой конструкции ИУВС является тщательная проработка цепей питания, в первую очередь в устройствах на элементах ТТЛ- и ЭСЛ-типа. Помехи в цепях питания возможны из-за падения напряжения на активном сопротивлении шин «Земля» и «Питание» при прохождении по ним постоянных токов; возникновения ЭДС самоиндукции в цепи шин «Питание» при прохождении по ним импульсных токов; «медленных» колебательных процессов в шинах «Питание» при «бросках» тока нагрузки.

Чтобы свести к минимуму *постоянную помеху в цепях питания*, необходимо выбрать такую конструкцию шин «Земля», при которой падение напряжения на ней от постоянного тока было бы меньше наперед заданного допустимого значения $U_{\text{пом. доп}}$, рассчитываемого из условия обеспечения помехозащищенности конструкции.

Конструктивными мерами по уменьшению постоянных помех являются: увеличение сечения шины «Земля»; увеличение числа заземляющих точек, что уменьшает длину общих участков протекания токов питания элементов; применение заземленных медных листов, к которым припаиваются все обратные провода ячеек или модулей;

использование для подвода питания отдельных слоев многослойной печатной платы.

Импульсные помехи в цепях питания обуславливаются наличием паразитной индуктивности шин «Земля» и бросками токов потребления интегральных микросхем при переключении последних из одного состояния в другое. Прохождение переменных составляющих токов потребления микросхем по шинам «Земля» и «Питание» приводит к возникновению ЭДС самоиндукции, выступающей как паразитная, импульсная помеха, приложенная к входу микросхемы, которая в данный момент времени находится в установившемся состоянии. Уменьшение импульсной помехи по цепям питания достигается либо выбором элементов с малыми бросками токов при переключении (сравните, например, элементы ТТЛ-типа с элементами ЭСЛ-типа, табл. 4.6), либо при заданной системе элементов путем уменьшения индуктивности общей шины «Питание», что в свою очередь может быть достигнуто увеличением числа заземляющих точек; применением заземленных медных листов; использованием для подвода питания слоев многослойной печатной платы; навесных шин «Питание» (рис. 4.15), обеспечивающих малую индуктивность, большую распределен-

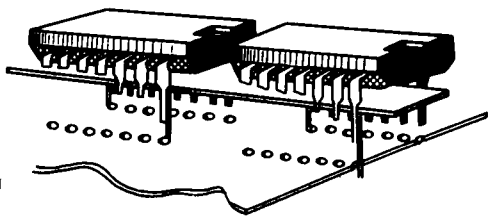


Рис. 4.15. Пример конструкции шин «Питание».

ную емкость, малое (единицы Ом) волновое сопротивление цепей питания и при необходимости выполняющих роль тепловых мостов, по которым теплота от микросхем передается к рамкам ячеек; индивидуальных конденсаторов развязки; групповых конденсаторов развязки.

Конструктивно индивидуальный конденсатор развязки устанавливается в непосредственной близости от микросхемы, выводы конденсатора возможно меньшей длины подпаиваются к контактным площадкам выводов питания микросхемы. В качестве индивидуальных конденсаторов развязки необходимо выбирать конденсаторы с малой собственной паразитной индуктивностью. Индивидуальные конденсаторы развязки позволяют локализовать протекание динамических токов потребления в рамках цепи «микросхема — индивидуальный конденсатор развязки» (паразитная индуктивность такой цепи пренебрежимо мала).

Выбор емкости индивидуального конденсатора развязки $C_{\text{ин}}$ производится из условия равенства заряда, накапливаемого конденсатором за время переключения логического элемента, заряду, переносимому динамической составляющей тока потребления $\Delta I_{\text{ин}}$ за время переключения элемента $t_{\text{нар}}$. При этом изменение напряжения на

конденсаторе не должно превышать наперед заданного значения, равного допустимой помехе $\Delta U_{\text{пом. доп}}$ по шине «Питание»:

$$\Delta U_{\text{пом. доп}} C_{\text{ир}} \geq \int_0^{t_{\text{нар}}} i_{\text{ип}}(t) dt \approx \Delta I_{\text{ип}} t_{\text{нар}} / 2,$$

откуда

$$C_{\text{ир}} \geq \Delta I_{\text{ип}} t_{\text{нар}} / (2 \Delta U_{\text{пом. доп}}).$$

Примечание. На практике часто принимают $\Delta I_{\text{ип}} = b I_{\text{кз}}$, где b — коэффициент, характеризующий серию ИС (для схем ТТЛ-типа коэффициент принимают равным 1/3); $I_{\text{кз}}$ — выходной ток короткого замыкания микросхемы.

Для серий К133, К155 интегральных микросхем значение емкости конденсатора индивидуальной развязки обычно выбирается из расчета $(7-10) \cdot 10^3$ пФ на вентиль; для серий К164, К564, К765 — $(1,5-2) \cdot 10^3$ пФ на вентиль.

Из-за бросков тока в системе питания могут возникать «медленные» колебания напряжения. При правильно спроектированной цепи питания (включение больших электролитических конденсаторов) эти колебания носят затухающий характер. Емкость группового конденсатора развязки C_p , удовлетворяющего этому условию, рассчитывается из соотношения

$$C_p \geq 4L_{\text{ш}} / R_{\text{ш}}^2,$$

где $L_{\text{ш}}$ и $R_{\text{ш}}$ — индуктивность и сопротивление шины «Питание» соответственно.

Принципы экранирования. Для защиты от внешних помех следует применять электростатическое, магнитостатическое и электромагнитное экранирование.

От *внешних электростатических полей* (преобладает емкостная связь) защитой может служить металлический экран (провод в металлической оплетке, коаксиальный кабель, корпус и крышка блока), заземленный хотя бы в одной любой точке.

От *постоянных магнитных полей* надежной защитой служит экран из магнитомягкого материала.

При *недопустимо больших электромагнитных внешних помехах*

(преобладает индуктивная связь) сигнальные проводники следует выполнять экранированными (коаксиальный кабель, свитая пара в экране,

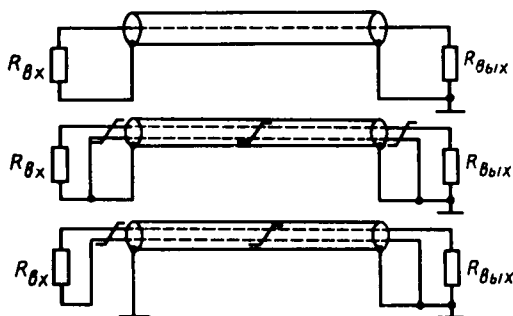


Рис. 4.16. Примеры эффективных способов экранирования проводников от внешних электромагнитных наводок.

экранированный отдельный проводник), причем экран нужно обязательно заземлять в крайних его точках, добиваясь уменьшения контура потокоцепления сигнальных проводников с внешним магнитным полем. Следует помнить, что защитное влияние экрана ухудшается по мере уменьшения частот внешних помех начиная с частот среза экранов $f_{ср.з} = R_s / (2\pi L_s)$, где R_s и L_s — сопротивление и индуктивность экрана; $f_{ср.з}$ не превышает диапазон звуковых частот, т. е. $f_{ср.з} \leq 10-15$ кГц.

При *низкочастотных внешних помехах* рекомендуется использовать экранированную «свитую» пару или триаксиальный кабель. На рис. 4.16 приведены наиболее эффективные способы экранирования от электромагнитных внешних помех.

4.3. Организация контроля микропроцессорных информационно-управляющих вычислительных систем

Под задачей контроля ИУВС будем понимать установление факта отказа сменного блока, входящего в систему. Отказ j -го сменного блока в простейшем случае считается установленным при условии

$$n_j / \bar{n}_j > k_j,$$

где n_j — количество зарегистрированных фактов неисправного состояния j -го блока в единицу времени; $\bar{n}_j$ — количество зарегистрированных фактов исправного состояния j -го блока в единицу времени; k_j — порог, устанавливаемый в зависимости от характера использования j -го блока и системы в целом.

Например, для удовлетворения требования устойчивости системы автоматического управления необходимо вырабатывать управляющие воздействия с частотой посылок не менее f_j . При выборе частоты управляющих воздействий $f_j^* \geq f_j$ допустимо формировать управляющие воздействия в виде потока перемежающихся правильных и неправильных сигналов управления. Соотношение таких сигналов управления в потоке определяется запасом устойчивости и требуемыми точностными характеристиками.

Установление фактов отказов сменных блоков иллюстрируется примером (рис. 4.17). Информационно-управляющая вычислительная система состоит из сменных блоков СБ₁, ..., СБ _{$j-1$} , СБ _{j} , ..., СБ _{n} . Каждый сменный блок СБ _{j} формирует совокупность управляющих воздействий $У_j$ на объект управления ОУ и получает совокупность сигналов обратных связей ОС _{j} .

В процессе работы ИУВС по сигналам опроса $С_j$, поступающим со схемы формирования сигналов опроса СФСО, схемы контроля СК _{j} формируют воздействия $В_j$ и получают информацию $И_j$ о состоянии блоков СБ _{j} . Для установления факта отказа j -го сменного блока и выработки сигнала $О_j$ подсчитывают количество сигналов неисправности n_j (исправности $\bar{n}_j$) на реверсивном счетчике СЧ _{j} . Причем количество сигналов исправности дополнительно делится на счетчике К _{j} . На выходе схемы сравнения СС _{j} формируется сигнал отказа блока

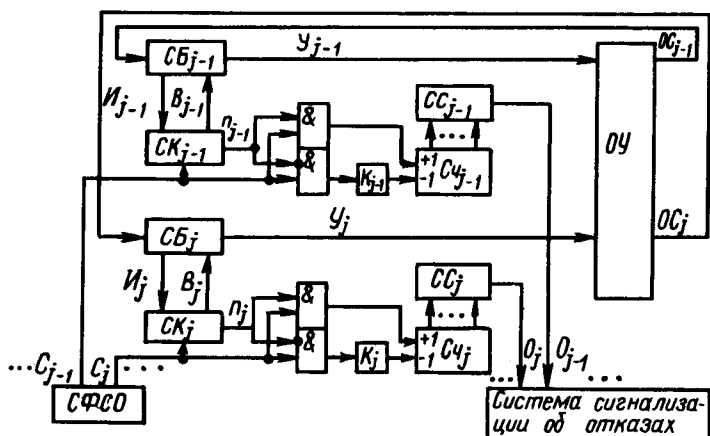


Рис. 4.17. Иллюстрация отказов сменных блоков.

СБ_г, если содержимое $\langle Cч_j \rangle$ счетчика $Cч_j$ удовлетворяет условию $\langle Cч_j \rangle = n_j - n_j K_j > 0$. Сигналы O_j отказов блоков выводятся на систему сигнализации об отказах. Реальные ИУВС значительно сложнее, однако в ряде случаев они могут быть сведены к приведенному примеру. Используемые в примере средства, связанные с формированием сигналов отказов (СК_г, СФСО и т. д.), характеризуют логику работы, но не конкретную реализацию.

В зависимости от характера использования ИУВС допускают два подхода в организации систем контроля: в виде систем встроенного контроля (СВК), конструктивно объединенных с блоками ИУВС; в виде контрольно-проверочной аппаратуры (КПА), которая конструктивно не объединяется с конкретной системой, а предназначена для обслуживания группы ИУВС. Однако в том и другом случае формируется и перерабатывается дополнительная, избыточная по отношению к основной информация. В свою очередь в зависимости от способа обработки дополнительной информации различают аппаратный, программный и программно-логический методы контроля.

Аппаратный метод контроля предусматривает введение в структуру ИУВС дополнительного оборудования, предназначенного для обнаружения ошибок при переработке и передаче информации. Принцип действия дополнительного оборудования основан на использовании следующих основных видов контроля: по модулю; с использованием специальных кодов и специальных схемных решений.

Использование контроля по модулю m позволяет обнаруживать все комбинации ошибок, за исключением ошибок, удовлетворяющих условию:

$$\left[\sum_{i=0}^n l_i 2^i \right] \bmod m = 0; \quad l_i = (\alpha_i^* - \alpha_i) \bmod 2,$$

где l_i — значение i -го разряда кода ошибки; α_i^* — значение i -го разряда контролируемого кода; α_i — значение, которое должно находиться в i -м разряде при отсутствии ошибок.

Из-за простоты аппаратной реализации наиболее распространен код с проверкой четности, образуемый добавлением к группе информационных разрядов одного контрольного разряда, равного сумме всех информационных разрядов по модулю 2. Таким образом, если код содержит n информационных разрядов, а $(n+1)$ -й разряд контрольный, то значение контрольного разряда

$$A_{n+1} \equiv \sum_{i=1}^{n+1} A_i \pmod{2},$$

где A_i — значение i -го информационного разряда числа (для $i=1, 2, \dots, n$).

Сигналы четности S_i и нечетности $\bar{S}_i$ по разрядам кода могут быть получены следующим образом:

$$\begin{aligned} S_i &= S_{i-1} \cdot \bar{A}_i + \bar{S}_{i-1} \cdot A_i = \overline{S_{i-1} \oplus A_i}; \\ \bar{S}_i &= S_{i-1} \cdot A_i + \bar{S}_{i-1} \cdot \bar{A}_i = \overline{S_{i-1} \oplus \bar{A}_i}, \end{aligned}$$

где $S_{i-1} = \sum_{j=1}^{i-1} A_j \pmod{2}$; A_i ; $\bar{A}_i$ — прямое и инверсное значения j -го разряда числа.

Схема контроля четности может быть построена в виде цепочки (рис. 4.18, а) или как многоярусная пирамида (рис. 4.18, б). Код

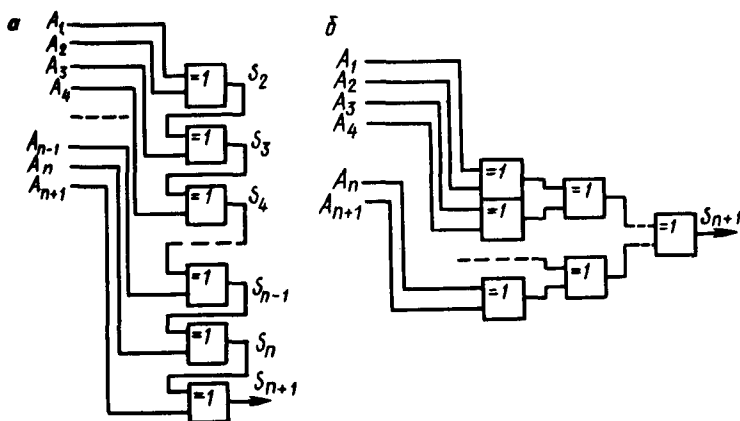


Рис. 4.18. Схемы контроля четности в виде цепочки (а) многоярусной пирамиды (б).

с проверкой четности обнаруживает все одиночные и все случаи нечетного количества ошибок. Контроль по модулю широко используется и при выполнении арифметических и логических операций, что основывается на следующих свойствах:

$$a_1 + a_2 + \dots + a_k = (b_1 + b_2 + \dots + b_k) \bmod m;$$

$$a_1 \cdot a_2 \cdot \dots \cdot a_k = (b_1 \cdot b_2 \cdot \dots \cdot b_k) \bmod m;$$

$$a_1 - a_2 = (b_1 - b_2) \bmod m,$$

где a_i и b_i — числа, равные по модулю m , т. е. $a_i = b_i \bmod m$.

В частности, структура сумматора, контролируемого по модулю, имеет вид, показанный на рис. 4.19. На сумматоре складываются

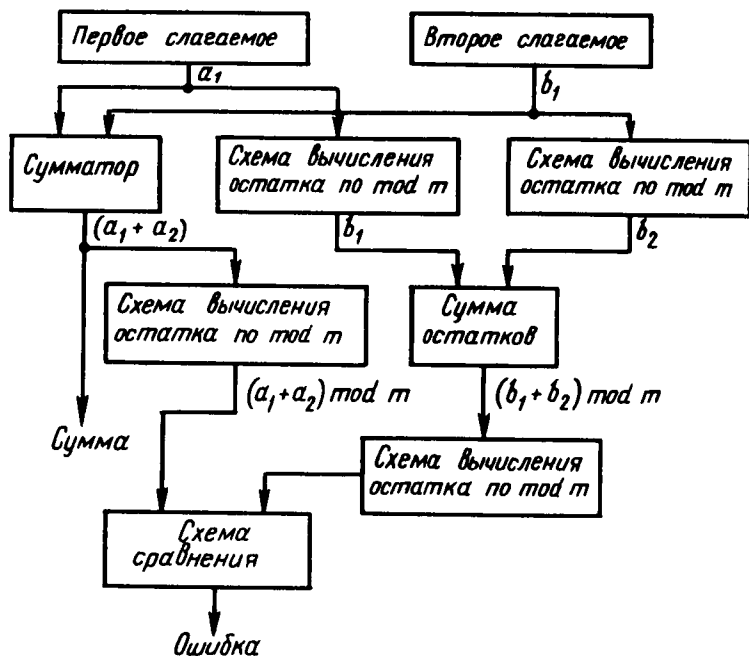


Рис. 4.19. Сумматор, контролируемый по модулю.

операнды a_1 и a_2 ; при этом образуются сумма и наименьший положительный остаток по модулю m , т. е. $(a_1 + a_2) \bmod m$. Одновременно вычисляются остатки по модулю b_1 и b_2 и формируется остаток суммы $(b_1 + b_2)$ по модулю m . Сигнал ошибки формируется в случае, если $(a_1 + a_2) \bmod m \neq (b_1 + b_2) \bmod m$. Наиболее часто в ИУВС используется контроль по модулю 3, 7, 9, 11, 15. Аналогично строятся схемы контроля выполнения вычитания, умножения, а также логических операций.

Задача обнаружения ошибок в ИУВС при выполнении арифметических действий, операций управления, а также при передаче информации часто удачно может быть решена применением алгебраических кодов. Использование свойств алгебраических кодов позволило создать в достаточной степени универсальные аппаратные средства для обнаружения ошибок в устройствах с цифровым представлением сигналов в контрольных точках. При этом для оценки

правильности получаемой в контрольной точке или на приемном конце линии информации, исходное сообщение должно быть предварительно закодировано. Существуют два основных способа такого кодирования: перемножения и деления.

В первом способе исходное сообщение, например 101 1001, представленное полиномом $M(X)=X^6+X^4+X^3+1$, пересылается как закодированное информационное сообщение $V(X)$ в виде

$$V(X)=G(X) \cdot M(X),$$

где $G(X)$ — образующий полином.

Над полиномами можно выполнять различные действия по законам обычной алгебры, однако сложение выполняется по модулю 2. Таким образом, если $G(X)=X^3+X^2+1$, то $V(X)$ определится как

$M(X)$	$X^6 +$	$X^4 + X^3 +$	1	
$G(X)$		$X^3 + X^2 + 1$		
		X^6	X^4	X^3
		X^6		
	X^8	X^6	X^5	X^3
	X^9	X^7		X^3
$V(X) = X^9 + X^8 + X^7 + X^6 + X^5 + X^4 + X^2 + 1$				

Следовательно, кодовая последовательность будет иметь вид:
1 1 1 1 1 0 1 0 1.

Во втором способе к k информационным разрядам добавляются $(n - k)$ контрольных разрядов:

$$V(X) = \frac{K \dots 2 \ 1}{\text{Информационные разряды}} \quad \frac{(n - k) \dots 2 \ 1}{\text{Контрольные разряды}}$$

Полином $V(X)$ степени n , содержащий информационные и контрольные разряды, должен быть сформирован так, чтобы он (как и при первом способе) делился без остатка на образующий полином $G(X)$.

Для кодирования полинома $M(X)$, содержащего информационные разряды, определяется остаток от деления

$$\frac{X^{n-k} M(X)}{G(X)} = Q(X) + \frac{R(X)}{G(X)}$$

или

$$X^{n-k} M(X) = Q(X) \cdot G(X) + R(X),$$

где $Q(X)$ — частное; $R(X)$ — остаток от деления $X^{n-k} M(X)$ на $G(X)$.

Следовательно, кодовый полином можно сформировать как $V(X) = X^{n-k} M(X) - R(X) = Q(X) \cdot G(X)$. Учитывая, что вычитание и сложение по модулю 2 эквивалентны, получим

$$V(X) = X^{n-k} M(X) + R(X) = Q(X) \cdot G(X).$$

Таким образом, $R(X)$ в результате деления имеет степень, меньшую $n-k$, и образует контрольные разряды, а полином $X^{n-k}M(X)$ не имеет коэффициентов, равных 1 при степенях X , меньших либо равных k . Следовательно, структура сообщения $V(X)$ имеет вид

$$\left| \begin{array}{c} \frac{X^{n-k}M(X)}{\text{информационных}} \\ \text{разрядов} \end{array} \right| \left| \begin{array}{c} \frac{R(X)}{\text{контрольных}} \\ \text{разрядов} \end{array} \right| \\ \left| \frac{\quad}{n\text{-разрядный кодовый полином } V(X)} \right|$$

Например, как и в предыдущем случае, требуется передать исходное сообщение $M(X)$ с разрядностью $k=7$ в виде полинома $V(X)$ разрядностью $n=10$. Воспользуемся образующим полиномом $G(X)=X^3+X^2+1$. Для кодирования сообщения необходимо полином $X^3M(X)=X^3(X^6+X^4+X^3+1)$ разделить на $G(X)$ и найти остаток. В результате получим

$$V(X)=(X^9+X^7+X^6+X^3)+(X^2+X)=\left|\frac{101\ 1001}{1}\right|\left|\frac{110}{2}\right|,$$

где 1 и 2 — соответственно информационные и контрольные разряды.

Если в канале передачи информации возникает ошибка $E(X)$, то на приемном конце сообщение будет иметь вид

$$H(X)=V(X)+E(X).$$

Обнаружение ошибки при приеме осуществляется делением $H(X)$ на образующий полином $G(X)$. Если остаток деления равен нулю, то сообщение считается правильным, если нет — то формируется сообщение об ошибке. Таким образом, ошибки не обнаруживаются, если остаток от деления $E(X)$ на $G(X)$ равен нулю.

Выбор вида образующего полинома $G(X)$ зависит от требования обнаружения ошибок определенного класса. В общем случае обнаруживающая способность кодов будет тем выше, чем большее число остатков может быть получено при делении полиномов $V(X)$ на образующий полином. Таким свойством обладают неприводимые (или примитивные) многочлены, которые и используются в качестве образующих полиномов. Метод обнаружения ошибок, основанный на делении на образующий многочлен, положен в основу сигнатурного анализатора.

Разнообразие схемных решений, применяемых при проектировании ИУВС, часто приводит к необходимости разработки нетиповых, специальных схем контроля. При этом используются свойства, присущие логическим элементам и узлам при выполнении ими некоторых операций, например:

1) контроль по характеру выходного кода (только одна из выходных шин дешифратора может находиться в единичном (нулевом) состоянии; выходы триггера должны иметь противоположное состояние и т. д.);

2) контроль дублированием — несколько идентичных схем, вы-

полняющих одинаковые операции, должны иметь на каждом шаге одинаковые значения выходного кода;

3) контроль сравнением исходного кода со значением входного кода, полученного восстановлением по выходным значениям.

Аппаратные методы контроля имеют следующие основные достоинства:

1) обеспечивают оперативный контроль практически в момент появления ошибки (неисправности) в работе устройства;

2) обнаруживаются случайные ошибки тем же оборудованием, что и систематические. В то же время введение только аппаратных средств контроля увеличивает общий объем оборудования и может привести к снижению суммарной надежности системы. Поэтому при разработке системы контроля руководствуются критериями эффективности, которые выбираются в зависимости от характера использования ИУВС. В частности, требование оптимального повышения коэффициента готовности достигается в том случае, если объем оборудования для контроля составляет 20—25 % от общего объема аппаратуры. Применение микропроцессоров для разработки ИУВС и особенно их основы — управляющих ЦВМ — суживает рамки применения аппаратных методов контроля. Это обстоятельство связано с использованием больших и сверхбольших интегральных схем (БИС и СБИС), составляющих микропроцессорные комплекты. Основная, операционная, обрабатывающая часть микроЭВМ оказывается скрытой в БИС и СБИС и недоступна для введения аппаратных средств контроля. При этом возрастающую роль приобретают программные методы контроля.

Программный метод контроля подразделяют на тестовый и программно-логический. Тестовый контроль представляет собой решение проверочной задачи с известными исходными данными и результатом. Такая задача может решаться в рабочем режиме (в определенные этапы исполнения основной программы) и в режиме контроля, когда ИУВС не выполняет основную задачу. Построение тестов для узлов и устройств (БИС и СБИС) состоит в отыскании минимальных последовательностей входных наборов, на которых обеспечивается поиск всех дефектов узла в заданном классе неисправностей и в случае необходимости места их возникновения.

Одним из возможных методов поиска входного набора X_d , диагностирующего дефект элемента G_d , является метод активизации одномерного пути. Суть метода состоит в выборе пути от неисправного элемента G_d к одному из выходов схемы G_n через последовательность элементов, находящихся на этом пути. Причем входы элементов в цепочке $G_d \dots G_j \dots G_n$ выбираются так, что выход элемента G_j зависит от входа, состояние которого определяется элементом G_{j-1} . Таким образом, о состоянии элемента G_d можно судить по значению выхода G_n . В свою очередь входы элемента G_d должны быть заданы так, чтобы о наличии или отсутствии дефекта можно было судить по его выходу. Установленный путь называется *активизированным* а сам процесс — *фазой продвижения вперед*. После выполнения фазы продвижения вперед выполняется обратный процесс с целью

определения входного набора, который обеспечит требуемые значения входов элементов, входящих в активизированный путь. Этот процесс называется *фазой продвижения назад*.

Существует достаточно большое количество других, более эффективных методов построения проверяющих тестов. В ИУВС применяют программно-логические методы контроля. Наиболее распространенным способом контроля является двойной и многократный счет со сравнением результатов вычислений. При сравнительной простоте реализации и достаточно высокой достоверности обнаружения случайных ошибок этот метод имеет существенные недостатки [33]: не позволяет обнаруживать систематические ошибки; более чем в два раза снижает производительность микроЭВМ; требует специального построения программы и распределения информации в оперативной памяти. Указанные недостатки ограничивают, а в ряде случаев делают невозможным применение данного метода, что приводит к необходимости использования других программно-логических методов. Основными из них являются: алгоритмический контроль; контроль исходной информации и результатов вычислений; программный контроль хода программы микроЭВМ и выполнения наиболее важных операций.

При реализации алгоритмического метода необходимая для контроля избыточная информация образуется тремя основными способами [31]:

- 1) контрольным решением основных задач по упрощенному алгоритму;
- 2) применением методов контроля по модулю и использованием специальных кодов;
- 3) использованием имеющихся или полученных косвенным путем дополнительных переменных.

Применение первого способа базируется на возможности представления основного алгоритма в виде упрощенного, дающего приближенное решение, и последующей совместной проверке результатов вычислений. При получении упрощенного алгоритма широко используются различные общие методы нахождения приближенных решений (например, интерполяции и аппроксимации), а также методы, использующие свойства конкретных алгоритмов. Достоинством данного способа является высокая вероятность обнаружения случайных ошибок и меньшие временные затраты по сравнению с методом двойного счета. Недостатком данного метода является сложность, а иногда и невозможность получения упрощенного алгоритма.

Второй способ основан на программной реализации методов по модулю и с помощью специальных кодов. При этом широко используются:

- контроль выполнения отдельных операций по модулю 2 и 3;
- контроль приема и передачи информации с использованием циклических кодов при обмене с внешними устройствами;
- получение контрольных сигнатур при выполнении стандартных (для данного алгоритма и типа микроЭВМ) процедур.

При реализации третьего способа в исходный алгоритм вводится

вспомогательный — за счет введения дополнительных переменных и ряда контрольных условий. Программная проверка выполнения этих условий позволяет контролировать правильность хода выполнения вычислительного процесса. В сложных ИУВС имеется развитая система датчиков входной информации, каждый из которых предназначен для точного измерения какого-либо параметра. В ряде случаев удастся по совокупности параметров, полученных от некоторых датчиков, косвенно, расчетным путем получить: приближенное значение параметров от других датчиков; приближенное значение переменных, используемых в основных алгоритмах как результат ряда промежуточных вычислений. Указанный способ широко используется в ИУВС, является весьма эффективным, позволяет обнаруживать ошибки микроЭВМ, а также ошибки в исходных данных.

Контроль исходной информации и результатов вычислений в ИУВС основан на использовании контрольных соотношений, имеющих место для реальных систем. Основные способы контроля следующие:

- стробирование входных данных, промежуточных и окончательных результатов;

- проверка на наличие неразрешенных кодовых комбинаций.

Стробирование — это сравнение контролируемого параметра X_i с границами его допустимых значений $X_i^{\max}$ и $X_i^{\min}$. В общем случае для ИУВС стробирование производят и по времени появления контролируемого параметра в допустимом диапазоне $t_i^{\max}$ и $t_i^{\min}$. Причем границы допустимых значений могут быть как фиксированными, так и переменными.

Контроль на наличие неразрешенных кодовых комбинаций осуществляется при формировании ИУВС дискретных управляющих воздействий в момент времени t_{n+1} по известным входным значениям, характеризующим состояние объекта управления в момент времени t_n . Контроль реализуется сравнением полученных входных значений и сформированных управляющих воздействий с заранее составленным и хранимым в памяти микроЭВМ списком допустимых значений.

Программный контроль хода программы и выполнения наиболее важных операций реализуется в микроЭВМ при отсутствии аппаратных средств контроля или их недостаточной эффективности.

ЗАКЛЮЧЕНИЕ

Большие интегральные схемы изменили все области электронной техники настолько быстро, что последствия их воздействия можно наблюдать почти каждый день. Главной движущей силой по-прежнему остается технология ИС, причем ее достижения дополняются соответствующими сдвигами и в других разделах электронной техники и технологии. Каковы же основные тенденции развития элементов и технологии средств вычислительной техники? На уровне кристаллов происходит завоевание КМДП-технологией ведущих позиций в области производства БИС, которые могут содержать одновременно аналоговые и цифровые схемы.

В области «чисто» аналоговых ИС продолжается совершенствование комбинированных технологий схем с полевыми и биполярными транзисторами.

В области полупроводниковых ЗУ большой емкости (это относится как к статическим и динамическим ОЗУ, так и к ППЗУ с электрическим и ультрафиолетовым стиранием) широко используют резервирование на уровне кристалла, хотя механизмы программирования этих резервных элементов и уровень информационной емкости, на котором их применение экономически оправдано, пока еще окончательно не установлены.

Для обработки аналоговых сигналов на современном этапе характерны цифровые методы, в результате чего операционный усилитель вытесняется микропроцессорами, ставшими универсальными компонентами электронных конструкций. Тем не менее специалисты по аналоговым схемам продолжают создавать микросхемы с более высокой степенью интеграции, предназначенные для универсальных подсистем. На базе АЦП, ЦАП, коммутаторов, схем выборки и хранения, операционных усилителей и других аналоговых элементов разрабатывают операционные узлы в виде БИС, способные обрабатывать аналоговую информацию без преобразования ее в цифровую форму.

Датчики, пожалуй, являются теми устройствами, в которых острее всего нуждаются производственные участки предприятий, особенно промышленные роботы.

В области преобразования данных основной движущей силой является стремление к повышению точности и быстродействию. Однако существенное значение начинают приобретать и новые факторы: сильный сдвиг в сторону КМДП-технологии, разработка преобразователей специального назначения и использование новых методов преобразования, в том числе схем коррекции погрешностей.

Весьма сложную задачу представляет собой организация ввода-вывода информации. Это связано с огромным разнообразием периферийных устройств, которые необходимы в микроЭВМ.

Применение языков программирования интерпретирующего типа ФОКАЛ (БЭЙСИК) для автоматизированных систем контроля в большинстве случаев оправдано благодаря их простоте, легкости изучения, гибкости, простым средствам отладки и редактирования. Однако языкам интерпретирующего типа свойственно их сравнительно небольшое быстродействие, которое сужает класс решаемых задач. Альтернативой будет использование компилирующих систем. Использование компиляторов для микроЭВМ, не имеющей внешней (дисковой) памяти, чрезвычайно сложно ввиду необходимости многократного ввода и вывода перфолент.

В тех случаях, когда к скорости выполнения программ в АСК предъявляются повышенные требования, целесообразно применение системы QUASIK. Система содержит компилятор для языка, близкого к языку БЭЙСИК, и оснащена средствами машинного уровня. Язык имеет средства для работы с аппаратурой в стандарте КАМАК. Вычислительные средства являются дополнительными и требуют большей памяти. Подготовка, трансляция и отладка программ производятся без промежуточного обращения к перфоленте. Минимальный объем памяти, необходимый для работы системы, составляет примерно 8К слов.

Быстрый прогресс в области вычислительной техники и ряде связанных с ней областей техники и технологии привел к созданию компьютеров с высокими техническими характеристиками и развитыми функциональными возможностями, имеющих низкую стоимость, малые габариты и удобство эксплуатации, благодаря чему ЭВМ в ближайшее время начнет использоваться в повседневной жизни практически каждого человека.

СПИСОК ЛИТЕРАТУРЫ

1. Автоматизированная система контроля сложных изделий на основе микро-ЭВМ//Г. И. Фролов, Р. А. Гембицкий, Г. Н. Капустина, Г. Н. Темирова// 26. научных трудов МИЭТ. Алгоритмы и программы системного математического обеспечения ЕС ЭВМ и кросс-средств. М., 1982.
2. Евреинов Э. В., Хорошевский В. Г. Однородные вычислительные системы.— Новосибирск: Наука, 1978.
3. Специализированные ЦВМ/Под ред. В. Б. Смолова.— М.: Высш. шк., 1981.
4. Наумов Б. Н. Мини-ЭВМ.— М.: Статистика, 1980.
5. Микропроцессорная шина, ориентированная на перспективные 32-разрядные системы//Экспресс-информация. Вычислительная техника. 1981. № 8.
6. Лестер, Сандерс. Однокристальный кодер-декодер для преобразования кода без возвращения к нулю в манчестерский код//Электроника. 1982. № 15.
7. ИС в качестве устройства сопряжения для бортовых систем информационного обмена//Электроника. 1981. № 26.
8. Разработка архитектуры локальной сети, объединяющей различные рабочие станции/Р. Райан, Дж. Маршалл, Р. Бич, С. Кермал//Электроника. 1981. № 17.
9. Преснухин Л. Н., Воробьев Н. В., Шишкевич А. А. Расчет элементов цифровых устройств/Под ред. Л. Н. Преснухина.— М.: Высш. шк., 1982.
10. Воробьев Н. В., Вернер В. Д. Микропроцессоры. Элементная база и схемотехника средств сопряжения/Под ред. Л. Н. Преснухина.— М.: Высш. шк., 1984.
11. Завьялов Ю. С. Экстремальное свойство кубических многозвенников (сплайнов) и задача сглаживания//Вычислительные системы, 42.— Новосибирск: Ин-т математики СО АН СССР, 1970.
12. Колин К. К., Линаев В. В. Проектирование алгоритмов управляющих ЦВМ.— М.: Сов. радио, 1980.
13. Горелик А. Л., Бутко Г. И., Белоусов Ю. А. Бортовые цифровые вычислительные машины.— М.: Машиностроение, 1975.
14. Мячев А. А. Организация управляющих вычислительных комплексов.— М.: Энергия, 1980, 262 с.
15. Надежность электронных элементов и систем/Под ред. Х. Шнайдера.— М.: Мир, 1977.
16. Шаньгин В. Ф., Костин А. Е. Микропроцессоры. Организация вычислительных процессов на микроЭВМ/Под ред. Л. Н. Преснухина.— М.: Высш. шк., 1984.
17. Горбунов В. Л., Панфилов Д. И., Преснухин Д. Л. Основы построения микроЭВМ/Под ред. Л. Н. Преснухина.— М.: Высш. шк., 1984.
18. Балибердин В. А. Методы анализа мультипроцессорных систем/Под ред. С. Д. Пашкеева.— М.: Радио и связь, 1982.
19. Манзелей И. А. Архитектура вычислительных систем и тенденции их развития//Зарубежная радиоэлектроника. 1980. № 11.
20. Прангис И. В., Стецюра Г. Г. Микропроцессорные системы.— М.: Наука, 1980.
21. Балашов Е. П., Пузанков Д. В. Микропроцессоры и микропроцессорные системы/Под ред. В. Б. Смолова.— М.: Радио и связь, 1981.
22. Нестеров П. В. Микропроцессоры. Архитектура и ее оценка/Под ред. Л. Н. Преснухина.— М.: Высш. шк., 1984.
23. СМ ЭВМ. СМ-3, СМ-4. Программное обеспечение. Перфоленточная диалоговая система программирования ДС СМ. Описание языка, 1978.
24. Архитектура устойчивых к отказам аэрокосмических ЭВМ.— Экспресс-информация//Вычислительная техника. 1979. Вып. 23.
25. Шахнов В. А., Романов Ф. И. Конструкционные системы микроЭВМ.— М.: Радио и связь, 1983.
26. Бут Н. И., Садовий И. Т., Банас В. И. Прогнозирование основных конструктивных параметров печатных узлов РЭА//Вопросы радиоэлектроники, вып. 11. Серия «Электронная вычислительная техника», 1978.
27. Карнушин В. Б. Виброшумы радиоаппаратуры.— М.: Сов. радио, 1977.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

Амортизация блоков 278

Варианты конструкций 268

Вероятность отказа 252

Вибропрочность 277

Восстановление аналоговых сигналов 39

Время апертурное 116

— выборки 116

Вычитатель 100

Датчики 22

Дельта-сети 202

Дешифратор 92

Доступ к памяти прямой 142

Изделия восстанавливаемые 251

— невозстанавливаемые 251

Индикаторы двончные 110

— сегментные 111

— — с мультиплексным управлением 112

Интенсивность отказов 252

Интерпретатор 139

Интерфейс 40

Источник опорного выражения 120

Канал общего пользования 51

— передачи 41

Ключ аналоговый 80

Код дополнительный 117

— Манчестер II 57

— прямой 117

— смещенный 117

Коды для передачи слов 57

Команда 190

Коммутатор 80

Компаратор аналоговый 81

Компилятор 139

Конструирование тепловое 273

Контроль систем 289

Критерий среднеквадратический 167

Линия связи 40, 50

Магистраль 40

Мажорирование 260

Матрица логическая некоммутированная 108

— — программируемая 107

Микропроцессор аналоговый 128

Мультиплексор 94

— аналоговый 112

Надежность изделия 251

Накопитель ЗУ с однокоординатной выборкой 104

— — двухкоординатной выборкой 104

Наработка до отказа 254

Обмен данными 141

Обращение рекурсивное 159

Параметр потока восстановлений 254

Передача данных 45

— сквозная 117

Подмагистраль 40

Полусумматор 99

Помехи 280

Поток восстановлений 254

— отказов 254

Преобразователь аналого-цифровой 117

— уровней 62

— цифроаналоговый 117

Прерывания 142, 191

— внешние 161

— внутренние 163

Программа 138

— компоновщик 140

— отладчик 140

Разрядность адреса 185

Регистр 88

Редактор текста 139

Режим обмена дуплексный 45

— — мультиплексный 45

— — полудуплексный 45

— — симплексный 45

Резервирование 258

Сбор данных 11, 15
Сдвигатель 90
Селектор 97
Синхронизаторы импульсные 86
Синхронизация 82
Система информационно-управляющая 166
— — архитектура 195
— — техническое задание 193
— контроля автоматизированная 133
— команд 155, 189, 219
— прерывания 161
— программирования 138
Скорость разрушения информации 116
Способы адресации 144
Структура устройств ввода 11
Сумматор двоичный полный 99
— многоразрядный 100
Схема выборки-хранения 115
Счетчик вычитающий 88
— реверсивный 88
— суммирующий 87

Таймер 87, 165
Такт машинный 85

Усилитель операционный 75
Устройство арифметическо-логическое 101
— запоминающее 104

— — оперативное 106
— — постоянное 106
— — программируемое 107
— множително-делительное аналого-цифровое 124
— пересчетное 88

Фильтрация сигналов 33
Фильтр Баттерворта 34
— Чебышева 34
ФОКАЛ 225
Функция дискретная 167

Частота отказов 253

Цикл командный 86
— машинный 85

Шифратор 92

Экранирование 288
Элемент памяти 104
— — статический 105
— КМДП-типа 73
— ТТЛ-типа 69
— ЭСЛ-типа 71

ОГЛАВЛЕНИЕ

Введение	3
I. СХЕМОТЕХНИКА СРЕДСТВ СОПРЯЖЕНИЯ	
1. Средства сопряжения микроЭВМ с объектом	11
1.1. Средства автоматического ввода и вывода данных в системах сбора и обработки информации	11
1.2. Датчики первичной информации	22
1.3. Интерфейсы	40
1.4. Схемотехника цифровых и аналоговых интегральных схем малой, средней и большой степени интеграции	67
1.5. Элементная база аналоговых устройств средств сопряжения	75
1.6. Узлы цифровых устройств средств сопряжения	82
1.7. Узлы цифроаналоговых устройств средств сопряжения	112
II. КОНТРОЛИРУЮЩИЕ И ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ	
2. Микропроцессорные контролирующие и информационно-управляющие вычислительные системы	
2.1. Особенности проектирования контролирующих систем на микроЭВМ с универсальной системой команд	133
2.2. Основные требования, предъявляемые к микропроцессорным информационно-управляющим вычислительным системам	166
2.3. Архитектура микропроцессорных информационно-управляющих вычислительных систем	195
2.4. Особенности проектирования информационно-управляющих вычислительных систем на микроЭВМ со специализированной системой команд	207
3. Программное обеспечение информационно-управляющих вычислительных систем	
3.1. Обоснование выбора языка программирования	223
3.2. Общие сведения о языке ФОКАЛ	225
3.3. Общие сведения об интерпретаторе языка ФОКАЛ	229
3.4. Разработка подпрограмм-функций обслуживания внешних устройств	232
3.5. Программирование устройств выдачи и приема релейных сигналов	240
3.6. Общие принципы построения программного обеспечения автоматизированных систем контроля	243
III. КОНСТРУИРОВАНИЕ И НАДЕЖНОСТЬ ИНФОРМАЦИОННО-УПРАВЛЯЮЩИХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ	
4. Обеспечение надежности и особенности конструирования микропроцессорных информационно-управляющих вычислительных систем	
4.1. Обеспечение надежности микропроцессорных информационно-управляющих вычислительных систем	251
4.2. Особенности конструирования микропроцессорных информационно-управляющих вычислительных систем	268
4.3. Организация контроля микропроцессорных информационно-управляющих вычислительных систем	289
Заключение	298
Список литературы	300
Предметный указатель	301

Владимир Дмитриевич Вернер, Николай Васильевич Воробьев,
Александр Васильевич Горячев, Георгий Иванович Фролов,
Александр Адамович Шишкевич

МИКРОПРОЦЕССОРЫ

Кн. 2

СРЕДСТВА СОПРЯЖЕНИЯ.



КОНТРОЛИРУЮЩИЕ И ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

Зав. редакцией С. Ю. Липец

Редактор С. С. Голод

Мл. редактор С. А. Когадеева

Художественный редактор Ю. С. Сергачев

Технический редактор И. П. Тихонова

Корректор Т. К. Скрипкина

ИБ № 2474

Сдано в набор 03.02.87. Подписано в печать 18.11.87. АТ 16802. Формат 60×
×90¹/₁₆. Бумага офсетная. Гарнитура литературная. Офсетная печать. Усл. печ.
л. 19. Усл. кр.-отт. 19. Уч.-изд. л. 20,73. Тираж 50 000 экз. Зак. 114. Цена 1 р. 10 к.

Издательство «Высшая школа» Государственного комитета БССР по делам
издательств, полиграфии и книжной торговли. 220048, Минск, проспект Машеро-
ва, 11.

Минский ордена Трудового Красного Знамени полиграфкомбинат МППО им. Я. Ко-
ласа. 220005, Минск, ул. Красная, 23.