

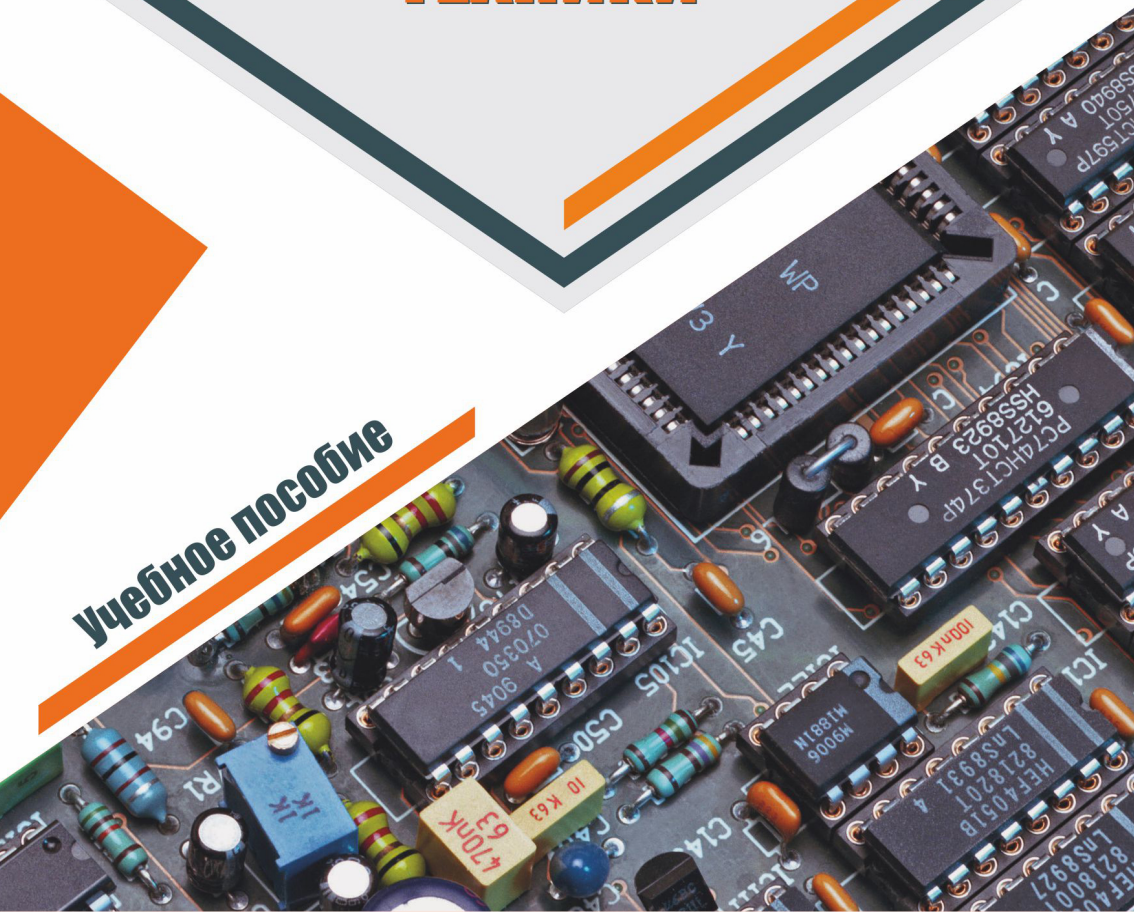


Центр
учебной книги
и средств обучения
РИПО

Д. А. Кушнер
А. В. Дробов
Ю. Л. Петроченко

ОСНОВЫ АВТОМАТИКИ И МИКРОПРОЦЕССОРНОЙ ТЕХНИКИ

Учебное пособие



Д. А. Кушнер
А. В. Дробов
Ю. Л. Петроченко

ОСНОВЫ АВТОМАТИКИ И МИКРОПРОЦЕССОРНОЙ ТЕХНИКИ

Допущено Министерством образования Республики Беларусь
в качестве учебного пособия для учащихся
учреждений образования, реализующих образовательные
программы среднего специального образования
по специальностям «Монтаж и эксплуатация
электрооборудования», «Городской электрический
транспорт», «Автоматизированные электроприводы»



МИНСК
РИПО
2019

УДК 681.5(075.32)
ББК 32.965я723
К96

А в т о р ы:

заместитель директора по учебно-производственной работе
Д. А. Кушнер; преподаватели *А. В. Дробов*,
Ю. Л. Петроченко УО «Гомельский государственный
профессионально-технический колледж электротехники».

Рецензенты:

цикловая комиссия общепрофессионального цикла и спецдисциплин
специальности «Электроника механических транспортных средств»
филиала БНТУ «Минский государственный политехнический колледж»
(*И. И. Буткевич*);

заведующий кафедрой «Автоматизированные системы управления
производством» УО «Белорусский государственный аграрный технический
университет», кандидат технических наук, доцент *А. Г. Сеньков*.

*Все права на данное издание защищены. Воспроизведение всей книги или
любой ее части не может быть осуществлено без разрешения издательства.*

*Выпуск издания осуществлен при финансовой поддержке Министерства
образования Республики Беларусь.*

Кушнер, Д. А.

К96 Основы автоматика и микропроцессорной техники : учеб. по-
собие / Д. А. Кушнер, А. В. Дробов, Ю. Л. Петроченко. — Минск :
РИПО, 2019. — 245 с. : ил.

ISBN 978-985-503-853-6.

В учебном пособии изложены основы теории автоматического управ-
ления. Особое внимание уделено замкнутым системам автоматического
управления непрерывного действия, дискретным автоматам. Рассмотрены
элементы систем автоматика, их классификация, характеристики и режи-
мы работы (датчики, усилительные устройства и элементы вычислитель-
ной техники, включающие в себя элементы памяти, триггеры, регистры,
счетчики, дешифраторы, шифраторы и формирователи импульсов).

Предназначено для учащихся учреждений среднего специального об-
разования по специальностям «Монтаж и эксплуатация электрообору-
дования», «Городской электрический транспорт», «Автоматизированные
электроприводы».

УДК 681.5(075.32)
ББК 32.965я723

ISBN 978-985-503-853-6

© Кушнер Д. А., Дробов А. В.,
Петроченко Ю. Л., 2019
© Оформление. Республиканский институт
профессионального образования, 2019

ВВЕДЕНИЕ

Автоматика (от греч. *αὐτόματος* – самодействующий) – отрасль науки и техники, которая разрабатывает технические средства и методы для осуществления технологических процессов без участия человека.

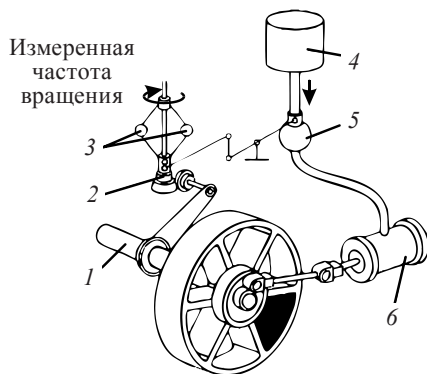
Первые машины, работающие без участия человека, были созданы древнегреческим ученым Героном Александрийским, жившим в I в. н. э. Существуют описания около 100 автоматических устройств, известных в те древние времена, которые использовались, как правило, в развлекательных и религиозных целях.

Первое автоматическое устройство практического назначения изобрел нидерландский ученый Христиан Гюйгенс в 1657 г. – автоматический регулятор маятникового типа для стабилизации скорости хода часового механизма.

Вместе с промышленной революцией в Европе в XVIII–XIX вв. появились и первые автоматические устройства промышленного назначения, когда получили свое развитие паровые машины и турбины.

В 1784 г. английский механик Джеймс Уатт получил патент на центробежный регулятор скорости паровой машины (см. рисунок), используемый для поддержания постоянства частоты вращения. Принцип работы регулятора заключался в поддержании заданного значения физического параметра не точно, а в некотором определенном диапазоне.

К первым промышленным регуляторам также относят программное устройство управления ткацким станком от перфокарты, построенное в 1808 г. французским изобретателем Жозефом Мари Жаккаром. Оно применялось для воспроизведения крупноузорчатых тканей.



Центробежный регулятор Уатта:

1 – выходной вал; 2 – регулятор; 3 – металлические шарики;
4 – паровой котел; 5 – клапан; 6 – паровая машина

Вышеприведенные изобретения дали старт изобретениям принципов регулирования и регуляторов, продолжающимся вплоть до середины XX в.

Первоначально работы по созданию автоматических систем в механике, электротехнике, теплотехнике и других научных отраслях велись независимо друг от друга. В начале 1940-х годов автоматика формируется как самостоятельная научная дисциплина, изучающая методы анализа и синтеза систем автоматического управления в технике независимо от их физической природы.

В настоящее время автоматические системы широко применяются во всех областях деятельности человека: в промышленности, на транспорте, в устройствах связи, при научных исследованиях и др. Способствовало этому и бурное развитие современной микропроцессорной техники, которая стала важнейшим средством при решении самых разнообразных задач в области сбора и обработки данных, а также разработки систем автоматического управления (САУ).

Развитие микроэлектроники, вычислительной техники в последние годы дало резкий скачок повышения качества функциональных возможностей цифровых САУ. Это заключается не только в повышении надежности и отказоустойчивости, обеспечении стабильности функционирования, но и в придании цифровым системам автоматического управления принципиально новых

свойств, таких как малогабаритность и перенастраиваемость структуры, адаптивность, способность решать вычислительные и логические задачи, самоконтроль и др.

Программируемость микропроцессоров определяет возможность гибкой оперативной перестройки как алгоритма работы САУ, так и ее структуры в целях приспособления их к меняющимся условиям работы. Свойство программируемости обеспечивает возможность внесения изменений в структуру и в программу работы системы на всех этапах ее существования — от предварительного проектирования до эксплуатации серийных образцов.

Значительная вычислительная мощность микропроцессоров и систем на их основе, величина которой, по оценкам специалистов, достигнет нескольких сотен миллионов операций в секунду, создает хорошие предпосылки для использования микропроцессоров в быстродействующих системах реального времени.

Применение микропроцессоров в САУ позволяет поднять на качественно новый уровень такие важные ее характеристики, как отказоустойчивость и живучесть. Отказоустойчивость, т. е. способность системы сохранять свою работоспособность при возникновении в системе разнообразных отказов, обеспечивается в микропроцессорных САУ введением аппаратурной, программной и информационной избыточности. Широкое распространение получают системы с резервированием, а также с программной реконфигурацией структуры и использованием самокорректирующихся кодов. Для оперативного контроля и диагностики все шире используют встроенный программно-аппаратный контроль, осуществляемый с привлечением относительно дешевых дополнительных ресурсов.

Таким образом, успехи автоматизации, в том числе и с применением микропроцессорной техники, обеспечили ей ведущее место практически во всех областях деятельности человека. Вместе с тем перспективы развития теории и приложений автоматического управления дают основания полагать, что главные достижения в этой области еще впереди.

1. СИСТЕМЫ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

1.1. Общие сведения о системе

Основным понятием теории управления, а также связанных с ней научных дисциплин (кибернетики, информатики, системного анализа и др.) является понятие «система».

Система — это совокупность упорядоченных объектов, в которой определены цели. Все, что не входит в систему, является частью среды. Система характеризуется связями между входными и выходными сигналами, вид которых определяется выбором границы между системой и средой. Например, системой может являться отдельный технологический агрегат, станок или машина, производственное предприятие или группа предприятий.

В каждой системе должны присутствовать определенные свойства.

1. Целостность. Система представляет собой целостную совокупность элементов. Это означает, что система — единое образование, в ее состав входят взаимосвязанные компоненты, которые в условиях системы приобретают системно-определенные свойства. Для системы первичным является признак целостности, а не множества входящих в нее элементов.

2. Устойчивые связи. В любой системе устанавливаются те или иные связи (отношения) между элементами. Связь интерпретируется как физический канал, по которому обеспечивается обмен между элементами системы и системы с окружающей средой, веществом, энергией и информацией.

Основные характеристики связи: физическое содержание, направленность, мощность и роль в системе. По физическому содержанию связи можно разделить на вещественные, энергетич-

ческие, информационные, смешанные и ненаполненные (отношения). По направленности различают следующие виды связей: прямые, обратные, контрсвязи и нейтральные. Важной характеристикой отношений и связей в системе является их сила (или мощность). Роль связей в системе определяется характером их влияния на процессы в системе: соединительные, ограничивающие, усиливающие, запаздывающие, селективирующие, преобразующие, координирующие, обратные и др.

3. Организация. Каждая система характеризуется наличием определенной организации. Мерой организации системы, т. е. ее устойчивой тенденции к сохранению или изменению в определенном направлении своего состояния вопреки внутренним деструктивным процессам и случайным внешним воздействиям, является информация.

4. Интегративные качества. Это такие свойства, которые присущи системе в целом, но не проявляются в отдельных элементах. Наличие интегративных качеств показывает, что свойства системы зависят от характеристик ее элементов, но не определяются ими полностью. Это означает, что система не сводится к простой совокупности элементов, а применение декомпозиции, т. е. разделение системы на отдельные части и изучение отдельных частей, не позволяет исследовать свойства системы в целом.

5. Целенаправленность функционирования. Определение цели системы — сложная задача, которую не всегда удается определить традиционным образом. В общем случае цель — желаемый результат деятельности системы, достижимый в пределах некоторого интервала времени.

Цели системы непосредственно связаны с процессами управления. Под **целью управления** принято понимать количественно определенную область требуемых значений с учетом параметров, критериев и времени, отвечающих при их полном соблюдении оптимальному функционированию системы. Следует отметить, что при частичном достижении целей управления системе может быть причинен «ущерб».

Под **эффективностью системы** понимают успешность достижения цели. Для оценки эффективности систем используют характеристики, являющиеся отношением между показателями полезности результатов управления в соответствии с целями и

затратами, потерями в процессе управления. При этом задачи управления принято разделять на два основных класса:

- задачи, в которых управление осуществляется в целях достижения определенного результата;
- задачи, в которых управление осуществляется в целях получения наилучшего (экстремального) значения некоторой величины, характеризующей конечный эффект управления.

По характеру взаимодействия со средой различают замкнутые и открытые системы. В замкнутых системах отсутствуют процессы обмена энергией, веществом и информацией со средой, в открытых – происходит постоянное взаимодействие со средой путем обмена энергией, веществом и информацией.

1.2. Основные понятия автоматизации

Автоматизация производства – это процесс в развитии производства, при котором функции контроля и управления, ранее выполнявшиеся человеком, передаются техническим устройствам (средствам и приборам автоматизации).

Цели автоматизации производства: сокращение численности обслуживающего персонала; повышение качества продукции; снижение расходов сырья и различных видов энергии; сокращение отходов производства; повышение ритмичности производства и производительности труда; увеличение объема выпускаемой продукции; улучшение условий труда работников; экологичность производства.

Автоматизация производственных процессов развивается в двух направлениях:

- создание систем локальной автоматизации технологических процессов, которые обеспечивают стабилизацию заданных рабочих режимов путем поддержания постоянными или изменяющимися по заданной программе параметров процесса;
- создание автоматизированных систем управления технологическими процессами и производством в целом с применением вычислительной техники.

Автоматизация производственных процессов в зависимости от поставленных задач может быть частичной, комплексной и полной.

Частичная автоматизация отдельных производственных процессов, устройств, элементов оборудования реализуется просты-

ми техническими средствами без сложной подготовки автоматизируемого оборудования и процессов.

Комплексная автоматизация проводится на участке или в цехе, которые функционируют как единый взаимосвязанный автоматизированный комплекс, т. е. при участии людей. Комплексная автоматизация возможна при высокоразвитом производстве с совершенной технологией и при прогрессивных методах управления с применением надежного производственного оборудования, действующего по заданной или саморегулирующейся программе.

При комплексной автоматизации функции человека состоят в контроле и управлении работой комплекса.

Полная автоматизация предусматривает передачу всех функций управления и контроля производством автоматическим системам управления. Полностью автоматизируют рентабельные производства с устойчивой программой развития, режимы работы которых неизменны в обозримом будущем. Особенно важна полная автоматизация в условиях производств, опасных для жизни и здоровья человека.

При автоматизации технологических процессов применяют локальные системы различных классов, в том числе системы автоматического управления (САУ).

Система автоматического управления – комплекс устройств, предназначенных для автоматического изменения одного или нескольких параметров объекта управления с целью установить требуемый режим его работы. Под САУ также понимают совокупность объекта управления и устройства управления.

В зависимости от уровня автоматизации производства под *объектом управления (ОУ)* понимают отдельный аппарат, машину, агрегат, технологический участок, технологическую линию или цех. *Устройством управления (УУ)* называют комплекс технических средств, обеспечивающих получение информации о состоянии ОУ, сравнение ее с заданными технологическими параметрами и, при рассогласовании параметров ОУ с заданием, выдачу управляющего воздействия в ОУ.

При этом качество работы ОУ могут определять несколько технологических параметров, так же как и управляющие воздействия могут вводиться по нескольким каналам. Технологические параметры, определяющие качество выпускаемой из объекта продукции, называют *качественными*. Параметры продуктов и

энергоносителей на входе в ОУ, которые влияют на качественные параметры, называют *количественными*.

Рассмотрим функциональную схему системы автоматического управления.

Разделение САУ на отдельные элементы целесообразно, поскольку это позволяет осуществить разработку САУ и произвести ее анализ, а также определить функциональную сущность отдельных ее элементов. Кроме того, предоставляется возможность построить функциональную схему САУ. В зависимости от числа контролируемых (регулируемых) качественных параметров САУ могут быть одно- и многопараметровыми. На рисунке 1.1 показана функциональная схема многоконтурной САУ.

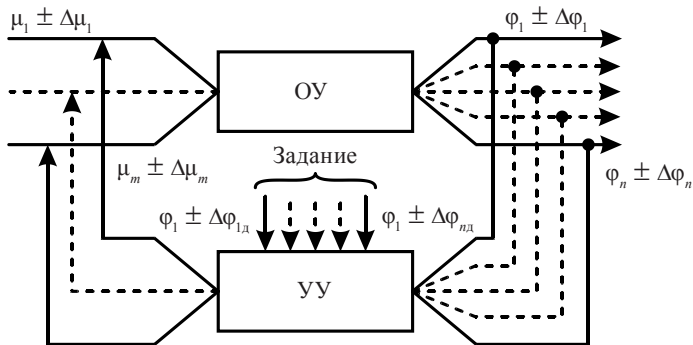


Рис. 1.1. Функциональная схема многоконтурной системы автоматического управления

Из объекта выходит продукт, который характеризуется качественными параметрами φ от 1 до n . Продукт имеет отклонения качественных параметров от номинального φ на величину $\Delta\varphi$. Эта информация поступает в устройство управления УУ, которое сравнивает отклонения качественных параметров $\Delta\varphi_1$ с допускаемыми $\Delta\varphi_{1д}$. При превышении $\Delta\varphi_1$ по отношению к $\Delta\varphi_{1д}$ УУ выдает управляющее воздействие $\mu_1 \pm \Delta\mu_1$ на количественные параметры m , поступающие в ОУ. Число количественных параметров m не обязательно должно совпадать с числом качественных параметров n , так как один и тот же количественный параметр может влиять на несколько качественных одинаково.

1.3. Назначение систем автоматического управления

По своему назначению САУ классифицируют на четыре основных типа:

- система автоматического регулирования (САР);
- система автоматической блокировки (САБ);
- система программного управления (СПУ);
- система операторного управления (СОУ).

Система автоматического регулирования (рис. 1.2, а) предназначена для автоматического управления ОУ (поддержания величины качественного параметра φ с допускаемыми отклонениями $\pm\Delta\varphi$). Система состоит из следующих элементов:

- Д(ПП) – *датчик (первичный преобразователь)* – устройство, предназначенное для восприятия величины качественного параметра φ и преобразования его в другую физическую величину, удобную для преобразования и передачи на расстояние;
- СП – *согласующий преобразователь*, предназначенный для согласования выходного сигнала с датчика во входной сигнал последующего элемента схемы (при их рассогласовании);
- ИП – *измерительный прибор*, предназначенный для преобразования выходного сигнала ПП или СП в визуальную информацию. Если она не требуется, то в САР может отсутствовать;
- БФЗР – *блок формирования закона регулирования*, предназначенный для сравнения текущего значения качественного параметра с заданным и, при их рассогласовании, выдачи управляющего воздействия в ОУ. В одноконтурных САР в качестве БФЗР используют регулирующий прибор, а в многоконтурных – управляющий микропроцессор;
- ИМ – *исполнительный механизм*, преобразующий управляющий сигнал от БФЗР в механическое перемещение;
- РО – *рабочий орган* – устройство непосредственного воздействия на потоки продукта или энергоносителя (кран, вентиль и т. д.).

Система автоматической блокировки (рис. 1.2, б) предназначена для отключения либо переключения потоков продукта или энергоносителя при возникновении аварийных ситуаций или резкого отклонения качественного параметра от заданного. На этом рисунке БУ – *блокирующее устройство*, предназначенное для выдачи управляющего воздействия на закрытие или переключение РО.

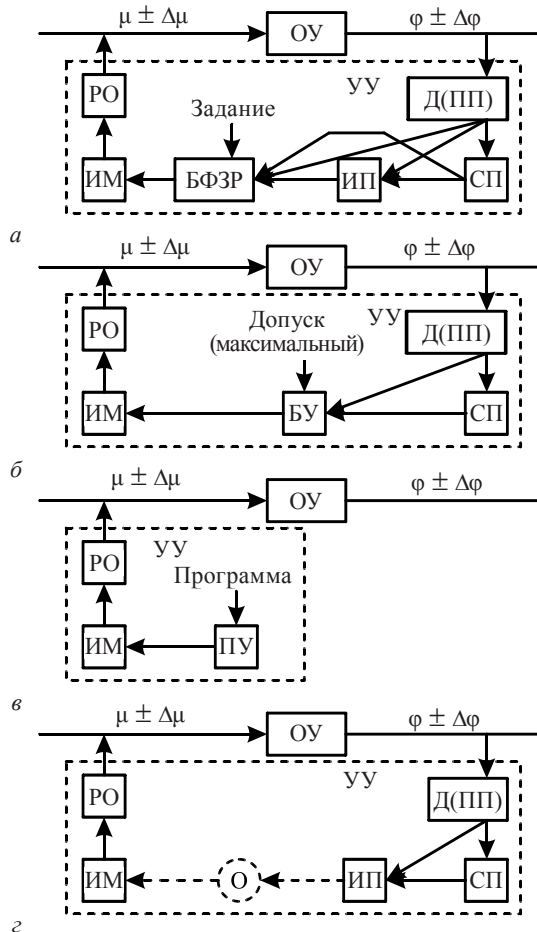


Рис. 1.2. Функциональные схемы системы автоматического управления

Остальные элементы схемы имеют то же назначение, что и в САР.

Система программного управления (рис. 1.2, *в*) предназначена для управления ОУ по заранее заданной временной программе, вне зависимости от величины качественного параметра. На этом рисунке ПУ – *программное устройство*, предназначенное для выдачи управляющих воздействий в ОУ по заданной временной

программе. В отличие от предыдущих (замкнутых САУ) относится к разомкнутым системам управления.

Система операторного управления (рис. 1.2, з) относится к человеко-машинным системам автоматизации, в которых информация о состоянии ОУ поступает с помощью технических средств, а управляющее воздействие в него вносится человеком-оператором.

1.4. Классификация элементов систем автоматического управления

Любая САУ состоит из большого числа элементов, которые чрезвычайно разнообразны по выполняемым функциям, конструкции, принципу действия, характеристикам, физической природе преобразуемых сигналов и т. д. Соответственно их можно классифицировать по ряду признаков. Например, в зависимости от того, как элементы получают энергию, необходимую для преобразования входных сигналов, они делятся на пассивные и активные.

Пассивные элементы автоматики — это элементы, у которых входное воздействие (сигнал $x_{вх}$) преобразуется в выходное воздействие (сигнал $x_{вых}$) за счет энергии входного сигнала (например, редуктор).

Активные элементы автоматики для преобразования входного сигнала используют энергию от вспомогательного источника (например, двигатель, усилитель).

В зависимости от энергии на входе и выходе элементы автоматики подразделяют на *электрические, гидравлические, пневматические, механические* и *комбинированные*.

По выполняемым функциям в САУ элементы автоматики можно разделить:

- на датчики;
- усилители;
- исполнительные устройства;
- реле;
- вычислительные элементы;
- согласующие элементы;
- вспомогательные элементы и т. д.

Датчики воспринимают поступающую на их вход информацию об управляемой величине объекта управления и преоб-

разуют ее в форму, удобную для дальнейшего использования в устройстве автоматического управления. Большинство датчиков преобразует входной неэлектрический сигнал $x_{вх}$ в выходной электрический сигнал $x_{вых}$. В зависимости от вида входного неэлектрического сигнала $x_{вх}$ выделяют:

- датчики механических величин (датчики перемещения, датчики скорости, датчики ускорения и т. д.);
- датчики тепловых величин (датчики температуры);
- датчики оптических величин (датчики излучения) и т. д.

Часто применяют датчики с двойным преобразованием сигнала. Например, входной неэлектрический сигнал $x_{вх}$ сначала преобразуется в перемещение, а затем перемещение преобразуется в выходной электрический сигнал $x_{вых}$.

Так, в САР при регулировании давления жидкости в трубопроводе применяют датчики давления с емкостными элементами преобразования, в которых воздействие давления вызывает изменение положения измерительной мембраны обкладки конденсатора. Это изменение приводит к изменению выходного значения датчика, которым является электрическая емкость.

Усилители — это элементы автоматики, которые осуществляют количественное преобразование, усиление мощности входного сигнала $x_{вх}$. В некоторых случаях одновременно с количественным преобразованием усилители осуществляют и качественное преобразование (например, преобразование постоянного тока в переменный, в пневматических и гидравлических усилителях преобразование перемещения в изменение давления).

В зависимости от вида энергии, получаемой усилителями, последние делятся на электрические, гидравлические, пневматические, электрогидравлические, электропневматические.

Наибольшее распространение получили электрические усилители, имеющие высокую чувствительность, большой коэффициент усиления и являющиеся наиболее удобными при эксплуатации.

Исполнительные устройства относят к элементам автоматики, создающим управляющие воздействия на объект управления. Они изменяют состояние или положение регулирующего органа объекта таким образом, чтобы регулируемый параметр соответствовал заданному значению. К исполнительным устройствам, создающим управляющее воздействие в виде силы или вращаю-

шего момента, относят силовые электромагниты, электромагнитные муфты, двигатели.

Двигатели в зависимости от вида применяемой для работы энергии могут быть электрическими, гидравлическими или пневматическими.

В качестве исполнительных устройств, изменяющих состояние регулирующего органа, можно использовать усилители или реле.

Реле — это элементы автоматики, у которых изменение выходного сигнала $x_{\text{вых}}$ происходит дискретно при достижении входным сигналом $x_{\text{вх}}$ определенного значения, вызывающего срабатывание реле. Это значение входного сигнала называют *уровнем срабатывания реле*.

Мощность входного сигнала, вызывающего срабатывание реле, значительно меньше мощности, которой реле может управлять. Поэтому реле применяют как усилительный и как исполнительный элемент.

Реле часто используют и как автоматически управляемые коммутаторы сигналов в многоканальных системах сбора и передачи данных, в которых обрабатывается информация от десятков, сотен и даже тысяч датчиков. Они применяются также в системах контроля, сигнализации, блокировки и защиты.

Вычислительные элементы в устройствах автоматического управления осуществляют математические преобразования с поступающими на их вход сигналами. Эти операции осуществляются с целью обеспечить заданный алгоритм работы системы.

В простейшем случае вычислительные элементы выполняют отдельные математические операции, такие как алгебраическое суммирование, дифференцирование, интегрирование, логическое сложение, логическое умножение и т. д.

В замкнутых САУ необходимо осуществлять суммирование сигнала датчика и сигнала обратной связи. В корректирующих устройствах используют дифференцирование и интегрирование сигналов. Для выполнения этих операций главным образом применяют вычислительные элементы аналогового типа.

В более сложных случаях в качестве вычислительного элемента можно использовать микропроцессор, специализированные и унифицированные ЭВМ цифрового и аналогового типов или комплекс этих машин. Такие задачи автоматического управ-

ления, как оптимизация, создание адаптивных (приспосабливающихся) САУ, использование алгоритмов управления, основанных на вероятностных и статистических методах обработки сигналов, невозможно осуществить без ЭВМ.

Согласующие и вспомогательные элементы включаются в устройство автоматического управления для улучшения его параметров, расширения функциональных возможностей основных элементов и т. д.

В качестве *согласующих элементов* часто используют трансформаторы, редукторы, позволяющие согласовывать параметры исполнительного элемента с параметрами объекта управления.

В САУ, в которых в качестве вычислительного элемента используют микропроцессор или ЭВМ, часто возникает необходимость согласования ЭВМ с датчиками информации и исполнительными элементами аналогового типа, широко применяемыми в автоматике. Для этой цели на входе ЭВМ устанавливают аналого-цифровые преобразователи (АЦП). Они преобразуют механический сигнал (перемещения, скорости и т. д.) или электрический сигнал (напряжения, силы тока, сопротивления и т. д.), получаемый от аналоговых датчиков, в дискретный кодовый сигнал, способный восприниматься ЭВМ. В таких системах управляющее воздействие получают в дискретной форме как результат обработки в ЭВМ поступившей информации.

Если в устройстве автоматического управления в качестве исполнительного элемента используют электродвигатели постоянного или переменного тока, электромагнитные муфты, усилители мощности постоянного или переменного тока и другие устройства, то возникает потребность обратного преобразования дискретного сигнала ЭВМ в аналоговый сигнал, воспринимаемый исполнительным элементом.

Эта задача решается с помощью цифроаналоговых преобразователей (ЦАП). Они преобразуют кодовый сигнал, полученный от ЭВМ, в перемещение, напряжение, ток, частоту и т. д.

Вспомогательные элементы автоматики — это стабилизаторы напряжения или тока, коммутаторы и распределители, генераторы напряжения специальной формы, формирователи импульсов, индикаторные и регистрирующие приборы, сигнальные и защитные устройства.

Эти элементы автоматики, не являясь принципиально необходимыми для работы устройства автоматического управления,

в то же время позволяют увеличить точность и стабильность его работы, облегчают наладку и эксплуатацию, расширяют возможности использования этого устройства при создании САУ.

1.5. Принципы регулирования

По принципу регулирования САУ подразделяют на три вида: системы регулирования по отклонению и возмущению, комбинированные системы.

В системах регулирования по отклонению регулирующее воздействие вводится при отклонении регулируемого параметра от заданного значения независимо от вида возмущения и места их возникновения. В этом отношении такие системы универсальны, однако они имеют недостаток: действие системы начинается лишь при наличии отклонения регулируемого параметра, следовательно, обеспечить точное поддержание его на заданном значении невозможно. Системы, работающие по данному принципу, являются замкнутыми системами автоматического регулирования с отрицательной обратной связью.

Системы регулирования по возмущению основаны на том, что регулирующее воздействие регулятора направлено на компенсацию возмущения по месту его возникновения на входе объекта, обеспечивая таким образом стабилизацию его входных параметров.

Поскольку в большинстве случаев в системах автоматического регулирования по возмущению происходит изменение нескольких параметров, возникает необходимость установки соответствующего числа регулирующих автоматических устройств, что усложняет и удорожает систему. Кроме того, автоматические системы регулирования по возмущению не компенсируют возможные внутренние возмущения, которые могут вызвать отклонение параметров на выходе из объекта, что является существенным недостатком САУ по возмущению. Автоматические системы регулирования по возмущению являются разомкнутыми системами.

Комбинированные системы регулирования сочетают принципы работы двух рассмотренных систем. В таких системах значительные внешние возмущения компенсируются регуляторами, действующими по принципу регулирования по возмущению, а влияние всех прочих возможных возмущений устраняется регулятором, действующим по принципу регулирования по отклонению.

Достоинства комбинированных САР – стабилизация внешних условий функционирования объекта и точность поддержания заданных значений параметров процесса, недостаток – большое число автоматических устройств, что усложняет эксплуатацию систем и ведет к их удорожанию.

✍ Контрольные вопросы и задания

1. Какие свойства присущи любой системе?
2. Дайте определение понятия «системы автоматического управления».
3. Опишите функциональную схему системы автоматического управления.
4. Приведите классификацию системы автоматического управления по назначению.
5. Раскройте назначение элементов системы автоматического регулирования.
6. Как классифицируют системы автоматического регулирования по принципу регулирования?
7. Перечислите основные элементы систем автоматического управления.

2. ДАТЧИКИ

2.1. Общие сведения

В технологических процессах управление осуществляется по таким параметрам, как время, энергия, температура, сила (давление), масса, скорость (ускорение), электропроводность, освещенность, цвет и т. д. Многие остальные параметры технологических процессов могут быть рассчитаны, смоделированы из вышеперечисленных.

Датчик представляет собой конструктивно законченное устройство, предназначенное для преобразования физической величины в электрическую или иную величину, удобную для дальнейшей передачи и преобразования.

Большинство известных в автоматике датчиков преобразуют контролируемую величину X в электрическую Y (например, движение в индуктивность или напряжение, температуру в электрическое напряжение и т. п.) или в неэлектрическую (движение в давление воздуха или жидкости и др.) величину.

Измерения производят путем внедрения датчика b в среду или приема излучений (электромагнитных, акустических и др.) из нее (рис. 2.1).

Основу датчика составляет чувствительный элемент 2 , непосредственно преобразующий измеряемую величину среды 1 в электрический сигнал. В состав датчика также могут входить устройства 3 , изменяющие форму или корректирующие сигнал чувствительного элемента, например преобразователь аналогового сигнала в цифровой или корректор нелинейности. Сигнал из датчика подают по проводам или по беспроводной системе 4 на приемник 5 для его обработки и дальнейшего использования.

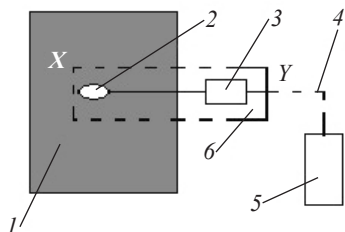


Рис. 2.1. Датчик и измеряемая среда:

1 – измеряемая величина среды; 2 – чувствительный элемент;
3 – преобразователь; 4 – линия связи; 5 – приемник; 6 – датчик

Используемые датчики весьма разнообразны, их можно классифицировать по различным признакам:

- в зависимости от вида входной (измеряемой) величины различают датчики механических перемещений (линейных и угловых), пневматические, электрические, расходомеры, датчики скорости, ускорения, усилия, температуры, давления и др.;

- по виду выходной величины, в которую преобразуется входная величина, различают неэлектрические и электрические: датчики постоянного тока (ЭДС или напряжения), датчики амплитуды переменного тока (ЭДС или напряжения), датчики частоты переменного тока (ЭДС или напряжения), датчики сопротивления (активного, индуктивного или емкостного) и др.;

- по принципу действия датчики разделяют на два класса: генераторные и параметрические (датчики-модуляторы). Генераторные датчики осуществляют непосредственное преобразование входной величины в электрический сигнал;

- по принципу действия датчики также можно разделить на омические, реостатные, фотоэлектрические (оптико-электронные), индуктивные, емкостные и др.

2.2. Потенциометрические датчики

В датчиках, чувствительным элементом в которых является потенциометр – переменный резистор, подвижной контакт механически связан с устройством, параметр которого необходимо измерить.

Потенциометрические устройства широко используют для преобразования линейного или углового перемещения (входная величина) в электрический сигнал (выходная величина).

Конструктивно потенциометрическое измерительное устройство (рис. 2.2) состоит из потенциометра 1 с движком 2 , к которому прикреплена жесткая механическая связь 3 с перемещаемым объектом 4 . Потенциометр может быть выполнен линейным или круговым. В первом случае измеряются линейные перемещения, во втором – угловые. Обмотка резистора выполняется из металлической проволоки (константана, манганина, изабелина, никелина), навитой на каркас, или металлических пленок. На вход потенциометра подается напряжение U_1 .

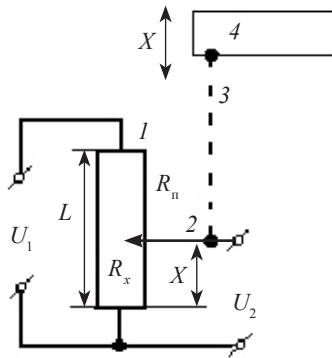


Рис. 2.2. Потенциометрический чувствительный элемент:
 1 – потенциометр; 2 – движок потенциометра; 3 – жесткая связь; 4 – перемещаемый объект; L – длина перемещения движка; X – перемещение объекта; R_n – полное сопротивление потенциометра; R_x – измеряемое сопротивление ползуна; U_1 – входное напряжение; U_2 – выходное напряжение датчика

Объект, перемещение X которого необходимо измерить, через механическую связь перемещает движок потенциометра на то же расстояние X или кратное ему либо на определенный угол в зависимости от конструкции связи.

Напряжение на выходе U_2 определяют по формуле

$$U_2 = U_1 R_x / R_n = (U_1 X) / L = kX,$$

где $k = U_1 / L$ – постоянная датчика.

Потенциометрические датчики просты по конструкции, дешевы, однако в связи с применением движущихся механических частей имеют значительную погрешность и низкую надежность.

2.3. Тензометрические датчики

Тензометрические датчики применяют для контроля за деформациями и напряжениями при статических и динамических нагрузках, а также для измерения крутящих и изгибающих моментов, возникающих на поверхности механизмов при их механической нагрузке.

Принцип действия тензометрических датчиков (тензорезистивных преобразователей) основан на явлении тензоэффекта, т. е. на изменении величины активного сопротивления проводниковых и полупроводниковых материалов под воздействием приложенных к ним механических напряжений (усилий).

В простейшем случае тензорезистор представляет собой тонкую проволоку, внедренную в бумагу, наклеиваемую на металлическую балку, которая подвергается механической деформации (растяжение, изгиб, скручивание, сжатие).

Полупроводниковый тензорезистор (рис. 2.3) представляет собой кристалл кремния (n -проводимость) с нанесенной на его поверхность дорожкой другого полупроводника с p -проводимостью – тензорезиста. Сопротивление дорожки тензорезиста

$$R = \rho l / s,$$

где ρ – удельное сопротивление материала тензорезистора; l – длина дорожки тензорезиста; s – площадь поперечного сечения тензорезиста.

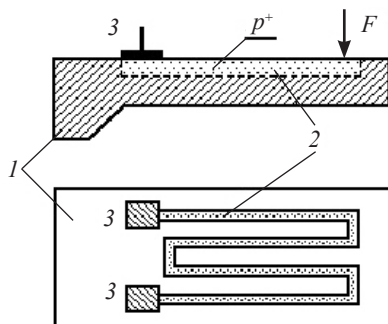


Рис. 2.3. Полупроводниковый тензорезистор:
1 – полупроводниковый кристалл; 2 – дорожка тензорезиста с p -проводимостью; 3 – электроды

При деформации дорожки под действием силы F (например, при ее сжатии) изменяются эффективная длина l (увеличивается) и площадь поперечного сечения s (уменьшается). В результате изменяется сопротивление подложки R (увеличивается), а значит, и всего p – n -перехода тензорезистора. Это изменение сопротивления фиксируется электронной схемой, на которой отображаются измерения силы F .

Тензорезистор крепится к детали, которая воспринимает какие-либо усилия F . Чувствительность тензорезисторных преобразователей определяется тензорезисторным коэффициентом: отношением изменения сопротивления активной части тензорезистора к изменению приложенной силы. Постоянная датчика у большинства проволочных тензорезистов $K \approx 2$, а у полупроводниковых $K \approx 100$.

Тензорезисторы применяют для измерения сил, деформаций и малых перемещений.

В зависимости от материала чувствительного элемента тензорезисторы подразделяют на проволочные, фольговые, полупроводниковые.

К достоинствам тензометрических датчиков можно отнести незначительную массу, малые размеры, простоту конструкции, возможность измерения статических и динамических процессов; к недостаткам — относительно невысокую чувствительность, возможность только разового использования (так как он разрушается при отсоединении от детали), необходимость использования мостовой измерительной схемы и компенсации температурных воздействий.

2.4. Индуктивные датчики

Широкое применение индуктивные датчики находят в промышленности для измерения перемещений в диапазоне от 1 мкм до 20 мм, а также для измерения давлений, сил, уровней расхода газа и жидкости. Измеряемый параметр с помощью различных чувствительных элементов преобразуется в изменение перемещения и затем эта величина подводится к индуктивному измерительному преобразователю.

Принцип действия индуктивных чувствительных элементов датчиков основан на изменении тока в обмотке индуктивной ка-

тушки или трансформатора в зависимости от магнитного сопротивления сердечника.

Индуктивный чувствительный элемент (рис. 2.4) состоит из обмотки катушки, намотанной на неподвижный сердечник, продолжением которого является подвижный сердечник.

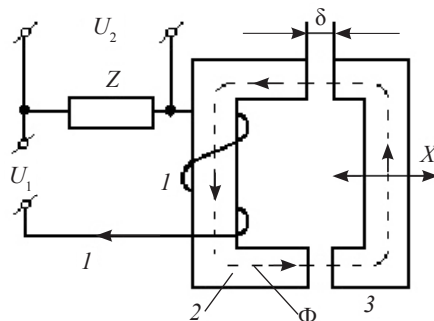


Рис. 2.4. Индуктивный чувствительный элемент:
 1 – обмотка катушки; 2 – неподвижный сердечник;
 3 – подвижный сердечник; U_1 – входное переменное напряжение;
 U_2 – выходной сигнал; Z – сопротивление нагрузки

Между неподвижным и подвижным сердечниками имеется зазор δ , который зависит от положения подвижного сердечника, связанного с объектом, для которого необходимо измерить перемещение X . На обмотку катушки подается переменное синусоидальное напряжение U_1 с частотой f . В сердечнике с магнитным сопротивлением R_m циркулирует магнитный поток Φ .

В цепи обмотки протекает ток, величина которого зависит от зазора δ :

$$I = \delta \Phi (2nf / \mu_0 S) = k \delta \Phi,$$

где $k = 2nf / \mu_0 S$ – постоянная датчика; μ_0 – магнитная постоянная; S – площадь сечения сердечника.

Выходной сигнал

$$U_2 = ZI = k \delta \Phi.$$

Поскольку магнитный поток также зависит от зазора δ , то зависимость тока от зазора является нелинейной, особенно в начале и в конце зависимости $I = f(\delta)$.

Для повышения чувствительности применяют дифференциальное включение двух обмоток в мост. Широко используется

трансформаторная схема включения, когда одна из обмоток питается внешним напряжением, а в другой наводится напряжение в зависимости от магнитного сопротивления, т. е. от зазора между сердечниками. Индуктивные датчики широко применяют для измерения малых перемещений и вибраций (0,1–0,5 мкм).

Достоинства индуктивных датчиков: простота и прочность конструкции, отсутствие скользящих контактов; возможность подключения к источникам промышленной частоты; относительно большая выходная мощность (до десятков ватт); значительная чувствительность.

Недостатки индуктивных датчиков: точность работы зависит от стабильности питающего напряжения по частоте; возможна работа только на переменном токе.

2.5. Емкостные датчики

Емкостные датчики используются в системах регулирования и управления производственными процессами почти во всех отраслях промышленности. Они применяются для контроля заполнения резервуаров жидким, порошкообразным или зернистым веществом, как конечные выключатели на автоматизированных линиях, конвейерах, роботах, обрабатывающих центрах, станках, в системах сигнализации, для позиционирования различных механизмов и т. д.

Емкостные датчики (рис. 2.5) представляют собой конденсаторы с изменяющимся зазором d между обкладками, площадью пластин S или диэлектрической проницаемостью ϵ . Изменение входной величины вызывает изменение емкости конденсатора:

$$C = \epsilon_0 \epsilon S / (d - \Delta d),$$

где ϵ_0 — абсолютная диэлектрическая проницаемость вакуума; ϵ — диэлектрическая проницаемость среды между обкладками; S — площадь обкладок измерительных емкостей; d — начальный зазор; Δd — изменение зазора.

Емкостные датчики с переменным зазором между пластинами (рис. 2.5, *а*) служат для измерения линейных перемещений x с точностью до 0,1–0,01 мкм; с переменной площадью (рис. 2.5, *б*) — для измерения линейных и угловых перемещений; с изменением диэлектрической проницаемости среды (рис. 2.5, *в*) — для измерения уровней, влажности, температуры, химического состава и др.

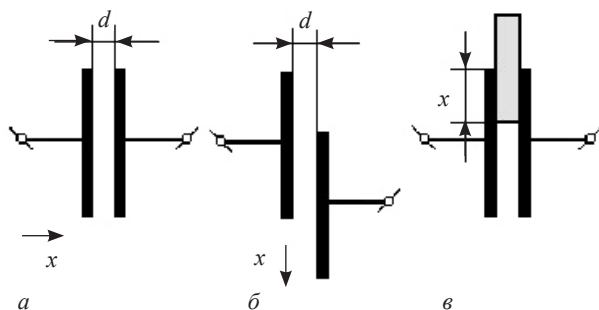


Рис. 2.5. Емкостные датчики:

- а* – с переменным зазором между пластинами;
б – с переменной площадью пластин; *в* – с изменением диэлектрической проницаемости среды

К достоинствам емкостных датчиков относятся простота изготовления, использование недорогих материалов для производства; малые габариты и масса; низкое потребление энергии; высокая чувствительность; отсутствие контактов (в некоторых случаях – один токосъем); долгий срок эксплуатации; потребность весьма малых усилий для перемещения подвижной части емкостного датчика; простота приспособления формы датчика к различным задачам и конструкциям.

К недостаткам емкостных датчиков следует отнести сравнительно небольшой коэффициент передачи (преобразования); высокие требования к экранировке деталей; необходимость работы на повышенной (по сравнению с 50 Гц) частоте.

2.6. Пьезоэлектрические датчики

В данного типа датчиках используется пьезоэлектрический эффект, сущность которого состоит в том, что под действием приложенного усилия на гранях некоторых кристаллов (кварца, титана, бария, турмалина, сегнетовой соли и др.) появляются электрические заряды (прямой пьезоэффект), а при внесении пьезоэлемента в электрическое поле он деформируется (обратный пьезоэффект).

При действии силы F на сжатие или растяжение на гранях пластины возникают разнополярные электрические заряды q (рис. 2.6).

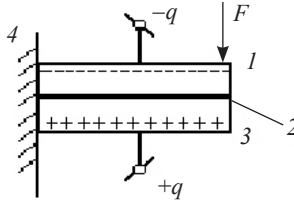


Рис. 2.6. Возникновение пьезоэффекта в кристалле кварца:
1, 3 – пластины кварца; 2 – электропроводящая фольга; 4 – консоль крепления пластин

В общем виде плотность зарядов δ зависит от механического напряжения кристалла σ :

$$\delta = d\sigma,$$

где d – величина, постоянная для данного типа материала и конструкции чувствительного элемента.

Если к кристаллу приложить внешнее электрическое поле E (подвести электрическое напряжение), он начнет деформироваться – изменять свои геометрические размеры, а значит, и электрическую емкость. Переменное внешнее электрическое поле приводит к возникновению в кристалле механических колебаний, частота которых зависит от его величины и конструкции. Такие пьезодатчики называются кварцевыми резонаторами, в них основная частота механических колебаний зависит от собственной частоты кристалла:

$$f \equiv \Delta d \sqrt{\frac{E_i}{p}},$$

где Δd – расстояние между краями пластины; E_i – модуль упругости пластины; p – плотность.

Если включить кварцевый резонатор Π_3 по схеме автогенератора (рис. 2.7), то на выходе операционного усилителя ОУ возникнет сигнал с частотой f , зависящей от приложенного внешнего воздействия – усилия, температуры, вибрации и т. п.

Резонатор Π_3 с кварцевой пластиной включен на инвертирующий вход операционного усилителя ОУ. Ток насыщения конденсатора создаст падение напряжения на инвертирующем входе ОУ, которое инвертируется и изменяет знак питания цепи $\Pi_3 - R$, в результате чего возникают колебания, частота которых

определяется ее постоянной времени. Частота колебаний генератора совпадает с собственной частотой кварца и возникают резонансные автоколебания.

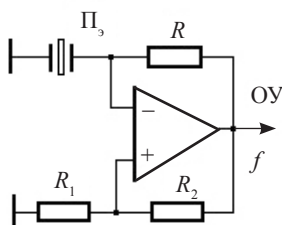


Рис. 2.7. Кварцевый автогенератор

Чувствительные элементы пьезорезонансных датчиков выполняют в виде резонатора той или иной формы — прямоугольной, квадратной, круглой и т. п. Пьезоэлектрические датчики используют при измерении усилий, давлений, вибраций и других физических величин.

Преимущества пьезоэлектрических датчиков: материал изготовления (кристаллокерамика); отличная эффективность при высоких и низких мощностях; простота обращения и эксплуатации; минимальная потеря энергии во время работы.

Недостатки: короткий срок эксплуатации; высокая стоимость; требует тщательного ухода и техобслуживания; толчки, падение, сильная вибрация могут повредить такой датчик, а высокие колебания температур — кристаллы; пьезоэлектрический наконечник нельзя подвергать воздействию ультразвука, который его непоправимо испортит.

2.7. Фотоэлектрические датчики

Фотоэлектрические чувствительные элементы преобразуют оптическое излучение в электрический сигнал. По принципу действия существует несколько видов преобразователей: с внешним фотоэффектом (вакуумные или газонаполненные); с внутренним фотоэффектом (фоторезисторы) и на основе $p-n$ -перехода (фотодиоды, фототранзисторы и т. п.).

Основными характеристиками фотоэлементов являются зависимость параметра (сопротивления, тока, напряжения) от облученности входного торца чувствительного элемента; спектраль-

ная — зависимость чувствительности от длины волны падающего излучения; частотная — зависимость чувствительности от частоты изменения излучения.

Фотоэлемент с внешним фотоэффектом представляет собой вакуумную или газонаполненную лампу с анодом и катодом в виде внутренней стенки с нанесенным на нее фоточувствительным слоем. Под действием светового потока в катод возникают свободные электроны, которые под действием электрического поля перемещаются к аноду, создавая внутри фотоэлемента электрический ток. В настоящее время подобные элементы применяются редко.

Фотоэлемент с внутренним фотоэффектом представляет собой фоторезистор (рис. 2.8), принцип действия которого состоит в том, что свободные электроны, образующиеся под действием излучения в слое чувствительного проводника (фоторезиста) 3, остаются (перераспределяются) в веществе, резко изменяя его сопротивление R . Чувствительный материал наносится на изоляционную подложку 1 и сверху покрыт защитной тонкой прозрачной лаковой пленкой 2. Наиболее часто применяют сернисто-кадмиевые, сернисто-свинцовые, сернисто-висмутовые и селенисто-кадмиевые фоторезисторы.

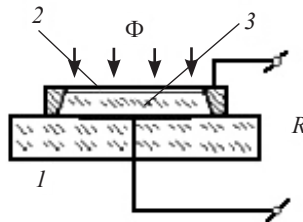


Рис. 2.8. Устройство фоторезистора:

1 — подложка; 2 — защитная пленка; 3 — фоторезист;
 R — сопротивление фоторезистора; Φ — оптическое излучение

Фотоэлементы с вентильным фотоэффектом работают на использовании явлений, происходящих в p - n -переходе под воздействием излучения. Они состоят из полупроводниковых слоев 1 и 3, являющихся p - n -переходом, металлического основания 2, выполняющего роль нижнего электрода, оптической линзы 4 (рис. 2.9).

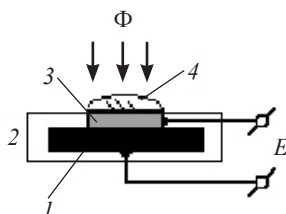


Рис. 2.9. Конструкция фотодиода:
 1 – материал n -типа; 2 – корпус; 3 – материал p -типа;
 4 – оптическая линза

При облучении p - n -переход сам генерирует электрическое напряжение. Как правило, напряжение E , возникающее на переходе, имеет положительную полярность со стороны области n . Это напряжение способно обеспечить протекание тока во внешние цепи. Чем выше интенсивность светового потока, тем больше ток I от него. Схемы включения фотодиода в режиме генератора приведены на рисунке 2.10.

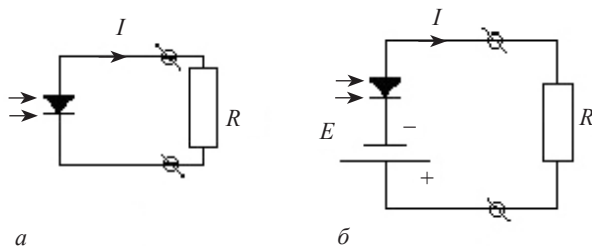


Рис. 2.10. Схемы включения фотодиода в генераторном режиме (а) и в обратном направлении (б)

При построении датчиков часто используют излучающие полупроводниковые диоды – *светодиоды*. Если приложить внешнее напряжение в прямом направлении диода, это приведет к протеканию в нем электрического тока, который вызывает свечение p - n -перехода. Длина волны излучаемых колебаний определяется используемым полупроводниковым материалом p - n -перехода. Интенсивность излучения возрастает при увеличении тока.

Чтобы обеспечить требуемое пространственное распределение света, светодиод применяют совместно с оптическими линзами. Для изготовления светодиодов применяют арсенид галлия, дающий излучение в инфракрасной области оптического излучения,

фосфид арсенида галлия, излучающий красный или оранжевый свет, и фосфид галлия (желтый или зеленый). Глаз человека обладает максимальной чувствительностью на зеленом участке спектра, поэтому фосфид галлия имеет наибольшую эффективность.

В датчиках используют диодные оптические пары — *оптроны*. Они состоят из фотоизлучающего (светодиод) D_1 и фоточувствительного D_2 (фотодиод, фоторезистор) приборов, сконструированных в едином устройстве (рис. 2.11). Такие устройства применяют для передачи сигнала с одной электрической схемы на другую в тех случаях, когда требуется, чтобы они были электрически изолированы одна от другой.

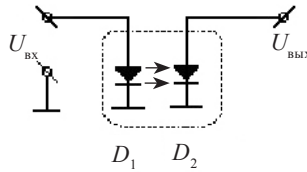


Рис. 2.11. Схема устройства оптрона

В настоящее время распространение получили твердотельные фотоэлектронные приборы с зарядовой связью (ПЗС). Основой ПЗС является конденсатор МОП-структуры (металл–оксид–полупроводник). Одной обкладкой конденсатора является металлический электрод, а другой — полупроводник. Между обкладками находится слой оксида полупроводника (рис. 2.12).

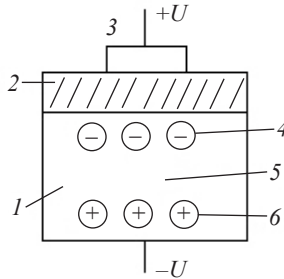


Рис. 2.12. Ячейка конденсатора прибора с зарядовой связью (МОП-структура):

- 1 — p -полупроводниковая подложка; 2 — оксид полупроводника;
3 — металлический электрод; 4 — неосновные носители;
5 — обедненная область; 6 — основные носители

Если между электродом и полупроводниковой подложкой (например, p -типа) приложить напряжение $+U$, в результате действия электрического поля дырки (основные носители), находящиеся под электродом, отойдут в толщу слоя полупроводника. При этом под электродом образуется область, обедненная основными носителями, — *потенциальная яма*. Изменяя величину напряжения $+U$, можно управлять глубиной потенциальной ямы.

В образовавшейся потенциальной яме происходит накопление неосновных носителей заряда — электронов, которые образуются за счет фотоэмиссии в результате засветки МОП-структуры.

2.8. Датчики температуры

Температура — один из основных параметров технологических процессов. Температура среды оказывает влияние на параметры многих материалов: на размеры, электрическое сопротивление, термоэлектродвижущую силу и т. д. В качестве чувствительных элементов датчиков температуры применяют тепломеханические элементы, термопары, терморезисторы, p - n -переходы, кристаллы кварца, кремния и их соединений и многие другие материалы и явления в них.

Тепломеханические датчики (термобиметаллические, dilatометрические) используют в качестве чувствительных элементов, преобразующих изменение фактического значения регулируемой температуры в перемещение.

Биметаллический чувствительный элемент (рис. 2.13, *a*) представляет собой две узкие металлические пластинки 1 и 3 с различными коэффициентами температурного расширения, жестко скрепленные между собой по всей плоскости касания (спаянные).

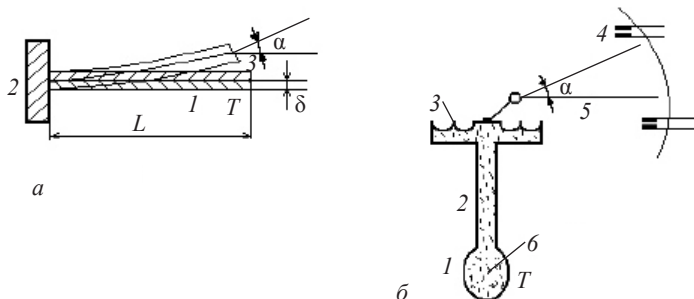


Рис. 2.13. Тепломеханические чувствительные элементы: *a* — биметаллический; *б* — dilatометрический

Один конец биметаллической пластинки неподвижно закреплен в основании, а второй — свободен. При длине L плоской пластинки l , значительно превышающей ее толщину δ , угол перемещения свободного конца α будет зависеть от температуры окружающей среды T . На пути изгиба биметаллической пластинки можно поставить, например, электрические контакты, которые будут включать электрическую сеть.

Подобные чувствительные элементы используют для двухпозиционного регулирования (включено-выключено) температуры.

На рисунке 2.13, б показан dilatометрический преобразователь с расширяющейся жидкостью или газом. Изменение температуры T металлического баллона l связано с изменением давления находящихся в нем жидкости или газа b , так как коэффициенты объемного расширения жидкостей и газов гораздо выше, чем металлов. При изменении давления перемещается свободный конец мембраны 3 , связанный с баллоном через капилляр 2 . Указатель 5 , связанный с мембраной, может включать электрические контакты 4 , установленные под определенным углом α , соответствующим температуре T .

В качестве термометрических жидкостей используют амилловый спирт, ацетон, ртуть и газовые наполнители — азот, гелий и др. Термометаллические и dilatометрические чувствительные элементы применяют для измерения температур в диапазоне от -60 до $+450$ °С. Погрешность преобразования составляет 1–5 %.

Терморезисторы — это элементы, электрическое сопротивление которых зависит от температуры. Для их изготовления используют медь, платину и другие металлы, удельное сопротивление ρ которых увеличивается с повышением температуры T по закону

$$\rho = \rho_0(1 + \alpha T),$$

где ρ_0 — удельное сопротивление металла при 0 °С; α — температурный коэффициент сопротивления, Ом/К, показывающий, на сколько увеличивается удельное сопротивление терморезистора при повышении температуры на 1 °С.

Чувствительный элемент металлического терморезистора помещается в стальной или латунный корпус с клеммной головкой и представляет собой бифилярную обмотку из проволоки диаметром 0,04–0,08 мм, размещенную на изолированном корпусе,

к концам которой припаивают выводы из серебряной, а при температуре до $100\text{ }^{\circ}\text{C}$ – из медной проволоки.

Погрешность измерений металлических терморезисторов не превышает $0,5\text{--}1,0\%$. Из металлических терморезисторов наибольшее распространение получили термометры сопротивления типа ТСП (платиновые) и ТСМ (медные).

Термисторы – это полупроводниковые терморезисторы с отрицательным температурным коэффициентом сопротивления, а **позисторы** – с положительным.

Нелинейность характеристик ограничивает применение термисторов при измерениях температуры в узких пределах. Большой разброс по номинальному сопротивлению затрудняет их взаимозаменяемость.

Распространены медно-марганцевые (ММТ), кобальто-марганцевые (КМТ), косвенного подогрева (ТКП) и другие термисторы для температур от 203 до 523 К . Чувствительность термисторов в $5\text{--}30$ раз выше чувствительности металлических терморезисторов.

Зависимость термистора от температуры и схема его включения приведены на рисунке 2.14. В практике измерения температуры в рабочем диапазоне от -60 до $+120\text{ }^{\circ}\text{C}$ с погрешностью измерения $0,1\%$ получили распространение кремниевые $p\text{--}n$ -переходы, выполненные в виде смещенных в прямом направлении диодов или базоэмиттерных переходов транзисторов. Отрицательный температурный коэффициент падения напряжения на базоэмиттерном переходе кремниевого биполярного транзистора составляет $2,2 \cdot 10^{-3}\text{ В}/^{\circ}\text{C}$, а германиевого – $2,7\text{--}3,1 \cdot 10^{-3}\text{ В}/^{\circ}\text{C}$.

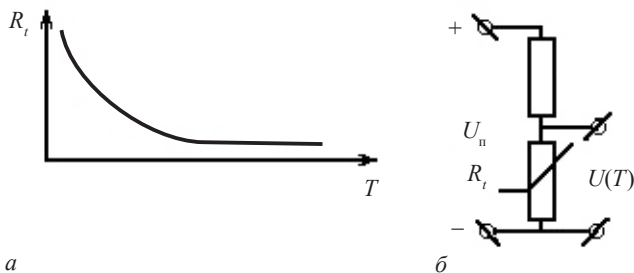


Рис. 2.14. Зависимость электрического сопротивления термистора R_t от температуры T (а) и схема его включения (б)

Терморезисторы и $p-n$ -переходы (диоды, транзисторы) используют для построения датчиков температуры, теплосчетчиков, термоанемометров и т. п.

Термопара – это термоэлектрический преобразователь (рис. 2.15). Состоит из двух 1 и 2 металлов, концы которых А и Б спаяны. Металлы обладают разными термоэлектрическими свойствами. Для изготовления термопар используют пары сплавов «хромель–копель» или «хромель–алюмель».

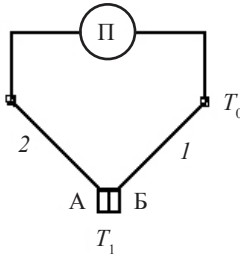


Рис. 2.15. Термопара

Спаянный конец называют рабочим спаем, он погружается в измеряемую среду (температура T_1), а свободные концы (холодный спай – температура T_0) подключают ко входу измерительной схемы П.

Если температуры рабочего спая и свободных концов различны, термопара вырабатывает термоэлектродвижущую силу. Она зависит от разности температур двух спаев термопары, и для снижения погрешности показаний необходимо знать температуру холодного спая, чтобы компенсировать эту разницу в дальнейших вычислениях. Подключение термопары к схеме измерения осуществляется проводами тех же металлов, что и термопары.

Термопары применяют в системах автоматического регулирования и контроля в диапазоне температур 373–3000 К. Чувствительность термопары обычно невелика и составляет для разных термопар 0,01–0,07 мВ/°С.

На рисунке 2.16 приведена схема цифрового датчика температуры и генератора импульсов на операционном усилителе. Эта схема позволяет получить выходной сигнал, частота которого F линейно зависит от температуры термочувствительного элемента R_t . Термочувствительный элемент R_t выполнен в виде термосопротивления.

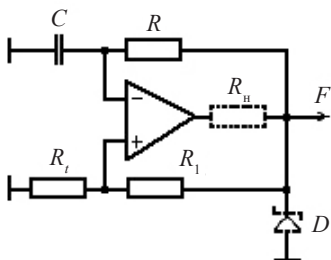


Рис. 2.16. Принципиальная схема включения терморезистора:
 R_n – сопротивление нагрузки; D – стабилитрон
(ограничитель напряжения односторонний)

Контрольные вопросы и задания

1. Дайте определение понятия «датчик».
2. Приведите классификацию датчиков.
3. Расскажите про общие характеристики датчиков.
4. В чем заключается работа потенциометрического датчика?
5. На чем основан принцип действия тензометрических датчиков?
6. Что собой представляют: а) емкостной датчик; б) фотоэлектрический датчик?
7. Объясните конструкцию и принцип действия датчиков температуры.

3. УСИЛИТЕЛЬНЫЕ УСТРОЙСТВА

3.1. Общие сведения

Усилитель — это устройство, в котором происходит усиление выходного сигнала датчика или преобразователя до величины, необходимой для питания исполнительных устройств за счет дополнительного источника энергии. В усилителях входная и выходная величины имеют одинаковую физическую природу. Усилители являются элементами датчиков, преобразователей, исполнительных устройств.

Классифицировать усилительные устройства можно по многим параметрам, в частности электронные усилители (ЭУ), которые будут рассмотрены, можно классифицировать следующим образом.

1. По *усиливаемому параметру (величине) электрического сигнала*: ЭУ могут быть напряжения, тока или мощности.

Любой усилитель (напряжения или тока) является усилителем мощности, которую можно получить на выходе как при высоком напряжении и малом токе, так и наоборот — большом токе и низком напряжении. Очевидно, что такая классификация усилителей определяется по наиболее выраженной отдаче в нагрузку соответственно напряжения, тока или мощности.

Следует также помнить, что повышать величину тока или напряжения можно и с помощью трансформатора, однако в отличие от усилителя трансформатор мощность не усиливает, а является лишь преобразователем напряжения или тока при неизменной мощности.

2. По *полосе и значению усиливаемых частот входного сигнала*: усилители постоянного тока (УПТ $\Delta f = 0-1$ МГц), усилители

низкой частоты (УНЧ $\Delta f = 0,1-1000$ Гц), высокой частоты (УВЧ $\Delta f = 0,1$ кГц–10 МГц), промежуточной частоты (УПЧ), а также узкополосные и широкополосные усилители (УПУ и ШПУ $\Delta f = 10$ Гц–10 МГц).

Поскольку полоса усиливаемых частот определяется видом нагрузки (резонансный контур или резистивная нагрузка), то в этой классификационной группе ЭУ различают резонансные и апериодические усилители.

3. По *характеру и виду усиливаемого сигнала*: ЭУ непрерывных сигналов (гармонических или квазигармонических, у которых сигнал медленно меняется во времени) и ЭУ импульсных сигналов, в которых импульсный сигнал прямоугольный, пилообразный и других форм меняется во времени настолько быстро, что форма выходного сигнала полностью определяется процессом установления колебаний в усилителе, т. е. переходным процессом.

4. По *виду используемого усилительного прибора*: ламповые, транзисторные, тиристорные, квантовые, парамагнитные (на сверхвысокочастотных вакуумных приборах – клистронах, магнетронах, лампах бегущей волны – ЛБВ) и др.

5. По *виду амплитудной характеристики* (зависимости выходного напряжения или тока от входной величины): линейные, логарифмические, усилители-ограничители и т. д.

Существуют и другие классификации внутри перечисленных классификационных групп, отражающих схемы включения усилителей и режимы их работы, например усилители с общей базой (ОБ), общим эмиттером (ОЭ), общим коллектором (ОК), функционирующие в режимах А, В и других режимах, а также усилители с обратной связью, распределенным усилением и т. д.

Для полной характеристики усилителя обычно используют совместные признаки, например транзисторный усилитель мощности низкой частоты на биполярных транзисторах (БПТ) по схеме с общим эмиттером.

Основные параметры электронных усилителей – усилительные, частотные, предельные, шумовые, параметры нелинейных искажений.

Усилительными параметрами являются коэффициенты усиления по напряжению:

$$K_U = \frac{U_{\text{ВЫХ}}}{U_{\text{ВХ}}};$$

току:

$$K_I = \frac{I_{\text{ВЫХ}}}{I_{\text{ВХ}}}$$

или мощности:

$$K_P = \frac{P_{\text{ВЫХ}}}{P_{\text{ВХ}}} = \frac{U_{\text{ВЫХ}} I_{\text{ВЫХ}}}{U_{\text{ВХ}} I_{\text{ВХ}}} = K_U K_I,$$

где $U_{\text{ВХ}}$, $U_{\text{ВЫХ}}$ – входное и выходное напряжение усилителя; $I_{\text{ВХ}}$, $I_{\text{ВЫХ}}$ – входной и выходной ток усилителя; $P_{\text{ВХ}}$, $P_{\text{ВЫХ}}$ – входная и выходная мощность усилителя.

Из *частотных параметров* одним из главных в усилителях является диапазон усиливаемых частот, задаваемый полосой пропускания усилителя, в пределах которой усиление падает от среднего (номинального) значения на допустимую величину (обычно в 1,7 раза), определяемую назначением усилителя. Этот параметр находят по *амплитудно-частотной характеристике (АЧХ)*.

Примеры АЧХ некоторых видов усилителей представлены на рисунке 3.1. Видно, что в области средних частот усиление не зависит от частоты. Форма АЧХ зависит от вида нагрузки, усиливаемых свойств усилителя и схемы усилителя.

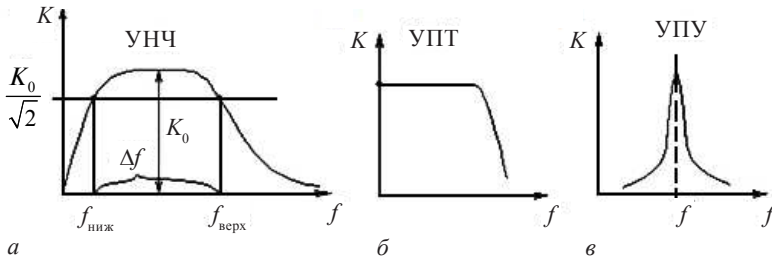


Рис. 3.1. Примеры амплитудно-частотных характеристик:
 а – усилителя низкой частоты; б – усилителя
 постоянного тока; в – узкополосного усилителя

По АЧХ определяют полосу пропускания частот усилителя Δf :

$$K = K_{\text{верх}} = \frac{K_0}{\sqrt{2}} = K_0 \cdot 0,707,$$

где K_0 – среднее значение коэффициента усиления.

Фазочастотная характеристика (ФЧХ) описывается зависимостью

$$\varphi = \Phi(f),$$

где φ — угол сдвига фаз между входным и выходным сигналами; f — основная частота усилителя.

Поскольку усилитель содержит реактивные элементы, то фаза сигнала на входе не совпадает с фазой сигнала на выходе, причем на разных частотах угол сдвига фаз будет разным.

Амплитудная характеристика — это зависимость величины выходного напряжения от входного при $f = \text{const}$ (рис. 3.2).

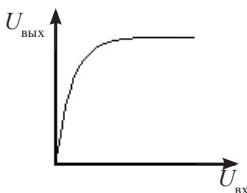


Рис. 3.2. Амплитудная характеристика

Предельные параметры: допустимые максимальные значения входного сигнала (при заданном коэффициенте усиления) и напряжения источника питания. Эти параметры зависят от типа УПТ и схемы его включения.

Шумовые параметры: коэффициент шума или уровень собственных шумов, создаваемый на выходе усилителя и мешающий усилению слабых сигналов. Шумы определяются физикой усилительных приборов.

Параметры нелинейных искажений в различных усилителях имеют разные наименования. В УНЧ для оценки нелинейных искажений обычно используют коэффициент гармонических составляющих, определяющий наличие в выходном сигнале частотных составляющих (гармоник), кратных частоте усиливаемого сигнала и искажающих форму полезного входного сигнала.

В УНЧ это воспринимается непосредственно ухом как искажение передаваемого звука, а в УВЧ нарушается достоверность слабой полезной информации, нуждающейся в усилении и качестве ее воспроизведения.

В УВЧ в качестве параметров нелинейных искажений часто используют коэффициенты (параметры) отдельных гармонических или

комбинационных составляющих (например, второго, третьего порядков и т. д.), имеющих наибольший уровень на выходе усилителя.

В современных УВЧ высокое качество электрических параметров достигается за счет максимальных значений коэффициента усиления и полосы усиливаемых частот при минимальных шумах и нелинейных искажениях, а также при минимальном потреблении энергии от источника питания. Часто усилители состоят из нескольких усилительных каскадов (УК) или просто каскадов, осуществляющих последовательное усиление сигнала. Их число зависит от требуемых коэффициентов усиления K_U , K_P , K_P .

Рассмотрим возможную структурную схему усилительного каскада (рис. 3.3).

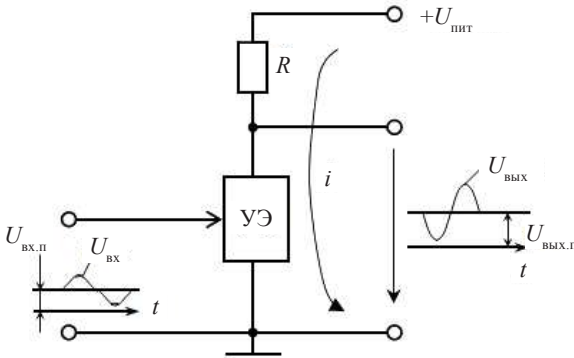


Рис. 3.3. Структурная схема усилительного каскада

Основными элементами здесь являются управляемый элемент (УЭ) и резистор R .

Будем считать УЭ линейным элементом с бесконечным сопротивлением в отсутствии входного напряжения.

Процесс усиления основывается на преобразовании энергии источника питания $+U_{\text{пит}}$ в энергию переменного напряжения в выходной цепи (*цепочка $+U_{\text{пит}}$ – резистор R – управляемый элемент*) за счет изменения сопротивления УЭ соответственно входному сигналу. Отметим, что усиление переменного сигнала обеспечивается при наличии некоторого постоянного напряжения на входе $U_{\text{вх.п}}$, на которое накладывается входной сигнал. Тогда при однополярном питании каскада выходное напряжение будет меняться относительно некоторого уровня постоянного напря-

жения $U_{\text{вых.п}}$. При отсутствии входного постоянного напряжения невозможно усилить переменный (двухполярный) сигнал. Очевидно, что усилительные свойства рассматриваемого каскада тем выше, чем больше сказывается изменение входного сигнала на изменение сопротивления УЭ и чем выше сопротивление R .

В качестве управляемого элемента можно взять транзистор. Показатели усилительного каскада будут зависеть от схемы включения транзистора – с общим эмиттером, с общим коллектором или с общей базой.

3.2. Транзисторные усилители

Усилительный каскад на биполярных транзисторах функционирует в схемах включения: с общей базой, общим эмиттером и общим коллектором (рис. 3.4).

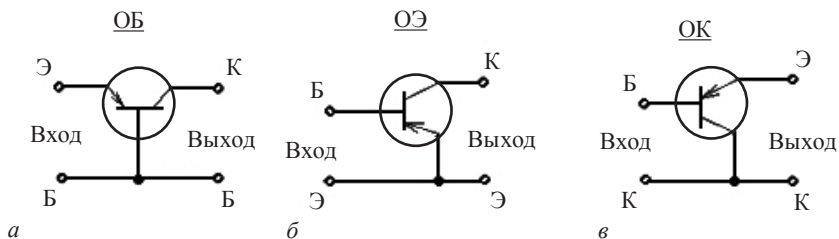


Рис. 3.4. Схемы включения биполярного транзистора:
 а – с общей базой; б – с общим эмиттером; в – с общим коллектором

Чаще применяется схема усилительного каскада с ОЭ, так как эта схема имеет наибольшее усиление.

Принцип работы усилителя на БПТ. Для пояснения принципа работы усилителя на БПТ воспользуемся типовой схемой усилительного каскада на БПТ с ОЭ (рис. 3.5), в которой отдельные источники базового напряжения (смещения $E_б$) и коллекторного питания $E_к$ не применяются, а используется делитель базового смещения на резисторах R_1 и R_2 , который совместно с эмиттерным резистором R_4 и коллекторным резистором R_3 обеспечивает выбор рабочей точки транзистора и ее термостабилизацию в процессе работы усилительного каскада.

Работу БПТ в типовой схеме с ОЭ (см. рис. 3.5) можно пояснить с помощью статических и динамических вольт-амперных характеристик (ВАХ) (рис. 3.6).

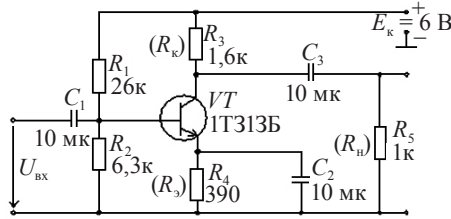
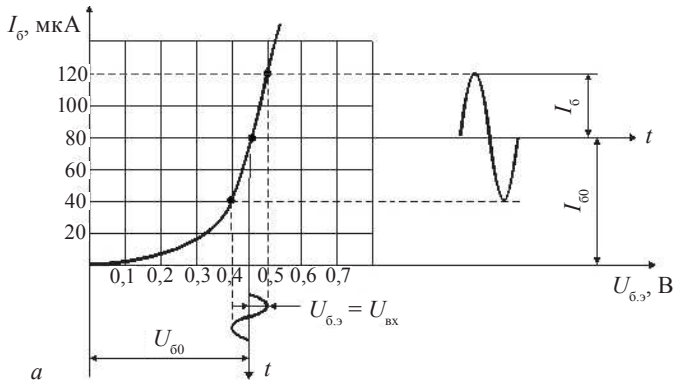
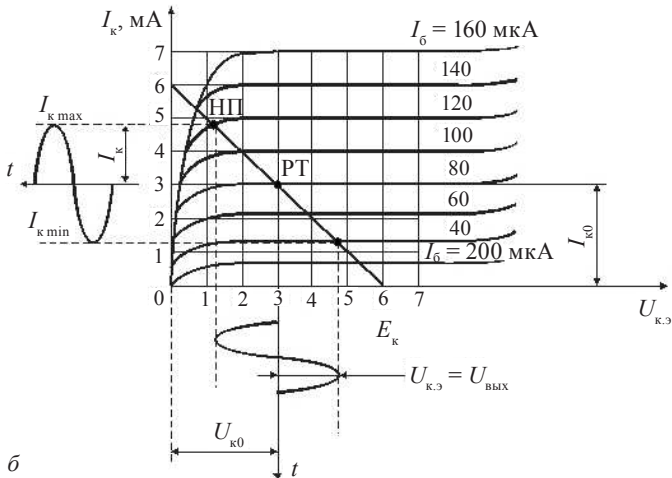


Рис. 3.5. Усилительный каскад на биполярном транзисторе в схеме с общим эмиттером



а



б

Рис. 3.6. Входная динамическая характеристика (а) и выходные статические характеристики усилительного каскада (б)

Для анализа схем на БПТ широко используют выходные $I_{\text{к}} = f(U_{\text{к.э}})$, входные $I_{\text{б}} = f(U_{\text{б.э}})$ и проходные $I_{\text{к}} = f(U_{\text{б.э}})$ статические и динамические характеристики. Независимыми из них являются входные и выходные, приводимые в справочниках.

В качестве входной динамической ВАХ обычно используют статическую при $U_{\text{к.э}} > 0$, например при $U_{\text{к.э}} = 5 \text{ В}$, что объясняется слабой зависимостью этих ВАХ от $U_{\text{к.э}}$ (см. рис. 3.6, а).

В качестве выходной динамической ВАХ в режиме малого переменного сигнала применяют нагрузочные прямые по постоянному току (см. рис. 3.6, б).

В соответствии с рисунком 3.6 процесс усиления можно представить следующей взаимосвязью электрических величин:

$$U_{\text{вх}} \uparrow \rightarrow I_{\text{б}} \uparrow \rightarrow I_{\text{к}} \uparrow \rightarrow I_{\text{к}} R_{\text{к}} \uparrow \rightarrow U_{\text{к.э}} = (E_{\text{к}} - I_{\text{к}} R_{\text{к}}) \downarrow \rightarrow U_{\text{вых}} \downarrow.$$

Видно, что входной сигнал с амплитудой $U_{\text{вх}} = U_{\text{б.э}}$ синфазно изменяет величину $I_{\text{к}}$, что влечет пропорциональное изменение $I_{\text{к}} R_{\text{к}}$ и противофазное изменение коллекторного напряжения $U_{\text{к.э}}$, причем амплитуда $U_{\text{к.э}}$ (с учетом масштаба по оси абсцисс) значительно больше амплитуды напряжения на базе $U_{\text{б.э}}$.

Рабочий режим и элементы схемы. В усилителе на БПТ линейный режим задается выбором рабочей точки (РТ) на середине нагрузочной прямой (НП) так, что равным отклонениям тока базы (см. рис. 3.6, б; отклонения тока базы $\Delta I_{\text{б}} = (80 \pm 40) \text{ мкА}$) соответствовали равные отрезки НП. При этом входной сигнал с амплитудами тока $I_{\text{б}} = I_{\text{вх}} = 40 \text{ мкА}$ и напряжения $U_{\text{вх}} = U_{\text{б.э}} = 0,05 \text{ В}$ вызывает изменения выходного сигнала с амплитудами тока $I_{\text{к}} = 1,8 \text{ мА}$ и напряжения $U_{\text{вх}} = U_{\text{к.э}} = 1,5 \text{ В}$.

Следует подчеркнуть, что одна из координат $I_{\text{к}} = E_{\text{к}} / R_{\text{б}}$ НП учитывает общее сопротивление эмиттерной и коллекторной цепей:

$$R_{\text{б}} = R_{\text{к}} + R_{\text{э}}.$$

В БПТ наблюдается значительный тепловой уход рабочей точки в процессе работы каскада. Поэтому термостабилизация РТ имеет существенное значение. Она создается протеканием тока отрицательной обратной связи (ООС) через эмиттерный резистор $R_{\text{э}}$ (см. рис. 3.5), который создает на базе напряжение ООС, компенсирующее тепловой уход РТ при повышении температуры $p-n$ -перехода БПТ. Действие ООС можно представить в следующей функциональной взаимосвязи электрических величин:

$$t^{\circ} \uparrow \rightarrow I_{\text{э}} \uparrow \rightarrow U_{\text{э}} = I_{\text{э}} R_{\text{э}} \uparrow \rightarrow U_{\text{б.э}} = (U_{\text{б}} - I_{\text{э}} R_{\text{э}}) \downarrow \rightarrow I_{\text{б}} \downarrow \rightarrow I_{\text{к}} \downarrow.$$

Величину R_3 обычно выбирают из соотношения

$$R_3 < U_{\text{к min}} / I_{\text{к0}},$$

где $U_{\text{к min}}$ — минимальная амплитуда коллекторного напряжения; $I_{\text{к0}}$ — постоянный ток коллектора в выбранной РТ.

Для определения резисторов R_1 и R_2 базового смещения обычно задается величина коэффициента стабилизации $K_{\text{ст}} = 3 \dots 20$, после чего рассчитываются основные величины элементов схемы, а также напряжений и токов в основных ее точках в соответствии с нижеприведенными соотношениями:

$$K_{\text{ст}} = 1 + \frac{\beta R_3}{R_3 + R_6}; R_6 = \frac{R_1 R_2}{R_1 + R_2}; U_6 = U_{6.э} + I_3 R_3 = \frac{E_{\text{к}} R_2}{R_1 + R_2};$$

$$R_1 = \frac{E_{\text{к}} - U_6}{I_{\text{д}} + I_6}; R_2 = \frac{U_6}{I_{\text{д}}},$$

где $\beta = h_{21э}$ — коэффициент передачи тока базы в схеме с ОЭ (в справочниках обычно берут среднее значение); U_6 — постоянное напряжение между базой и корпусом; $U_{6.э}$ — постоянное напряжение между базой и эмиттером; I_3 — ток эмиттера; $E_{\text{к}}$ — ЭДС источника коллекторного напряжения (см. рис. 3.6); $I_{\text{д}}$ — ток, протекающий через конденсатор.

Выбор конденсатора C_2 , шунтирующего эмиттерный резистор, производят по формуле

$$I_{\partial} = \frac{E_{\text{к}} - I_6 R_1}{R_1 + R_2};$$

$$U_{6.э} = U_6 - I_3 R_3 = I_{\text{д}} R_2 - I_3 R_3 = E_{\text{к}} - (I_{\text{д}} + I_6) R_1 - I_3 R_3.$$

Основные статические и динамические параметры усилительных каскадов. Существует несколько так называемых малосигнальных параметров БПТ. Наибольшее применение находит система h -параметров, вносящая минимальные погрешности при анализе в расчете усилительного каскада на БПТ. Для определения h -параметров используют уравнения эквивалентной схемы БПТ в виде h -четыреполюсника (рис. 3.7):

$$U_1 = h_{11} I_1 + h_{12} U_2;$$

$$I_2 = h_{21} I_1 + h_{22} U_2.$$

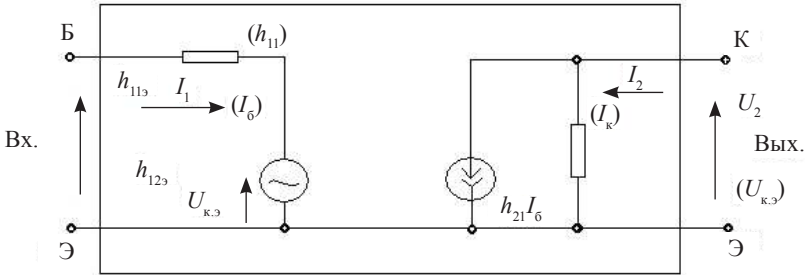


Рис. 3.7. h -четыреполюсник

В качестве независимых переменных здесь приняты одновременно ток на входе I_1 и напряжение на выходе U_2 , поэтому h -параметры имеют различные размерности, а сама система называется *смешанной (гибридной)*. Полагая, что $U_1 = U_{6,э}$, $I_1 = I_6$, $U_2 = U_{к,э}$ и $I_2 = I_к$, h -четыреполюсник представлен в виде эквивалентной схемы усилителя при включении БПТ с ОЭ, а ее h -параметры определяются по статическим характеристикам транзистора, которые выражаются следующим образом:

$$\begin{aligned} h_{11э} &= \Delta U_{6,э} / \Delta I_6 \text{ при } U_{к,э} = \text{const} (\Delta U_к = 0); \\ h_{12э} &= \Delta U_{6,э} / \Delta U_{к,э} \text{ при } U_6 = \text{const} (\Delta I_6 = 0); \\ h_{21э} &= \Delta I_к / \Delta I_6 \text{ при } U_{к,э} = \text{const} (\Delta U_{к,э} = 0); \\ h_{22э} &= \Delta I_к / \Delta U_{к,э} \text{ при } I_6 = \text{const} (\Delta I_6 = 0). \end{aligned}$$

Здесь h_{11} – входное сопротивление БПТ при коротком замыкании на выходе, Ом; h_{12} – коэффициент обратной передачи по напряжению при холостом ходе входа; h_{21} – коэффициент передачи тока при коротком замыкании выхода; h_{22} – выходная проводимость при холостом ходе входа, См.

Параметрами типа h чаще пользуются при анализе и расчете усилителей низкой частоты (УНЧ) или усилителей постоянного тока (УПТ) для определения динамических параметров каскада: входного $R_{вх}$, выходного $R_{вых}$ сопротивлений; коэффициентов усиления по току K_I , напряжению K_U или мощности K_P :

$$K_U = \frac{U_{\text{ВЫХ}}}{U_{\text{ВХ}}} = \frac{U_{к,э}}{U_{6,э}} = -\frac{h_{21} R_{\text{н.ЭКВ}}}{h_{11} + h R_{\text{н.ЭКВ}}} \approx -\frac{h_{21} R_{\text{н}}}{h_{11}},$$

где $R_{\text{н.ЭКВ}} = R_к R_{\text{н}} / (R_к + R_{\text{н}})$ – знак приближения соответствует практическим случаям, когда $R_{\text{н}} \ll R_{\text{ВЫХ}}$, т. е. $h_{22} R_{\text{н}} \ll 1$ ($K_U \gg 1000$ в УНЧ). Знак « \rightarrow » означает переворачивание фазы выходного сигнала;

$$K_I = \frac{I_{\text{ВЫХ}}}{I_{\text{ВХ}}} = \frac{I_K}{I_6} = \frac{h_{21}}{1 + h_{22}R_{\text{Н.ЭКВ}}} \approx h_{21} \quad (K_I \geq 30);$$

$$K_P = K_U K_I = K_U^2 R_{\text{ВХ}} / R_{\text{Н}} = \frac{K_I^2 R_{\text{Н}}}{R_{\text{ВХ}}} \approx h_{21}^2 R_{\text{Н}} / h_{11} \quad (K_P \geq 30);$$

$$R_{\text{ВХ}} = \frac{U_{\text{ВХ}}}{I_{\text{ВХ}}} = \frac{U_{\text{б.э}}}{I_6} = \frac{h_{11} + hR_{\text{Н.ЭКВ}}}{1 + h_{22}R_{\text{Н.ЭКВ}}} \approx h_{11} \quad (R_{\text{ВХ}} \leq 2 \text{ кОм}),$$

где $h = h_{11}h_{22} - h_{12}h_{21}$.

Из приведенных соотношений следует, что динамические параметры можно определять как графически, так и аналитически (по средним значениям h -параметров).

3.3. Усилители с обратными связями

Обратной связью называют воздействие выходного сигнала усилителя на его вход.

Существует *отрицательная* (ООС) и *положительная* (ПОС) *обратная связь*. При ООС напряжение обратной связи вводится на вход усилителя в противофазе с входным напряжением, а при ПОС – в фазе с входным напряжением.

Введем следующие величины и параметры, характеризующие усилитель с обратной связью (ОС):

$K = U_{\text{ВЫХ}} / U_{\text{ВХ}}$ – коэффициент усиления по напряжению усилителя без цепи ОС;

$\beta = U_{\text{В}} / U_{\text{ВЫХ}}$ – коэффициент обратной передачи напряжения или возвратное отношение;

$K\beta$ – петлевой коэффициент усиления (фактор ОС);

$\gamma = U_{\text{ВХОД}} / U_{\text{ВЫХ}}$ – глубина ОС;

U_{β} – часть выходного напряжения, вводимого цепью ОС на вход усилителя;

$U_{\text{ВХОД}}$ – входное напряжение усилителя с цепью ОС.

Усилитель с цепью ОС усиливает сигнал $U_{\text{ВХОД}}$, равный сумме входного $U_{\text{ВХ}}$ и добавленного цепью ОС U_{β} :

$$U_{\text{ВХОД}} = U_{\text{ВХ}} + U_{\beta} = U_{\text{ВХ}} + \beta U_{\text{ВЫХ}}.$$

Тогда коэффициент усиления усилителя с цепью ОС

$$K_{\text{ОС}} = \frac{U_{\text{ВЫХОД}}}{U_{\text{ВХ}}} = \frac{K}{1 - K\beta} \rightarrow \text{ПОС},$$

где $U_{\text{выход}} = KU_{\text{вход}} = K(U_{\text{вх}} + \beta U_{\text{выход}}) = KU_{\text{вх}} + K\beta U_{\text{выход}}$,

т. е. $U_{\text{выход}} = \frac{KU_{\text{вх}}}{1 - K\beta}$.

Выражение характеризует ПОС, при которой $K_{\text{ОС}} > K$. Напротив, при ООС $K_{\text{ОС}} < K$, а формула приобретает следующий вид:

$$K_{\text{ОС}} = \frac{K}{1 + K\beta} \rightarrow \text{ООС},$$

тогда $U_{\text{вхОС}} = U_{\text{вх}} - U_{\beta}$.

Заметим, что при $K\beta = 1$ ПОС приводит к самовозбуждению усилителя, превращая его в генератор колебаний, так как в этом случае $K_{\text{ОС}} = \infty$.

В зависимости от способа подачи сигнала обратной связи на вход усилителя применяют следующие ее виды:

- непосредственная параллельная и последовательная ПОС, превращая усилитель в двухполюсник;
- параллельная и последовательная ООС по напряжению;
- параллельная и последовательная ООС по току.

Можно полагать, что обе ООС по напряжению уменьшают выходное сопротивление усилителя, а ООС по току его увеличивает.

Что касается входного сопротивления, то обе параллельные ООС его уменьшают, а последовательные увеличивают:

$$\beta = \frac{R_{\Gamma} \parallel R_{\beta} \parallel R_{\text{вх}}}{R_{\text{ОС}} + R_{\Gamma} \parallel R_{\beta} \parallel R_{\text{вх}}},$$

где $R_{\Gamma} \parallel R_{\beta}$ и т. д. — параллельные соединения соответствующих резисторов.

Наибольшее применение в усилительной технике находит ООС (по сравнению с ПОС), действие которой сводится к следующему:

- уменьшается общий коэффициент усиления $K_{\text{ОС}}$ усилителя;
- повышается стабильность усиления при изменении температуры и режима транзистора;
- снижается уровень нелинейных искажений;
- изменяются входные и выходные сопротивления.

Ранее упоминалось, что электронные усилители подразделяются на усилители напряжения, тока и мощности, хотя все они

по существу являются усилителями мощности, а соответствующее название определяется по наибольшей выраженности в нагрузке той или иной электрической величины: напряжения, тока или мощности.

В курсе электронных усилителей показано, что максимальная отдача мощности в нагрузке достигается при условии согласования входной и выходной цепей усилителя с сопротивлениями соответственно источника сигнала (генератора R_{Γ}) и нагрузки (R_{Π}), т. е. равенству этих сопротивлений:

$$R_{\Gamma} \approx R_{\text{вх}}; R_{\text{вых}} \approx R_{\Pi}.$$

При этих условиях коэффициент по мощности усилителя определяется формулой

$$K_p = \frac{P_{\text{вых}}}{P_{\Gamma}} = \frac{E_{\text{вых}}^2}{E_{\Gamma}^2} \frac{R_{\Gamma}}{R_{\text{вых}}},$$

в которой входные и выходные параметры соответствуют функциональной схеме усилителя в общем виде (рис. 3.8).

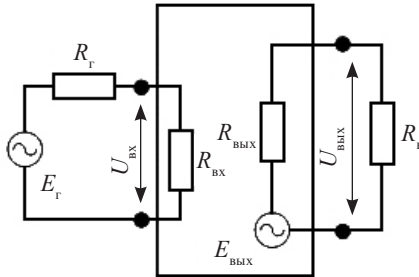


Рис. 3.8. Входные и выходные параметры усилителя

Однако в реальных случаях сопротивления генератора и нагрузки существенно отличаются от соответствующих входного и выходного сопротивлений усилителя, что не позволяет получить высокий КПД.

В инженерной практике существенное значение имеет КПД выходной цепи, равный отношению выходной мощности усиленного сигнала к потребляемой от источника питания мощности P_0 :

$$\text{КПД} = \frac{P}{P_0}.$$

При этом основной характеристикой усилителя считается выходная (отдаваемая в нагрузку) мощность:

$$P = P_{\text{вых}} = U_{\text{вых}}^2 + R_{\text{н}}.$$

В соответствии с классификацией усилителей напряжения и мощности усилителями мощности называют такие, у которых выходная мощность сигнала сравнима с мощностью, потребляемой от источника питания. Практически усилители с КПД более 50 % считаются усилителями мощности, а менее 50 % – усилителями напряжения.

3.4. Операционные усилители

Операционный усилитель (ОУ) представляет собой усилитель постоянного и переменного тока с большим коэффициентом усиления и глубокой отрицательной обратной связью.

Широкое распространение ОУ связано с их универсальностью (возможность построения на их основе различных электронных устройств, причем как аналоговых, так и импульсных), широким диапазоном частот (усиление сигналов постоянного и переменного тока), независимостью основных параметров от внешних дестабилизирующих факторов (изменение температуры, напряжения питания и др.). В основном используют интегральные операционные усилители (ИОУ).

Присутствие в названии слова «операционные» объясняется возможностью выполнения данными усилителями ряда математических операций – суммирования, вычитания, дифференцирования, интегрирования и др.

На рисунке 3.9 изображены условные графические обозначения ИОУ. Усилитель имеет два входа (прямой и инверсный) и один выход. При подаче входного сигнала на неинвертирующий (прямой) вход выходной сигнал имеет ту же полярность или фазу (рис. 3.10, а).

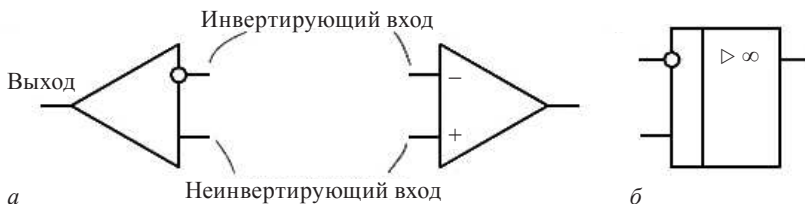


Рис. 3.9. Условные графические обозначения операционных усилителей:
а – на схемах; б – структурное

При использовании инвертирующего входа фаза выходного сигнала будет сдвинута на 180° по отношению к фазе входного сигнала — полярность изменяется на противоположную (рис. 3.10, б). Инверсные входы (выходы) обозначают кружком.

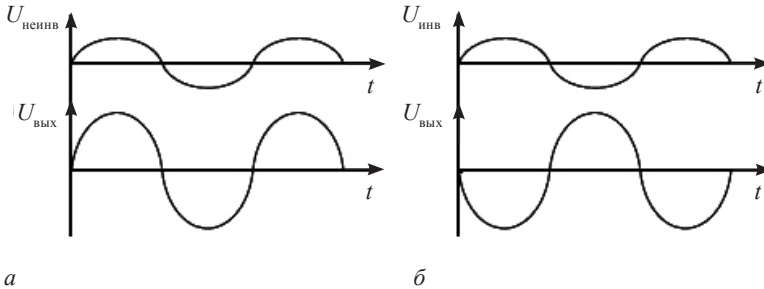


Рис. 3.10. Временные диаграммы операционных усилителей:
а — неинвертирующего; б — инвертирующего

При подаче напряжения на оба входа выходное напряжение пропорционально разности входных напряжений, т. е. сигнал на инвертирующем входе берется со знаком «−»:

$$U_{\text{вых}} = K(U_{\text{нсинв}} - U_{\text{инв}}),$$

где K — коэффициент усиления.

Питание ОУ осуществляется от двухполярного источника, обычно $+15$ В и -15 В. Также допускается однополярное питание. Остальные выводы ИОУ указывают по мере их использования.

Работу ОУ поясняет амплитудная характеристика, приведенная на рисунке 3.11. На характеристике можно выделить линейный участок, на котором с повышением входного напряжения пропорционально увеличивается выходное и два участка насыщения $U_{\text{нас}}^+$ и $U_{\text{нас}}^-$. При определенном значении входного напряжения $U_{\text{вх max}}$ усилитель переходит в режим насыщения, при котором выходное напряжение принимает максимальное значение (при значении $U_{\text{п}} = 15$ В примерно $U_{\text{нас}} = 13$ В) и остается неизменным при дальнейшем увеличении входного сигнала. Режим насыщения используют в импульсных устройствах на ОУ.

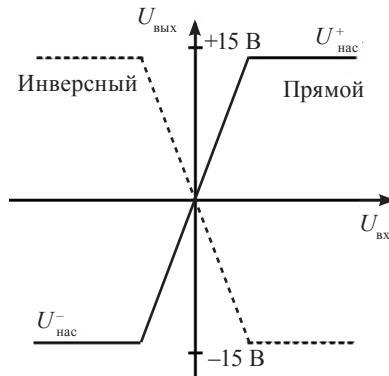


Рис. 3.11. Амплитудная характеристика операционного усилителя

Неинвертирующий усилитель (рис. 3.12) не переворачивает фазу входного сигнала. Напряжение ООС через делитель на резисторах R_1 и R_2 подается на инвертирующий вход.

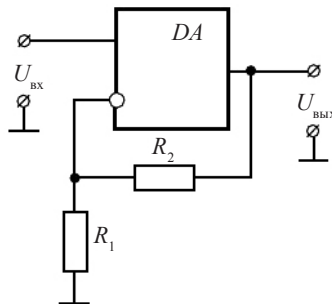


Рис. 3.12. Неинвертирующий усилитель

Коэффициент усиления определяют по формуле

$$K = \frac{U_{\text{ВЫХ}}}{U_{\text{ВХ}}} = 1 + \frac{R_2}{R_1}.$$

Для обеспечения нужного коэффициента усиления необходимо задать соответствующие значения R_1 и R_2 . Значение R_1 выбирают в диапазоне 2–10 кОм.

Инвертирующий усилитель (рис. 3.13) не только усиливает входной сигнал, но и инвертирует (переворачивает) фазу сигнала на 180° .

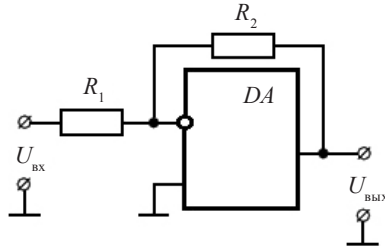


Рис. 3.13. Инвертирующий усилитель

Коэффициент усиления определяют по формуле

$$K = U_{\text{ВЫХ}} / U_{\text{ВХ}} = -R_2 / R_1.$$

Знак «-» показывает инвертирование фазы (изменение полярности).

Суммирующий усилитель (рис. 3.14) выполняет операцию сложения с усилением суммы в K раз.

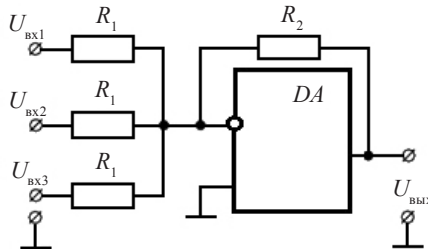


Рис. 3.14. Суммирующий усилитель

При равенстве входных сопротивлений выходное напряжение определяют по формуле

$$U_{\text{ВЫХ}} = -\frac{R_2}{R_1} (U_{\text{ВХ1}} + U_{\text{ВХ2}} + U_{\text{ВХ3}}).$$

Сложение входных сигналов происходит с учетом их полярности.

Вычитающий усилитель (рис. 3.15) выполняет операцию вычитания двух напряжений с усилением разности в K раз. При условии $R_2 / R_1 = R_4 / R_3$ выходное напряжение определяют по формуле (также с учетом полярности входных напряжений):

$$U_{\text{ВЫХ}} = R_2 / R_1 (U_{\text{ВХ2}} - U_{\text{ВХ1}}).$$

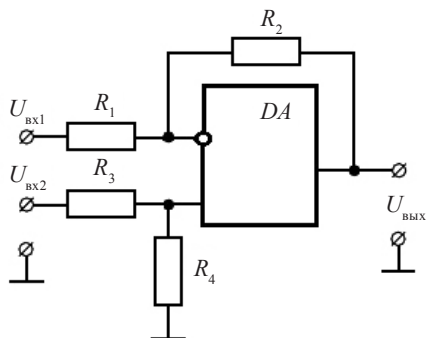


Рис. 3.15. Вычитающий усилитель

Контрольные вопросы и задания

1. Дайте определение понятия «усилитель».
2. Как классифицируют усилительные устройства?
3. Что является основными параметрами ЭУ?
4. Опишите общую структуру усилительного каскада.
5. Поясните принцип работы усилителя на БПТ.
6. Что называют обратной связью усилителя?
7. Расскажите об усилителях напряжения, тока и мощности и их характерных особенностях.
8. Расскажите о принципе работы и видах операционных усилителей.
9. Где применяют интегральные микросхемы?

4. ТИПОВЫЕ ЗВЕНЬЯ СИСТЕМ АВТОМАТИЧЕСКОГО РЕГУЛИРОВАНИЯ

4.1. Режимы работы объекта. Возмущающие воздействия

При исследовании объектов управления используют уравнения взаимосвязи выходного сигнала объекта (регулируемого параметра) с входным сигналом.

Объект управления может работать в двух режимах: статическом и динамическом.

Статический режим – это режим работы, при котором приток энергии или вещества в объект равен оттоку и объект находится в состоянии равновесия. Например, движение поезда по горизонтальной плоскости с постоянной скоростью, когда сила тяги уравновешена силой трения.

Динамический режим – режим работы объекта, при котором нарушено равновесие между притоком и оттоком энергии или вещества.

Поскольку в динамическом режиме работы входной и выходной сигналы объекта изменяются во времени или пространстве, его динамическая характеристика, как правило, описывается дифференциальным уравнением, являющимся основным инструментом при математическом описании всего, что изменяется во времени и пространстве: от процесса набухания почек растения до перемещения космических тел.

Если при исследовании не учитывать физико-химическую сущность технологических процессов, протекающих в объекте, то различные по своему объему, конфигурации и назначению объекты управления в динамическом режиме описываются од-

ними и теми же типами уравнений взаимосвязи выходного и входного сигналов.

Удачно подобрано всего шесть типов таких уравнений взаимосвязи выходного и входного сигналов объектов (или элементов) САУ, которые названы **типowymi динамическими звеньями** и которые составляют математический аппарат, применяемый для исследования объектов с целью получить математические модели.

Методика применения совокупности шести типовых динамических звеньев заключается в следующем.

На действующий объект управления по входному каналу подается одно из трех типовых возмущающих воздействий (рис. 4.1): единичный скачок; единичный импульс; синусоидальные колебания с различной частотой ω или периодом T .

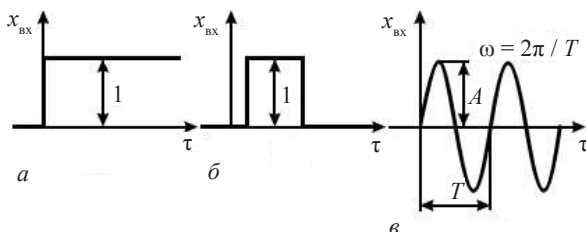


Рис. 4.1. Типовые возмущающие воздействия:
 a – единичный скачок; b – единичный импульс;
 v – синусоидальные колебания

Амплитуда A синусоидальных колебаний может быть постоянной, но по значению она должна быть достаточной, чтобы выходной сигнал также был синусоидальным.

Чаще всего применяют возмущающее воздействие в виде единичного скачка. Реакцию объекта на это возмущение (график изменения во времени выходного сигнала объекта после подачи его на вход возмущения в виде единичного скачка) называют **кривой разгона**.

Каждое типовое динамическое звено имеет свою типовую кривую разгона и ряд других типовых характеристик.

4.2. Апероодическое (инерционное, статическое) звено

Типовое дифференциальное уравнение взаимосвязи выходного и входного сигналов апероодического типового динамического звена имеет следующий вид:

$$T_0 \frac{dx_{\text{ВЫХ}}}{dt} + x_{\text{ВЫХ}} = kx_{\text{ВХ}},$$

где T_0 – постоянная времени; k – коэффициент передачи.

Дифференциальное уравнение является неудобной формой математической модели звена или объекта, так как решение большинства дифференциальных уравнений – сложная процедура. Более удобна математическая модель объекта, записанная в виде передаточной функции.

Передаточной функцией называют преобразованное по Лапласу исходное дифференциальное уравнение, т. е. уравнение, записанное в виде отношения преобразованных по Лапласу выходного и входного сигналов звена (объекта).

В преобразовании по Лапласу исходное дифференциальное уравнение называется оригиналом, а преобразованное и записанное в операторной форме – его изображением. Суть преобразования Лапласа заключается в замене функций вещественных переменных – $x_{\text{ВЫХ}}(\tau)$ и $x_{\text{ВХ}}(\tau)$ на функции комплексных переменных – $x_{\text{ВЫХ}}(p)$ и $x_{\text{ВХ}}(p)$, где p – оператор Лапласа (комплексное число $p = \pm m \pm in$). Эти функции связаны между собой интегралом Лапласа:

$$x_{\text{ВЫХ}}(p) = \int_0^{\infty} x_{\text{ВЫХ}}(\tau) e^{-p\tau} d\tau;$$

$$x_{\text{ВХ}}(p) = \int_0^{\infty} x_{\text{ВХ}}(\tau) e^{-p\tau} d\tau.$$

Для большинства дифференциальных уравнений, используемых в описании типовых динамических звеньев, чисто формальным условием перехода от оригинала к изображению будут следующие замены:

$$dx / dt \text{ на } p; \quad d^2x / dt^2 \text{ на } p^2 \text{ и т. д.}$$

Таким образом легко можно получить из оригинала изображение, т. е. операторную форму записи дифференциального уравнения аперiodического звена следующего вида:

$$T_0 p x_{\text{ВЫХ}}(p) + x_{\text{ВЫХ}}(p) = k x_{\text{ВХ}}(p).$$

Преимущество такого преобразования заключается в том, что записанное в операторной форме исходное дифференциаль-

ное уравнение становится алгебраическим, которое можно преобразовать следующим образом:

$$(T_0 p + 1)x_{\text{вых}}(p) = kx_{\text{вх}}(p).$$

Из этого выражения легко получить отношение $x_{\text{вых}}(p) / x_{\text{вх}}(p)$, которое называется передаточной функцией и для аperiodического звена имеет следующий вид:

$$W(p) = \frac{x_{\text{вых}}(p)}{x_{\text{вх}}(p)} = \frac{k}{T_0 p + 1}.$$

Каждое типовое динамическое звено имеет ряд типовых частотных характеристик: амплитудно-частотную (АЧХ); фазочастотную (ФЧХ); амплитудно-фазовую частотную АФЧХ (или АФХ); логарифмическую амплитудно-частотную (ЛАЧХ); логарифмическую фазочастотную (ЛФЧХ).

На практике чаще используют амплитудно-фазовую частотную характеристику (или АФХ).

На рисунке 4.2 приведена типовая кривая разгона аperiodического звена — *экспонента*. Любая экспонента обладает одним

свойством: если к любой ее точке провести касательную, а затем точку касания и точку пересечения касательной с асимптотой, к которой со временем приближается экспонента, спроецировать на ось времени, то получится один и тот же отрезок на оси времени. Эта проекция, называемая *постоянной времени*, соответствует коэффициенту T_0 в передаточной функции и АФХ аperiodического звена, а ордината асимптоты, к которой стремится экспонента, — коэффициенту k в его передаточной функции. Таким образом по кривой разгона легко найти коэффициенты k и T_0 в передаточной функции аperiodического звена.

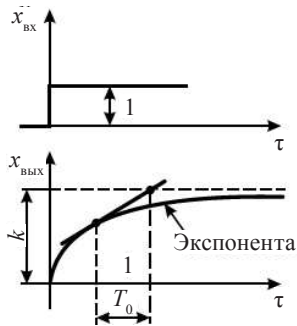


Рис. 4.2. Входной сигнал и типовая кривая разгона аperiodического звена

коэффициенты k и T_0 в передаточной функции аperiodического звена.

Примером реализации аperiodического звена является электродвигатель небольшой мощности, который после включения в электросеть (подачи единичного скачка) набирает обороты по экспоненте.

Еще одним примером реализации аperiodического звена может быть установка, изображенная на рисунке 4.3.

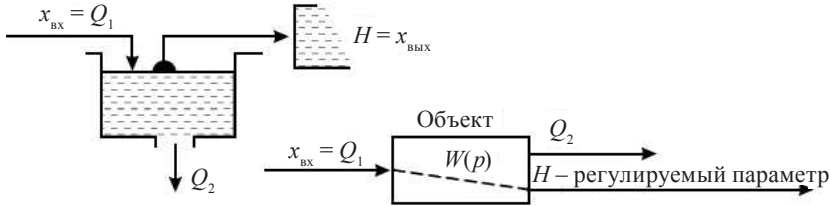


Рис. 4.3. Пример реализации аperiodического звена

В бак поступает поток воды с расходом Q_1 , из бака вытекает свободно поток воды с расходом Q_2 . Регулируемый параметр $x_{\text{вых}}$ — это уровень H воды в баке.

При подаче единичного скачка Q_1 уровень H в баке повышается, при этом увеличивается гидростатическое давление и возрастает Q_2 . Затем уровень воды H стабилизируется, т. е. экспонента приближается к асимптоте. Эта способность самостоятельно восстанавливать равновесие, присущая объектам, аппроксимируемым аperiodическим типовым динамическим звеном, за счет притока или стока энергии либо вещества называется **самовыравниванием**. Количественно самовыравнивание определяется коэффициентом p , равным обратному значению коэффициента k в передаточной функции звена, т. е. $p = 1 / k$.

Объекты с передаточной функцией аperiodического звена называют **статическими**.

4.3. Астатическое (интегрирующее) звено

Типовое дифференциальное уравнение астатического звена

$$T \frac{dx_{\text{вых}}}{dt} = x_{\text{вх}}.$$

Это уравнение можно решить в общем виде, используя интегралы левой и правой его частей:

$$\int T \frac{dx_{\text{вых}}}{dt} = \int x_{\text{вх}} dt \quad \text{или} \quad T x_{\text{вых}} = x_{\text{вх}}.$$

Посредством преобразования Лапласа получаем передаточную функцию астатического звена.

Операторная форма записи дифференциального уравнения:

$$Tp x_{\text{вых}}(p) = x_{\text{вх}}(p).$$

Передаточная функция имеет вид

$$W(p) = \frac{x_{\text{вых}}(p)}{x_{\text{вх}}(p)} = \frac{1}{Tp}.$$

На рисунке 4.4 изображена типовая кривая разгона астатического звена. По этому графику видно, что объект, аппроксимируемый астатическим звеном, не обладает свойством самовыравнивания.

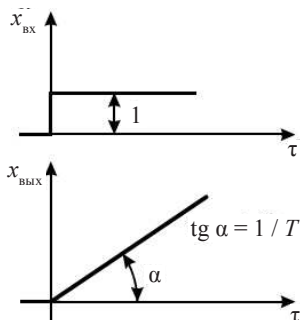


Рис. 4.4. Входной сигнал и типовая кривая разгона астатического звена

По кривой разгона легко определить коэффициент T в передаточной функции астатического звена:

$$T = 1 / \operatorname{tg} \alpha = \operatorname{arctg} \alpha.$$

Иногда передаточную функцию астатического звена записывают в виде

$$W(p) = \frac{k}{p} = \frac{1}{\frac{1}{k} p} = \frac{1}{Tp}.$$

Примером реализации астатического звена может служить цилиндрический бак, из которого вода откачивается насосом постоянной производительности (рис. 4.5).

Равновесие в этой системе наступает только при равенстве входного потока Q_1 производительности насоса Q_2 . В остальных случаях будет непрерывное наполнение или опорожнение бака в соответствии с кривой разгона типового астатического звена.

Объекты, которые аппроксимируются астатическим (интегрирующим) звеном, называют **астатическими**. Такие объекты не обладают свойством самовыравнивания.

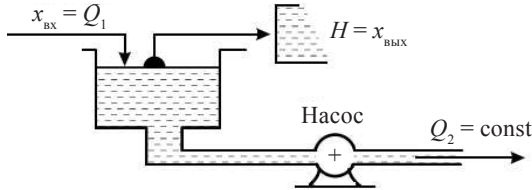


Рис. 4.5. Пример реализации астатического звена

4.4. Колебательное звено (апериодическое звено второго порядка)

Дифференциальное уравнение колебательного звена

$$T_1 \frac{dx_{\text{ВЫХ}}^2}{d\tau^2} + T_2 \frac{dx_{\text{ВЫХ}}}{d\tau} + x_{\text{ВЫХ}} = kx_{\text{ВХ}}.$$

Заменяв d^2 / dx^2 на p^2 , а d / dx на p , получим операторную форму этого уравнения:

$$T_1 p^2 x_{\text{ВЫХ}}(p) + T_2 p x_{\text{ВЫХ}}(p) + x_{\text{ВЫХ}}(p) = kx_{\text{ВХ}}(p).$$

Преобразуя последнее уравнение в отношении выходного сигнала к входному, получаем передаточную функцию колебательного звена:

$$W(p) = \frac{x_{\text{ВЫХ}}(p)}{x_{\text{ВХ}}(p)} = \frac{k}{T_1 p^2 + T_2 p + 1}.$$

Типовые кривые разгона колебательного и апериодического звена второго порядка приведены на рисунке 4.6. Различие кривых определяют соотношением коэффициентов T_1 и T_2 в исходном типовом дифференциальном уравнении: если $T_2^2 < 4T_1$, то система ведет себя как колебательное звено, если же $T_2^2 \geq 4T_1$ – как апериодическое звено второго порядка.

Очевидно, что объекты, аппроксимируемые колебательным или апериодическим звеном второго порядка, обладают свойством самовыравнивания, т. е. способностью самостоятельно восстанавливать состояние равновесия после возмущающего воздействия.

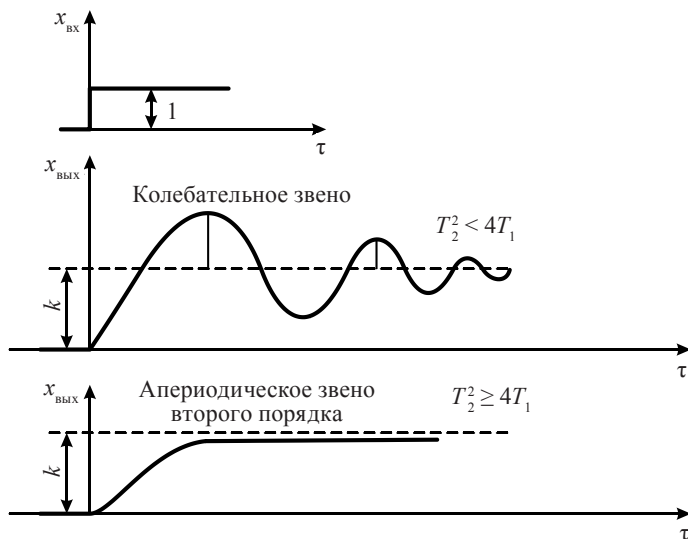


Рис. 4.6. Входной сигнал и типовые кривые разгона колебательного и апериодического второго порядка звеньев

Примером колебательного звена может служить механическая система (колесная пара вагона), изображенная на рисунке 4.7, *а*, а апериодического звена второго порядка – система из двух проточных прудов (рис. 4.7, *б*), в которой регулируемым параметром $x_{\text{вых}}$ является уровень H_2 воды во втором пруду.

По кривым разгона колебательного и апериодического звеньев легко найти значение коэффициента k в их передаточной функции. Определить коэффициенты T_1 и T_2 сложнее.

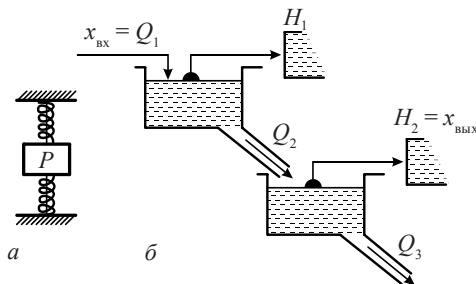


Рис. 4.7. Примеры реализации колебательного (*а*) и апериодического второго порядка (*б*) звеньев

4.5. Пропорциональное (усилительное, безынерционное) звено

Типовое уравнение взаимосвязи выходного и входного сигналов пропорционального звена является алгебраическим, т. е. оно может служить операторной формой записи уравнения звена:

$$x_{\text{ВЫХ}} = kx_{\text{ВХ}}.$$

Передаточная функция пропорционального звена

$$W(p) = \frac{x_{\text{ВЫХ}}(p)}{x_{\text{ВХ}}(p)} = \frac{x_{\text{ВЫХ}}}{x_{\text{ВХ}}} = k.$$

Пропорциональное звено мгновенно (без инерции) реагирует на возмущающее воздействие. По типовой кривой разгона (рис. 4.8) видно, что выходной сигнал этого звена пропорционален входному сигналу, и ордината выходного сигнала равна коэффициенту пропорциональности k .

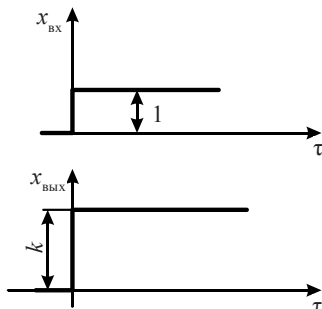


Рис. 4.8. Входной сигнал и типовая кривая разгона пропорционального звена

Примером реализации пропорционального звена может служить жесткий стержень, лежащий на опоре (рис. 4.9), при перемещении одного конца которого ($x_{\text{ВХ}}$) мгновенно перемещается другой его конец ($x_{\text{ВЫХ}}$).



Рис. 4.9. Пример реализации пропорционального звена

4.6. Дифференцирующее звено

Различают идеальное дифференцирующее и реальное дифференцирующее типовые динамические звенья.

Рассмотрим *идеальное дифференцирующее звено*, дифференциальное уравнение которого имеет вид

$$x_{\text{вых}} = k \frac{dx_{\text{вх}}}{d\tau}.$$

Операторная форма записи этого уравнения:

$$x_{\text{вых}}(p) = kp x_{\text{вх}}(p).$$

Передаточная функция идеального дифференцирующего звена:

$$W(p) = \frac{x_{\text{вых}}(p)}{x_{\text{вх}}(p)} = kp.$$

Типовая кривая разгона идеального дифференцирующего звена своеобразна (рис. 4.10). Выходной сигнал этого звена пропорционален первой производной входного сигнала, т. е. тангенсу угла наклона вектора АФХ. В момент подачи входного воздействия этот угол равен $+90^\circ$, а $\text{tg}(+90^\circ) = +\infty$, но далее входное воздействие устанавливается равным единице, при этом угол наклона становится равным -90° , а $\text{tg}(-90^\circ) = -\infty$.

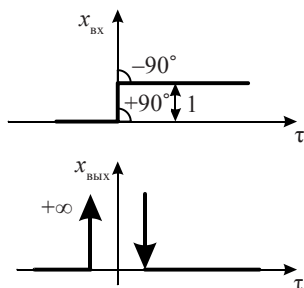


Рис. 4.10. Входной сигнал и типовая кривая разгона идеального дифференцирующего звена

Следовательно, выходной сигнал идеального дифференцирующего звена в момент подачи входного воздействия принимает значение $+\infty$. Здесь же из $+\infty$ вычитается $-\infty$, и выходной сигнал возвращается в исходное нулевое состояние (см. рис. 4.10).

Примером реализации идеального дифференцирующего звена может быть электрическая RC -цепь (рис. 4.11), состоящая из конденсатора с емкостью C и резистора R , обладающего сверхпроводимостью ($R = 0$).

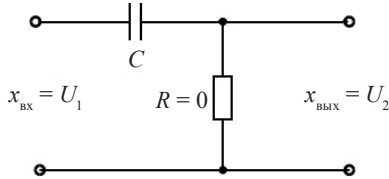


Рис. 4.11. Пример реализации идеального дифференцирующего звена

Рассмотрим реальное дифференциальное звено. Его типовое дифференциальное уравнение имеет вид

$$T_0 \frac{dx_{\text{ВЫХ}}}{d\tau} + x_{\text{ВЫХ}} = k \frac{dx_{\text{ВХ}}}{d\tau}.$$

Операторная форма записи этого уравнения

$$T_0 p x_{\text{ВЫХ}}(p) + x_{\text{ВЫХ}}(p) = k p x_{\text{ВХ}}(p)$$

или

$$(T_0 p + 1) x_{\text{ВЫХ}}(p) = k p x_{\text{ВХ}}(p).$$

Получим аналитическое выражение передаточной функции реального дифференцирующего звена:

$$W(p) = \frac{x_{\text{ВЫХ}}(p)}{x_{\text{ВХ}}(p)} = \frac{k p}{T_0 p + 1}.$$

Типовая кривая разгона реального дифференцирующего звена (рис. 4.12) показывает, что после подачи на его вход возмущения в виде единичного скачка выходной сигнал мгновенно возрастает на величину k / T_0 , а затем по экспоненте постепенно приближается к нулю. Таким образом по кривой разгона легко определить коэффициенты T_0 и k передаточной функции звена, т. е. сначала с помощью касательной (см. п. 4.2) находят значение T_0 , а затем, умножив ординату величины k / T_0 на T_0 , получают значение k .

Примером реализации реального дифференцирующего звена будет RC -цепь (см. рис. 4.11), в которой сопротивление $R \neq 0$, что реально имеет место.

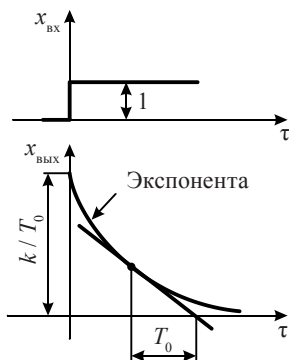


Рис. 4.12. Входной сигнал и типовая кривая разгона реального дифференцирующего звена

4.7. Запаздывающее звено

В качестве примера для рассмотрения запаздывающего звена можно привести ленточный транспортер длиной L , перемещающийся со скоростью v (рис. 4.13). Если расход сыпучего материала в начале такого транспортера принять $Q_1 = x_{\text{вх}}$, а расход сбрасываемого в конце с транспортера материала $Q_2 = x_{\text{вых}}$, то время движения материала Δt будет $L / v = \tau_{\text{зап}}$, где $\tau_{\text{зап}}$ — время запаздывания.

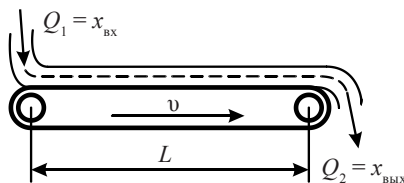


Рис. 4.13. Пример реализации запаздывающего звена

Если на вход ленточного транспортера подать возмущение в виде единичного скачка, т. е. открыть подачу материала, то этот же единичный скачок появится на его выходе через отрезок времени, равный времени запаздывания. Следовательно, типовая кривая разгона будет иметь следующий вид (рис. 4.14).

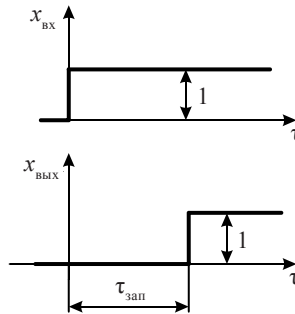


Рис. 4.14. Входной сигнал и типовая кривая разгона запаздывающего звена

Предположим, на вход транспортера подан единичный импульс. Через определенное время запаздывания на выходе транспортера его же получим. То же будет и при входном потоке, изменяющемся по синусоиде или любому другому закону во времени. Следовательно, выходной сигнал запаздывающего звена повторяет входной сигнал, но с некоторым временем запаздывания. Исходя из этого, общее уравнение взаимосвязи входного и выходного сигналов запаздывающего звена в динамическом режиме работы можно записать в виде

$$x_{\text{вх}}(\tau) = x_{\text{вых}}(\tau + \tau_{\text{зап}})$$

или

$$x_{\text{вых}}(\tau + \tau_{\text{зап}}) = x_{\text{вх}}(\tau).$$

Передаточная функция запаздывающего звена в общем виде

$$W(p) = \frac{x_{\text{вых}}(p)}{x_{\text{вх}}(p)} = e^{-p\tau_{\text{зап}}}.$$

4.8. Показатели качества процесса регулирования

Для различных систем регулирования важен характер затухания переходного процесса. Так, затухание переходного процесса может происходить медленно или быстро. Медленно – значит, система долго выходит на новый установившийся режим, т. е. она обладает недостаточным быстродействием и, следовательно, применение ее ограничено. Если затухание переходного процесса в САР происходит быстро, система обладает высокой степенью

работоспособности. Переходный процесс может быть с большим или малым отклонением регулируемого параметра от заданного значения. Следовательно, устойчивость – необходимое, но недостаточное условие работоспособности САР. Достаточным условием является качество процесса регулирования, которое оценивается по форме переходного процесса, полученного в результате единичного скачкообразного возмущения ΔQ , относительно его номинального значения $Q_{\text{ном}}$ (рис. 4.15, а).

Продолжительность регулирования T_p – это длительность переходного процесса с момента отклонения регулируемой величины от заданного значения до момента возвращения ее регулятором к заданному значению или новому установившемуся значению с заданной точностью.

Практически считают, что переходный процесс заканчивается тогда, когда отклонение регулируемой величины $x(\tau)$ от нового установившегося значения $x_{\text{уст}}$ не будет превышать допустимых пределов ε (рис. 4.15, б).

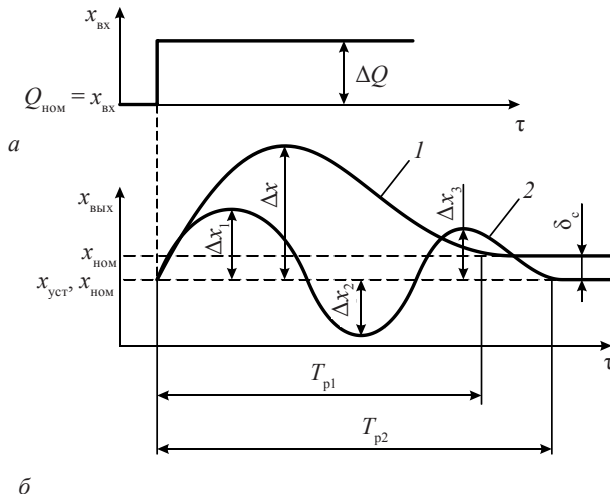


Рис. 4.15. Показатели качества переходных процессов САР:
 а – график скачкообразного возмущающего воздействия;
 б – переходные процессы

Обычно принимают $\varepsilon = (0,03...0,05)x_{\text{уст}}$ (кривая 1).

Продолжительность регулирования характеризует быстроедействие системы.

Перерегулированием δ называется отношение амплитуды второй полуволны Δx_2 колебательного переходного процесса к амплитуде колебаний в первом периоде Δx_1 (кривая 2), выраженное в процентах: $\delta = (\Delta x_2 / \Delta x_1) \cdot 100$.

Колебательность переходного процесса регулирования характеризуется степенью затухания.

Степень затухания ψ – отношение разности между положительными амплитудами первого и второго периодов колебательного возмущения к величине амплитуды первого периода колебаний (кривая 2), выраженное в процентах: $\psi = [(\Delta x_1 - \Delta x_2) / \Delta x_1] \cdot 100$.

Чем выше степень затухания, тем лучше качество регулирования. Для устойчивых САР $0 < \psi \leq 1$.

Статическая ошибка δ_c – максимальное остаточное отклонение регулируемой величины от номинального ее значения в конце переходного процесса (кривая 1), которое получается при максимально возможных в данной системе возмущениях. Принято статическую ошибку выражать в процентах от номинального значения регулируемой величины $x_{ном}$, т. е. $\eta = (\delta_c / x_{ном}) \cdot 100$.

Для реальных автоматических систем регулирования статическая ошибка не должна превышать 0,03–0,05 % от $x_{ном}$. Обычно величина допустимой статической ошибки задается технологическими требованиями к процессу регулирования.

Максимальное динамическое отклонение $\Delta x_{max} = \Delta x_1 = \Delta x$ представляет собой величину максимального отклонения регулируемого параметра от заданного. Эта величина соответствует первой полуволне переходного процесса регулирования.

Отклонение называют динамическим, поскольку оно имеет временной характер. Величина динамического отклонения ограничивается технологическими требованиями к процессу регулирования.

Контрольные вопросы и задания

1. Раскройте сущность динамического режима работы объекта управления.
2. Какие типы возмущающего воздействия входного канала действуют на объект управления?

3. Перечислите типовые динамические звенья систем автоматического регулирования.
4. Какие типовые частотные характеристики присущи динамическим звеньям систем автоматического регулирования?
5. Приведите пример реализации интегрального звена систем автоматического регулирования.
6. Определите зависимость входного и выходного сигналов в колебательном звене системы автоматического регулирования.
7. Какие условия определяют работоспособность системы автоматического регулирования?

5. ДИСКРЕТНЫЕ АВТОМАТЫ

5.1. Понятие дискретного автомата

Законченный технологический цикл на производстве представляет совокупность отдельных технологических операций, выполняемых в определенной последовательности. Управление технологическим циклом подразумевает формирование дискретной во времени последовательности (программы) команд для исполнительных элементов, участвующих в этом цикле (электро-, гидропривода и т. д.). Формирование команд осуществляется устройством управления (схемой управления) на основе логического анализа ситуации, информацию о которой выдают различные датчики, используемые в технологическом цикле.

Устройство управления с жесткой логикой, обрабатывающее входные сигналы датчиков и аппаратов управления и вырабатывающее выходные двоичные управляющие переменные, назовем **дискретным управляющим автоматом**, в отличие от программируемого контроллера – устройства, выполняющего функции дискретного управляющего автомата, но реализованного на программируемой логике. Тогда структура управления технологическим циклом при помощи дискретного автомата может быть представлена следующей схемой (рис. 5.1).

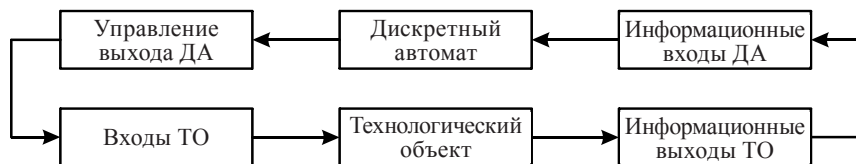


Рис. 5.1. Структура управления технологическим циклом:
ДА – дискретный автомат; ТО – технологический объект

Приведем пример контактной схемы управления, рассматриваемой как дискретный автомат (рис. 5.2).

Выходами схемы являются промежуточное реле $K1$, $K2$, а входами – кнопки управления $S1$, $S2$. Эта схема управления может быть представлена в виде модели дискретного автомата (рис. 5.3) и описана функциями $Z1 = f_1(X1, X2)$, $Z2 = f_2(X1, X2)$.

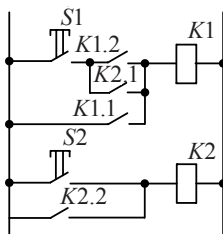


Рис. 5.2. Схема управления

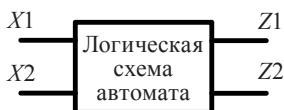


Рис. 5.3. Модель дискретного автомата

Условия срабатывания реле $K1$, $K2$ представлены как логические функции $Z1$, $Z2$ от входных логических переменных $X1$, $X2$, отображающих реальные состояния кнопок $S1$, $S2$, а в качестве логической схемы автомата рассматривается логический алгоритм функционирования реле $K1$, $K2$.

Дискретные управляющие автоматы в общем случае могут иметь один или несколько входов и выходов, а состояние сигнала каждого из них характеризуется двумя уровнями: логические нуль или единица. Для случая нескольких входов (выходов) комбинацию нулевых и единичных уровней можно воспринимать двояко:

- как наличие или отсутствие на каждом входе (выходе) задающего или командного сигнала;
- как кодовую комбинацию, обозначающую, например, двоичное число, количество разрядов которого равно количеству входов (выходов).

В первом случае дискретный автомат решает задачи логического анализа ситуации, во втором – это арифметическое устройство, преобразующее информацию путем выполнения арифметических операций над числами.

Все дискретные управляющие автоматы можно разделить на две большие группы: комбинационные логические схемы (комбинационные автоматы) и последовательностные логические схемы (дискретные автоматы с памятью).

Комбинационные автоматы — это схемы, состояние выходных логических переменных которых однозначно зависит от состояния входных переменных на рассматриваемый момент времени. При изменении состояния входов мгновенно изменяется и состояние на выходе (при пренебрежении инерционностью элементов схемы).

Дискретные автоматы с памятью — это схемы, у которых состояние выходных логических переменных зависит не только от состояния входных переменных на рассматриваемый момент времени, но и определяется последовательностью входных сигналов, поступивших в предшествующий момент времени.

Основной задачей проектирования комбинационных дискретных автоматов являются разработка схемы автомата по заданным условиям его работы на выбранной элементной базе, причем схемы наиболее простой (с минимальным числом элементов), и проверка ее на состязания. Эта задача решается в следующем порядке:

1) по словесному (содержательному) описанию условий работы автомата формализуется работа автомата в виде, позволяющем в дальнейшем использовать функции алгебры логики (так называемое задание автомата);

2) на основании выбранного способа задания работа автомата описывается функциями алгебры логики;

3) минимизировать полученные функции и при необходимости привести их к выбранному базису (к виду используемой элементной базы);

4) построить принципиальную электрическую схему автомата или алгоритма для работы программируемого контроллера.

Поскольку функционирование комбинационного дискретного автомата описывается логическими функциями, связывающими выход автомата с логическими переменными (входами автомата), то для выполнения пунктов 1) и 3) можно использовать способы задания и минимизации логических функций.

5.2. Основы алгебры логики

Основоположителем алгебры логики является немецкий математик Готфрид Вильгельм Лейбниц (1646–1716). Он сделал попытку построить универсальный язык, с помощью которого споры между людьми можно было бы разрешать посредством вычислений. На заложенном Лейбницем фундаменте ирландский

математик Джордж Буль (1815–1864) в середине XIX в. создал новую науку – алгебру логики, которая в отличие от обычной алгебры оперирует не числами, а высказываниями.

Высказывание – это любое утверждение, относительно которого можно сказать, истинно оно или ложно, т. е. соответствует оно действительности или нет.

Дж. Буль произвел революцию в науке, о которой сам не подозревал. Его разработки много позже были взяты за основу построения электронно-вычислительных устройств. Поэтому алгебру логики также называют **булевой алгеброй**. Поскольку в булевой алгебре используют всего два значения – *истина* и *ложь*, то, присвоив им соответственно код «1» и «0», получили отличный математический инструмент синтеза и анализа различных цифровых устройств.

5.3. Основные функции и законы алгебры логики

Алгебра логики базируется на трех функциях, определяющих три основные логические операции. Реализуют функции алгебры логики с помощью **логических элементов**, которые используют для построения преобразователей цифровых сигналов комбинационного типа. В комбинационных устройствах отсутствует внутренняя память. Сигналы на их выходах в любой момент однозначно определяются сочетаниями сигналов на входах и не зависят от предыдущих состояний схемы. Характерной особенностью комбинационных устройств является отсутствие обратной связи.

1. Функция логического умножения (конъюнкция). Функция логического умножения записывается в виде $f = X1 \& X2$ (символы логического умножения: «&», «^», «·», «x») и читается как *f* есть (эквивалентна) $X1$ и $X2$, поскольку функция истинна тогда, когда истинны первый и второй аргументы (переменные) (табл. 5.1). Конъюнкцию называют **функцией И (AND)**, а логический элемент, реализующий эту функцию, **элементом И**.

Для изображения логических элементов всегда используют условные графические обозначения. В настоящее время в мире существует несколько общепринятых стандартов условных обозначений. Наиболее распространенными являются стандарт обозначения элементов по ГОСТ 2.743–91 (рис. 5.4, *а*), стандарт IEC 60617-12:1997 (рис. 5.4, *б*), созданный Международной электротехнической комиссией, и американский стандарт US ANSI 91–1984 (рис. 5.4, *в*).

Таблица 5.1

Значения истинности элемента И

$X1$	$X2$	f
0	0	0
1	0	0
0	1	0
1	1	1

Необходимо отметить, что входов у логического элемента может быть и один, и несколько, поэтому при обозначении логического элемента всегда указывается их количество, т. е. в представленном примере элемент должен называться 2И.

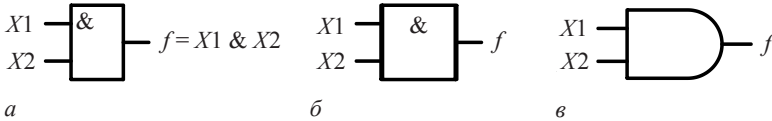


Рис. 5.4. Условные графические обозначения логического элемента И:
 а – по ГОСТ 2.743–91; б – по IEC 60617-12:1997;
 в – по US ANSI 91–1984

2. Функция логического сложения (дизъюнкция). Функция логического сложения записывается в виде $f = X1 \vee X2$ (символы логического сложения: «+», « \vee ») и читается как f есть $X1$ или $X2$, поскольку функция истинна, когда истинна одна или другая переменная (хотя бы одна) (табл. 5.2), поэтому функцию дизъюнкции часто называют функцией ИЛИ (OR) (рис. 5.5).

Таблица 5.2

Значения истинности элемента ИЛИ

$X1$	$X2$	f
0	0	0
1	0	1
0	1	1
1	1	1

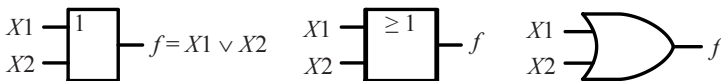


Рис. 5.5. Условные графические обозначения логического элемента ИЛИ

3. Функция отрицания (инверсия). Записывается в виде $f = \bar{X}$ и читается как f есть (эквивалентна) не X (табл. 5.3). Элемент, реализующий функцию НЕ (NOT), называют **инвертором** (рис. 5.6).

Таблица 5.3

Значения истинности элемента НЕ

X	f
0	1
1	0

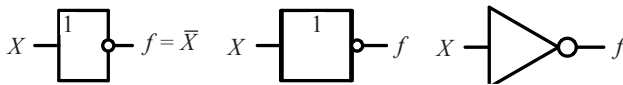


Рис. 5.6. Условные графические обозначения логического элемента НЕ

Основные законы алгебры логики (табл. 5.4) позволяют проводить эквивалентные преобразования функций, записанных с помощью операций И, ИЛИ, НЕ, приводить их к удобному виду для дальнейшего использования и упрощать имеющиеся записи. Кроме того, булевой алгебре свойственен принцип двойственности (исключение п. 6 – закон двойного отрицания).

Таблица 5.4

Основные законы алгебры логики

№ п/п	Часть I	Часть II	Наименование
1	$\bar{0} = 1$	$\bar{1} = 0$	Аксиомы (тождества)
2	$X + 0 = X$	$X \cdot 1 = X$	
3	$X + 1 = 1$	$X \cdot 0 = 0$	
4	$X + X = X$	$X \cdot X = X$	
5	$X + \bar{X} = 1$	$X \cdot \bar{X} = 0$	
6	$\overline{\bar{X}} = X$		Закон двойного отрицания
7	$X + X \cdot Y = X$	$X \cdot [X + Y] = X$	Закон поглощения
8	$X + Y = Y + X$	$X \cdot Y = Y \cdot X$	Закон коммутативности
9	$[\overline{X+Y}] = \bar{X} \cdot \bar{Y}$	$[\overline{X \cdot Y}] = \bar{X} + \bar{Y}$	Правило де Моргана (закон дуальности)

Окончание табл. 5.4

№ п/п	Часть I	Часть II	Наименование
10	$[X + Y] + Z = X + Y + Z$	$[X \cdot Y] \cdot Z = X \cdot Y + X \cdot Z$	Закон ассоциативности
11	$X + Y \cdot Z = [X + Y] \cdot [X + Z]$	$X \cdot [Y + Z] = X \cdot Y + X \cdot Z$	Закон дистрибутивности

5.4. Базисы в алгебре логики

Важным понятием в алгебре логики является понятие о базисе. Совокупность элементарных функций, с помощью которых можно записать любую, сколь угодно сложную функцию, называют **базисом**.

Функционально полными в алгебре логики являются три базиса:

- И–НЕ – базис Шеффера (рис. 5.7, а, табл. 5.5);
- ИЛИ–НЕ – базис Пирса, или функция Вебба (рис. 5.7, б, табл. 5.6);
- И–ИЛИ–НЕ – базис конъюнкции, дизъюнкции, инверсии (рис. 5.7, в, табл. 5.7).

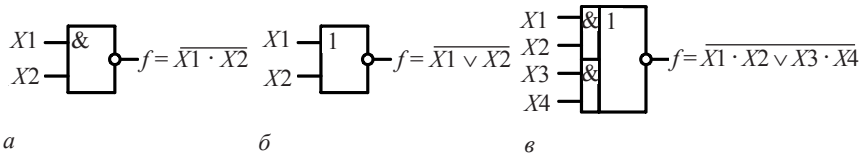


Рис. 5.7. Условные графические обозначения базисов:
а – И–НЕ; б – ИЛИ–НЕ; в – И–ИЛИ–НЕ

Таблица 5.5

Значения истинности элемента И–НЕ

X1	X2	f
0	0	1
1	0	1
0	1	1
1	1	0

Таблица 5.6

Значения истинности элемента ИЛИ–НЕ

X_1	X_2	f
0	0	1
1	0	0
0	1	0
1	1	0

Таблица 5.7

Значения истинности элемента И–ИЛИ–НЕ

X_1	X_2	X_3	X_4	f
0	0	0	1	1
0	0	1	1	0
0	1	1	1	0
1	1	1	1	0
... и т. д.				

5.5. Схемотехническая реализация основных логических функций

Поскольку логические переменные способны принимать только два дискретных значения, пригодны лишь те схемы, у которых имеются два четко различимых рабочих состояния. Ключ S (рис. 5.8) является простейшим из возможных представлений логической переменной. Надо лишь условиться о том, что разомкнутый ключ представляет логический нуль, а замкнутый – логическую единицу. Следовательно, ключ S отображает переменную X , если он замкнут и $X = 1$, и переменную \bar{X} , если он разомкнут и $X = 0$.



Рис. 5.8. Представление логической переменной с помощью ключа

Логические переменные также можно представить с помощью напряжений, различающихся высоким H и низким L уровнями, которым сопоставляются логические состояния 1 и 0. Систему обозначений $H = 1$ и $L = 0$ называют **позитивной логикой**; допустима и обратная система обозначений $H = 0$ и $L = 1$, именуемая **отрицательной логикой**.

Логические функции реализуются посредством соответствующих электронных схем с одним или несколькими входами и единственным выходом. Уровень напряжения на входах и способ реализации логической функции определяют уровень напряжения на выходе.

Для реализации отдельных основных логических функций имеется множество схемотехнических средств, различающихся энергопотреблением, напряжением питания, уровнями 0 и 1, быстродействием логического элемента и нагрузочными свойствами. Для выбора оптимального варианта надо хотя бы в общих чертах знать внутреннее устройство таких схем, в основном за счет интегральных микросхем.

5.6. Резисторно-транзисторная логика

Элементы *резисторно-транзисторной логики* (РТЛ) представляют собой вариант переключающих схем на транзисторах в режиме насыщения.

Если в элементе РТЛ, представленном на рисунке 5.9, входное напряжение отвечает высокому уровню, соответствующий транзистор открыт, а на выходе образуется низкий уровень. В таких условиях при положительной логике реализуется функция ИЛИ–НЕ.

Сравнительно низкоомные резисторы в цепи базы гарантируют проводимость транзистора даже при небольшом усилении по току. Однако отсюда следует низкий коэффициент разветвления по выходу. В настоящее время элементы РТЛ вышли из употребления.

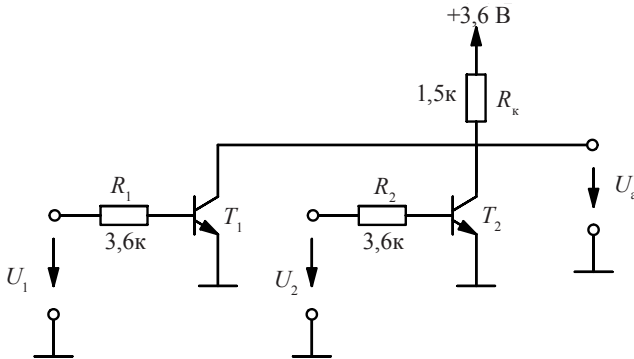


Рис. 5.9. Логический элемент резисторно-транзисторной логики ИЛИ–НЕ (R_k – сопротивление коллектора; U_a – напряжение выхода)

5.7. Диодно-транзисторная логика

В схеме *диодно-транзисторной логики* (ДТЛ), представленной на рисунке 5.10, ток базы выходного транзистора протекает через резистор R_1 при запертых входных диодах D_1 и D_2 , т. е., когда все входные напряжения находятся в состоянии H , транзистор T_1 открыт и выходное напряжение находится на уровне L . Таким образом, если выбрана положительная логика, реализуется функция И–НЕ. Если к выходу подключить аналогичный логический элемент И–НЕ, нагрузка выхода в состоянии H останется прежней. Поэтому уровень H выходного напряжения равен напряжению питания U^+ . Схемы ДТЛ в настоящее время не находят применения из-за недостаточного быстродействия, обусловленного насыщением транзисторов.

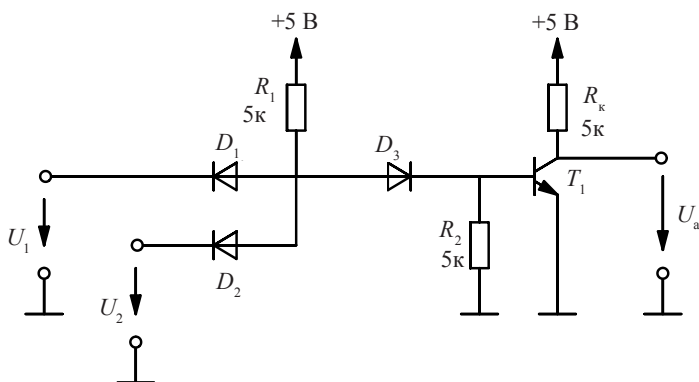


Рис. 5.10. Логический элемент диодно-транзисторной логики И–НЕ

5.8. Медленнодействующая помехоустойчивая логика

Элементы медленнодействующей помехоустойчивой логики (МПЛ) используют в приборах, где возможны значительные импульсные помехи, и представляют собой модифицированные схемы ДТЛ, у которых сдвоенный диод D_3 заменен стабилитроном (рис. 5.11). В результате уровень переключения по входу поднят до 6 В, и при напряжении питания 12 В запас помехоустойчивости достигает 5 В.

Транзисторы, обладающие малым быстродействием, позволяют искусственно увеличить время переключения, а примене-

ние внешних конденсаторов делает его еще бóльшим. Благодаря этому короткие импульсы помех не страшны даже тогда, когда их амплитуда превышает запас помехоустойчивости. Элементы МПЛ называют также схемами с высоким логическим уровнем.

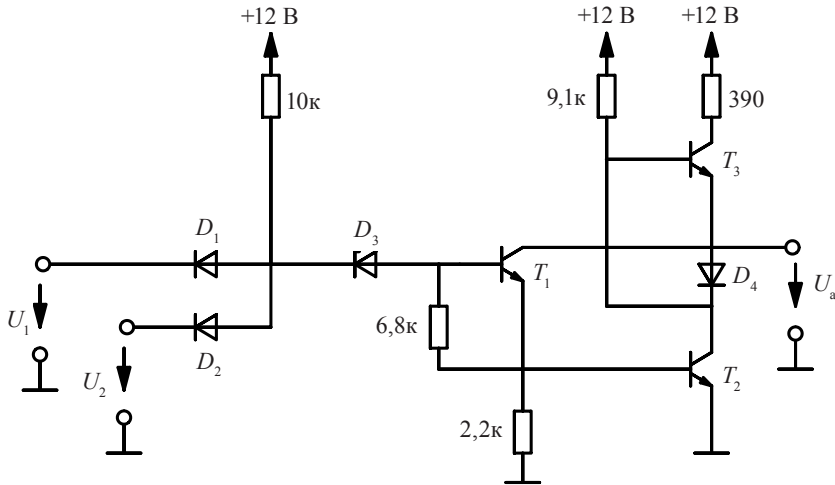


Рис. 5.11. Логический элемент медленнодействующей помехоустойчивой логики И–НЕ

5.9. Транзисторно-транзисторная логика

Принцип действия схем *транзисторно-транзисторной логики* (ТТЛ) аналогичен работе схем ДТЛ, а сами схемы различаются только диодными элементами и усилителем. У стандартной схемы ТТЛ (рис. 5.12) диодный элемент заменен транзистором T_1 с несколькими эмиттерами. Когда все входные уровни отвечают состоянию H , ток через резистор R_1 протекает по открытому в прямом направлении переходу база – коллектор входного транзистора в цепь базы транзистора T_2 и переводит его в открытое состояние.

Если на один из входов подан низкий потенциал, соответствующий переходу база – эмиттер открывается и потенциал базы T_2 оказывается близким к нулю, вследствие чего T_2 запирается, а выходное напряжение соответствует состоянию H .

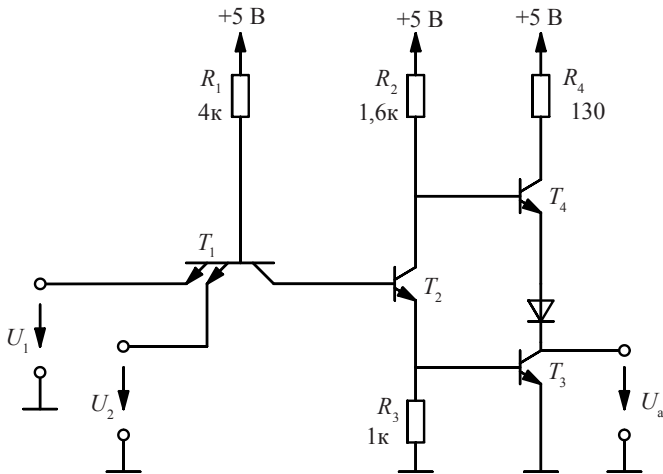


Рис. 5.12. Стандартный логический элемент транзисторно-транзисторной логики И–НЕ (U_a – напряжение на выходе логического элемента)

В схемах ТТЛ усилитель состоит из управляющего транзистора T_2 и двухтактного оконечного каскада. Когда T_2 открыт, T_3 также открыт, а T_4 заперт. На выходе возникает сигнал уровня L , и транзистор T_3 приобретает способность пропускать большие токи, например от подключенных входов логических элементов (в состоянии L ток вытекает из этих входов). Вместе с T_2 запирается T_3 , но при этом открывается T_4 , что приводит к появлению уровня H на выходе. В таком случае транзистор в режиме эмиттерного повторителя может создавать в нагрузке большие выходные токи и, следовательно, быстро заряжать емкости нагрузки. Стандартные схемы ТТЛ, подобные показанной на рисунке 5.12, в настоящее время не находят применения из-за недостаточного быстродействия, обусловленного насыщением транзисторов.

Включение диода Шоттки параллельно переходу коллектор – база (рис. 5.13) предотвращает насыщение транзистора. При открытом транзисторе благодаря обратной связи по напряжению диод не позволяет напряжению коллектор – эмиттер упасть ниже 0,3 В.

На рисунке 5.14 показан логический элемент ТТЛ, который построен из подобных «транзисторов Шоттки», представляющих собой маломощный элемент ТТЛ с диодами Шоттки. Номиналы

резисторов схемы выбираются впятеро более высокоомными по сравнению со схемой стандартного логического элемента ТТЛ, благодаря чему потребляемая мощность снижается столько же раз и достигает всего 2 мВт. При этом быстродействие элемента не ухудшается, оставаясь равным 10 нс. Подобно схемам ДТЛ, диодный входной логический элемент строится на отдельных диодах. Транзистор Дарлингтона T_3 заменяет диод D в оконечном каскаде, необходимый для смещения потенциала в стандартной схеме (см. рис. 5.12).

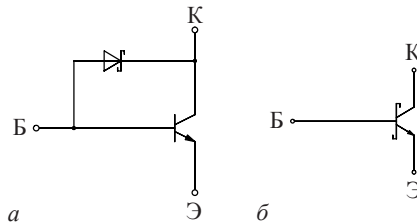


Рис. 5.13. Транзистор с диодом Шоттки для устранения насыщения (а) и соответствующее схемное обозначение (б)

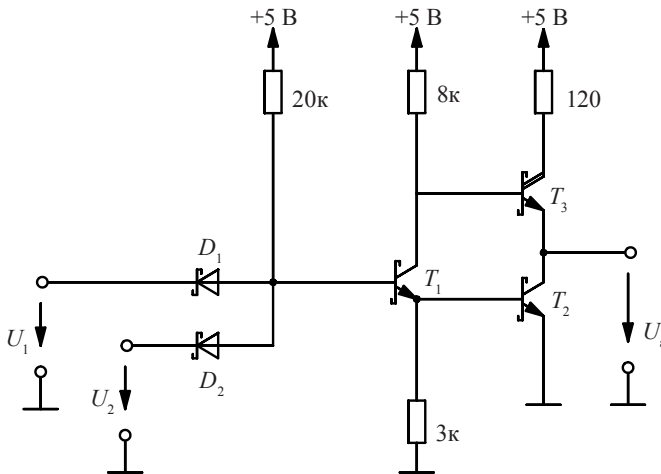


Рис. 5.14. Маломощный логический элемент транзисторно-транзисторной логики с диодом Шоттки

Порог переключения по входу составляет около 1,1 В, что с большим запасом перекрывает заданные допуски: при макси-

мально допустимом уровне L на входе величиной $0,8$ В выходное напряжение получается равным $2,4$ В, что отвечает уровню H . При минимальном входном уровне H величиной $2,0$ В выходной уровень L должен составлять не более $0,4$ В.

Выходы с открытым коллектором. Иногда требуется логически соединять выходы многих элементов. Например, для 20 выходов потребовалось бы проложить 20 проводов к элементу с 20 входами.

Использование элементов с открытым коллектором позволяет обойти эту трудность. У подобного элемента выходным каскадом служит $n-p-n$ -транзистор, эмиттер которого соединен с землей (рис. 5.15). В отличие от обычных двухтактных выходных каскадов, такие выходы можно соединять параллельно друг другу и подключать к общему коллекторному резистору.

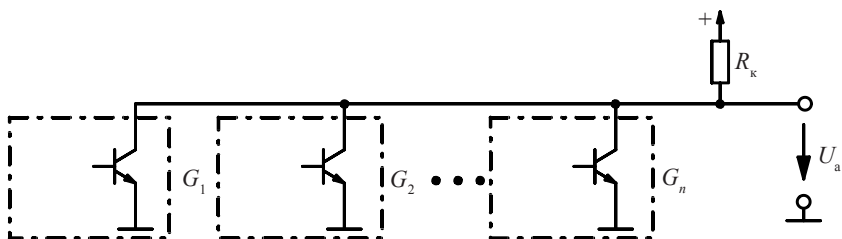


Рис. 5.15. Реализация логической функции путем соединения выходов элементов с открытым коллектором

Выходное напряжение U_a принимает значение, соответствующее состоянию H , только в том случае, когда все выходы оказываются в состоянии H . В позитивной логике это равнозначно функции И. Вместе с тем выходное напряжение соответствует низкому уровню L , когда один или несколько выходов оказываются в состоянии L . Так реализуется функция ИЛИ в отрицательной логике. Поскольку подключения осуществляются с помощью внешнего монтажа, такие соединения называют «монтажным И» и «монтажным ИЛИ». Выходы элементов характеризуются низким выходным сопротивлением только в состоянии L (логический ноль), поэтому их называют также схемами с низкоомным выходным активным нулем. Представление «монтажного И» с помощью условных обозначений показано на рисунке 5.16.

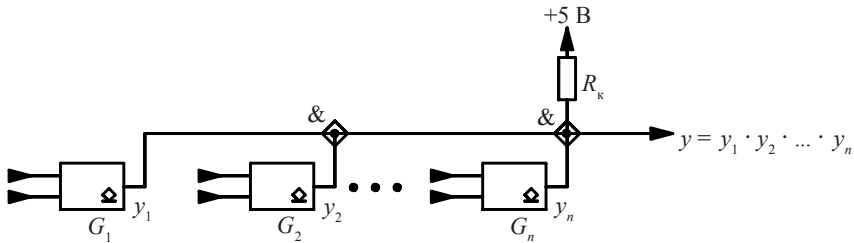


Рис. 5.16. Условные обозначения функции «монтажное И». Символ в виде подчеркнутого ромба означает выход с открытым коллектором

Выходы с открытым коллектором позволяют также реализовать функцию ИЛИ, если «монтажное И» применить к инвертированным переменным. Согласно теореме де Моргана, имеет место

$$y_1 + y_2 + \dots + y_n = \overline{\overline{y_1} \cdot \overline{y_2} \cdot \dots \cdot \overline{y_n}}$$

Соответствующая схема функции ИЛИ приведена на рисунке 5.17.

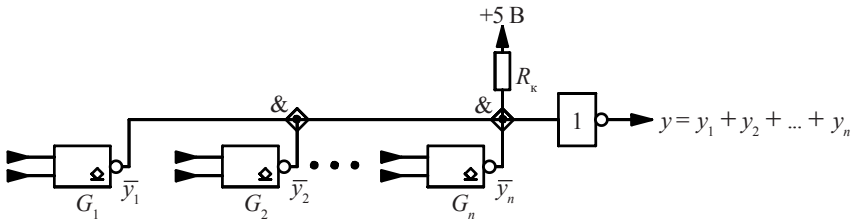


Рис. 5.17. Функция ИЛИ, реализованная посредством выходов с общим коллектором

Недостаток применения выходов с открытым коллектором заключается в более медленном нарастании выходного напряжения по сравнению с двухтактным выходом, так как паразитные емкости могут заряжаться только через резистор R_k . В этом отношении элементы ТТЛ с открытым коллектором сходны со схемами РТЛ (см. рис. 5.15), в которых логическая функция может интерпретироваться как «монтажное И».

Трехстабильные выходы. Имеется еще одно важное применение параллельного включения выходов логических элементов, упрощающее схемы. Это ситуация, когда один из нескольких элементов по выбору должен определять логическое состояние

сигнального проводника. В таком случае говорят о системе с шинной организацией.

Проблема решается также посредством элементов с открытым коллектором (см. рис. 5.15), если все выходы переводятся в высокоомное состояние H . Здесь удастся обойти принципиальный недостаток медленного нарастания сигнала, применив элементы не с открытым коллектором, а с трехстабильным выходом. Это схема с двухтактным выходом, отличающимся тем, что он может переводиться в состояние с высоким выходным сопротивлением при помощи специального управляющего сигнала. Описанное состояние называют состоянием Z .

Принцип схмотехнической реализации показан на рисунке 5.18. Когда разрешающий сигнал $EN = 1$, схема работает как обычный инвертор: при $x = 0$ имеем $z_1 = 0$ и $z_2 = 1$, т. е. T_1 заперт, а T_2 открыт. При $x = 1$ T_1 открыт, а T_2 заперт. Однако, если управляющая переменная $EN = 0$, $z_1 = z_2 = 0$, так что оба выходных транзистора заперты, получаем состояние Z с высоким выходным сопротивлением.

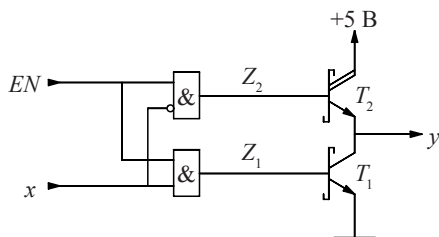


Рис. 5.18. Инвертор с трехстабильным выходом

Самое многочисленное семейство логических элементов образуют маломощные схемы ТТЛ с диодом Шоттки, отличающиеся высокими электрическими параметрами, разнообразием типов и дешевизной.

5.10. Логические схемы на комплементарных МОП-транзисторах

Семейство логических схем на **комплементарных МОП-транзисторах** (КМОП) отличается особой экономичностью по электропитанию. Схема комплементарного МОП-инвертора приведена на рисунке 5.19. Характерно, что она образована исклю-

чительно МОП-транзисторами с индуцированным каналом. При этом истоковые выводы n -канального МОП-транзистора соединены с землей, а p -канального — с источником напряжения питания VDD . Таким образом, оба МОП-транзистора функционируют в схеме с общим истоком, усиливают и инвертируют входное напряжение, причем в каждый момент времени один из них играет роль рабочего нагрузочного резистора для другого.

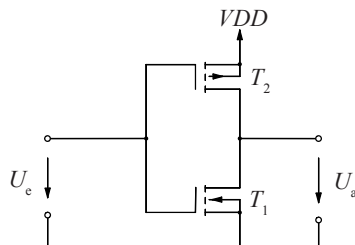


Рис. 5.19. Схема комплементарного МОП-инвертора (U_c — входное напряжение; U_a — выходное напряжение)

Модуль порогового напряжения обоих МОП-транзисторов составляет около 1,5 В, поэтому при напряжении питания 5 В по крайней мере один из них открыт.

При $U_c = 0$ открыт p -канальный МОП-транзистор T_2 , а n -канальный T_1 закрыт. Выходное напряжение становится равным VDD . При $U_c = VDD$ закрыт T_2 , а T_1 открыт и выходное напряжение принимает нулевое значение. Ясно, что в стационарных условиях ток через схему не идет.

Логические уровни зависят от выбора напряжения питания. Диапазон допустимого напряжения питания КМОП-приборов весьма широк. У схем с кремниевым затвором он лежит между 3 и 6 В, а у схем с металлическим затвором простирается от 3 до 15 В. По соображениям симметрии порог переключения всегда выбирают равным половине напряжения питания. Поэтому при питающем напряжении 5 В уровень H должен быть выше 3,5 В. По указанной причине для управления КМОП-элементом с помощью ТТЛ выхода требуется дополнительный резистор установки рабочей точки.

Выводы затвора МОП-транзисторов крайне чувствительны к статическим зарядам. Во избежание повреждений входы интегральных МОП-схем защищаются диодами (рис. 5.20), однако это не избавляет от необходимости проявлять осторожность.

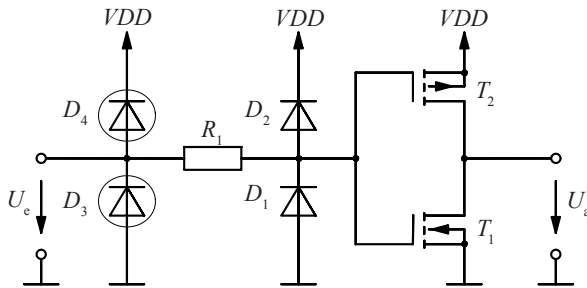


Рис. 5.20. Входная защита схемы посредством КМОП-элементов D_3 , D_4 , реализованных транзисторами T_1 , T_2

Однако с защитными диодами появляется новое ограничение, которое следует учитывать при применении КМОП-схем. Вследствие изоляции обоих МОП-транзисторов T_1 и T_2 обратным смещенным переходом возникает паразитный тиристор между выводами напряжения питания (рис. 5.21).

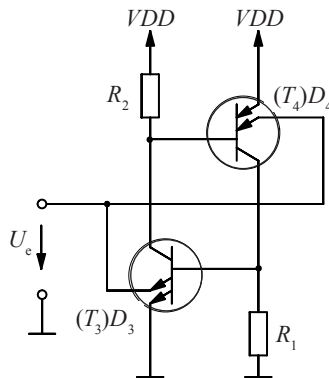


Рис. 5.21. Паразитный тиристор, возникающий вследствие изоляции МОП-транзисторов обратным смещенным переходом

В норме данный тиристор не мешает, так как транзисторы T_3 и T_4 заперты, их обратные токи протекают через резисторы R_1 и R_2 . Однако, если один из защитных диодов, действующих как дополнительные эмиттеры, включить в прямом направлении, может сработать один из тиристоров (T_3 , T_4). В результате откроются оба транзистора, питающее напряжение накоротко замкнется и возникшие при этом сильные токи разрушат интегральную схему. Во избежание подобного эффекта входные напряжения не

должны быть ниже потенциала земли или выше напряжения питания. Если такая ситуация не исключается, следует хотя бы ограничить токи, протекающие через защитные диоды, величиной 1–100 мА в зависимости от технологии схемы, как правило, с помощью гасящего резистора. Паразитный тиристор способен сработать и в случае, когда к выходу приложено напряжение, превосходящее напряжение питания.

На рисунке 5.22, *a* представлен КМОП-элемент ИЛИ–НЕ, функционирующий по тому же принципу, что и описанный инвертор. Чтобы сделать управляемый резистор высокоомным при переходе одного из входных напряжений в состояние *H*, требуется последовательная цепочка из соответствующего числа *p*-канальных полевых транзисторов. Заменяя последовательную цепочку параллельной, вместо логического элемента ИЛИ–НЕ получают элемент И–НЕ (рис. 5.22, *б*).

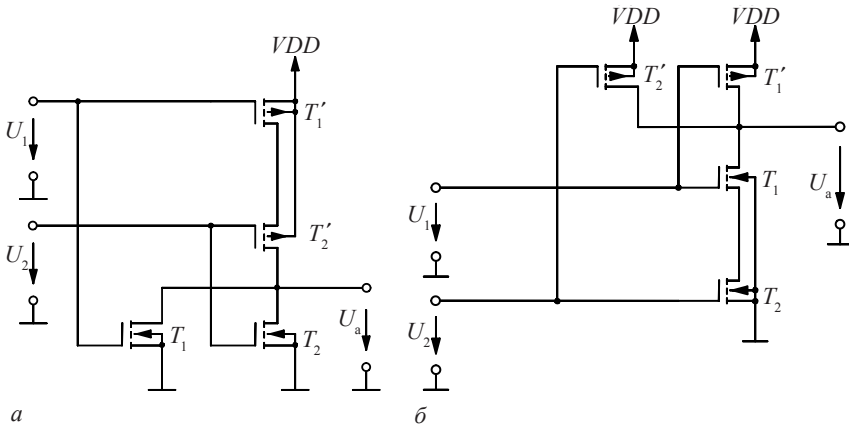


Рис. 5.22. Комплектарные МОП-элементы:
a – ИЛИ–НЕ; *б* – И–НЕ

5.11. *n*-канальная МОП-логика

Отличительная черта интегральных *n*-канальных МОП-схем состоит в том, что они строятся исключительно из *n*-канальных МОП-транзисторов. Это упрощает их изготовление и делает особенно подходящими для применения в приборах высокой степени интеграции.

Логический n -канальный МОП-элемент ИЛИ–НЕ (рис. 5.23) весьма сходен с РТЛ элементом ИЛИ–НЕ (см. рис. 5.9). По технологическим соображениям вместо омического рабочего нагрузочного резистора используют МОП-транзистор, причем работающий в режиме обогащения, как и входные полевые транзисторы. Чтобы сделать его проводящим, необходимо обеспечить высокий потенциал затвора U_{GG} . Если выходное напряжение, соответствующее состоянию H , должно подниматься до потенциала стока U_{DD} , вспомогательный потенциал U_{GG} должен превышать U_{DD} по крайней мере на величину порогового напряжения. Кроме того, обычно требуется напряжение смещения подложки U_{BB} для надежного запираания входных полевых транзисторов и снижения барьерных емкостей.

Из рисунка 5.23 видно, что транзистор T_3 работает как истоковый повторитель для U_{GG} . При этом внутреннее сопротивление желательно сделать более высокоомным, что обеспечивается путем снижения крутизны по сравнению с крутизной полевых транзисторов на входе.

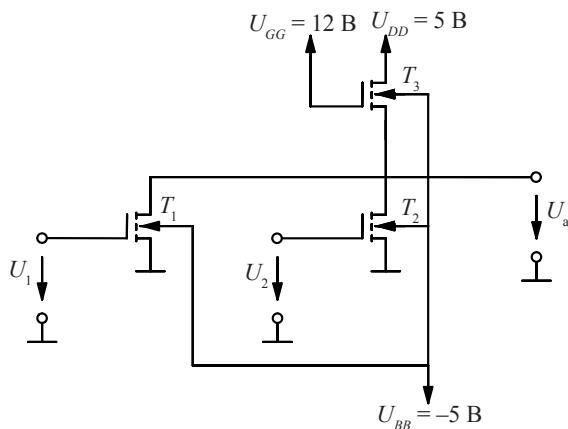


Рис. 5.23. Стандартная схема логического n -канального МОП-элемента ИЛИ–НЕ

Если в качестве T_3 применить МОП-транзистор со встроенным каналом, легко обойтись без источника напряжения U_{GG} . Такая возможность иллюстрируется схемой (рис. 5.24), в которой T_3 функционирует как источник постоянного тока. Однако полевые транзисторы на входе по-прежнему должны работать в

режиме обогащения; в противном случае управляющее напряжение пришлось бы сделать отрицательным, тогда как выходное напряжение всегда положительное. В результате прямое соединение таких логических элементов было бы невозможным.

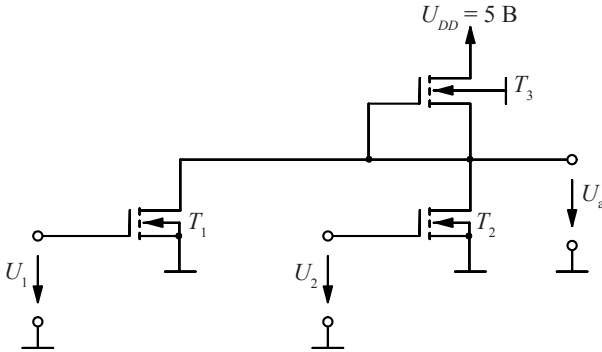


Рис. 5.24. Схема логического *n*-канального МОП-элемента ИЛИ–НЕ с обобщенной нагрузкой

Импантация ионов позволяет интегрировать МОП-транзисторы, функционирующие в режимах обеднения и обогащения. Нетрудно справиться с проблемой вспомогательного отрицательного напряжения, если правильно выбрать пороговые напряжения или получить его из положительного напряжения питания с помощью преобразователя напряжения, встроенного в микросхему.

В *n*-канальной МОП-технологии предлагаются только интегральные схемы высокой степени интеграции, а не простенькие схемы вроде одиночного логического элемента.

Признаком качества семейства интегральных микросхем служит производство мощности на время переключения. Оно свидетельствует о том, характеризуется ли элемент только малыми потерями или также обладает хорошим быстродействием.

Логические микросхемы сильно разнятся по потребляемой мощности. В этом отношении весьма удобно применять КМОП-схемы на низких частотах.

Однако в диапазоне, лежащем выше 1 МГц, различия в мощности потерь между маломощными схемами с диодами Шоттки и КМОП-элементами становятся незначительными. Следует отметить, что в данной частотной области возрастает и потребляе-

мая мощность микросхем ТТЛ. Дело в том, что через каскадный выход протекает сквозной ток при каждом переключении, в результате потребляемая мощность заметно возрастает с повышением частоты коммутации. Указанного недостатка лишены схемы серий ЭСЛ, поэтому при работе на частотах выше 30 МГц, не считая более высокой стоимости, можно говорить лишь об их преимуществах.

Контрольные вопросы и задания

1. Перечислите основные функции алгебры логики.
2. Что называют базисом в алгебре логики?
3. При каких значениях входов будет истинность элемента ИЛИ–НЕ?
4. Какое преимущество имеется у элемента медленнодействующей логики?
5. Опишите принцип работы элемента транзисторно-транзисторной логики.
6. Расскажите о назначении применения диода Шоттки в элементе транзисторно-транзисторной логики.
7. Какие преимущества дает применение в логических схемах комплементарных МОП-транзисторов?
8. Каким образом можно упростить схемы параллельного включения выходов логических элементов?

6. ЭЛЕМЕНТЫ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

6.1. Триггеры

Триггер – это устройство, имеющее два устойчивых состояния – «1» или «0», которые могут сохраняться сколь угодно долго. Переход из одного состояния в другое может происходить под воздействием управляющих сигналов. Триггер имеет два выхода (прямой Q и инверсный \bar{Q}) и один или несколько входов.

Триггеры можно классифицировать по следующим признакам.

1. По функциональному назначению: *RS*-триггеры; *D*-триггеры; *T*-триггеры; *JK*-триггеры.
2. По способу управления: асинхронные; синхронные.
3. По типу входов триггера: статические; динамические.

Асинхронные триггеры меняют свое состояние по приходу соответствующего управляющего импульса; синхронные триггеры – при наличии управляющего импульса в момент прихода синхронизирующего (тактового) импульса.

Триггеры с динамическими входами управляются либо фронтом, либо срезом управляющего сигнала, а статические – уровнем сигнала. Условное обозначение входов триггера на логических схемах показано на рисунке 6.1.

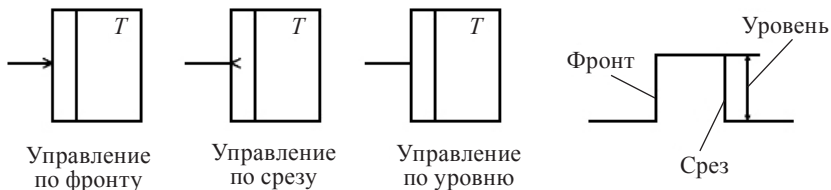


Рис. 6.1. Условное обозначение входов триггера на логических схемах

RS-триггеры. В современной электронике триггеры выполняются в виде микросхем, построенных на основе логических элементов. На рисунке 6.2 представлены схемы асинхронного и синхронного RS-триггеров на логических элементах И–НЕ и их условные обозначения.

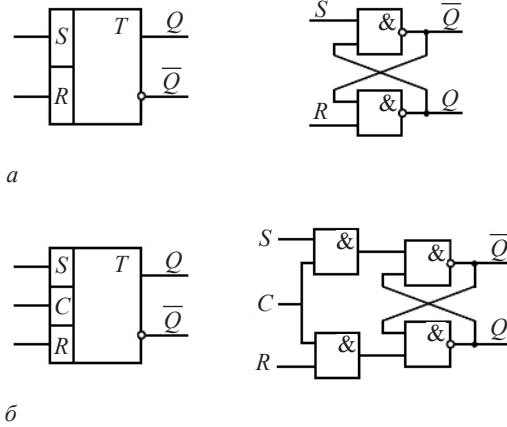


Рис. 6.2. Условные обозначения асинхронного (а) и синхронного (б) RS-триггеров и соответствующие им схемы, выполненные на логических элементах

Для синхронного варианта RS-триггер имеет три входа: S – информационный, C – синхронизирующий и R – вход сброса.

На рисунках 6.3 и 6.4 приведены таблицы истинности и временные диаграммы соответственно для асинхронного и синхронного RS-триггера.

S	R	Q	\bar{Q}
0	0	Предыдущее состояние	
0	1	1	0
1	0	0	1
1	1	Запрещенная комбинация	

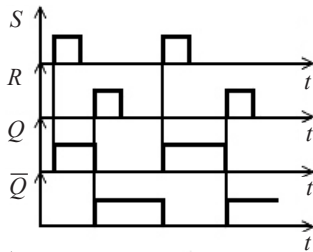
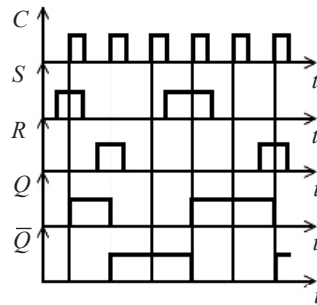


Рис. 6.3. Асинхронный RS-триггер: а – таблица истинности; б – временные диаграммы

Согласно таблице истинности (см. рис. 6.3), асинхронный RS -триггер меняет свое состояние в момент прихода управляющего импульса на соответствующий вход. Если на вход S приходит единица, триггер принимает единичное состояние и на прямом выходе Q появляется единица, а на инверсном \bar{Q} — ноль. Если единица приходит на вход R , триггер принимает нулевое состояние и на прямом выходе появляется ноль, а на инверсном — единица. В случае прихода на оба входа нулей триггер не меняет своего состояния. При приходе на входы S и R единицы одновременно триггер может принять неопределенное (неоднозначное) состояние, и такой режим нежелателен.

C	S	R	Q	\bar{Q}
0	1	0	0	1
0	0	1	0	1
1	1	0	1	0
1	0	1	0	1
1	0	0	0	1
1	1	1	Запрещенная комбинация	

а



б

Рис. 6.4. Синхронный RS -триггер:

а — таблица истинности; б — временные диаграммы

Синхронный RS -триггер (см. рис. 6.4) работает как асинхронный только тогда, когда на синхронизирующий вход C подан единичный сигнал. При его отсутствии триггер выведен из работы, т. е. сохраняет свое состояние без изменений.

D -триггеры. Для приема информации по одному входу используют D -триггеры. На рисунке 6.5 приведены условное обозначение и схема D -триггера на элементах И–НЕ. D -триггер имеет вход C для подачи тактовых импульсов и информационный вход D .

D -триггер переходит в единичное состояние $Q = 1$, если в момент прихода синхронизирующего импульса $C = 1$ на его информационном входе единичный сигнал $D = 1$. В этом состоянии триггер остается и после окончания сигнала на входе D до прихода очередного синхронизирующего импульса, возвращающего триггер в нулевое состояние. Таким образом, D -триггер «задер-

живает» поступающую на его вход информацию на время, равное периоду синхронизирующих сигналов.

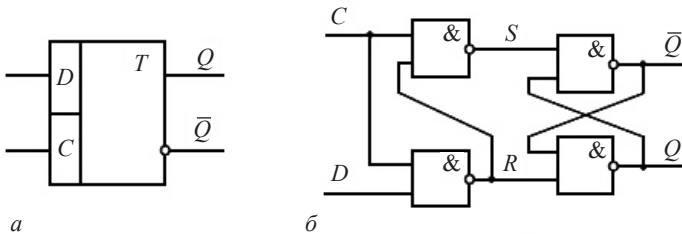


Рис. 6.5. D -триггер на элементах И–НЕ:
a – условное обозначение; *б* – схема

На рисунке 6.6 приведены таблица истинности и временные диаграммы работы динамического (по срезу синхронизирующего импульса) D -триггера.

C	D	Q	\bar{Q}
0	0	0	1
1	1	0	1
0	1	1	0
1	1	1	0
0	0	0	1
1	0	0	1

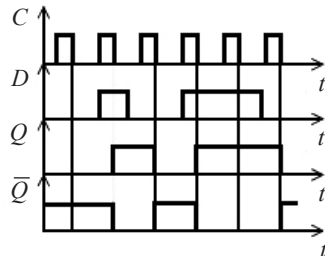


Рис. 6.6. Динамический (по срезу синхронизирующего импульса) D -триггер:
a – таблица истинности; *б* – временные диаграммы

Существует только синхронный D -триггер, в связи с чем он получил наибольшее распространение в качестве ячейки памяти для хранения однобитовой информации.

T -триггеры. Триггер T -типа, или счетный триггер, имеет один информационный вход и переходит в противоположное состояние в результате воздействия на его вход каждого очередного сигнала. Название «счетный» (или со «счетным запуском») связано с широким применением T -триггеров в счетчиках импульсов.

На рисунке 6.7 приведены условное обозначение T -триггера и схема на элементах И–НЕ.

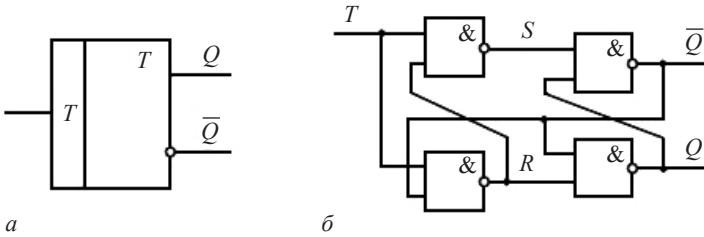


Рис. 6.7. T -триггер на элементах И–НЕ:
a – условное обозначение; *б* – схема

На рисунке 6.8 приведена таблица истинности и временные диаграммы работы динамического (по фронту информационного сигнала) T -триггера.

T	Q	\bar{Q}
0	0	1
1	1	0
0	1	0
1	0	1
0	0	1
1	1	0

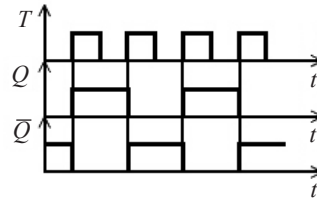


Рис. 6.8. Динамический (по фронту информационного сигнала) T -триггер: *a* – таблица истинности; *б* – временные диаграммы

JK -триггеры. Триггер JK -типа, или универсальный триггер, имеет информационные входы установки J и сброса K , подобные входам RS -триггера, а также синхронизирующий вход C (рис. 6.9, *a*). В отличие от RS -триггера, JK -триггер допускает ситуацию с одновременной подачей сигналов на оба входа J и K .

При $J = 1$ и $K = 1$ триггер меняет свое состояние на противоположное в момент окончания каждого синхронизирующего сигнала. Таким образом, соединяя входы JK -триггера по схеме (рис. 6.9, *б*), получают T -триггер.

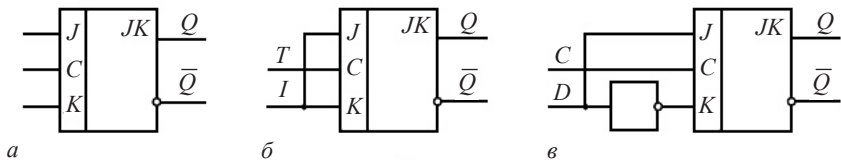


Рис. 6.9. Условное обозначение JK -триггера (а), схема T -триггера (б) и D -триггера (в) на его основе

При использовании входа J как входа S , а K как R реализуют синхронный RS -триггер, особенность которого состоит в том, что при комбинации $S = R = 1$, запрещенной для RS -триггера, он переключается на каждый синхронизирующий сигнал. Добавлением инвертора на входе JK -триггера получают D -триггер (рис. 6.9, в).

Все типы триггеров, реализуемые на базе JK -триггера, дают задержку в появлении выходного сигнала, равную длительности синхронизирующего сигнала.

6.2. Регистры

Регистром называют устройство, предназначенное для записи и хранения дискретного «слова» — двоичного числа или другой кодовой комбинации.

Основные элементы регистра — двоичные ячейки, в качестве которых применяют триггеры различных типов. Число двоичных ячеек определяют числом двоичных разрядов «слова» (длиной слова), на которое рассчитан регистр.

По способу заполнения регистра информацией различают два типа регистров: параллельный и последовательный.

Параллельный регистр. Применяют для записи и хранения информации в параллельном коде, т. е. дискретное «слова» может быть записано в регистр или считано из него за один такт.

На рисунке 6.10 приведены схема параллельного регистра на триггерах RS -типа и его условное обозначение. Перед записью данных в регистр по шине сброса подается сигнал на обнуление триггеров регистра. После этого данные в ячейки регистра записываются по команде с шины разрешения записи. Тогда сигналы n входов установят в соответствующие состояния триггеры $T_1 - T_n$.

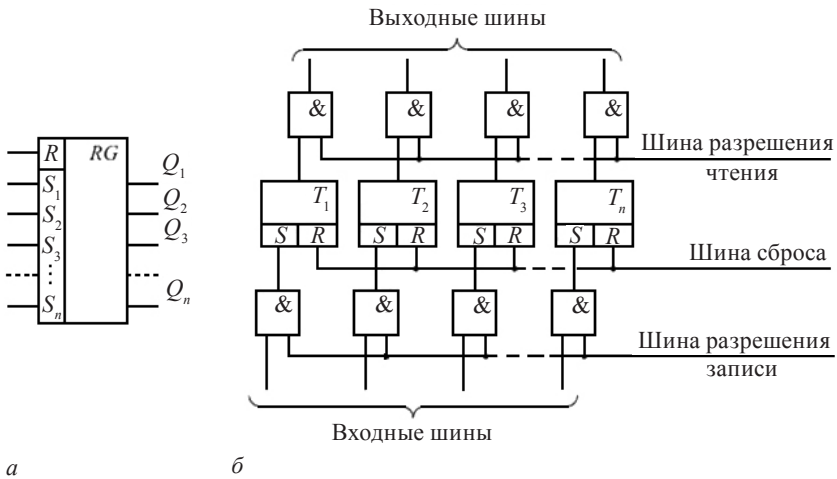


Рис. 6.10. Параллельный регистр:
a – условное обозначение; *б* – схема

На выходе регистра информация появится по команде с шины разрешения чтения, в ее отсутствие на выходах – нули. При считывании информация, записанная в регистре, сохраняется до момента записи новой информации.

Последовательный регистр. Для более экономичной передачи информации применяют последовательный код, когда используется одна линия для последовательной (во времени) передачи комбинации дискретного «слова». Для записи и хранения информации в последовательных кодах используют последовательные регистры (регистры сдвига).

На рисунке 6.11 приведена схема последовательного регистра на D -триггерах. Здесь информация, поступившая на информационный вход, по окончании каждого синхронизирующего импульса передается («сдвигается») из предыдущего триггера в следующий.

Рассмотрим работу последовательного регистра на базе D -триггеров (см. рис. 6.11). Пусть требуется записать в регистр трехразрядное двоичное слово $S = 101$, имеющее разряды $S_1 = 1$, $S_2 = 0$, $S_3 = 1$. При приходе первого тактового импульса при наличии на информационном входе первого триггера единичного сигнала триггер перейдет в единичное состояние и на его прямом

выходе появится единица. При этом остальные триггеры останутся в нулевом состоянии.

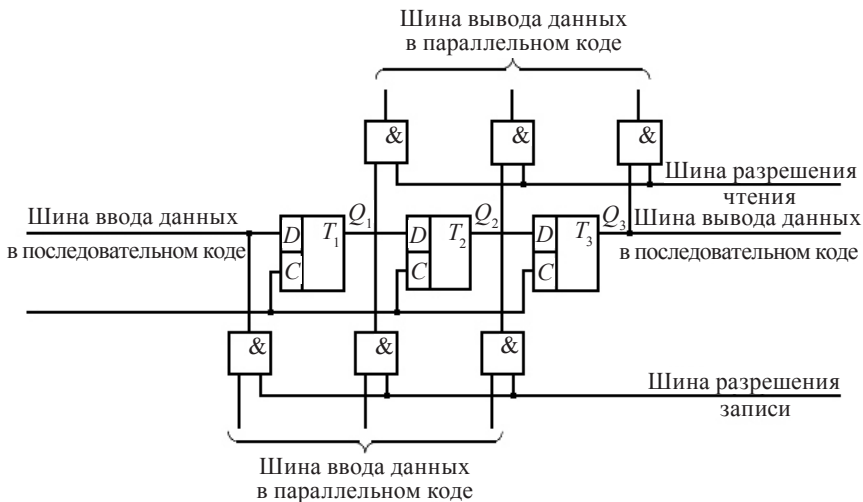


Рис. 6.11. Схема последовательного регистра

При приходе следующего тактового импульса триггеры примут состояния в соответствии с состояниями своих информационных входов, т. е. первый триггер изменит свое состояние на нулевое (так как на его информационный вход подан второй разряд двоичного слова), а триггер T_2 перейдет в единичное состояние. Таким образом, произойдет сдвиг информации из первого разряда регистра во второй.

При приходе третьего тактового импульса единичный сигнал будет на информационных входах первого и третьего триггеров, а на информационном входе триггера T_2 будет нулевой сигнал. Следовательно, триггеры T_1 и T_3 примут единичное состояние, а триггер T_2 — нулевое, и все слово будет записано в регистр.

На рисунке 6.12 приведены условное обозначение и временные диаграммы работы последовательного регистра.

Считать информацию из последовательного регистра можно либо в последовательном коде, продвигая информацию через все разряды регистра к выводу, либо в параллельном коде одновременно. Таким образом, последовательный регистр может использоваться не только для хранения информации, но и для преобразования параллельного кода в последовательный и наоборот.

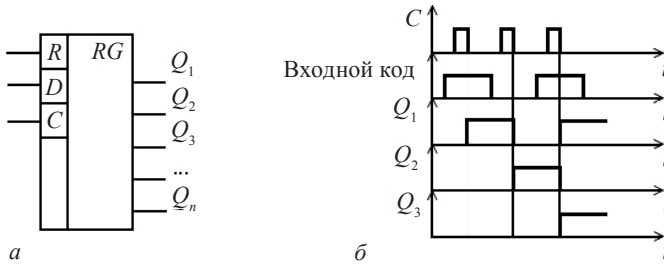


Рис. 6.12. Последовательный регистр:
 а – условное обозначение; б – временные диаграммы

6.3. Счетчики импульсов

Счетчиком импульсов называют устройство, реализующее счет числа входных импульсов и фиксирующее это число в каком-либо коде.

Обычно счетчики строят на основе триггеров, поэтому счет импульсов ведется в двоичной системе исчисления.

Функциональная схема простейшего двоичного трехразрядного счетчика импульсов представлена на рисунке 6.13. Счетчик состоит из трех последовательно соединенных T -триггеров, имеющих вход R для установки в нулевое состояние.

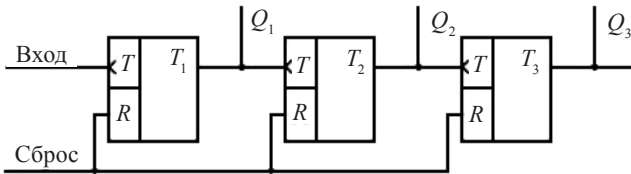


Рис. 6.13. Функциональная схема трехразрядного двоичного счетчика

На рисунке 6.14 показаны временные диаграммы счетчика и его обозначение на логических схемах. Счетчик импульсов CT_2 означает двоичный счетчик; выходы 1, 2, 3 обозначают двоичные разряды ($2^0 = 1$, $2^1 = 2$, $2^2 = 4$), соответствующие выходам Q_1 , Q_2 , Q_3 схемы, изображенной на рисунке 6.13; C_1 – счетный вход; R – установка нуля.

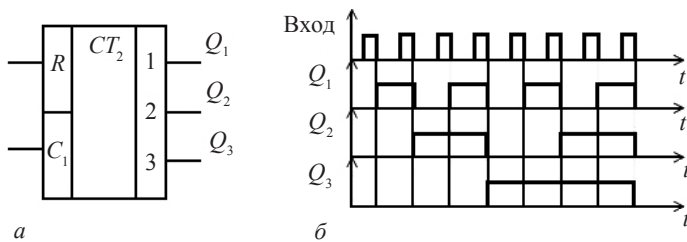


Рис. 6.14. Трехразрядный двоичный счетчик:
 а – условное обозначение; б – временные диаграммы

Таблица 6.1 иллюстрирует состояние триггеров. Если в исходном положении все триггеры будут в нулевом состоянии, то по окончании первого входного импульса триггер T_1 перейдет в единичное состояние ($Q_1 = 1$). По окончании второго входного импульса триггер T_1 переходит в нулевое состояние ($Q_1 = 0$). По окончании импульса Q_1 триггер T_2 переходит в единичное состояние ($Q_2 = 1$) и т. д. После восьмого входного импульса все триггеры переходят в нулевое состояние, и счет повторяется.

Таблица 6.1

Состояние триггеров

Такт	Вход	Q_1	Q_2	Q_3	Такт	Вход	Q_1	Q_2	Q_3
1	1	0	0	0	6	1	1	0	1
	0	1	0	0		0	0	0	1
2	1	1	0	0	7	1	0	1	1
	0	0	1	0		0	1	1	1
3	1	0	1	0	8	1	1	1	1
	0	1	1	0		0	0	0	0
4	1	1	1	0	9	1	0	0	0
	0	0	0	1		0	1	0	0
5	1	0	0	1	10	1	1	0	0
	0	1	0	1		0	0	1	0

Из таблицы 6.1 следует, что состояние триггеров отражает число поступивших на вход счетчика импульсов в двоичной системе исчисления (двоичном коде). Общее число возможных состояний N счетчика определяют числом триггеров n : $N = 2^n$. В нашем случае $N = 8$.

6.4. Дешифраторы

Дешифратором (декодером) называют устройство, предназначенное для распознавания различных кодовых комбинаций (слов).

Каждому слову на входе дешифратора соответствует логическая единица на одном из его выходов. На рисунке 6.15 приведены обозначение двухразрядного дешифратора и его функциональная схема, выполненная на элементах И и НЕ, объединенных соответствующими логическими связями.

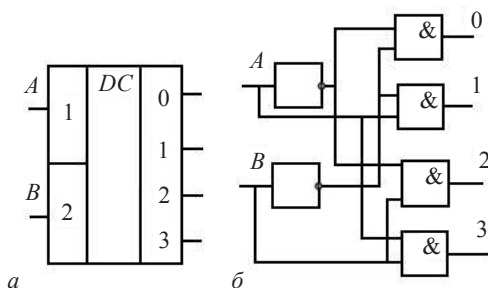


Рис. 6.15. Двухразрядный дешифратор:
а – условное обозначение; б – схема

Логическое состояние выходных каналов в зависимости от кода, подаваемого на дешифратор, приведено в таблице 6.2.

Таблица 6.2

Логическое состояние выходных каналов дешифратора

Входной код		Состояние выходов			
<i>B</i>	<i>A</i>	0	1	2	3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Если на вход *A* подать единицу, а на вход *B* – нуль (код входного сигнала 10), то на обоих входах элемента И под номером 1 будут одновременно единичные сигналы. Поэтому на выходе только этого элемента будет единичный сигнал.

Все остальные выходные сигналы будут обнулены, так как на входах их элементов И будет присутствовать хотя бы по одному нулевому входному сигналу.

Дешифраторы находят разнообразное применение в информационно-измерительной и вычислительной технике. Одно из них – управление световой индикацией. На рисунке 6.16, *а* представлена схема счета и отображения числа импульсов. Она состоит из двоичного счетчика CT_2 , который представляет число поступивших на его вход импульсов в двоичном коде, двоичного семисегментного дешифратора DC , управляющего транзисторными ключами T_1-T_7 , и светодиодного сегментированного индикатора D_1-D_7 . Условное обозначение сегментов семисегментного индикатора показано на рисунке 6.16, *б*.

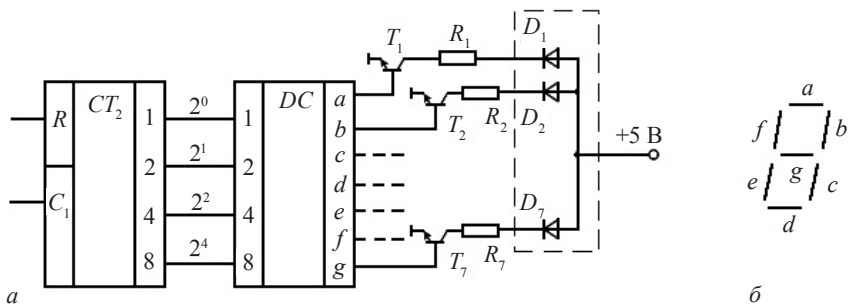


Рис. 6.16. Схема счета и отображения числа импульсов (*а*) и обозначение сегментов (*б*) семисегментного индикатора

Таблица 6.3 иллюстрирует порядок функционирования двоичного семисегментного дешифратора.

Таблица 6.3

Логическое состояние выходных каналов счетчика импульсов

Цифра	Двоичный код				Семисегментный выход						
	2^3	2^2	2^1	2^0	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	<i>g</i>
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1

6.5. Элементы памяти

Под *элементами памяти* подразумевают простейшие схемы и физические среды, предназначенные для приема, хранения и выдачи 1 бита информации. Память микропроцессорной системы – функциональная часть, используемая для хранения обрабатываемых данных, запоминания результатов выполнения операций или отдельных команд и т. п. Запоминающие устройства (ЗУ) представляют собой комплекс технических средств, предназначенных для приема, хранения и выдачи информации, представленной двоичным словом. Этот комплекс включает в себя средства адресации информации, накопитель информации и устройство управления. В микропроцессорных системах используют исключительно полупроводниковую память.

Для хранения программ, которые обеспечивают работу микропроцессорной системы, и некоторых данных многократного применения требуется память, позволяющая только считывать информацию. Такое ЗУ называют *памятью только для чтения* или *постоянным запоминающим устройством* (ПЗУ). Содержимое ПЗУ не может быть стерто. Оно используется как память программы, составленной изготовителем в соответствии с требованиями ее пользователей. В таких случаях говорят, что программа жестко «защита» в запоминающем устройстве. Помимо ПЗУ, используются также программируемое постоянное запоминающее устройство и репрограммируемое постоянное запоминающее устройство.

Программируемое постоянное запоминающее устройство (ППЗУ) отличается от ПЗУ тем, что пользователь может самостоятельно запрограммировать ПЗУ (ввести в него программу) с помощью специального устройства – программатора, но только один раз. *Репрограммируемое постоянное запоминающее устройство* (РППЗУ), называемое также стираемым ПЗУ, допускает перепрограммирование с помощью программатора. Предыдущая информация при этом стирается.

Существует большое количество различных типов ПЗУ. На рисунке 6.17 показаны условное графическое обозначение электрически программируемого ПЗУ емкостью 256 бит с организацией 32×8 (K155PE3), фрагмент схемы и временные диаграммы режима программирования.

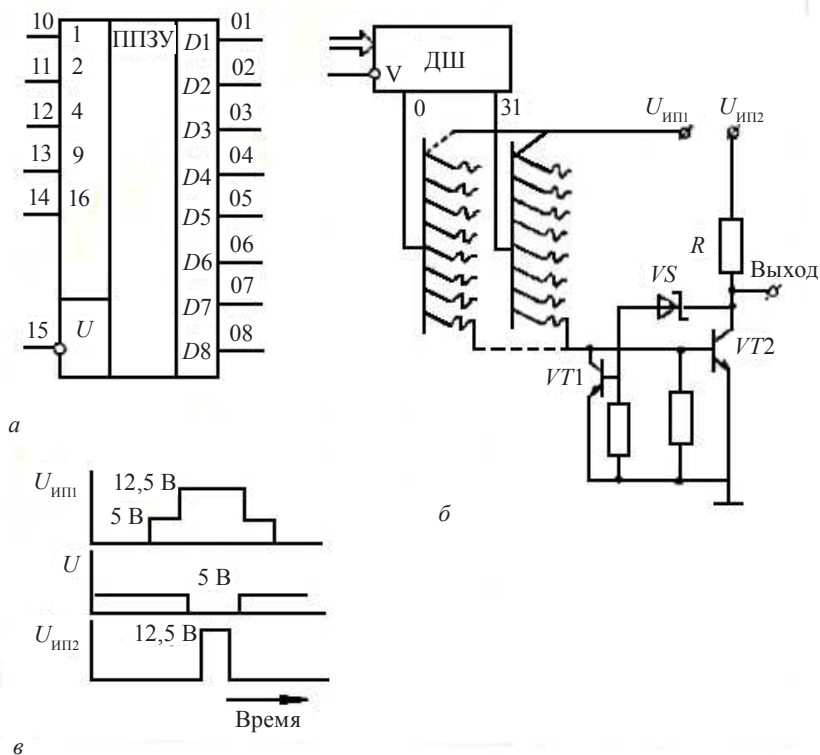


Рис. 6.17. Постоянное запоминающее устройство

Технология программирования включает следующие этапы. Первоначально во всех ячейках ППЗУ записаны 0. Для записи 1 на адресные входы необходимо подать требуемый адрес, установить $U_{ин1} = 12,5$ В, на вход выбора кристалла подать 0, а на резистор $R = 390$ Ом напряжение $U_{ин2} = 12,5$ В. В этом случае пробивается стабилитрон VS , открывается транзистор $VT1$ и пережигается нихромовая перемычка у соответствующего эмиттера многоэмиттерного транзистора. В одном цикле осуществляется программирование только одного разряда. Выпускают ППЗУ с организацией 256×4 ; 512×8 ; 2048×8 , для программирования которых разработаны специальные программаторы.

Программируемая логическая матрица (ПЛМ) – матрица логических элементов, связи между которыми могут задавать-

ся пользователем в целях реализации определенных логических функций. В отличие от ПЗУ, в которых с помощью адресных дешифраторов обеспечивается обращение к одному из n слов памяти, в ПЛМ реализуется обращение к требуемому числу слов. Это связано с тем, что часто не требуется дешифрация всех возможных комбинаций, поступающих по адресным входам. В отличие от ПЗУ, в ПЛМ программируются не только данные, но и адреса.

Оперативное запоминающее устройство (ОЗУ), называемое также запоминающим устройством с произвольной выборкой (ЗУПВ), служит памятью данных, подлежащих обработке, и программ. Время доступа к любой ячейке ОЗУ не зависит от адреса этой ячейки. Информацию в ОЗУ можно многократно записывать и считывать из нее. В отличие от ПЗУ, информация в ОЗУ стирается автоматически, если прерывается напряжение питания. Таким образом, важнейшим недостатком ОЗУ перед ПЗУ является то, что оно энергозависимо, а достоинством — возможность многократной записи и считывания.

Контрольные вопросы и задания

1. Что представляет собой логический элемент?
2. Расскажите, какие операции включает в себя логическое преобразование двоичных сигналов.
3. Что означает понятие «триггер»?
4. Приведите классификацию триггеров.
5. Поясните работу: а) RS -триггера; б) D -триггера; в) T -триггера; г) JK -триггера.
6. Дайте определение регистрам. Какие виды регистров бывают?
7. Расскажите о логике работы счетчика импульсов.
8. Для чего применяют дешифраторы?
9. Опишите виды памяти.

7. АРИФМЕТИЧЕСКИЕ И ЛОГИЧЕСКИЕ ОСНОВЫ ЭВМ

7.1. Системы счисления, их назначение и виды

Система счисления — это знаковая система, в которой числа записываются по определенным правилам, с помощью символов некоторого алфавита, называемых цифрами.

Самая простая система счисления — *единичная*, или *унарная*. В ней используется только один символ: палочка, камушек и т. д.

Такая система счисления использовалась в основном народами, не имеющими письменности, примерно 10–11 тыс. лет до н. э. Однако и сейчас такой системой счисления пользуются, отмечая, например, зарубками количество прошедших дней.

В зависимости от того, меняется ли значение цифры от ее положения в числе, выделяют две системы счисления: позиционную и непозиционную.

В *позиционных системах* значение цифры меняется вместе с ее положением в числе. Так, если взять число 234, то цифра 4 в ней означает единицы, если же рассмотреть число 243, то здесь она уже будет означать десятки, а не единицы.

В *непозиционных системах* значение цифры статично, вне зависимости от ее положения в числе. Наиболее яркий пример — *палочковая система*, где каждая единица обозначается с помощью черточки. Независимо, куда вы припишете палочку, значение числа изменится лишь на единицу.

К позиционным системам счисления относятся десятичная, двоичная, шестидесятеричная и др. Название позиционной системы счисления зависит от того, сколько символов используется для записи чисел.

Основанием позиционной системы счисления называют количество символов, используемых для записи чисел. Например, в *двоичной* системе счисления используют две цифры — 0 и 1; основание ее равно 2. Изначально именно с помощью двоичного кода компьютеры получали всю необходимую информацию. В *восьмеричной* системе счисления восемь цифр (0, 1, ..., 7); основание — 8. Применялась в основном в тех областях знаний, которые связаны с цифровыми устройствами. Однако в последнее время она употребляется значительно реже, так как на смену ей пришла *шестнадцатеричная система счисления*.

В системах счисления с основанием больше 10 для представления чисел после цифр 0, 1, 2, ..., 9 используют латинские буквы: А (10), В (11), С (12), D (13), Е (14), F (15). Например, алфавит шестнадцатеричной системы счисления выглядит следующим образом: 0, 1, 2, ..., 9, А, В, С, D, Е, F. Основание этой системы счисления — 16.

Перевод чисел из десятичной в двоичную систему счисления.

Перевод чисел в системах счисления происходит по определенным правилам. Чаще осуществляется перевод из двоичной в десятичную систему и наоборот. Для того чтобы перевести число из десятичной системы в двоичную, необходимо последовательно делить его на основание системы счисления, т. е. число *два*. При этом остаток от каждого деления необходимо фиксировать. Так будет происходить до тех пор, пока остаток от деления не будет меньше или равен единице. Проводить вычисления лучше всего в столбик. Затем полученные остатки от деления записывают в строку в обратном порядке.

Например, переведем число 7 в двоичную систему. Делим 7 на 2. Поскольку число не делится нацело, берем число 6, остаток будет $7 - 6 = 1$. Делим 3 на 2, так как число не делится нацело, берем число 2, остаток будет $3 - 2 = 1$. В итоге деления у нас получается 1. Далее записываем все полученные нами остатки в обратном порядке, начиная с итога деления: 111. Вне зависимости от итоговой системы счисления перевод чисел из десятичной в любую другую будет происходить по принципу деления числа на основу позиционной системы.

Перевод чисел из двоичной в десятичную систему счисления.

Очень просто переводить числа и в десятичную систему счисле-

$$\begin{array}{r} 7 \quad | \quad 2 \\ \hline 6 \quad | \quad 3 \quad | \quad 2 \\ \hline 1 \quad | \quad 2 \quad | \quad 1 \\ \hline \quad \quad \quad 1 \end{array}$$

ния из двоичной. Для этого достаточно знать правила возведения чисел в степень, в данном случае в степень двойки. Последовательность перевода следующая: каждую цифру из кода двоичного числа необходимо умножить на двойку, причем первая двойка будет в степени $m - 1$, вторая $m - 2$ и так далее, где m — количество цифр в коде. Затем сложить результаты умножения, получив целое число.

Для примера разберем полученное ранее число 111, переводя его в десятичную систему, и заодно проверим правильность наших вычислений. Выглядеть это будет следующим образом: $1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 4 + 2 + 1 = 7$.

Перевод десятичных чисел в восьмеричную систему счисления. Переводим 126_{10} в восьмеричную систему счисления. Для этого данное число делим на восемь:

$$\begin{array}{r} 126 \overline{)8} \\ -120 \overline{)15} \overline{)8} \\ \hline 6 \quad \underline{8} \overline{)1} \\ \quad \quad \quad 7 \end{array} \quad \begin{array}{l} 126 / 8 = 15 - \text{остаток } 6; \\ 15 / 8 = 1 - \text{остаток } 7; \\ 7 / 8 = 0 - \text{остаток } 1. \end{array}$$

Записав цифры всех остатков в обратном порядке, получим число 176_8 .

Таким образом, $126_{10} = 176_8$.

Для перевода в десятичную систему необходимо каждую цифру (начиная справа) восьмеричного числа умножить на 8 в степени, соответствующей разряду этой цифры.

Перевод десятичных чисел в шестнадцатеричную систему счисления. Как правило, перевод десятичных чисел в шестнадцатеричную систему вызывает наибольшее затруднение, поэтому возьмем более сложный пример.

Переводим 46877_{10} в шестнадцатеричную систему счисления:

$$46877 / 16 = 2929 - \text{остаток } 13 = D;$$

$$2929 / 16 = 183 - \text{остаток } 1;$$

$$183 / 16 = 11 - \text{остаток } 7;$$

$$11 = B.$$

Записав цифры всех остатков в обратном порядке, получим число $B71D_{16}$.

Таким образом, $46877_{10} = B71D_{16}$.

Для перевода в десятичную систему необходимо каждую цифру (начиная справа) шестнадцатеричного числа умножить на 16 в степени, соответствующей разряду этой цифры.

Представление чисел с фиксированной и плавающей запятой.

В ЭВМ применяются две формы представления двоичных чисел:

- естественная, или форма с фиксированной запятой (точкой);
- нормальная, или форма с плавающей запятой (точкой).

В форме с **фиксированной запятой** все числа изображаются в виде последовательности цифр с постоянным для всех чисел положением запятой, отделяющей целую часть от дробной.

Например, в десятичной системе счисления имеется 5 разрядов в целой части числа (до запятой) и 5 разрядов в дробной части числа (после запятой). Числа, записанные в такую разрядную сетку, имеют следующий вид:

$$+00375,35500; +00000,00638; -12301,53262.$$

Эта форма наиболее простая, естественная, но имеет небольшой диапазон представления чисел, поэтому чаще всего не применяется при вычислениях.

Если в результате операции получится число, выходящее за допустимый диапазон, происходит переполнение разрядной сетки и дальнейшие вычисления теряют смысл. В современных ЭВМ естественная форма представления используется как вспомогательная и только для целых чисел.

В форме представления с **плавающей запятой** каждое число изображается в виде двух групп цифр. Первая группа цифр называется *мантиссой*, вторая — *порядком*, причем абсолютная величина мантиссы должна быть меньше 1, а порядок — целым числом. В общем виде число в форме с плавающей запятой может быть представлено так:

$$N = \pm MP^{\pm r},$$

где M — мантисса числа ($|M| < 1$); P — основание системы счисления; r — порядок числа (r — целое число).

Например, приведенные ранее числа в нормальной форме записываются так:

$$+0,541395 \cdot 10^3; +0,623 \cdot 10^{-3}; -0,123212026 \cdot 10^5.$$

Нормальная форма представления имеет огромный диапазон отображения чисел и является основной в современных компьютерах.

Следует заметить, что все числа с плавающей запятой хранятся в компьютере в нормализованном виде. *Нормализованным*

называют такое число, в старшем разряде мантиссы которого стоит единица.

Форматы данных. Формат данных связан с разрядностью ячеек памяти и регистров, в которых хранятся данные. Он определяет возможную длину представления данных в процессоре.

Различают следующие основные форматы представления данных: байт; полуслово; слово; двойное слово; расширенное слово.

Основными из этих форматов являются слово, двойное слово и расширенное слово.

Слово отображает содержимое одной из n -разрядных ячеек памяти или одного из n -разрядных регистров, поэтому слово обычно характеризует внешнее представление исходных и конечных данных.

Длина слова n измеряется в битах и по величине равна разрядности соответствующей ячейки памяти данных или регистра.

Двойное слово отображает содержимое пары соседних n -разрядных ячеек памяти, одного $2n$ -разрядного регистра или пары соседних n -разрядных регистров. Поэтому двойное слово характеризует представление данных с удвоенной точностью; длина двойного слова равна $2n$ битам.

Расширенное слово отображает содержимое k -разрядного аккумулятора или выходного регистра и характеризует внутреннее представление данных — результатов промежуточных и конечных вычислений.

Длина расширенного слова равна k битам; в зависимости от архитектуры процессора, формы представления данных и назначения регистра для величины k выполняется условие: $k > 2n$ (в процессорах с фиксированной точкой) или $n < k < 2n$ (в процессорах с плавающей точкой). Длина расширенного слова всегда больше длины слова, что позволяет повысить точность промежуточных и конечных вычислений.

Содержимое аккумулятора отображается словом аккумулятора, которое в зависимости от архитектуры процессора имеет формат двойного и расширенного слова.

Байт и **полуслово** отображают содержимое соответствующих частей ячейки памяти данных или регистров.

Основные форматы (слово, двойное слово и расширенное слово) показаны на рисунке 7.1.

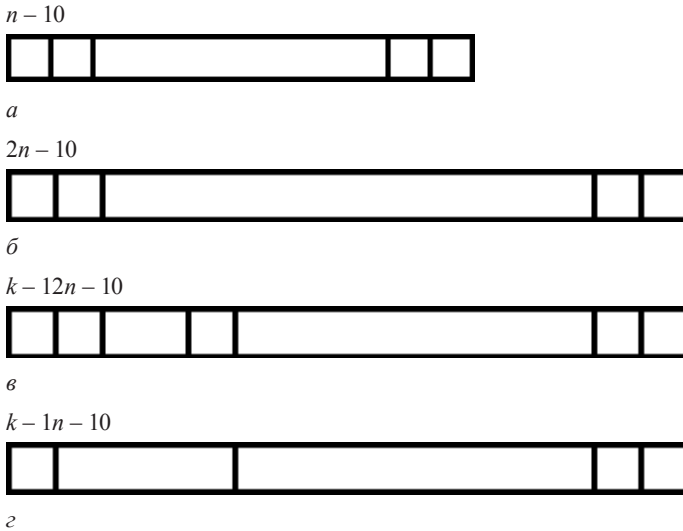


Рис. 7.1. Основные форматы данных:
a – слово; *b* – двойное слово; *v* – расширенное слово $k > 2n$;
z – расширенное слово $n < k < 2n$

Формат является важнейшей характеристикой представления данных. Базовым форматом в процессоре считается *слово*, его длина определяет диапазон и точность представления данных, объем памяти, разрядность шины данных и т. д.

7.2. Кодирование информации

Для представления информации могут использоваться разные коды, соответственно надо знать определенные правила – законы записи этих кодов, т. е. уметь кодировать.

Код – набор условных обозначений для представления информации.

Кодирование – процесс представления информации в виде кода.

В качестве *элементов кодируемой информации* могут выступать:

- буквы, слова и фразы естественного языка;
- различные символы, такие как знаки препинания, арифметические и логические операции, операторы сравнения и т. д. Следует отметить, что сами знаки операций и операторы сравнения – это кодовые обозначения;
- числа;

- аудиовизуальные образы;
- ситуации и явления;
- наследственная информация и т. д.

Кодовые обозначения могут представлять собой:

- буквы и сочетания букв естественного языка;
- числа;
- графические обозначения;
- электромагнитные импульсы;
- световые и звуковые сигналы;
- набор и сочетание химических молекул и т. д.

Кодирование может выполняться в целях:

- удобства хранения, обработки и передачи информации (как правило, закодированная информация представляется более компактно, а также пригодна для обработки и передачи автоматическими программно-техническими средствами);

- удобства информационного обмена между субъектами;
- наглядности отображения;
- идентификации объектов и субъектов;
- сокрытия секретной информации и т. д.

Одна и та же информация может быть представлена (закодирована) в нескольких формах. С появлением компьютеров возникла необходимость кодирования всех видов информации, с которыми имеют дело и отдельный человек, и человечество в целом. Однако решать задачу кодирования информации человечество начало задолго до появления компьютеров. Грандиозные достижения человечества — письменность и арифметика — есть не что иное, как система кодирования речи и числовой информации. Информация никогда не появляется в чистом виде, она всегда как-то представлена, как-то закодирована.

Двоичное кодирование — один из распространенных способов представления информации. В вычислительных машинах, в роботах и станках с числовым программным управлением, как правило, вся информация, с которой имеет дело устройство, кодируется в виде слов двоичного алфавита.

Кодирование символьной информации. Это основная операция, производимая над отдельными символами текста — сравнение символов.

При сравнении символов наиболее важными аспектами являются уникальность кода для каждого символа и длина этого

кода, а сам выбор принципа кодирования практически не имеет значения.

Для кодирования текстов используют различные таблицы перекодировки. Важно, чтобы при кодировании и декодировании одного и того же текста использовалась одна и та же таблица.

Таблица перекодировки — таблица, содержащая упорядоченный некоторым образом перечень кодируемых символов, в соответствии с которой происходит преобразование символа в его двоичный код и обратно.

Наиболее популярные таблицы перекодировки: ДКОИ-8, ASCII, CP1251, Unicode.

Исторически сложилось, что в качестве длины кода для кодирования символов было выбрано 8 бит, или 1 байт. Поэтому чаще одному символу текста, хранимому в компьютере, соответствует 1 байт памяти.

Различных комбинаций из 0 и 1 при длине кода 8 бит может быть $2^8 = 256$, поэтому с помощью одной таблицы перекодировки можно закодировать не более 256 символов. При длине кода в 2 байта (16 бит) можно закодировать 65 536 символов.

Кодирование числовой информации. Сходство в кодировании числовой и текстовой информации состоит в следующем: чтобы можно было сравнивать данные этого типа, у разных чисел (как и у разных символов) должен быть различный код. Основное отличие числовых данных от символьных заключается в том, что над числами, кроме операции сравнения, производятся разнообразные математические операции: сложение, умножение, извлечение корня, вычисление логарифма и пр. Правила выполнения этих операций в математике подробно разработаны для чисел, представленных в позиционной системе счисления.

Основной системой счисления для представления чисел в компьютере является двоичная позиционная система счисления.

Прямые, обратные и дополнительные коды. При выполнении арифметических операций в ЭВМ применяют специальные коды для представления чисел (в целях упрощения арифметических операций): прямой, обратный и дополнительный коды чисел.

Прямой код (представление в виде абсолютной величины со знаком) двоичного числа — это само двоичное число, в котором все цифры, изображающие его значение, записываются как в математической записи, а знак числа — двоичной цифрой.

Прямой код используют при хранении чисел в памяти ЭВМ, а также при выполнении операций умножения и деления. Однако формат представления чисел в прямом коде неудобен для использования в вычислениях, поскольку сложение и вычитание положительных и отрицательных чисел выполняют по-разному, поэтому требуется анализировать знаковые разряды операндов.

Обратный код положительного числа совпадает с прямым, а при записи отрицательного числа все его цифры, кроме цифры, изображающей знак числа, заменяются на противоположные (0 заменяется на 1, а 1 на 0).

Дополнительный код (представление в виде дополнения до двойки) положительного числа совпадает с прямым, а код отрицательного числа образуется как результат увеличения на 1 его обратного кода. Иными словами, процесс построения дополнительного кода отрицательного числа можно разбить на два этапа: построить обратный код, а затем из него построить дополнительный.

Основным достоинством дополнительного кода является то, что в нем единообразно реализуются операции сложения чисел разных знаков, а операцию вычитания можно свести к операции сложения заменой знака вычитаемого на обратный, причем для реализации дополнительного кода не требуется никаких дополнительных аппаратных устройств.

Модифицированные коды. МПК – модифицированный прямой код; МОК – модифицированный обратный код; МДК – модифицированный дополнительный код.

Эти коды отличаются от прямого, обратного и дополнительного кодов тем, что на изображение знака отводятся два разряда: если число положительное – 00, если число отрицательное – 11. Такие коды удобны для выявления переполнения разрядной сетки. Если знаковые разряды результата принимают значения 00 и 11, то переполнения разрядной сетки не было, а если 01 или 10, то было переполнение.

Кодирование текстовой информации. В настоящее время большая часть пользователей при помощи компьютера обрабатывает текстовую информацию, которая состоит из символов: букв, цифр, знаков препинания и др. Подсчитаем, сколько всего символов и какое количество бит нам нужно.

Например: 10 цифр, 12 знаков препинания, 15 знаков арифметических действий, буквы русского и латинского алфавита. Всего: 155 символов, что соответствует 8 битам информации.

Суть кодирования заключается в том, что каждому символу ставят в соответствие двоичный код от 00000000 до 11111111 или соответствующий ему десятичный код от 0 до 255.

Необходимо помнить, что в настоящее время для кодирования русских букв используют пять различных кодовых таблиц (КОИ-8, CP1251, CP866, Mac, ISO), причем тексты, закодированные при помощи одной таблицы, не будут правильно отображаться в другой.

Основным отображением кодирования символов является код ASCII (American Standard Code for Information Interchange – американский стандартный код обмена информацией), который представляет собой таблицу 16 на 16, где символы закодированы в шестнадцатеричной системе счисления.

Кодирование графической информации. Важным этапом кодирования графического изображения является разбиение его на дискретные элементы (дискретизация).

Основные способы представления графики для ее хранения и обработки с помощью компьютера – векторные и растровые изображения.

Векторное изображение представляет собой графический объект, состоящий из элементарных геометрических фигур (чаще всего отрезков и дуг). Положение этих элементарных отрезков определяется координатами точек и величиной радиуса. Для каждой линии указываются двоичные коды в виде линии (сплошная, пунктирная, штрихпунктирная), толщины и цвета.

Растровое изображение представляет собой совокупность точек (пикселей), полученных в результате дискретизации изображения в соответствии с матричным принципом.

Матричный принцип кодирования графических изображений заключается в том, что изображение разбивается на заданное количество строк и столбцов. Затем каждый элемент полученной сетки кодируется по выбранному правилу.

Pixel (picture element – элемент рисунка) – минимальная единица изображения, цвет и яркость которой можно задать независимо от остального изображения.

В соответствии с матричным принципом строятся изображения, выводимые на принтер, отображаемые на экране дисплея, получаемые с помощью сканера.

Качество изображения будет тем выше, чем «плотнее» расположены пиксели, т. е. чем больше разрешающая способность устройства и чем точнее закодирован цвет каждого из них.

Для черно-белого изображения код цвета каждого пикселя задается 1 битом.

Если рисунок цветной, для каждой точки задается двоичный код ее цвета.

Поскольку и цвета кодируются в двоичном коде, то если, например, вы хотите использовать 16-цветный рисунок, для кодирования каждого пикселя потребуется 4 бита ($16 = 2^4$), а если есть возможность использовать 16 битов (2 байта) для кодирования цвета одного пикселя, тогда вы можете передать $2^{16} = 65\,536$ различных цветов. Использование 3 байтов (24 битов) для кодирования цвета одной точки позволяет отразить $2^{24} = 16\,777\,216$ (или около 17 млн) различных оттенков цвета, так называемый режим «истинного цвета» (true color).

Цветовая модель RGB используется при описании цветов, получаемых смешением световых лучей. Ее составляющими являются три базовых цвета: Red (красный), Green (зеленый) и Blue (синий). Все остальные цвета в этой модели образуются за счет смешения базовых цветов в различных пропорциях. Каждый из трех базовых цветов может варьировать в пределах 0–255, образуя разные цвета и обеспечивая таким образом доступ ко всем другим цветам. Для их хранения необходимо 3 байта – 24 бита (цветовая разрешающая способность аддитивной модели составляет 24 bpp (bits per pixel – бит на пиксел)).

Каждому цвету RGB-модели можно сопоставить код, который содержит яркости его трех составляющих. Для записи кода используют десятичное (тройка десятичных чисел, разделенных запятой) или шестнадцатеричное (#XXXXXX; # – признак шестнадцатеричного числа, далее каждые две цифры – шестнадцатеричное число яркости каждой составляющей) представление кода. Например, код желтого цвета можно записать как 255,255,0 или #FFFF00.

На экране монитора RGB-схема – это свечение трех зерен триады люминофора (красного, зеленого и синего). Для получения белого цвета необходимо, чтобы ярко светились все три зерна триады – красное, зеленое и синее (R:255/G:255/B:255). Отсутствие свечения дает черный цвет (R:0/G:0/B:0).

Кодирование звуковой информации. Из курса физики известно, что звук — это колебания воздуха. По своей природе звук является непрерывным сигналом. Если преобразовать звук в электрический сигнал (например, с помощью микрофона), то увидим плавно изменяющееся со временем напряжение.

Этот процесс называют *дискретизацией* (или *оцифровкой*), а устройство, выполняющее его, — *аналого-цифровым преобразователем* (АЦП).

Чтобы воспроизвести закодированный таким образом звук, необходимо сделать обратное преобразование (для этого служит *цифроаналоговый преобразователь* — ЦАП), а затем сгладить получившийся ступенчатый сигнал.

Чем выше частота дискретизации и чем больше разрядов отводится для каждого отсчета, тем точнее будет представлен звук, но при этом увеличивается и размер звукового файла. Поэтому в зависимости от характера звука, требований, предъявляемых к его качеству и объему занимаемой памяти, выбирают некоторые компромиссные значения.

Параметры дискретизации. Важными параметрами дискретизации являются разрядность и частота.

Разрядность указывает, с какой точностью происходят изменения амплитуды аналогового сигнала. Точность, с которой при оцифровке передается значение амплитуды сигнала в каждый из моментов времени, определяет качество сигнала после цифроаналогового преобразования. Именно от разрядности зависит достоверность восстановления формы волны.

Для кодирования значения амплитуды используют принцип двоичного кодирования. Звуковой сигнал должен быть представлен в виде последовательности электрических импульсов (двоичных нулей и единиц). Обычно используют 8-, 16-битное или 20-битное представление значений амплитуды. При двоичном кодировании непрерывный звуковой сигнал заменяют последовательностью дискретных уровней сигнала. Например (рис. 7.2):

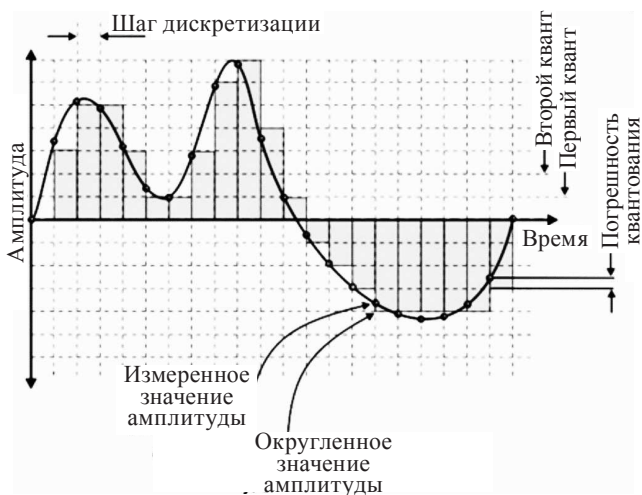


Рис. 7.2. Пример кодирования звукового сигнала

Частота дискретизации — количество измерений амплитуды аналогового сигнала в секунду.

От частоты дискретизации (количества измерений уровня сигнала в единицу времени) зависит качество кодирования. С увеличением частоты дискретизации увеличивается точность двоичного представления информации. При частоте 8 кГц (количество измерений в секунду 8000) качество оцифрованного звукового сигнала соответствует качеству радиотрансляции, а при частоте 48 кГц (количество измерений в секунду 48 000) — качеству звучания аудио.

Квантование — это процедура преобразования непрерывного диапазона всех возможных входных значений измеряемой величины в дискретный набор выходных значений.

Обычно при квантовании диапазон возможных значений измеряемой величины разбивается на несколько поддиапазонов. При измерении определяется поддиапазон, в который попадает значение, и в компьютере сохраняется только номер поддиапазона.

7.3. Арифметические операции над двоичными числами

Арифметические действия в двоичной системе счисления выполняют по тем же правилам, что и в десятичной системе, с той лишь разницей, что основание системы счисления равно двум и используются только две цифры.

Сложение	Вычитание	Умножение
$0 + 0 = 0$	$0 - 0 = 0$	$0 \cdot 0 = 0$
$0 + 1 = 1$	$1 - 0 = 1$	$0 \cdot 1 = 0$
$1 + 0 = 1$	$1 - 1 = 0$	$1 \cdot 0 = 0$
$1 + 1 = 10$	$10 - 1 = 1$	$1 \cdot 1 = 1$

Сложение чисел. Независимо от системы счисления сложение чисел выполняется поразрядно, начиная с младшего. Если при сложении цифр одного разряда сумма равна основанию системы счисления, образуется единица переноса в соседний старший разряд, а в разряде суммы записывается нуль. При сложении двух единиц разряда с единицей переноса из младшего разряда образуется единица переноса в старший разряд, а в разряде суммы записывается единица.

Действия двоичной арифметики можно проверить, если перевести все операнды (слагаемые, сумму и т. д.) в десятичную систему счисления:

$$1101 + 101 = 10010.$$

Вычитание чисел. При вычитании в данном разряде из нуля необходимо занять единицу из соседнего старшего разряда, которая равна двум единицам данного разряда:

$$1101 - 101 = 1001.$$

Умножение двоичных чисел. Осуществляется путем образования частичных произведений и последующего их суммирования. В соответствии с таблицей двоичного умножения каждое частичное произведение равно нулю, если в соответствующем разряде множителя стоит нуль, или равно множимому, сдвинутому на соответствующее число разрядов влево, если в разряде множителя стоит единица. Таким образом, операция умножения многоразрядных чисел сводится к операциям сдвига и сложения. Положение запятой определяется так же, как и при умножении десятичных чисел:

$$\begin{array}{r} 1110 \\ \cdot 10 \\ + 0000 \\ \hline 1110 \\ \hline 11100 \end{array}$$

Деление чисел. Производится по правилам умножения и вычитания, как и в десятичной системе счисления:

$$\begin{array}{r|l}
 1110 & 10 \\
 - 10 & 111 \\
 \hline
 11 & \\
 10 & \\
 10 & \\
 10 & \\
 0 &
 \end{array}$$

В большинстве компьютеров операция вычитания не используется. Вместо нее производится сложение обратных или дополнительных кодов уменьшаемого и вычитаемого. Это позволяет существенно упростить конструкцию арифметико-логического устройства.

Сложение обратных кодов. Здесь при сложении чисел A и B имеют место четыре основных и два особых случая.

1. A и B положительные. При суммировании складываются все разряды, включая разряд знака. Так как знаковые разряды положительных слагаемых равны нулю, разряд знака суммы тоже равен нулю. Например:

Десятичная запись	Двоичные коды
$ \begin{array}{r} + 3 \\ + 7 \\ \hline 10 \end{array} $	$ \begin{array}{r} + 00000011 \\ + 00000111 \\ \hline 00001010 \end{array} $

Получен правильный результат.

2. A положительное, B отрицательное и по абсолютной величине больше, чем A . Например:

Десятичная запись	Двоичные коды
$ \begin{array}{r} + 3 \\ + -10 \\ \hline -7 \end{array} $	$ \begin{array}{r} + 00000011 \\ + 11110101 \\ \hline 11111000 \end{array} $ <p style="margin-left: 20px;">Обратный код числа -10 Обратный код числа -7</p>

Получен правильный результат в обратном коде. При переводе в прямой код биты цифровой части результата инвертируются: $1\ 0000111 = -7_{10}$.

3. A положительное, B отрицательное и по абсолютной величине меньше, чем A . Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} + 10 \\ -3 \\ \hline 7 \end{array}$	$\begin{array}{r} + 00001010 \\ + 11111100 \text{ Обратный код числа } -3 \\ \hline 00000110 \\ \xrightarrow{+1} \\ 00000111 \end{array}$

Компьютер исправляет полученный первоначально неправильный результат (6 вместо 7) переносом единицы из знакового разряда в младший разряд суммы.

4. A и B отрицательные. Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} + -3 \\ -7 \\ \hline -10 \end{array}$	$\begin{array}{r} + 11111100 \text{ Обратный код числа } -3 \\ + 11111000 \text{ Обратный код числа } -7 \\ \hline 11110100 \\ \xrightarrow{+1} \\ 11110101 \text{ Обратный код числа } -10 \end{array}$

Полученный первоначально неправильный результат (обратный код числа -11_{10} вместо обратного кода числа -10_{10}) компьютер исправляет переносом единицы из знакового разряда в младший разряд суммы. При переводе результата в прямой код биты цифровой части числа инвертируются: $1\ 0001010 = -10_{10}$.

При сложении может возникнуть ситуация, когда старшие разряды результата операции не помещаются в отведенной для него области памяти. Такая ситуация называется переполнением разрядной сетки формата числа. Для обнаружения переполнения и оповещения о возникшей ошибке в компьютере используют специальные средства. Ниже приведены два возможных случая переполнения.

5. A и B положительные, сумма $A + B$ больше либо равна $2n - 1$, где n — количество разрядов формата чисел (для однобайтового формата $n = 8$, $2n - 1 = 27 = 128$). Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} + 65 \\ + 97 \\ \hline 162 \end{array}$	$\begin{array}{r} + 01000001 \\ + 01100001 \\ \hline 10100010 \text{ Переполнение} \end{array}$

Семи разрядов цифровой части числового формата недостаточно для размещения 8-разрядной суммы ($162_{10} = 10100010_2$),

поэтому старший разряд суммы оказывается в знаковом разряде. Это вызывает несовпадение знака суммы и знаков слагаемых, что является свидетельством переполнения разрядной сетки.

6. *A и B отрицательные, сумма абсолютных величин A и B больше либо равна 2^{n-1} .* Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} + \quad -63 \\ \quad \underline{-95} \\ -158 \end{array}$	$\begin{array}{r} + \quad 11000000 \text{ Обратный код числа } -63 \\ \quad \underline{10100000} \text{ Обратный код числа } -95 \\ \quad \left[\begin{array}{l} 01100000 \\ \rightarrow +1 \end{array} \right] \text{ Переполнение} \end{array}$

Здесь знак суммы тоже не совпадает со знаками слагаемых, что свидетельствует о переполнении разрядной сетки.

Сложение дополнительных кодов. Здесь также имеют место рассмотренные выше шесть случаев:

1. *A и B положительные.* В данном случае нет отличий от случая 1, рассмотренного для обратного кода.

2. *A положительное, B отрицательное и по абсолютной величине больше, чем A.* Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} + \quad 3 \\ \quad \underline{-10} \\ -7 \end{array}$	$\begin{array}{r} + \quad 00000011 \\ \quad \underline{11110110} \text{ Дополнительный код числа } -10 \\ 11111001 \text{ Дополнительный код числа } -7 \end{array}$

Получен правильный результат в дополнительном коде. При переводе в прямой код биты цифровой части результата инвертируются и к младшему разряду прибавляется единица: $1\ 0000110 + 1 = 1\ 0000111 = -7_{10}$.

3. *A положительное, B отрицательное и по абсолютной величине меньше, чем A.* Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} + \quad 10 \\ \quad \underline{-3} \\ 7 \end{array}$	$\begin{array}{r} + \quad 00001010 \\ \quad \underline{11111101} \text{ Дополнительный код числа } -3 \\ \quad \left[\begin{array}{l} 00000111 \\ \rightarrow \text{Перенос отбрасывается} \end{array} \right] \end{array}$

Получен правильный результат. Единицу переноса из знакового разряда компьютер отбрасывает.

4. *A* и *B* отрицательные. Например:

Десятичная запись		Двоичные коды		
+	-3	+	1111101	Дополнительный код числа -3
	<u>-7</u>		<u>1111001</u>	Дополнительный код числа -7
	-10		11110110	Дополнительный код числа -10
			└───┬───┘	
			└───┬───┘	→ Перенос отбрасывается

Получен правильный результат в дополнительном коде. Единицу переноса из знакового разряда компьютер отбрасывает.

Случаи переполнения для дополнительных кодов рассматривают по аналогии со случаями 5 и 6 для обратных кодов.

Сравнение рассмотренных форм кодирования целых чисел со знаком показывает:

- на преобразование отрицательного числа в обратный код компьютер затрачивает меньше времени, чем на преобразование в дополнительный код, так как последнее состоит из двух шагов – образования обратного кода и прибавления единицы к его младшему разряду;

- время выполнения сложения для дополнительных кодов чисел меньше, чем для их обратных кодов, потому что в таком сложении нет переноса единицы из знакового разряда в младший разряд результата.

Машинные методы умножения чисел. Умножение двоичных чисел машинным методом производится по следующему алгоритму:

- определяют знак произведения путем сложения по модулю двух знаков высшего разряда множителей;

- перемножение модулей чисел, начиная с младшего разряда множителя, одним из способов умножения: а) множимое неподвижно, сумматор частичных произведений сдвигается вправо; б) сумматор частичных произведений неподвижен, множимое сдвигается влево.

Оба способа дают одинаковый результат. В обоих случаях умножение начинается с анализа младшего разряда множителя и производится в прямом коде.

Способ 1 – множимое неподвижно, сумматор сдвигается вправо.

Анализируется очередной разряд множителя: если он равен единице, к содержимому сумматора прибавляется множимое и

сумматор сдвигается вправо; если он равен нулю, производится только сдвиг сумматора вправо. Причем количество сдвигов равно количеству разрядов множителя, а количество прибавлений множимого к сумматору равно количеству единиц во множителе.

Пример:

$[x1]_{\text{пр}} = 01101$

$[x2]_{\text{пр}} = 11001$

1. Определяем знак произведения: $0 + 1 = 1$.

2. Находим произведение модулей чисел первым способом: множимое – 1101, множитель – 1001:

1101 (множимое)

01101000 (сдвиг)

00110100 (сдвиг)

01110101 (сдвиг)

Способ 2 – сумматор неподвижен, множимое сдвигается влево.

Анализируется очередной разряд множителя: если он равен единице, множимое прибавляется к сумматору и сдвигается влево на один разряд; если он равен нулю, производится только сдвиг множимого влево.

3. Находим произведение вторым способом:

1101 (множимое)

Ответ: -01110101

7.4. Физические формы представления информации

В электронных вычислительных машинах обрабатывается цифровая, т. е. дискретная, информация, а не непрерывная. Изменение информации происходит в определенные моменты времени, временной интервал между двумя соседними моментами дискретного времени называют *тактом*, или *периодом представления информации*.

В ЭВМ предусмотрены специальные блоки, которые вырабатывают сигналы, отмечающие такты работы машины, – *блоки синхронизации*.

Наиболее распространенными способами физического представления информации являются потенциальный и импульсный.

Потенциальный способ. Код единицы при данном способе – это один уровень напряжения, код нуля – противоположный ему. Например, если 1 – высокий потенциал определенного

уровня, 0 – нулевой потенциал, или если 1 – нулевой потенциал, 0 – низкий потенциал. Уровень напряжения и его полярность определяются электрическими схемами и параметрами тех приборов, на которых строится узел ЭВМ. Потенциальный сигнал не изменяется в пределах всего такта.

Импульсный способ. Код единицы представляется наличием электрического импульса, код нуля – его отсутствием. Либо единица может представляться импульсом одной полярности, ноль – противоположной. Это зависит от тех базовых элементов, на которых построена электронная схема. Импульсный сигнал характеризуется амплитудой и длительностью. Длительность импульса должна быть не больше такта машины.

В ЭВМ широко используют *импульсно-потенциальные схемы*, в которых информация представляется и тем, и другим способом.

Обработка информации внутри ЭВМ, передача информации от одного узла ЭВМ к другому производится многоразрядными числами (словами). Число может быть передано параллельным или последовательным способом.

При параллельном способе передачи или обработки числа все разряды этого числа поступают по цепям ЭВМ одновременно, т. е. в один и тот же такт машины. Происходит развертка слова в пространстве, для чего необходимо иметь столько электрических цепей, сколько разрядов содержит двоичное число.

При последовательном способе передачи числа все разряды этого числа передаются по одной электрической цепи развернутыми во времени. Для передачи n -разрядного числа в этом случае потребуется n тактов машинного времени.

Устройства ЭВМ могут быть предназначены для любого способа передачи и обработки информации, в соответствии с этим они называются параллельными или последовательными. Например, сумматор может быть параллельного или последовательного действия. Обработка и передача информации внутри ЭВМ ведутся, как правило, в параллельном коде. Это обеспечивает более высокое быстродействие, но требует большего объема аппаратуры. Последовательный способ передачи информации используется при передаче информации вне ЭВМ, т. е. там, где практически невозможно обеспечить параллельную пересылку чисел или слов. Этот способ при всей своей экономичности резко снижает скорость обработки информации.

Весьма распространенным является **параллельно-последовательный способ** передачи информации. Слово разбивается на слоги, слог передается параллельно, слог за слогом – последовательно. Такой способ передачи информации используется для обмена информацией между внутренними и внешними устройствами ЭВМ.

Контрольные вопросы и задания

1. Объясните назначение систем счисления.
2. Опишите виды систем счисления.
3. Поясните порядок представления чисел: а) с фиксированной запятой; б) с плавающей запятой.
4. Как происходит кодирование сигнала?
5. Какие математические операции можно производить над двоичными числами?
6. Назовите формы представления информации.

8. СТРУКТУРА ЭЛЕКТРОННО-ВЫЧИСЛИТЕЛЬНЫХ МАШИН, МИКРОЭВМ

8.1. Общие сведения

Электронно-вычислительная машина (ЭВМ) — комплекс технических средств, объединенных общим управлением и предназначенных для автоматического выполнения преобразований цифровых кодов в соответствии с заданным алгоритмом.

В структуре любой ЭВМ можно выделить следующие компоненты: память; процессор; устройства ввода-вывода.

Память — запоминающее устройство для хранения информации. Память делится на два базовых вида — внутреннюю (оперативную) и внешнюю.

Оперативная память предназначена для временного хранения информации. Она состоит из ячеек, которые имеют адрес и содержимое. Ячейки состоят из более мелких элементов памяти — байтов.

Адрес ячейки — n -разрядное слово, используемое для указания определенного места, обычно ячейки памяти.

Содержимое — это информация, находящаяся в ячейке и закодированная в двоичном коде.

Внешняя память предназначена для длительного хранения информации и обмена с оперативной памятью.

Единицами измерения памяти являются байты, килобайты, мегабайты, гигабайты, терабайты и т. д.

Процессор — устройство для выполнения арифметических, логических операций и контроля над работой всей ЭВМ.

Процессоры характеризуются своей *архитектурой* (набором команд и др.), *разрядностью*, *скоростью* выполнения операций.

Устройства ввода-вывода — устройства, предназначенные для общения пользователя с компьютером.

Алгоритм — совокупность точных предписаний (команд), выполнение которых ЭВМ приводит к решению поставленной задачи путем преобразования исходных цифровых кодов в цифровые коды результата.

В общем виде ЭВМ является электронной системой сбора, хранения, накопления, переработки и выдачи цифровой информации.

Основные свойства ЭВМ — программное управление, алгоритмическая универсальность, высокие точность и скорость вычисления.

Структура ЭВМ — абстрактная модель, устанавливающая состав, порядок и принципы взаимодействия основных функциональных частей ЭВМ без учета их реализации.

Функциональная часть ЭВМ — составная часть ЭВМ, имеющая определенное функциональное назначение.

МикроЭВМ — это вычислительная машина, имеющая разрядность в один или два байта, малые габариты и низкую стоимость.

Разрядность — количество разрядов в цифровых кодах, обрабатываемых в данной машине.

Типовая **архитектура микроЭВМ** содержит пять основных элементов: устройства ввода; устройство управления; микропроцессор (микроконтроллер); память; устройства вывода.

Вариант архитектуры и структура функциональных элементов ЭВМ представлены на рисунке 8.1.

Архитектура ЭВМ — размещение соединенных между собой элементов, отражающее структуру связей, влияющих на основные характеристики ЭВМ.

8.2. Устройство процессора

Основными частями микропроцессора являются арифметико-логическое устройство, устройство управления и регистры.

Арифметико-логическое устройство (АЛУ) — это один из узлов центрального процессора, который отвечает за реализацию всех логических и арифметических преобразований различных уровней сложности, которые протекают в электронно-вычислительной машине. В качестве выполняемых операций используют операнды, длина которых соответствует длине слова или его размеру.

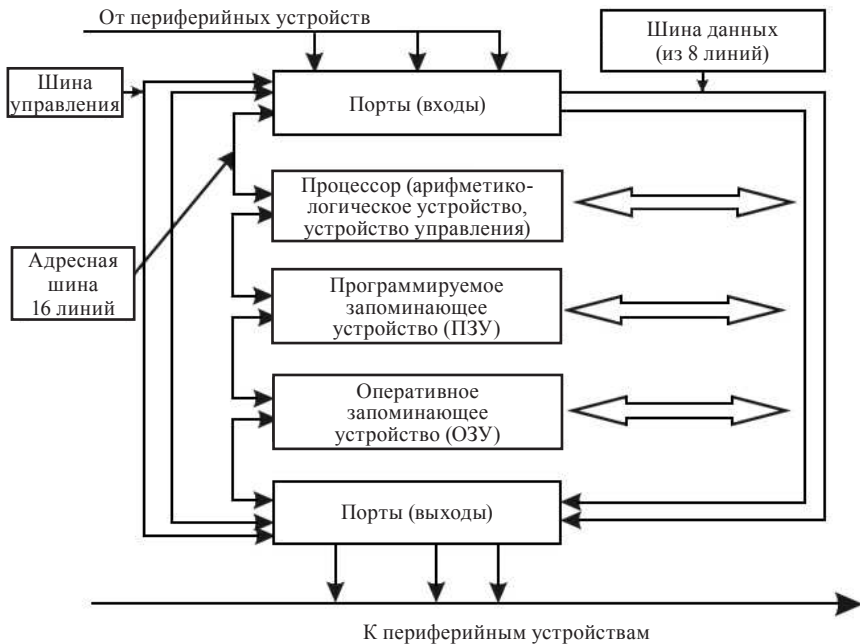


Рис. 8.1. Структурная схема ЭВМ

Главное предназначение АЛУ – обработка данных, которые находятся в оперативном запоминающем устройстве компьютера. Кроме того, АЛУ генерирует управляющие сигналы, которые являются для ЭВМ своего рода инструкциями, указывающими, что необходимо делать с определенным набором данных. Каждая из операций, происходящих в процессоре, выполняется при помощи электронных схем, которые могут состоять из нескольких тысяч электронных элементов, установленных очень плотно друг к другу.

Поступающие на вход АЛУ электронные сигналы генерируют определенный тип обработки данных, выраженных в двоичной системе исчисления. Существуют четыре разновидности заданных типов действий, которыми способен работать этот узел центрального процессора. Таким образом, количество базовых операций является основной характеристикой этого узла процессора.

В свою очередь для выполнения определенной операции с данными в АЛУ реализованы четыре элемента, на каждый из

которых возложено выполнение определенных задач: хранение, передача, управление и преобразование данных, поступающих на вход узла из оперативной памяти.

К **узлу хранения** относят следующие электронные элементы:

- триггеры, отвечающие за хранение вспомогательных данных и конечных результатов, необходимых для выполнения различных процессов;
- регистры, основным предназначением которых является сохранение целостной структуры аргументов операций промежуточных и конечных результатов.

В некоторых случаях совокупность регистров может выступать в качестве блока памяти, а триггеры – в качестве общего регистра состояния.

Составляющие **узла передачи** центрального процессора:

- шины, отвечающие за объединение блоков АЛУ и их обоюдное взаимодействие;
- мультиплексоры и вентили, благодаря которым задается нужное направление выполняемых преобразований.

Элементами **узла преобразования** выступают:

- сумматоры, отвечающие за выполнение микроопераций;
- сдвигатели;
- схемы выполнения логических операций;
- корректоры для десятичной арифметики;
- преобразователи кода, принимающие участие в операциях с обратными и дополнительными данными, которые могут потребоваться в процессе преобразования;
- счетчики, осуществляющие подсчет завершенных циклов преобразования данных.

Основные элементы **узла управления**:

- контрольный блок;
- дешифратор сигналов;
- схемы преобразования логических признаков, которые формируют ветви выполнения заданных команд.

В процессе работы процессора в узле управления создаются электронные сигналы, подаваемые на другие узлы центрального процессора строго в определенной последовательности, которые необходимы для выполнения поставленной задачи. Этот процесс осуществляется поэтапно и состоит из нескольких тактов.

Управляющее устройство отвечает за выполнение программ в автоматическом режиме, однако в процессе его работы могут

применяться прерывания, необходимые для вовлечения в работу прочих узлов процессора. Работа устройства управления возможна благодаря основному принципу микропрограммирования, который обладает определенным набором характеристик.

В зависимости от *принципа работы* и *метода обработки массивов данных* АЛУ можно разделить на две категории: последовательные и параллельные. Эти категории различаются методами представления элементов операций и их реализации.

Кроме того, существует классификация АЛУ по *характеру использования*, в которой этот узел подразделяется на блочные и многофункциональные. Для первого типа характерно распределение данных по однотипным блокам и последующее их выполнение. В процессе работы второго типа для обработки данных задействуются все имеющиеся микросхемы, которые в свою очередь могут приспособливаться под различные типы данных. При этом высокая скорость работы АЛУ достигается благодаря тому, что все задачи выполняются в параллельном режиме, что обеспечивает их многозадачность. Однако такой принцип работы сопровождается и большим энергопотреблением.

По *способу представления данных* АЛУ можно применять для работы с десятичными числами, а также с числами, использующими плавающую или фиксированную запятую.

Арифметические операции и логические процедуры, происходящие в узле арифметико-логических преобразований, можно подразделить на несколько категорий. Арифметические операции представляют собой основные математические операции, такие как деление, умножение, вычитание модулей, обычное вычитание и сложение.

В логических преобразованиях чаще используют двоичные слова, в состав которых входят множественные разряды.

В любом преобразовании данных, происходящем в АЛУ, присутствуют свои аргументы операции, а конечные результаты обработки данных выражаются в виде 16-разрядных битовых строк. Обработанные во время преобразования данные выражаются в цифровом виде со знаком «+» или «-», в зависимости от наличия переполнения, при помощи специальных флагов, которые используются при непредсказуемых изменениях знака, который выражается в виде отдельного бита. Логика переноса бита между системами различных разрядов происходит по принципу

без знаковой арифметики. Таким образом, система присваивает знак «+» в том случае, если обрабатываемые данные в старшем разряде не могут быть выражены конечным результатом в младшем разряде.

8.3. Устройство управления

Устройство управления выполняет функции управления вычислительным процессом и осуществляет автоматическое исполнение программы, обеспечивая для этого следующие основные действия: извлечение из памяти очередных команд, их расшифровку и преобразование в последовательность стандартных элементарных операций, занесение в АЛУ исходных данных, сохранение полученных в АЛУ результатов. Устройство управления обеспечивает синхронизацию работы процессора, оперативной памяти, а также других устройств.

Состав устройства управления: регистр команд; счетчик команд; микропрограммное устройство управления; очередь команд.

Регистр команд предназначен для приема из памяти команды и ее хранения. Команда, которая находится в регистре команд, называется текущей, именно ее процессор выполняет в данное время.

Счетчик (указатель) команд хранит относительный адрес команды. После выборки команды из памяти содержимое счетчика команд увеличивается на длину команды. Программисту счетчик команд недоступен. Содержимое счетчика команд изменяется при выполнении следующих команд:

- *безусловного перехода* — перезаписывает указатель команд (регистр IP или CS), что заставляет процессор «переключиться» на выполнение команды по указанному адресу;

- *условных переходов* — если условие истинно, выполняется переход на указанную метку, если нет, процессор продолжит выполнять программу со следующей команды;

- *перехода на подпрограмму* — проверяют значения разрядов регистра состояния и определяют последующий ход выполнения программ в зависимости от результата проверки, позволяют организовывать в программах циклы и разветвления;

- *прерывание* — инициируемый определенным образом процесс, временно переключающий микропроцессор на выполнение

другой программы с последующим возобновлением прерванной программы.

Микропрограммное устройство управления обеспечивает преобразование кода операции команды в сигналы, управляющие блоками процессора.

Для каждой команды разработчик процессора разрабатывает микропрограмму. Микропрограммы всех команд процессора хранятся в постоянной памяти процессора.

Очередь команд увеличивает скорость работы микропроцессора. Пока выполняется команда, находящаяся в регистре команд, если магистраль процессора свободна, из памяти считывается следующая команда и загружается в очередь команд.

Архитектура отображает те аспекты ЭВМ, которые являются видимыми для пользователя: систему команд, режимы адресации, форматы и длину данных, набор регистров ЭВМ.

Физически реализованные в виде соответствующей аппаратуры функциональные блоки ЭВМ называют **аппаратными средствами**.

Структурная схема микроЭВМ показана на рисунке 8.2.

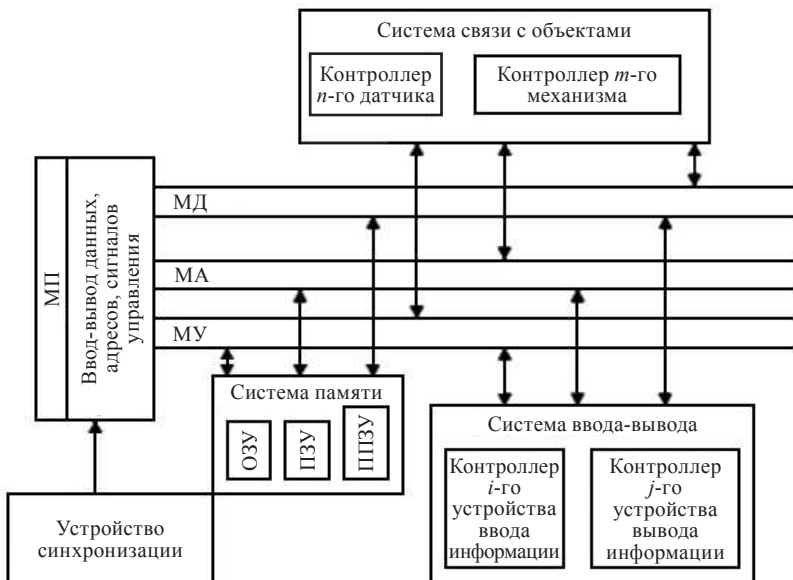


Рис. 8.2. Структурная схема микроЭВМ

Модель в процессе проектирования преобразуется в структуру микроЭВМ, определяющую состав, назначение и взаимные связи необходимых аппаратурных компонентов, реализующих архитектуру.

Необходимость выполнения сложных функций управления привела к созданию **микроконтроллеров** – управляющих устройств, выполненных на одном или нескольких кристаллах. Микроконтроллеры выполняют функции логического анализа и управления. Поэтому за счет исключения арифметических операций можно уменьшить их аппаратурную сложность или развить функции логического управления.

В приведенных схемах (см. рис. 8.1, 8.2) обработку информации осуществляет микропроцессор, синхронизируемый тактовыми импульсами устройства синхронизации. Для того чтобы все блоки микроЭВМ работали в правильной последовательности, применяют специальный генератор тактовых импульсов, синхронизирующих работу всей системы.

Обмен информацией между микропроцессором и остальными блоками микроЭВМ осуществляется по трем магистралям (называемым также шинами): адресной, данных и управляющей.

Шина (BUS) – группа линий передачи информации, объединенных общим функциональным назначением. Например, шина данных, шина адресов, шина управления.

По каждой линии шины может передаваться только один двоичный разряд (бит) информации. Поэтому разрядность шины указывает на количество параллельных линий, из которых она состоит.

Магистраль (шина) адреса служит для передачи кода адреса, по которому проводится обращение к устройствам памяти, ввода-вывода или другим внешним устройствам, подключенным к микропроцессору.

Обрабатываемая информация и результаты вычислений передаются по магистрали данных.

Магистраль управления передает управляющие сигналы на все блоки ЭВМ, настраивая на нужный режим устройства, участвующие в выполнении команды.

Программные средства – это общий термин, охватывающий все программы.

8.4. Запоминающие устройства микропроцессорных систем

Когда программа помещается в память постоянно, она располагается в **постоянном запоминающем устройстве** (ПЗУ). Это устройство является кристаллом программируемой интегральной микросхемы с неизменяемой программой. В ПЗУ хранится программное обеспечение, используемое при работе с каждой программой (например, программа инициализации ЭВМ, управляющие программы).

По *типу исполнения* выделяют:

1) ПЗУ, в которых массив данных совмещен с устройством выборки (считывающим устройством):

- микросхему ПЗУ;
- один из внутренних ресурсов однокристалльного компьютера (микроконтроллера), как правило, FlashROM;

2) ПЗУ, в которых массив данных существует самостоятельно:

- компакт-диск;
- USB-накопитель;
- штрих-коды.

По *разновидностям микросхем* выделяют ПЗУ:

1) по технологии изготовления кристалла:

- ROM (Read Only Memory, постоянное запоминающее устройство) – масочное ПЗУ, изготавливаемое фабричным методом;
- PROM (Programmable Read Only Memory, программируемое ПЗУ (ППЗУ)) – ПЗУ, однократно «прошиваемое» пользователем;
- EPROM (Erasable Programmable Read Only Memory) – перепрограммируемое ПЗУ;

• EEPROM (Electrically Erasable Programmable Read Only Memory – электрически стираемое перепрограммируемое ПЗУ, память которого может стираться и заполняться данными несколько десятков тысяч раз, используется в твердотельных накопителях, одной из разновидностей EEPROM является флеш-память;

• на магнитных доменах, которое имело сложное устройство выборки и хранило довольно большой объем данных в виде намагниченных областей кристалла, при этом, не имея движущихся частей, обеспечивает неограниченное количество циклов перезаписи;

- NVRAM (Non Volatile Random Access Memory, «неразрушающаяся» память) – ПЗУ, выполняющее роль ОЗУ небольшого объема, конструктивно совмещенное с батареей;

2) по виду доступа:

- ПЗУ с параллельным доступом – ПЗУ, которое в системе может быть доступно в адресном пространстве ОЗУ;

- ПЗУ с последовательным доступом – ПЗУ, часто используемые для однократной загрузки констант либо «прошивки» в процессор или в ПЛИС, используемые для хранения, например, настроек каналов телевизора и других данных;

3) по способу программирования:

- непрограммируемые;

- программируемые только с помощью специального устройства – программатора ПЗУ (как однократно, так и многократно прошиваемые); использование программатора необходимо, в частности, для подачи нестандартных и относительно высоких напряжений (до ± 27 В) на специальные выводы;

- внутрисхемно перепрограммируемые ПЗУ (ISP – In System Programming) – микросхемы, имеющие внутри генератор всех необходимых высоких напряжений; могут быть перепрошиты программным способом, т. е. без программатора и без выпайки из печатной платы.

Практически всегда эта информация не требуется одновременно, поэтому простейшие устройства для запоминания постоянной информации можно построить на мультиплексорах. Иногда в переводной литературе ПЗУ называют ROM памятью, доступной только для чтения). Схема такого ПЗУ приведена на рисунке 8.3. В этой схеме ПЗУ разделено на восемь одноразрядных ячеек. Запоминание конкретного бита в одноразрядную ячейку производится запайкой провода к источнику питания (запись единицы) или запайкой провода к корпусу (запись нуля).

На принципиальных схемах такое устройство обозначается следующим образом (рис. 8.4).

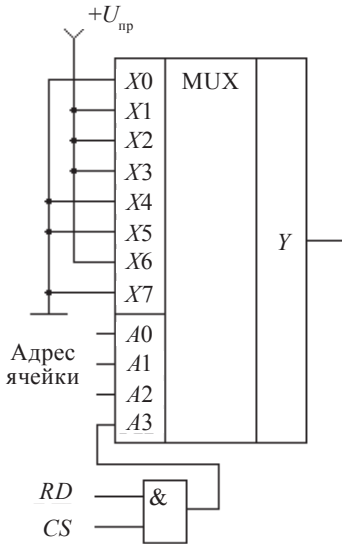


Рис. 8.3. Схема постоянного запоминающего устройства, построенного на мультиплексе

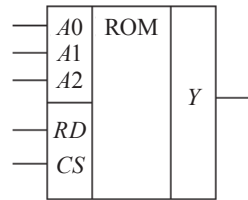


Рис. 8.4. Обозначение постоянного запоминающего устройства на принципиальных схемах

Для того чтобы увеличить разрядность ячейки памяти ПЗУ, эти микросхемы можно соединять параллельно (выходы и записанная информация естественно остаются независимыми). Схема параллельного соединения одноразрядных ПЗУ приведена на рисунке 8.5.

В реальных ПЗУ запись информации осуществляется при помощи последней операции производства микросхемы — металлизации. Металлизация производится при помощи маски, поэтому такие ПЗУ получили название *масочных*. Еще одно отличие реальных микросхем от упрощенной модели, приведенной выше, — это использование, кроме мультиплексора, еще и демультиплексора. Такое решение позволяет превратить одномерную запоминающую структуру в двухмерную и тем самым существенно сократить объем схемы дешифратора, необходимого для работы схемы ПЗУ.

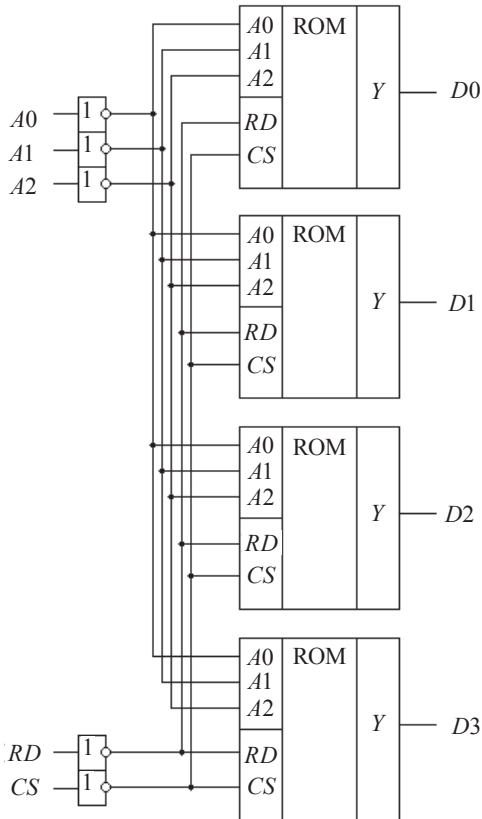


Рис. 8.5. Схема многоадресного постоянного запоминающего устройства (ROM)

Память изменяющихся данных называют **оперативным запоминающим устройством** (ОЗУ) и составляют также из интегральных микросхем.

Оперативная память (ОЗУ, RAM) — относительно быстрая энергозависимая память компьютера с произвольным доступом, в которой осуществляется большинство операций обмена данными между устройствами. Энергозависима, т. е. при отключении питания все данные на ней стираются.

Оперативная память является хранилищем всех потоков информации, которые необходимо обработать процессору, или

же они дожидаются в оперативной памяти своей очереди. Все устройства связываются с оперативной памятью через системную шину, а с ней в свою очередь обмениваются через кэш-память или напрямую (рис. 8.6).

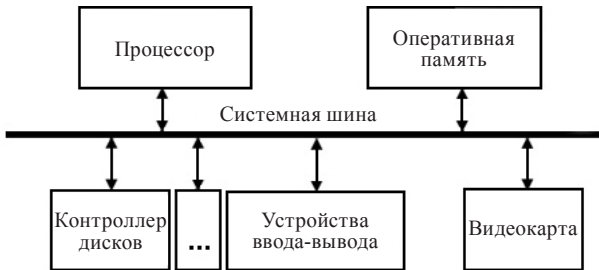


Рис. 8.6. Схема работы оперативного запоминающего устройства

Random Access Memory (RAM) – память с произвольным (прямым) доступом (рис. 8.7), которая означает то, что при необходимости память может напрямую обратиться к одному, необходимому блоку, не затрагивая при этом остальные. Скорость произвольного доступа не меняется от места нахождения нужной информации, что является огромным плюсом.

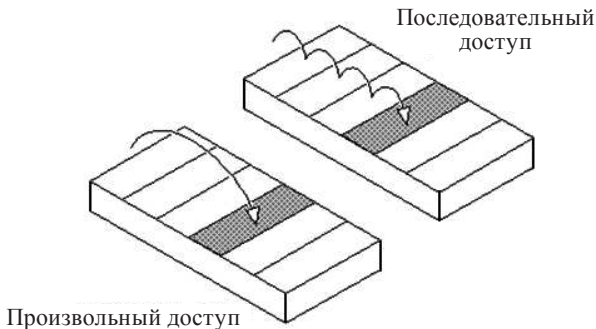


Рис. 8.7. Память с последовательным и произвольным (прямым) доступом

Оперативная память выгодно отличается от энергозависимой памяти практически нулевым влиянием количества операций чтения-записи на срок службы и долговечность. При соблюдении всех тонкостей при производстве оперативная память очень редко выходит из строя. В большинстве случаев поврежденная память

начинает допускать ошибки, которые приводят к краху системы или к нестабильной работе многих устройств компьютера.

Оперативная память может быть как отдельным модулем, который можно менять и добавлять дополнительные (компьютер, например), так и отдельным блоком устройства или чипа (как в микроконтроллерах или в простейших SoC – однокристалльных системах).

Современные операционные системы активно используют оперативную память для хранения и обработки в ней важных и часто используемых данных. Если бы в электронных устройствах не использовалась оперативная память, то все операции происходили бы гораздо медленнее и для считывания с постоянного источника памяти (ПЗУ) требовалось бы значительно больше времени. Да и более-менее многопоточная обработка была бы практически невозможна.

Использование оперативной памяти позволяет приложениям работать и запускаться быстрее. Данные беспрепятственно могут обрабатываться и ждать своей очереди благодаря адресуемости (все машинные слова имеют свои собственные адреса).

В большинстве устройств используется динамическая память с произвольным доступом DRAM (Dynamic Random Access Memory), но она медленнее статической SRAM (Static Random Access Memory). Эта память нашла свое применение в быстрой кэш-памяти процессоров, видеочипов и контроллеров. Поскольку статическая память занимает на кристалле гораздо больше места, чем динамическая, во времена быстрого развития компьютерной периферии и операционных систем производители пошли по пути большего объема, а не по пути более высокой скорости, что было более оправдано.

Аналогично используется регистровая *кэш-память* – высокоскоростная память большой емкости, являющаяся буфером между ОЗУ и микропроцессором и позволяющая увеличить скорость выполнения операций.

Регистры кэш-памяти (участок памяти, забранный с жесткого диска, в котором хранится информация, наиболее часто необходимая процессору) недоступны для пользователя, отсюда и название кэш (англ. cache – тайник).

В кэш-памяти хранятся данные, которые микропроцессор получил и будет использовать в ближайшие такты своей работы.

Быстрый доступ к этим данным позволяет сократить время выполнения очередных команд программы. При выполнении программы данные, считанные из ОЗУ с небольшим опережением, записываются в кэш-память.

По принципу записи результатов различают два типа кэш-памяти:

- *кэш-память «с обратной записью»* — результаты операций, прежде чем их записать в ОЗУ, фиксируются в кэш-памяти, а затем ее контроллер самостоятельно перезаписывает эти данные в ОЗУ;
- *кэш-память «со сквозной записью»* — результаты операций одновременно параллельно записываются и в кэш-память, и в ОЗУ.

Все указанные типы памяти имеют основные параметры: емкость памяти и ее быстродействие.

Быстродействие микропроцессорной системы, кэш-памяти и ОЗУ измеряется временем обращения к ним (сумма времени поиска, считывания и записи информации).

Быстродействие внешних запоминающих устройств характеризуется двумя параметрами: временем доступа (время поиска информации на носителе) и скоростью считывания (скорость считывания смежных байтов информации подряд — трансфер).

8.5. Периферийное оборудование микроЭВМ

Внешние устройства сопрягаются с магистралями через порты ввода-вывода, которые преобразуют данные в параллельную или последовательную форму.

Порт (контроллер) — специальная микросхема, с помощью которой микропроцессор связывается с внешними устройствами.

Порты ввода данных преобразуют данные из последовательной формы в параллельную.

Порты вывода, наоборот, преобразуют данные в последовательную форму.

Последовательная передача данных — поразрядная передача данных.

Параллельная передача данных — одновременная передача двух и более разрядов.

Линии передачи данных обычно буферизируются, прежде чем они будут подключены к соответствующим портам.

Буфер – схема, обеспечивающая прохождение данных между внешними устройствами и шиной данных, а также электрическую изоляцию между отдельными частями системы, согласование их импедансов (полных электрических сопротивлений).

Регистр – ячейка памяти микропроцессора, обычно содержащая одно слово.

Слово – совокупность двоичных разрядов, воспринимаемая при обработке как единое целое, помещается для хранения в одну ячейку памяти.

Некоторые устройства ввода и вывода снабжаются специальными схемами, выполняющими специальную процедуру управления обменом, при котором набор управляющих сигналов, т. е. их интерфейс (интерфейс внешних устройств) несовместим с системным интерфейсом микроЭВМ. Для сопряжения микроЭВМ с внешним устройством используют контроллеры. Микропроцессор имеет возможность «опросить» внешнее устройство, готово ли оно к обмену информацией, и данные передаются только при утвердительном ответе.

Одно из основных различий между разными типами ЭВМ – это организация системных шин, обеспечивающих связь между отдельными блоками ЭВМ. Самой популярной структурой шин, которая используется во многих ЭВМ, является структура с общей шиной. В таком случае при наличии общей шины для всех блоков ЭВМ возникает задача подключения выходов нескольких элементов к одной шине. Известны три способа решения задачи: логическое объединение, объединение с помощью схем с открытым коллектором и объединение с использованием схем с тремя состояниями.

Периферийное устройство – устройство, входящее в состав внешнего оборудования микроЭВМ, обеспечивающее ввод-вывод данных, организацию промежуточного и длительного хранения данных. Периферийные устройства микроЭВМ подключаются к его интерфейсам и предназначены для выполнения вспомогательных операций. Благодаря им компьютерная система приобретает гибкость и универсальность.

По назначению периферийные устройства можно подразделить:

- на устройства ввода данных;
- устройства вывода данных;
- устройства хранения данных;
- устройства обмена данными.

8.6. Устройства ввода данных

К устройствам ввода данных относятся клавиатура, сканеры, устройства управления курсором, графические планшеты (дигитайзеры), устройства ввода акустических сигналов, цифровые фото- и видеокамеры, TV-устройства ввода и т. д.

Клавиатура — это одно из основных устройств ввода в ЭВМ, позволяющее вводить различные виды информации. Вид вводимой информации определяется программой, интерпретирующей нажатые или отпущенные клавиши. С помощью клавиатуры можно вводить любые символы: от букв и цифр до иероглифов и знаков музыкальной нотации.

Клавиатура позволяет управлять курсором на экране дисплея — устанавливать его в нужную точку экрана, перемещать по экрану, «прокручивать» экран в режиме скроллинга, отправлять содержимое экрана на принтер, производить выбор при наличии альтернативных вариантов и т. д.

Стандартная клавиатура имеет несколько групп клавиш:

- алфавитно-цифровые и знаковые клавиши (с латинскими и русскими буквами, цифрами, знаками пунктуации, математическими знаками);
- специальные клавиши: Esc, Tab, Enter, BackSpace;
- функциональные клавиши: (F1, F2 и т. д.);
- служебные клавиши для управления перемещением курсора (стрелки Up, Down, Left, Right, клавиши Home, End, PgUp, PgDn и клавиша 5, иногда обозначаемая значком «[]», в центре дополнительной цифровой клавиатуры);
- служебные клавиши для управления редактированием: Ins, Del;
- служебные клавиши для смены регистров и модификации кодов других клавиш: Alt, Ctrl, Shift;
- служебные клавиши для фиксации регистров: CapsLock, ScrollLock, NumLock;
- разные вспомогательные клавиши: PrtScr, Break.

Сканеры подразделяют на планшетные, ручные и штрих-сканеры.

Планшетные сканеры предназначены для ввода графической информации с прозрачного или непрозрачного листового материала. Принцип действия этих устройств состоит в том, что луч света, отраженный от поверхности материала или прошед-

ший сквозь прозрачный материал, фиксируется специальными элементами, называемыми приборами с зарядовой связью (обычно их конструктивно оформляют в виде линейки, располагаемой по ширине исходного материала). Перемещение линейки относительно листа бумаги выполняется механическим протягиванием линейки при неподвижной установке листа или протягиванием листа при неподвижной установке линейки.

Основные потребительские параметры планшетных сканеров:

- разрешающая способность;
- производительность;
- динамический диапазон;
- максимальный размер сканируемого материала.

Разрешающая способность планшетного сканера зависит от плотности размещения приборов с зарядной связью на линейке, а также от точности механического позиционирования линейки при сканировании.

Производительность сканера определяется продолжительностью сканирования листа бумаги стандартного формата и зависит как от совершенства механической части устройства, так и от типа интерфейса, использованного для сопряжения с компьютером.

Принцип действия *ручных сканеров* в основном соответствует планшетным. Разница заключается в том, что протягивание линейки приборов зарядовой связи (ПЗС) в данном случае выполняется вручную. Равномерность и точность сканирования при этом обеспечиваются неудовлетворительно.

Штрих-сканеры (разновидность сканеров) предназначены для ввода данных, закодированных в виде штрих-кода. Такие устройства имеют применение, например, в розничной торговой сети и складском хозяйстве.

Устройства управления курсором включают в себя мышь, джойстик, тачпад и др. Они используются для перемещения курсора по экрану и для отметки позиции, в которой находится курсор.

Манипулятор «мышь» — координатное устройство для управления курсором и отдачи различных команд компьютеру. Управление курсором осуществляется путем перемещения мыши по поверхности стола или коврика для мыши. Клавиши и колесико мыши вызывают определенные действия, например активацию указанного объекта или вызов контекстного меню. В настоящее время распространение получили оптические, в том числе лазерные, мыши.

Подключение к компьютеру может осуществляться как через USB-интерфейс, так и по различным протоколам радиосвязи.

Трекбол (англ. trackball – шаровый манипулятор) – указательное устройство ввода информации об относительном перемещении для компьютера. Аналогично мыши по принципу действия и по функциям. Трекбол функционально представляет собой оптико-механическую шариковую конструкцию. Шар находится сверху или сбоку корпуса и пользователь может вращать его ладонью или пальцами, при этом не перемещая корпус устройства. Несмотря на внешние различия, трекбол и мышь конструктивно похожи: при движении шар приводит во вращение пару валиков или, в более современном варианте, его сканируют оптические датчики перемещения (как в оптической мыши).

Основная область применения трекболов – в основном работа с графическими пакетами, пакетами для автоматизированного проектирования и им подобными, т. е. такими приложениями, в которых наиболее остро ощущается необходимость плавного перемещения и точного позиционирования курсора.

Тачпад (англ. touchpad – сенсорная площадка, сенсорная панель) – указательное устройство ввода, применяемое чаще всего в портативных ЭВМ (ноутбуках). Как и другие указательные устройства, тачпад обычно используется для управления «указателем» путем перемещения пальца по поверхности устройства.

Работа тачпадов основана на измерении емкости пальца или измерении емкости между сенсорами. Емкостные сенсоры расположены вдоль вертикальной и горизонтальной осей тачпада, что позволяет определить положение пальца с нужной точностью.

Современные тачпады также поддерживают технологию *мультитач*, т. е. реагируют на множественные касания и распознают определенные манипуляции.

Джойстик (англ. joystick – досл.: палочка счастья) – устройство ввода информации; представляет собой манипулятор, посредством которого можно задавать экранные координаты графического объекта; также может выполнять функции клавиатуры.

Джойстик представляет собой ручку, наклоном которой можно задавать направление в двухмерной плоскости. На ручке, а также в платформе, на которой она крепится, обычно располагаются кнопки и переключатели различного назначения. Помимо координатных осей X и Y , возможно также изменение коорди-

наты Z за счет вращения рукояти вокруг оси, наличия второй ручки, дополнительного колесика и т. п.

Трехмерные джойстики позволяют осуществлять управление перемещением контролируемого объекта в трех плоскостях. Наибольшее применение нашли в системах автоматизированного проектирования.

Задача получения 3D-моделей реальных объектов стоит перед промышленными дизайнерами, инженерами, художниками, аниматорами, разработчиками игровых приложений. Измерение геометрии сложных пространственных форм является основным требованием для современных производителей технологической оснастки.

Графические планшеты (дигитайзеры) – устройства, предназначенные для ввода художественной графической информации. Существует несколько различных принципов действия графических планшетов, но в основе всех их лежит фиксация перемещения специального пера относительно планшета. Такие устройства удобны для художников и иллюстраторов, поскольку позволяют им создавать экранные изображения привычными приемами, наработанными для традиционных инструментов (карандаш, перо, кисть).

В состав графических планшетов входит специальный указатель с датчиком, называемый пером. Собственный контроллер посылает импульсы по ортогональной сетке проводников, расположенной под плоскостью планшета. Получив два таких сигнала, контроллер преобразует их в координаты, передаваемые в ЭВМ. Компьютер переводит эту информацию в координаты точки на экране монитора, соответствующие положению указателя на планшете.

Дигитайзер (англ. digitize – оцифровывать) – это кодирующее устройство, обеспечивающее ввод двухмерного (в том числе и полутонного) или трехмерного (3D-дигитайзеры) изображения в компьютер в виде растровой таблицы. С помощью специального программного обеспечения он позволяет преобразовывать движение руки оператора в формат векторной графики. Дигитайзер является типичным внешним специализированным устройством графического ввода.

Дигитайзер предназначен для профессиональных графических работ. Первоначально дигитайзер был разработан для приложений систем автоматизированного проектирования, так как в этом случае необходимо определять и задавать точное значение координат большого количества точек. В отличие от мыши, дигитайзер способен точно определять и обрабатывать абсолютные координаты.

3D-дигитайзер представляет собой настольное устройство, способное отображать объекты небольших и средних размеров в виде точных трехмерных файлов.

Например, при использовании механической технологии оцифровки 3D-объектов контактный щуп, установленный на складной арматуре с шарнирными соединениями, считывает информацию о том, в каком месте находится головка, и транслирует эту информацию в координаты X , Y и Z в трехмерном пространстве. Оцифрованные данные в дальнейшем обрабатывают с помощью специальных прикладных программ.

Кроме того, есть и другие технологии трехмерного сканирования: ультразвуковое; магнитное; лазерное.

Из всех систем по оцифровке 3D-объектов *ультразвуковые* (или *сонарные*) дигитайзеры наименее точные и надежные, но при этом самые чувствительные к изменениям в окружающем пространстве. Они представляют собой систему передатчиков, жестко закрепленных по периметру помещения. Передатчики излучают звуковые волны, на основании информации об отражении которых вычисляются координаты точек поверхности 3D-модели. Поскольку скорость звука зависит от атмосферного давления, температуры и других условий (например, влажности), результаты оцифровки одного и того же объекта являются функцией состояния воздуха. Кроме того, данные системы очень восприимчивы к шуму, производимому различным оборудованием (компьютерами, кондиционерами).

Принцип работы *электромагнитных* 3D-дигитайзеров такой же, как у ультразвуковых систем, только для построения пространственной модели вместо звуковых волн используются электромагнитные. Результат работы этих сканеров не зависит от погодных условий, но находящиеся поблизости металлические предметы или источники магнитного поля снижают точность измерений. Даже в специальных помещениях, не содержащих ничего металлического, погрешность магнитных систем составляет не менее 0,7 мм.

Лазерные дигитайзеры обладают самой высокой точностью, но область их применения также имеет значительные ограничения. Большие трудности вызывает сканирование объектов с зеркальными, прозрачными и полупрозрачными поверхностями, а также предметов большого размера либо имеющих впадины или выступы, препятствующие прямому прохождению лазерного

пучка. Лазерные дигитайзеры — это полностью автоматизированные системы.

Устройства ввода акустических сигналов делятся на устройства ввода музыкальных произведений, звуковых эффектов и речи. Для ввода акустических сигналов любого типа необходимо, чтобы ЭВМ была оснащена звуковой картой. Для ввода звуковых эффектов и речи используют микрофон или магнитофон. Музыкальные произведения могут вводиться с магнитофона, через специальный интерфейс с MIDI-устройств или с клавиатуры ЭВМ. Звуковые эффекты могут создаваться программным путем.

Цифровая фотокамера — это устройство, предназначенное для записи изображения, используя фотоэлектрический принцип. При этом полупроводниковая фотоматрица преобразует свет в электрические сигналы, которые трансформируются в цифровые данные, сохраняемые энергонезависимым запоминающим устройством. Изображения, полученные цифровой фотокамерой, могут быть загружены в компьютер для обработки, передачи по вычислительным сетям или для хранения, а затем просмотрены на экране монитора либо отпечатаны.

Чувствительным элементом цифровых фотокамер является ССD-матрица, которая преобразует падающий на нее свет в аналоговый электрический сигнал. Встроенный аналого-цифровой преобразователь конвертирует его в цифровую форму.

Качество электронных фотографий напрямую зависит от количества элементов (пикселей) ССD-матрицы. Чем больше элементов, тем выше разрешение матрицы и тем точнее цветопередача получаемого изображения.

Каждый снимок образует файл. В память цифровой фотокамеры фотоснимок записывается в сжатом виде. В цифровых фотокамерах для компрессии используется формат JPEG. Практически любая камера позволяет проводить съемку в нескольких режимах сжатия (mode). При наименьшей степени сжатия размер файлов получается достаточно большим, зато записываемое в память изображение качественнее. При максимальной степени сжатия размер файлов сокращается, но ухудшается качество изображений.

Сенсорный экран (от англ. touch screen) — координатное устройство, позволяющее путем прикосновения к области экрана монитора производить выбор необходимого элемента данных, меню или осуществлять ввод данных в какое-либо ЭВМ.

Сенсорные экраны наиболее пригодны для организации гибкого и интуитивно понятного интерфейса.

Выделяют четыре базовых типа работы сенсорных экранов: резистивный, емкостный, акустический и инфракрасный.

Резистивный сенсорный экран обладает максимальной стойкостью к загрязнениям и износу, что позволяет не бояться попадания на рабочую поверхность жидкостей, конденсата, паров и надежно работать, когда другие типы экранов выходят из строя. Экран выдерживает 35 млн прикосновений к одной точке.

Резистивные сенсорные экраны нашли применение в составе POS-терминалов, в промышленности, медицине и на транспорте. Прикоснувшись к экрану пальцем, рукой в перчатке, ногтем или кредитной картой, получите точный ответ на прикосновение.

Конструкция экрана представляет собой стеклянную либо акриловую пластину, покрытую двумя токопроводящими слоями. Слои разделены прокладками, которые предохраняют сеть вертикальных и горизонтальных проводников от соприкосновения. В момент нажатия слои контактируют и контроллер регистрирует электрический сигнал. Координаты нажатия определяют исходя из того, на пересечении каких проводников было зарегистрировано воздействие (рис. 8.8). К недостаткам технологии можно отнести снижение на 75–80 % мощности светового потока, излучаемого монитором.

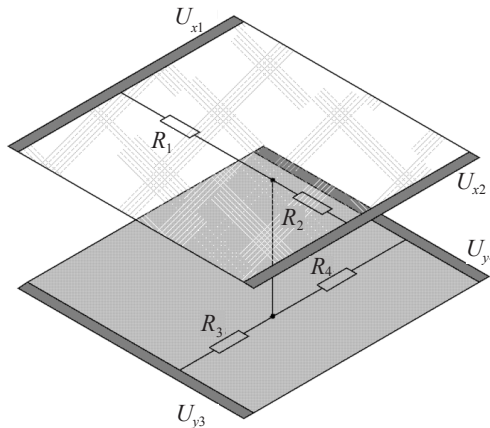


Рис. 8.8. Принцип действия четырехпроводного резистивного сенсорного экрана (R_1 – R_4 – резисторы; U_{x1} , U_{x2} – напряжения по оси x ; U_{y3} , U_{y4} – напряжения по оси y)

В работе *емкостного экрана* человек участвует не только механически. До прикосновения экран обладает некоторым электрическим зарядом. Прикосновение пальца меняет картину заряженности, «оттягивая» часть заряда к точке нажатия. Датчики экрана, расположенные по всем четырем углам, следят за течением заряда в экране, определяя таким образом координаты «утечки» электронов (рис. 8.9).

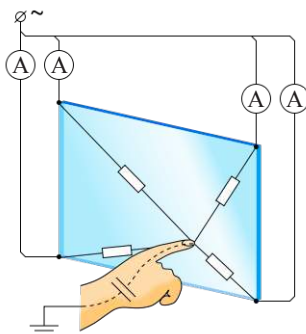


Рис. 8.9. Принцип действия емкостного сенсорного экрана

Емкостные экраны также отличаются высокой надежностью (в них отсутствуют гибкие мембраны) и высокой степенью прозрачности.

Акустический сенсорный экран построен с использованием миниатюрных пьезоэлектрических излучателей звука, не слышимого человеком. Стекло такого экрана постоянно незаметно вибрирует под воздействием излучателей, установленных в трех углах экрана. Специальные отражатели особым образом распространяют акустическую волну по всей поверхности экрана. Прикосновение к экрану меняет картину распространения акустических колебаний, что и регистрируется датчиками. По изменению характера колебаний можно вычислить координаты возмущений, внесенных нажатием на экран. Кроме того, анализируя степень изменения колебаний, можно вычислить силу нажатия на экран. Это следует учитывать при проектировании систем управления промышленным оборудованием, например для плавного изменения скорости вращения двигателей и других параметров.

В ряде случаев к качеству изображения, воспроизводимого *инфракрасным сенсорным экраном*, предъявляют строгие требова-

ния. Для определения точки касания используют две линейки светодиодов, расположенные по вертикали и горизонтали, и две линейки фотодиодов, расположенные на противоположных сторонах экрана.

Каждому светодиоду соответствует свой фотодиод. Работает такая оптическая пара следующим образом. При подаче напряжения на светодиод он излучает невидимый для человека инфракрасный свет в пределах очень небольшого телесного угла, чтобы попасть на «свой» фотодиод и «не задеть» соседние. Любое препятствие (например, касающийся экрана палец руки), частично или полностью перекрывающее световой луч, приводит к уменьшению или прекращению прохождения электрического тока через соответствующий фотодиод. Это изменение фиксируется микроконтроллером, позволяя вычислить координаты касания с высокой точностью. Обычно светодиод (и соответственно фотодиод) в линейке имеет размеры порядка 2,5 мм, т. е. на каждый квадратный сантиметр панели приходятся четыре горизонтальных и четыре вертикальных сканирующих луча. Однако механизмы интерполяции, используемые микроконтроллером, позволяют вычислять положение препятствия с большей точностью. Инфракрасный сенсорный экран выполнен в виде рамки, которая не имеет никаких стекол или прозрачных пленок. Поэтому изменение яркости, контраста и цветопередачи изображения, а также появление дополнительных бликов исключено, что является несомненным достоинством экрана.

8.7. Устройства вывода данных

Устройства вывода данных (информации) включают принтеры, плоттеры, видеосистемы.

Принтеры. Это внешние устройства ЭВМ, предназначенные для вывода информации на твердый носитель в символьном или графическом виде.

По принципу действия различают матричные, лазерные, светодиодные и струйные принтеры.

Матричные принтеры – это простейшие печатающие устройства. Данные выводятся на бумагу в виде оттиска, образующегося при ударе цилиндрических стержней (иглолок) через красящую ленту. Качество печати матричных принтеров напрямую зависит от количества иглолок в печатающей головке. Наибольшее рас-

пространение имеют 9-игольчатые и 24-игольчатые матричные принтеры. Последние позволяют получать оттиски документов, не уступающие по качеству документам, исполненным на пишущей машинке.

Производительность работы матричных принтеров оценивают по количеству печатаемых знаков в секунду (cps — characters per second).

Лазерные принтеры обеспечивают высокое качество печати. Отличаются высокой скоростью печати, которая измеряется в страницах в минуту (ppt — page per time). Как и в матричных принтерах, итоговое изображение формируется из отдельных точек.

Принцип действия лазерных принтеров следующий:

- в соответствии с поступающими данными лазерная головка испускает световые импульсы, которые отражаются от зеркала и попадают на поверхность светочувствительного барабана;
- горизонтальная развертка изображения выполняется вращением зеркала;
- участки поверхности светочувствительного барабана, получившие световой импульс, приобретают статический заряд;
- барабан при вращении проходит через контейнер, наполненный красящим составом (тонером), и тонер закрепляется на участках, имеющих статический заряд;
- при дальнейшем вращении барабана происходит контакт его поверхности с бумажным листом, в результате чего тонер переносится на бумагу;
- лист бумаги с нанесенным на него тонером протягивается через нагревательный элемент, в результате чего частицы тонера спекаются и закрепляются на бумаге.

Принцип действия *светодиодных принтеров* похож на принцип действия лазерных принтеров. Разница заключается в том, что источником света является не лазерная головка, а линейка светодиодов. Поскольку эта линейка расположена по всей ширине печатаемой страницы, отпадает необходимость в механизме формирования горизонтальной развертки, и вся конструкция получается проще, надежнее и дешевле. Типичная величина разрешения печати для светодиодных принтеров составляет порядка 600 dpi.

В *струйных принтерах* изображение на бумаге формируется из пятен, образующихся при попадании капель красителя на бумагу. Выброс микрокапель красителя происходит под давлением, кото-

рое развивается в печатающей головке за счет парообразования. В некоторых моделях капля выбрасывается щелчком в результате пьезоэлектрического эффекта. Этот метод позволяет обеспечить более стабильную форму капли, близкую к сферической.

Качество печати изображения во многом зависит от формы капли и ее размера, а также от характера впитывания жидкого красителя поверхностью бумаги. В таких условиях особую роль играют вязкостные свойства красителя и свойства бумаги.

К положительным свойствам струйных печатающих устройств следует отнести относительно небольшое количество движущихся механических частей. Основным недостатком, по сравнению с лазерными принтерами, является нестабильность получаемого разрешения.

Плоттеры. Предназначены для вывода графической информации на твердый носитель (бумагу). Планшетный плоттер имеет линейку, по которой может перемещаться печатающий механизм. Перемещение линейки сдвигает печатающий механизм по вертикали, а перемещение механизма по линейке сдвигает его по горизонтали. Благодаря этому можно установить печатающий механизм в любую точку планшета. На планшете крепится лист бумаги. Плоттер может воспроизводить на бумаге очень сложные штриховые изображения, но работает очень медленно. Для управления плоттером разработаны специальные алгоритмические языки.

Плоттеры нашли применение в строительном и машиностроительном черчении, картографии, метеорологии. Для работы с ними существуют специальные пакеты прикладных программ.

Системы визуального отображения информации (видеосистемы). Видеосистемы предназначены для оперативного отображения информации, доведения ее до сведения оператора ЭВМ. Обычно они состоят из двух частей: монитора и адаптера. Монитор служит для визуализации изображения, адаптер — для связи монитора с микропроцессорным комплектом.

Классификацию мониторов можно провести по используемым физическим эффектам, принципу формирования изображения на экране, способу управления, длительности хранения информации на экране, цветности.

По *принципу формирования изображения* мониторы делят на плазменные, электролюминесцентные, жидкокристаллические, дисплей с эмиссией полей и электронно-лучевые.

Плазменные, электролюминесцентные, жидкокристаллические мониторы и дисплеи с эмиссией полем относятся к дисплеям с плоским экраном. Для них характерно то, что экран имеет малые физические размеры и не мерцает. Мониторы этого вида имеют малую массу и незначительное потребление энергии, большую механическую прочность и длительный срок службы.

Жидкокристаллические мониторы пассивные. Они работают только при наличии постороннего источника света: отраженного или проходящего. Используют способность жидких кристаллов изменять свою оптическую плотность или отражающую способность под воздействием электрических сигналов.

По *длительности хранения информации на экране* мониторы делят на регенерируемые и запоминающие.

В регенерируемых мониторах изображение после однократной прорисовки держится на экране недолго (доли секунды), постепенно угасая. Угасание изображения иногда заметно на глаз, например нижние строки могут быть ярче верхних. Для поддержания постоянной яркости изображение приходится повторно прорисовывать (регенерировать). Чтобы яркость в различных частях экрана не очень отличалась, и для снижения полосы пропускания применяют чересстрочную развертку: при каждой регенерации сначала рисуются нечетные строки, а затем четные. Регенерируемые мониторы незаменимы при визуализации быстротекущих динамических процессов.

В запоминающих мониторах после однократной прорисовки изображение держится на экране в течение нескольких часов. Для его стирания приходится подавать на экран специальное стирающее напряжение.

Запоминающие мониторы эффективны там, где выведенное изображение нуждается в длительной обработке, например подвергается редактированию или должно быть воспринято (изучено) оператором.

По *способу управления яркостью луча* мониторы делят на цифровые и аналоговые. В цифровых мониторах для управления яркостью на сетку подаются дискретные сигналы, которые в зависимости от настройки могут полностью запирают трубку или полностью отпирать ее, снижать яркость до $1/2$ или обеспечивать полную яркость и т. д.

В аналоговых мониторах на сетку подается непрерывный (аналоговый) сигнал, который может плавно изменять яркость от полного запыриания до полного отпыриания.

По *цветности изображения* различают монохромные и цветные мониторы. В цветных мониторах в качестве основных цветов применяются красный (Red), зеленый (Green) и синий (Blue), в связи с чем они получили название RGB-мониторы.

Связь ЭВМ с монитором осуществляют с помощью *адаптера* – устройства, которое должно обеспечивать совместимость различных мониторов с микропроцессорным комплектом ЭВМ. В начальный период существования персональных компьютеров адаптеры старались стандартизировать, чтобы в полной мере обеспечить совместимость различных по конструкции мониторов с ЭВМ. Были разработаны следующие стандарты:

- MDA – монохромный дисплейный адаптер;
- CGA – цветной графический адаптер;
- MGA – монохромный графический адаптер;
- EGA – улучшенный графический адаптер;
- VGA – видеографическая матрица;
- SVGA – улучшенная видеографическая матрица.

Адаптер SVGA не стандартизован, вследствие чего каждая фирма, выпускающая мониторы, обязательно снабжает их драйверами, позволяющими работать с различными адаптерами.

Исторически сложилось, что дисплеи могут работать в одном из двух режимов: символьном и графическом.

В символьном режиме на экран может выводиться ограниченный состав символов, имеющих четко определенный графический образ: буквы, цифры, знаки пунктуации, математические знаки и знаки псевдографики. Состав этих символов определен системой кодирования, применяемой в данной ЭВМ. В Robotron 1715 состав символов определяется кодом КОИ-7, в ЕС ЭВМ – кодом ДКОИ, в IBM PC – кодом ASCII, который в последнее время теснится кодом UNICODE.

Для вывода символа на экран дисплея сначала определяют позицию, в которой должен появиться символ (номер символа в строке и порядковый номер строки), а затем по коду символа определяют его форму, которая и высвечивается на экране. Предельное количество символов, одновременно размещаемых на экране, называют *информационной емкостью экрана*.

В символьном режиме на экране монитора может быть высвечено 40, 80 или 132 символа в строке. Всего на экране помещается 25, 50 или 60 строк. Форму выводимого символа определяют знакогенератором дисплея, в котором хранятся коды формы всех символов ASCII или UNICODE.

В графическом режиме изображение на экране формируется из отдельных точек (пиксел), имеющих свои адреса (номер пиксела в строке и номер пиксельной строки). В простейшем случае каждому пикселу экрана соответствует один пиксел видеопамяти, который и определяет, светлым или темным должна быть соответствующая точка на экране. Кроме того, если необходимо указывать цвет пиксела, количество битов видеопамяти, характеризующих каждый пиксел, приходится увеличивать. Поэтому для графического режима требуется бóльшая память, чем для символьного, при той же разрешающей способности экрана.

Основу адаптера любого типа составляет видеопамять: обычная динамическая (DRAM), в которой для доступа к биту надо задать адрес строки и столбца; FPM DRAM (Fast Page Mode DRAM), в которой адрес строки задается один раз для нескольких доступов к близким элементам памяти; EDO DRAM (Extended Data Out DRAM), в которой модифицированы схемы тактирования, за счет чего новое обращение к памяти может начаться до завершения предыдущего; BEDO DRAM (Burst Extended Data Out DRAM), в которой EDO-память скомбинирована с конвейерной технологией и специальными триггерами с защелкой, что позволяет заметно сократить время доступа; RDRAM (RAMBus DRAM), в которой для доступа к данным вместо обычных 32- или 64-битных шин используется 8-битный скоростной интеллектуальный канал; специальная двухпортовая (VRAM), допускающая одновременное обращение как со стороны системной магистрали, так и со стороны монитора.

Кроме видеопамяти, в состав адаптера входят блок сопряжения с монитором, блок управления, различные ускорители (графический, 3D-ускоритель и др.), которые предназначены для выполнения вычислительных операций без обращения к процессору.

В *мультимедийном проекторе* проекционная лампа, жидкокристаллическая матрица и оптическая система конструктивно размещаются в одном корпусе. По принципу действия мультимедийный проектор создает изображение с помощью мощной

проекторной лампы и встроенного в проектор электронно-оптического модулятора, управляемого сигналом видеографического адаптера ЭВМ, а затем посредством оптической системы проецируется на внешний экран. Основным отличием в мультимедийных проекторах являются конструкция модулятора и способы построения и переноса изображения на экран.

8.8. Устройства хранения данных и обмена данными

Устройства хранения данных делят на два вида: внешние (периферийные) и внутренние.

К внешним устройствам относят магнитные диски, диски CD, DVD, BD, стримеры, жесткий диск (винчестер), а также флеш-карту. Также большинство устройств внешней памяти может переноситься с одного компьютера на другой. Главный их недостаток в том, что они работают медленнее устройств внутренней памяти.

К внутренним устройствам относятся оперативная память, кэш-память, CMOS-память, BIOS. Главное их достоинство — скорость обработки информации. Однако в то же время устройства внутренней памяти довольно дорогостоящие.

К *устройствам обмена данными* относят модемы и сетевые платы.

Модем (МОдулятор + ДЕМОдулятор) — это устройство, предназначенное для обмена информацией между удаленными компьютерами по каналам связи. При этом под каналом связи понимают физические линии (проводные, оптоволоконные, кабельные, радиочастотные), способ их использования (коммутируемые и выделенные) и способ передачи данных (цифровые или аналоговые сигналы).

В зависимости от типа канала связи устройства приема-передачи подразделяют на радиомодемы, кабельные модемы и пр. Наибольшее распространение получили модемы, ориентированные на подключение к коммутируемым телефонным каналам связи.

Цифровые данные, поступающие в модем из компьютера, преобразуются в нем путем модуляции (по амплитуде, частоте, фазе) в соответствии с избранным стандартом (протоколом) и направляются в телефонную линию. Модем-приемник, понимающий данный протокол, осуществляет обратное преобразование

(демодуляцию) и пересылает восстановленные цифровые данные в свой компьютер. Таким образом обеспечивается удаленная связь между компьютерами и обмен данными между ними.

Основные параметры модемов:

- производительность (бит/с);
- поддерживаемые протоколы связи и коррекции ошибок;
- шинный интерфейс, если модем внутренний (ISA или PCI).

От производительности модема зависит объем данных, передаваемых в единицу времени.

От поддерживаемых протоколов зависит эффективность взаимодействия данного модема с сопредельными модемами (вероятность того, что они вступят во взаимодействие друг с другом при оптимальных настройках).

От шинного интерфейса в настоящее время пока зависит только простота установки и настройки модема (в дальнейшем при общем совершенствовании каналов связи шинный интерфейс начнет оказывать влияние и на производительность).

Сетевая плата (в англоязычной среде NIC – Network Interface Controller/card) – периферийное устройство, позволяющее компьютеру взаимодействовать с другими устройствами сети.

В настоящее время в персональных компьютерах и ноутбуках контроллер и компоненты, выполняющие функции сетевой платы, довольно часто интегрированы в материнские платы для удобства, в том числе для унификации драйвера и удешевления всего компьютера в целом.

Обмен данными с другими микроконтроллерами в управляющей системе, содержащей несколько совместно работающих микроконтроллеров, может осуществляться по последовательному каналу или путем совместного использования внешней памяти.

Обмен данными по последовательному каналу выполняют с использованием последовательного порта (SP – Service Processor). Обмен производится путем посылки отдельных кадров, каждый из которых содержит стартовый бит, семь или восемь информационных битов и один стоповый бит. В состав кадра должен быть включен дополнительный бит, который используют для контроля по четности правильности пересылки данных или для различения кадров, содержащих адреса абонентов, и кадров, содержащих данные, при включении контроллера в простейшую локальную сеть.

Последовательный порт также может осуществлять последовательный ввод или вывод байтов с использованием внешних сдвигающих регистров, которые в данном случае выполняют функции дополнительных параллельных портов ввода или вывода.

У некоторых микроконтроллеров в число периферийных устройств входит второй последовательный порт (SSIO), с помощью которого осуществляется непосредственный обмен байтами между двумя микроконтроллерами путем последовательной передачи байта и сопровождающей серии импульсов сдвига. Порт SSIO содержит два последовательных канала, каждый из которых может работать в режиме передачи или в режиме приема. Максимальная скорость обмена (на частоте 16 МГц): в асинхронном режиме – 1 МБод (1 МБод = 1 Мбит/с); в синхронном режиме – 4 МБод.

Микроконтроллеры почти всех модификаций имеют аппаратные средства, обеспечивающие совместное использование внешней памяти несколькими микроконтроллерами. Согласование работы микроконтроллеров при обращении к внешней памяти реализуется с помощью сигналов HOLD (захват шин), HLDA (подтверждение захвата), BREQ (сигнал запроса магистрали) и дополнительной внешней аппаратуры.

У некоторых микроконтроллеров имеется «подчиненный» порт (slave port), предназначенный для обмена данными с центральным процессором в иерархической управляющей системе. Через «подчиненный» порт микроконтроллер подключается непосредственно к системной магистрали микропроцессорной системы. Обмен данными происходит под управлением центрального процессора, который обращается к микроконтроллеру, как к собственному порту ввода и вывода. При необходимости передать данные в центральный процессор микроконтроллер посылает запрос прерывания.

Контрольные вопросы и задания

1. Назовите структурные элементы ЭВМ.
2. Дайте определение понятиям «архитектура ЭВМ», «арифметико-логическое устройство».
3. Как работает устройство управления?

4. Расскажите о назначении и видах памяти.
5. Чем отличаются архитектура ЭВМ и микроЭВМ?
6. Дайте классификацию периферийных устройств.
7. Сформулируйте основные принципы работы устройств ввода.
8. Перечислите классы сканеров.
9. Назовите типы дисплеев, физические принципы формирования изображения.
10. Сравните принтеры различных технологий.
11. Назовите виды внешней памяти персонального компьютера.

9. АРХИТЕКТУРА МИКРОПРОЦЕССОРОВ

9.1. Общие сведения

В данной главе речь пойдет о микропроцессорах (МП) или процессорах (в первом случае подчеркивается, что процессор – интегральная микросхема), занимающих самую верхнюю строчку в иерархии интегральных микросхем как по количеству транзисторов, расположенных на кристалле, так и по своим функциональным возможностям. В компьютере МП является самым главным элементом, так как отвечает за все вычислительные процессы (благодаря встроенному АЛУ) и управление другими элементами системы.

Первый микропроцессор, носивший название i4004, появился в 1971 г. и предназначался для работы в калькуляторе, работал на частоте 759 кГц и содержал 2300 транзисторов. Выпустила его фирма Intel, до сих пор являющаяся одним из лидеров среди разработчиков микропроцессоров.

Дальнейшая эволюция микропроцессоров привела к их повсеместному использованию в самых разнообразных системах управления и обработки данными, и в первую очередь персональных компьютерах.

9.2. Классификации микропроцессоров

Микропроцессоры классифицируют по следующим направлениям.

1. По числу больших интегральных схем, задействованных в микропроцессорном комплекте: однокристалльные; многокристалльные; многокристалльные секционные.

Однокристалльные микропроцессоры представляют собой один кристалл, на котором реализованы все аппаратные и функциональные блоки процессора (арифметико-логический, регистров, стека, прерываний, интерфейса, управления и синхронизации и др.).

Многокристалльные микропроцессоры появились в связи с тем, что возможности однокристалльных ограничены технологическими ресурсами их кристалла, и порой реализовать многопроцессорный комплект быстрее, чем перейти на новую технологию получения высокопроизводительного однокристалльного микропроцессора. В связи с этим для повышения производительности отдельных устройств логическую структуру проектируемого микропроцессора подразделяют на функционально законченные части, которые затем реализуют в виде отдельных сверхбольших интегральных схем (СБИС) для выполнения своих определенных функций автономно.

Многокристалльные секционные микропроцессоры выполняют ту же задачу, что и многокристалльные. Отличие от последних состоит в способе разбиения логической структуры процессора. Обычно микропроцессорная секция – это СБИС, предназначена для обработки нескольких разрядов данных или выполнения определенных управляющих операций. Секционность БИС микропроцессора определяет возможность «наращивания» разрядности обрабатываемых данных или усложнения устройств управления микропроцессора при параллельном включении большего числа БИС.

2. По функционально-структурным особенностям и областям применения микропроцессоры разделяют:

- на универсальные с CISC-архитектурой;
- универсальные с RISC-архитектурой;
- специализированные;
- микроконтроллеры.

Универсальные микропроцессоры с CISC-архитектурой (Complicated Instruction Set Computer – компьютер со сложным набором команд) применяют главным образом в персональных компьютерах и серверах. Кроме того, микропроцессоры этого семейства широко используют в устройствах управления, встраивают в различные приборы и системы.

Универсальные микропроцессоры с RISC-архитектурой (Reduced Instruction Set Computer – компьютер с сокращенным набором ко-

манд) применяют в основном в рабочих станциях, мощных серверах и при построении мультипроцессорных систем.

В классе *специализированных микропроцессоров* в настоящее время наиболее широко представлены DSP (Digital Signal Processor – процессор цифровой обработки сигналов).

Кроме DSP, выпускают микропроцессоры, специализированные для передачи информации в системах коммуникации, – коммуникационные контроллеры для обработки графической информации и др.

Микроконтроллеры являются наиболее массовым представителем микропроцессорной техники. Интегрируя на одном кристалле высокопроизводительный процессор, память и набор периферийных устройств, микроконтроллеры позволяют с минимальными затратами реализовать широкий спектр систем управления различными системами и процессами. Так, использование микроконтроллеров в объектах управления и обработки информации обеспечивает исключительно высокие показатели эффективности при низкой их стоимости. Во многих случаях система вообще может состоять только из одного микроконтроллера.

3. По *виду обрабатываемых входных сигналов* различают цифровые и аналоговые микроконтроллеры.

Разбиение на цифровые и аналоговые связано с тем, что привычные цифровые микропроцессоры могут иметь встроенные в свою структуру *аналого-цифровые* и *цифроаналоговые* преобразователи, которые «переводят» их из категории цифровых в категорию аналоговых микропроцессоров.

Например, входные аналоговые данные передаются в МП через аналого-цифровой преобразователь в цифровой форме, обрабатываются в цифровом виде, после чего возвращаются в систему (после обратного преобразования) в аналоговом виде.

С архитектурной точки зрения такие микропроцессоры представляют собой ничто иное, как аналоговые функциональные преобразователи сигналов и называются *аналоговыми микропроцессорами*. Они выполняют функции любой аналоговой схемы. При этом применение аналогового микропроцессора значительно повышает точность обработки аналоговых сигналов и их воспроизводимость, а также расширяет функциональные возможности за счет программной «настройки» цифровой части микропроцессора на различные алгоритмы обработки сигналов.

4. По *характеру временной организации работы* микропроцессоры делят на синхронные и асинхронные.

В **синхронных** микропроцессорах начало и конец выполнения операций задаются устройством синхронизации (управления), время выполнения операций при этом не зависит от вида выполняемых команд и величин операндов.

В **асинхронных** микропроцессорах начало выполнения каждой следующей операции определяется сигналом фактического окончания выполнения предыдущей. Чтобы эффективно использовать все узлы микропроцессорной системы, в их состав вводят электронные цепи, обеспечивающие автономное функционирование устройств. Закончив выполнение какой-либо команды, узел вырабатывает сигнал запроса, означающий его готовность к выполнению следующей операции.

Роль распределителя заданий играет память, которая в соответствии с заранее определенными приоритетами выполняет поочередно запросы устройств по обеспечению их командами и данными.

5. По *организации структуры микропроцессорных систем* различают одномагистральные и многомагистральные микропроцессоры.

В **одномагистральных** микропроцессорных системах все устройства имеют один интерфейс и подключаются к единой информационной магистрали, по которой передаются данные, адреса и управляющие сигналы.

В **многомагистральных** микропроцессорных системах устройства подключаются каждый к своей информационной магистрали, что позволяет осуществить одновременную передачу данных и сигналов управления сразу многим устройствам. Такая организация систем усложняет их конструкцию, но позволяет увеличить производительность.

6. По *количеству выполняемых программ* выделяют однопрограммные и многопрограммные микропроцессоры.

В **однопрограммных** микропроцессорах в текущий момент времени выполняется только одна программа. Переход к выполнению следующей программы происходит после завершения предыдущей.

В **многопрограммных** (мультипрограммных) микропроцессорах, по определению, одновременно выполняется несколько

программ. Кроме повышения производительности микропроцессорных систем, это позволяет осуществлять одновременный контроль за состоянием и управлением большим числом источников или приемников информации.

9.3. Архитектура микропроцессора

Для комплексной оценки свойств отдельных микропроцессоров введено понятие *архитектуры*. Под этим понимают схемотехническую и логическую организацию микропроцессора.

Лучше понять работу микропроцессора можно, рассмотрев его в составе системы, с которой он взаимодействует. Наибольшее применение получили микропроцессорные системы (в том числе и в ЭВМ), использующие *магистрально-модульную структуру построения* (рис. 9.1). Такая структура характеризуется четко выраженной общей шиной (ОШ), к которой подсоединяются все устройства системы, выполняемые в виде отдельных модулей.

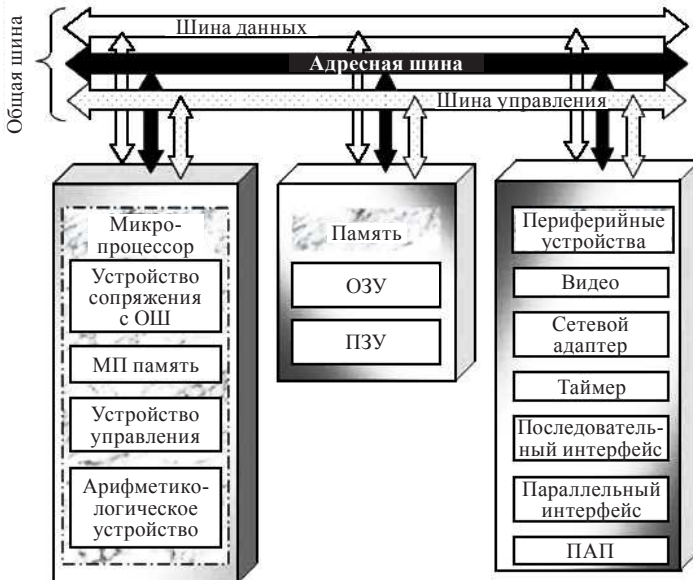


Рис. 9.1. Типовая архитектура микропроцессора, включенного в ЭВМ с магистрально-модульной структурой

Микропроцессор (МП) координирует работу всех устройств цифровой системы с помощью общей (системной) шины, которая

включает в себя *шину управления*, *адресную шину* (служащую для выбора определенной ячейки памяти, порта ввода или порта вывода и др.) и *шину данных* (по которой осуществляется двунаправленная пересылка данных к микропроцессору и обратно).

В состав МП (см. рис. 9.1) входит следующий стандартный набор блоков:

- *устройство сопряжения с общей шиной (ОШ)* — представляет собой интерфейс связи МП с системной шиной;

- *микропроцессорная память* — включает в себя кэш-память и регистры равного назначения (общие, сдвиговые и т. п.), служит для хранения исходных, промежуточных и конечных данных работы процессора;

- *устройство управления (УУ)* — предназначено для управления и контроля за выполнением операций, выбором команд программы в необходимой последовательности, их дешифровки и обработки;

- *арифметико-логическое устройство (АЛУ)* — предназначено для выполнения арифметических и логических операций над числовой и символьной информацией, поступающей в МП; является самым главным блоком процессора.

Имея в своем составе вышеперечисленные блоки, МП может выполнять следующие операции:

- *арифметические*: сложение, вычитание, деление, умножение;
- *логические*: И, ИЛИ, исключающее ИЛИ, отрицание (инверсия), сдвиг;

- *пересылки*;

- *управление процессором*: переключение регистров разного назначения, управление энергопотреблением и т. п.

Выполнение этих операций МП осуществляет под воздействием поступающих на его вход микрокоманд. К ним относятся, например, команды изменения содержимого ячеек памяти, обновления содержимого регистров общего назначения, выполнения операций с памятью-стеком, выполнения операций в АЛУ, изменения содержимого регистра состояния и многие другие. Разнообразие команд управления МП представляет собой его *систему команд*. Чем больше команд выполняет МП, тем эффективнее может быть реализован алгоритм решения какой-либо прикладной задачи. Системы команд различных МП отличаются друг от друга, так как каждый МП предназначен для определенной области применения. Кроме отличий системы команд,

МП могут различаться организацией режима прерываний, т. е. возможностью МП использовать свободные промежутки времени для выполнения другой части программы.

Рассмотрим в качестве примера особенности архитектур процессоров K8 AMD и Intel Core P8. Эти двухъядерные процессоры компаний AMD и Intel представлены соответственно на рисунках 9.2, 9.3.

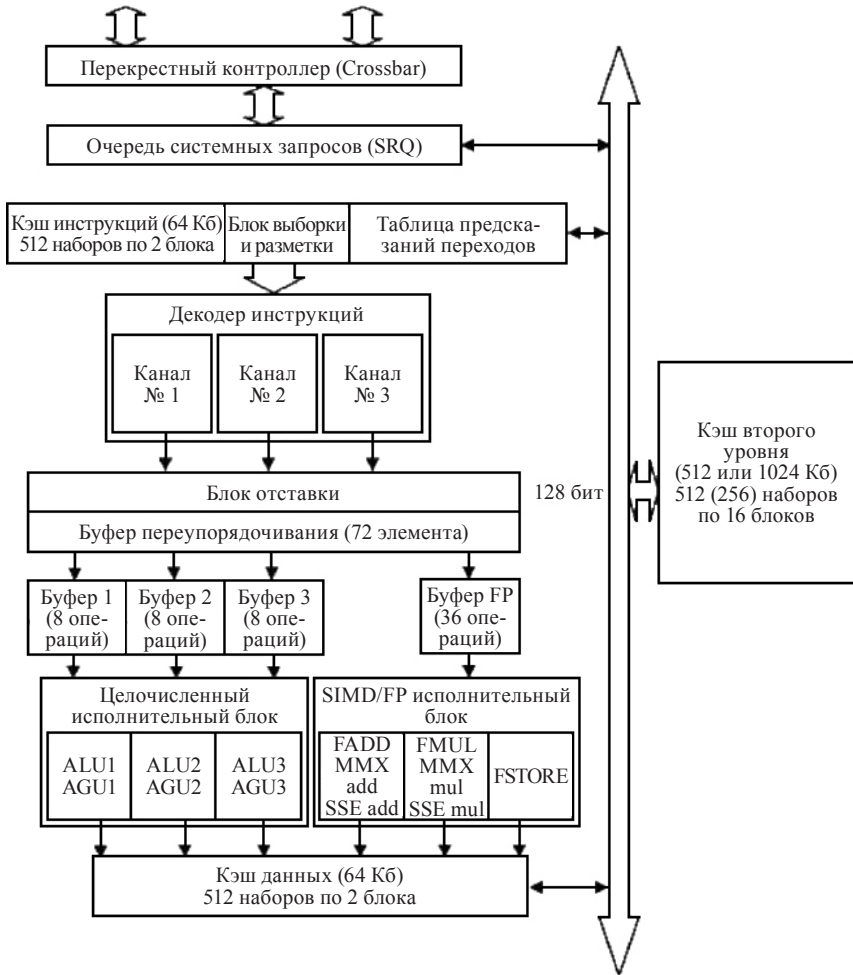


Рис. 9.2. Архитектура процессора AMD K8

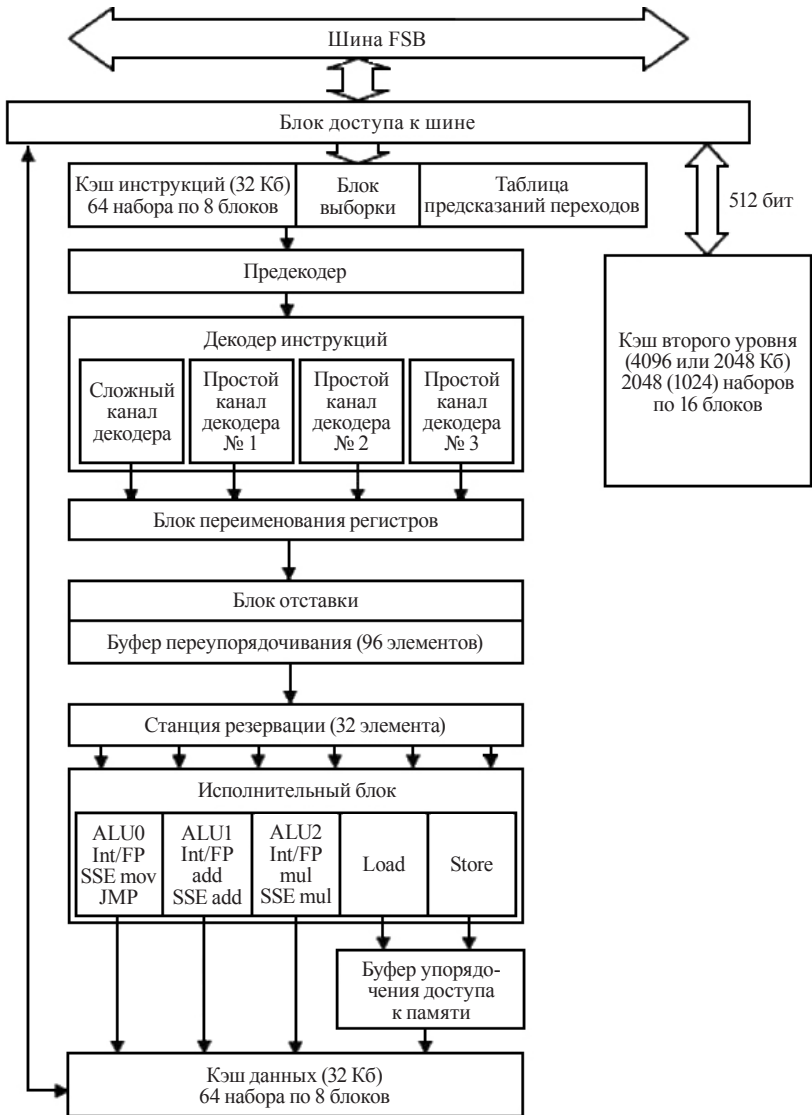


Рис. 9.3. Архитектура процессора Intel Core

В основе микроархитектуры процессоров K8 и P8 лежат одни и те же принципы построения. Оба ядра являются суперскаляр-

ными и конвейерными, т. е. параллельно обрабатывают несколько команд с переопределением порядка их следования в целях оптимальной загрузки вычислительных ресурсов.

Выборка команд в обоих процессорах производится из отдельного кэша первого уровня для инструкций (*L1I*) и выбора данных из кэша данных (*L1D*). Оба кэша организованы одинаково, с использованием блочно-ассоциативного принципа выборки. Чтение команд из кэша выполняется с опережением и предсказанием условных переходов. Имеется также расположенный на кристалле полноскоростной кэш второго уровня. Обращение к памяти также выполняется с опережением, чтобы минимизировать простой процессора в ожидании данных.

В архитектурах представленных процессоров имеются отличия. В процессоре AMD K8 обработка команд целочисленных и с плавающей запятой выполняется на отдельных устройствах. Для целочисленных команд предусмотрены три практически равноценных блока, выполняющих как арифметико-логические, так и адресные операции. Операции с плавающей запятой, а также SIMD-команды MMX/SSE/3DNow тоже выполняются в трех блоках, но рассчитаны на разные действия. Декодирование (преобразование во внутренние микрооперации) также выполняется по три. Именно так, тройками, они и «путешествуют» по вычислительным ресурсам. Процессор еще на этапе выборки определяет, как будут выполняться команды. Эффективность загрузки вычислительных блоков при этом не будет максимальной, но количество и длительность стадий, через которые проходят команды в процессе выполнения, уменьшаются. При выполнении ненужных команд, что случается при неправильном предсказании направления ветвления в программе, процессор AMD K8 очень быстро очищает и перезагружает конвейер. Согласно общепринятому мнению, именно эта особенность является главным преимуществом процессора AMD K8 перед Pentium 4.

В вычислительном ядре процессора Intel Core P8 выборка команд идет четырьмя, а не тремя устройствами одновременно, причем процессор «научился» в некоторых случаях объединять две исходные команды в одну. Далее полученные микрооперации, также пригодные к объединению в пары, помещаются в общую очередь большого размера, откуда поступают на выполнение. Два исполнительных блока занимаются доступом в память

и три — выполнением, причем команд как целочисленных, так и с плавающей запятой. Отметим возможность обрабатывать 128-битные и упакованные команды SSE целиком, а не разбивать их на две отдельные операции, как в случае с процессором AMD.

Несмотря на более сложный способ диспетчеризации команд, общая длина конвейера у Intel Core P8 увеличилась незначительно. Кроме того, в данном процессоре имеется сложный блок предсказания ветвлений, который использует самые передовые алгоритмы, сводящие к минимуму вероятность неправильного предсказания и следующей за ним перезагрузки конвейера.

Оба процессора могут эффективно обрабатывать 64-разрядные инструкции AMD 64/Intel 64.

Кэши первого уровня у сравниваемых процессоров отличаются. У AMD K8 оба кэша имеют размер по 64 Кб, но содержат всего два блока в наборе, у Intel Core P8 кэши вдвое меньше, но их ассоциативность (число блоков в наборе) в 4 раза выше. Поэтому эффективность кэширования данных и команд обоими процессорами примерно одинакова. У Intel Core P8 размер кэша второго уровня намного больше, но у AMD K8 имеется встроенный контроллер памяти, обеспечивающий намного меньшие задержки при доступе к памяти. Процессор работает через системную шину FSB, а значит, использует внешний контроллер памяти. С другой стороны, эффективный механизм доступа к памяти с опережением (аппаратная предвыборка) позволяет ему значительно уменьшить влияние больших задержек на общую производительность, а механизм устранения неоднозначностей при доступе к памяти — не обращаться к памяти без необходимости. Таким образом, разработчики Intel Core P8 смогли добиться хорошей скорости работы с памятью другими способами.

Оба процессора являются двухъядерными, но по принципу объединения ядер отличаются (рис. 9.4). В процессоре AMD K8 два ядра соединены в точке *SRQ* (System Request Queue — очередь системных запросов). Заметим, что этот блок изначально был рассчитан на подключение двух процессоров. К *SRQ* подключен общий на два ядра контроллер Crossbar, который соединяет *SRQ* с контроллером памяти и контроллером шины HyperTransport. Кэш второго уровня у каждого ядра свой, поскольку работа с ним выполняется минуя *SRQ*.

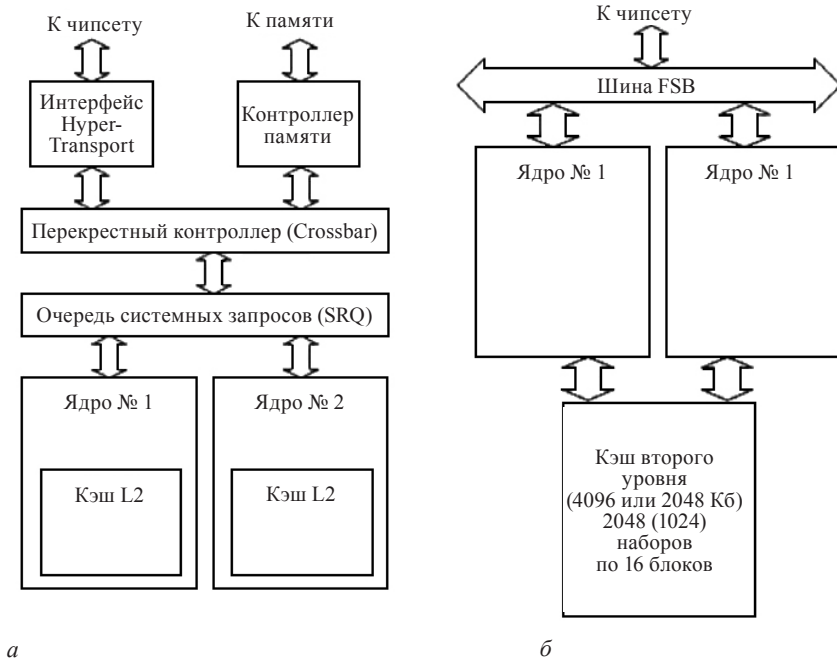


Рис. 9.4. Сравнение двухъядерных архитектур:
 а – AMD K8; б – Intel Core P8

У процессора Intel Core P8 архитектура иная, в нем два ядра объединены в точке блока доступа к системной шине. К этому же блоку подключен и кэш второго уровня, поэтому разработчики решили оставить один кэш на два процессора. С одной стороны, это позволяет не только исключить необходимость синхронизации двух кэшей (обеспечения идентичности совпадающих данных), но и добиться более эффективного распределения объема между ядрами. То ядро, которое чаще обращается к кэшу второго уровня, и заполнит его своими данными. Однако, с другой стороны, оба ядра вынуждены обращаться к одному кэшу одновременно.

Таким образом, сравнивая архитектуры представленных процессоров, можно заключить, что особенности архитектур каждого из них позволяют им получить дополнительные преимущества в работе.

Например, для процессора Intel Core P8 характерна наиболее эффективная выборка и распределение команд по исполнительным устройствам, что дает более высокую производительность в расчете на 1 МГц частоты. У процессора AMD K8 имеется встроенный контроллер памяти, обеспечивающий намного меньшие задержки при доступе к памяти.

9.4. Основные характеристики микропроцессоров

К основным характеристикам микропроцессора (МП) относятся его разрядность и тактовая частота, а также наличие определенного объема кэш-памяти и особенности построения архитектуры процессора.

Разрядность — это максимально возможное число одновременно обрабатываемых МП двоичных разрядов — бит.

Например, в настоящий момент в персональных компьютерах на смену 32-разрядным процессорам пришли 64-разрядные. В данном случае говорится о разрядности внутренних общих регистров МП, однако следует учитывать, что еще существуют не менее важные для определения характеристики процессора: *разрядность шины данных* (определяющая скорость передачи информации) и *разрядность шины адреса* (определяющая размер адресного пространства).

Тактовая (внутренняя) частота МП определяет максимальное время, требующееся на переключение микроэлементов (транзисторов) процессора, поэтому она определяет скорость его работы. Обмен с системной шиной осуществляется на внешней частоте (частоте шины), обычно меньше внутренней. Частное между внешней и внутренней частотой выражается коэффициентом умножения и учитывается при подключении процессора.

Все выпускаемые в настоящее время процессоры снабжены внутренней быстродействующей кэш-памятью первого уровня Level 1 (L1) и второго уровня Level 2 (L2). Чем больше объем кэш-памяти и частота, на которой она работает, тем быстрее обрабатываются процессором команды и данные.

В качестве примера, отражающего основные характеристики процессоров, в таблице приведены данные для рассмотренных выше МП Intel Core P8 и AMD K8.

Характеристики процессоров Intel Core P8 и AMD K8

Параметр	Intel Core 2 Duo E630Q	AMD Athlon 64 X2 4200+
Тактовая частота, ГГц	1,83	2,2
Частота шины, МГц	266	200
Множитель	7	11
Кэш L2, Кб	1 × 1024	2 × 512
Технология, нм	65	90
Энергопотребление, Вт	65	89

Среди современных процессоров, отличающихся более высокими характеристиками, можно выделить, например, 18-ядерный процессор компании Intel: Core I9-7980XE.

Выполнен процессор по 14 нм технологическим нормам, частота процессора 2,6 ГГц; объем кэш-памяти: L2 – 18 × 1024 Кб; потребляемая мощность – 165 Вт.

Контрольные вопросы и задания

1. Какую роль играет микропроцессор в ЭВМ?
2. По каким признакам классифицируют микропроцессоры?
3. Какие основные блоки присутствуют в архитектуре любого микропроцессора?
4. Для чего предназначено АЛУ?
5. Что означает понятие «система команд» микропроцессора?
6. Одинакова ли для всех микропроцессоров система команд? Почему?
7. Назовите основные микропроцессорные характеристики.
8. Что имеют в виду, когда говорят о разрядности микропроцессоров?
9. Какова разрядность современных микропроцессоров в составе персональных ЭВМ?
10. Назовите самую широко распространенную структуру современных ЭВМ, в которой главную роль играет микропроцессор.

10. ОРГАНИЗАЦИЯ ВВОДА-ВЫВОДА В МИКРОПРОЦЕССОРНОЙ СИСТЕМЕ

10.1. Общие сведения

Вводом-выводом информации является процесс передачи данных между микропроцессором и основной памятью (ОЗУ, ПЗУ), между микропроцессором и внешними периферийными устройствами, а также между основной памятью и внешними устройствами. Различают три режима ввода-вывода информации: программный, по прерываниям и прямой доступ к памяти.

Программный ввод-вывод инициируется программой работы микропроцессора, т. е. в определенных местах программы записываются команды опроса портов или обращения к памяти, под управлением которых осуществляется процесс ввода-вывода. Поэтому программный ввод-вывод называют еще *синхронным режимом* ввода-вывода, т. е. управляемым командами микропроцессора. Эти команды выполняются немедленно, если внешние устройства имеют свойство постоянной готовности к приему или передаче данных. Такими устройствами могут быть, например, светодиодные или жидкокристаллические индикаторы на выходе порта вывода либо набор кнопок на входе порта ввода.

Если внешнее устройство таким свойством не обладает, при выполнении команды необходимо осуществлять проверку управляющего сигнала готовности внешнего устройства к передаче или приему данных. Если готовность устройства не подтверждается, то микропроцессор переходит в режим ожидания активного уровня сигнала готовности. Недостатком такой организации ввода-вывода являются большие временные затраты на ожидание готовности медленного внешнего устройства. Однако этот способ ввода-вывода имеет преимущество, заключающееся в простой

аппаратной реализации. Поэтому такой режим ввода-вывода используется преимущественно для осуществления обмена данными с устройствами, постоянно готовыми к этому обмену.

Ввод-вывод по прерываниям производится по инициативе внешних устройств и вводится, если кроме выполнения основной программы необходимо организовать обработку информации от этих устройства по мере ее поступления. По прерыванию может подключаться клавиатура в микропроцессорной системе. Поскольку режим ввода-вывода по прерыванию активизируется внешним устройством, его еще называют **асинхронным** вводом-выводом. При этом микропроцессор должен прервать выполнение основной программы и перейти к подпрограмме обработки устройства, запросившего прерывания.

Переход к обработке запроса прерывания осуществляется по сигналу запроса прерывания *INT* (выдается перепрограммируемым контроллером прерываний от микропроцессора) от внешнего устройства по мере готовности на его выходе данных к передаче или по мере потребности получения им данных от микропроцессора.

В микропроцессорной системе может быть подключено несколько внешних устройств по прерываниям. Поэтому внешнее устройство должно сообщить не только о факте прерывания, но и свой номер, чтобы микропроцессор знал, какую именно обрабатывающую подпрограмму запускать. Для этих целей в микропроцессорных комплектах микросхем обычно имеются специальные микросхемы — **контроллеры прерываний**. Контроллер прерывания воспринимает хотя бы одно прерывание с линий и формирует сигнал запроса прерывания *INT* на шину управления микропроцессора. Далее он должен в виде двоичного кода сформировать на шину данных микропроцессора команду запроса прерывания, где определенные биты этого кода будут указывать на номер устройства, запросившего прерывание. Поскольку сигнал запроса прерывания может поступить в любое время, даже когда микропроцессор выполняет очередную команду программы и его шина данных занята, анализ этого сигнала микропроцессором должен выполняться только после завершения текущей выполняемой команды. Поэтому контроллер прерывания должен дожидаться от микропроцессора подтверждения прерывания в виде сигнала *INTA* (сигнал — подтверждение прерывания) и только после этого выставить на шину данных код команды прерывания.

Однако не всегда можно допустить, чтобы работа микропроцессора прерывалась. Прерывания делят на маскируемые и немаскируемые. *Немаскируемые прерывания* вызываются внешними устройствами и выполняются всегда. *Маскируемые прерывания* отличаются от немаскируемых тем, что могут быть разрешены или запрещены программой путем включения в нее специальных команд разрешения или запрещения прерываний. Обычно по умолчанию маскируемые прерывания при включении питания микропроцессора запрещены. Поэтому, если в программе предполагается реализация ввода-вывода по прерываниям, их нужно в определенном месте программы соответствующей командой разрешить. И хотя бы одно обслуженное маскируемое прерывание запрещает остальные. Если и в дальнейшем необходимо обслуживать эти прерывания, их нужно разрешить.

Различают также прерывания одноуровневые и многоуровневые. Уровень прерывания задается его *приоритетом*, т. е. преимущественным обслуживанием того или иного внешнего устройства при поступлении сразу нескольких прерываний. В многоуровневых системах прерываний подпрограмма обслуживания прерывания более низкого уровня может быть прервана подпрограммой обслуживания прерывания более высокого уровня. Иными словами, реализуется режим прерывания во время прерывания.

Ввод-вывод с прямым доступом к памяти (ПДП) используется в случаях, когда необходимо реализовывать обмен между основной памятью и внешними устройствами большими массивами данных. Как правило, ввод-вывод данных в микропроцессоре реализуется через один из его регистров — аккумулятор. Таким образом, передача данных из внешнего устройства в основную память осуществляется с помощью последовательности команд для каждого элемента массива данных: вначале данные заносятся из внешнего устройства в аккумулятор, затем из аккумулятора в основную память. То есть данные пересылаются через микропроцессор и под его управлением. Это значительно снижает производительность системы при пересылке пакетов данных определенного объема. Причем такие пакеты данных обычно записываются (считываются) с внешнего устройства в соседние ячейки памяти. Это легко сделать в режиме ПДП. Суть его заключается в следующем: буферные регистры данных и адреса микропроцессора переводятся в высокоимпедансное состояние, благодаря

чему микропроцессор оказывается отключенным от шины данных и шины адреса.

Отключение микропроцессора позволяет внешнему устройству самостоятельно обмениваться данными с памятью. Для этого устройство должно само устанавливать на шине адрес требуемой ячейки памяти, а также формировать управляющие сигналы чтения и записи. Для этих целей в микропроцессорных комплектах микросхем имеются специальные микросхемы, называемые контроллерами прямого доступа к памяти. Микросхемы берут на себя задачу управления памятью, а именно формируют адреса соседних ячеек памяти в режиме автоинкремента (это таблица, в которой присутствуют уникальные значения для столбца) или автодекремента (использование автоматического уменьшения содержимого регистра) и управляющие сигналы чтения и записи памяти и портов.

Ввод-вывод в режиме ПДП обычно характерен для микропроцессорных систем универсального назначения, где в разные моменты времени в соседние ячейки оперативной памяти загружаются с внешних накопителей разные программы в виде файлов, а также сохраняются из памяти в виде файлов большие количества промежуточных и конечных результатов. Для специализированных автономных микропроцессорных систем такой режим организуется редко, поскольку эти системы работают в основном с одной программой, уже хранимой в ПЗУ, и быстродействие обращения к внешним устройствам в них не столь значимо.

В качестве примера использования можно рассмотреть **встроенные порты RS-485**.

Все модели центральных процессоров семейства программируемых логических контроллеров фирмы Siemens AG – Simatic S7-200 оснащены одним или двумя встроенными портами RS-485. Каждый встроенный порт имеет универсальное назначение и может использоваться в следующих режимах.

С поддержкой на уровне операционной системы контроллера:

- порт PPI (Point to Point Interface);
- порт MPI (Multi Point Interface);
- свободно программируемый порт.

С поддержкой на уровне программы пользователя:

- свободно программируемый порт;
- порт USS;
- порт ведущего или ведомого устройства MODBUS RTU.

10.2. Интерфейсы микропроцессорных систем

Одним из ключевых моментов в микропроцессорной системе является выбор интерфейсов. **Интерфейсы** представляют собой совокупность унифицированных аппаратных, программных и конструктивных средств, необходимых для реализации алгоритмов взаимодействия различных функциональных устройств. 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. Скорость передачи данных по интерфейсу измеряется в Бодах. **Бод** определяет, сколько бит передается в единицу времени, т. е. в секунду.

К основным характеристикам интерфейса относят функциональное назначение, принцип обмена информацией, способ обмена, режим обмена, количество линий, число линий для передачи данных (разрядность), количество адресов, количество команд, быстродействие, длина линий связи, число подключаемых устройств (нагрузочная способность), тип линии связи.

По *функциональному назначению* интерфейсы разделяют на **внутренние** (межплатные, межэлементные и системные) и **внешние** (для внешних периферийных устройств, локальных сетей, распределенных систем). Внутренние интерфейсы определяют комплекс принятых стандартов по организации связей между микросхемами, модулями и блоками, составляющими микропроцессорную систему. Внешние интерфейсы организуют связь между несколькими микропроцессорными системами.

По *принципу обмена информацией* выделяют интерфейсы с **параллельной, последовательной и параллельно-последовательной передачей** информации. В параллельных интерфейсах в единицу времени данные пересылаются по нескольким сигнальным линиям одновременно, в последовательных — по одной линии бит за битом. Аналогом здесь может служить принцип организации параллельных и последовательных регистров. Методически интерфейсы с параллельной передачей должны обладать более высокой пропускной способностью, однако это не всегда так. Дело в том, что в параллельных интерфейсах из-за большого количества сигнальных линий, объединенных в жгуты, существенное влия-

ние оказывают паразитные емкости этих жгутов, увеличивающиеся с увеличением длины линии. Кроме того, на такие жгуты оказывает влияние поперечная помеха, также усиливающая свое влияние с увеличением длины линии. Это накладывает ограничение на длину соединительных жгутов. В последовательных интерфейсах линии обычно представляют витую пару и их длину можно реализовывать значительно большую, чем в параллельных интерфейсах.

При проектировании специализированных микропроцессорных систем на базе микроконтроллеров разработчики часто сталкиваются с проблемой нехватки портов для подключения большого количества периферий. Поэтому выводы микросхем микроконтроллеров часто приходится экономить. По этой причине в микроконтроллерных и микропроцессорных системах в настоящее время часто используются последовательные интерфейсы, особенно внутренние.

По *режиму обмена информацией* различают интерфейсы с симплексным, полудуплексным, дуплексным и мультиплексным режимами обмена.

Симплексный режим обмена характеризуется тем, что только один из абонентов в любой момент времени может инициировать передачу информации.

Полудуплексный режим обмена обеспечивает инициализацию связи любым абонентом, если линия связи интерфейса свободна.

Дуплексный режим обеспечивает связь любому абоненту в любой момент времени.

Мультиплексный режим позволяет связываться только между парами абонентов и в единственном направлении от одного к другому. Характерен для магистральной организации, когда периодически разные устройства включаются в общую шину.

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). В первом случае передача синхронизируется специальными синхроимпульсами *CLK* (сигнал тактовой синхронизации) в виде последовательности прямоугольных импульсов, во втором – управляющими сигналами «Готовность» к обмену, «Начало», «Конец» и «Контроль» обмена.

В качестве примера магистральных интерфейсов можно привести асинхронные мультиплексные интерфейсы с параллельным способом передачи: 8-разрядные интерфейсы Microbus, Z-bus, шины iSBX, ISA; 16-разрядные интерфейсы Unibus (отечественный аналог ОШ), Multibus I, Q-bus, EISA; 32-разрядные интерфейсы Vercabus, PCI. В качестве внешних интерфейсов периферийных устройств наиболее широкое распространение получили последовательные интерфейсы RS-232, USB и параллельный интерфейс Centronix.

В микропроцессорной технике для согласования микросхем аналого-цифровых и цифроаналоговых преобразователей, памяти и прочих схем с микропроцессорами и микроконтроллерами широко используются внутренние последовательные интерфейсы I²C, SPI и MicroLAN. Для программирования внутренней памяти программ и отладки программного обеспечения микроконтроллеров используется интерфейс JTEG.

Следует отметить, что в составе микропроцессорных систем имеются специальные интерфейсные большие интегральные схемы (БИС), обеспечивающие совместимость по электрическим, конструктивно-технологическим и эксплуатационным параметрам процессорного модуля с внешними устройствами. Каждый микропроцессорный комплект, как правило, содержит параллельные и последовательные интерфейсные устройства, выполненные в виде отдельных микросхем, представляющих собой регистры, буферные схемы и специализированные контроллеры. Такие микросхемы используют как шинные формирователи для организации внутримашинных магистралей и как внешние интерфейсы периферийных устройств.

10.3. Интерфейс RS-232

В качестве примера приведем принцип действия наиболее часто используемых в микроконтроллерных системах интерфейсов. Начнем с интерфейса RS-232 (также принято обозначение EIA-232). Этот интерфейс обеспечивает последовательный мультиплексный режим связи. Для него сигнал логической единицы имеет уровень напряжения менее -3 В (обычно -10 В) и сигнал логического нуля – уровень напряжения более $+3$ В (обычно $+10$ В). Гарантированная длина кабеля для надежной работы интерфейса может быть до 30 м в зависимости от скорости

передачи. Есть микросхемы, которые выполняют преобразование стандартных ТТЛ-уровней в уровни стандарта RS-232. Примером может служить микросхема SP232A производства фирмы Siprex, условное графическое обозначение и типовая схема включения которой показаны на рисунке 10.1.

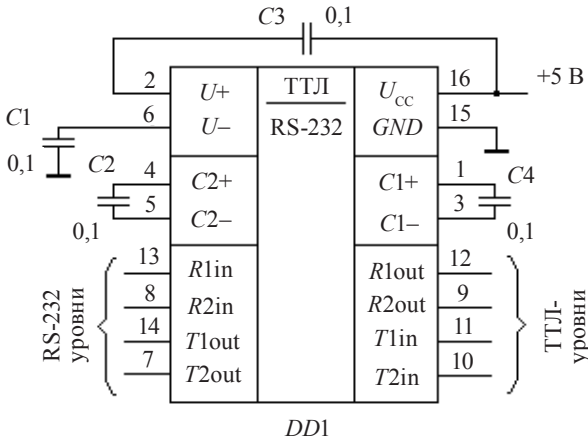


Рис. 10.1. Условное графическое обозначение и схема включения микросхемы SP232A

Микросхема SP232A преобразует уровни напряжений стандарта ТТЛ (уровень нуля $\leq 0,2$ В и уровень единицы $\geq 3,5$ В, реально 0 В и 5 В) в уровни напряжения стандарта RS-232 (уровень единицы -10 В и уровень нуля $+10$ В) и, наоборот, для двух равноценных каналов 1 и 2. Вывод $T1in$ микросхемы является входом усилителя первого канала с входными напряжениями ТТЛ, вывод $T1out$ – выходом усилителя первого канала с уровнями напряжений RS-232. Вывод $R1in$ является входом усилителя первого канала с входными напряжениями RS-232, вывод $R1out$ – выходом усилителя первого канала с уровнями напряжений ТТЛ. Аналогично назначение выводов второго канала. Аналогичная микросхема SP240 отличается от рассмотренной тем, что у нее ТТЛ-выходы усилителей обоих каналов выполнены по трехстабильной схеме. Для управления состояниями этих выходов микросхема SP240 имеет дополнительный управляющий вход.

Передача данных по протоколу стандарта RS-232 осуществляется обычно побайтно бит за битом (в общем случае от 5 до 8 бит)

по независимым двум каналам для каждого направления. Канал передачи данных от передатчика называется каналом TxD (передача данных), а канал приема данных в приемник — каналом RxD (чтение данных). Пассивному состоянию линии передачи соответствует уровень напряжения логической единицы (-10 В).

Наибольшее распространение получил асинхронный режим передачи интерфейса RS-232, поскольку именно он реализован для COM-портов персонального компьютера. В этом режиме передаче основного байта предшествует выдача на линию передатчиком логического нуля (стартовый бит), а завершение передачи заканчивается выдачей логической единицы (стоповый бит) (рис. 10.2).

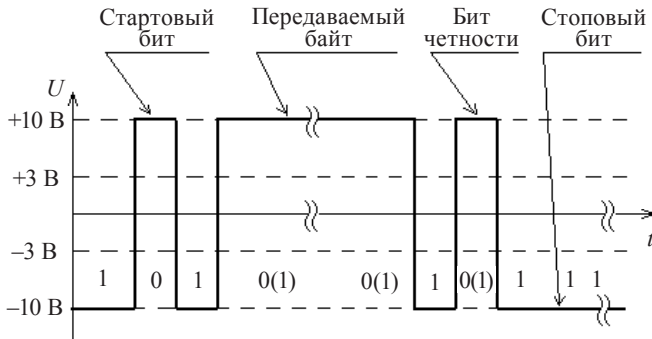


Рис. 10.2. Временная диаграмма протокола интерфейса RS-232

Кроме того, можно использовать контрольный бит четности, предшествующий передаче стопового бита. Бит четности дополняет общую сумму единиц основного байта до четного (нечетного) количества единиц. Такой способ контроля может выявить на принимающей стороне ошибку, если общее количество единиц приходит нечетным (четным). При пересылке данных между устройствами необходимо, чтобы оба устройства были настроены в один режим и на одну скорость. Типовые значения скорости передачи данных для интерфейса RS-232 составляют 2400, 4800, 9600, 19 200 Бод.

Как правило, функции интерфейса RS-232 реализуются аппаратно. Для этого существуют специализированные микросхемы последовательных приемопередатчиков. Если такие микросхемы реализуют асинхронный режим последовательной связи,

они называются *UART-преобразователями* (Universal Asynchronous Receiver/Transmitter). Примером подобной микросхемы является i8251A (отечественный аналог КР580ВВ51А). Эта микросхема не является в чистом виде асинхронным приемопередатчиком. Она поддерживает как асинхронный режим, так и синхронный, т. е. ее можно охарактеризовать как USART-преобразователь. Обмен может быть как полудуплексным, так и дуплексным. Микросхема i8251A (рис. 10.3) выполняет следующие функции:

- прием данных из устройства, осуществляющего передачу, преобразование их из параллельного формата в последовательный и передачу через последовательный канал периферийному устройству;
- прием данных по последовательному каналу из периферийного устройства, преобразование их к параллельному формату и передачу в устройство назначения.

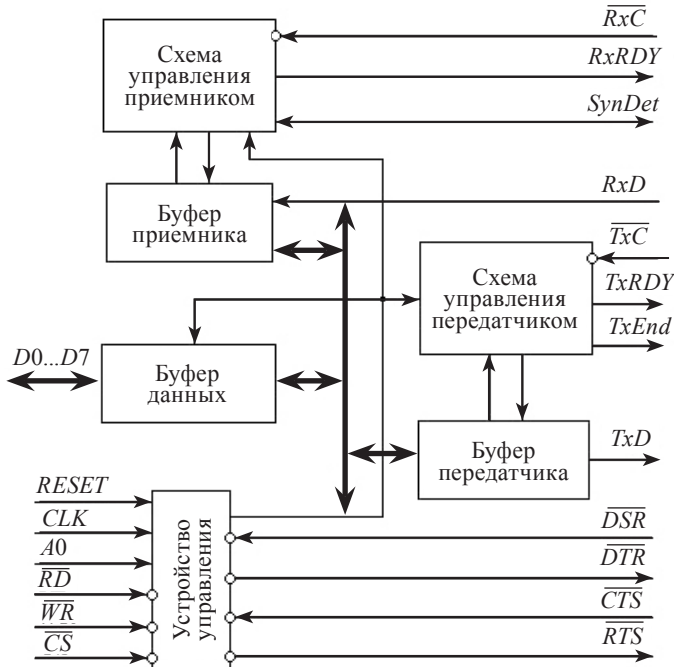


Рис. 10.3. Структурная схема микросхемы i8251A

Назначение выводов микросхемы следующее (стрелка внутрь соответствует входу, стрелка наружу – выходу):

- $D0...D7$ – двунаправленные трехстабильные выходы параллельных данных;
- $RESET$ – сброс;
- CLK – синхронизация (вход подачи последовательности прямоугольных импульсов с типовым значением частоты 2 МГц);
- $A0$ – адрес. Высокий уровень сигнала на этом входе определяет возможность записи или чтения данных по параллельному каналу, а низкий – возможность записи управляющих кодов для задания режима работы микросхемы или чтение слова состояния;
- RD – разрешение вывода данных или слова состояния на выходы параллельного канала;
- WR – разрешение ввода данных с параллельного канала;
- CS – выбор кристалла;
- RTS – готовность микросхемы передавать данные по последовательному каналу;
- CTS – готовность приемника терминала;
- DTR – готовность микросхемы принять данные по последовательному каналу;
- DSR – готовность передатчика терминала;
- TxD – последовательный канал передачи данных;
- $TxEnd$ – буфер передатчика пустой;
- $TxRDY$ – готовность передатчика к приему данных с выводов параллельного канала;
- TxC – синхронизация передачи данных;
- RxD – последовательный канал приема данных;
- $SynDet$ – определяет вид синхронизации (внутренней или внешней). Состояние вывода определяется программно;
- $RxRDY$ – данные на выводах параллельного канала готовы для считывания их микропроцессором;
- RxC – синхронизация приема данных.

Несмотря на то что микросхема имеет большое количество управляющих выводов, для связи терминала и периферийного устройства в асинхронном режиме обычно используют всего два вывода: RxD и TxD . Причем вход RxD терминала соединяется кабелем с выходом TxD периферийного устройства и выход TxD терминала соединяется с входом RxD периферийного устройства.

Рассмотрим работу микросхемы в наиболее часто используемом режиме – асинхронном. В этом режиме скорость обмена кратна частоте сигналов синхронизации TxC и RxC в соответствующих направлениях обмена. На эти входы подаются синхроимпульсы в виде последовательности прямоугольных импульсов. Коэффициент кратности задается программно и может быть выбран из ряда 1:1, 1:16 и 1:64. Для задания режима работы в микросхему по параллельному каналу необходимо записать слово инициализации. Это слово определяет асинхронный или синхронный режим работы микросхемы, кратность коэффициента деления, длину передаваемых/принимаемых данных по последовательному каналу, наличие или отсутствие бита контроля на четность или нечетность, длительность стоп-бита.

В асинхронном режиме биты параллельного кода слова инициализации определяются следующими значениями:

- $D0 = 1, D1 = 0$ – коэффициент кратности равен 1;
- $D0 = 0, D1 = 1$ – коэффициент кратности равен 16;
- $D0 = 1, D1 = 1$ – коэффициент кратности равен 64;
- $D2 = 0, D3 = 0$ – формат передаваемых данных – 5 бит;
- $D2 = 1, D3 = 0$ – формат передаваемых данных – 6 бит;
- $D2 = 0, D3 = 1$ – формат передаваемых данных – 7 бит;
- $D2 = 1, D3 = 1$ – формат передаваемых данных – 8 бит;
- $D4 = 1, D5 = 0$ – контроль по четности;
- $D4 = 1, D5 = 1$ – контроль по нечетности;
- $D4 = 0, D5 = 1$ – без контроля;
- $D6 = 1, D7 = 0$ – длина стоп-бита 1;
- $D6 = 0, D7 = 1$ – длина стоп-бита 1,5;
- $D6 = 1, D7 = 1$ – длина стоп-бита 2.

Слово инициализации устанавливается один раз. Смена слова инициализации необходима тогда, когда необходимо сменить режим работы микросхемы. В процессе передачи может возникнуть потребность задавать слова команды, определяющие функции и этапы передачи.

Разряды слова команды задают следующие функции:

- $D0$ – разрешение на передачу (при нулевом значении разряда передача невозможна, при единичном – возможна);
- $D1$ – запрос готовности передатчика (при единичном значении разряда на выводе DTR микросхемы устанавливается ноль);

- *D2* – разрешение приема (при нулевом значении разряда прием невозможен, при единичном – возможен);
- *D3* – конец передачи (при нулевом значении разряда – нормальная работа канала передачи, при единичном – установка высокого уровня на выходе *TxD* микросхемы);
- *D4* – сброс меток ошибок (при единичном значении разряда сбрасываются все флажки-признаки ошибок);
- *D5* – запрос готовности приемника терминала (при единичном значении разряда на выводе *RTS* микросхемы устанавливается нуль);
- *D6* – программный сброс (единичное значение разряда равноценно подаче сигнала *RESET*);
- *D7* – поиск синхроимпульсов (при единичном значении разряда выполняется поиск синхроимпульсов в синхронном режиме).

10.4. Последовательный интерфейс USB

В настоящее время стандарт последовательной передачи RS-232 считается морально устаревшим. На многих современных персональных компьютерах он даже не предусмотрен. На смену ему пришел интерфейс USB (Universal Serial Bus или универсальная последовательная шина). Этот интерфейс широко применяется для связи внешних периферийных устройств с персональным компьютером. Он может быть полезен, когда необходимо подключить прибор, построенный на базе микропроцессора или микроконтроллера как внешнее устройство к персональному компьютеру.

Интерфейс USB представляет собой шину, т. е. к одному физическому каналу можно подключать несколько устройств. Для подключения периферийных устройств к шине USB используется четырехжильный кабель. При этом два провода «D–» и «D+» (витая пара) в дифференциальном включении используются для приема и передачи данных, а два провода «+5 В» и «Общий» – для питания периферийного устройства. Благодаря встроенным линиям питания интерфейс USB позволяет подключать периферийные устройства без собственного источника питания. Максимальная сила тока, потребляемого устройством по линиям питания шины USB, не должна превышать 500 мА. Существует несколько версий стандарта шины USB, отличающихся режи-

мами работы и пропускной способностью, но схемотехническое подключение устройств к этой шине одинаковое для всех версий (кроме последней версии 3.0 и выше).

Шина USB строго ориентирована, имеет понятие «главное устройство» (хост, он же USB контроллер) и «периферийные устройства». Шина представляет древовидную топологию, т. е. периферийным устройством может быть *разветвитель* (хаб), к которому в свою очередь могут подключаться другие периферийные устройства. Разветвитель всегда активный и представляет собой сложное электронное устройство. К одному контроллеру шины USB можно подсоединить до 127 устройств по топологии «звезда», в том числе и разветвители. На одной шине USB может быть до пяти уровней каскадно включенных разветвителей.

Шина USB поддерживает «горячее» подключение периферийных устройств в режиме *Plug&Play*. Это означает, что подключаемое к контроллеру шины USB периферийное устройство должно быть опознано персональным компьютером и должна автоматически запуститься программа обработки этого подключения. Попутно решается историческая проблема нехватки ресурсов на внутренних системных шинах персонального компьютера: контроллер USB занимает только одно прерывание независимо от количества подключенных к шине USB устройств. Для этого каждое периферийное устройство должно иметь уникальный код, который считывается контроллером USB (хостом) при подключении этого устройства.

Несмотря на то что стандарт RS-232 на сегодняшний момент считается устаревшим и во многих новых моделях персональных компьютеров не используется, существует огромный парк периферийных устройств, имеющих интерфейс RS-232. Поэтому возникает задача стыковки периферии, имеющей интерфейс RS-232, с новыми моделями персональных компьютеров, особенно с переносными мобильными компьютерами. Для этих целей разработаны микросхемы преобразователей RS-232 (с асинхронным режимом работы) в стандарт шины USB. Примером такой микросхемы является микросхема FT232R фирмы Future Technology Devices International Ltd., условное графическое обозначение и типовая схема включения которой приведены на рисунке 10.4.

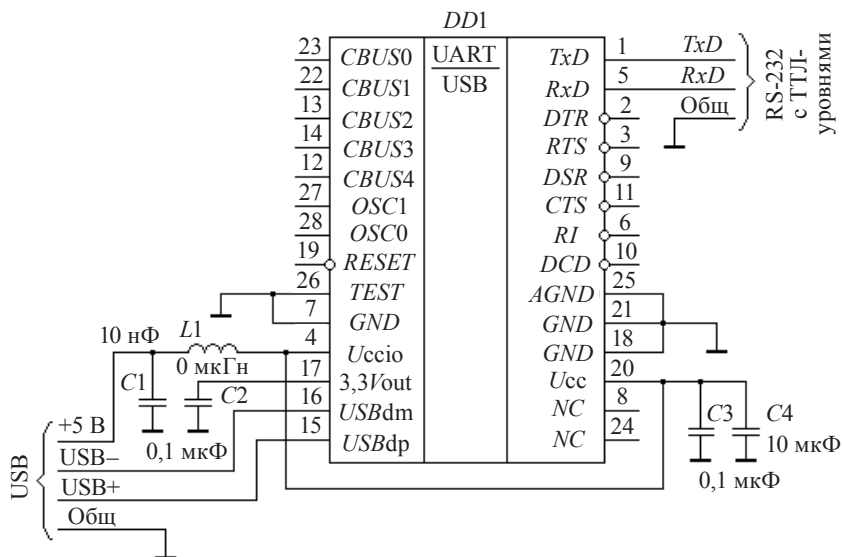


Рис. 10.4. Условное графическое обозначение и схема включения микросхемы FT232R

В микросхеме имеются все сигналы и линии стандарта RS-232 со стороны периферийного устройства, рассмотренные ранее, и дополнительные линии *RI* (обнаружен вызов) и *DCD* (обнаружена несущая), задействованные обычно в работе модема. Для связи периферийного микропроцессорного устройства с персональным компьютером достаточно использовать только линии *TxD* и *RxD*. В микросхему FT232R запрограммирован индивидуальный идентификационный код, что позволяет использовать ее без каких-либо специальных процедур программирования. Если предусматривается возможность подключения нескольких устройств через несколько микросхем FT232R к одному хосту, то в каждую из них нужно запрограммировать индивидуальный номер, чтобы хост мог корректно идентифицировать устройство и запустить необходимую процедуру обработки. Для этих целей микросхема имеет встроенное перепрограммируемое ПЗУ, программирование которого осуществляется через выходы *CBUS0...CBUS4*. Выводы *OSC1* и *OSC0* используются в опциональных применениях и в обычном режиме не задействованы. Вывод *TEST* также используется в специальных режимах, в обыч-

ном режиме на него подается логический нуль. Вывод 3,3V_{out} может быть использован как стабилизированный источник напряжения для подключения дополнительных потребителей с потребляемым током не более 50 мА, если такие есть в схеме.

10.5. Интерфейс I²C

В цифровой микропроцессорной технике широко применяют интерфейс I²C (часто встречается обозначение I2C). Этот интерфейс представляет собой шину, поддерживающую мультиплексный синхронный последовательный обмен данными. Название интерфейса расшифровывают как «Inter IC», т. е. «межмикросхемная шина».

Интерфейс I²C был предложен фирмой Philips в начале 1990-х годов специально для связи различных цифровых микросхем между собой. Сегодня этот интерфейс используют для простой, не требующей большого количества выводов и сигнальных линий, стыковки микроконтроллеров с микросхемами памяти, АЦП, ЦАП, часами реального времени, таймерами, различными преобразователями, индикаторами и прочей периферией. В периферийных устройствах протокол шины I²C обычно реализуется аппаратно, в микроконтроллерах и микропроцессорах он может реализовываться как аппаратно, так и программно.

Шина I²C использует две двунаправленные линии, подтянутые к напряжению питания резисторами R_{Π} (рис. 10.5). На шину параллельно подключаются все микросхемы и функциональные блоки микропроцессорной системы. Выходы всех устройств, подключаемых к шине, должны быть организованы по схеме с открытым коллектором или открытым стоком. Одна линия имеет обозначение *SDA* (Serial Data) и предназначена для последовательной передачи/приема данных, а вторая имеет обозначение *SCL* (Serial Clock) и предназначена для последовательного тактирования. Поскольку в шине прием и передача данных организованы по одной линии *SDA* в обоих направлениях, то выводы *SDA* и *SCL* микросхем должны быть как входами, так и выходами.

Стандартные напряжения логических уровней соответствуют ТТЛ-уровням, однако допускаются и другие значения напряжений. Основной режим работы реализуется на скорости 100 кбит/с, режим с пониженной скоростью 10 кбит/с. В более поздних спецификациях появились ускоренный режим со ско-

ростью 400 кбит/с и режим повышенной скорости 3,4 Мбит/с. Стандарт шины допускает также приостановку тактирования для работы с медленными устройствами.

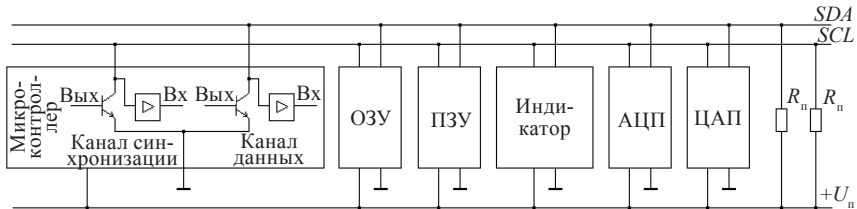


Рис. 10.5. Организация шины I²C

Устройства, подключаемые к шине, по своему назначению делятся на ведущие (Master) и ведомые (Slave). В шине должно быть как минимум одно ведущее устройство. Как правило, таким устройством является микроконтроллер. Он берет на себя функции управления шиной, вырабатывает тактовые сигналы *SCL*, инициирует сеансы приема/передачи. Информация может передаваться как от ведущего устройства к ведомому, так и в обратном направлении, но обязательно под управлением ведущего устройства. На рисунке 10.5 показана внутренняя организация выводов *SDA* и *SCL* только для одного устройства. Такая же организация характерна для всех микросхем, подключаемых к шине.

В исходном состоянии на выходах *SDA* и *SCL* всех микросхем, подключенных к шине I²C, должны быть установлены логические единицы. Благодаря схеме с открытыми коллекторами, в которой транзисторные ключи выходов *SDA* и *SCL* оказываются закрытыми, через резисторы R_n потенциалы указанных выходов подтягиваются к напряжению питания U_n . Внутренние сопротивления закрытых транзисторов выходов микросхем оказываются большими и все микросхемы находятся в режиме чтения. Если какое-либо устройство начнет процесс передачи, остальные устройства окажутся приемниками. Открывающиеся под управлением синхроимпульсов и импульсов передаваемых данных выходные транзисторы передающего устройства будут подсаживать потенциалы линий *SDA* и *SCL* шины на землю. Переданные сигналы данных и синхронизации поступят на входы всех остальных микросхем. Каждая микросхема, работающая в режиме приемника, считает переданный сигнал.

Кроме самих передаваемых данных, в каждом пакете импульсов закладывается адрес устройства, для которого эти данные предназначены. Каждое подключаемое к шине I²C периферийное устройство имеет свой уникальный адрес. В стандартном режиме для адресации периферийного устройства используется 7 бит информации с 16 зарезервированными адресами. Это означает, что к шине может быть подключено до 112 ведомых устройств. В более поздних спецификациях шины для адресации стали использовать 10 бит, что увеличило количество подключаемых устройств до 1008. Если передаваемый адрес совпадет с индивидуальным адресом микросхемы, эта микросхема примет предназначенные для нее данные и обработает их. Остальные микросхемы текущую посылку проигнорируют. Адрес в ведомых микросхемах в зависимости от их типа может быть задан жестко производителем или может программироваться разработчиком устройства. Иногда микросхема содержит неизменяемую часть адреса в старших разрядах и часть адреса в младших разрядах, доступную для программирования разработчиком. В таком случае упрощается адресация при подключении однотипных микросхем.

Информация по шине I²C передается побайтно, бит за битом. Передача каждого бита по линии *SDA* сопровождается синхроимпульсом на линии *SCL*. Процедура обмена начинается с того, что ведущий передатчик формирует состояние «Старт», обозначаемое в спецификации шины символом «S». Для этого он генерирует переход сигнала линии *SDA* из высокого «отпущенного» состояния в низкое «подсаженное» при высоком уровне *SCL* (рис. 10.6). Этот переход воспринимается всеми устройствами, подключенными к шине, как признак начала процедуры обмена.

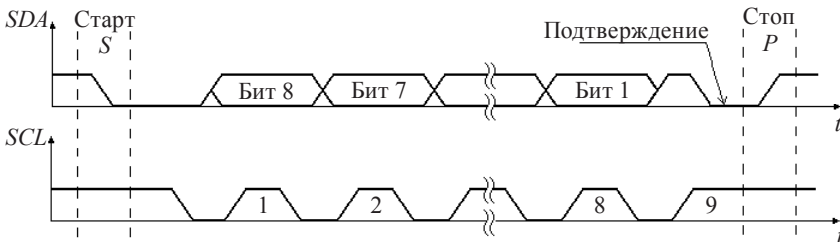


Рис. 10.6. Временная диаграмма протокола интерфейса I²C

Генерация синхросигнала – всегда обязанность ведущего устройства. Каждый ведущий генерирует свой собственный сигнал синхронизации при пересылке данных по шине. Процедура обмена завершается тем, что ведущий формирует состояние «Стоп» (обозначаемое символом «P»), в котором осуществляет переход линии *SDA* из низкого уровня в высокое «отпущенное» при высоком уровне линии *SCL*. Состояния «Старт» и «Стоп» всегда вырабатываются ведущим. Считается, что шина занята после фиксации состояния «Старт» и свободна через некоторое время после фиксации состояния «Стоп». После формирования состояния «Старт» ведущий опускает уровень линии *SCL* в низкое состояние и выставляет на линию *SDA* старший бит первого байта сообщения.

Передача байта по спецификации шины I²C всегда происходит начиная со старшего бита. Количество байтов в сообщении не ограничено. Спецификация шины I²C при передаче данных разрешает изменения на линии *SDA* только при низком уровне сигнала на линии *SCL*. Данные действительны и должны оставаться стабильными только во время высокого состояния синхроимпульса. Исключение составляют состояния «Старт» и «Стоп», когда уровень линии *SDA* меняется при высоком уровне линии *SCL*.

Для подтверждения приема байта от ведущего передатчика ведомым приемником в протоколе обмена по шине I²C вводится специальный бит подтверждения, выставляемый на шину *SDA* после приема последнего младшего бита данных. Для этого передатчик устанавливает логическую единицу на выходе *SDA*, тем самым «отпускает» эту линию, и передает девятый синхроимпульс по линии *SCL*. В ответ на этот синхроимпульс приемник должен сам «подсадить» линию *SDA*. Если это произойдет, передатчик считает низкий уровень с линии *SDA* и продолжит или окончит передачу в зависимости от того, все ли байты переданы. Если от приемника подтверждения не последует, передача приостановится (см. рис. 10.6).

Механизм передачи, когда ведущее устройство является приемником, а ведомое передатчиком, аналогичен описанному выше. В этом режиме формированием синхроимпульсов и состояний «Старт» и «Стоп» также будет заниматься ведущее устройство. Однако только в данном случае оно будет приемником. В от-

вет на состояние «Старт» ведомый передатчик под управлением синхроимпульсов *SCL* с ведущего приемника будет выставлять побитно данные на линию *SDA*, которые будут считаны ведущим приемником. После передачи 8 бит данных ведомый передатчик приступит к ожиданию от ведущего приемника бита подтверждения. Если прием прошел без ошибок, ведущий приемник передаст этот бит путем подсаживания линии *SDA* и формирования синхроимпульса на линии *SCL*. В противном случае на линии *SDA* останется присутствовать высокий потенциал.

Процесс обмена информацией начинается с того, что ведущее устройство выдает на шину I²C первый байт, в семи старших разрядах которого присутствует адрес активизируемого устройства, а в младшем бите отражается направление пересылки данных. При нуле в этом бите передача осуществляется от ведущего устройства к ведомому, а при единице — от ведомого к ведущему. Все ведомые устройства получают условие «Старт» и примут первый байт. Только в одном устройстве адрес, закодированный первым байтом, совпадет с индивидуальным адресом этого устройства. И именно оно продолжит участвовать в пересылке, остальные ведомые устройства останутся в режиме ожидания. Все они будут ждать новое условие «Старт», после которого вновь первым передаваемым байтом будет адрес активизируемого устройства. Ведомое устройство, для которого адрес совпал, завершит процедуру приема первого байта, для чего оно выработает бит подтверждения. Получив подтверждение, ведущее устройство начнет в зависимости от направления пересылки управлять приемом или передачей данных. Между этими пересылками условие «Стоп» формироваться не будет, а будет формироваться только бит подтверждения после пересылки очередного байта. Как только все байты данных будут пересланы, ведущее устройство сформирует условие «Стоп» и пересылка будет считаться завершенной.

10.6. Интерфейс SPI

В микропроцессорной технике широко используют интерфейс SPI (Serial Peripheral Interface, SPI bus — последовательный периферийный интерфейс, шина SPI). Изначально интерфейс SPI был придуман компанией Motorola, а в настоящее время используется в продукции многих производителей. Шина SPI организована по принципу «ведущий—ведомый» и предназначена для

подключения внешних устройств. В качестве ведущего устройства обычно выступает микроконтроллер (или микропроцессор с контроллером указанного интерфейса в системе), а также может быть ПЛИС или специализированная микросхема. Подключенные к ведущей шине внешние устройства являются ведомыми. В их роли выступают различного рода микросхемы, в том числе запоминающие устройства (ОЗУ, ПЗУ, ППЗУ, Flash-ROM), часы реального времени, АЦП или ЦАП и др.

Главный составной блок интерфейса SPI – обычный регистр сдвига, сигналы синхронизации и ввода-вывода которого образуют интерфейсные сигналы. Протокол SPI является протоколом обмена данными между двумя сдвиговыми регистрами, каждый из которых одновременно выполняет и функцию приемника, и функцию передатчика. Непременное условие передачи данных по шине SPI – генерация сигнала синхронизации. Этот сигнал имеет право генерировать только ведущее устройство. От этого сигнала полностью зависит работа ведомого устройства.

Стандартом шины SPI определены четыре сигнала:

- *MOSI* (Master Output Slave Input) – выходной сигнал передачи последовательных данных для ведущего устройства, входной сигнал приема последовательных данных для ведомого устройства;

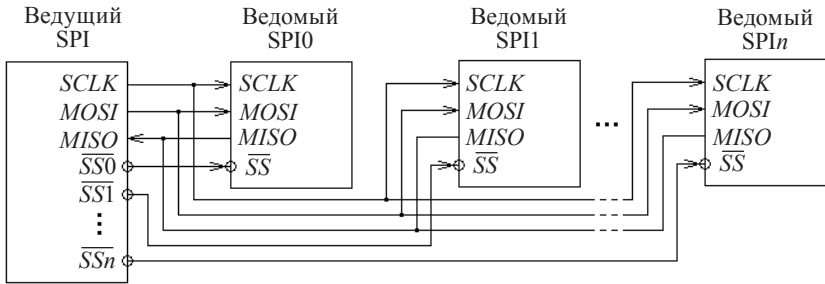
- *MISO* (Master Input Slave Output) – входной сигнал приема последовательных данных для ведущего устройства, выходной сигнал передачи последовательных данных для ведомого устройства;

- *SCLK* (Select Clock) – выходной сигнал синхронизации для ведущего устройства и аналогичный входной сигнал для ведомого устройства;

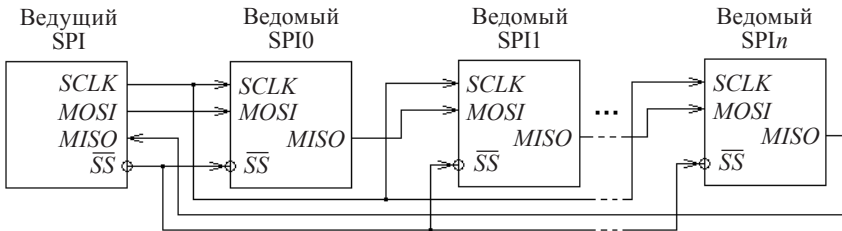
- \overline{SS} (Select Slave) – выходной сигнал выбора одного из ведомых устройств для ведущего устройства, входной сигнал активизации для ведомого устройства.

Существуют два типа подключения нескольких устройств к шине SPI – параллельное и последовательное. Параллельное подключение (рис. 10.7, а) более распространено, поскольку позволяет организовывать подключение любых SPI-совместимых устройств. При таком подключении все линии, кроме линии сигнала выбора \overline{SS} , соединены параллельно. Ведущее устройство переводом того или иного сигнала \overline{SS} в низкое состояние задает, с каким ведомым устройством оно будет обмениваться данными. Главным недостатком такого подключения является необходи-

мость в дополнительных линиях \overline{SS} для адресации нескольких ведомых устройств, что существенно неудобно в случае большого их количества.



а



б

Рис. 10.7. Варианты подключения устройств по стандарту интерфейса SPI:

а – параллельное; б – последовательное

Самое простое подключение последовательное (см. рис. 10.7, б), в котором участвуют только два устройства – ведущее и ведомое, требует всего одной линии сигнала \overline{SS} . Ведущее устройство передает данные по линии $MOSI$ синхронно со сгенерированным им же сигналом $SCLK$, а ведомое захватывает переданные биты данных по определенным фронтам принятого сигнала синхронизации. Одновременно с этим ведущее устройство отправляет свою посылку данных.

Представленную схему можно упростить исключением одной из линий $MISO$ или $MOSI$, если в процессе передачи предполагается только односторонняя связь. Одностороннюю передачу данных можно встретить у таких микросхем, как ЦАП, АЦП,

цифровые потенциометры, программируемые усилители и аналогичные по смыслу. Чтобы ведомое устройство принимало и передавало данные, кроме наличия сигнала синхронизации, необходимо также, чтобы линия \overline{SS} была переведена в низкое состояние. Когда используется только одно ведомое устройство, может возникнуть соблазн исключения и линии \overline{SS} за счет жесткой установки низкого уровня на входе выбора ведомого устройства. Такое решение крайне нежелательно и может привести к сбоям или вообще к невозможности передачи данных, так как вход выбора ведомого устройства служит для перевода его в исходное состояние и иногда инициирует вывод первого бита данных.

Каскадное включение избавлено от большого количества линий \overline{SS} в случае большого количества ведомых устройств, поскольку при данном подключении из нескольких микросхем образуется один большой сдвиговый регистр. Для этого выход передачи данных $MISO$ текущего ведомого устройства соединяется с входом приема данных $MOSI$ последующего. Входы выбора \overline{SS} всех устройств соединяются параллельно и, таким образом, общее число линий связи сохраняется равным четырем. Однако использование последовательного подключения возможно только в том случае, если оно обеспечивается характеристиками всех ведомых устройств. В технической документации на подключаемые в шину устройства по последовательной схеме должна стоять отметка «daisy-chaining» (последовательное подключение).

Протокол передачи по интерфейсу SPI отражает логику работы сдвигового регистра, которая заключается в выполнении операции сдвига и соответственно побитного ввода и вывода данных по определенным фронтам сигнала синхронизации. Установка данных при передаче и выборка при приеме всегда выполняются по противоположным фронтам импульсов синхронизации (рис. 10.8).

Импульсы синхронизации $SCLK$ могут иметь как положительную, так и отрицательную полярность. При этом передача или прием осуществляются как по переднему, так и по заднему его фронту. Таким образом, возможны четыре варианта логики работы интерфейса SPI. Эти варианты получили название режимов SPI и описываются двумя параметрами:

- CPOL – исходный уровень сигнала синхронизации (если CPOL = 0, линия синхронизации до начала цикла передачи и после его окончания имеет низкий уровень, если же CPOL = 1 – высокий);

• \overline{CPHA} – фаза синхронизации (если $\overline{CPHA} = 0$, по переднему фронту в цикле синхронизации будет выполняться прием данных, а затем по заднему фронту передача данных; если же $\overline{CPHA} = 1$, передача данных будет выполняться по переднему фронту в цикле синхронизации, а прием – по заднему).

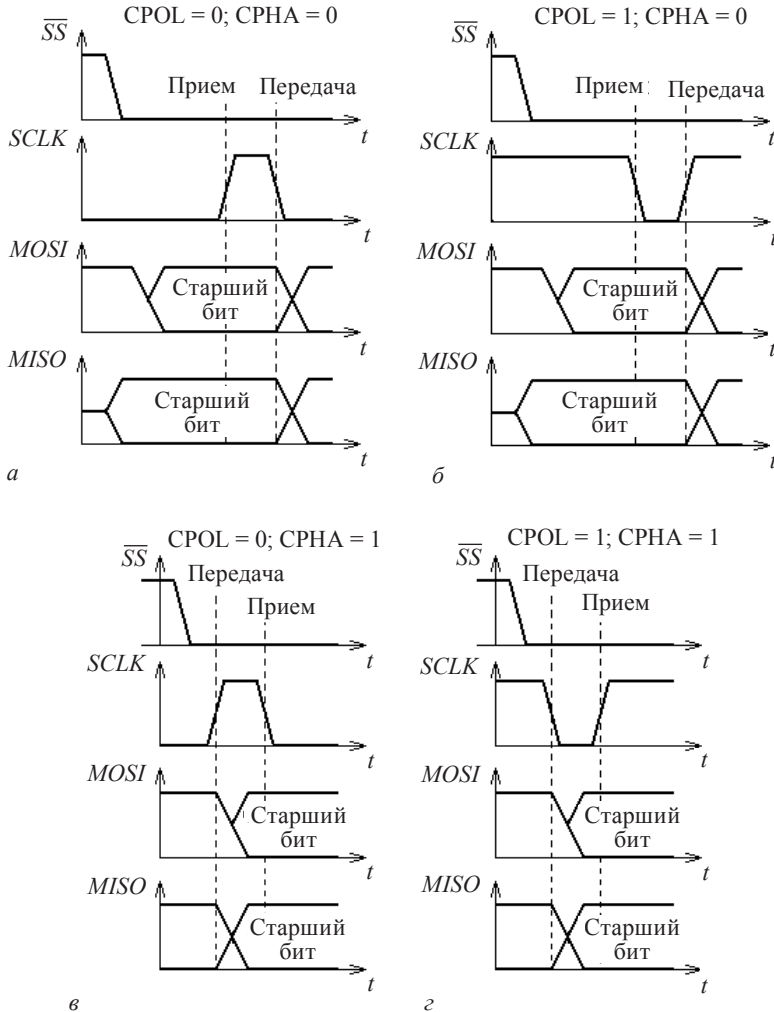


Рис. 10.8. Временная диаграмма протокола интерфейса SPI:
а–г – побитный ввод и вывод данных

Ведущее и ведомое устройства, работающие в различных режимах SPI, несовместимы. Поэтому необходимо обращать внимание на характеристики ведомых устройств при подключении их в шину. Аппаратные модули SPI, интегрированные в микроконтроллеры, в большинстве случаев поддерживают возможность выбора любого режима SPI. Карты флеш-накопителей форматов SD и MMC подключаются аналогично протоколу с модулем SPI.

10.7. Интерфейс PPI

Интерфейс PPI (Point To Point Interface) может быть использован для подключения программатора, устройств человеко-машинного интерфейса, других контроллеров. Каналы связи выполняются витой парой. Максимальная скорость обмена данными может достигать 187,5 кбит/с.

На основе интерфейса PPI могут создаваться простейшие сетевые структуры, объединяющие в своем составе программируемые контроллеры Simatic S7-200 фирмы Simens, программатор, компьютер, а также устройства человеко-машинного интерфейса. Управление обменом данными из программы пользователя выполняется с помощью инструкций NETR/NETW. В такой сети каждый программируемый контроллер S7-200 выступает в роли равноправного партнера по связи, способного генерировать запросы к другим сетевым станциям или отвечать на их запросы.

Общее количество станций в одной PPI-сети может достигать 32.

10.8. Свободно программируемый порт (Freeport)

Этот режим позволяет поддерживать обмен данными с использованием протокола ASCII. Управление обменом данными из программы пользователя осуществляется с помощью инструкций XMT/RCV. Подключение к устройствам с интерфейсом RS-232 допускается выполнять через RS-232/PPI кабель. Максимальная скорость обмена данными может достигать 38,4 кбит/с.

Свободно программируемый режим может быть использован для организации:

- связи с приборами, оснащенными последовательным интерфейсом;
- модемной связи с использованием внешнего модема;
- непосредственной связи между двумя контроллерами S7-200.

10.9. Порт ведущего-ведомого устройства MODBUS RTU

Modbus — коммуникационный протокол, основан на архитектуре ведущий-ведомый (master-slave). Для передачи данных использует интерфейсы RS-485, RS-422, RS-232, а также Ethernet сети TCP/IP (протокол Modbus TCP).

Благодаря своей простоте и универсальности Modbus получил широкое распространение и стал де факто стандартом в малых распределенных вычислительных системах. Практически все современные контроллеры поддерживают работу в сетях Modbus.

В сети Modbus контроллеры, как правило, соединены по топологии «Общая шина». Взаимодействие контроллеров происходит в соответствии с моделью ведущий-ведомый.

В сети есть главное устройство — ведущее, а также несколько подчиненных устройств — ведомые. Обмен может быть инициирован только ведущим устройством.

Транзакция (последовательность операций при обмене данными) состоит из запроса и ответа.

Ведущее устройство может адресовать запрос любому ведомому контроллеру или инициировать широковещательное сообщение для всех ведомых устройств одновременно.

Ведомое устройство, определив свой адрес в запросе, формирует ответ.

В запросе от ведущего устройства обязательно содержится код функции, т. е. что надо делать. В зависимости от функции в запросе также могут содержаться данные.

Существуют три варианта протокола Modbus:

Modbus ASCII — текстовый протокол. В нем используются только ASCII символы. Каждый байт передается как два шестнадцатеричных символа.

Modbus RTU — числовой протокол. Данные передаются в двоичном виде. Байт, передаваемый по сети, — это число протокола.

Modbus TCP — протокол для передачи данных в сетях TCP/IP.

Контрольные вопросы и задания

1. Дайте определение понятия «интерфейс».
2. Назовите виды интерфейсов.
3. Как производится организация системных интерфейсов?

10. Организация ввода-вывода в микропроцессорной системе

4. Поясните принцип работы контроллеров периферийных устройств.
5. Расскажите об устройстве и назначении портов ввода-вывода.
6. Как происходит прерывание в микропроцессорной системе?
7. Каким образом осуществляется порядок организации прямого доступа к памяти?

11. ВЫЧИСЛИТЕЛЬНЫЕ И МИКРОПРОЦЕССОРНЫЕ УСТРОЙСТВА

11.1. Общие сведения

При построении микропроцессорных систем (МПС) используют два основных принципа обмена информацией: модульный и магистральный.

Модульный принцип предполагает, что система строится на основе ограниченного количества типов конструктивно и функционально завершенных модулей. Исходя из классической схемы компьютера, любая МПС должна состоять, как минимум, из модуля процессора, модуля памяти и модуля управления вводом-выводом. Основой модуля процессора является микропроцессор.

Магистральный принцип обмена информацией определяет характер связей между модулями МПС. Существуют два принципа взаимодействия элементов модулей и самих модулей в системе: принцип произвольных связей, реализующих правило «каждый с каждым», и принцип упорядоченных связей — магистральный или шинный.

При использовании *принципа произвольных связей* все сигналы между устройствами передаются по отдельным линиям связи. Каждое устройство, входящее в систему, передает свои сигналы и коды независимо от других устройств. При этом в системе получается много линий связи и разных правил (протоколов) обмена информацией.

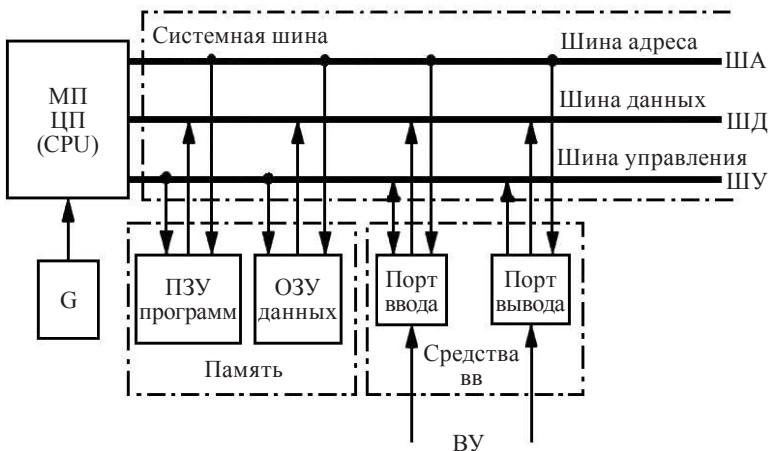
При *магистральном принципе* все сигналы между устройствами передаются по одним и тем же линиям связи, но в разное время (это называется временным мультиплексированием). Обмен

информацией по отдельным линиям связи может осуществляться в одном направлении (однаправленная передача) или в двух направлениях (двухнаправленная передача). Группу линий связи, по которым передаются сигналы и коды, называют *магистралью*, или *шиной*.

Применение принципа магистрального обмена позволяет минимизировать число связей между устройствами системы, сократить число выводов БИС. Большое достоинство магистрального принципа связи состоит в том, что все устройства должны принимать и передавать информацию по одним и тем же правилам (протоколам обмена). Это обеспечивает стандартизацию интерфейсов всех устройств, входящих в систему.

При использовании магистрального принципа обмена информацией все модули МПС соединяются с единой магистралью, которую часто называют *системной шиной*. Шина представляет собой набор электрических проводников, объединенных функционально и часто физически, например на печатной плате.

На рисунке приведена типичная структура МПС, состоящая из модуля процессора, модуля памяти и модуля ввода-вывода. По сложившейся практике при описании МПС слово *модуль* опускают и говорят: процессор, память, устройство ввода-вывода (УВВ).



Магистранно-модульная структура микропроцессорных систем

Весь информационный поток, циркулирующий в МПС, обычно разделяют на три группы: адреса, данные и сигналы управления. В соответствии с этим в системной шине (магистрале) выделяются три шины нижнего уровня:

- шина адреса – ША;
- шина данных – ШД;
- шина управления – ШУ.

Шина адреса предназначена для однозначного определения адреса элемента МПС (например, ячейки памяти или устройства ввода-вывода). Обычно адрес задает процессор, поэтому шина адреса чаще всего однонаправленная.

Шина данных служит для обмена данными между элементами МПС. Она всегда двунаправленная.

Шина управления предназначена для управления работой элементов МПС. По ней передаются управляющие сигналы. Отдельные линии шины управления могут быть одно- или двунаправленными.

Для организации шин используют специальные буферные микросхемы (шинные формирователи), отличающиеся высокой нагрузочной способностью. Поэтому возможность подключения к шине нескольких входов логических элементов ограничивается лишь нагрузочной способностью буферных элементов, к выходу которых эта шина присоединена.

Сложнее организуется подключение выходов нескольких элементов к одной шине. Возможны два способа такого подключения: при помощи схем, имеющих выходы с тремя состояниями, и схем, имеющих выходы с открытым коллектором (открытым стоком).

11.2. Построение и применение вычислительных средств в системах управления

В производственную деятельность прочно входят ЭВМ, и в настоящее время нет необходимости доказывать целесообразность использования вычислительной техники в системах управления технологическими процессами, проектирования, научных исследований, административного управления, в учебном процессе, банковских расчетах, здравоохранении, сфере обслуживания и т. д.

При этом последние годы как за рубежом, так и в Беларуси характеризуются резким увеличением производства мини- и микроЭВМ (персональные ЭВМ).

На основе мини- и персональных ЭВМ можно строить локальные сети ЭВМ, что позволяет решать сложные задачи по управлению производством.

Исследования показали, что из всей информации, образующейся в организации, 60–80 % используется непосредственно в этой же организации, циркулируя между подразделениями и сотрудниками, и только оставшаяся часть в обобщенном виде поступает в министерства и ведомства. Это значит, что средства вычислительной техники, рассредоточенные по подразделениям и рабочим местам, должны функционировать в едином процессе, а сотрудникам организации должна быть поставлена возможность общения с помощью абонентских средств между собой, с единым или распределенным банком данных. Одновременно должна быть обеспечена высокая эффективность использования вычислительной техники.

Решению этой задачи в значительной степени способствовало появление микроэлектронных средств средней и большой степени интеграции, персональных ЭВМ, оборудования со встроенными микропроцессорами. В результате наряду с региональными сетями ЭВМ, построенными на базе крупных ЭВМ и распределенных на большой территории, появились и находят все большее распространение **локальные вычислительные сети** (ЛВС).

Локальные вычислительные сети представляют собой открытую для подключения дополнительных абонентских и вычислительных средств сеть, функционирующую в соответствии с принятыми протоколами (правилами). Устройства обработки, передачи и хранения в ЛВС располагаются друг от друга на расстоянии до нескольких километров, т. е. в пределах одного или группы зданий.

Взаимодействие устройств ЛВС осуществляется по единому каналу связи (моноканалу), обеспечивающему высокую скорость передачи информации (до 10–15 Мбит/с). В сеть могут объединяться ЭВМ как одних типов (однородные сети) или разных типов (неоднородные сети), так и разной производительности. Однородные сети проще и дешевле, так как для их создания необходимы относительно простое оборудование и программное

обеспечение, не требующие большого числа типов средств сопряжения. Это значит, что такие сети создать проще и дешевле.

В настоящее время ЛВС являются универсальной базой современной индустрии обработки информации и характеризуются большим разнообразием методов построения любых видов информации. Концепция локальных сетей ЭВМ — одна из самых полезных системных концепций, возникших в результате длительных научных исследований и прогресса в области микроэлектроники.

Локальные вычислительные сети позволяют небольшим предприятиям воспользоваться возможностью объединить персональные, микро- и миниЭВМ в единую вычислительную сеть, а крупным предприятиям — освободить вычислительный центр от некоторых функций по обработке информации «цехового значения» и обеспечить их решение в цехе, отделе. Кроме того, эксплуатация сети одним заказчиком позволит упростить решение вопроса о закрытии информации.

Использование ЛВС дает высокий экономический эффект. Например, создание сквозного маршрута проектирования микропроцессоров на базе ЛВС позволило уменьшить сроки разработки на 35 % и одновременно снизить стоимость на 48 %. При этом специалисты-разработчики могут находиться на своих рабочих местах и вести совместное проектирование с использованием абонентских средств. «Узкие» места изделия определяются при проектировании, что позволило сократить объем работ при доводке изделия до промышленного образца в 2 раза. Одновременно обеспечивается автоматизация разработки документации.

По своей архитектуре (структуре) ЛВС являются упрощенным вариантом архитектуры региональных и глобальных сетей ЭВМ и могут создаваться на базе любых ЭВМ.

Внедрение ЛВС доступно массовому пользователю и позволяет создать в организациях и учреждениях распределенные вычислительные мощности и базы данных, информационно-поисковые и справочные службы, объединить в единую систему автоматизированные рабочие места, печатающие и копирующие устройства, графопостроители, кассовые аппараты и т. д. ЛВС обеспечивают надежность обработки информации благодаря дублированию ресурсов сети, редактирование писем, справок, отчетов, осуществляют обмен документами без распечатки их на

бумажном носителе, ведение бухгалтерского и складского учета, управление роботами, машинами, станками, передачу информации в заданное время, используют систему приоритетов, направляют циркулярные распоряжения всем, некоторым или одному подразделению организации, проводят телесовещания.

По мере развития ЛВС можно изменить ее конфигурацию, объединить с другими ЛВС (например, на крупном предприятии или объединении), подключить ее к региональной вычислительной сети, что позволит реализовать интегрированные автоматизированные системы управления (АСУ). На определенном этапе развития ЛВС может стать безбумажным бюро, в котором информация записывается на магнитные диски, ленты с возможностью при необходимости получить твердую копию и ее размножение, а также, наоборот, получить машинные носители с твердой копии.

Все многообразие ЛВС условно можно разделить на четыре группы:

- ориентированные на массового потребителя и строящиеся в основном на базе персональных ЭВМ;
- включающие, кроме персональных ЭВМ, микроЭВМ и микропроцессоры, встроенные в средства автоматизированного проектирования и разработки документальной информации, электронной почты;
- построенные на базе микропроцессорных средств, мини- и микроЭВМ и ЭВМ средней производительности;
- создаваемые на базе всех типов ЭВМ, включая высокопроизводительные.

Первые из них применяют в учебных процессах, торговле, мелких и средних учреждениях, вторые — в системах автоматизированного проектирования и конструирования (САПР), третьи — в автоматизированных системах научных исследований (АСНИ), управления сложными производственными процессами и гибких автоматизированных производствах, четвертые — в системах управления крупным производством, отраслью.

Внедрение ЛВС окажет серьезное влияние на организацию производства, где информационно-управляющие системы будут связаны с автоматизированными технологическими системами. Одновременно ЛВС, ориентированные на автоматизацию основных направлений деятельности предприятий, могут быть связа-

ны с системами обработки информации объединений, главков, министерств. При этом будет значительно повышена скорость обмена информацией на всех уровнях управления, т. е. будет создана иерархическая сеть обмена информацией.

При решении вопроса о создании ЛВС должно быть проведено обследование объекта автоматизации и определены количество и тип устройств, включаемых в сеть, условия эксплуатации сети, расстояния между объектами сети, интенсивность потока данных, максимальная скорость передачи данных, необходимость обеспечения приоритетности обслуживания абонентов сети, максимальное время ожидания для оператора рабочей станции, необходимость реализации режима диалога, расширения или изменения конфигурации сети в будущем, затраты на создание и эксплуатацию сети и другие параметры.

Структура ЛВС должна четко соответствовать организационной структуре объекта автоматизации и его информационным связям, а также учитывать полный спектр проблем, связанных с ее использованием в течение периодов максимальной нагрузки. Это значит, что на каждую ЛВС для конкретного объекта необходимо иметь проектную документацию, ориентированную на промышленные технические и программные средства.

Для решения проблемы массового внедрения локальных сетей ЭВМ промышленными министерствами в соответствии с единой нормативной документацией и стандартами должен быть создан ряд комплексов технических и программных средств для ЛВС, ориентированных на разное максимальное число подключаемых к сети узлов и скорость передачи информации, с технико-экономическими характеристиками на уровне лучших образцов и обеспечена поставка их потребителям как комплектных изделий производственно-технического назначения. При этом должны быть разработаны средства сопряжения с ЛВС широкой номенклатуры средств вычислительной техники, имеющейся у потребителей и планируемой к освоению в производстве. Наиболее реальным направлением решения этой проблемы является организация выпуска специализированных СБИС.

Решение указанных проблем, безусловно, окажет серьезное влияние на эффективность всего народного хозяйства.

Главными системными применениями вычислительной техники являются автоматизированные системы управления

экономико-организационного типа (ОАСУ, АСУП и т. п.), системы автоматизации проектирования и конструирования (САПР), информационно-поисковые системы и системы управления сложными технологическими процессами (АСУ ТП). Остановимся кратко на последних системах, так как они дают наибольший социальный и экономический эффект.

Сегодня технологические процессы постоянно усложняются, а агрегаты, реализующие их, выполняются все более мощными. Например, в энергетике действуют энергоблоки мощностью 1000–1500 МВт, установки первичной переработки нефти пропускают до 6 млн т сырья в год, работают доменные печи объемом 3,5–5,0 тыс. м³, создаются гибкоперестраиваемые производственные системы в машиностроении. Человек не может уследить за работой таких агрегатов и технологических комплексов и тогда на помощь ему приходит АСУ ТП.

В АСУ ТП за работой технологического комплекса следят многочисленные датчики-приборы, изменяющие параметры технологического процесса (например, температуру и толщину прокатываемого металлического листа), контролирующие состояние оборудования (температуру подшипников турбины) или определяющие состав исходных материалов и готового продукта. Таких приборов в одной системе может быть от нескольких десятков до нескольких тысяч.

Датчики постоянно выдают сигналы, меняющиеся в соответствии с измеряемым параметром (аналоговые сигналы), в устройство связи с объектом (УСО) ЭВМ. В УСО сигналы преобразуются в цифровую форму и затем по определенной программе обрабатываются вычислительной машиной. ЭВМ сравнивает полученную от датчиков информацию с заданными результатами работы агрегата и вырабатывает управляющие сигналы, которые через другую часть УСО поступают на регулирующие органы агрегата.

Например, если датчики подали сигнал, что лист прокатного стана выходит толще, чем предписано, то ЭВМ вычислит, на какое расстояние нужно сдвинуть валки прокатного стана, и подаст соответствующий сигнал на исполнительный механизм, который переместит валки на требуемое расстояние.

Системы, в которых управление ходом процесса осуществляется подобно сказанному выше без вмешательства человека,

называют автоматическими. Однако, если не известны точные законы управления, человек вынужден брать управление (определение управляющих сигналов) на себя (такие системы называются автоматизированными). В этом случае ЭВМ предоставляет оператору всю необходимую информацию для управления технологическим процессом при помощи дисплеев, на которых данные могут высвечиваться в цифровом виде или в виде диаграмм, характеризующих ход процесса, могут быть представлены и технологические схемы объекта с указанием состояния его частей. ЭВМ может также «подсказать» оператору некоторые возможные решения.

Чем сложнее объект управления, тем производительнее, надежнее требуется для АСУ ТП вычислительная машина. Чтобы избежать все увеличивающегося наращивания мощности ЭВМ, сложные системы стали строить по иерархическому принципу. Как правило, в сложный технологический комплекс входит несколько относительно автономных агрегатов, например в энергоблок тепловой электростанции входят парогенератор (котел), турбина и электрогенератор. В иерархической системе для каждой составной части создается своя локальная система управления, как правило, автоматическая на базе микропроцессорной техники. Теперь, чтобы все части работали как единый энергоблок, необходимо скоординировать работу локальных систем. Это осуществляется ЭВМ, устанавливаемой на пульте управления блоком. Для этого уже потребуется небольшая вычислительная машина.

Перспективные АСУ ТП имеют ряд характерных признаков. Прежде всего это автоматические системы, осуществляющие автоматическое управление рабочим режимом, а также пуском и остановом оборудования (режимами, на которые при ручном управлении приходится наибольшее число аварийных ситуаций из-за ошибок операторов).

В системах предусматривается оптимизация управления ходом процесса по выбранным критериям. Например, можно задать такие параметры процесса, при которых стоимость, себестоимость продукции будут минимальными, или, если необходимо, настроить агрегат на максимум производительности, не считаясь с некоторым увеличением расхода сырья и энергоресурсов на единицу продукции.

Системы должны быть адаптивными, т. е. иметь возможность изменять ход процесса при изменении характеристик исходных материалов или состояния оборудования.

Одним из важнейших свойств АСУ ТП является обеспечение безаварийной работы сложного технологического комплекса. Для этого в АСУ ТП предусматривается возможность диагностирования технологического оборудования. На основе показаний датчиков система определяет текущее состояние агрегатов, тенденции к аварийным ситуациям и может дать команду на ведение облегченного режима работы или остановку вообще. При этом оператору предоставляют данные о характере и местоположении аварийных участков.

Таким образом, АСУ ТП обеспечивают лучшее использование ресурсов производства, повышение производительности труда, экономию сырья, материалов и энергоресурсов, исключение тяжелых аварийных ситуаций, увеличение межремонтных периодов работы оборудования.

11.3. Контроль работы вычислительных средств

Несмотря на относительно невысокую стоимость современных элементов технологической среды сферы информатизации, общая стоимость вычислительных комплексов на многих предприятиях становится сопоставимой со стоимостью основных производственных фондов (их активной части), не говоря о тех предприятиях, где вычислительная техника и есть производительное оборудование. Кроме того, по мере освоения и распространения средств информатизации руководители разных уровней управления предприятием осознают, что информационные технологии и информационные системы – не просто средство, облегчающее выполнение вычислительных процедур, но мощный инструмент выработки и оптимизации управленческих решений, что в конечном счете обеспечивает эффективность деятельности организации. Все это привело к тому, что проблема рационального использования вычислительных средств и информационных ресурсов становится все более актуальной.

В силу определенной специфики функционирования вычислительных комплексов и сферы информатизации в целом одним из наиболее доступных и очевидных путей повышения их отдачи является интенсификация использования либо во времени, либо по мощности, либо и то и другое вместе.

Говоря об интенсификации использования вычислительных комплексов и информационных ресурсов во времени, следует иметь в виду, что рассматриваемые процессы носят дискретный характер, так как обусловлены обязательным взаимодействием «человек — машина». Кроме того, быстроедействие современных ЭВМ позволяет производить вычислительные процедуры практически мгновенно, а значит, время выполнения одного производственного задания на одном рабочем месте значительно короче продолжительности рабочей смены, установленной режимом работы предприятия. Отсюда структура фонда машинного времени должна быть аналогичной дискретному производству (например, в промышленности), но с учетом особенностей технологического процесса работы с информацией.

При анализе использования вычислительных комплексов во времени исходным показателем является эффективный фонд рабочего времени одного рабочего места (или просто — рабочее время), который определяется исходя из количества рабочих дней в расчетном периоде (устанавливается Правительством), продолжительности рабочей смены или дня (регулируется Трудовым кодексом) и коэффициента сменности работы предприятия (устанавливается на основании коллективного договора данного предприятия). Рабочее время уменьшается на величину потерь, обусловленных необходимостью проведения планово-предупредительных ремонтов, диагностики, испытаний и т. п. Эти потери являются оправданными, поскольку связаны с поддержанием элементов технологической среды в работоспособном состоянии.

Говоря об интенсификации использования вычислительных комплексов во времени, в общем фонде рабочего времени следует выделить долю полезного машинного времени. Это возможно на основе классификации затрат рабочего времени, которая позволяет проводить анализ целесообразности использования рабочего времени по отношению к оборудованию, работнику и производственному процессу обработки информации в целом.

В общем случае рабочее время складывается из времени выполнения полезной работы и времени перерывов. В свою очередь время выполнения полезной работы включает в себя время на выполнение производственного задания и потери времени, не обусловленные технологическим процессом обработки информации и производственным заданием. Время перерывов может не

зависеть от работника (время, необходимое для обмена данными, время ожидания, связанное с разной производительностью смежных технических средств и т. п.) и зависеть от работника (время на отдых и личные надобности, необоснованное отсутствие на рабочем месте, опоздание на работу и т. п.).

Время на выполнение производственного задания включает в себя следующие элементы:

- *подготовительно-заключительное время* (ознакомление с производственным заданием, подготовка первичной информации, установление взаимодействия всех компонентов технологического процесса, безопасное отключение внешних технических устройств по окончании их использования, регистрация результатов работы по окончании рабочей смены и т. п.);

- *машинное время* (выполнение всех технологических операций, связанных непосредственно с обработкой информации, иными словами, время взаимодействия «человек—машина» при выполнении производственного задания);

- *вспомогательное время* (подключение/отключение внешних устройств в процессе выполнения работы, вспомогательные ручные операции, текущее тестирование и т. п.);

- *время обслуживания* (техническое обслуживание — устранение различного рода сбоев, переключение с одной технологии на другую и организационное обслуживание — удобная организация рабочего места).

К потерям времени, не обусловленным производственным заданием, относят:

- работу на себя (время, когда элемент технологического комплекса находится в состоянии программно-технического обслуживания или ремонта вследствие неожиданного отказа);

- резерв — не востребованность вычислительных ресурсов.

Каждый из выделенных показателей может стать основой для оценки эффективности использования вычислительных комплексов. При этом уровень интенсивности их использования определяется отношением времени, затраченного на выполнение производственного задания, к суммарной величине рабочего времени. Этот подход может быть использован для построения базовых соотношений различных элементов рабочего времени и далее — при планировании и оценке фактических составляющих эффективного фонда времени для каждого вычислительного подразделения и для сферы обработки информации предприятия в

целом. На этой основе могут строиться варианты стратегии снижения непроизводительных затрат рабочего времени по всем его отдельным составляющим и в совокупности.

Что касается оценки интенсивности использования вычислительных ресурсов по мощности, т. е. определения количества работы (продукции) в единицу времени, то решение этой задачи наталкивается на определенные трудности. В принципе вычислительная система может быть охарактеризована некоторой достижимой потенциальной мощностью, которую она способна развивать в том или ином процессе. Это можно сделать на основе информации о технической производительности отдельных элементов, а также в процессе специального их тестирования. Однако такая оценка будет не точна, так как она должна базироваться не только на совокупной мощности отдельных элементов, но и с учетом особенностей организации их взаимодействия.

Поскольку вычислить мощность (производительность) вычислительного комплекса точными методами практически невозможно, следует использовать эмпирические методы. Например, можно набрать статистику на основе стандартных процедур контроля производительности за ряд лет и на основе регрессионных зависимостей создать опытно-статистические нормативы. Другой путь – проведение эксперимента на основе имитационного моделирование загрузки вычислительной системы в целях определения предельных параметров.

Полученные оценки можно принять в качестве исходных значений нормативов, а затем в процессе контроля и анализа параметров эксплуатации реальной вычислительной системы совершенствовать нормативную базу управления использованием ресурсов сферы информатизации предприятия.

Контрольные вопросы и задания

1. Дайте определение микропроцессорной системы.
2. Какие особенности возникают при построении микропроцессорной сети?
3. Опишите процессы построения и применения вычислительных систем в системах управления.
4. Как осуществляется порядок контроля работы вычислительных систем?

12. ОСНОВЫ ПРОГРАММИРОВАНИЯ

12.1. Языки программирования и их классификация

Язык программирования — язык (знаковая система), предназначенный для записи компьютерных программ.

Язык программирования определяет набор лексических, синтаксических и семантических правил, определяющих внешний вид программы и действия, которые выполнит исполнитель (обычно ЭВМ) под управлением программы. Он предназначен для написания компьютерных программ, которые представляют собой набор правил, позволяющих компьютеру выполнить тот или иной вычислительный процесс, организовать управление различными объектами и т. п. Язык программирования отличается от естественных языков тем, что предназначен для взаимодействия человека с ЭВМ, в то время как естественные языки используют для общения людей между собой.

Программирование ПЛК (программируемые логические контроллеры) осуществляется с помощью стандарта МЭК 61131-3. Стандарт МЭК 61131-3 устанавливает пять языков программирования ПЛК, три графических и два текстовых. Первоначально стандарт назывался IEC 1131-3 и был опубликован в 1993 г., но в 1997 г. МЭК (IEC) перешел на новую систему обозначений и в названии стандарта добавилась цифра «6».

Основная цель стандарта — повышение скорости и качества разработки программ, а также создание языков программирования, ориентированных на технологов, обеспечение соответствия ПЛК идеологии открытых систем, исключение этапа дополнительного обучения при смене типа ПЛК.

Системы программирования, основанные на МЭК 61131-3, характеризуются следующими показателями:

- надежностью создаваемого программного обеспечения. Надежность обеспечивается тем, что программы для ПЛК создаются с помощью специально предназначенной для этого среды разработки, которая содержит все необходимые средства для написания, тестирования и отладки программ с помощью эмуляторов и реальных ПЛК, а также множество готовых фрагментов программного кода;
- возможностью простой модификации программы и наращивания ее функциональности;
- переносимостью проекта с одного ПЛК на другой;
- возможностью повторного использования отработанных фрагментов программы;
- простотой языка и ограничением количества его элементов.

Языки МЭК 61131-3 появились не как теоретическая разработка, а как результат анализа множества языков, уже используемых на практике и предлагаемых рынку производителями ПЛК. Стандарт устанавливает пять языков программирования со следующими названиями:

- структурированный текст (ST – Structured Text);
- последовательные функциональные схемы (SFC – Sequential Function Chart);
- диаграммы функциональных блоков (FBD – Function Block Diagram);
- релейно-контактные схемы, или релейные диаграммы (LD – Ladder Diagram);
- список инструкций (IL – Instruction List).

Графическими языками являются SFC, FBD, LD, текстовыми – IL и ST.

В стандарт было введено несколько языков (а не один) для того, чтобы каждый пользователь мог применить наиболее понятный ему язык. Программисты чаще выбирают язык IL, похожий на ассемблер, или ST, похожий на язык высокого уровня Паскаль. Специалисты, имеющие опыт работы с релейной логикой, выбирают язык LD, специалисты по системам автоматического управления (САУ) и схемотехники выбирают привычный для них язык FBD.

Выбор одного из пяти языков определяется не только предпочтениями пользователя, но и смыслом решаемой задачи. Если

исходная задача формулируется в терминах последовательной обработки и передачи сигналов, то для нее проще и нагляднее использовать язык FBD. Если задача описывается как последовательность срабатываний некоторых ключей и реле, то для нее нагляднее всего будет язык LD. Для задач, которые изначально формулируются в виде сложного разветвленного алгоритма, удобнее будет язык ST.

Языки по стандарту МЭК 61131-3 базируются на следующих принципах:

- вся программа разбивается на множество функциональных элементов – Program Organization Units (POU), каждый из которых может состоять из функций, функциональных блоков и программ. Любой элемент МЭК-программы может быть сконструирован иерархически из более простых элементов;

- стандарт требует строгой типизации данных. Указание типов данных позволяет легко обнаруживать большинство ошибок в программе до ее исполнения;

- имеются средства для исполнения разных фрагментов программы в разное время, с разной скоростью, а также параллельно. Например, один фрагмент программы может сканировать концевой датчик с частотой 100 раз в секунду, в то время как второй фрагмент будет сканировать датчик температуры с частотой один раз в 10 с;

- для выполнения операций в определенной последовательности, которая задается моментами времени или событиями, используется специальный язык последовательных функциональных схем (SFC);

- стандарт поддерживает структуры для описания разнородных данных. Например, температуру подшипников насоса, давление и состояние «включено-выключено» можно описать с помощью единой структуры и передавать ее внутри программы как единый элемент данных;

- стандарт обеспечивает совместное использование всех пяти языков, поэтому для каждого фрагмента задачи может быть выбран любой наиболее удобный язык;

- программа, написанная для одного контроллера, может быть перенесена на любой контроллер, совместимый со стандартом МЭК 61131-3.

Любой ПЛК работает в циклическом режиме. Цикл начинается со сбора данных с модулей ввода, затем выполняется про-

грамма ПЛК и оканчивается цикл выводом данных в устройства вывода. Поэтому величина контроллерного цикла зависит от времени исполнения программы и быстродействия процессорного модуля.

Язык релейно-контактных схем (LD). Графический язык релейной логики впервые появился в виде электрических схем, которые состояли из контактов и обмоток электромагнитных реле. Такие схемы использовались в автоматике конвейеров для сборки автомобилей до эры микропроцессоров. Язык релейной логики был интуитивно понятен людям, слегка знакомым с электротехникой, и поэтому оказался наиболее распространенным в промышленной автоматике. Обслуживающий персонал легко находил отказ в оборудовании, прослеживая путь сигнала по релейной диаграмме.

Однако язык LD проблематично использовать для реализации сложных алгоритмов, поскольку он не поддерживает подпрограммы, функции, инкапсуляцию (упаковка данных и функций в единый компонент) и другие средства структурирования программ с целью повысить качество программирования. Эти недостатки затрудняют многократное использование программных компонентов, что делает программу длинной и сложной для обслуживания (рис. 12.1).

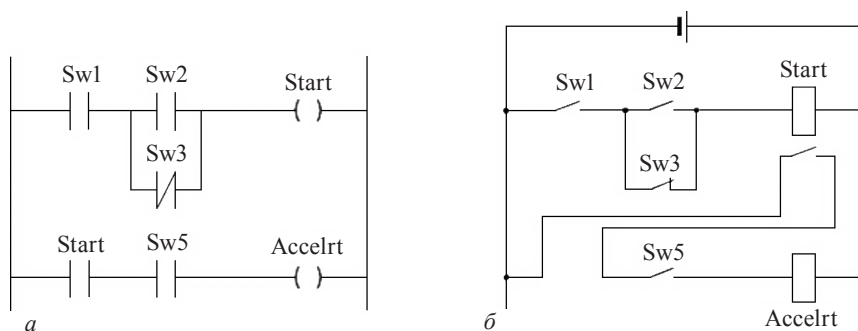


Рис. 12.1. Пример программы на языке LD (а) и ее эквивалент в виде электрической цепи с реле и выключателями (б)

Для выполнения арифметических функций в язык LD были добавлены функциональные блоки, которые выполняли операции сложения, умножения, вычисления среднего и т. д. Сложные вычисления в этом языке невозможны. Недостатком является

также то, что только малая часть программы умещается на мониторе компьютера или панели оператора при программировании.

Несмотря на указанные недостатки, язык LD относится к наиболее распространенным в мире, хотя используется для программирования только простых задач.

Список инструкций (IL). Язык IL напоминает ассемблер, его применяют для реализации функций, функциональных блоков и программ, а также шагов и переходов в языке SFC. Основным достоинством языка является простота его изучения. Чаще язык IL используют в случаях, когда требуется получить оптимизированный код для реализации критических секций программы, а также для решения небольших задач с малым количеством разветвлений алгоритма.

В основе языка лежит понятие аккумулятора и переходов по меткам. Пример программы на языке IL с комментариями приведен ниже. Начинается программа с загрузки в аккумулятор значения переменной. Дальнейшие шаги программы состоят в извлечении содержимого аккумулятора и выполнении над ним ограниченного числа допустимых действий (их в языке всего 24).

Пример программы на языке IL.

Метки	Операторы	Операнды	Комментарии
	LD	Voltage	(*Загрузить Voltage в аккумулятор*)
	GT	220	(*Если >220*)
	JMPCN	M1	(*Перейти к метке, если «>220» не верно*)
	LD	Current	(*Загрузить Current в аккумулятор*)
	SUB	10	(*Вычесть из аккумулятора 10*)
	ST	Current	(*Присвоить Current значение аккумулятора*)
M1:	LD	0	(*Загрузить в аккумулятор значение «0»*)
	ST	Out	(*Присвоить Out значение аккумулятора*)

Структурированный текст (ST). Язык ST является текстовым языком высокого уровня и очень напоминает язык Паскаль.

Пример программы на языке ST:

```
IF Voltage>220 THEN
```



```

Current:=Current - 10; (*Если V>220 В, то уменьшить ток
на 10*)
ELSE
Current:=50; Speed:= ON; (*Установить ток 50 А и включить
мотор*)
END_IF;

```

Язык ST отличается от языка Паскаль и разработан специально для программирования ПЛК. Он содержит множество конструкций для присвоения значений переменным, для вызова функций и функциональных блоков, написания выражений условных переходов, выбора операторов, построения итерационных процессов. Этот язык предназначен в основном для выполнения сложных математических вычислений, описания сложных функций, функциональных блоков и программ.

Диаграммы функциональных блоков (FBD). Графический язык FBD предназначен для программирования процессов прохождения сигналов через функциональные блоки. Он удобен для схемотехников, которые легко могут составить электрическую схему системы управления на «жесткой логике», но не имеют опыта программирования.

Функциональные блоки представляют собой фрагменты программ, написанных на IL, SFC или других языках, которые могут быть многократно использованы в разных частях программы и которым соответствует графическое изображение, принятое при разработке функциональных схем электронных устройств (рис. 12.2).

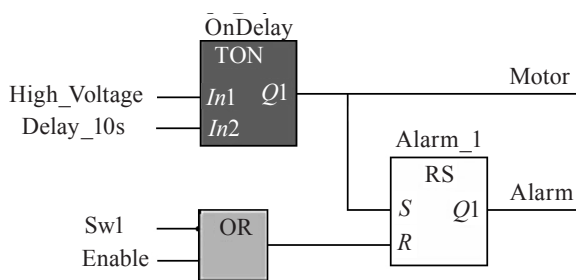


Рис. 12.2. Пример программы на языке FBD

Язык FBD можно использовать для программирования функций, функциональных блоков и программ, а также для описания

шагов и переходов в языке SFC. Функциональные блоки инкапсулируют данные и методы, напоминая объектно-ориентированные языки программирования, но не поддерживают наследование и полиморфизм.

Типичным применением языка FBD является описание «жесткой логики» и замкнутых контуров систем управления. Язык функциональных блоков также удобен для создания и пополнения библиотеки типовых функциональных блоков, которую можно многократно использовать при программировании задач промышленной автоматизации. К типовым блокам относят блок таймера, ПИД-регулятора, блок секвенсора, триггера, генератора импульсов, фильтра и т. п.

Последовательные функциональные схемы (SFC). Язык программирования SFC специально предназначен для программирования последовательности выполнения действий системой управления, когда эти действия должны быть выполнены в заданные моменты времени или при наступлении некоторых событий. В его основе лежит представление системы управления с помощью понятий состояний и переходов между ними.

Язык SFC применяют для описания системы управления на самом верхнем уровне абстракции, например в терминах «Старт», «Наполнение автоклава», «Выполнение этапа № 1», «Выполнение этапа № 2», «Выгрузка из автоклава». Язык SFC может быть использован также для программирования отдельных функциональных блоков, если алгоритм их работы естественным образом описывается с помощью понятий состояний и переходов. Например, алгоритм автоматического соединения модема с коммутируемой линией описывается состояниями «Включение», «Обнаружение тона», «Набор номера», «Идентификация сигнала» и переходами «Если длинный, то ждать 20 с», «Если короткий, перейти в состояние «Набор номера».

На рисунке 12.3 приведен фрагмент программы на языке SFC. Программа состоит из шагов и условий переходов, выполняется сверху вниз. Шаги на схеме показаны прямоугольниками, условия переходов – жирной перечеркивающей линией, начальный шаг – в виде двойного прямоугольника. Условия переходов записываются рядом с их обозначениями. Каждый шаг программы может представлять собой реализацию сложного алгоритма, написанного на одном из МЭК-языков.

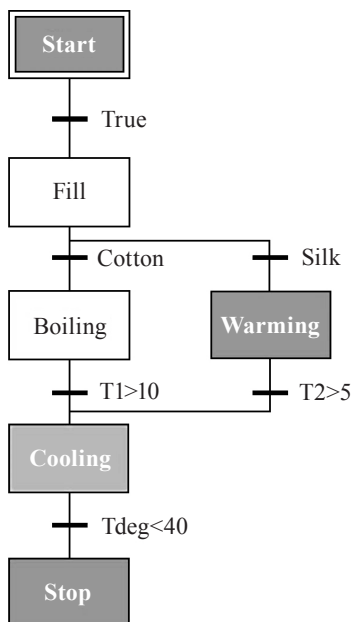


Рис. 12.3. Пример программы на языке SFC

Программное обеспечение. Программирование ПЛК на описанных выше языках по стандарту МЭК 61131-3 осуществляется с помощью специализированного программного обеспечения, которое разрабатывается производителями ПЛК или фирмами, специализирующимися на создании программного обеспечения для систем автоматизации. Наиболее известными в мире являются системы CoDeSys фирмы 3S и ISaGRAF фирмы ICS Triplex.

Система CoDeSys (Controller Development System) представляет собой комплекс программ для проектирования прикладного программного обеспечения, отладки в режиме эмуляции и загрузки программы в ПЛК. Основными частями системы являются среда разработки программы и среда ее исполнения (CoDeSys SP), которая находится в ПЛК.

В CoDeSys входят графические и текстовые редакторы для всех пяти языков стандарта МЭК 61131-3. Этот комплекс полностью реализует требования стандарта и дополнительно вводит ряд оригинальных расширений, самым удобным из которых является объектно-ориентированное программирование. Однако расшире-

ниями языка можно не пользоваться, чтобы сохранить требование к совместимости языков, предъявляемое к открытым системам.

В одном проекте можно использовать несколько контроллеров разных производителей. Каждый из них может программироваться как независимое устройство или с учетом их взаимодействия в промышленной сети. Проект состоит из нескольких приложений, распределенных по нескольким контроллерам. В одном ПЛК может существовать несколько независимых приложений.

Система ISaGRAF также состоит из среды разработки и среды исполнения. Среда исполнения может функционировать практически на любой операционной системе и любой аппаратной платформе, включая персональный компьютер. Среда разработки поддерживает все пять языков стандарта МЭК 61131-3 и функциональные блоки стандарта МЭК 61499, имеет средства для редактирования, компиляции, документирования, управления библиотеками, архивирования, моделирования системы при отсутствии реального ПЛК и отладки с подключенным ПЛК.

Связь между пакетом SCADA и контроллером, запрограммированным с помощью системы ISaGRAF, осуществляется с помощью стандартного OPC-сервера.

Среда исполнения создается и загружается в контроллер производителем ПЛК и является независимой от исполняемой в ней программы пользователя.

Среда разработки имеет знакомый по Windows-приложениям интерфейс с подсказками, панелями инструментов, окнами, с функциями вставки и замены и т. п. Код, полученный на выходе среды разработки, может исполняться на любой аппаратно-программной платформе без изменений, если на ней предварительно установлена среда исполнения. Среда разработки может также транслировать пользовательскую программу, написанную на МЭК-языках, в текст на языке Си.

12.2. Алгоритм программы и его свойства

В 1983 г. отмечалось 1200-летие со дня рождения величайшего средневекового ученого Средней Азии Мухаммада ибн Мусы аль-Хорезми. С именем этого ученого связано понятие алгоритма. Итак, **алгоритм** — это определенная последовательность действий, которые необходимо выполнить, чтобы получить результат. Алгоритм может представлять собой некоторую последовательность вычислений или последовательность действий не-

математического характера. Для любого алгоритма справедливы общие закономерности – *свойства алгоритма*.

Дискретность – свойство алгоритма, когда алгоритм разбивается на конечное число элементарных действий (шагов).

Понятность – свойство алгоритма, при котором каждое из этих элементарных действий (шагов) является законченным и понятным.

Детерминированность – свойство, когда каждое действие (операция, указание, шаг, требование) должно пониматься в строго определенном смысле, чтобы не оставалось места произвольному толкованию, чтобы каждый, прочитавший указание, понимал его однозначно.

Массовость – свойство, когда по данному алгоритму должна решаться не одна, а целый класс подобных задач.

Результативность – свойство, при котором любой алгоритм в процессе выполнения должен приводить к определенному результату. Отрицательный результат также является результатом.

Алгоритм может быть записан различными способами: на естественном языке в виде описания, графических блок-схем, на специальном алгоритмическом языке. Запись алгоритмов на родном языке доступна и удобна. Существенным недостатком такой записи является недостаточная наглядность, что особенно сказывается, когда алгоритм имеет много ветвлений. Поэтому наши алгоритмы будем записывать в виде блок-схемы (рис. 12.4).



Рис. 12.4. Основные блоки алгоритмов

Все имеющиеся алгоритмы разделяют на простые (линейные), циклические и ветвящиеся.

Простой (линейный) тип алгоритмов. Алгоритмы, в которых команды выполняются друг за другом, независимо от каких-либо

условий, называют алгоритмами линейного типа. Например, алгоритм вычисления по самым простейшим формулам, не имеющим ограничений на значения входящих в них переменных.

Пример.

Постановка задачи: вычислить площадь круга, если известен радиус.

Дано: R – радиус круга.

Найти: S – площадь круга.

Решение: $S = 3,14R^2$.

Словесная форма записи алгоритма. Для записи алгоритма в этой форме выберем русский язык и запишем последовательность команд, выполнение которых при заданном значении радиуса позволит найти площадь.

1. Прочитать значение R .
2. Умножить значение R на 3,14.
3. Умножить результат второго действия на значение R .
4. Записать полученный результат как значение S .

Форма записи в виде алгоритма (рис. 12.5).

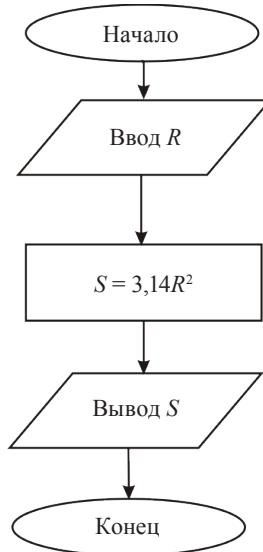


Рис. 12.5. Простой алгоритм

Циклический тип алгоритмов. Циклический алгоритм – описание действий, которые должны повторяться указанное число

раз или пока не выполнено заданное условие. Перечень повторяющихся действий называют телом цикла.

Циклические алгоритмы бывают двух типов:

- *циклы со счетчиком* (рис. 12.6), в которых какие-то действия выполняются определенное число раз;
- *циклы с условием* (рис. 12.7), в которых тело цикла выполняется в зависимости от какого-либо условия.

Циклы со счетчиком используют, когда заранее известно, какое число повторений тела цикла необходимо выполнить. Например, на уроке физкультуры необходимо пробежать некоторое количество кругов вокруг стадиона.

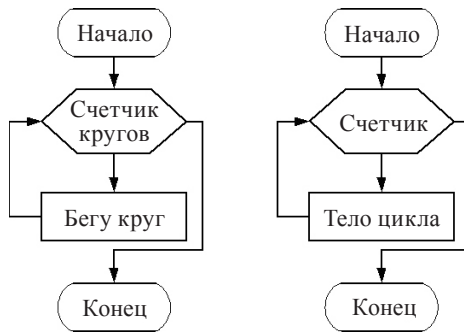


Рис. 12.6. Пример схем циклических алгоритмов со счетчиками

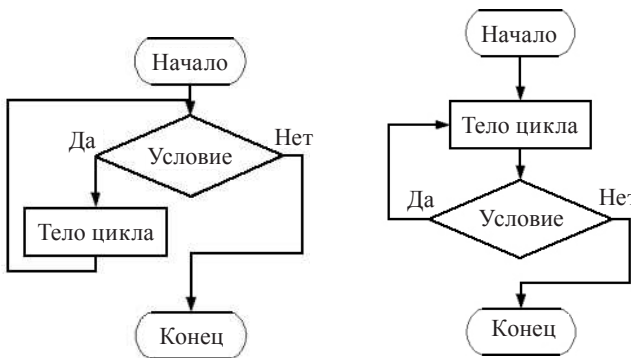


Рис. 12.7. Пример схем циклических алгоритмов с условием

В общем случае схема циклического алгоритма со счетчиком будет выглядеть следующим образом:

Для счетчика от начального значения до конечного значения **выполнить действие**. Часто бывает так, что необходимо повторить тело цикла, но заранее не известно, какое количество раз это надо сделать. В таких случаях количество повторений зависит от некоторого условия. Такие циклы называют циклами с условием. Циклы, в которых сначала проверяется условие, а затем, возможно, выполняется тело цикла, называют циклами с предусловием. Если условие проверяется после первого выполнения тела цикла, циклы называют циклами с постусловием.

В общем случае схема циклического алгоритма с условием будет выглядеть следующим образом:

Пока условие повторять действие.

При составлении циклических алгоритмов важно думать о том, чтобы цикл был конечным. Ситуация, при которой выполнение цикла никогда не заканчивается, называется заикливанием.

Ветвящийся тип алгоритмов. Решение задач не всегда можно представить в виде линейного алгоритма.

Алгоритмы, в которых требуется организовать выбор последовательности действий в зависимости от каких-либо условий, называют алгоритмами разветвляющегося типа.

При графическом способе (рис. 12.8) ветвление организуется с помощью логического элемента (ромб), имеющего один вход и два выхода. Назначение логического элемента – проверка заданного условия. В зависимости от выполнения (истинности) или невыполнения (ложности) проверяемого условия возможен выход соответственно на ветвь «Да» или «Нет».

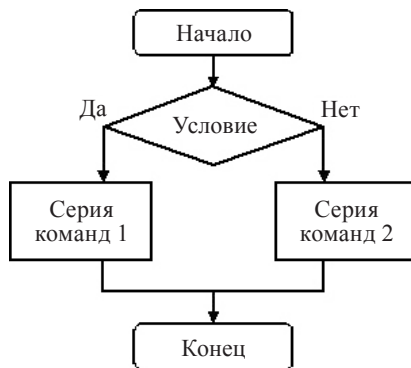


Рис. 12.8. Блок-схема ветвящегося алгоритма

Выделяют *полную* и *неполную* условную конструкцию ветвящихся алгоритмов (рис. 12.9).

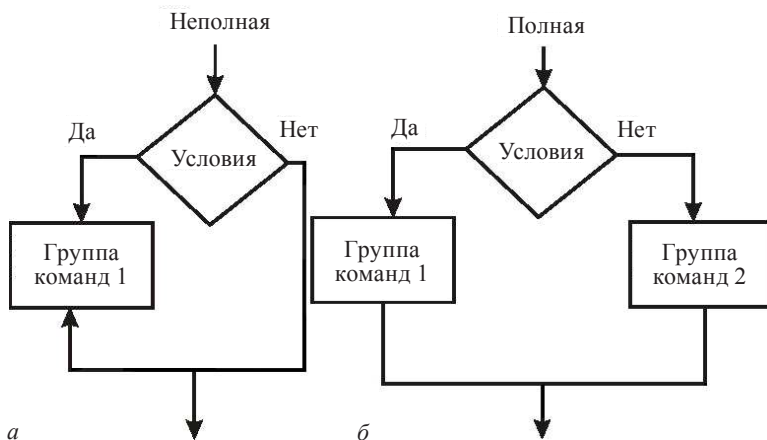


Рис. 12.9. Примеры ветвящихся алгоритмов с условной конструкцией:
 а – неполной; б – полной

12.3. Характеристики программного продукта

Все программы по характеру использования и категориям пользователей можно разделить на два класса: утилитарные программы и программные продукты (изделия).

Утилитарные программы (утилиты) предназначены для использования самими разработчиками. Чаще всего это программные решения функциональных задач, не предназначенных для широкого распространения.

Программный продукт – комплекс взаимосвязанных программ для решения определенной проблемы (задачи) массового спроса, подготовленный к реализации как любой вид промышленной продукции.

Программный продукт должен быть соответствующим образом подготовлен к эксплуатации, иметь необходимую техническую документацию, представлять сервис и гарантию надежной работы программы, иметь товарный знак изготовителя, а также желательно наличие кода государственной регистрации.

Программные продукты предназначены для широкого распространения и продажи.

Процесс создания программных продуктов достаточно долгий, он связан с изменениями технической и программной среды разработки и эксплуатации программ, с появлением и развитием самостоятельной отрасли — информационного бизнеса, для которого характерны разделение труда фирм-разработчиков программ, их дальнейшая специализация, формирование рынка программных средств и информационных услуг.

Программный продукт разрабатывается на основе промышленной технологии выполнения проектных работ с применением современных инструментальных средств программирования. Специфика заключается в уникальности процесса разработки алгоритмов и программ, зависящего от характера обработки информации и используемых инструментальных средств.

Как правило, программные продукты требуют сопровождения, которое осуществляется специализированными фирмами-распространителями программ (дистрибьюторами), реже — фирмами-разработчиками. Сопровождение программ массового применения сопряжено с большими трудозатратами — исправление ошибок, создание новых версий программ и т. п.

12.4. Основные характеристики программ

Основные характеристики программ:

- алгоритмическая сложность (логика алгоритмов обработки информации);
- состав и глубина проработки реализованных функций обработки;
- полнота и системность функций обработки;
- объем файлов программ;
- требования к операционной системе и техническим средствам обработки со стороны программного средства;
- объем дисковой памяти;
- размер оперативной памяти для запуска программ;
- тип процессора;
- версия операционной системы;
- наличие вычислительной сети и др.

Программные продукты имеют многообразие показателей качества.

Мобильность программных продуктов означает их независимость от технического комплекса системы обработки данных,

операционной среды, сетевой технологии обработки данных, специфики предметной области и т. п. Мобильный (многоплатформный) программный продукт может быть установлен на различных моделях компьютеров и операционных систем, без ограничений на его эксплуатацию в условиях вычислительной сети. Функции обработки такого программного продукта пригодны для массового использования без каких-либо изменений.

Надежность работы программного продукта определяется устойчивостью в работе программ, точностью выполнения предписанных функций обработки, возможностью диагностики возникающих в процессе работы программ ошибок.

Эффективность программного продукта оценивается как с позиций прямого его назначения – требований пользователя, так и с точки зрения расхода вычислительных ресурсов, необходимых для его эксплуатации.

Расход вычислительных ресурсов оценивается через объем внешней памяти для размещения программ и объем оперативной памяти для запуска программ.

Учет человеческого фактора означает обеспечение дружелюбного интерфейса для работы конечного пользователя, наличие контекстно-зависимой подсказки или обучающей системы в составе программного средства, хорошей документации для освоения и использования заложенных в программном средстве функциональных возможностей, анализ и диагностику возникших ошибок и др.

Модифицируемость программных продуктов означает способность к внесению изменений, например расширение функций обработки, переход на другую техническую базу обработки и т. п.

Коммуникативность программных продуктов основана на максимально возможной их интеграции с другими программами, обеспечении обмена данными в общих форматах представления (экспорт-импорт баз данных, внедрение или связывание объектов обработки и др.).

Приведем основные этапы создания программного продукта. Всего их пять. Они так или иначе характерны для любой методологии разработки программного обеспечения: будь то классическая каскадная либо современная гибкая методология. Во всех из них разработчики проходят через следующие этапы создания программного обеспечения.

Составление требований заказчика. На данном этапе проводится работа с заказчиком и документирование его видения и требований к программе. Выработка и понимание требований всеми участниками данного этапа – чрезвычайно важное мероприятие.

Проектирование программного продукта. Разобравшись в предметной области, разработчики приступают к проектированию. На данном этапе создания программного продукта разрабатывается архитектура компонентов программного обеспечения, выбираются нужные шаблоны проектирования и составляется схема информационной базы данных системы.

Разработка. Когда требования сформулированы и архитектура готова, приступают к разработке программного продукта. На этапе разработки также выполняется документирование системы.

Тестирование. После разработки необходимо произвести тестирование системы в целом, тем самым подтвердить ее соответствие требованиям заказчика.

Модульные тесты (unit-тесты), т. е. тесты отдельных частей программы, обычно выполняются на этапе разработки программистом конкретного модуля.

Когда все тесты пройдены, программное обеспечение готово к выпуску.

Сопровождение программного продукта. После выпуска фирма-разработчик отвечает за поддержку программного продукта и выпуск новых версий, которые исправляют ошибки и привносят новый функционал. Также необходимо осуществлять поддержку пользователей разработанного программного обеспечения.

Контрольные вопросы и задания

1. Дайте определение понятия «язык программирования».
2. Приведите классификацию языков программирования.
3. Раскройте суть понятия «алгоритм программы».
4. Какими свойствами обладает алгоритм?
5. Назовите типы алгоритмов.
6. Как описывается алгоритм?
7. Расскажите о порядке создания программного продукта.

13. СОВРЕМЕННЫЕ МИКРОПРОЦЕССОРНЫЕ СИСТЕМЫ, ТЕНДЕНЦИИ ИХ РАЗВИТИЯ

Одним из ключевых назначений автоматизации являются наблюдение за изменением состояния объекта и способность контролировать этот процесс. Снижение процессов изменений приводит к повышению производительности и эффективности. Машинное зрение и управление движением помогают уменьшить изменения и добавить гибкости в современные системы автоматизации. В свою очередь, увеличение гибкости и функциональных возможностей систем автоматического регулирования может вытеснить некоторые более старые системы.

Современные программируемые логические контроллеры (ПЛК) являются вполне надежными устройствами, а их языки программирования стандартизированы. Среды разработки программного обеспечения для ПЛК пока не имеют общих единых стандартов, так как все основные игроки рынка электронных компонентов предлагают свои собственные уникальные решения. Программирование, а также поиск и устранение неисправностей в ПЛК может быть даже легче, чем в персональном компьютере, который каждый из нас, казалось бы, знает очень хорошо. Программируемый логический контроллер имеет модульную структуру и возможность подключения различных модулей в зависимости от требований проекта: дополнительные порты ввода-вывода, модули безопасности, а также конкретные коммуникационные карты Ethernet и др.

Используют следующие виды ПЛК:

ОВЕН ТРМ232М – предназначен для регулирования температуры в системах отопления и управления насосными группами.

В комплекте с датчиками и исполнительными механизмами ОВЕН ТРМ232М обеспечивает контроль и регулирование температуры и давления, управляет циркуляционными насосами контуров, насосами и контурами подпитки.

ОВЕН ПР200 (см. рисунок) – это свободно программируемое реле. Применяется для решения локальных задач автоматизации: водоподготовка, водоочистка, вентиляция, отопление и др.



Программируемое реле с дисплеем ОВЕН ПР200

Написание алгоритма осуществляется пользователем на языке FBD (МЭК 61131-3) с помощью бесплатной среды программирования OWEN Logic. Подключение к ЭВМ производится посредством стандартного MiniUSB-кабеля (USB – MiniUSB). Для интеграции в SCADA-системы и управления внешними устройствами в прибор может быть установлено до двух интерфейсов RS-485 с поддержкой протоколов Modbus RTU/ASCII.

Для увеличения количества дискретных входов-выходов программируемого реле ОВЕН ПР200 применяют модули расширения ОВЕН ПРМ.

Модульная конструкция дает ПЛК основное преимущество – *расширяемость*. Есть и другие преимущества, такие как стоимость, простота устройства и прочность конструкции. Такие элементы САУ, как реле, периодически нужно осматривать и проводить замену, и здесь появляется еще одно преимущество ПЛК – минимум движущихся механических частей. Существуют возможности интеграции с более сложными системами, например с программируемым компьютерным контроллером.

В результате многие производственные линии, которые уже используют ПЛК, скорее всего будут работать с ними долгие годы. Привычность и простота дают очень хороший импульс, который будет продвигать ПЛК в будущем. Тем не менее в последние

годы виден очень большой прогресс в развитии промышленных компьютеров, и это напрямую связано с развитием систем машинного зрения, более усложнившейся робототехники, а также промышленных интернет вещей IoT.

Программируемый контроллер автоматизации (ПКА). На смену ПЛК приходит новая система – программируемый контроллер автоматизации (англ. PAC – Programmable Automation Controller), способный решить некоторые проблемы ПЛК. Большинство специалистов утверждают, что ПКА – более коммерческое название, но это не совсем так. К сожалению, существуют различия между их определениями, а с технологической точки зрения принципиальные различия между ними найти довольно трудно.

Как правило, ПКА включает в себя функциональные возможности ПЛК. Оба являются цифровыми устройствами, но ПКА предоставляет расширенные возможности программирования и часто имеет более расширенную функциональность, память и периферические способности. ПКА предлагает более сложные архитектурные системы в случае необходимости большей связности вводов-выводов. Кроме того, он имеет обычно встроенные возможности передачи данных из памяти на USB-накопитель, а также часто присутствует возможность прямого взаимодействия с базами данных.

Существуют различные модели устройств ПКА. Они могут предлагать модели, сосредоточенные на системах машинного зрения, или другие, предназначенные для одновременного контроля нескольких процессов. При выборе модели или технологии должны учитывать и требования будущего (модернизация и расширение производства), и стандарты (например, безопасности). Планирование может не только продлить срок службы контроллера путем удовлетворения будущих потребностей, но и заложить фундамент под использования промышленных интернет вещей IoT и децентрализованного управления.

Программируемые логические контроллеры все еще актуальны, однако развитие систем машинного зрения, динамических роботизированных процессов и управления движением, стремление к большей автоматизации производства с использованием IoT требуют от них значительно большей мощности обработки данных или памяти, которые они не в состоянии обеспечить. Децентрализованная технология может помочь расширить устарев-

шую линейку путем предложения продуктов типа SoCs и FPGAs (программируемая логическая интегральная схема – ПЛИС, конфигурация ее может быть загружена после включения питания), которые обрабатывают информацию непосредственно на самом датчике. Это означает, что добавление сложного процесса к существующей линии может не требовать обязательной установки дорогого ПКА, но будет необходима группа интеллектуальных датчиков, способных самостоятельно хранить и обрабатывать данные своих измерений.

Режим реального времени. Многие понимают режим реального времени как выполнение задачи «как можно скорее». Однако это не так. Система реального времени гарантирует, что все входы, выходы и вычислительные процессы будут обрабатываться за какой-то фиксированный промежуток времени, часто упоминаемый в технической литературе как *дедлайн* (англ. *deadline* – предельный срок). В системах жесткого реального времени нарушение дедлайнов приравнивается к отказу системы. В свою очередь мягкая система реального времени допускает небольшое превышение дедлайнов, но только тогда, когда это приводит к допустимому снижению качества работы системы. Например, в видеоконференции небольшое запаздывание звука или видео не приведет к катастрофическим последствиям.

При компиляции программы ПЛК рассчитывает, есть ли необходимые ресурсы для выполнения пользовательских инструкций, после чего переходит к выполнению поставленной задачи в нужный срок.

Мягкая система реального времени (softPLC). Как упоминалось выше, мягкая система реального времени не дает гарантии своевременного выполнения поставленной задачи. Поэтому ее не применяют для систем управления движением. Вместо этого softPLC предпочтительна для подключения связей завод–цех, человеко-машинных интерфейсов, систем диспетчерского контроля и сбора данных (SCADA). Вполне возможно, для некоторых ПКА должна быть softPLC.

Встраиваемый промышленный компьютер (ПК). Он разработан и оптимизирован для одного пользовательского приложения. Все его компоненты, как правило, размещены на одной плате, включая и микроконтроллеры или микропроцессоры, шины ввода-вывода, память и другие пользовательские микросхемы.

Устройство включает в себя даже программное обеспечение или прошивку (прошивка обычно находится в ПЗУ или памяти только для чтения). Встраиваемые ПК (embedded PC) – это пересечение между аппаратным и программным обеспечением, поскольку существует тесная взаимосвязь между этими двумя частями: одна не может работать без другой. Проекты с использованием встраиваемых ПК могут реагировать на потребности жесткого или мягкого режима реального времени.

Путь к смарт-производству. Умное производство глобально развивается благодаря его преимуществам. Обеспечивая большую связь между промышленными мощностями и максимально автоматизируя технологические процессы, смарт-производства предоставляют производителям практически неограниченные возможности для улучшения работы. Например, некоторые производители используют искусственный интеллект для прогнозирования процента отказов оборудования и повышения производительности труда.

Использование удаленных систем мониторинга для слежения за несколькими системами одновременно с одного центра позволяет значительно повысить производительность. Центры обмена данными расширяют возможности локального обмена информацией между отдельными производственными циклами, что позволяет строить более эффективные алгоритмы обработки данных. Применение современных облачных технологий способно изменить бизнес модели и создать новые источники доходов.

Построение инфраструктуры. Принятие ключевых перспективных технологий является неотъемлемой частью смарт-производственного подхода. Внедрение IoT включает в себя использование постоянно растущего числа интеллектуальных устройств для улучшения качества обработки информации, эффективности, безопасности и сохранности данных, а также стратегию использования облачных вычислений, мобильности и анализа данных.

Большинство производителей пока еще не готовы к внедрению смарт-технологий в производство, однако они четко видят возможности их использования. Согласно исследованиям MPI, выделено пять целей, в которых производители могли бы использовать IoT:

- улучшение качества продукции;
- увеличение скорости операций;

- снижение производственных затрат;
- улучшение технической надежности;
- бесперебойная работа.

Достижение этих целей требует построения комплексной архитектуры и стратегии использования смарт-производства. В частности, производители должны сводить свои системы информационных технологий (ИТ) и технологических операций (ТО) в единую унифицированную сетевую инфраструктуру. Кроме того, они должны определить возможности для использования технологий IoT, которые обеспечивают беспрепятственную связь и обмен информацией между людьми, процессами и вещами.

В то же время производители должны гарантировать эффективное управление большими объемами данных, так как это поможет сделать работу систем управления более производительной. Данная концепция включает в себя использование интернет вещей IoT, подключение систем сбора и анализ данных к облачным системам для улучшения работы с большими объемами данных, необходимых для балансировки производственной деятельности. Производители стремятся создать и внедрить связанные системы IoT для создания смарт-производств. Для реализации этого существуют четыре основных направления, рассмотренных ниже.

Повышение производительности и качества. Программы управления качеством и непрерывным улучшением могут создаваться сколь угодно только в случае, если они основываются на информации, ограниченной или недоступной в режиме реального времени. На производстве подавляющее большинство оборудования, ведущего наблюдение и контроль, работает в режиме реального времени. Более того, эти интеллектуальные устройства активно применяют для повышения качества выпускаемой продукции и поиска дефектов в процессе производства.

Более эффективный контроль и прозрачность производственных процессов с помощью встроенного интеллекта также создают новые возможности для повышения производительности. Например, операторы многих заводов и фабрик производят анализ производственных процессов в режиме реального времени. Это помогает выявлять недостатки прямо в процессе производства и оперативно их устранять. Менеджеры и логистические специалисты используют смарт-производственные технологии для предоставления критически важной информации, такой как

прогнозы и графики, заказчикам, а также мониторинг исполнения поставки.

Совершенствование процесса принятия решений. Наиболее оптимальное решение может быть принято только при обладании оперативной информацией. Тем не менее большинство производителей имеют старые системы сбора и обработки информации, которые необходимо обновлять до следующего поколения производительности. Это включает в себя согласование разрозненных источников данных технологических операций с их информационно-технологическими системами, извлечение нужных данных из смарт-технологий производства и преобразование их в полезную информацию.

Производители, которые предприняли шаги по модернизации и вооружились более качественными системами сбора и обработки информации, используют их для оптимизации активов, повышения реакции на быстро изменяющиеся потребности клиентов, усовершенствования рабочих потоков и сокращения расходов. Более того, они приобретают новые стратегические идеи, которые помогают им понять свой бизнес в более глубоких отношениях, в том числе:

- определение эксплуатационных сильных и слабых сторон;
- анализ процессов и инициатив по улучшению планирования;
- разработка и внедрение более совершенных систем производства;
- разработка целевых программ подготовки;
- установка безопасных и надежных решений.

Достижение безопасных, совместимых и надежных решений является постоянной проблемой для любого производителя, и смарт-производства открывают новые возможности для работы с некоторыми из этих извечных проблем. Наиболее очевидные возможности — это замена устаревших и изолированных систем автоматизации, исчерпавших свой жизненный цикл, которые трудносовместимы с другими системами и больше не поддерживаются изготовителями. Тем не менее перед началом модернизации производитель должен определиться с требованиями к новой смарт-системе на основе прошлого опыта в таких сферах, как травмоопасность для рабочего персонала, время простоя оборудования, перерывы в работе.

Исходя из этого, можно расставить приоритеты процессов и оборудования при модернизации. Необходимо рассмотреть вопрос об использовании встроенного интеллекта для сбора дан-

ных в режиме реального времени, в том числе состояния оборудования и отчетности на основе исключений, которые могут быть вписаны в отчет и доставлены в качестве аналитических данных в таких областях, как качество, безопасность, выполнение, использование энергии, а также вопросы времени простоя.

Различные заинтересованные стороны (например, инженеры по качеству, инженеры по охране труда, специалисты по техническому обслуживанию оборудования) могут использовать эти данные для оптимизации производительности машин, производственных процессов, соблюдения основ охраны труда и пр. Производители также должны позаботиться о процессах совместной работы механизмов, где более мощные и производительные машины смогут вести наблюдение и контроль за большим количеством производственных процессов.

Обеспечение безопасности инфраструктуры. Большая доступность информации, а также огромное количество точек подключения вводит большой риск для производственной среды как в виде внутренних, так и в виде внешних угроз. Современные киберпреступники подвергают атакам не только корпоративные сервера, но и некоторые устройства IoT, что является более опасным для промышленных систем, которые имеют более низкую степень защиты.

Ни одна методология или технология безопасности не смогут дать стопроцентной гарантии защищенности от перечисленных угроз. Для этого производители должны применять комплексный, углубленный защитный подход, который устанавливает гарантии безопасности на разных уровнях, для предотвращения угроз на нескольких фронтах.

Надежная и безопасная сетевая инфраструктура должна строиться на стандартном и неизменном Ethernet, который стал более предпочтительным в промышленности для повышения безопасности. Он также должен обеспечивать безопасность управлением и установкой программного обеспечения техническим персоналом, процедурой обновления и исправления, а также включать политику строгой безопасности для управления всем оборудованием.

Контрольные вопросы и задания

1. Укажите основное назначение автоматизации.
2. Опишите назначение современных программируемых логических контроллеров.

3. Приведите примеры современных программируемых логических контроллеров.
4. Какие устройства приходят на смену современным программируемым логическим контроллерам? Охарактеризуйте данные устройства.
5. Что понимают под системой реального времени?
6. Дайте определение встраиваемому промышленному компьютеру.
7. Расскажите о развитии умного производства.
8. Опишите технологию IoT.
9. Какую роль играют программы управления производительностью и качеством в современном производстве?
10. Какие стратегические идеи приобретают производители для модернизации, вооружившись более качественными системами сбора и обработки информации?
11. Какую роль на производстве играет обеспечение безопасности инфраструктуры?

ЛИТЕРАТУРА

- Автоматика** : учеб. / А.В. Шавров [и др.]. М., 1999.
- Александровская, А.Н.** Автоматика : учеб. / А.Н. Александровская. М., 2014.
- Аристов, Е.В.** Основы микропроцессорной и преобразовательной техники : учеб. пособие / Е.В. Аристов. Пермь, 2008.
- Архитектуры** и топологии многопроцессорных вычислительных систем / А.В. Богданов [и др.]. М., 2004.
- Балашов, Е.П.** Микро- и мини-ЭВМ / Е.П. Балашов. СПб., 2004.
- Вавилов, В.Д.** Интегральные датчики : учеб. / В.Д. Вавилов. Н. Новгород, 1999.
- Водовозов, А.М.** Элементы систем автоматки : учеб. пособие / А.М. Водовозов. М., 2006.
- Гордеев, А.С.** Основы автоматки : учеб. пособие / А.С. Гордеев. Мичуринск, 2006.
- Гуров, В.В.** Архитектура микропроцессоров / В.В. Гуров. М., 2010.
- Гусев, В.Г.** Электроника и микропроцессорная техника : учеб. / В.Г. Гусев, Ю.М. Гусев. М., 2013.
- Калиш, Г.Г.** Основы вычислительной техники / Г.Г. Калиш. М., 2000.
- Крупский, В.Н.** Теория алгоритмов : учеб. пособие / В.Н. Крупский. М., 2009.
- Малафеев, С.И.** Основы автоматки и системы автоматического управления : учеб. / С.И. Малафеев, А.А. Малафеева. М., 2010.
- Новиков, Ю.В.** Основы микропроцессорной техники / Ю.В. Новиков, П.К. Скоробогатов. М., 2008.

Селевцов, Л.И. Автоматизация технологических процессов : учеб. / Л.И. Селевцов, А.Л. Селевцов. М., 2014.

Смирнов, Ю.А. Основы микроэлектроники и микропроцессорной техники : учеб. пособие / Ю.А. Смирнов, С.В. Соколов, Е.В. Титов. СПб., 2013.

Столлингс, У. Структурная организация и архитектура компьютерных систем / У. Столлингс. М., 2002.

Титце, У. Полупроводниковая схемотехника : в 2 т. / пер. с нем. : У. Титце, К. Шенк. 12-е изд. М., 2008.

Фомин, Д.В. Основы компьютерной электроники : учеб. пособие / Д.В. Фомин. Саратов, 2017.

Шишмарев, В.Ю. Автоматика : учеб. / В.Ю. Шишмарев. М., 2017.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
1. СИСТЕМЫ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ	6
1.1. Общие сведения о системе	6
1.2. Основные понятия автоматизации.....	8
1.3. Назначение систем автоматического управления.....	11
1.4. Классификация элементов систем автоматического управления	13
1.5. Принципы регулирования.....	17
2. ДАТЧИКИ.....	19
2.1. Общие сведения.....	19
2.2. Потенциометрические датчики	20
2.3. Тензометрические датчики.....	22
2.4. Индуктивные датчики.....	23
2.5. Емкостные датчики.....	25
2.6. Пьезоэлектрические датчики.....	26
2.7. Фотоэлектрические датчики	28
2.8. Датчики температуры	32
3. УСИЛИТЕЛЬНЫЕ УСТРОЙСТВА	37
3.1. Общие сведения.....	37

3.2. Транзисторные усилители	42
3.3. Усилители с обратными связями.....	47
3.4. Операционные усилители.....	50
4. ТИПОВЫЕ ЗВЕНЬЯ СИСТЕМ АВТОМАТИЧЕСКОГО РЕГУЛИРОВАНИЯ	55
4.1. Режимы работы объекта. Возмущающие воздействия	55
4.2. Аперриодическое (инерционное, статическое) звено	56
4.3. Астатическое (интегрирующее) звено.....	59
4.4. Колебательное звено (аперриодическое звено второго порядка)	61
4.5. Пропорциональное (усилительное, безынерционное) звено	63
4.6. Дифференцирующее звено	64
4.7. Запаздывающее звено	66
4.8. Показатели качества процесса регулирования.....	67
5. ДИСКРЕТНЫЕ АВТОМАТЫ	71
5.1. Понятие дискретного автомата	71
5.2. Основы алгебры логики.....	73
5.3. Основные функции и законы алгебры логики	74
5.4. Базисы в алгебре логики	77
5.5. Схемотехническая реализация основных логических функций.....	78
5.6. Резисторно-транзисторная логика.....	79
5.7. Диодно-транзисторная логика.....	80
5.8. Медленнодействующая помехоустойчивая логика.....	80
5.9. Транзисторно-транзисторная логика.....	81
5.10. Логические схемы на комплементарных МОП- транзисторах	86
5.11. n-канальная МОП-логика.....	89
6. ЭЛЕМЕНТЫ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ	93
6.1. Триггеры	93

6.2. Регистры	98
6.3. Счетчики импульсов.....	101
6.4. Дешифраторы.....	103
6.5. Элементы памяти	105
7. АРИФМЕТИЧЕСКИЕ И ЛОГИЧЕСКИЕ ОСНОВЫ ЭВМ.....	108
7.1. Системы счисления, их назначение и виды	108
7.2. Кодирование информации	113
7.3. Арифметические операции над двоичными числами	120
7.4. Физические формы представления информации.....	126
8. СТРУКТУРА ЭЛЕКТРОННО-ВЫЧИСЛИТЕЛЬНЫХ МАШИН, МИКРОЭВМ.....	129
8.1. Общие сведения.....	129
8.2. Устройство процессора	130
8.3. Устройство управления.....	134
8.4. Запоминающие устройства микропроцессорных систем ...	137
8.5. Периферийное оборудование микроЭВМ	143
8.6. Устройства ввода данных.....	145
8.7. Устройства вывода данных	153
8.8. Устройства хранения данных и обмена данными	159
9. АРХИТЕКТУРА МИКРОПРОЦЕССОРОВ.....	163
9.1. Общие сведения.....	163
9.2. Классификации микропроцессоров	163
9.3. Архитектура микропроцессора	167
9.4. Основные характеристики микропроцессоров.....	174
10. ОРГАНИЗАЦИЯ ВВОДА-ВЫВОДА В МИКРОПРОЦЕССОРНОЙ СИСТЕМЕ.....	176
10.1. Общие сведения	176
10.2. Интерфейсы микропроцессорных систем.....	180
10.3. Интерфейс RS-232	182

10.4. Последовательный интерфейс USB	188
10.5. Интерфейс I ² C	191
10.6. Интерфейс SPI	195
10.7. Интерфейс PPI	200
10.8. Свободно программируемый порт (Freeport)	200
10.9. Порт ведущего-ведомого устройства MODBUS RTU.....	201
11. ВЫЧИСЛИТЕЛЬНЫЕ И МИКРОПРОЦЕССОРНЫЕ УСТРОЙСТВА.....	203
11.1. Общие сведения	203
11.2. Построение и применение вычислительных средств в системах управления.....	205
11.3. Контроль работы вычислительных средств	212
12. ОСНОВЫ ПРОГРАММИРОВАНИЯ.....	216
12.1. Языки программирования и их классификация.....	216
12.2. Алгоритм программы и его свойства	224
12.3. Характеристики программного продукта	229
12.4. Основные характеристики программ	230
13. СОВРЕМЕННЫЕ МИКРОПРОЦЕССОРНЫЕ СИСТЕМЫ, ТЕНДЕНЦИИ ИХ РАЗВИТИЯ.....	233
ЛИТЕРАТУРА.....	242

Учебное издание

Кушнер Дмитрий Александрович,
Дробов Андрей Владимирович,
Петроченко Юрий Леонидович

ОСНОВЫ АВТОМАТИКИ И МИКРОПРОЦЕССОРНОЙ ТЕХНИКИ

Учебное пособие

Редактор *Л.В. Рутковская*
Технический редактор *Е.В. Потапейко*
Корректор *И.В. Счеснюк*
Дизайн обложки *Е.Н. Самусевич*

Подписано в печать 15.02.2019. Формат 60×84/16.
Бумага офсетная. Ризография.
Усл. печ. л. 14,46. Уч.-изд. л. 10,64.
Тираж 700 экз. Заказ 31.

Издатель и полиграфическое исполнение:
Республиканский институт профессионального образования.
Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий № 1/245 от 27.03.2014.
Ул. К. Либкнехта, 32, 220004, Минск. Тел.: 226 41 00, 200 43 88.

Отпечатано в Республиканском институте
профессионального образования. Тел. 200 69 45.