

**САПР В АВТОМОБИЛЕ- И ТРАКТОРОСТРОЕНИИ**

Высшее профессиональное образование

**Ю. В. ДЕМЕНТЬЕВ  
Ю. С. ЩЕТИНИН**

# **САПР В АВТОМОБИЛЕ- И ТРАКТОРОСТРОЕНИИ**

Учебник

Учебник



Транспорт

**ACADEMIA**



Ю. В. ДЕМЕНТЬЕВ, Ю. С. ЩЕТИНИН

# САПР В АВТОМОБИЛЕ- И ТРАКТОРОСТРОЕНИИ

Учебник

Под общ. редакцией д-ра техн. наук, проф. В. М. Шарипова

*Допущено*

*Министерством образования Российской Федерации в качестве учебника для студентов высших учебных заведений, обучающихся по специальности «Автомобиле- и тракторостроение» направления подготовки дипломированных специалистов «Транспортные машины и транспортно-технологические комплексы»*

УДК 629.113/.115:656.137(075.8)

ББК 39.33:39.34я73

Д302

Рецензенты:

первый заместитель генерального директора ФГУП «НАМИ»  
д-р техн. наук, проф. *О. И. Гируцкий*;  
зав. кафедрой «Автомобили и двигатели» Московского индустриального  
университета, канд. техн. наук, проф. *Р. П. Кушвид*

**Дементьев Ю. В.**

Д302 САПР в автомобиле- и тракторостроении: Учебник для студ. высш. учеб. заведений / Ю. В. Дементьев, Ю. С. Щетинин; Под общ. ред. В. М. Шарипова. — М.: Издательский центр «Академия», 2004. — 224 с.

ISBN 5-7695-1758-1

Рассмотрены методы и средства автоматизированного проектирования автомобилей, тракторов и составляющих их элементов, принципы построения систем автоматизированного проектирования и их классификация по структуре и функциональным возможностям. Дано описание средств автоматизированного проектирования, используемых в автомобиле- и тракторостроении.

Для студентов вузов и специалистов.

УДК 629.113/.115:656.137(075.8)

ББК 39.33:39.34я73

ISBN 5-7695-1758-1

© Дементьев Ю. В., Щетинин Ю. С., 2004

© Издательский центр «Академия», 2004

## ПРЕДИСЛОВИЕ

Целью каждой проектно-конструкторской разработки является создание и выпуск изделий на уровне лучших мировых образцов. Достичь этой цели можно лишь путем применения комплекса наиболее эффективных технических решений. Для этого требуется синтезировать и проанализировать очень много вариантов решений, что по ряду причин затруднительно или невозможно без применения вычислительной техники.

Идея использования электронной вычислительной техники при решении задач проектирования возникла практически сразу после появления первых ЭВМ. Однако только в 1970-е гг. развитие ЭВМ и периферийных устройств, позволяющих наиболее полно использовать технические возможности ЭВМ, привели к появлению первых систем автоматизированного проектирования (САПР).

В наши дни наблюдается быстрое развитие САПР в таких отраслях, как авиастроение, автомобиле- и тракторостроение, тяжелое машиностроение, архитектура, строительство, нефтегазовая промышленность, картография, геоинформационные системы, а также в производстве товаров народного потребления, например бытовой электротехники. САПР в машиностроении используется для проведения конструкторских и технологических работ, в том числе работ по технологической подготовке производства. С помощью САПР выполняется разработка чертежей, производится трехмерное моделирование изделия и процесса сборки, проектируется вспомогательная оснастка, например штампы и пресс-формы, составляется технологическая документация и управляющие программы для станков с числовым программным управлением, ведется архив. Современные САПР применяются для сквозного автоматизированного проектирования, технологической подготовки, анализа и изготовления изделий в машиностроении, для электронного управления технической документацией.

Применение САПР в автотракторной отрасли позволяет существенно снизить затраты времени и средств на создание новых и модернизацию существующих моделей автомобилей и тракторов. В связи с этим возникает необходимость подготовки специалистов — пользователей САПР для предприятий отрасли.

Основой данного учебника послужили лекции, читаемые авторами в Московском государственном техническом университете «МАМИ» студентам, обучающимся по направлению подготовки бакалавра 551400 «Наземные транспортные системы» и по направлению подготовки дипломированного специалиста 653200 «Транспортные машины и транспортно-технологические комплексы» (специальность 150100 «Автомобиле- и тракторостроение»).

Главы 1, 3—6 написаны Ю. В. Дементьевым, глава 2 — Ю. С. Щетининым.

## СОКРАЩЕНИЯ

- АРМ — автоматизированное рабочее место;  
АСТПП — автоматизированная система технологической подготовки производства;  
БД — база данных;  
БИС — большая интегральная схема;  
БПУ — безударное печатающее устройство;  
ВЗУ — внешнее запоминающее устройство;  
ГМД — гибкий магнитный диск;  
ДВК — диалоговый вычислительный комплекс;  
ДУЧП — дифференциальное уравнение в частных производных;  
ЕС — единая система;  
ЖК — жидкокристаллический;  
ЛАУ — линейное алгебраическое уравнение;  
ЛВС — локальная вычислительная сеть;  
МД — магнитный диск;  
ММ — математическая модель;  
НМД — накопитель на магнитном диске;  
ОА — область адекватности;  
ОДУ — обыкновенное дифференциальное уравнение;  
ОЗУ — оперативное запоминающее устройство;  
ПК — персональный компьютер;  
ПП — пакет программ;  
ПУ — печатающее устройство;  
ПЭВМ — персональная электронно-вычислительная машина;  
САПР — система автоматизированного проектирования;  
СУБД — система управления базами данных;  
ТГЭ — типовой графический элемент;  
ТЗ — техническое задание;  
УВГИ — устройство ввода графической информации;  
ЭВМ — электронно-вычислительная машина;  
ЭЛТ — электронно-лучевая трубка.

## ОБЩИЕ СВЕДЕНИЯ О СИСТЕМАХ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

### 1.1. Применение ЭВМ для автоматизации проектирования и технологической подготовки производства автомобилей и тракторов

*Проектирование технического объекта* связано с созданием, преобразованиями и представлением в принятой форме образа этого объекта. Образ объекта или его составных частей может создаваться в воображении человека в результате творческого процесса или генерироваться по некоторым алгоритмам в процессе взаимодействия человека и ЭВМ. В любом случае проектирование начинается при наличии задания на проектирование, которое отражает потребности общества в получении некоторого технического изделия; это задание представляется в виде тех или иных документов и является *исходным (первичным) описанием объекта*. Результатом проектирования, как правило, служит полный комплект документации, содержащий достаточные сведения для изготовления объекта в заданных условиях; эта документация представляет собой *окончательное описание объекта*.

*Проектирование* — процесс, заключающийся в преобразовании исходного описания объекта в окончательное описание на основе выполнения комплекса работ исследовательского, расчетного и конструкторского характера.

Преобразование исходного описания в окончательное порождает *промежуточные описания*, которые являются предметом рассмотрения с целью определения окончания проектирования или выбора путей его продолжения. Такие описания называют *проектными решениями*.

Проектирование, при котором все или часть проектных решений получают путем взаимодействия человека и ЭВМ, называют *автоматизированным*, а проектирование, при котором ЭВМ не используются, — *неавтоматизированным*.

Возможности проектирования сложных объектов обусловлены использованием ряда принципов, основными из которых являются декомпозиция и иерархичность описаний объектов, многоэтапность и итерационность проектирования, типизация и унификация проектных решений и средств проектирования.

Широкое внедрение вычислительной техники в условиях научно-технического прогресса обеспечивает рост производительности труда в различных областях производства. Главное внима-

ние при этом обращается на те из них, где рост производительности труда до применения ЭВМ проходил крайне медленно; это, в первую очередь, области, связанные с приложением умственного труда человека, т.е. управление производством, проектирование и исследование объектов и процессов. Если производительность труда в сфере производства с начала XX в. возросла в сотни раз, то в области проектирования только в 1,5—2 раза; это обуславливает большие сроки проектирования новых объектов, что не отвечает потребностям развития экономики.

Очевидность того факта, что развитие новой техники в современных условиях замедляется не столько отсутствием научных достижений и инженерных идей, сколько сроками и не всегда удовлетворительным качеством их реализации в процессе конструкторско-технологической разработки, ни у кого не вызывает сомнения. Одним из направлений решения этой проблемы является создание и развитие САПР.

Качество проектирования в значительной степени определяет темпы технического прогресса.

Прогресс производства в современных условиях связывают с достижениями в области автоматизации производства. Поскольку проектирование и разработка технологии являются ступенью производства (логическим уровнем), прогресс на этой ступени также должен определяться автоматизацией.

Таким образом, автоматизация проектирования — закономерность, соответствующая естественному пути совершенствования производства вообще. Дело лишь в том, что логический уровень подготовки производства (проектирование) связан непосредственно с интеллектуальной деятельностью человека, которую пока еще формализовать сложно.

При неавтоматизированном проектировании результаты во многом определяются инженерной подготовкой конструкторов, их производственным опытом, профессиональной интуицией и другими факторами. Автоматизированное проектирование позволяет значительно сократить субъективизм при принятии решений, повысить точность расчетов, выбрать наилучшие варианты для реализации на основе строгого математического анализа всех или большинства вариантов проекта с оценкой технических, технологических и экономических характеристик производства и эксплуатации проектируемого объекта, значительно повысить качество конструкторской документации, существенно сократить сроки проектирования и передачи конструкторской документации в производство, эффективнее использовать технологическое оборудование с программным управлением. Автоматизация проектирования способствует более полному использованию унифицированных изделий в качестве стандартных компонентов проектируемого объекта.



Характер применения ЭВМ при проектировании различных объектов с течением времени претерпевает значительные изменения. С появлением вычислительной техники был сделан переход от традиционных «ручных» методов проектирования к реализации отдельных задач проектирования на ЭВМ. Такой подход, характеризовавший использование ЭВМ на первом этапе, носит название «позадачного» и заключается в том, что каждая вновь возникающая задача решается с помощью автономно создаваемой программы, которая функционирует независимо от других программ данной предметной области. Коренной недостаток такого подхода заключается в том, что подобные программы строятся по принципу «натурального хозяйства», когда для решения отдельной задачи требуется полная подготовка вспомогательных средств (технических, информационных, программных и др.). Поскольку проектирование объекта, как правило, предполагает и его оптимизацию, машинная программа в этом случае представляет собой «симбиоз» модели проектируемого объекта и некоторого алгоритма оптимизации. Естественно, что в этом случае ни модель, ни алгоритм оптимизации не могут использоваться для других целей (например, чтобы провести оптимизацию модели трансмиссии с помощью другого алгоритма, необходимо разработать новую программу).

Однако применение ЭВМ на этом этапе, несмотря на отмеченный недостаток, позволило перейти от упрощенных методов расчета и проектирования с субъективным выбором «лучшего» варианта к научно обоснованным методам, обеспечивающим рассмотрение всего многообразия технически приемлемых вариантов с учетом заданных условий и ограничений и объективный выбор среди них оптимального.

По мере совершенствования ЭВМ акцент в использовании вычислительной техники постепенно смещался от факта применения ЭВМ в качестве электронного арифмометра в сторону более эффективного и производительного использования ЭВМ в системах с режимом «ЭВМ — человек».

С появлением вычислительной техники новых поколений и совершенствованием методов ее использования наметился новый, системный подход к организации процесса проектирования на ЭВМ, заключающийся в создании крупных программных комплексов в виде пакетов программ (ПП) и САПР, ориентированных на определенный класс задач. Такие комплексы строятся по модульному принципу с универсальными информационными и управляющими связями между модулями. При решении задач данного класса используются единые информационные массивы, организованные в банки данных.

Объединение нескольких ПП в единую систему, предназначенную для реализации вполне определенных функций, позволя-

ет говорить о новом, более высоком уровне в иерархии программных комплексов, т.е. САПР. При этом качественные изменения претерпевают и организация информационного, технического и других видов обеспечения, и, что особенно важно, условия обмена информацией между человеком и ЭВМ. Как правило, эти изменения направлены на повышение гибкости и универсальности системы, улучшение характеристик взаимодействия проектировщика с ЭВМ, повышение качества получаемого результата и снижение времени его получения.

В начале 1950-х гг. существовало мнение, что несколько крупных (по тем временам) ЭВМ могут удовлетворять все запросы на вычисления в любой промышленно развитой стране. Однако жизнь очень скоро опровергла эти предположения. Количественный и качественный рост парка ЭВМ позволил обратиться к решению таких задач, о которых раньше даже не помышляли. Так, например, различные динамические и колебательные процессы требуют для своего описания десятков дифференциальных уравнений. При решении этих задач вручную приходилось сильно упрощать расчетные схемы, что приводило к значительным количественным, а иной раз и к качественным ошибкам. ЭВМ позволяет использовать очень сложные математические модели, значительно более точно описывающие реальные процессы. Более того, открылась возможность для широкого применения расчетных методов, разработанных еще до создания ЭВМ, но неудобных либо неприемлемых при ручных расчетах.

Применение ЭВМ для решения инженерных задач началось сразу же после появления их первых образцов. Однако это применение требовало от пользователя трудоемкой подготовки задач к решению, заключающейся в математической формулировке задачи, выборе численного метода, разработке алгоритма и его записи на одном из языков программирования. Автоматизированное проектирование отличается от подобного использования ЭВМ прежде всего тем, что почти все из перечисленных операций автоматизированы и выполняются на ЭВМ с помощью заранее разработанного программного обеспечения, рассчитанного на многократное применение при решении определенного класса проектных задач. От пользователя требуется лишь описать исходные данные задачи на проблемно-ориентированном языке и быть готовым к оценке результатов и принятию решений по полученным от ЭВМ сведениям.

В каждой новой сфере применения ЭВМ обычно довольно скоро начинают сказываться ограничения в их характеристиках (прежде всего в скорости вычислений и объеме памяти), и, таким образом, стимулируется следующий виток в развитии вычислительной техники, который в свою очередь делает возможным создание и использование программного обеспечения более высокого

уровня, а также применение ЭВМ в областях человеческой деятельности, появление и существование которых обусловлено вычислительной техникой. Так, в условиях стремительного развития ЭВМ и быстрого расширения сферы их приложения в начале 1960-х гг. родилась новая область их применения — интерактивная машинная графика, предметом которой стало представление и преобразование графической информации.

Говоря об интерактивной машинной графике, следует отметить ее влияние на организацию самих вычислений. Во-первых, появилась возможность наглядного представления результатов расчета, объем которых в численном виде огромен. Так, при решении газодинамических задач получаемый ответ может состоять из нескольких миллионов чисел, которые человеку осмыслить трудно. Методами же машинной графики всю эту информацию можно уложить в пятиминутный фильм, хорошо воспринимаемый человеком. Во-вторых, ЭВМ теперь позволяют в буквальном смысле рассматривать работу графической модели изучаемого объекта, т. е. решать задачи, в которых обработка графической информации занимает значительное место. Например, можно видеть поведение автомобиля в аварийных ситуациях или колебание остова автомобиля или трактора при движении по неровностям пути. Без графического вывода результат решения подобных задач воспринимается трудно, поэтому многие системы автоматизации научных исследований и САПР немислимы без использования средств машинной графики.

#### Современные САПР:

характеризуются наличием банка данных с справочными характеристиками материалов, с прежними техническими решениями, чертежами конструкций, патентами, стандартами и другой информацией, необходимой проектировщику;

обеспечивают возможность корректировки баз данных в процессе проектирования;

могут осуществлять моделирование (физическое, математическое, графическое) как отдельных элементов, так и всей конструкции в целом;

имеют возможность развития путем присоединения нового программного обеспечения в пакеты имеющихся программ;

содержат развитые графические подсистемы, которые могут совмещать различные виды и проекции изделий, преобразовывать масштабы, осуществлять аффинные преобразования, заменять отдельные элементы конструкции другими;

могут обеспечивать одновременную работу нескольких проектировщиков.

#### Применение САПР позволяет:

ускорить процесс проектирования и улучшить его качество за счет применения современных расчетно-оптимизационных методов;

в ряде случаев снизить затраты трудовых и материальных ресурсов на проектирование;

избавить проектировщика от рутинной работы, многократно увеличивая производительность и качество его труда и освобождая его время для творческой работы.

## **1.2. САПР и роль проектировщика в автоматизированном проектировании**

САПР — комплекс технических средств и программного обеспечения, предназначенный для автоматизированного проектирования технических объектов с участием человека.

Проектировщик организует процесс проектирования с использованием данной САПР, направляет ход решения задачи в зависимости от полученных промежуточных результатов, анализирует и оценивает полученные результаты, принимает решение по проекту и несет ответственность за его качество.

В ГОСТ 23501.0—79 САПР определена как организационно-техническая система, состоящая из комплекса средств автоматизации проектирования, взаимодействующего с подразделениями проектной организации, и выполняющая автоматизированное проектирование.

САПР создается как человекомашина система, в которой для автоматизации интеллектуально-формальных процессов деятельности специалистов используются возможности вычислительной техники. Процесс проектирования в системе организуется таким образом, чтобы повысить отдачу интеллектуального труда человека, стимулировать его творческую деятельность.

При этом возникают некоторые социальные эффекты. Использование САПР превращает инженера-проектировщика, инструментами которого были счетная линейка, арифмометр и кульман, в пользователя САПР, использующего ЭВМ с набором периферийных устройств. Содержание работы специалистов радикально меняется, хотя и может иметь различный характер. Например, если высококвалифицированная работа, требующая знаний и точности исполнения, но небольшого воображения, передается машине и выполняется пакетом программ системы, то инженеры, постоянно занимающиеся такой работой, становятся операторами, вводящими данные в машину и получающими для анализа результаты.

В развитых диалоговых системах, где проектные решения принимаются человеком, а промежуточная обработка данных выполняется машиной, ритм работы радикально меняется. Здесь проектировщик нередко испытывает значительные перегрузки, так как темп работы задается ЭВМ.

Таким образом, проектировщики достаточно высокого уровня с помощью САПР могут существенно повысить производительность труда и качество продукта. Однако их работа при этом становится более напряженной, поскольку они лишаются периодов восстановления, которые до внедрения САПР определялись выполнением рутинной работы. Так, по некоторым данным, доля творческой деятельности в общем бюджете времени проектировщика, не использующего САПР, составляет всего 15 %, с трудом формализуемая рутинная работа составляет 42 %, формализуемая рутинная работа — 35 %, остальное — вспомогательные и подготовительные работы. Следовательно, с внедрением САПР потребность в проектировщиках низкой квалификации, круг обязанностей которых ограничивался именно рутинной работой, будет уменьшаться.

Что же определяет специфику труда проектировщика в САПР?

Задачи проектирования существенно отличаются от чисто вычислительных задач, хотя последние являются составной частью процесса проектирования. Отличия заключаются прежде всего в том, что в большинстве случаев оказывается весьма затруднительным дать формальную постановку задачи и поэтому человек должен сам доопределять эту задачу на основе своего опыта, интуиции и т. д. Кроме того, если в традиционных, сложившихся сферах применения ЭВМ между машиной и специалистом, использующим ее, находится еще, как правило, программист-оператор, то в автоматизированных системах проектирования это исключено по двум причинам. Во-первых, из-за массового характера проектной деятельности, в которой одновременно участвуют большие коллективы специалистов, и, во-вторых, проблема коммуникации в системе «человек — человек — ЭВМ» оказывается более сложной, чем в системе «человек — ЭВМ».

Все это потребовало разработки нового подхода к проблеме организации интерфейса человека в САПР, определяющей качественно новые условия деятельности человека в режиме автоматизированного проектирования. Решение данной проблемы имеет два аспекта: совершенствование связей между проектировщиками в режиме автоматизированного проектирования и между человеком и машиной.

Первый аспект обусловлен необходимостью работы над сложным проектом коллектива проектировщиков примерно с тем же разделением функций по конструктивным элементам и уровням иерархии проекта, какое имеет место в обычном неавтоматизированном проектировании. Различие состоит в том, что конструктор работает за своим дисплеем без помощников и вспомогательного персонала. Одновременная работа с проектом нескольких проектировщиков значительно повышает требования к программному обеспечению САПР, которая в этом случае должна иметь специ-



альную подсистему, выполняющую функции координатора в коллективном процессе принятия решений.

Что касается второго аспекта, а именно связей между человеком и машиной, то для ее реализации в первую очередь необходимо качественное периферийное оборудование, средства отображения и приема информации, дистанционная связь терминалов с центральным процессором. Кроме того, проектировщику должны быть созданы комфортные условия для работы с системой. От этих условий во многом зависит устранение социальных и психологических факторов, препятствующих внедрению САПР. Уровень комфортного взаимодействия проектировщика с системой характеризуется информационной обусловленностью процесса принятия решения.

Обеспечение комфортных условий проектировщику при работе с САПР осуществляется за счет настройки и модификации средств отображения и приема информации и, главным образом, посредством персонифицированного информационного взаимодействия с проектировщиком; это предполагает изменяющуюся интенсивность информационного потока, направленного к человеку в зависимости от квалификации проектировщика, его психофизиологического состояния и результатов текущей деятельности. Такое взаимодействие должно быть организовано так, чтобы каждый специалист работал близко к своему пределу компетентности и текущему показателю утомленности, но не выше их.

Другим аспектом взаимодействия проектировщика с машиной является распределение функций между человеком и ЭВМ, которое, в идеале, должно осуществляться динамически в процессе взаимодействия на основе некоторых критериев эффективности (время решения, стоимость решения и т.д.). В настоящее время распределение, как правило, определяется заранее на основе априорных предположений типа, например, такого, что человек располагает преимущественными по отношению к ЭВМ знаниями или ЭВМ быстрее выполняет вычислительные операции и т.п.

Что касается форм диалога, то самой идеальной для человека явилась бы форма естественного языка, т.е. возможность беседовать с машиной как с человеком. Большую трудность в реализации этой формы представляет весьма сложная задача формализации входной информации на естественном языке, поэтому при современном уровне развития вычислительной техники диалог на основе естественного языка оказывается нереальным. Следовательно, приходится накладывать ограничения на возможные способы общения проектировщика с машиной, что приводит к созданию специальных языков общения и реализации с их помощью различных форм диалога с ЭВМ, эти формы зависят от числа и вида накладываемых ограничений.

Например, диалог, инициируемый ЭВМ, или, как его еще называют, *выбор меню* имеет место, когда проектировщик выби-

рает один из предлагаемых системой вариантов. Вся совокупность вариантов ответа заложена в систему заранее, и проектировщик лишь указывает конкретный вариант, отвечая машине ДА/НЕТ; этот простейший диалог применяется, как правило, в наиболее простых САПР и дает возможность работы с ней практически без специальных знаний в области вычислительной техники.

Следующей по сложности формой диалога является так называемый *ординарный диалог*, который сейчас распространен довольно широко. Если в предыдущем случае инициатором диалога выступала лишь система, то здесь проектировщик, кроме ответов на запросы системы, получает возможность указывать режим работы, формы ввода-вывода данных, останавливать и запускать вновь систему по своей инициативе, а также задавать численные данные, необходимые для ее работы. Возможность общаться с системой проектировщик получает лишь в специально предусмотренных в программах местах (точках диалога), причем заранее оговариваются возможные ответы пользователя на запрос системы.

Характерной чертой этих двух форм диалога является отсутствие специального математического обеспечения.

Более совершенной формой диалога является диалог на основе *директивного языка*. Запросы и ответы проектировщика формируются в виде директив, имеющих строго определенную структуру. Директива может выполнять самые разнообразные функции. Запросы системы поступают как в виде развернутых сообщений, так и в виде коротких кодов, идентифицируемых с рядом вероятных состояний системы (требованиями на ввод данных, подключением определенных устройств, указанием на ошибки проектировщика или сбой системы).

Данная форма диалога осуществляется в режиме с разделением времени и требует наличия специального математического обеспечения, состоящего из монитора ввода-вывода, интерпретатора директив языка и архива. Функция монитора заключается в иницировании интерпретатора, который обеспечивает расшифровку и выполнение директивы пользователя. Структура подобной диалоговой системы сложна, но и довольно совершенна. Директивный язык представляет возможность гибкого многовариантного использования программных модулей, включенных разработчиком в систему.

Наконец, последней формой диалога является диалог на основе *процедурно-ориентированных языков* типа BASIC, Pascal, C, C++ и др. Процесс общения пользователя с системой сводится в этом случае к написанию программы на соответствующем расширенном варианте процедурно-ориентированного языка. Достоинством такой формы общения является предоставление проектировщику возможности изменения созданных разработчиком программ и включения в систему новых программ самостоятельной разработ-

ки. Однако к пользователю предъявляются в этом случае наиболее высокие требования — он должен владеть программированием на алгоритмическом языке.

Таким образом, эффективность диалогового взаимодействия в САПР во многом определяется квалификацией проектировщика — пользователя системы.

### 1.3. Структурная схема и классификация САПР

**Структурная схема САПР.** Укрупненно структурную схему САПР можно представить в виде функциональной и обеспечивающей частей (рис. 1.1).

Функциональная часть САПР на схеме представлена в виде набора подсистем, удовлетворяющих поставленным целям проектирования: технологической подготовки производства, моделирования, информационного поиска, инженерных расчетов, управления САПР, испытаний, изготовления, машинной графики. Подсистемы являются основными структурными звеньями САПР



Рис. 1.1. Структурная схема САПР

и различаются по назначению и по отношению к объекту проектирования.

Каждая из составляющих САПР подсистем может быть определена как комплекс программных средств, предназначенных для выполнения определенного этапа процесса проектирования. Необходимо отметить, что программные компоненты неразрывно взаимосвязаны с техническими средствами данной САПР.

Существующий отечественный и зарубежный опыт в области автоматизации проектирования свидетельствует о том, что разработка, внедрение и эффективное использование программных комплексов, предназначенных для автоматизации процесса проектирования и реализуемых на базе современных ЭВМ, требуют комплексного решения широкого спектра проблем: организационных, технических, математических, программных, лингвистических, информационных и др. Решение этих проблем базируется на соответствующих видах обеспечения, которые будут рассмотрены ниже.

Сложность разработок больших комплексов взаимосвязанных программ заключается в том, что эффективность решения каждой конкретной проблемы, как правило, определяется на завершающем этапе работы, когда вся или большая часть системы начинает функционировать; это предопределяет сложность создания высокоэффективных программных комплексов при первоначальной разработке. Система становится действенной в ходе сравнительно длительного процесса создания, испытаний, совершенствования и доводки.

*Подсистема информационного поиска* — это комплекс языково-алгоритмических средств, предназначенный для хранения, поиска в каком-либо множестве элементов (документов, стандартов, нормалей, чертежей выполненных конструкций, патентов, характеристик материалов и т. п.) и представления информации, отвечающей на запрос, предъявленный этой подсистеме.

*Подсистема инженерных расчетов* наряду с *подсистемой машинной графики* обычно встречается уже на начальном этапе создания САПР и представляет собой совокупность программных средств, предназначенных для выполнения различных расчетов (геометрических, прочностных и т. п.) в режиме диалога «человек — машина».

Работа развитой подсистемы инженерных расчетов тесно связана с использованием различного рода математических моделей проектируемых объектов или процессов, для автоматизированного получения которых предназначена *подсистема моделирования*.

Большинство современных САПР, помимо вычислительных, обладает широким спектром возможностей для ввода, обработки, хранения и вывода графической информации, реализуемых программными средствами подсистемы машинной графики.

*Подсистема испытаний* представляет собой комплекс программных средств, предназначенных для создания программ управле-

ния испытательным оборудованием, обработки результатов испытаний, проведения «численного эксперимента» с использованием математических моделей объекта проектирования и процесса его нагружения. Численный эксперимент очень важен в процессе проектирования, поскольку позволяет определить свойства проектируемого объекта без изготовления опытных образцов и вовремя отказаться от бесперспективных вариантов, что позволяет значительно уменьшить затраты времени и материальных средств на создание объекта.

*Подсистема изготовления* может быть предназначена, например, для подготовки программ для станков и автоматических линий с числовым программным управлением.

*Подсистема технологической подготовки производства*, как правило, выходит за рамки САПР и представляет собой самостоятельную систему (АСТПП — автоматизированная система технологической подготовки производства). Тем не менее ее наличие в САПР весьма желательно, хотя бы в упрощенном виде.

*Подсистема управления* предназначена для увязывания работ других подсистем на различных этапах процесса проектирования и выполнения функций координатора в коллективном процессе принятия решений.

Не обязательно в каждой САПР должен быть представлен весь набор функциональных подсистем — они могут сочетаться произвольно в зависимости от задач, стоящих перед системой. Необходимо отметить, что все функциональные подсистемы тесно взаимосвязаны, поэтому зачастую невозможно провести между ними четкие границы. Так, например, подсистема машинной графики может выдавать результаты в виде программы для оборудования с числовым программным управлением, что непосредственно связывает ее с подсистемами технологической подготовки производства и изготовления. Та же подсистема машинной графики может позволить получать на выходе вместо чертежей раскрой листового материала, что роднит ее с подсистемой изготовления.

Некоторые подсистемы в зависимости от степени их развития или назначения, как было отмечено в примере с АСТПП, могут существовать как самостоятельные системы (например, информационно-поисковые системы, графические и др.).

Обеспечивающая часть представляет собой технические средства и документацию на машинных и других информационных носителях, необходимые в процессе проектирования.

В отличие от функциональной обеспечивающая часть должна входить в систему всеми своими компонентами даже в случае различной степени совершенства каждой из них. При отсутствии любой составляющей обеспечивающей части нельзя говорить о существовании САПР в целом, поскольку все компоненты тесно взаимосвязаны.



*Техническое обеспечение САПР* представляет собой совокупность взаимосвязанных и взаимодействующих технических средств, включающих ЭВМ и работающие под ее управлением внешние устройства, предназначенных для выполнения автоматизированного проектирования. Техническое обеспечение делится на группы средств программной обработки данных (процессоры и запоминающие устройства, в которых реализуются преобразования данных и программное управление вычислениями), подготовки, ввода и отображения данных (служащие для общения человека с ЭВМ), вывода, хранения и передачи данных (различные запоминающие, печатающие и другие графические устройства и средства связи с удаленными терминалами).

*Математическое обеспечение САПР* объединяет описание математических моделей проектируемых объектов и математических методов, реализованных в данной САПР. Элементы математического обеспечения чрезвычайно разнообразны. Среди них имеются инвариантные элементы, широко применяемые в САПР; к ним относятся принципы построения функциональных моделей, методы численного решения алгебраических и дифференциальных уравнений, постановки и решения задач на определение экстремума. Специфика предметных областей проявляется прежде всего в математических моделях проектируемых объектов.

*Программное обеспечение САПР* представляет собой описание алгоритмов проектирования, использованных в данной САПР, а также документы с исходными текстами программ, программы на машинных носителях и эксплуатационные документы.

*Информационное обеспечение САПР* объединяет различные данные, необходимые для выполнения автоматизированного проектирования, которые могут быть представлены в виде документов на различных носителях, содержащих сведения справочного характера о материалах, комплектующих изделиях, типовых проектных решениях, параметрах элементов, сведения о состоянии текущих разработок в виде проектных решений, параметров проектируемых объектов и т. п.

*Лингвистическое обеспечение САПР* представлено совокупностью языков, применяемых для описания процедур автоматизированного проектирования и проектных решений, а также языками программирования.

*Методическое обеспечение САПР* составляют документы, содержащие правила проектирования в данной системе.

*Организационное обеспечение САПР* включает в себя положения, инструкции, приказы, штатные расписания, квалификационные требования и другие документы, регламентирующие организационную структуру подразделений проектной организации и взаимодействие подразделений с комплексом средств автоматизированного проектирования.

**Классификация САПР.** Классифицировать САПР можно по следующим признакам:

- степень формализации решаемых задач;
- функциональное назначение;
- специализация;
- техническая организация.

По степени формализации решаемых задач САПР могут быть построены на решении:

- полностью формализуемых задач;
- частично формализуемых задач;
- неформализуемых задач.

Системы, построенные на решении полностью формализуемых задач, для проектирования машиностроительных конструкций обычно не пригодны, поскольку математические модели объектов проектирования и процессов их функционирования настолько сложны, что полное и точное их математическое описание на сегодняшний день невозможно. Такие системы могут применяться только для решения простейших задач проектирования.

Системы, построенные на решении неформализуемых задач, в настоящее время находятся в стадии исследований и разработки («искусственный интеллект») и в целях проектирования также не применяются.

Необходимо отметить, что в обоих случаях процесс проектирования происходит без вмешательства человека. Таким образом, речь здесь идет о системах не автоматизированного, а автоматического проектирования.

Для решения задач в области автомобиле- и тракторостроения в настоящее время пригодны только системы, построенные на решении частично формализуемых задач.

Безусловно, часть задач, связанных с проектированием некоторых простейших элементов конструкции автомобиля и трактора, может быть решена с использованием автоматического проектирования, но для проектирования сложных агрегатов и систем этих машин сегодня полная автоматизация невозможна. Кроме того, если речь идет о таких понятиях, как форма кабины и кузова, детали интерьера, то на их конструкцию помимо функциональных требований (аэродинамические свойства, эргономика, безопасность) влияют и субъективные факторы, например мода, что также невозможно описать языком математических зависимостей.

По функциональному назначению САПР могут быть подразделены в зависимости от решаемых задач, определяемых составом функциональной части системы. Так, можно выделить следующие системы:

- расчетно-оптимизационные;
- графические;

- графоаналитические;
- информационные и т. п.

По специализации САПР можно подразделить на специализированные и инвариантные. Поскольку задачи автоматизированного проектирования очень сложны, то, как правило, САПР представляют собой специализированные системы, создаваемые для решения достаточно узких задач в одной области техники.

По технической организации САПР бывают одноуровневые, построенные на базе одной достаточно производительной ЭВМ с набором необходимых периферийных устройств, и многоуровневые, включающие в себя помимо базовой ЭВМ ряд подчиненных автоматизированных рабочих мест (АРМ), построенных на основе ЭВМ более низкого уровня.

#### 1.4. Подходы и методы проектирования в САПР

В современных САПР используют различные подходы и методы проектирования. На практике, особенно при проектировании объектов машиностроения, редко встречаются случаи, когда существует возможность полного описания объекта в рамках одной программы. Обычно задачи проектирования настолько сложны, что это невозможно.

Описания технических объектов должны быть по сложности согласованы с возможностями восприятия человеком и возможностями оперирования описаниями в процессе их преобразования с помощью имеющихся средств проектирования. Однако выполнить это требование в рамках некоторого единого описания, не расчлняя его на составные части, удастся лишь для простых изделий. Как правило, требуется структурирование описаний и соответствующее расчленение представлений о проектируемых объектах на иерархические уровни и аспекты; это позволяет распределять работы по проектированию сложных объектов между подразделениями проектной организации, что способствует повышению эффективности и производительности труда проектировщиков.

**Блочно-иерархический подход.** Разделение описаний по степени детализации отображаемых свойств и характеристик объекта лежит в основе *блочно-иерархического подхода к проектированию* и приводит к появлению *иерархических уровней (уровней абстрагирования)* в представлениях об объекте.

На каждом иерархическом уровне используются свои понятия системы и элементов. На верхнем уровне подлежащий проектированию сложный объект рассматривается как система взаимосвязанных и взаимодействующих элементов. Каждый из этих элементов представляет собой также довольно сложный объект, кото-

рый, в свою очередь, рассматривается как система на следующем, нижележащем уровне. Как правило, выделение элементов происходит по функциональному признаку. Подобное разделение продолжается вплоть до получения на некотором уровне элементов, описания которых дальнейшему делению не подлежат; эти элементы по отношению к объекту называют *базовыми элементами*.

Таким образом, *принцип иерархичности* означает структурирование представлений об объектах проектирования по степени детальности описания, а *принцип декомпозиции (блочности)* — разбиение представлений каждого уровня на ряд составных частей (блоков) с возможностями раздельного (поблочного) проектирования объектов на каждом из уровней.

Конструкция машины как объекта проектирования представляет собой сложную систему. Вследствие этого, математическое описание конструктивных элементов должно базироваться на блочно-иерархическом подходе к процессу конструирования.

Для объектов машиностроения (в том числе автомобилей и тракторов) характерны следующие иерархические уровни: машина — агрегат — узел — деталь. Уровень IV (низший уровень) составляют детали машин, уровень III — совокупность деталей (узел — сборочная единица), уровень II — агрегат (совокупность узлов), уровень I — машина (совокупность агрегатов). Для более точного представления иерархии системы машин могут быть предусмотрены дополнительные подуровни, например узлы машины могут быть разбиты на подузлы и т.д.

Рассмотрим иерархическое представление транспортной машины (рис. 1.2). Уровень I представлен самой транспортной машиной, уровень II — ее агрегатами и системами (рулевое управление

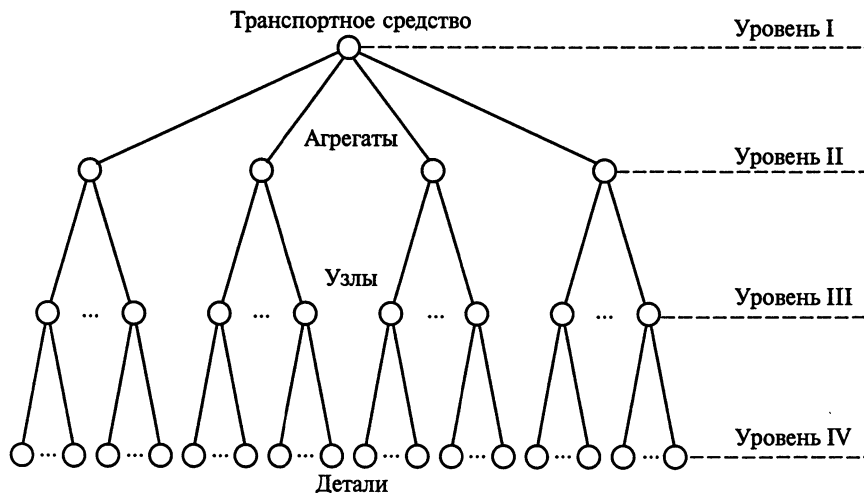


Рис. 1.2. Пример иерархического представления транспортной машины

ние, тормозное управление, сцепление, коробка передач и т.д.). Элементами уровня III являются узлы агрегатов и систем (рулевой и тормозные механизмы, синхронизаторы и пр.). На уровне IV располагаются детали узлов. Возможно введение подуровней, например между уровнями I и II можно расположить трансмиссию, в которую входят ее агрегаты (сцепление, коробка передач, карданные передачи и т.д.).

Соответственно иерархии объектов проектирования можно построить иерархию их математических моделей.

Итак, блочно-иерархический подход к проектированию — подход, основанный на разделении описаний проектируемых объектов на *иерархические уровни* по степени подробности отражения свойств объектов, а также на соответствующем разделении процесса проектирования на группы *проектных процедур*, связанных с получением и преобразованием описаний выделенных иерархических уровней.

Иерархическая структура описаний имеет место в каждом из *аспектов описаний* сложных систем. Например, описания функционального аспекта делятся на *мета-, макро- и микроуровни*. В описаниях конструкций выделяют уровни комплектов оборудования, агрегатов, сборочных единиц, деталей; в описаниях технологических процессов — уровни принципиальных схем, маршрутной и операционной технологии. При переходе с более высокого иерархического уровня на более низкий степень подробности описания объекта возрастает. Для сохранения приемлемой сложности описаний (приемлемой размерности решаемых задач) при таком переходе приходится производить *декомпозицию* описаний на блоки с последующим поблочным рассмотрением и преобразованием описаний. В результате появляется возможность свести решение малого числа чрезмерно сложных задач к решению большого числа задач, но приемлемой сложности.

Если решение задач высоких иерархических уровней предшествует решению задач более низких иерархических уровней, проектирование называют нисходящим. Если раньше выполняются этапы, связанные с низшими иерархическими уровнями, проектирование называют восходящим. У каждого из этих двух видов проектирования имеются преимущества и недостатки.

**Нисходящее проектирование.** При таком проектировании сначала создаются описания на более высоких иерархических уровнях, а затем на более низких (проектирование сверху вниз).

Например, последовательность проектирования может быть такой: структурная схема транспортного средства — модели агрегатов и систем — расчетные схемы узлов и деталей. *Функциональное проектирование* сложных систем чаще всего бывает нисходящим вплоть до уровня, на котором элементы — унифицированные объекты.



При нисходящем проектировании система разрабатывается в условиях, когда ее элементы еще не определены и, следовательно, сведения об их возможностях и свойствах носят предположительный характер.

**Восходящее проектирование.** Проектирование, при котором выполнение процедур по получению описаний низких иерархических уровней предшествует выполнению процедур по получению описаний высоких иерархических уровней, называется восходящим (проектирование снизу вверх).

Например, объекты могут проектироваться в такой последовательности: детали — сборочные единицы — агрегаты. Восходящее проектирование характерно для объектов, содержащих унифицированные элементы.

При восходящем проектировании элементы проектируются раньше системы, и, следовательно, предположительный характер имеют требования к элементам.

В обоих случаях из-за отсутствия исчерпывающей исходной информации имеют место отклонения от потенциально возможных оптимальных технических результатов. Однако нужно помнить, что подобные отклонения неизбежны при блочно-иерархическом подходе к проектированию и что какой-либо приемлемой альтернативы блочно-иерархическому подходу при проектировании сложных объектов не существует. Поэтому оптимальность результатов блочно-иерархического проектирования следует рассматривать с позиций технико-экономических показателей, включающих в себя, в частности, материальные и временные затраты на проектирование.

Поскольку принимаемые предположения могут не оправдаться, часто требуется повторное выполнение проектных процедур предыдущих этапов после выполнения проектных процедур последующих этапов. Такие повторения обеспечивают последовательное приближение к оптимальным результатам и обуславливают итерационный характер проектирования. Следовательно, итерационность нужно относить к важным принципам проектирования сложных объектов.

На практике обычно сочетают и восходящее, и нисходящее проектирование. Например, восходящее проектирование имеет место на всех тех иерархических уровнях, на которых используются унифицированные элементы. Очевидно, что такие элементы, ориентированные на применение в различных системах определенного класса, разрабатываются раньше, чем та или иная конкретная система из этого класса.

Весьма полезным в практике разработки новых конструкций может быть использование эвристических приемов синтеза.

**Эвристический прием синтеза.** Неформализованный прием, используемый при синтезе технических объектов и указывающий, в

каком направлении искать нужное техническое решение, называется эвристическим.

Имеются фонды эвристических приемов, помещаемые в *базы данных*, для использования инженерами в интерактивных процедурах синтеза. Эвристические приемы делятся на группы преобразований вида движения, материала, геометрической формы в пространстве, во времени, добавлением, исключением, заменой, по аналогии и т. п. Примерами эвристических приемов могут служить «изменить направление вращения», «перейти от однородных материалов к композиционным», «вывернуть форму наизнанку», «поменять местами противоположно размещенные элементы», «превратить асинхронный процесс в синхронный», «расходуемые элементы восстановить непосредственно в процессе работы», «исключить наиболее напряженный элемент», «заменить механическую обработку обработкой без снятия стружки», «обратить внимание на способ решения задачи, обратной данной» и др.

## 1.5. Способы представления графической информации в ЭВМ

**Машинная графика.** Раздел информатики, изучающий средства и методы создания и преобразования графических изображений объектов с помощью ЭВМ, называется машинной графикой. Машинная графика используется для ввода информации, первоначально имеющей графическую форму, в ЭВМ и вывода информации в графической форме из ЭВМ. При этом под графической формой представления информации понимают чертежи и эскизы деталей и сборочных единиц, разнообразные схемы, диаграммы, графики, гистограммы и т. п. Если ввод и вывод графической информации происходит в процессе диалога человека с ЭВМ, машинную графику называют интерактивной.

К основным техническим средствам машинной графики относятся графические дисплеи вместе с устройствами управления маркером, графопостроителями, координатографы, кодировщики графической информации.

Математическое обеспечение машинной графики включает в себя геометрические модели, методы и алгоритмы их преобразования. Геометрические модели можно представить состоящими из отдельных элементарных частей — графических примитивов (отрезки сплошной и пунктирной прямых линий, дуги окружности и эллипса, прямоугольники, многоугольники), а алгоритмы преобразования геометрических моделей — из элементарных операций (масштабирование изображения — сжатие или расширение, его поворот, сдвиг, мультиплицирование, зеркальное отображение, выделение окна).

Наряду с автоматическим вводом информации непосредственно с графических документов применяют также предварительное кодирование — представление информации на графических входных языках. Программное обеспечение машинной графики представлено в САПР, во-первых, языковыми процессорами, преобразующими информацию, вводимую с кодировщика или представленную на входном графическом языке, в структуры данных применяемых прикладных программ, и, во-вторых, программами, преобразующими результаты выполнения проектных процедур прикладными программами в команды управления устройствами вывода.

Основным понятием машинной графики является графический объект, представляющий собой систему уравнений, описывающих геометрические характеристики объекта проектирования в трехмерном пространстве.

На устройствах отображения графический объект изображается в виде проекций (ортогональных, аксонометрических, перспективных и др.), разрезов и сечений.

Существуют три вида моделей объектов машинной графики — стержневая, оболочечная и объемная.

*Стержневая модель* (рис. 1.3, а) основана на представлении объекта в виде прямолинейных стержней, представляющих собой ребра модели, соединенные между собой в узлах — вершинах (1, 2, 3, ...). Основными уравнениями, применяемыми для описания модели, являются уравнения прямой линии в трехмерном пространстве. Ребра являются границами граней объекта.

Математическое описание моделей такого рода сравнительно простое, что обуславливает высокое быстродействие программного обеспечения, основанного на их использовании. К недостаткам следует отнести сложность или невозможность представления

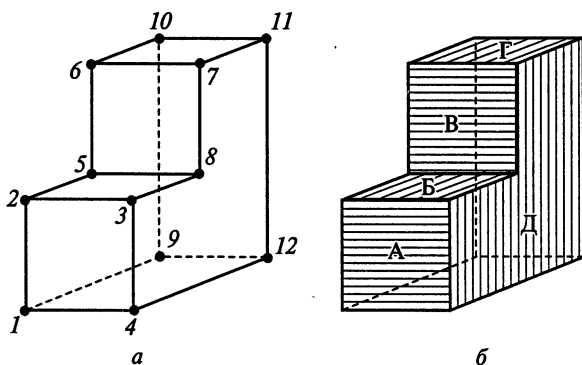


Рис. 1.3. Модели представления объектов машинной графики:

а — стержневая; б — оболочечная

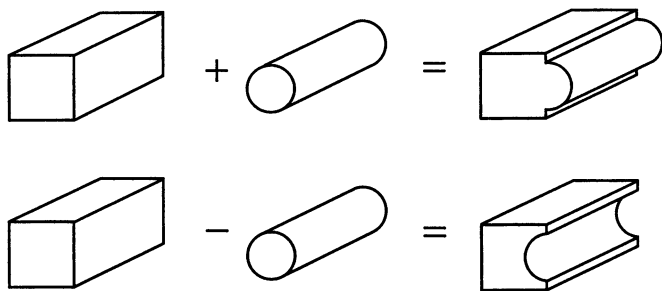


Рис. 1.4. Операции с объемными элементами

неплоских граней объекта, а также невозможность представления внутреннего облика объекта, построения произвольных его разрезов и сечений.

*Оболочечная модель* (рис. 1.3, б) основана на представлении внешнего облика геометрического объекта в виде совокупности поверхностей, являющихся гранями модели (А, Б, В, ...). Линии пересечения поверхностей образуют ребра модели. Такая модель описывается системой уравнений поверхностей и может быть использована для моделирования внешнего облика объектов любой формы.

Основной ее недостаток — невозможность представления внутреннего облика объекта, построения произвольных его разрезов и сечений.

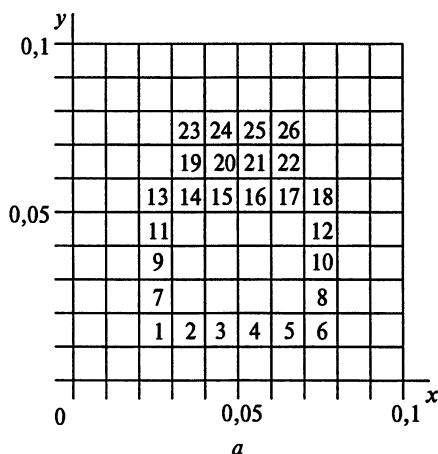
Наиболее современной моделью, нашедшей широкое применение в САПР, является *объемная (твердотельная) модель*. Общепринятым порядком моделирования твердого тела является последовательное выполнение булевых операций (объединение, вычитание и пересечение) над объемными элементами (сферы, призмы, цилиндры, конусы, пирамиды и т.д.). Эти элементы описываются теми же уравнениями, что и поверхности оболочечной модели, однако объемные элементы считаются заполненными. Пример выполнения операций с объемными элементами показан на рис. 1.4.

Такая модель требует более сложного программного обеспечения, но при использовании реализующих ее программных средств на современных ЭВМ может быть обеспечена достаточная для диалогового режима скорость графических преобразований. Эта модель не имеет ограничений в возможностях построения и отображения любых проекций, разрезов и сечений.

**Представление графической информации в ЭВМ.** Графическая информация может быть представлена разными способами, это зависит от назначения этой информации и типа устройств, для которых она предназначена.

*Координатный способ* представления графической информации основан на представлении плоского (монокромного) изображения в виде координат прямоугольных растрэлементов. При этом фиксируются только растрэлементы, принадлежащие изображению, а фоновая область не рассматривается. На рис. 1.5, *а* показано разбиение изображения на растрэлементы, представляющие собой квадрат со стороной 0,01 мм. Данные о координатах растрэлементов заносятся в таблицу, пример которой приведен на рис. 1.5, *б*. Как видно из рисунка, для достаточно точного представления графической информации требуется огромный массив данных, включающий в себя порядковые номера и координаты растрэлементов, вследствие чего этот способ неудобен для хранения графической информации. Область его применения — сканирующие и растровые устройства ввода-вывода графической информации. Размер растрэлемента определяется техническими характеристиками указанных устройств.

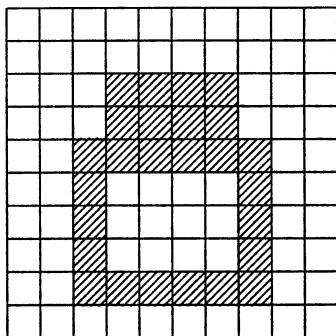
Разновидностью координатного способа является *рецепторный способ* представления графической информации. Он основан на представлении всего поля изображения (в том числе фоновой области) в виде прямоугольных областей, называемых *рецепторами* (рис. 1.6, *а*). При этом рецептор, принадлежащий линиям изображения, кодируется двоичной единицей и называется «возбужденным», а рецептор, принадлежащий фоновой области изображения, кодируется двоичным нулем и носит название «белого» (рис. 1.6, *б*). Видно, что кодирование каждого отдельного рецептора более компактно, чем растрэлемента в координатном способе. Однако необходимость кодирования фоновой области, занимаю-



Номера точек	x	y
1	0,03	0,02
2	0,04	0,02
3	0,05	0,02
~~~~~		
23	0,04	0,08
24	0,05	0,08
25	0,06	0,08
26	0,07	0,08

Рис. 1.5. Координатный способ представления графической информации: *а* — разбиение изображения на растрэлементы; *б* — таблица с координатами растрэлементов





*a*

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	0	0	0
0	0	0	1	1	1	1	0	0	0
0	0	1	1	1	1	1	1	0	0
0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	1	0	0
0	0	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0

*б*

Рис. 1.6. Рецепторный способ представления графической информации:  
*a* — в виде прямоугольных областей (рецепторов); *б* — в двоичном коде

шей, например, на чертежах значительное пространство, приводит к тому, что и этот способ не позволяет получить компактную запись информации. Область применения данного способа представления графической информации та же, что и в случае координатного способа.

Более экономичным с точки зрения объема хранимой информации является способ поэлементного представления графической информации. Он основан на представлении изображения в виде совокупности графических примитивов, в качестве которых могут выступать отрезок прямой линии, дуга, окружность. Для представления любого из этих элементов достаточно записать код этого элемента, координаты его опорных точек и (или) ориентацию в пространстве, коды типа и цвета линий. Естественно, такой способ позволяет представить графическую информацию значительно компактнее рассмотренных выше.

Область его применения — устройства кодирования графической информации (дигитайзеры). Он также применяется для представления в более компактном виде графической информации, полученной рассмотренными выше способами. Для этого используются программы векторизации растровых изображений, позволяющие определять упорядоченные множества рецепторов и растр-элементов, а затем представлять их в виде графических примитивов (обычно отрезков прямой).

Дальнейшим развитием способа поэлементного представления графической информации является структурно-символический способ. Он основан на использовании для формирования изображения типовых графических элементов (ТГЭ). В качестве ТГЭ могут применяться сложные графические изображения, состоящие из множества графических примитивов. Для описания каждого эле-

мента достаточно указать его тип, координаты опорной точки и ориентацию в пространстве, а также модуль его размера. Этот способ привлекателен тем, что часто встречающиеся в данной области применения элементы графических изображений могут быть систематизированы и занесены в библиотеку. Способ является более компактным при хранении графического изображения и весьма удобен при формировании изображений.

*Аналитический способ* представления графической информации основан на ее представлении в виде уравнений поверхностей. При этом предполагается, что любое изображение можно представить множеством уравнений, соответствующих множеству поверхностей, которые пересекаются с плоскостью изображения. Такой способ обычно является математической основой современного программного обеспечения машинной графики.

## 1.6. Задачи синтеза и анализа. Оптимальное проектирование конструкций

Проектирование — сложный и трудно формализуемый процесс, объединяющий такие важные процедуры, как синтез структуры, выбор параметров элементов, анализ и принятие решения. Особенно важна начальная стадия проектирования, когда выбираются эффективный физический принцип действия, рациональное техническое решение и определяются оптимальные значения параметров.

**Задачи синтеза и анализа.** Под *синтезом* понимаются проектные процедуры, направленные на получение новых описаний проектируемого объекта в соответствии с заданными показателями его функционирования. *Анализ* — это проектные процедуры, имеющие целью получение информации о свойствах проектируемого объекта по заданному его описанию.

Задачи синтеза связаны с созданием проектных документов и самого проекта, а задачи анализа связаны с оценкой проектных документов.

Процедуры синтеза делятся на процедуры структурного и параметрического синтеза.

Поиск рационального технического решения при выбранном физическом принципе действия осуществляется методами *структурного синтеза*. Определение оптимальных значений параметров элементов технической системы известной структуры представляет собой задачу *параметрического синтеза*, или *параметрической оптимизации*.

Целью структурного синтеза является определение структуры объекта — перечня типов элементов, составляющих объект, и способа связи элементов между собой в составе объекта.

Параметрический синтез заключается в определении числовых значений параметров элементов при заданных структуре и условиях работоспособности, влияющих на выходные параметры объекта, т. е. при параметрическом синтезе нужно найти точку или область в пространстве внутренних параметров, в которых выполняются те или иные условия (обычно условия работоспособности).

Классификация проектных процедур приведена на рис. 1.7.

Процедуры анализа делятся на процедуры одно- и многовариантного анализа.

При *одновариантном анализе* заданы значения внутренних и внешних параметров, требуется определить значения выходных параметров объекта. Удобно использовать геометрическую интерпретацию этой задачи, связанную с понятием пространства внутренних параметров; это  $n$ -мерное пространство, в котором для каждого из  $n$  внутренних параметров  $x_i$  выделена координатная ось. При одновариантном анализе задается также некоторая точка в пространстве внутренних параметров и требуется в этой точке определить значения выходных параметров. Подобная задача обычно сводится к однократному решению уравнений, составляющих



Рис. 1.7. Классификация проектных процедур

математическую модель, что и обуславливает название данного вида анализа.

*Многовариантный анализ* заключается в исследовании свойств объекта в некоторой области пространства внутренних параметров. Такой анализ требует многократного решения систем уравнений (многократного выполнения одновариантного анализа).

Синтез называется *оптимизацией*, если определяются наилучшие в заданном смысле структура и значения параметров. При расчетах оптимальных значений параметров при заданной структуре говорят о параметрической оптимизации. Задачу выбора оптимальной структуры называют *структурной оптимизацией*.

Постановка задачи оптимизации имеет содержательный смысл только в том случае, когда появляется необходимость выбора одного из конкурирующих вариантов, полученных при ограниченности ресурсов. Техническое проектирование всегда ведется в условиях жестких ограничений на материальные, энергетические, временные и прочие виды ресурсов. Вместе с тем средства САПР позволяют выполнить разработку нескольких альтернативных вариантов, поэтому окончательный выбор технического объекта (принятие решения) необходимо проводить с учетом выработанных *правил предпочтения* на основании установленных критериев. Выбор критерия является одним из важных этапов постановки задачи оптимизации, так как все последующие действия направлены на поиск объекта, наиболее близкого к оптимальному по выбранному критерию.

В основе построения правила предпочтения лежит *целевая функция*, количественно выражающая качество объекта и потому называемая также *функцией качества*, или *критерием оптимальности*. Формирование целевой функции всегда выполняется с учетом различных выходных параметров проектируемого устройства. В зависимости от содержательного смысла этих параметров и выбранного способа их сочетания в целевой функции качество объекта будет тем выше, чем больше ее значение (максимизация) или чем меньше ее значение (минимизация).

Выбор целевой функции носит субъективный характер, и поэтому объект может быть оптимален только в смысле данного критерия.

В большинстве подходов к оценке технического объекта принято ориентироваться на эталонные образцы, на мнение ведущих специалистов отрасли (экспертные оценки) или на технико-экономические показатели, определяемые техническим заданием (ТЗ) на проектирование. При подготовке ТЗ обычно учитываются достижения, полученные в мировой практике, а также в той или иной мере экспертные оценки, поэтому более объективной следует считать ориентацию на ТЗ. Наиболее полная оценка проектных решений может быть выполнена на основе анализа технико-

экономических показателей с учетом требований, сформулированных в ТЗ.

Качество функционирования любой системы характеризуется вектором выходных параметров  $Y = (y_1, y_2, \dots, y_m)$ . Некоторые из выходных параметров могут быть измерены количественно, другие представляют собой качественную сторону объекта, поэтому все выходные параметры обычно делят соответственно на количественные и качественные. В дальнейшем под вектором  $Y$  будем подразумевать вектор количественных параметров.

К выходным параметрам, определяющим качество автомобиля или трактора, можно отнести тяговые и скоростные характеристики, показатели разгонной и тормозной динамики, управляемости и устойчивости, комфортабельности, расход топлива, количественный и качественный состав загрязняющих веществ, выбрасываемых в атмосферу с выхлопными газами, и многие другие.

Значение целевой функции может возрастать или убывать с увеличением качества выходного параметра, поэтому в первом случае необходимо искать максимум, а во втором — минимум целевой функции.

Каждый из составляющих вектор  $Y$  выходных параметров зависит от множества внутренних параметров проектируемого объекта. Следует отметить, что значения некоторых внутренних параметров назначаются и не подлежат изменению. К таким параметрам, например, можно отнести параметры унифицированных элементов или те из них, значения которых оговорены в техническом задании на проектирование. Остальные параметры можно выбрать, ориентируясь на прототипы с учетом собственного опыта и оценки возможных перспектив развития конструкций подобных объектов.

Внутренние параметры, значения которых могут меняться в процессе оптимизации и которые являются аргументами целевой функции, называют *управляемыми параметрами*.

Пусть в проектируемом объекте имеется  $n$  управляемых параметров, образующих вектор  $X = (x_1, x_2, \dots, x_n)$ . Обозначим целевую функцию через  $F(X)$ , а область ее определения — через  $XP$ . Вектор  $X$  определяет координаты точки в области определения  $XP$ . Если элементы вектора  $X$  принимают только дискретные значения,  $XP$  является *дискретным множеством точек* и задача оптимизации относится к области *дискретного* (в частном случае целочисленного) *программирования*.

Большинство задач параметрической оптимизации технических объектов формулируется в терминах непрерывных параметров. Если экстремум целевой функции ищется в неограниченной области  $XP$ , его называют *безусловным*, а методы поиска — *методами безусловной оптимизации*. Если экстремум целевой функции ищется в ограниченной области  $XP$ , его называют *условным*.

Для решения задач проектирования в машиностроении характерны методы условной оптимизации.

Таким образом, задачу поиска оптимального решения можно в общем случае сформулировать следующим образом:

$$\min(\max) F(\mathbf{X}), \mathbf{X} \in \mathbf{XP},$$

где  $\mathbf{X}$  — вектор управляемых параметров;  $F(\mathbf{X})$  — целевая функция;  $\mathbf{XP}$  — область допустимых значений вектора управляемых параметров.

**Постановка задачи оптимизации.** Основная проблема постановки задачи оптимизации заключается в выборе целевой функции. Сложность выбора целевой функции состоит в том, что любой технический объект первоначально имеет векторный характер критериев оптимальности (многокритериальность), причем улучшение одного из выходных параметров, как правило, приводит к ухудшению другого, так как все выходные параметры являются функциями одних и тех же управляемых параметров и не могут изменяться независимо друг от друга. Такие выходные параметры называют *конфликтными*.

Целевая функция должна быть одна (принцип однозначности). Сведение многокритериальной задачи к однокритериальной называют *сверткой векторного критерия*. Задача поиска его экстремума сводится к задаче математического программирования.

В зависимости от того, каким образом выбираются и объединяются выходные параметры в скалярной функции качества, различают частные, аддитивные, мультипликативные, минимаксные, статистические критерии и т. д.

*Частные критерии* могут применяться в случаях, когда среди выходных параметров можно выделить один основной параметр  $u_j(\mathbf{X})$ , наиболее полно отражающий эффективность проектируемого объекта; этот параметр принимают за целевую функцию. Различные группы и категории транспортных и тяговых средств очень сильно отличаются по назначению (например, тракторы, грузовые автомобили и спортивные автомобили), поэтому для них в качестве выходных параметров, применимых как целевая функция, могут быть и грузоподъемность, и расход топлива, и динамические качества, и многие другие. Условия работоспособности объекта по всем остальным выходным параметрам относят при этом к функциональным ограничениям. Оптимизация на основе такой постановки называется *оптимизацией по частному критерию*.

Достоинство такого подхода — его простота, а существенный недостаток — то, что возможно получить высокое качество объекта только по тому параметру, который принят в качестве целевой функции, в то время как остальные выходные параметры будут только приемлемыми.

Теоретически возможен следующий подход. Можно выбрать один, наиболее значимый выходной параметр и провести его оптимизацию. Затем выбрать следующий по значимости параметр и провести его оптимизацию при условии, что значения управляемых параметров, входящие в выражение как для первого, так и для второго выходных параметров, при оптимизации второго параметра должны быть зафиксированы на уровне, достигнутом при оптимизации первого параметра, а оптимизация должна проводиться только с использованием управляемых параметров, не вошедших в описание выходного параметра, оптимизируемого первым. Далее по такому же принципу можно оптимизировать и все остальные выходные параметры. Однако это возможно только в тех случаях, когда выходные параметры зависят хотя бы частично от различных наборов управляемых параметров, что на практике бывает далеко не всегда.

*Взвешенный аддитивный критерий* применяют тогда, когда условия работоспособности позволяют выделить две группы выходных параметров. В первую группу входят выходные параметры  $y_i^+(\mathbf{X})$ , значения которых в процессе оптимизации нужно увеличивать (производительность, вероятность безотказной работы, для транспортных средств — грузоподъемность, максимальная скорость и т. п.), во вторую — выходные параметры  $y_i^-(\mathbf{X})$ , значения которых следует уменьшать (расход топлива, масса транспортного средства, время разгона и пр.).

В этом случае целевая функция свертки будет иметь вид:

$$F(\mathbf{X}) = \sum_{i=1}^k \lambda_i y_i^-(\mathbf{X}) - \lambda_i y_i^+(\mathbf{X}), \quad (1.1)$$

где  $\lambda_i > 0$  — весовой коэффициент, определяющий степень важности  $i$ -го выходного параметра (обычно значения  $\lambda_i$  выбираются проектировщиком и в процессе оптимизации остаются постоянными).

Объединение нескольких выходных параметров, имеющих в общем случае различную физическую размерность и величину, в одной скалярной целевой функции требует предварительного нормирования этих параметров. Способы нормирования параметров могут быть различны. Наиболее простым является отнесение этого параметра к некоторой заранее заданной средней величине, определяемой из опыта предыдущего проектирования, или к допускаемой величине (например, допускаемые напряжения). В этом случае все составляющие выражения (1.1) будут безразмерными величинами одного порядка, что делает удобным манипулирование коэффициентами  $\lambda_i$  в процессе проектирования.

Наиболее типичным случаем параметрической оптимизации технических объектов является поиск значений вектора управляемых

параметров  $\mathbf{X}$ , определяющих экстремум целевой функции при наличии ограничений. Для дальнейшего изложения будем считать, что в процессе оптимизации ищется минимум целевой функции.

Таким образом, задачу поиска оптимального решения можно в общем случае сформулировать следующим образом:

$$\min F(\mathbf{X}). \quad (1.2)$$

Здесь  $\mathbf{X} \in \mathbf{XP}$  при прямых ограничениях  $x_{ni} \leq x_i \leq x_{vi}$  для любого  $i \in [1, n]$ , где  $x_{ni}$ ,  $x_{vi}$  — максимально и минимально допустимые значения  $i$ -го управляемого параметра;  $n$  — размерность пространства управляемых параметров и при функциональных ограничениях.

*Функциональные ограничения*, как правило, представляют собой условия работоспособности выходных параметров, не вошедших в целевую функцию, и могут быть:

типа равенств

$$\psi(\mathbf{X}) = 0;$$

типа неравенств

$$\phi(\mathbf{X}) > 0,$$

где  $\psi(\mathbf{X})$  и  $\phi(\mathbf{X})$  — вектор функции.

Выбор метода решения зависит от способа постановки задачи оптимизации.

Любая из точек  $\mathbf{X} \in \mathbf{XP}$  является *допустимым решением задачи*. Часто параметрический синтез ставится как задача определения любого из допустимых решений. Однако гораздо важнее решить задачу оптимизации — найти оптимальное решение среди допустимых.

## 1.7. Методы решения задач оптимизации

Расчет выходных параметров объекта в САПР, как правило, выполняется через численное решение систем алгебраических и дифференциальных уравнений. Целевая функция и функциональные ограничения определяются неявным образом, что не позволяет оценить их свойства (выпуклость, вогнутость и т. д.) с приемлемыми затратами вычислительных ресурсов. Поэтому решение задач оптимизации в САПР ведется с помощью *поисковых методов* математического программирования, использующих предшествующую информацию для построения улучшенного решения задачи, т. е. эти методы являются итерационными.

По ряду признаков задачи поиска экстремума могут быть отнесены к тому или иному классу. Большинство постановок задач параметрической оптимизации технических систем сводятся к задачам нелинейного программирования, так как целевая функция



и ограничения описываются нелинейными зависимостями от вектора управляемых параметров. В отдельных случаях при проектировании удается так сформулировать задачу, что целевая функция и ограничения являются линейными функциями своих аргументов. Тогда имеет место задача линейного программирования. В зависимости от типа искомого экстремума различают методы локальной и глобальной, условной и безусловной оптимизации. Практически используемые методы в основном являются методами локального поиска.

Одновременно надежные и экономичные методы поиска глобального экстремума в настоящее время неизвестны. Надежным, но крайне неэкономичным методом глобального поиска является *метод сканирования*. При его применении область определения  $F(X)$  в пространстве управляемых параметров разбивается на  $k$  подобластей, в центре каждой из которых вычисляется значение целевой функции. Если функция зависит от  $n$  параметров, необходимо выполнить  $k^n$  вариантов расчетов. Чтобы получить достоверную картину поведения гиперповерхности отклика целевой функции, необходимо сканировать допустимую область с достаточно малым шагом, поэтому даже для сравнительно несложных задач затраты машинного времени на поиск становятся недопустимо большими. Этот недостаток характерен и для *методов случайного поиска глобального экстремума*. Однако затраты ресурсов на случайный поиск можно сделать приемлемыми, если не предъявлять высоких требований к надежности определения экстремума.

Наиболее многочисленную группу составляют методы безусловной оптимизации. Некоторое представление о широко применяемых методах этой группы дает рис. 1.8. В зависимости от порядка используемых производных целевой функции по управляемым параметрам методы безусловной оптимизации делят на методы нулевого, первого и второго порядков.

В методах нулевого порядка (прямых методах) информация о производных не используется. Для методов первого порядка необходимо вычислять как значение функции качества, так и ее первые частные производные (градиентные методы). В методах второго порядка организация поиска экстремума ведется с учетом значений целевой функции, ее первых и вторых производных.

В зависимости от количества управляемых параметров целевой функции различают методы одномерного и многомерного поиска. Одномерный поиск может рассматриваться как самостоятельная задача, если аргументом целевой функции является один параметр. Такой же поиск используется в качестве части процедуры многомерной оптимизации в тех случаях, когда необходимо найти оптимальный шаг в выбранном направлении.

Задача условной оптимизации может быть сформулирована как задача безусловной оптимизации с помощью методов Лагранжа

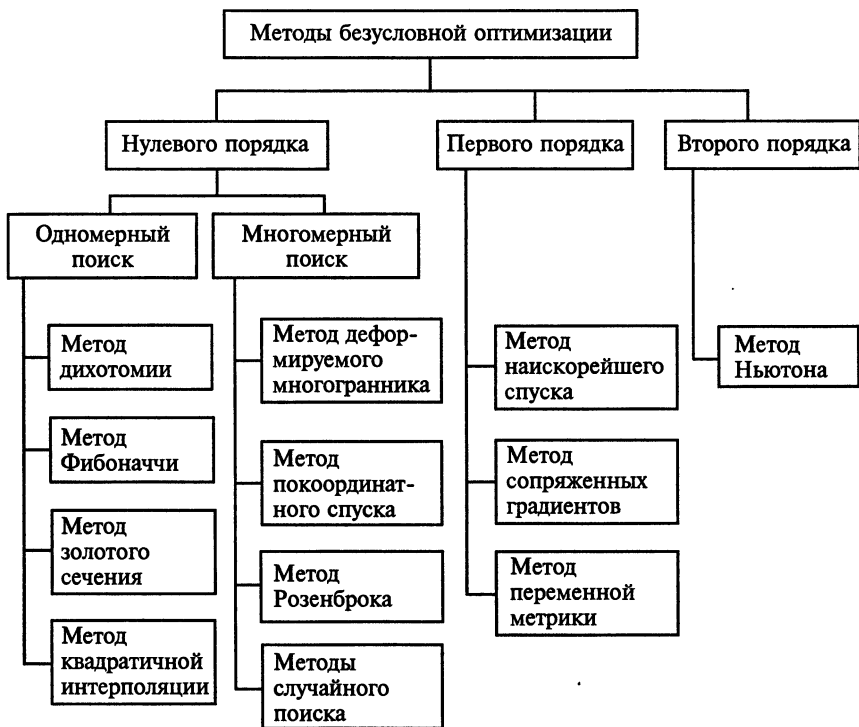


Рис. 1.8. Классификация методов оптимизации

или штрафных функций. Тогда для ее решения применяются методы безусловной оптимизации.

Задача условной оптимизации может быть решена и методами условной оптимизации (методы проекции градиента, допустимых направлений и др.).

Практически во всех методах оптимизации стремятся построить такую последовательность значений  $X_0, X_1, X_2, \dots$ , при которой  $F(X_0) > F(X_1) > F(X_2) > \dots$ . В этом случае обеспечивается сходимость результатов и можно надеяться, что минимум функции будет найден.

Важной характеристикой методов является их скорость сходимости. Однако оценка сходимости того или иного метода обычно базируется на некоторых теоретических предпосылках относительно особенностей целевой функции (например, функция дважды непрерывно дифференцируема или сильно выпукла) и зависит от выбора начальной точки поиска. Теоретические предпосылки относительно реальных целевых функций могут не удовлетворяться, поэтому скорость сходимости в этих случаях можно рассматривать как сравнительную оценку метода.

Численные методы поиска оптимума позволяют построить последовательность шагов от начальной точки  $X_0$  через некоторые промежуточные точки  $X_k$  к локальному экстремуму  $X^*$ .

Схема алгоритма поиска оптимального решения для общего случая показана на рис. 1.9. Как отмечалось выше, выбор исходной точки поиска  $X_0$  во многом определяет успех решения всей задачи. Очевидно, что  $X_0$  должна принадлежать области определения целевой функции и, чем ближе к экстремуму выбрана  $X_0$ , тем быстрее и с большей вероятностью экстремум будет найден. Сущность метода оптимизации определяется этапами 2 и 3 алгоритма, на которых выбирается направление дальнейшего поиска и вычисляются координаты очередной точки  $X_{k+1}$  на траектории поиска. Далее в точке  $X_{k+1}$  вычисляются значения целевой функции  $F(X_{k+1})$  и функций-ограничений, т.е. определяется информация, позволяющая судить о достигнутом успехе. Инженер может назначить различные условия прекращения поиска, и, в зависимости от степени их выполнения, поиск будет продолжен или прекратится.

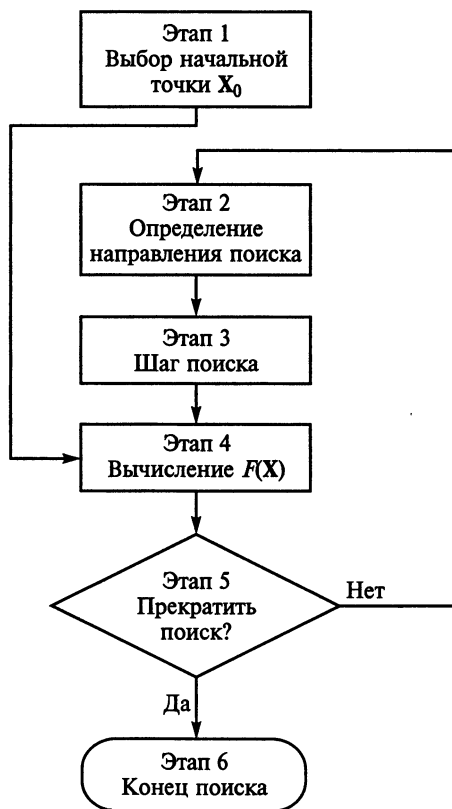


Рис. 1.9. Алгоритм поиска оптимального решения

Методы одномерного поиска строятся в предположении уни-модальности (одноэкстремальности) функции  $F(x)$  на заданном интервале  $[a, b]$ . К функции не предъявляются требования дифференцируемости или непрерывности. Предполагается, что для любого  $x \in [a, b]$  значение  $F(x)$  может быть вычислено, т. е. найдено путем вычислительного эксперимента.

Методы одномерного поиска можно разделить на методы последовательного поиска (методы дихотомии или половинного деления, Фибоначчи и золотого сечения) и методы, использующие аппроксимацию функции (методы квадратичной и кубической интерполяции и др.).

Для большинства задач, связанных с поиском оптимальных решений при проектировании таких сложных технических систем, как автомобиль и трактор, методы одномерного поиска практически не применимы, поскольку выходные параметры этих машин, как правило, зависят от множества управляемых параметров. Обычно в этом случае применяются методы многомерного поиска.

В качестве примера многомерного поиска рассмотрим *метод покоординатного спуска* (метод Гаусса — Зейделя). Рассмотрим функцию двух переменных, поскольку в этом случае возможна геометрическая интерпретация рассматриваемого метода.

Таким образом, необходимо решить задачу

$$\min F(\mathbf{X}), \mathbf{X}(x_1, x_2) \in \text{XP}.$$

На рис. 1.10, *а* изображена поверхность, представляющая собой изображение функции  $F(\mathbf{X})$  двух переменных (управляемых параметров)  $x_1$  и  $x_2$ . Проекция этой поверхности на плоскость переменных  $x_1$  и  $x_2$  показана на рис. 1.10, *б*.

В соответствии с алгоритмом, приведенным на рис. 1.9, на плоскости управляемых параметров  $x_1 O x_2$  выберем начальную точку  $\mathbf{X}_0$  с координатами  $(x_{10}; x_{20})$  и определим для нее значение целевой функции  $F_0$ . Затем, в соответствии с указанным алгоритмом, выберем направление поиска. Для этого зафиксируем значение управляемого параметра  $x_2 = x_{20} = \text{const}$ , а значение управляемого параметра  $x_1 = x_{10}$  изменим на величину заранее выбранного шага. Для полученной пары значений управляемых параметров вычислим значение целевой функции. Если полученное значение целевой функции будет больше первоначального, необходимо изменить значение управляемого параметра  $x_{10}$  на величину шага в противоположном направлении и для этой точки снова вычислить значение целевой функции. Направление выбирается с таким расчетом, чтобы значение целевой функции стало меньше первоначального. Далее в выбранном направлении продолжаем перемещение с выбранным шагом до тех пор, пока значение целевой функции уменьшается. При нарушении этого условия зафиксируем

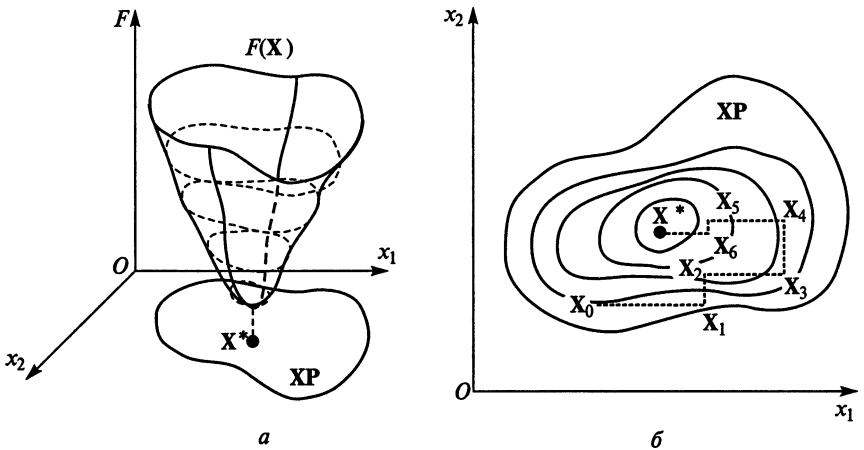


Рис. 1.10. Графическая интерпретация метода покоординатного спуска: *а* — изображение поверхности в виде функции двух переменных; *б* — проекция поверхности на плоскость переменных

ем достигнутое на предыдущем шаге значение управляемого параметра  $x_1$  (точка  $X_1$ ), а затем начинаем перемещение в направлении управляемого параметра  $x_2$  с определенным шагом, пользуясь закономерностями, описанными выше, до точки  $X_2$ , в которой достигнут минимум целевой функции.

После этого зафиксируем значение переменной  $x_2$ , снова возвращаемся к параметру  $x_1$ , и цикл повторяется до тех пор, пока изменение любого управляемого параметра на величину выбранного шага будет приводить к увеличению значения целевой функции. Поиск заканчивается достижением локального минимума целевой функции, соответствующего значениям вектора управляемых параметров  $X^*$ .

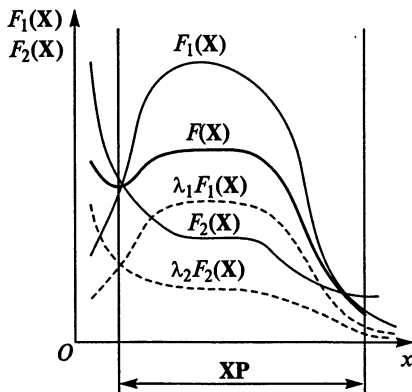


Рис. 1.11. Графическая интерпретация метода свертки

Использование *метода свертки* можно проиллюстрировать следующим примером. Пусть имеются две целевые функции  $F_1(x)$  и  $F_2(x)$  одного управляемого параметра. Предположим, что для каждой из них необходимо найти минимум и значимость каждой из них с точки зрения оптимизации выходных параметров конструкции равнозначна.

В этом случае постановка задачи, в общем виде выраженной формулой (1.1), сводится к следующей:

$$\min F(\mathbf{X}) = \lambda_1 F_1(x) + \lambda_2 F_2(x), \quad x \in \mathbf{XP}.$$

Поскольку значимость целевых функций в процессе оптимизации одинакова, весовые коэффициенты равны. Допустим,  $\lambda_1 = \lambda_2 = 0,5$ . Графическая интерпретация решения этой задачи представлена на рис. 1.11.

Следует иметь в виду, что на практике решение задач проектирования таких сложных объектов, как автомобиль или трактор, связано с нахождением экстремумов целевых функций значительно большего количества управляемых параметров и столь простая графическая интерпретация описанных методов невозможна.

Рассмотрим пример постановки задачи оптимального проектирования независимой подвески на двух поперечных рычагах, упругим элементом которой является круглый торсион. Расчетная схема подвески приведена на рис. 1.12.

Направляющим устройством этой подвески являются нижний 1 и верхний 2 рычаги, связанные между собой поворотной стойкой 3, на которой расположена ось вращения колеса 5. С осью нижнего рычага связан торсион 4 диаметра  $d$ , расположенный вдоль

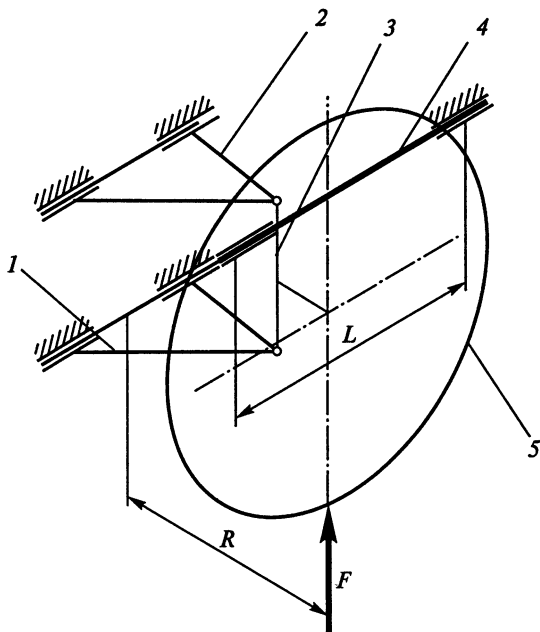


Рис. 1.12. Расчетная схема торсионной подвески:

- 1 — нижний рычаг; 2 — верхний рычаг; 3 — поворотная стойка; 4 — торсион;  
5 — колесо;  $F$  — реакция опорной поверхности;  $L$  — рабочая длина торсиона;  
 $R$  — радиус направляющего устройства

оси транспортного средства. На колесо воздействует нормальная реакция опорной поверхности  $F$ , которая на радиусе направляющего устройства  $R$  создает момент, закручивающий торсион.

Потенциальная энергия деформации торсиона определяется зависимостью

$$U = c\varphi^2/2,$$

где  $c$  — крутильная жесткость торсиона;  $\varphi$  — угол закрутки торсиона.

Жесткость торсиона

$$c = T/\varphi,$$

где  $T$  — крутящий момент, действующий на торсион, определяемый произведением силы  $F$  на радиус  $R$ .

Отсюда следует

$$\varphi = T/c \text{ и } U = T^2/(2c). \quad (1.3)$$

С другой стороны:

$$c = GJ/L = G\pi d^4/(32L), \quad (1.4)$$

где  $G$  — модуль упругости второго рода;  $J$  — полярный момент инерции сечения торсиона ( $J = \pi d^4/32$ );  $L$  — рабочая длина торсиона.

Касательные напряжения, возникающие в материале торсиона, определяются формулой

$$\tau = T/W_k,$$

где  $W_k = \pi d^3/16$  — момент сопротивления кручению сечения торсиона.

Отсюда

$$T = \pi d^3\tau/16. \quad (1.5)$$

Подставляя зависимости (1.4) и (1.5) в (1.3), имеем

$$U = \tau^2\pi d^2L/(16G).$$

Потенциальная энергия деформации торсиона определяет одну из важнейших характеристик подвески — ее энергоемкость. Чем выше энергоемкость подвески, тем менее вероятны ее пробои, что положительно сказывается как на комфортабельности автомобиля, так и на нагруженности элементов подвески и несущего основания автомобиля. Однако условия максимальной энергоемкости подвески недостаточны для того, чтобы обеспечить ее основное назначение — плавность хода автомобиля, которая оценивается частотой собственных колебаний неподрессоренной массы автомобиля на подвеске. При известной неподрессоренной массе автомобиля и радиусе  $R$  для определения частоты собственных колебаний на торсионной подвеске достаточно знать жесткость торсиона. Кроме того, необходимым условием работоспособности

подвески является прочность ее элементов, в данном случае — торсиона.

Таким образом, задачу оптимального проектирования подвески (см. рис. 1.12) можно сформулировать следующим образом.

За целевую функцию принимаем потенциальную энергию деформации торсиона, определяющую энергоемкость подвески, причем ищем максимум целевой функции.

$$\max F(\mathbf{X}) = \tau_{\max}^2 \pi x_1^2 x_2 / (16G),$$

$$\mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} d \\ L \\ R \end{bmatrix}.$$

Необходимо также ввести ограничения по жесткости и прочности торсиона:

$$\tau_{\max} = 16Fx_3 / (\pi x_1^3) \leq [\tau] / n,$$

где  $[\tau]$  — допускаемые напряжения;  $n$  — запас прочности;

$$c_{\min} \leq G\pi x_1^4 / (32x_2) \leq c_{\max}.$$

Минимальные и максимальные значения жесткости торсиона могут быть предварительно выбраны из условия обеспечения плавности хода — обеспечения необходимой частоты собственных колебаний неподрессоренной массы автомобиля на подвеске.

Теперь необходимо ограничить область определения составляющих вектора управляемых параметров  $\mathbf{X}$ .

$$d_{\min} \leq x_1 \leq d_{\max}; \quad L_{\min} \leq x_2 \leq L_{\max}; \quad R_{\min} \leq x_3 \leq R_{\max}.$$

Данные ограничения могут быть заданы проектировщиком исходя из опыта предшествующих разработок и из конструктивных особенностей проектируемого автомобиля.

Указанная задача может быть решена методами нелинейного программирования, поскольку целевая функция и ряд ограничений выражены нелинейными зависимостями.



# МАТЕМАТИЧЕСКИЕ МОДЕЛИ ОБЪЕКТОВ ПРОЕКТИРОВАНИЯ

### 2.1. Общие сведения

Знание возможностей технических и языковых средств и умение ими пользоваться является необходимым условием успешной работы инженера в САПР. Однако для эффективного использования этих средств он должен хорошо ориентироваться в вопросах математического обеспечения САПР, которое определяет внутреннее содержание процедур взаимодействия инженера с ЭВМ. Знание особенностей математических моделей, методов и алгоритмов решения проектных задач необходимо инженеру для постановки задач и правильной формулировки исходных данных и интерпретации получаемых результатов, при принятии решения об использовании тех или иных компонентов математического обеспечения в процессе решения проектных задач.

*Математической моделью (ММ)* называют совокупность математических объектов (чисел, переменных, матриц, множеств и т. п.) и связей между ними, отображающих важнейшие для проектирования свойства технического объекта. Выполнение проектных операций и процедур в САПР основано на оперировании ММ. С их помощью прогнозируются характеристики и оцениваются возможности предложенных вариантов схем и конструкций, проверяется их соответствие предъявляемым требованиям, проводится оптимизация параметров, разрабатывается техническая документация и т. д.

При автоматизации проектирования специфика проектируемых объектов находит свое отражение прежде всего в их ММ. При проектировании таких сложных объектов, как автомобиль или трактор, приходится иметь дело с множеством ММ отдельных агрегатов, узлов, деталей, причем каждый из элементов конструкции требует, как правило, разработки нескольких ММ, описывающих ограниченный круг свойств элемента. Так, относительно простой, на первый взгляд, элемент конструкции, как пневматическое колесо, имеет отдельные ММ для описания его состава, формы, тяговых, упругих и амортизирующих свойств, свойств, определяющих его грузоподъемность, влияние на управляемость и устойчивость движения машины, и т. д.

Несмотря на многообразие ММ, применяемых в САПР автомобиля и трактора, они имеют много общего; в частности, это

относится к классификации, требованиям, принципам и методам создания ММ и использования их в процессе моделирования.

Прежде чем знакомиться с математическими моделями, остановимся вкратце на таких понятиях, как параметры и фазовые переменные объекта (модели).

*Параметр* — величина, характеризующая свойства или режим работы объекта. Среди параметров объекта проектирования следует выделить показатели эффективности, являющиеся количественной оценкой степени соответствия объекта его целевому назначению. Показатели эффективности делят на показатели надежности, стоимости, массы, габаритных размеров, точности. В зависимости от конкретных условий и типов объектов те или иные из показателей имеют решающее значение. Термин «показатель эффективности» чаще всего используется на высших иерархических уровнях проектирования применительно к сложным системам.

Различают выходные, входные и внутренние параметры.

*Выходные параметры* — показатели качества, по которым можно судить о правильности функционирования системы, т.е. это понятие аналогично понятию «показатель эффективности», но применяется к системам на любом иерархическом уровне. Выходные параметры зависят как от свойств элементов, так и от особенностей связи элементов друг с другом, определяемой структурой (конфигурацией) системы. Каждый новый способ связи задает новую структуру и приводит к качественным изменениям в работе системы. К таким же изменениям приводит и смена типа какого-либо элемента, если новый тип качественно отличается от предыдущего. Если структура системы определена, ее выходные параметры зависят только от параметров элементов и параметров внешних условий.

*Внутренние параметры* — параметры элементов.

*Внешние параметры* — параметры внешней по отношению к объекту среды, оказывающие влияние на его функционирование.

Иными словами, на каждом иерархическом уровне выходные параметры характеризуют свойства системы, а внутренние — свойства элементов. Следует отметить, что при переходе к новому уровню рассмотрения внутренние параметры могут стать выходными и наоборот. Например, геометрические размеры упругого элемента подвески автомобиля или трактора являются внутренними параметрами подвески, а ее жесткость — выходным параметром. В свою очередь жесткость подвески — внутренний параметр всей колебательной системы. Типичными примерами внешних параметров могут служить параметры нагрузки, влажность и температура окружающей среды и т.п.

Введем обозначения:  $\mathbf{Y} = (y_1, y_2, \dots, y_m)$  — вектор выходных параметров некоторой системы;  $\mathbf{X} = (x_1, x_2, \dots, x_n)$  — вектор внутренних параметров;  $\mathbf{Q} = (q_1, q_2, \dots, q_k)$  — вектор внешних параметров.

Тогда

$$Y = F(X, Q),$$

где вид функциональной зависимости  $F$  определяется структурой системы.

В большинстве случаев связь между выходными, внутренними и внешними параметрами известна не в виде явной зависимости  $Y$  от  $X$  и  $Q$ , а задается в алгоритмической форме, например через числовое решение системы уравнений.

Уравнения, решение которых требуется для определения выходных параметров, обычно являются математическим описанием функционирования проектируемого объекта. В этих уравнениях независимыми переменными могут быть время, частота, пространственные координаты, а зависимыми переменными — *фазовые переменные* (величины, характеризующие состояние объекта и поэтому называемые также *переменными состояния*). Примерами фазовых переменных могут служить скорости, силы, напряжения и деформации в механических системах, давления и расходы в гидравлических системах, напряжения, токи и заряды в электрических системах и т.д. Вектор фазовых переменных задает точку в пространстве, называемом *фазовым пространством*.

**Классификация математических моделей.** В зависимости от характера отображаемых свойств объекта модели делятся на структурные и функциональные.

В процедурах, относящихся к процессу проектирования, преобладает использование математических моделей, отражающих только структурные свойства объекта, например его геометрическую форму, размеры, взаимное расположение элементов в пространстве. Такие модели называют *структурными*.

Различают структурные ММ топологические и геометрические. В топологических моделях отображаются состав и взаимосвязь элементов объекта. Их часто используют для описания объектов, состоящих из большого числа элементов, при решении задач привязки элементов к определенным пространственным позициям (например, задача компоновки машины) или к относительным моментам времени (например, при разработке технологического процесса). Топологические модели часто имеют форму графов, таблиц, списков и т.п. В геометрических моделях помимо сведений о взаимном расположении элементов объекта содержатся сведения о форме элементов.

Такие модели могут выражаться совокупностью уравнений линий и поверхностей, алгебраических соотношений, описывающих области, составляющие тело объекта. При описании конструкции, состоящей из типовых элементов, используются графы и списки.

В проектных процедурах, связанных с функциональным аспектом проектирования, как правило, используются ММ, отражаю-

щие закономерности процессов функционирования объектов. Такие модели называют *функциональными*. Типичная функциональная модель представляет собой систему уравнений, описывающих механические, гидравлические, пневматические, электрические, тепловые процессы. Поскольку характер функционирования объекта в большинстве случаев невозможно описать без учета его структуры, в функциональных ММ отражаются также и структурные свойства объекта. Функциональные модели более сложные по сравнению со структурными и считаются основным типом моделей в САПР.

Существует классификация математических моделей в зависимости от степени детализации описываемых свойств и процессов, протекающих в объекте. Рассмотрим ее применительно к функциональным ММ.

Блочно-иерархический подход к проектированию технических объектов включает в качестве своей основы и иерархию математических моделей. На каждом иерархическом уровне используются свои ММ, сложность которых согласована с возможностями анализа. Деление моделей по иерархическим уровням происходит по степени детализации описываемых свойств и процессов, протекающих в объекте. При этом на каждом иерархическом уровне используют свои понятия «система» и «элементы». Так, система  $k$ -го уровня рассматривается как элемент на соседнем более высоком  $(k-1)$ -м уровне абстрагирования.

Структуру объекта на любом иерархическом уровне можно представить в виде совокупности элементов и связей между ними. Свойства каждого элемента описываются математической *моделью элемента*. Она представляет собой соотношения, связывающие внешние по отношению к элементу фазовые переменные. Некоторые подмножества элементов с их связями можно по каким-либо общим признакам объединить в группы или *блоки*. Математическая модель, полученная непосредственным объединением моделей элементов блока в общую систему уравнений, называется *полной моделью* блока. Характерной ее особенностью является присутствие в ней вектора внутренних фазовых переменных, т. е. она описывает и состояние каждого из элементов блока.

При большом количестве элементов порядок системы уравнений становится чрезмерно большим и требуются упрощения. При переходе к более высокому иерархическому уровню упрощения основаны на исключении из модели вектора внутренних переменных. Такая модель называется *макромоделью*. Она уже не описывает процессы внутри блока, а характеризует только процессы взаимодействия данного блока с другими в составе системы блоков. Понятие макромодели имеет существенное значение в блочно-иерархическом подходе к проектированию. Замена полных моделей блоков их макромоделями позволяет перейти на более

высокий иерархический уровень, где блок низшего уровня выступает уже в качестве элемента нового, укрупненного блока (или системы полностью).

Существует также понятие многоуровневых моделей, когда одни блоки системы описываются полными моделями, а другие — макромоделями.

В зависимости от сложности объекта при его проектировании используют большее или меньшее число уровней абстракции. Число используемых иерархических уровней при проектировании конкретных объектов зависит от традиций предприятия, принятой организации САПР, возможностей используемого математического и программного обеспечения. Увеличение числа уровней позволяет использовать более простые ММ на каждом из них, однако усложняет согласование результатов, полученных на различных уровнях.

В иерархии функциональных моделей для большинства проектируемых сложных объектов объединение уровней, родственных по характеру используемого математического аппарата, приводит к образованию трех укрупненных уровней: микро-, макро- и метауровень.

На микроуровне используют математические модели, описывающие физическое состояние и процессы в сплошных средах. Фазовые переменные являются в данном случае функциями нескольких независимых переменных, таких, как пространственные координаты и время, при этом и пространство, и время непрерывны.

Примерами таких моделей служат дифференциальные уравнения в частных производных — уравнения упругости, электродинамики, теплопроводности, гидродинамики, газовой динамики, которые описывают напряженно-деформированное состояние деталей механических конструкций, поля электрического потенциала и температуры и т. п.

К типичным фазовым переменным на микроуровне относятся механические напряжения и деформации, давления, температуры, электрические потенциалы, концентрации частиц, плотности токов. В связи с учетом характера воздействий и фазовых переменных, распределенных в пространстве, эти модели называют *распределенными*. Подобные модели используются, например, для определения распределения напряжений в деталях конструкции, распределения температуры по поверхности и внутри накладок фрикционного сцепления в процессе его включения, исследования процесса взаимодействия пневматического колеса с дорогой.

Анализ распределенных моделей сводится к решению краевых задач математической физики и представляет значительные трудности вычислительного характера. Использование их ограничива-

ется случаями объектов с малым числом участков. Усложнение задачи при увеличении протяженности пространственных и временных областей приводит к необходимости перехода к следующему иерархическому уровню — макроуровню.

На макроуровне производится дискретизация пространства с выделением в качестве элементов отдельных деталей. Такая дискретизация означает переход от распределенных моделей к *сосредоточенным*, при этом из числа независимых переменных исключают пространственные координаты. Элементами этого уровня являются объекты, которые на микроуровне рассматривались как системы (например, валы, пружины, элементы сопротивления). Параметры этих элементов, будучи на микроуровне выходными, становятся внутренними. Примерами выходных параметров макроуровня являются касательная сила тяги колеса, время и работа буксования фрикционного сцепления, уровень нагрузки в отдельных элементах конструкции.

Математические модели на макроуровне представляют собой системы обыкновенных дифференциальных уравнений, которые в частных случаях решения статических задач превращаются в системы алгебраических или трансцендентных уравнений. Для их получения и решения используют соответствующие численные методы. В качестве фазовых переменных фигурируют силы, скорости, температуры, расходы, электрические напряжения, токи и т.д. Они характеризуют проявления внешних свойств элементов при их взаимодействии между собой и внешней средой.

С увеличением числа элементов системы возможности решения задач с использованием ММ макроуровня резко сужаются. В этом случае целесообразен переход к следующему, более высокому иерархическому уровню.

На метауровне с помощью дальнейшего абстрагирования от характера физических процессов удается получить приемлемое по сложности описание процессов, протекающих в проектируемых объектах. Математические модели на метауровне — системы обыкновенных дифференциальных уравнений, системы алгебраических уравнений, системы логических уравнений, имитационные модели систем массового обслуживания. Здесь роль элементов и внутренних параметров выполняют системы и выходные параметры предыдущего иерархического уровня. Так, элементами автомобиля или трактора на метауровне можно считать двигатель, коробку передач, ведущий мост, колесо. Моделирование на метауровне позволяет выполнить тягово-динамический расчет автомобиля, тяговый расчет трактора, решить вопросы компоновки машины, выполнить основные расчеты на прочность и сопротивление усталости деталей.

Математические модели различают также в зависимости от формы их представления.

*Инвариантная форма* — запись соотношений модели с помощью традиционного математического языка безотносительно к методу решения уравнений модели.

*Алгоритмическая форма* — запись соотношений модели и выбранного численного метода решения в форме алгоритма. Вычисление значений искомых величин производится путем решения систем уравнений.

*Аналитическая форма* — запись модели в виде результата аналитического решения исходных уравнений модели. При этом модели в аналитической форме обычно представляют собой явные выражения выходных параметров как функций внутренних и внешних параметров. С точки зрения удобства реализации на ЭВМ они выгодно отличаются от других, однако их сложно и не всегда возможно получить.

*Схемная форма* (называемая также *графической*) — представление модели на некотором графическом языке, например на языке графов, эквивалентных схем, диаграмм и т. п. Графические формы удобны для восприятия человеком. Использование таких форм возможно при наличии правил однозначного истолкования элементов чертежей и их перевода на язык инвариантных или алгоритмических форм.

Математические модели в алгоритмической и аналитической формах называют соответственно *алгоритмическими* и *аналитическими*. Среди алгоритмических моделей важный класс составляют *имитационные модели*, предназначенные для имитации происходящих в объекте процессов при задании различных зависимостей входных воздействий от времени. Собственно имитацию названных процессов называют *имитационным моделированием*. Результат имитационного моделирования — зависимости фазовых переменных в избранных элементах системы от времени. Примерами имитационных моделей могут служить модели разгона и торможения автомобиля и трактора, входа автомобиля в поворот, переезда транспортного средства через препятствие, модель рабочего цикла бульдозера и т. д.

**Требования к математическим моделям.** Основными требованиями, предъявляемыми к ММ, являются требования адекватности, универсальности и экономичности.

*Адекватность.* Модель считается адекватной, если она отражает заданные свойства объекта с приемлемой точностью. Точность определяется как степень совпадения предсказанных с помощью модели значений выходных параметров объекта с истинными значениями этих параметров.

Точность модели оценивается относительной погрешностью

$$\varepsilon = \frac{y_m - y_{\text{ист}}}{y_m},$$

где  $y_m$  — выходной параметр, рассчитанный с помощью модели;  $y_{\text{ист}}$  — тот же выходной параметр, имеющий место в моделируемом объекте.

Количественная оценка точности модели в большинстве случаев вызывает затруднения по ряду причин.

Во-первых, реальные объекты и их модели характеризуются не одним, а несколькими выходными параметрами, поэтому для возможности сопоставления моделей друг с другом используют сведение векторной оценки точности к скалярной.

Пусть объект характеризуется  $m$  выходными параметрами. Относительные погрешности  $\varepsilon_j$  по каждому из них будут различны. В связи с этим для общей оценки погрешности модели  $\varepsilon_m$  по совокупности учитываемых выходных параметров обычно используют одну из норм вектора  $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_m)$ :

$$\varepsilon_m = \max_{j=1, m} |\varepsilon_j| \quad \text{или} \quad \varepsilon_m = \sqrt{\sum_{j=1}^m \varepsilon_j^2}.$$

Во-вторых, так как характер проявления свойств объекта зависит от особенностей его взаимодействия с внешней средой и другими объектами системы, то и показатели точности отображения этих свойств в модели будут зависеть от условий функционирования объекта. В результате оценка точности становится неоднозначной.

Кроме того, значения  $y_{\text{ист}}$  обычно отождествляют с экспериментально полученными. Однако погрешности эксперимента во многих случаях оказываются соизмеримыми с погрешностью ММ или даже превышают их. Чтобы уменьшить влияние этих факторов на результаты оценки погрешности модели, сравнение моделей проводят по результатам их использования в некоторых стандартных ситуациях, соответствующих наиболее характерным условиям работы реального объекта, причем число этих ситуаций стараются максимально ограничить.

Конечно, ММ должна иметь высокую точность отображения свойств объекта в конкретных условиях его эксплуатации. Однако целью ее создания является возможность использования модели для исследования поведения объекта в некоторых областях изменения его внутренних и внешних параметров. Поскольку выходные параметры системы являются функциями внешних  $\mathbf{Q} = (q_1, q_2, \dots, q_k)$  и внутренних  $\mathbf{X} = (x_1, x_2, \dots, x_n)$  ее параметров, погрешность  $\varepsilon_m$  зависит от  $\mathbf{Q}$  и  $\mathbf{X}$ . Обычно внутренние параметры ММ выбираются из условия минимизации погрешности  $\varepsilon_m$  в некоторой точке  $\mathbf{Q}_{\text{ном}}$  пространства внешних параметров, и величина погрешности модели становится функцией  $\mathbf{Q}$ .

Если задаться предельно допустимой погрешностью  $\delta$ , то в пространстве внешних параметров можно выделить область, в которой выполняется условие  $\varepsilon_m < \delta$ .



Такую область называют областью адекватности (ОА) модели. Возможно введение индивидуальных предельных значений  $\delta_j$  для каждого выходного параметра и определение ОА как области, в которой одновременно выполняются все  $m$  условий вида  $|\varepsilon_j| \leq \delta_j$ .

Графическая иллюстрация ОА для двумерного пространства внешних параметров  $\mathbf{Q} = (q_1, q_2)$  представлена на рис. 2.1, где область адекватности ограничена линиями  $j = 1, j = 2$  и  $j = 3$ , задаваемыми уравнениями

$$|\varepsilon_j(\mathbf{Q})| = \delta_j, j = 1, 2, 3.$$

Определение областей адекватности для конкретных моделей — сложная процедура, требующая больших вычислительных затрат. Эти затраты и трудности представления ОА быстро растут с увеличением размерности пространства внешних параметров. Определение ОА — более трудная задача, чем, например, задача параметрической оптимизации, поэтому для моделей вновь проектируемых объектов ОА не рассчитывают. Однако для моделей унифицированных элементов расчет ОА становится оправданным в связи с однократностью определения и многократностью их использования при проектировании различных систем. Знание ОА позволяет правильно выбирать модели элементов из числа имеющихся и тем самым повышать достоверность результатов машинных расчетов.

В общем случае ОА (см. рис. 2.1) может иметь произвольную форму и неудобна в использовании, поэтому на практике вместо истинных ОА применяют те или иные их аппроксимации. Наиболее просто представляются и используются сведения об областях, имеющих форму гиперпараллелепипеда, который задается  $p$  двусторонними неравенствами:

$$q_k' < q_k < q_k'', k = 1, 2, \dots, p,$$

где  $p$  — размерность пространства внешних параметров.

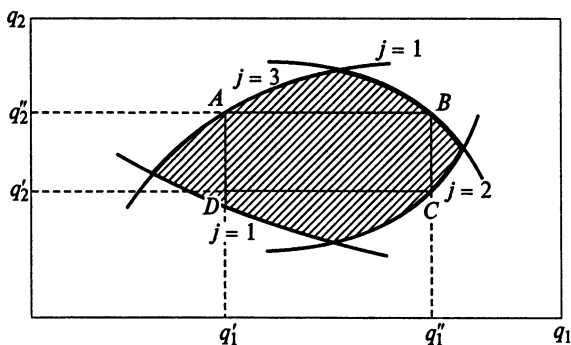


Рис. 2.1. Пример области адекватности

В библиотеку моделей элементов наряду с алгоритмом, реализующим модель, и номинальными значениями параметров должны включаться граничные значения внешних параметров  $q'_k$  и  $q''_k$ , задающие ОА. На рис. 2.1 область адекватности аппроксимирована параллелепипедом  $ABCD$ . Такое представление удобно для двумерных случаев. Возможно использование и других аппроксимаций ОА, например областей с линеаризованными границами в виде участков гиперплоскостей, областей в форме гиперсфер и т. п.

**Универсальность.** Степень универсальности ММ характеризует полноту отображения в них свойств реального объекта и определяется возможностью использования модели для анализа более или менее многочисленной группы однотипных объектов, а также числом доступных для анализа режимов функционирования. Использование машинных методов проектирования станет неудобным, если в процессе анализа объекта при каждом изменении режима функционирования пользователю потребуется смена ММ.

Универсальность модели в первую очередь зависит от числа и состава учитываемых в модели внешних и выходных параметров. Увеличение их расширяет применимость модели, но существенно усложняет ее разработку. Тем не менее степень использования универсальных ММ в САПР является одним из основных критериев ее выбора.

**Экономичность.** Экономичность ММ характеризуется затратами вычислительных ресурсов для ее реализации, а именно затратами машинного времени и памяти. Общие затраты на выполнение в САПР какой-либо проектной процедуры зависят как от особенностей выбранных моделей, так и от методов решения.

В большинстве случаев при реализации численного метода происходят многократные обращения к модели элемента, входящего в состав моделируемого объекта. Тогда удобно экономичность модели элемента характеризовать затратами машинного времени, получающимися при обращении к модели, а число обращений к модели должно учитываться при оценке экономичности метода решения.

Экономичность модели по затратам памяти оценивается объемом оперативной памяти, необходимой для реализации модели.

Требования широких областей адекватности, высокой степени универсальности, с одной стороны, и высокой экономичности, с другой, являются противоречивыми. Наилучшее компромиссное удовлетворение этих требований оказывается неодинаковым в различных применениях. Данное обстоятельство обуславливает использование в САПР многих моделей для объектов одного и того же типа.

**Методы получения математических моделей.** Математическая модель технического объекта в САПР обычно создается пользова-

телем на основе уже разработанных и имеющихся в библиотеке ММ элементов и соответствующего программного обеспечения.

Получение моделей элементов (моделирование элементов) в общем случае — процедура неформализованная. Основные решения, касающиеся выбора вида математических соотношений, характера используемых переменных и параметров, принимает разработчик модели. В то же время такие операции, как расчет численных значений параметров модели, определение областей адекватности и др., алгоритмизированы и решаются на ЭВМ. Поэтому моделирование элементов обычно выполняется специалистами конкретных технических областей с помощью традиционных средств экспериментальных исследований и средств САПР.

Методы получения функциональных моделей элементов делят на теоретические и экспериментальные.

*Теоретические методы* основаны на изучении физических закономерностей протекающих в объекте процессов, определении соответствующего этим закономерностям математического описания, обосновании и принятии упрощающих предположений, выполнении необходимых выкладок и приведении результата к принятой форме представления модели. Основу получаемых моделей обычно составляют системы уравнений, отражающих зависимости фазовых переменных. Такие модели чаще всего относятся к алгоритмическим и адекватны в сравнительно широких диапазонах изменения переменных.

*Экспериментальные методы* основаны на использовании экспериментально полученных зависимостей между параметрами и фазовыми переменными объекта. Эксперименты при этом могут проводиться на самих объектах, на их физических моделях (макетах и стендах) или с использованием их полных ММ. Для целей моделирования используются пассивные и активные эксперименты. При пассивных экспериментах условия опыта остаются постоянными. В случае использования активного целенаправленного эксперимента опыты проводятся по заранее разработанному плану, определяющему количество опытов и значения факторов в каждом опыте. В зависимости от методов планирования преимущества активных экспериментов перед пассивными могут выражаться в сокращении сроков разработки модели и в получении оптимального положения области ее адекватности.

В процессе преобразования экспериментальных данных в ММ возможны их аппроксимация, усреднение, статистическая обработка. Последнее характерно при постановке пассивных экспериментов, когда связь между выходными и внешними параметрами носит не функциональный, а статистический характер. Для получения модели в такой ситуации часто применяют регрессионный анализ.

Экспериментальные методы получения ММ удобны для моделирования безынерционных объектов с относительно гладкими

зависимостями между переменными. Результатом применения этих методов становятся модели, имеющие частный характер.

Несмотря на неформальный характер большинства операций, используемых при разработке моделей элементов различных объектов, имеется ряд общих положений и приемов. Достаточно общий характер имеют методика макромоделирования, математические методы планирования экспериментов, а также алгоритмы формализуемых операций расчета численных значений параметров и определения ОА.

В общем случае методика получения ММ элементов объекта включает в себя этапы, приведенные ниже.

1. Выбор свойств объекта, которые подлежат отражению в модели. Этот выбор основан на анализе возможных применений модели и определяет степень ее универсальности.

2. Сбор априорной информации о свойствах моделируемого объекта. Источниками собираемых сведений могут служить знания и опыт инженера, научно-техническая литература, прежде всего справочная, ММ и результаты эксплуатации существующих аналогичных объектов и т. п.

3. Синтез структуры ММ — получение общего вида уравнений модели без конкретизации численных значений фигурирующих в них параметров. Часто проектировщику модели удобнее оперировать не уравнениями, а эквивалентными схемами, с помощью которых инженеру проще устанавливать физический смысл различных элементов ММ.

4. Определение численных значений параметров модели. Данная задача решается как задача минимизации погрешности модели заданной структуры. Возможны следующие приемы выполнения этого этапа:

- использование специфических расчетных соотношений с учетом собранных на этапе 2 сведений;
- решение экстремальной задачи, в которой в качестве целевой функции выбирается степень совпадения известных значений выходных параметров объекта с результатами использования модели, а управляемыми параметрами являются параметры модели;
- проведение экспериментов и обработка полученных результатов.

5. Оценка точности полученной модели и определение области ее адекватности. Для оценки точности должны использоваться значения  $u_{\text{ист}}$ , которые не фигурировали в задаче минимизации погрешности. При неудовлетворительных результатах оценки выполняют итерационное приближение к желаемому результату повторением этапов 3—5.

6. Представление полученной модели в форме, принятой в используемой библиотеке моделей.

Наличие в методике макро моделирования эвристических и формальных операций обуславливает целесообразность разработки моделей элементов в диалоговом режиме с ЭВМ. Язык взаимодействия человека с ЭВМ должен позволять оперативный ввод исходной информации о структуре модели, об известных характеристиках и параметрах объекта, о плане экспериментов. Диалоговое моделирование должно иметь программное обеспечение, в котором реализованы алгоритмы статистической обработки результатов экспериментов, расчета выходных параметров эталонных моделей и создаваемых макромоделей, в том числе расчета параметров по методам планирования экспериментов и регрессионного анализа, алгоритмы методов поиска экстремума, расчета ОА и др. Пользователь, разрабатывающий модель, должен иметь возможность менять уравнения модели, задавать их в аналитической, схемной или табличной формах, обращаться к нужным подпрограммам и тем самым оценивать результаты своих действий, приближаясь к получению модели с требуемыми свойствами.

Не следует, однако, считать, что необходимость использования при получении моделей элементов в основном неформальных методов снижает уровень автоматизации выполнения большинства проектных процедур. Дело в том, что при моделировании любого технического объекта используется типовый набор элементов, причем количество типов элементов в объектах определенного назначения, как правило, существенно меньше количества самих элементов. Более того, эти типы элементов повторяются во многих проектируемых объектах не только аналогичного, но и иного назначения, и даже в объектах другой физической структуры. Поэтому получение моделей элементов требуется однократно при многократном их использовании в моделях различных объектов; это позволяет для каждого типа элементов тщательно отработать ряд моделей, различающихся показателями точности, экономичности и универсальности. Модели всех унифицированных типов элементов в САПР данной предметной области заносят в библиотеку моделей элементов. При эксплуатации САПР повседневно решаемыми задачами становятся задачи формирования и анализа моделей систем объекта.

В отличие от процедур получения ММ элементов, процедуры получения модели блока или всего объекта могут быть полностью формализованы. Используемые при этом методы инвариантны по отношению ко многим областям техники. Примерами могут служить методы перемещений в механике, методы узловых потенциалов в электротехнике и др.

При моделировании на метауровне часто принимаются допущения о однонаправленности распространения внешних воздействий от входов к выходам элементов, т.е. предполагается, что изменение состояния элемента-нагрузки не передается к элемен-

ту-источнику и все внешние связи разделяются на входные и выходные. На макроуровне подобные допущения чаще всего неправомерны. Используемые здесь методы более сложны в реализации, их инвариантность обусловлена наличием аналогий физических систем, поэтому такие методы называют *методами моделирования на основе прямой аналогии*.

## **2.2. Преобразование математических моделей в процессе получения рабочих программ анализа**

Процедура анализа в САПР заключается в определении свойств объекта, отражаемых в его ММ, на основе решения уравнений. Реализация ММ на ЭВМ подразумевает выбор численного метода решения уравнений и преобразование их в соответствии с особенностями выбранного метода. Конечная цель преобразований — получение рабочей программы в виде последовательности элементарных действий над числами (арифметических и логических операций).

Все преобразования исходной ММ в последовательность элементарных действий ЭВМ выполняются автоматически по специальным программам, создаваемым разработчиком САПР. Пользователь САПР должен лишь указать, какие программы из имеющихся он хочет использовать. Однако ему важно знать методы решения уравнений, прежде всего для правильного выбора прикладных программ в конкретных ситуациях. Обычно в САПР имеется несколько программ одинакового целевого назначения, но с разными реализованными в них численными методами. Программы различаются затратами машинного времени и памяти, вероятностью получения решения, точностью результатов. Значения перечисленных показателей зависят от многих факторов. Их успешный прогноз при анализе конкретного заданного объекта основан на знании особенностей численных методов. Кроме того, знание численных методов позволяет инженеру в необходимых случаях составлять свои оригинальные программы решения проектных задач. При этом для инженера важно не абстрактное знание численных методов, а знание показателей эффективности и резервов повышения эффективности методов в условиях их применения к задачам проектирования.

Подробно эти методы рассматриваются при изучении дисциплин «Математика» и «Информатика». Здесь остановимся на них лишь вкратце.

Процесс преобразования ММ, относящихся к различным иерархическим уровням, иллюстрирует рис. 2.2.

Ветви 1 соответствует постановка задачи, относящейся к микроуровню. Описание объекта в данном случае обычно сводится к

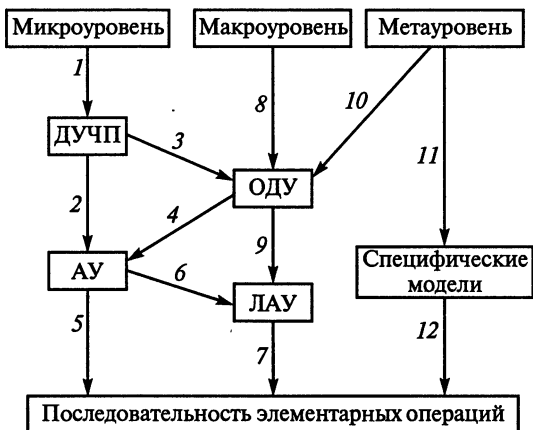


Рис. 2.2. Преобразование математической модели

составлению системы дифференциальных уравнений в частных производных (ДУЧП), численные методы решения которых основаны на дискретизации переменных и алгебраизации задачи. Дискретизация заключается в замене переменных конечным множеством их значений в заданных пространственном и временном интервалах, алгебраизация — в замене производных алгебраическими соотношениями.

Используют два основных подхода к дискретизации и алгебраизации задачи, составляющих сущность методов конечных разностей и конечных элементов.

Если ДУЧП стационарные (описывают статическое состояние и время не фигурирует в качестве независимой переменной), дискретизация и алгебраизация преобразуют систему ДУЧП в систему алгебраических уравнений (АУ) — ветвь 2 (см. рис. 2.2). Если система ДУЧП нестационарная, дискретизацию и алгебраизацию можно представить состоящими из двух этапов: устранение производных по пространственным координатам (ветвь 3), в результате чего получается система обыкновенных дифференциальных уравнений (ОДУ), и устранение производных по времени (ветвь 4 или ветвь 9).

Для решения ОДУ при заданных начальных условиях разработано большое количество численных методов. Как правило, эти методы являются пошаговыми: на каждом шаге интегрирования производится алгебраизация уравнений с помощью аппроксимирующих выражений, связывающих производные переменных по времени в некоторой точке  $t_k$  со значениями переменных в этой же точке и в одной или нескольких соседних точках. Каждый метод использует свое аппроксимирующее выражение и алгоритм его вычисления.

С точки зрения пользователя методы отличаются степенью универсальности, скоростью вычислений и требованиями к объему оперативной памяти. В САПР широко используются методы Гира, Адамса и Рунге — Кутта.

Алгебраические уравнения в общем случае нелинейные. Если нелинейность несущественная, уравнения предварительно линеаризуют (см. рис. 2.2, ветвь 6). Основу решения системы нелинейных алгебраических уравнений (ветвь 5) составляют итерационные методы.

Различия между методами в основном заключаются в способе вычисления поправки на каждом шаге итерации. Необходимо, чтобы итерационный процесс был сходящимся к решению задачи. Однако сходимость имеет место не всегда; для каждого метода существуют свои условия сходимости, поэтому важной характеристикой метода является его надежность, оцениваемая как вероятность получения решения с заданной точностью. Очевидно, что такая вероятность зависит не только от метода, но и от особенностей решаемой системы уравнений. Сравнение методов между собой следует производить также с учетом объема требующихся вычислений и затрат машинной памяти.

В САПР для решения систем нелинейных алгебраических уравнений обычно применяют метод Ньютона, как обладающий наибольшей эффективностью по показателю затрат машинного времени. Основным его недостатком является то, что сходимость к решению имеется не всегда, причем заранее предсказать ее наличие или отсутствие довольно сложно. Стремление повысить надежность метода привело к появлению ряда его модификаций. В случаях большой размерности задач при нехватке емкости оперативной памяти вместо метода Ньютона применяют метод последовательной верхней релаксации, в отдельных случаях — методы Якоби и Зейделя.

Ветви 7 (см. рис. 2.2) соответствует решение систем линейных алгебраических уравнений (ЛАУ). Методы простой итерации, Якоби, Зейделя и др. используются редко из-за малой скорости сходимости. Наиболее часто применяется метод Гаусса и его разновидности.

Ветви 8 соответствует преобразование исходного описания задачи, относящейся к макроуровню, в систему ОДУ с начальными условиями. Если это система нелинейных ОДУ, дальнейшее преобразование происходит по ветвям 4, 6, 7 или 4, 5; если это система линейных ОДУ — осуществляется переход к системе ЛАУ (ветвь 9).

Для анализа объектов на метауровне осуществляют либо переход к системе ОДУ (ветвь 10), либо переход к системам логических уравнений, моделям массового обслуживания или аналитическим моделям, отображающим упрощенно технико-экономи-



ческие показатели объекта (ветвь 11). Сведение этих форм моделей в последовательность элементарных вычислительных операций (ветвь 12) не вызывает затруднений.

### 2.3. Математические модели объектов на макроуровне

Основными методами создания ММ на макроуровне являются инвариантные методы. Они предполагают разбиение всей системы на отдельные элементы, описание свойств каждого элемента на уровне взаимодействия их между собой с помощью ММ и последующее объединение их в единую структуру на основе связей между однотипными фазовыми переменными. Обычно процедура выделения элементов выполняется человеком, а составление общей системы уравнений для полученной структуры объекта возлагается на ЭВМ.

Математические модели элементов получают одним из способов, рассмотренных выше. Уравнения, входящие в ММ элементов, называются *компонентными*. Они отражают законы функционирования элемента и связывают, как правило, разнородные фазовые переменные, относящиеся к этому элементу. Так, уравнение второго закона Ньютона связывает силу и ускорение. Уравнения могут быть алгебраическими или дифференциальными, линейными или нелинейными.

Для объединения элементов в систему используются *топологические уравнения*. Они отражают способ связи элементов между собой в составе системы. Топологические уравнения могут выражать законы сохранения, условия неразрывности, равновесия и т. п. Например, уравнения равновесия (принцип Д'Аламбера) устанавливают, что сумма сил (моментов), действующих на тело с учетом инерционных, равна нулю. Топологические уравнения связывают однотипные фазовые переменные, относящиеся к разным

элементам системы. Их получают на основе сведений о структуре системы.

Рассмотрим пример одномассовой колебательной системы. Пусть требуется, например, определить вертикальные ускорения на месте водителя транспортного средства, если известен закон вертикального перемещения кузова (остова) в месте установки подпрессоренного сиденья. Задача может быть решена с использованием одномассовой схемы динамической системы, показанной на рис. 2.3. Здесь масса сиденья и водителя  $m$  выделена в

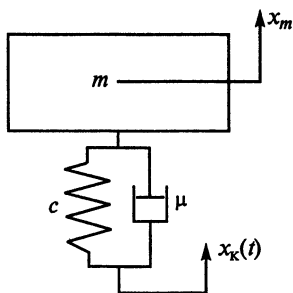


Рис. 2.3. Схема одномассовой динамической системы

отдельный инерционный элемент, связанный с внешней средой посредством двух элементов, учитывающих упругие  $c$  и демпфирующие  $\mu$  свойства сиденья. Рассмотрим движение системы вдоль вертикальной оси  $x$ , совершающей перемещение в горизонтальном направлении совместно с кузовом (остовом).

Фазовыми переменными для инерционного элемента являются сила инерции  $F_m$  и перемещение  $x_m$ , а компонентное уравнение имеет вид

$$F_m = md^2x_m/dt^2,$$

где  $t$  — текущее время (независимая переменная).

Для упругого элемента фазовыми переменными являются сила упругости  $F_c$  и деформация  $\Delta$ , а компонентное уравнение имеет вид

$$F_c = c\Delta.$$

Здесь внутренний параметр  $c$  — жесткость упругого элемента.

Фазовыми переменными элемента сопротивления будут сила сопротивления амортизатора  $F_R$  и скорость деформации  $d\Delta/dt$ , а внутренним параметром — коэффициент демпфирования  $\mu$ . Компонентное уравнение имеет вид

$$F_R = \mu d\Delta/dt.$$

Источником колебаний системы являются вертикальное перемещение кузова  $x_k$  (фазовая переменная источника), закон изменения которого известен:  $x_k = x_k(t)$ .

Для получения ММ объекта необходимо полученные компонентные уравнения дополнить топологическими. Согласно схеме динамической системы (см. рис. 2.3) и принципу Д'Аламбера

$$F_m + F_c + F_R = 0.$$

Из уравнения непрерывности

$$\Delta = x_k - x_m.$$

Решение системы этих уравнений при выбранных значениях коэффициентов и законе перемещения кузова позволяет определить вертикальное ускорение на месте водителя  $a_m = d^2x_m/dt^2$  для любого момента времени.

В качестве фазовой переменной в механических системах часто используется не перемещение, а скорость, поэтому компонентные и топологические уравнения рассматриваемой системы могут иметь несколько иной вид.

В используемых САПР методах формирования ММ объектов принято моделируемую систему представлять в виде совокупности физически однородных подсистем. Каждая из них описывает процессы определенной физической природы, например механи-

ческие, гидравлические, тепловые, электрические. Важно, что форма компонентных и топологических уравнений одинакова для большинства из них, что объясняется наличием аналогий между разнородными физическими подсистемами. Подобные аналогии замечены давно и широко используются в науке и технике для анализа объектов. Установление аналогий важно для эффективного использования САПР, так как позволяет использовать одни и те же математические методы для решения многих проектных задач с разным физическим содержанием. Аналогии дают возможность повысить степень универсальности многих элементов САПР, в частности программного обеспечения. Важно это и с методологической точки зрения, так как обуславливает возможность изложения вопросов моделирования различных технических объектов с единых позиций.

Использование аналогий предполагает, что для описания состояния одной подсистемы достаточно использовать фазовые переменные двух типов, а в большинстве технических объектов можно выделить три типа простейших элементов: элемент диссипации, преобразующий энергию соответствующего данной подсистеме вида в теплоту, и элементы, накапливающие потенциальную или кинетическую энергию.

Аналогию компонентных и топологических уравнений удобно проводить на основе электрической подсистемы. Рассмотрим использование аналогий на примерах подсистем, наиболее характерных для автомобиля и трактора — механической, гидравлической (пневматической), тепловой и электрической.

**Электрическая подсистема.** Типовыми простейшими элементами электрической подсистемы являются электрическое сопротивление  $R$ , электрическая емкость  $C$  и электрическая индуктивность  $L$ . При описании их функционирования используются фазовые переменные типа потока (сила тока  $I$ ) и типа потенциала (напряжение  $U$ ). Значение напряжения на этих элементах совпадает с разностью значений электрического потенциала на концах элементов, напряжение на элементе источника тока равно его ЭДС.

Согласно основным законам электротехники компонентные уравнения этих типовых элементов соответственно имеют вид

$$I = U/R; \quad I = CdU/dt; \quad U = LdI/dt.$$

Уравнения равновесия (первый закон Кирхгофа) и непрерывности (второй закон Кирхгофа), устанавливающие равенство нулю суммы токов в узлах схемы и суммы напряжений на элементах схемы при их обходе по произвольному контуру, дают топологические уравнения подсистемы

$$\sum_{j=1}^n I_j = 0; \quad \sum_{i=1}^k U_i = 0,$$

где  $n$  — число ветвей в узле схемы;  $k$  — число элементов в контуре схемы.

**Механические поступательная и вращательная подсистемы.** Типовыми элементами этих подсистем являются элементы трения, элементы массы и упругие элементы. В качестве фазовых переменных выступают сила  $F$  и скорость  $v$  для поступательной подсистемы и момент  $M$  и угловая скорость  $\omega$  — для вращательной.

Математические модели элементов трения получаются из уравнений вязкого трения:

$$F = \mu v_{от} = v_{от}/(1/\mu); \quad M = k\omega_{от} = \omega_{от}/(1/k),$$

где  $\mu$  и  $k$  — коэффициенты диссипации соответственно для поступательной и вращательной подсистем;  $v_{от}$ ,  $\omega_{от}$  — относительные скорости контактируемых тел.

Введем понятия механического сопротивления при поступательном движении  $R_m = 1/\mu$  и вращательном  $R_{вр} = 1/k$ .

Тогда

$$F = v_{от}/R_m; \quad M = \omega_{от}/R_{вр}.$$

Математическую модель массы выражает второй закон Ньютона:

$$F = mdv/dt; \quad M = Jd\omega/dt,$$

где  $m$  — масса элемента;  $J$  — момент инерции массы относительно ее оси вращения.

Математическая модель упругого элемента получается из закона Гука. Для поступательной подсистемы

$$\sigma = E\Delta l/l,$$

где  $\sigma$  и  $\Delta l$  — соответственно напряжение и удлинение стержня в продольном направлении;  $E$  — модуль упругости первого рода;  $l$  — начальная длина стержня.

Так как  $\sigma = F/S$ , где  $S$  — площадь поперечного сечения стержня, после дифференцирования по времени получим

$$dF/dt = SEv_{от}/l = cv_{от} \text{ или } v_{от} = (1/c)(dF/dt) = L_m dF/dt,$$

где  $c$  — жесткость;  $v_{от} = dl/dt$  — скорость деформации;  $L_m = 1/c$  — податливость.

Для вращательной системы при закрутке вала на угол  $\varphi$  получаем аналогичное компонентное уравнение

$$\omega_{от} = (1/c_\varphi)(dM/dt) = L_{вр} dM/dt,$$

где  $c_\varphi$  — крутильная жесткость;  $\omega_{от} = d\varphi/dt$  — угловая скорость деформации;  $L_{вр} = 1/c_\varphi$  — крутильная податливость.

Топологические уравнения подсистемы получаются из уравнений равновесия (принцип Д'Аламбера) и уравнений непрерыв-

ности (сумма абсолютной, переносной и относительной скоростей равна нулю):

$$\sum_{j=1}^n F_j = 0; \quad \sum_{i=1}^k v_{\text{оти}} = 0;$$

$$\sum_{j=1}^n M_j = 0; \quad \sum_{i=1}^k \omega_{\text{оти}} = 0.$$

Особенностью механической поступательной системы является то, что в топологических уравнениях фигурируют не алгебраические, а геометрические суммы. Другими словами, топологические уравнения в виде алгебраических сумм должны записываться для проекций сил и скоростей на каждую ось координат.

**Гидравлическая и пневматическая подсистемы.** Основными фазовыми переменными этих подсистем являются массовый расход  $Q_m$  и давление  $p$ .

Рассмотрим компонентные уравнения гидравлической подсистемы для участка трубопровода длиной  $l$  и диаметром  $d$ .

Потери давления  $\Delta p = p_1 - p_2$  ( $p_1, p_2$  — давление на начальном и конечном участках трубопровода соответственно) при стационарном ламинарном течении жидкости с кинематической вязкостью  $\nu$  могут быть определены по формуле Пуазейля

$$\Delta p = 128\nu l Q_m / (\pi d^4) = R_r Q_m \text{ или } Q_m = \Delta p / R_r,$$

где  $R_r = 128\nu l / (\pi d^4)$  — гидравлическое сопротивление.

Любая полость в гидравлической подсистеме может рассматриваться как емкость. Если полость заполнена жидкостью, компонентное уравнение гидравлической системы получается из обобщенного закона Гука:

$$Q_m = C_r dp/dt,$$

где  $C_r$  — приведенная гидравлическая емкость,  $C_r = V\rho/E_{\text{ж}}$ ;  $V$  — объем емкости;  $E_{\text{ж}}$  — модуль упругости жидкости.

Инерционные свойства жидкости позволяет учесть одномерное уравнение Эйлера (закон движения идеальной жидкости):

$$dv/dt = -(1/\rho)(\delta p/l),$$

где  $v$  — скорость течения жидкости;  $\delta p = p_2 - p_1$  — изменение давления на участке трубопровода длиной  $l$ , обусловленное инерционными свойствами жидкости.

Умножив обе части уравнения на  $\rho S$  и заменив  $\delta p$  на  $\Delta p$ , получим

$$dQ_m/dt = (S/l)\Delta p \text{ или } \Delta p = (l/S)(dQ_m/dt) = L_r dQ_m/dt,$$

где  $L_r = l/S$  — гидравлическая индуктивность.

Топологические уравнения гидравлической подсистемы выводятся на основе законов, утверждающих, что сумма потоков в любой общей точке системы и сумма перепадов давления на замкнутом контуре равны нулю:

$$\sum_{j=1}^n Q_{mj} = 0; \quad \sum_{i=1}^k \Delta P_i = 0.$$

Для пневматической подсистемы все уравнения имеют аналогичный вид, разница будет заключаться только в значениях коэффициентов.

**Тепловая подсистема.** Компонентные и топологические уравнения для тепловой подсистемы идентичны по форме уравнениям рассмотренных подсистем за исключением того, что в тепловой подсистеме присутствуют только два типа основных элементов: элементы теплопроводности и теплоемкости. Фазовыми переменными являются температура  $T$  и тепловой поток  $\Phi = dQ/dt$ , где  $Q$  — количество теплоты.

При выделении элементов твердое тело разбивается на участки, каждый из которых характеризуется средней теплоемкостью  $C_T$ . По определению

$$C_T = dQ/dT,$$

следовательно:

$$dQ/dt = C_T dT/dt \text{ или } \Phi = C_T dT/dt.$$

Кроме того, каждый участок обладает теплопроводностью. Математическая модель теплового сопротивления получается из уравнения Фурье

$$q = -\lambda \text{grad } T,$$

где  $q$  — плотность теплового потока;  $\lambda$  — коэффициент теплопроводности;  $T$  — температура.

Если рассматривать передачу теплоты вдоль участка однородного тела с площадью поперечного сечения  $S$  и длиной  $l$ , то

$$\Phi = \lambda S(T_1 - T_2)/l = (T_1 - T_2)/R_T = \Delta T/R_T,$$

где  $R_T = l/(\lambda S)$  — тепловое сопротивление;  $\Delta T = T_1 - T_2$  — разность температур на границах элемента.

Для участков контакта твердого тела с жидкой или газообразной средой тепловое сопротивление определяется теплопередачей конвекцией. Компонентное уравнение элемента сопротивления имеет в данном случае аналогичный вид, но  $R_T = 1/(S\alpha)$ , где  $S$  — площадь контакта;  $\alpha$  — коэффициент теплопередачи.

Топологические уравнения для тепловых подсистем следуют из уравнения теплового баланса для стационарного процесса теплопередачи и закона, устанавливающего, что сумма изменений температуры при обходе по замкнутому контуру равна нулю:

Таблица 2.1

## Фазовые переменные и компоненты для различных подсистем

Подсистемы	Фазовые переменные		Компоненты		
	типа потока $I$	типа потенциала $U$	типа $R$	типа $C$	типа $L$
Электрическая	Ток $I$	Напряжение $U$	Сопротивление $R$	Емкость $C$	Индуктивность $L$
Механическая поступательная	Сила $F$	Скорость $v$	Механическое сопротивление $R_m$	Масса $m$	Податливость $L_m$
Механическая вращательная	Момент $M$	Угловая скорость $\omega$	Вращательное сопротивление $R_{вр}$	Момент инерции $J$	Крутильная податливость $L_{вр}$
Гидравлическая (пневматическая)	Расход $Q_m$	Давление $p$	Гидравлическое сопротивление $R_T$	Гидравлическая емкость $C_T$	Гидравлическая индуктивность $L_T$
Тепловая	Тепловой поток $\Phi$	Температура $T$	Тепловое сопротивление $R_T$	Теплоемкость $C_T$	—

$$\sum_{j=1}^n \Phi_j = 0; \quad \sum_{i=1}^k \Delta T_i = 0.$$

В уравнениях рассмотренных подсистем используются фазовые переменные двух типов: типа потока и типа потенциала — и компоненты трех типов: типа сопротивления  $R$ , типа емкости  $C$  и типа индуктивности  $L$  (у тепловой подсистемы компонент типа индуктивности  $L$  отсутствует). Нетрудно убедиться, что при правильном выборе аналогий (табл. 2.1) для описания всех рассмотренных подсистем можно использовать идентичные уравнения. Данное обстоятельство дает возможность автоматизировать процесс получения ММ объектов после выполнения предварительных процедур, связанных с получением формального представления структуры этих объектов.

## 2.4. Формальное представление структуры объекта на макроуровне

Для формального представления структуры объекта удобно использовать графические изображения, облегчающие пользователю восприятие модели и приводящие к представлению модели в той или иной схемной форме. Топологические уравнения систем, представленных выше, записывались применительно к узлам и контурам. Следовательно, сама форма топологических уравнений требует отождествления участков реальной структуры объектов или характеризующих эти участки величин с некоторыми ветвями и узлами, поэтому первым этапом моделирования объектов должна быть замена реального объекта некоторым графическим представлением, состоящим из связанных между собой ветвей, — графами и эквивалентными схемами.

Графы применяются не только для представления топологических уравнений в моделях макроуровня. Ими широко пользуются для получения структурных и функциональных математических моделей на разных уровнях проектирования, а также при решении задач синтеза структуры.

*Графом* называют совокупность вершин (узлов) и связывающих их ребер (ветвей). Если для ребер графа указываются определенные направления, такой граф является *направленным*. Любая последовательность ребер, в которой соседние ребра инцидентны одной и той же вершине, называется *маршрутом*. Термин «инцидентность» означает соотношение объектов типа «проходит через...» или «находится на...».

Если в маршруте нет повторяющихся ребер, маршрут называют *цепью*. Если цепь начинается и заканчивается в одной и той же вершине, имеем *цикл*.



*Связным графом* называют граф, в котором можно указать маршрут, связывающий любые вершины. Частным случаем связного графа без циклов является *дерево*. Его ребра называют *ветвями*.

Граф — удобное средство отображения структуры проектируемого объекта. Чаще всего при таком отображении вершины графа отождествляются с элементами структуры, а ребра соответствуют связям между элементами. При моделировании объектов, наоборот, ребрами изображают элементы, а вершинами — связи между элементами.

Если граф отображает только такие структурные особенности, как способ связи элементов друг с другом, то *эквивалентная схема* содержит более полную информацию об объекте, отображает физическую сущность отдельных элементов. Рассмотрим особенности представления моделей в виде эквивалентных схем.

В разных областях техники применяют специфические системы обозначений элементов на эквивалентных схемах. Будем использовать в дальнейшем единую систему обозначений для элементов всех подсистем, обычно применяемую при изображении электрических эквивалентных схем, а при составлении схем придерживаться правил, приведенных ниже.

1. Эквивалентная схема, как и граф, состоит из множества ветвей и множества узлов.

2. Каждая ветвь относится к одному из пяти возможных типов, изображенных на рис. 2.4.

3. Каждой ветви соответствует свое компонентное уравнение:

для емкостной (рис. 2.4, а)  $I = CdU/dt$ ;

для индуктивной (рис. 2.4, б)  $U = LdI/dt$ ;

для резистивной (рис. 2.4, в)  $I = U/R$ ;

для ветви источника типа потенциала (рис. 2.4, г) уравнение, связывающее разность потенциалов  $U$  на узлах данной ветви с независимой переменной  $t$  и, возможно (для зависимой ветви), с фазовыми переменными других ветвей:  $U = f_1(\mathbf{V}, t)$ , где  $\mathbf{V}$  — вектор фазовых переменных;  $t$  — время. В частном случае  $U = \text{const}$ ;

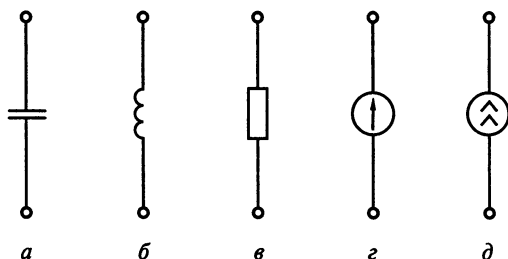


Рис. 2.4. Типы ветвей в эквивалентных схемах:

а — емкостная; б — индуктивная; в — резистивная; г — источника типа потенциала; д — источника типа тока

для ветви источника типа потока (рис. 2.4, д) уравнение, связывающее поток  $I$  в данной ветви со временем  $t$  и, возможно (для зависимой ветви), с фазовыми переменными других ветвей:  $I = f_2(\mathbf{V}, t)$ . В частном случае  $I = \text{const}$ .

4. Каждому узлу схемы соответствует определенное значение фазовой переменной типа потенциала, а каждой ветви — значения переменных  $I$  и  $U$ , фигурирующих в компонентных уравнениях. Соединение ветвей друг с другом, т.е. образование узлов, должно отражать взаимодействие элементов в системе. Выполнение этого условия обеспечивает справедливость топологических уравнений для узлов и контуров схемы.

*Зависимая ветвь* — ветвь, параметр которой ( $U$  в источнике типа потенциала,  $I$  в источнике типа потока или  $R, C, L$ ) зависит от каких-либо фазовых переменных.

Для отражения взаимосвязей подсистем в их эквивалентные схемы вводят специальные преобразовательные элементы. Различают три вида связей подсистем. Трансформаторная и гираторная связи выражают соотношения между фазовыми переменными двух подсистем. Этим типам связей соответствуют элементы, представляемые парами источников тока или напряжения. Их называют зависимыми источниками. Третий вид связи выражает влияние фазовых переменных одной подсистемы на параметры элементов другой и задается в виде зависимостей  $C, L$  или  $R$  от фазовых переменных. Примеры изображения связей в эквивалентных схемах приведены на рис. 2.5. Здесь запись вида  $A(B)$  означает, что фазовая переменная  $A$  является функцией фазовой переменной  $B$ .

Таким образом, если нужно отобразить только топологические уравнения, достаточно использования графов; если нужно отобразить топологические и компонентные уравнения, целесообразно воспользоваться эквивалентными схемами. Следует, однако, отметить, что, если эквивалентные схемы строить по приведенным выше правилам, между ними и графами в отношении структуры имеется строгое соответствие. Тогда можно проставить у ветвей графа условные буквенные обозначения типов ветвей. В этом случае граф содержит ту же информацию, что и эквива-

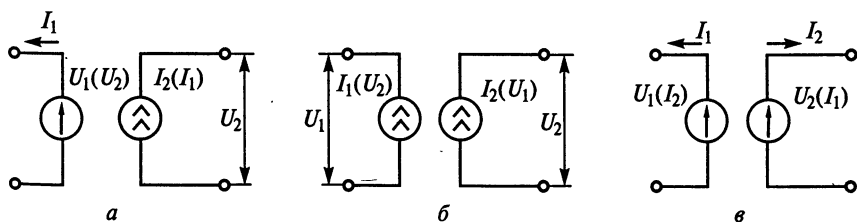


Рис. 2.5. Примеры изображения связей в эквивалентных схемах:

*a* — трансформаторной; *б, в* — гираторной

лентная схема, и можно с равным успехом использовать оба графических представления объекта.

Рассмотрим общие правила составления эквивалентной схемы на примере механической системы. Сначала выбирается инерциальная система отсчета, с которой связывается система координат (обычно декартова). После этого разрабатывается структурная схема, отражающая состав объекта и связи его элементов. Схема должна содержать все элементы, свойства которых предполагается учитывать в рамках поставленной задачи. На схеме наносятся источники внешних силовых и кинематических воздействий в виде сил (моментов) и скоростей. Инерционные элементы представляются в виде сосредоточенных масс. Элементы, отражающие упругие и демпфирующие свойства и внешние воздействия, часто ориентируются относительно осей координат. Назначаются *обобщенные координаты* системы — взаимно-однозначные функции координат, определяющие положение объекта в пространстве в каждый момент времени. Число обобщенных координат равно числу степеней свободы. Обобщенные координаты фигурируют в дальнейшем в компонентных и топологических уравнениях: фазовые переменные типа потока являются функциями производных по времени от обобщенных координат (функциями обобщенных скоростей).

На основании структурной схемы строится эквивалентная схема:

- задается базовый узел, который в дальнейшем интерпретируется как инерциальная система отсчета. В приведенных ниже примерах он обозначен номером  $0$ ;

- для каждого тела с учитываемыми инерционными свойствами в эквивалентной схеме выделяется отдельный узел. Если тело имеет несколько степеней свободы относительно осей координат, каждой из них должен соответствовать свой узел тела. Узел тела в дальнейшем интерпретируется как само тело, совершающее движение относительно соответствующей оси координат: к узлу прикладываются силы (моменты), действующие на тело в выбранном направлении, потенциал узла характеризует скорость тела в этом направлении относительно инерциальной системы координат (обобщенную скорость);

- между узлом тела и базовым узлом включается емкостная ветвь; она характеризует силу (момент) инерции;

- трение между контактируемыми телами отражаются резистивной ветвью, включенной между их узлами;

- упругие элементы, соединяющие тела, отражаются индуктивной ветвью между соответствующими узлами схемы;

- внешние силы (моменты) и внешние источники скорости, действующие на систему, учитываются в схеме соответственно ветвями источника типа потока и источника типа потенциала,

включенными между базовым узлом и узлом элемента, к которому они приложены;

- узлы схемы нумеруются в произвольном порядке, начиная с 1.

В ряде случаев невозможно непосредственно связать все ветви схемы подобным образом. Наличие в системе преобразователей скорости и силы (момента), таких, например, как рычажные механизмы или зубчатые передачи, приводит к тому, что фазовые переменные контактируемых элементов требуют дополнительного согласования. То же самое может возникнуть, когда тело совершает одновременно поступательное и вращательное движения. В подобных случаях при построении эквивалентной схемы используются схемы, разработанные для отдельных подсистем и объединенные между собой с помощью трансформаторных связей. Если в системе не учитываются упругие свойства связи между контактируемыми телами, в одну из подсистем между базовым узлом и узлом тела помещают зависимый источник типа потока, а в другую — зависимый источник типа потенциала. При этом безразлично, какой из источников в какую подсистему будет включен. Если в системе присутствуют элементы податливости между контактируемыми телами, их характеристики должны быть приведены к фазовой переменной типа потока одной из связываемых подсистем, а сами они должны быть включены в эту подсистему посредством ветвей, один конец которых подсоединен к узлу тела. Между другим концом ветвей и базовым узлом помещают зависимый источник типа потенциала. Зависимый источник типа потока размещают между узлом тела и базовым узлом другой подсистемы.

Следует отметить, что скорости и силы (моменты) — это векторы, и поэтому при размещении в схеме их источников следует правильно указывать их направление.

Более подробно правила составления эквивалентных схем с зависимыми источниками будут рассмотрены в примерах.

На рис. 2.6 изображены эквивалентная схема и соответствующий ей граф для ранее рассмотренного примера одно-массовой колебательной системы, структурная схема которой приведена на рис. 2.3. Начало координат  $0$  здесь связано с инерциальной системой отсчета, совершающей движение в гори-

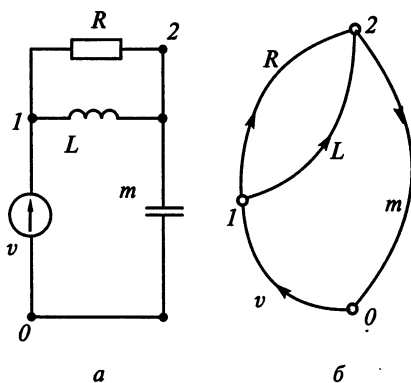


Рис. 2.6. Эквивалентная схема (а) и граф (б) одно-массовой колебательной системы

зонтальной плоскости совместно с корпусом транспортного средства. Система имеет одну степень свободы: рассматривается движение сиденья водителя в вертикальном направлении. В качестве обобщенной координаты выбрано вертикальное перемещение центра масс водителя и сиденья. Инерционные свойства водителя и сиденья учтены емкостной ветвью  $m$ , включенной между узлами 0 и 2. Потенциал узла 2 характеризует вертикальную скорость сиденья, а поток в емкостной ветви — силу инерции. Упругие и демпфирующие свойства подвески сиденья отражены в схеме ветвями  $L$  и  $R$ . Разность потенциалов в узлах 1 и 2 соответствует скорости деформации элементов подвески, а потоки в ветвях — действующим в них силам. Согласно условию задачи, колебания водителя на сиденье совершаются вследствие вертикального перемещения корпуса, что отражено в схеме введением ветви  $v$  источника типа потенциала (источника кинематических воздействий). Изменение потенциала в узле 1 характеризует изменение вертикальной скорости корпуса. Закон этого изменения должен быть задан в характеристике источника кинематических воздействий.

## 2.5. Примеры составления эквивалентных схем технических объектов

**Механические поступательные системы.** Рассмотрим пример поступательного движения тягача с нагруженным прицепом. Закон изменения силы тяги колес тягача известен. Пусть требуется определить скорости поступательного перемещения тягача, прицепа и груза, а также усилия взаимодействия этих тел. Основными факторами, влияющими на поведение системы, являются инерционные свойства тягача, прицепа и груза, упругие свойства сцепки и элементов крепления груза, силы сопротивления перемещению тягача и прицепа с грузом, сила трения груза о платформу прицепа. Допустим, что прочие факторы в расчет не принимаются, жесткость учитываемых упругих элементов постоянна, силы сопротивления и трения пропорциональны относительной скорости перемещения элементов системы. Структурная схема динамической системы тягача с прицепом в этом случае будет иметь вид, представленный на рис. 2.7, а. Здесь  $m_1$  — масса тягача;  $m_2$  — масса прицепа;  $m_3$  — масса груза;  $L_1$  — упругость сцепки;  $L_2$  — упругость элементов крепления груза;  $F$  — касательная сила тяги колес тягача;  $R_1$  — сила сопротивления перемещению тягача;  $R_2$  — сила сопротивления перемещению прицепа с грузом. Внутренняя сила  $R_3$  трения груза о платформу прицепа на схеме не показана. Система имеет три степени свободы. В качестве обобщенных координат выбираем продольные перемещения тягача  $q_1$ , прицепа  $q_2$

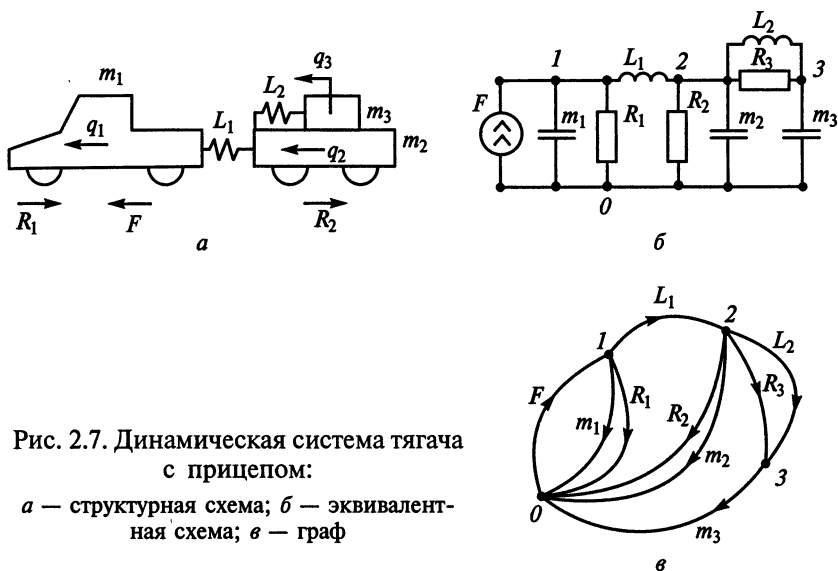


Рис. 2.7. Динамическая система тягача с прицепом:

*a* — структурная схема; *б* — эквивалентная схема; *в* — граф

и груза  $q_3$ . Начало координат связываем с дорогой. На рис. 2.7, *б* представлена эквивалентная схема объекта, а на рис. 2.7, *в* — ее граф. Инерционные свойства тягача, прицепа и груза учтены емкостными ветвями, включенными между узлом 0 (дорогой) и узлами 1 (тягач), 2 (прицеп) и 3 (груз). Ветвь  $F$  источника силы между узлами 0 и 1 отражает воздействие на тягач внешней силы со стороны дороги. Для учета силы сопротивления движению тягача введена ветвь  $R_1$  между узлами 0 и 1, а для учета силы сопротивления движению прицепа — ветвь  $R_2$  между узлами 0 и 2. Трение груза о платформу прицепа учитывается введением ветви  $R_3$  между узлами 2 и 3. Упругая связь между тягачом и прицепом учтена посредством введения индуктивной ветви  $L_1$  между узлами 1 и 2, а упругость элементов крепления груза — индуктивной ветвью  $L_2$ , включенной между узлами 2 и 3. Потенциал узла 1 характеризует скорость перемещения тягача, узла 2 — скорость перемещения прицепа, узла 3 — скорость перемещения груза относительно дороги. Разность потенциалов в узлах 1 и 2 соответствует скорости перемещения прицепа относительно тягача, в узлах 2 и 3 — скорости перемещения груза относительно прицепа, а в узлах 1 и 3 — скорости перемещения груза относительно тягача. Величины потоков в ветвях схемы соответствуют: для ветви  $m_1$  — силе инерции тягача, ветви  $m_2$  — силе инерции прицепа, ветви  $m_3$  — силе инерции груза, ветви  $R_1$  — силе сопротивления передвижению тягача, ветви  $R_2$  — силе сопротивления передвижению прицепа, ветви  $R_3$  — силе трения груза о платформу прицепа, ветви  $L_1$  — усилию в сцепном устройстве, ветви  $L_2$  — усилию в элементах крепления груза.

На рис. 2.8 представлены структурная схема трехмассовой динамической системы, эквивалентная схема и ее граф, которые могут быть использованы для исследования вертикальных колебаний остова и сиденья водителя транспортного средства в продольной плоскости. В отличие от примера рис. 2.3 здесь дополнительно учтены: радиальная податливость шин  $L_1$  и демпфирование в них  $R_1$ , податливость элементов подвески  $L_2$  и трение в них  $R_2$ , неподрессоренная масса  $m_1$ . Масса остова представлена сосредоточенной массой  $m_2$ ; масса сиденья с водителем — массой  $m_3$ ; элементы  $L_3$  и  $R_3$  учитывают упругие и демпфирующие свойства сиденья. Начало координат  $0$  связано с инерциальной системой, совершающей движение вдоль горизонтальной оси совместно с машиной. Система имеет три степени свободы вдоль вертикальной оси. Обобщенными координатами являются вертикальные перемещения  $q_1$  неподрессоренной массы, центра масс остова  $q_2$  и сиденья с водителем  $q_3$ . Кинематические воздействия на колеса со стороны дороги учтены источником скорости  $v$ . Потенциалы в узлах 2, 3 и 4 эквивалентной схемы характеризуют абсолютные вертикальные скорости соответственно неподрессоренных масс, остова и водителя на сиденье. Следует заметить, что эквивалент-

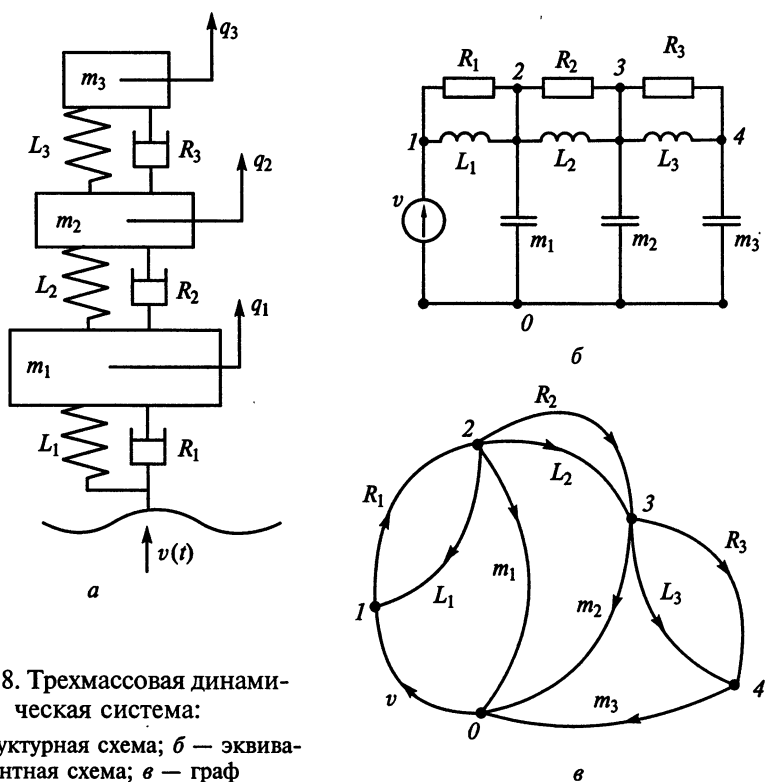


Рис. 2.8. Трехмассовая динамическая система:

*a* — структурная схема; *б* — эквивалентная схема; *в* — граф

ная схема составлена без учета внешних постоянных сил, обусловленных весом остова, неподдрессоренных масс и водителя, поэтому математическая модель будет описывать поведение системы относительно ее статического состояния.

**Механические вращательные системы.** Рассмотрим фрагмент динамической системы зубчатого редуктора (рис. 2.9, *a*). Входной вал под действием крутящего момента  $M_{\text{вх}}$  вращается на подшипниках совместно с зубчатым колесом. Потери на трение в подшипниках учитываются в эквивалентной схеме сопротивлениями  $R_1$  и  $R_2$ , крутильные податливости участков валов — элементами  $L_1$  и  $L_2$ , инерционные свойства вала и колеса — элементом  $J$ . Влияние остальных элементов редуктора на рассматриваемую подсистему учитывается посредством внешнего момента нагрузки  $M_{\text{н}}$ . Эквивалентная схема подсистемы зубчатого редуктора и граф представлены на рис. 2.9, *б* и *в* соответственно. Здесь базовый узел  $0$  соответствует невращающейся части редуктора — корпусу. В качестве обобщенной координаты  $q_1$  выбран угол поворота колеса относительно оси вала в направлении действия момента  $M_{\text{вх}}$ .

Другим примером механической вращательной системы может служить фрагмент расчетной схемы для исследования динамических процессов, возникающих при включении фрикционного сцепления. Схема динамической системы фрикционного сцепления, его эквивалентная схема и соответствующий ей граф представлены

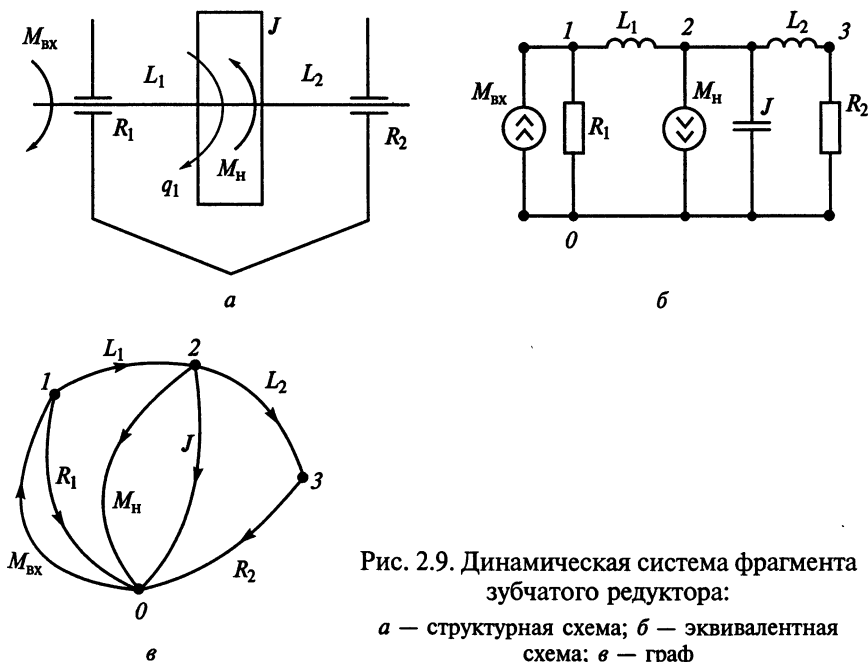


Рис. 2.9. Динамическая система фрагмента зубчатого редуктора:

*a* — структурная схема; *б* — эквивалентная схема; *в* — граф



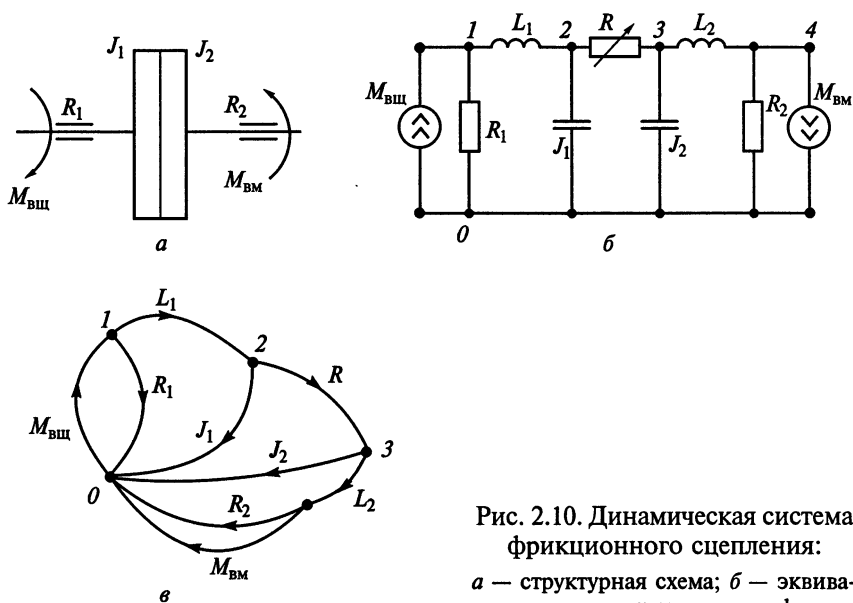


Рис. 2.10. Динамическая система фрикционного сцепления:  
*a* — структурная схема; *б* — эквивалентная схема; *в* — граф

на рис. 2.10. Здесь  $M_{\text{вщ}}$ ,  $M_{\text{вм}}$  — соответственно моменты на ведущем и ведомом валах сцепления;  $J_1$ ,  $J_2$  — вращающиеся массы, учитывающие инерционные свойства соответственно ведущих и ведомых частей;  $L_1$ ,  $L_2$  — крутильные податливости валов;  $R_1$ ,  $R_2$  — элементы, учитывающие трение в подшипниках;  $R = R(t)$  — элемент, определяющий закон изменения величины момента трения между дисками в процессе включения сцепления.

**Сложные механические подсистемы.** Рассмотрим схему зубчатого редуктора (рис. 2.11, *a*). Здесь  $M_{\text{вщ}}$ ,  $M_{\text{вм}}$  — крутящие моменты соответственно на ведущем и ведомом валах редуктора;  $J_1$ — $J_4$  — моменты инерции вращающихся масс зубчатых колес и примыкающих к ним участков валов относительно осей вращения;  $L_1$ — $L_3$ ,  $L_5$ — $L_7$ ,  $L_9$  — крутильные податливости участков валов;  $L_4$ ,  $L_8$  — крутильные податливости зубчатых колес первой и второй пары зацепления, приведенные соответственно к угловой скорости промежуточного и выходного валов;  $R_1$ — $R_6$  — элементы, учитывающие трение в подшипниках;  $q_1$ — $q_4$  — обобщенные координаты (углы поворота колес).

В этой системе две пары зубчатых колес исполняют роль преобразователей фазовых переменных (крутящего момента и скорости), поэтому можно выделить три подсистемы, элементы которых имеют общие фазовые переменные: подсистему ведущего вала, подсистему промежуточного вала и подсистему выходного вала.

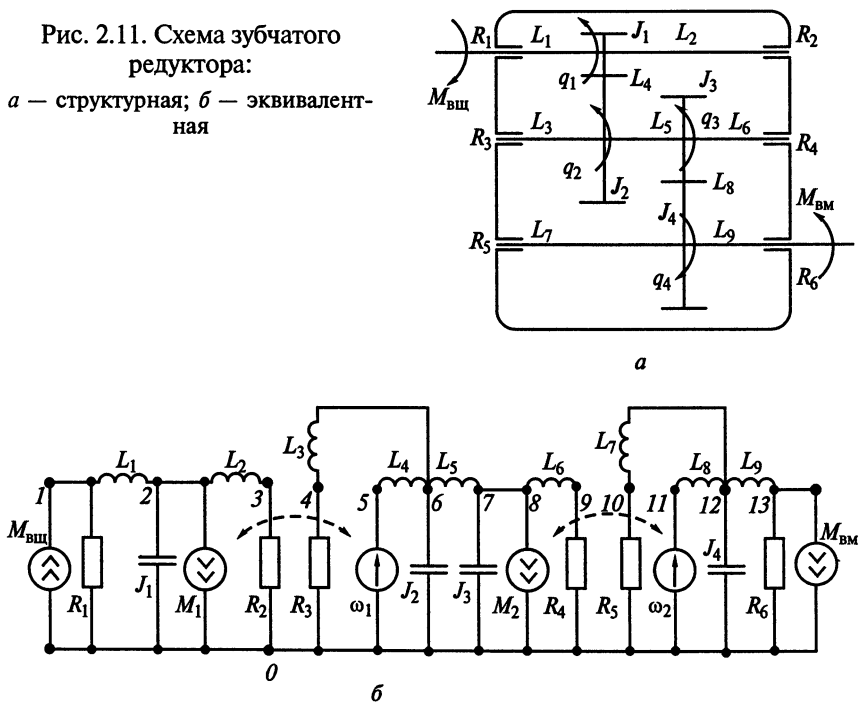
Рассмотрим механизм взаимодействия первой пары зубчатых колес. Колесо с моментом инерции  $J_1$ , вращаясь с угловой скоростью  $\omega_1 = dq_1/dt$ , имеет контакт с одним концом податливого элемента  $L_4$  колеса с моментом инерции  $J_2$  и сообщает ему (концу податливого элемента) угловую скорость  $\omega_2$ . Если в данный момент времени  $\omega_2 \neq dq_2/dt$ , податливый элемент деформируется и в нем возникает упругий момент  $M_{уп}$ . Этот момент воспринимается колесом с моментом инерции  $J_2$  непосредственно, а колесом с моментом инерции  $J_1$  в виде реакции со стороны упругого элемента. Обозначим:  $M'$  — крутящий момент на первом колесе;  $M'' = M_{уп}$  — крутящий момент на втором колесе;  $D_1$  и  $D_2$  — диаметры начальных окружностей соответственно первого и второго колес;  $u_1$  — передаточное число первой пары зубчатых колес.

Тогда, пренебрегая потерями в зацеплении, получим

$$M''/M' = \omega_1/\omega_2 = D_2/D_1 = u_1.$$

Таким образом, между фазовыми переменными рассматриваемых подсистем существуют однозначные связи, определяемые передаточными числами зубчатых колес, находящихся в зацеплении; это позволяет объединить подсистемы и создать математическую модель объекта.

Рис. 2.11. Схема зубчатого редуктора:  
 а — структурная; б — эквивалентная



На рис. 2.11, б приведена эквивалентная схема данного редуктора, состоящая из 14 узлов и 25 ветвей. Они являются элементами трех подсистем — по числу валов редуктора. Базовый узел 0 связан с корпусом редуктора. Для учета взаимного влияния подсистем ведущего и промежуточного валов используется трансформаторная связь: в подсистему ведущего вала между базовым узлом и инерционным элементом включен зависимый источник момента  $M_1$ , а в подсистему промежуточного вала зависимый источник угловой скорости  $\omega_1$  между базовым узлом и ветвью  $L_4$ . Поток в ветви  $M_1$  пропорционален потоку в ветви  $\omega_1$ , а потенциал узла 5 пропорционален потенциалу узла 2. Коэффициенты этих зависимостей определяются передаточным числом  $u_1$  первой пары зубчатых колес. Компонентные уравнения источников момента и скорости с учетом выбранного направления обобщенных координат имеют вид

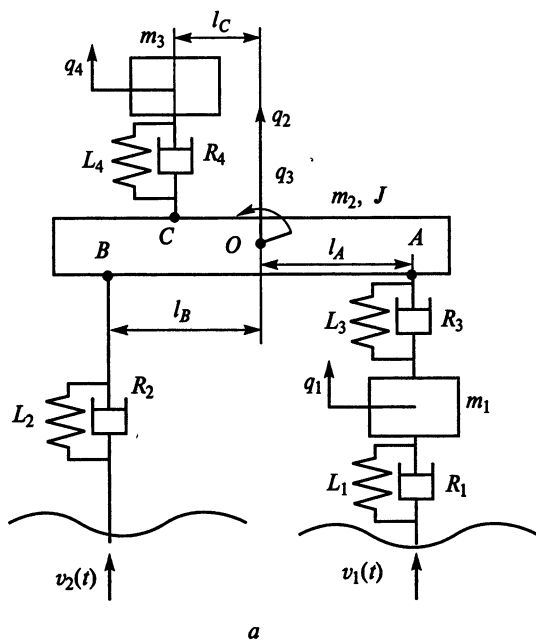
$$M_{0-2} = M_{0-6}/u_1; \quad \omega_5 = \omega_2 u_1.$$

Здесь индексы при переменных соответствуют номерам узлов схемы.

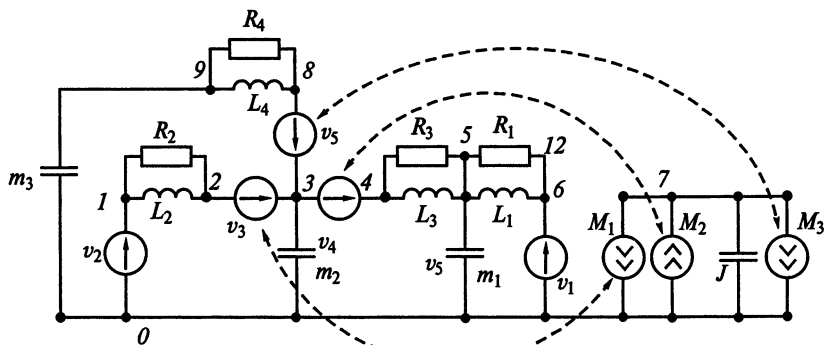
Аналогичная трансформаторная связь с использованием зависимых источников  $M_2$  и  $\omega_2$  включена между второй и третьей подсистемами.

В рассмотренном примере упругие свойства первого зубчатого зацепления учитывались крутильной податливостью колес, приведенной к угловой скорости промежуточного вала, которую можно было бы привести к угловой скорости ведущего вала и учитывать в первой подсистеме. В этом случае индуктивную ветвь следует подсоединить к узлу 2, между ней и базовым узлом поместить зависимый источник скорости, а между базовым узлом и инерционным элементом второй подсистемы (узел б) — зависимый источник момента.

Рассмотрим пример сложной механической поступательной системы (рис. 2.12), которая представляет собой транспортное средство, движущееся по дороге на пневматических шинах с поддрессоренным передним мостом и поддрессоренным сиденьем водителя. Схема динамической системы транспортного средства представлена на рис. 2.12, а. Требуется составить эквивалентную схему объекта для исследования вертикальных колебаний на сиденье водителя, вертикальных колебаний переднего моста и вертикальных и угловых колебаний остова машины в продольной плоскости. При решении задачи в расчет принимаются: инерционные свойства переднего моста (масса  $m_1$ ); инерционные свойства остова (масса  $m_2$  и момент инерции  $J$ ); инерционные свойства сиденья с водителем (масса  $m_3$ ); упругие и демпфирующие свойства передних шин ( $L_1$  и  $R_1$ ); упругие и демпфирующие свойства задних шин ( $L_2$  и  $R_2$ ); упругие и демпфирующие свойства передней подвески ( $L_3$  и  $R_3$ );



a



б

Рис. 2.12. Схема транспортного средства:

a — структурная; б — эквивалентная

упругие и демпфирующие свойства сиденья ( $L_4$  и  $R_4$ ); место расположения элементов конструкции относительно центра масс остова  $O$  ( $l_A$ ,  $l_B$  и  $l_C$ ). Принято, что законы изменения высоты неровностей известны и одинаковы для левого и правого колес.

Основным отличием данного примера от примера рис. 2.8 является то, что здесь учитывается вращение остова в продольной плоскости вокруг его центра масс. Система имеет четыре степени

свободы: вертикальные перемещения сосредоточенных масс  $m_1$ ,  $m_2$ ,  $m_3$  и поворот массы с моментом инерции  $J$  вокруг оси, проходящей через центр  $O$  и перпендикулярной продольной плоскости. В соответствии с этим выбраны обобщенные координаты  $q_1, q_2, q_3, q_4$ .

Здесь, как и в примере с редуктором (см. рис. 2.11), уравнения контактируемых элементов содержат разные фазовые переменные. Так, точка  $A$  (см. рис. 2.12, *a*) является общей для элементов  $m_2$ ,  $J$  и  $L_3$ . Но в компонентных уравнениях элемента  $L_3$  фигурируют вертикальная скорость точки  $A$  и вертикальная сила ( $v_A$  и  $F_A$ ), для массы остова  $m_2$  уравнения содержат переменные  $dq_2/dt$  и  $F_A$ , а в уравнения элемента  $J$  входят переменные  $dq_3/dt$  и момент  $M_A$ . Однако между однородными фазовыми переменными этих подсистем существуют однозначные связи:

$$v_A = dq_2/dt + (dq_3/dt)l_A; M_A = F_A l_A.$$

Аналогичные связи имеют место и в других общих точках системы ( $B$  и  $C$ ):

$$v_B = dq_2/dt - (dq_3/dt)l_B; M_B = -F_B l_B;$$

$$v_C = dq_2/dt - (dq_3/dt)l_C; M_C = -F_C l_C.$$

Наличие таких связей позволяет с помощью дополнительных уравнений, отражающих законы функционирования зависимых источников типа потока и типа потенциала, связать между собой отдельные подсистемы.

На рис. 2.12, *б* приведена эквивалентная схема данного объекта. Здесь базовый узел  $0$  связан с инерциальной системой, совершающей движение вдоль горизонтальной оси совместно с машиной. Массе  $m_2$  остова, совершающей поступательное движение, соответствует узел  $3$ , вращающейся массе с моментом инерции  $J$  — узел  $7$ , массе  $m_1$  переднего моста — узел  $5$ , массе  $m_3$  водителя — узел  $9$ .

Для связи подсистем использованы зависимые источники скорости  $v_3, v_4, v_5$  и зависимые источники момента  $M_1, M_2, M_3$ . Внешние кинематические возмущения со стороны дороги учтены источниками скорости  $v_1$  и  $v_2$ . Точке  $A$  структурной схемы соответствует узел  $4$  в эквивалентной схеме, точке  $B$  — узел  $2$ , а точке  $C$  — узел  $8$ . Влияние вращательной подсистемы на поступательную отражено введением зависимых источников скорости, а влияние поступательной на вращательную — введением зависимых источников момента. Чтобы правильно учесть взаимное влияние подсистем, необходимо должным образом задать направления действия источников.

Если вместо расстояний используются продольные координаты общих точек относительно центра масс остова  $O$ , направ-

ления моментов и скоростей учитываются в компонентных уравнениях зависимых источников. На схеме же все зависимые источники скорости будут направлены от узла 3, а все источники момента — к узлу 7.

**Гидравлические (пневматические) подсистемы.** При составлении эквивалентных схем гидравлических и пневматических подсистем за базовый узел обычно принимается внешняя среда. Любой трубопровод можно представить состоящим из нескольких секций, эквивалентная схема каждой из которых имеет вид, представленный на рис. 2.13.

Пользователь, учитывая конкретные условия, может пренебрегать какими-либо из элементов  $R_r$ ,  $C_r$  или  $L_r$ . Резервуары, скорость течения жидкости в которых относительно мала и поэтому потери на трение и изменение кинетической энергии потока незначительны, часто изображаются только емкостными ветвями, подсоединенными одним полюсом к базовому элементу, а вторым — к другим элементам трубопровода. Участок трубы относительно большой длины и с малым поперечным сечением изображается ветвями гидравлического сопротивления и гидравлической индуктивности, включенными последовательно; изменением объема жидкости пренебрегают.

На рис. 2.14 представлены гидравлическая и эквивалентная схемы гидравлической подсистемы.

Гидравлический насос  $p$  качает жидкость, которая по трубопроводу поступает в открытый резервуар  $C$ . В эквивалентной схеме насос представлен источником давления  $p$ , участок трубопровода — индуктивной  $L$  и резистивной  $R$  ветвями, а резервуар — переменной гидравлической емкостью  $C$ .

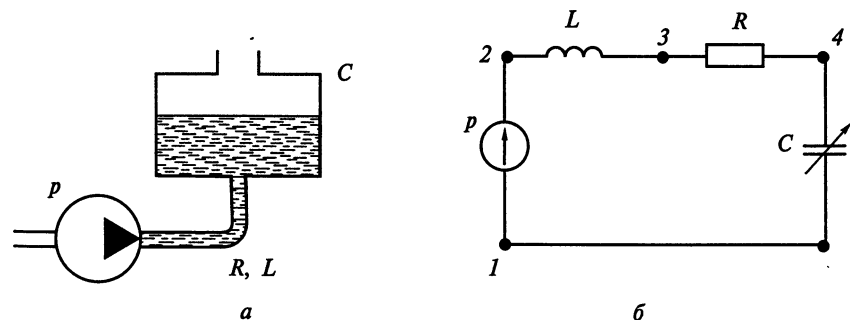


Рис. 2.14. Схема гидравлической подсистемы:

*a* — общая; *б* — эквивалентная

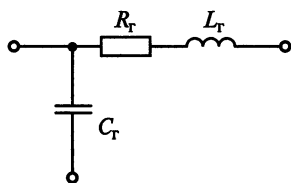


Рис. 2.13. Эквивалентная схема секции трубопровода

**Разнородные физические подсистемы.** В сложных технических системах часто используют элементы, для описания процессов в которых требуются фазовые переменные, характерные для разных физических систем. В подобных случаях целесообразно составлять системы уравнений отдельно для электрической, тепловой и механической частей (подсистем) сложной системы. Взаимосвязь частей должна проявляться в объединении таких автономно составленных систем уравнений в общую систему уравнений на определенном этапе моделирования.

В эквивалентных схемах каждая подсистема физической системы имеет свое самостоятельное изображение.

Физические элементы, в которых происходит преобразование или иное воздействие переменной одной подсистемы на переменные другой, должны иметь описание в более чем одной подсистеме. Компонентные уравнения этих элементов должны содержать соотношения между фазовыми переменными двух разнородных подсистем.

Ранее было введено понятие внешних параметров как количественных оценок факторов внешней по отношению к подсистеме среды. Предполагалось, что внешние параметры при анализе процессов в подсистеме считаются постоянными (например, температура) или зависящими только от времени (например, параметры внешней нагрузки). Если же анализу подвергаются процессы взаимодействия подсистем, то очевидно, что многие из внешних параметров уже не могут быть заданы до начала анализа как функции времени. Их текущие значения становятся известными только в процессе совместного решения систем уравнений всех подсистем.

Таким образом, взаимовлияние подсистем может быть отражено, если внешние параметры одной подсистемы рассматривать как функции фазовых переменных другой подсистемы. При составлении эквивалентных схем это взаимовлияние находит отражение либо включением дополнительных ветвей типа источников, либо учетом зависимостей внутренних параметров (сопротивлений, емкостей, индуктивностей) уже имеющихся в схеме ветвей от фазовых переменных других подсистем.

Для примера рассмотрим систему с силовым гидравлическим цилиндром, схема которой представлена на рис. 2.15, а. Гидравлический насос качает жидкость в цилиндр и заставляет перемещаться поршень, нагруженный внешней силой. Давление жидкости на выходе из насоса и закон изменения силы на штоке поршня известны.

Можно выделить две подсистемы: гидравлическую и механическую поступательную. Для каждой из них строится своя эквивалентная схема. В эквивалентной схеме гидравлической подсистемы насос отображается ветвью внешнего источника дав-

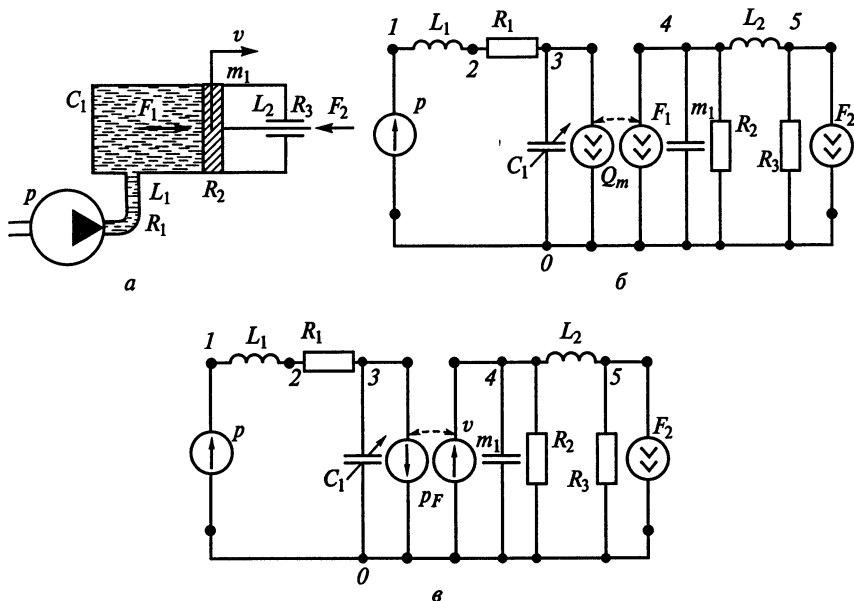


Рис. 2.15. Схема силового гидравлического цилиндра:  
*a* — общая; *б, в* — эквивалентная

ления  $p$ , потери в трубопроводе учитываются ветвями  $L_1$  и  $R_1$ , а влияние внутренней полости гидроцилиндра — емкостной ветвью  $C_1$ . В схеме механической подсистемы учитываются инерционные свойства поршня (ветвь  $m_1$ ), трение поршня о стенки цилиндра (ветвь  $R_2$ ), трение штока в контакте с уплотнением (ветвь  $R_3$ ), податливость штока (ветвь  $L_2$ ) и внешняя нагрузка (ветвь  $F_2$ ).

Взаимодействие подсистем заключается в том, что давление  $p$  в цилиндре (узел 3) создает силу  $F_1$ , действующую на поршень. Под ее воздействием поршень перемещается со скоростью  $v$  (узел 4), изменяя поток  $Q_m$  в цилиндре. На рис. 2.15, *б* связь подсистем учитывается включением в схему зависимых источников типа потока — ветви  $Q_m$  и  $F_1$ , компонентные уравнения которых имеют вид

$$F_1 = pS; \quad Q_m = \rho Sv,$$

где  $S$  — площадь поршня;  $\rho$  — плотность рабочей жидкости.

Взаимодействие подсистем можно учесть и другим образом. Поток жидкости в полости цилиндра вызывает перемещение поршня со скоростью  $v$ . При этом давление жидкости в цилиндре зависит от силы, действующей со стороны поршня. Тогда связь между подсистемами может быть учтена включением зависимых



источников типа потока — ветви  $p_F$  и  $v$  (рис. 2.15, в), компонентные уравнения которых имеют вид

$$p = F_1/S; \quad v = Q_m/\rho S.$$

## 2.6. Моделирование работы технических объектов на макроуровне

В программах анализа, используемых в САПР, автоматизировано как решение, так и формирование ММ объектов. Наиболее общий случай ММ на макроуровне есть ее представление в виде системы ОДУ вида

$$F\left(\frac{d\mathbf{V}}{dt}, \mathbf{V}, t\right) = 0,$$

где  $F$  — вектор-функция;  $\mathbf{V}$  — вектор фазовых переменных;  $t$  — время.

Данная система состоит из компонентных и топологических уравнений. Обычно пользователь САПР не касается непосредственно вопросов, связанных с получением уравнений. Он оперирует эквивалентными схемами, состоящими из набора элементов, соединенных между собой надлежащим образом. Однако использование элементов в схеме предполагает их наличие в библиотеке моделей программного комплекса анализа динамических систем. Первоначально библиотека моделей создается разработчиком САПР и содержит типовой набор моделей, достаточный, по мнению разработчика, для решения общих задач анализа ряда технических систем. По мере использования САПР возникает потребность в расширении набора моделей. В этом случае пользователь обращается к разработчику комплекса или сам пополняет библиотеку.

Задача создания ММ для библиотеки непростая. С ней может справиться только квалифицированный пользователь, обладающий знаниями методов и алгоритмов автоматического получения топологических уравнений. Рядовому пользователю достаточно уметь правильно составить эквивалентную схему, описать ее на входном языке и сформулировать задание на выполнение. После этого ММ объекта будет получена автоматически в соответствии с программой, созданной разработчиком САПР.

В САПР применяются несколько методов формирования ММ объектов, различающихся выбором системы базисных переменных. В основе некоторых из них лежат известные методы, традиционно используемые в различных областях техники, например *узловой метод*. В теории электрических цепей он известен как *метод узловых потенциалов*, а в механике — как *метод перемещений*. В качестве

основного топологического уравнения в узловом методе принимают уравнение равновесия, а в качестве базисных координат — фазовые переменные, относящиеся к узлам эквивалентной схемы. Узловой метод успешно применяется и для моделирования гидравлических систем. Достаточно известен, но менее распространен в САПР *контурный метод*; в механике его называют *методом сил*.

Для эффективного применения некоторых методов численного интегрирования ОДУ эти уравнения удобно представлять в нормальной форме Коши:

$$\frac{d\mathbf{V}}{dt} = F(\mathbf{V}, t).$$

Для получения таких уравнений разработан специальный метод, называемый *методом переменных состояния*. Иногда применяется *табличный метод*, который сравнительно прост в реализации, но приводит к созданию модели большей размерности. Формальное сходство исходных компонентных и топологических уравнений в системах различной физической природы позволяет применять любой из названных методов для моделирования всей номенклатуры технических объектов.

Реализация в программе конкретного метода формирования ММ объекта может потребовать от пользователя выполнения дополнительных правил составления эквивалентных схем. Так, в классическом варианте метода узловых потенциалов имеются ограничения на вид компонентных уравнений. Применительно к схемной форме представления моделей эти ограничения выражаются в недопустимости таких ветвей, как идеальные источники типа потенциала, и любых ветвей, параметры которых зависят от каких-либо потоков. Использование метода переменных состояния подразумевает отсутствие в эквивалентной схеме контуров, состоящих только из емкостных ветвей (возможно, с источниками типа потенциала), и узлов, в которых объединяются только индуктивные ветви или только резистивные ветви (возможно, с источниками типа потока). Подобные ограничения можно снять использованием фиктивных ветвей.

Каждый метод имеет свои особенности, касающиеся формы описания исходных данных, но синтаксис входного языка описания объекта в большей мере определяется все же разработчиком программного комплекса, который и создает этот язык.

Итак, чтобы иметь возможность использовать в своей работе процедуры математического моделирования работы технических объектов на макроуровне пользователю САПР необходимо:

- уметь составлять эквивалентные схемы объектов;
- иметь программный комплекс для анализа систем;
- изучить особенности комплекса в части составления эквивалентных схем и язык комплекса.

Для выполнения моделирования пользователю достаточно:

- разработать эквивалентную схему объекта с учетом особенностей программного комплекса и возможностей его библиотеки ММ элементов;
- активизировать программу анализа;
- с помощью входного языка комплекса описать структуру объекта и параметры элементов схемы;
- с помощью входного языка описать задание на выполнение моделирования;
- обработать полученные результаты.

# ПРОГРАММНОЕ И ЛИНГВИСТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САПР

### 3.1. Общее программное обеспечение

Программное обеспечение САПР — это совокупность программ, представленных в заданной форме, вместе с необходимой программной документацией, предназначенной для использования в САПР.

Программное обеспечение ЭВМ представляет собой сложные программные комплексы с различными наборами функциональных возможностей. В первую очередь среди этих комплексов необходимо назвать операционные системы, которые поставляются, как правило, с ЭВМ и ориентированы на ее конкретную конфигурацию. По существу операционные системы являются как бы продолжением технических (аппаратных) средств ЭВМ и обеспечивают взаимодействие пользователя с вычислительной системой.

Разработка программного обеспечения для различных сфер использования вычислительной техники является исключительно трудоемким и дорогостоящим процессом, поэтому значительную часть стоимости аппаратно-программного комплекса для использования в САПР составляет программное обеспечение, которое представляет собой документы с текстами программ, программы на машинных носителях и эксплуатационные документы.

Программное обеспечение САПР делится на общесистемное (общее) и прикладное (специальное).

Общее программное обеспечение инвариантно к объекту проектирования; его основу составляют операционные системы, используемые в САПР и ЭВМ.

Специальное программное обеспечение полностью определяется классом объектов, проектируемых с помощью САПР; его основу составляют программы, реализующие алгоритмы отдельных проектных процедур.

Операционная система имеет модульную структуру, которая позволяет приспособить систему к конкретной конфигурации технических средств, так как отдельные программные компоненты могут быть включены в операционную систему по желанию пользователя.

Процесс настройки операционной системы на конкретную вычислительную систему и режимы ее использования называются

генерацией операционной системы. Основные программы операционной системы (ядро) постоянно находятся в оперативной памяти ЭВМ; редко используемые программы операционной системы с целью экономии оперативной памяти могут храниться на магнитных дисках и вызываться по мере необходимости.

Историю развития операционных систем удобно связать с поколениями вычислительных машин, которые сменяли друг друга в процессе достаточно бурной эволюции вычислительных систем.

Вычислительные машины первого поколения не имели операционных систем. Они выполняли программы, загруженные в оперативную память. Все действия по подготовке и загрузке программ, а также обеспечению программы необходимыми данными выполнялись пользователем вручную. Каждый пользователь для выполнения своей программы получал вычислительную машину в монопольное пользование. Таким образом, на ЭВМ первого поколения существовал только режим пакетной обработки одного пользователя.

Вместе с ЭВМ второго поколения появились и операционные системы. Первые из них выполняли мониторинговые функции для автоматического прогона набора (пакета) заданий. В них главное внимание уделялось автоматической смене заданий в процессе выполнения пакета.

В процессе развития операционных систем стали появляться мультипрограммные режимы работы для одновременного выполнения нескольких заданий. Сформировались понятия «задание» и «задача», соответственно как внешняя и внутренняя единицы работы вычислительной системы, и концепция диалогового режима, основанного на системах разделения времени.

Бурное развитие операционных систем началось с появлением ЭВМ третьего поколения, имеющих аппаратные поддержки для функционирования операционной системы: систему прерываний, средства защиты программ от взаимного влияния, каналы и процессоры ввода-вывода, несколько процессоров, развитые средства микропрограммирования. Окончательно сформировался режим пакетной обработки, позволяющий одновременно выполнять большое количество потоков заданий с высокой степенью автоматизации. Стали функционировать развитые системы разделения времени, обеспечивающие диалоговые режимы для большого числа терминалов пользователей в сочетании с пакетной обработкой. Развивались средства телеобработки данных, обеспечивающие удаленный доступ к вычислительной системе. Появились операционные системы, обеспечивающие режим реального времени, в котором функционируют системы управления объектами или процессами (конвейером, ракетой, технологическим процессом). Операционные системы при этом характеризовались универсальностью, многофункциональностью и многорежимностью, с одной стороны,

и отсутствием единой концептуальной основы и внутренней структурной упорядоченности — с другой. В результате операционные системы отличались громоздкостью, большими расходами на разработку, сопровождение, изучение и обслуживание.

Современный период развития операционных систем, предназначенных для обеспечения функционирования вычислительной техники, характеризуется развитием «демократических начал», ориентирующих их на малоподготовленных пользователей. Появилась концепция виртуального ресурса и виртуальной машины, которая является основой текущих и будущих операционных систем и обеспечивает их концептуальное и структурное единство. Получают свое развитие также сетевая телеобработка данных и средства комплексирования вычислительных систем. В сочетании с системами баз и банков данных они обеспечивают движение в направлении распределенной обработки данных.

Операционные системы занимают важнейшее место в совокупности современных системных программных средств, составляющих программное обеспечение ЭВМ. Они являются основой организации вычислительного процесса в системе и определяют эффективность использования ее аппаратных компонентов решения поставленных задач; от их возможностей зависит эффективность труда персонала, использующего и обслуживающего эту, пока еще достаточно дорогую, технику: инженеров, операторов, руководителей и исполнителей выполняемых с помощью ЭВМ проектов, администрации вычислительных центров.

Наиболее распространенным является определение операционной системы как набора программ (часто говорят — управляющих программ), предназначенных для управления ресурсами вычислительной системы. Иногда под назначением операционной системы подразумевают распределение и планирование ресурсов (или динамическое и статическое распределение ресурсов).

Таким образом, на первый план выдвигается проблема распределения ресурсов (рис. 3.1), причем под ресурсами понимают не

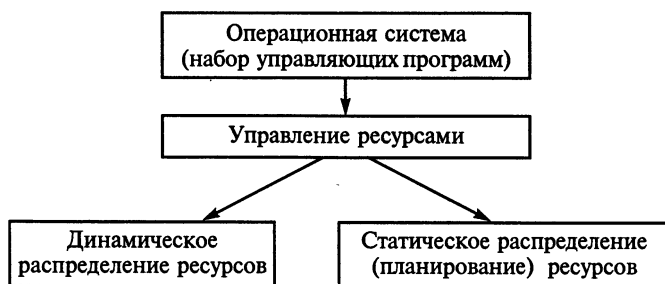


Рис. 3.1. Функции распределения ресурсов в операционной системе

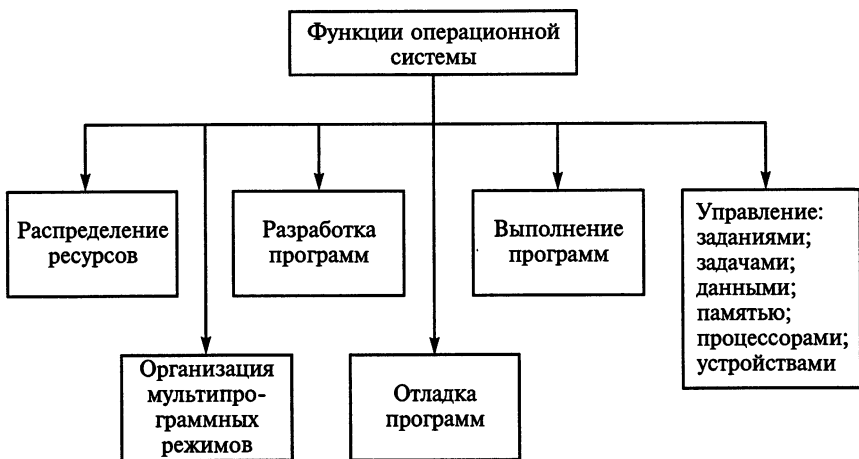


Рис. 3.2. Функции операционной системы

только традиционные виды ресурсов, такие, как время работы отдельных устройств (машинное время), пространство памяти разных уровней, функции отдельных устройств, наборы данных и, что бывает значительно реже, отдельные программы и программные комплексы, допускающие совместное использование, но иногда и человека (например, запрос на выполнение определенной функции оператором ЭВМ).

Другое определение операционной системы характеризуется функциональным подходом. В этом случае операционная система представляется перечислением функций, которые она должна выполнять (рис. 3.2).

К функциям операционной системы относят также обеспечение высоких показателей по двум важнейшим характеристикам вычислительных систем: эффективности и надежности (рис. 3.3).

Повышение эффективности подразумевает повышение эффективности использования аппаратных средств, выражающееся в ряде

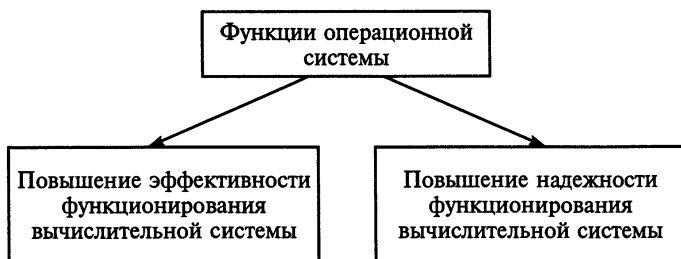


Рис. 3.3. Функции повышения эффективности и надежности

показателей (комплексным показателем является процент полезного машинного времени), повышение пропускной способности вычислительной системы, выражающееся в количестве выполненной работы (задач, заданий, программ и др.) в единицу времени, снижение системных издержек, повышение производительности труда программистов, проектировщиков, операторов и других участников вычислительного процесса, повышение удобства использования средств вычислительной техники.

Повышение надежности функционирования вычислительной системы как одна из функций операционной системы подразумевает наличие средств обеспечения достоверности получаемых результатов, уменьшения влияния сбоев и отказов аппаратуры, уменьшения времени восстановления работоспособности после сбоев и отказов, а также наличие средств для создания контрольных точек и повторения вычислений после сбоев с контрольной точки, средств резервирования данных, программ, процессов. В последнее время большое внимание уделяется повышению надежности программирования, что также находит отражение в функциях операционной системы.

Следует особо упомянуть о функциях операционных систем по обеспечению различных режимов использования вычислительной системы — режима пакетной обработки, диалоговых режимов, режима реального времени, режима разделения времени (рис. 3.4), а также о важных функциях по обеспечению различных категорий пользователей вычислительных систем: инженеров, операторов, административных работников и в первую очередь программистов и проектировщиков (рис. 3.5).

Большинство реально встречающихся в литературе определений операционных систем отличаются акцентами на те или иные стороны, упомянутые выше, и, как правило, характеризуются их комбинацией. По-видимому, наиболее полным определением операционной системы следует признать совокупность перечисленных выше характеристик.

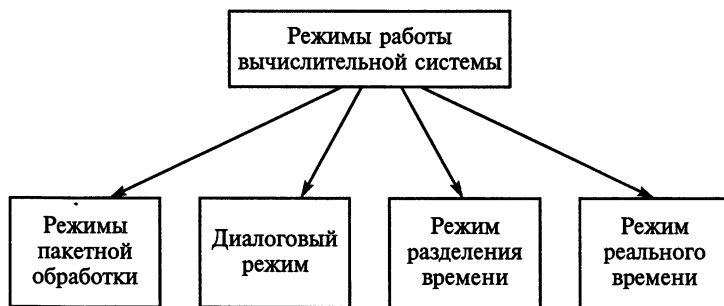


Рис. 3.4. Режимы работы, обеспечиваемые операционной системой



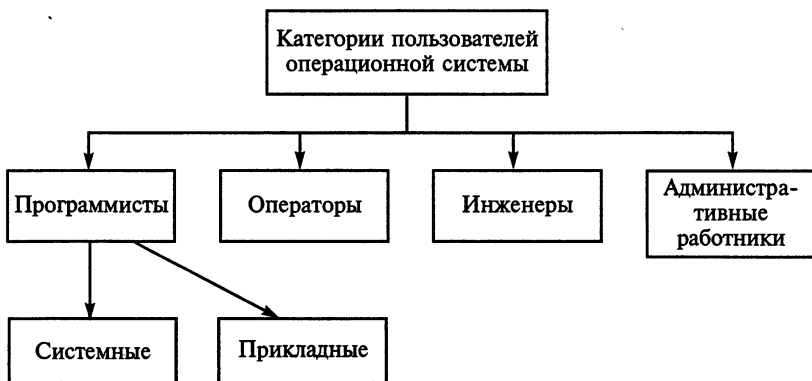


Рис. 3.5. Категории пользователей операционной системы

Следовательно, *операционная система* — это сложный многоцелевой и многофункциональный комплекс программ, являющийся составной частью практически всех современных вычислительных систем. Множество целей и функций, для реализации и выполнения которых предназначены операционные системы, перечислены выше.

При рассмотрении операционных систем следует упомянуть еще одну их функцию, а именно: операционная система является посредником (связующим звеном, интерфейсом) между ЭВМ, с одной стороны, и человеком (пользователем, программистом, оператором, инженером и т.д.) — с другой. Иначе говоря, операционная система — это логическое расширение функций аппаратуры в сторону человека. Она позволяет от «физического» уровня аппаратуры перейти к более высокому «логическому» уровню, который становится уровнем вычислительной системы и более удобен для человека.

Изложение настоящего материала ограничено универсальными общецельевыми операционными системами и не затрагивает специализированных.

**Состав операционной системы.** Компонентный состав операционной системы определяется набором функций, для выполнения которых она предназначена. Все ее программы можно разбить на две группы: управляющие и системные обрабатывающие (рис. 3.6).

Блок управляющих программ — неотъемлемый компонент любой операционной системы, это основная часть (ядро) операционной системы, без которого она не существует. Его функции — планирование прохождения непрерывного потока заданий, управление распределением ресурсов, реализация принятых методов организации данных, управление операциями ввода-вывода,



Рис. 3.6. Состав операционной системы

организация мультипрограммной работы, управление работоспособностью системы после сбоев и ряд других.

Блок управляющих программ состоит из ряда компонентов, среди которых следует выделить четыре основных:

- управление статическими ресурсами (управление заданиями);
- управление динамическими ресурсами (управление задачами);
- управление данными;
- управление восстановлением.

*Программы управления заданиями* осуществляют предварительное планирование потока заданий для выполнения и статическое распределение ресурсов между одновременно выполняемыми заданиями в процессе подготовки к выполнению (инициализации). К таким ресурсам обычно относят разделы памяти (основной, виртуальной, внешней), доступные для использования заданием, устройства, допускающие только монопольное использование, наборы данных и т. п.; эти ресурсы закрепляются за заданием или его частью с момента его инициализации до момента завершения и используются обычно в монопольном режиме.

*Программы управления задачами* осуществляют динамическое распределение ресурсов системы между несколькими задачами, решаемыми одновременно в мультипрограммном режиме для выполняемого потока заданий.

*Программы управления данными* обеспечивают все операции ввода-вывода (обмен между оперативной памятью и периферийными устройствами), исходящие как от программы пользователя, так и от программы операционной системы и реализуют различные структуры данных и возможность доступа к ним. Управление данными предназначено для выполнения следующих функций:

централизованное осуществление операций ввода-вывода (операций обмена) с использованием внешних (периферийных) устройств;

обеспечение хранения данных на устройствах внешней памяти; обеспечение различных способов организации и идентификации данных;

управление каталогом данных, позволяющее осуществлять их поиск, используя лишь символическое имя без указания местонахождения;

автоматическое распределение памяти на устройствах прямого доступа;

автоматический поиск данных по их символическому имени; обеспечение независимости программ от характеристик данных, которые они обрабатывают, и типов используемых внешних устройств;

обеспечение различных методов доступа к данным в зависимости от их организации и логического уровня способа доступа.

Все функции управления данными, перечисленные выше, можно разделить на два вида:

- управление процессами ввода-вывода в вычислительной системе;

- организация данных, хранимых в вычислительной системе; эта функция имеет отношение только к данным, хранимым на внешних запоминающих устройствах.

*Программы управления восстановлением* регистрируют машинные сбои и отказы и, если это возможно, восстанавливают работоспособность системы после сбоев.

*Системные обрабатывающие программы* выполняются под управлением блока управляющих программ, так же как любая обрабатывающая программа, в частности любая программа автоматизированного проектирования. Это значит, что они в полном объеме могут пользоваться услугами управляющих программ и не могут самостоятельно выполнять системные функции. Так, обрабатывающие программы не могут самостоятельно осуществлять собственный ввод-вывод. Операции ввода-вывода обрабатывающие программы реализуют с помощью запросов к управляющей программе, которая и выполняет непосредственно ввод или вывод данных. Централизованное выполнение системных функций управляющей программой позволяет выполнять их более эффективно и обеспечивает высокий уровень услуг для пользователя.

К системным обрабатывающим программам относятся программы, входящие в состав операционной системы: ассемблеры, трансляторы, сервисные программы обслуживания и ряд других. Трансляторы, компоновщик программ и загрузчик образуют основу систем программирования, построенных на базе операционной системы.

*Ассемблеры* — машинно-ориентированные языки низкого уровня и программные средства для их преобразования в язык машинных команд.

*Трансляторы* — программные средства, служащие для преобразования программы, написанной на одном из языков программирования высокого уровня (Fortran, Pascal, PL/1, C, C++ и др.) в язык машинных команд.

Как правило, операционные системы имеют ряд функций, не требующих дополнительного описания в разрабатываемых программах. Такие функции размещаются в специальном формате в библиотечных модулях (библиотеках) и по мере необходимости автоматически подключаются к пользовательским объектным модулям.

К *сервисным программам* можно отнести компоновщик программ, загрузчик, программы обнаружения неисправностей и т. п.

**Операционная система в процессе разработки программ.** В процессе реализации программных проектов операционная система используется на всех этапах проектирования программ, начиная от формулирования алгоритмов и структурных схем и заканчивая отладкой и выполнением программ.

Рассмотрим функции операционной системы при разработке программ с помощью систем программирования с использованием различных языков. В современных системах программирования широко применяется модульный принцип, позволяющий для одной системы разрабатывать параллельно и независимо большое число модулей, принадлежащих как одному, так и нескольким программным проектам. При этом операционная система поддерживает библиотеки модулей на различных стадиях разработки.

При разработке программ программист пользуется услугами операционной системы. Система программирования позволяет проводить программирование и отладку задачи по частям с использованием различных языков программирования. В состав системы программирования входит набор трансляторов, компоновщик программ (редактор связей) и загрузчик.

Для повышения эффективности программирования используются сегментация больших программ на менее крупные, их автономная разработка и объединение перед выполнением или в процессе его. Каждая программа представляет собой модуль. Несколько модулей могут быть объединены в более крупный модуль, кроме того, модули могут динамически вызывать друг друга.

Программы, написанные на одном из языков программирования, представляют собой исходные модули, тексты которых помещаются в наборы данных операционной системы. Набор данных, содержащий исходный модуль, представляет собой входные данные для соответствующего транслятора или ассемблера. В результате трансляции получается объектный модуль на машинном языке, непригодный пока для загрузки в оперативную память и выполнения. Специфика исходного языка в объектном модуле будет потеряна. Объектные модули состоят из текста в машинном коде, содержащего команды и константы программы, и управляющих словарей, содержащих информацию для объединения объектных модулей и последующей загрузки и настройки программ в оперативной памяти, и могут храниться во внешней памяти ЭВМ.

Отдельный программный модуль может содержать ссылку на другие программные модули. На некоторые величины данной программы могут быть ссылки и из других модулей. Величины в программе, которые осуществляют ссылку вне модуля и на которые производится ссылка извне, называются внешними символами. Информация о внешних символах (адрес, тип и т. д.) помещается в управляющем словаре внешних символов, который используется при объединении нескольких модулей в один для согласования разрешения внешних ссылок, т. е. ссылок модулей между собой. Формат объектных модулей — стандартный для всех трансляторов, используемых в данной операционной системе, что позволяет одинаково обрабатывать их редактором связей и объединять модули, написанные на разных языках программирования.

Объектный модуль проходит дальнейшую обработку — компоновку (редактирование связей) и загрузку в оперативную память. Компоновка заключается в увязке отдельных модулей в единую программу путем настройки связей между модулями и выполняется обслуживающей программой, называемой компоновщиком (редактором связей). Благодаря компоновщику пользователь может собирать программу из отдельных модулей, написанных в разное время и даже на различных алгоритмических языках. В результате получается программа, называемая загрузочным модулем.

Загрузочный модуль является перемещаемой программой и может быть помещен в любую область оперативной памяти. Различают простую и оверлейную структуры загрузочных модулей. Модуль имеет простую структуру, если все подпрограммы, входящие в него, одновременно загружаются в оперативную память и постоянно в ней находятся. Модуль называется оверлейным (с перекрытием), если в оперативной памяти постоянно находятся лишь некоторые подпрограммы.

Часть подпрограмм находится во внешней памяти (например, на магнитных дисках) и загружается в оперативную память лишь в

момент обращения к ним, причем разные подпрограммы могут загружаться на одно и то же место в оперативной памяти, перекрывая друг друга.

Планирование схемы перекрытий осуществляется на этапе компоновки загрузочного модуля. Применение оверлейных структур позволяет создавать большие и сложные программы в условиях ограниченного объема оперативной памяти ЭВМ. Большинство подпрограмм, применяемых в САПР, имеет оверлейную структуру.

После загрузки модуля управление передается в точку входа и начинается его выполнение.

**Режимы работы вычислительных систем.** Идея распределения ресурсов занимает центральное место при построении операционных систем на основе различных принципов. В идеале любая прикладная программа должна иметь возможность получать необходимые ресурсы, когда бы они ей не понадобились. Однако бывают ситуации, когда общая потребность в ресурсах превышает имеющиеся у системы возможности. В этом случае операционная система должна найти компромиссное решение и удовлетворить те запросы, которые она считает наиболее важными. С точки зрения пользователя (например, проектировщика, использующего САПР) можно выделить пакетный и диалоговый режимы работы вычислительной системы.

*Режим пакетной обработки задач* — режим выполнения определенной совокупности заданий, при котором эти задания обрабатываются в основном автоматически, без синхронизации с событиями, происходящими вне вычислительной системы, в частности без связи с лицами, представившими задание для выполнения.

Пакет заданий образует входной поток, который содержит их описание на языке управления заданиями; это описание может содержать входные наборы данных. Операционная система автоматически выполняет непрерывный поток заданий, что уменьшает необходимость ручного вмешательства оператора в процесс обработки.

При пакетной обработке каждая задача может использовать все ресурсы системы, за исключением основной памяти, занятой резидентной частью управляющей программы операционной системы. В этом режиме операционная система проверяет правильность задания, выделяет требуемые ему ресурсы, определяет программу, которая должна выполняться в данном задании, передает ей управление, следит за ходом выполнения программы до ее завершения (нормального или аварийного). При этом допускается совмещать выполнение программы одного задания с операциями ввода-вывода программы того же или другого задания. В случае, если устройство вывода занято, результаты размещаются в системной библиотеке, откуда они выводятся управляющей программой при освобождении устройства вывода.

Одним из недостатков режима пакетной обработки является «простаивание» ресурсов, которое возникает из-за того, что задания поступают на обработку в порядке их следования.

*Диалоговый (интерактивный) режим* — режим взаимодействия пользователя с вычислительной системой, при котором человек и вычислительная система обмениваются данными в темпе, соизмеримом с темпом обработки данных человеком.

Обмен информацией в диалоговом режиме осуществляется с помощью специальных диалоговых, с точки зрения пользователя, языков. Различают диалог пассивный и активный. При пассивном диалоге инициатива принадлежит вычислительной системе, она «ведет» за собой пользователя, требуя от него в точках ветвления вычислительного процесса дополнительной информации, необходимой для принятия заложенных в алгоритм решений. В этом режиме вычислительная система обеспечивает пользователя информационными сообщениями (в том числе результатами расчетов) и подсказками, облегчающими использование диалоговой системы. Запросы к пользователю строятся обычно либо в виде меню, либо в виде шаблонов. Меню представляет собой выводимый на экран дисплея список альтернативных возможностей продолжения решения в текущей точке вычислительного процесса, пользователь же должен выбрать какую-то одну возможность, указав соответствующую строку меню, например курсором или вводом с клавиатуры номера этой строки. Шаблоном называется форматизованный кадр изображения, выводимого на экран дисплея, содержащий тексты запросов к пользователю и специальные поля, предназначенные для занесения туда ответов пользователя в виде текстов или чисел. Активный диалог характеризуется равноправием его участников. Для организации такого диалога обычно используются формальные директивные (командные) языки или языки, близкие к естественным.

Диалоговый режим работы, обеспечиваемый современными вычислительными системами, в настоящее время используется в САПР чаще, чем режим пакетной обработки. Однако в тех случаях, когда время реакции системы на запрос пользователя слишком велико, диалоговый режим становится менее эффективным, поскольку человек не автомат и для организации цепочки прерванных длительным бездействием рассуждений ему требуется время. В этом случае предпочтителен режим пакетной обработки. Пакетный режим применяют также в простых случаях, когда существует возможность проектирования объекта без участия проектировщика (САПР, основанные на решении полностью формализуемых задач, — автоматическое проектирование).

С точки зрения организации работы вычислительных систем можно отметить однопрограммный и многопрограммный (мультипрограммный) режимы работы.

*Однопрограммный режим работы* — это режим, при котором ресурсы вычислительной системы одновременно используются только одним программным процессом.

*Мультипрограммный режим работы* — режим, в котором выполняется несколько программ одновременно. Современные вычислительные системы функционируют, как правило, в этом режиме. Для вычислительных систем с несколькими процессорами число одновременно выполняемых программ обычно существенно превосходит число используемых процессоров.

В режиме мультипрограммирования в основной памяти одновременно находится несколько программ, загруженных для выполнения. Число одновременно выполняемых программ определяет уровень мультипрограммирования. Центральный процессор в каждый момент времени может выполнять лишь одну программу. Таким образом, параллельно выполняемые программы конкурируют между собой за обладание ресурсами вычислительной системы, и в первую очередь за время центрального процессора. Каждая программа представляется в системе как задача (процесс). Таким образом, принято говорить об одновременно выполняемых задачах (или процессах), основой которых являются соответствующие программы, причем задача является единицей мультипрограммирования.

Задача, обладающая в текущий момент центральным процессором, называется активной. Она выполняется до тех пор, пока не окажется в состоянии ожидания какого-либо события (например, завершения операции ввода-вывода) или не будет прервана по каким-либо причинам операционной системой. Остальные задачи (кроме активной) находятся в состоянии готовности использования центрального процессора либо ожидания какого-либо события (завершения операции ввода-вывода, истечения заданного интервала времени, завершения выполнения какой-либо программы и т. д.). После перевода активной задачи в состояние ожидания выбирается одна из готовых для выполнения задач, которая становится активной.

Программы, одновременно претендующие на использование центрального процессора, в мультипрограммном режиме упорядочиваются по приоритетам. В случае конфликтов управление получает программа с наивысшим приоритетом. Она переводится в активное состояние. Остальные конкурирующие программы, находящиеся в состоянии готовности, получают управление, если программы с более высокими приоритетами окажутся в состоянии ожидания.

Основой мультипрограммирования является совмещение операций центрального процессора с операциями ввода-вывода. Такое совмещение возможно в связи с тем, что центральный процессор не занимается выполнением операций ввода-вывода, а толь-



ко инициирует их. После этого операции ввода-вывода выполняются каналами или процессорами ввода-вывода по самостоятельным программам параллельно с другими каналами и центральным процессором.

*Режим разделения времени* — мультипрограммирование, при котором ресурсы вычислительной системы предоставляются каждому процессу из группы процессов обработки данных, находящихся в вычислительной системе, на интервалы времени, длительность и очередность предоставления которых определяется управляющей программой этой системы с целью обеспечения одновременной работы в интерактивном режиме. Основным критерием оптимального планирования прохождения потока заданий является не эффективность использования вычислительной техники, а наименьшая продолжительность обслуживания одного пользователя при ограниченном сроке ожидания.

Механизм разделенного времени позволяет одновременно выполнять несколько задач путем выделения для каждой из них определенного кванта времени центрального процессора. Множество задач различных пользователей образует очередь заданий, которая размещается во внешней памяти ЭВМ (например, на магнитных дисках). В оперативной памяти для заданий выделяются специальные разделы. Задания из очереди последовательно вводятся в разделы оперативной памяти и обрабатываются в течение определенного кванта времени, по истечении которого задания «сворачиваются» и вновь помещаются во внешнюю память. Благодаря процессам свертки и развертки одновременно могут выполняться задания большого числа пользователей. При оптимальном размере кванта времени у каждого пользователя создается впечатление непрерывной работы с ЭВМ.

*Режим реального времени* — режим обработки данных, при котором обеспечивается взаимодействие вычислительной системы с внешними по отношению к ней процессами в темпе, соизмеримом со скоростью протекания этих процессов.

Режимы разделения времени и реального времени — наиболее важные для САПР режимы, при которых некоторое число пользователей одновременно имеют доступ к вычислительной машине с помощью удаленных терминалов.

Помимо использования стандартных операционных систем ЭВМ возможно создание специализированных операционных систем, ориентированных только на автоматизированное проектирование. Универсальные операционные системы имеют ряд функций, которые не востребованы САПР, что ведет к увеличению объема оперативной памяти, занимаемого ядром операционной системы, и времени, которое система тратит на выполнение этих невостребованных функций. Кроме того, в специализированной операционной системе могут отсутствовать некоторые системные об-

рабатывающие программы, например трансляторы, если ее работа связана только с обеспечением процесса автоматизированного проектирования и не связана с разработкой программ. В связи с этим применение специализированных операционных систем позволяет повысить производительность функционирования САПР и снизить уровень требований к системному программному обеспечению этих систем. Однако следует иметь в виду, что специализированная операционная система — очень сложный и дорогостоящий программный продукт, что ограничивает возможности ее повсеместного применения.

### 3.2. Специальное программное обеспечение

В системах автоматизированного проектирования, кроме программного обеспечения ЭВМ, может существовать системное программное обеспечение САПР — инвариантная часть программного обеспечения САПР, играющая роль связующего звена между операционной системой и программами автоматизированного проектирования.

Системное программное обеспечение САПР может быть использовано для разных САПР, близких по функциональному назначению, разработанных на основе общих концепций, например для САПР различных технических объектов. Системное программное обеспечение САПР разрабатывается с четкой ориентацией на конкретные ЭВМ и, следовательно, на используемые операционные системы. Оно является, по существу, продолжением операционной системы и обеспечивает:

- управление процессом вычислений;
- ввод, вывод, контроль и обработку информации;
- оперативную корректировку информации;
- хранение, поиск, анализ и защиту информации;
- диалоговую связь пользователя с ЭВМ в процессе проектирования;
- контроль и диагностику выполняемых проектных операций.

Системное программное обеспечение САПР в соответствии с перечисленными функциями включает в себя:

- управляющую программу;
- программу ввода и обработки входной информации;
- трансляторы проблемно-ориентированных входных языков;
- системы или программы управления базами данных;
- программы обработки и выдачи проектной документации;

включая чертежи.

При создании конкретных САПР может возникнуть необходимость и в других программах, реализующих те или иные функции. Но в каждом случае при разработке системного программного

обеспечения САПР учитывается технология проектирования изделий или группы изделий, требования и состав специального программного обеспечения, структура и объем переработки информации и т. д.

Собственно, специальное программное обеспечение САПР, как отмечалось выше, предназначено для выполнения автоматизированного проектирования объекта.

В состав специального программного обеспечения САПР включаются прикладные программы или пакеты прикладных программ, ориентированные на решение конкретных задач, и сервисные программы, выполняющие в рамках САПР сервисные функции и обеспечивающие использование полученных результатов.

Таким образом, специальное программное обеспечение САПР направлено на решение определенного круга проектных задач. Именно это программное обеспечение является в первую очередь предметом разработки системы. От его характеристик и особенностей определяющим образом зависят функциональные возможности САПР.

Специальное программное обеспечение не может функционировать без общесистемного программного обеспечения, поэтому его можно рассматривать как узкоспециализированное продолжение общесистемного обеспечения ЭВМ и САПР.

Решение задач автоматизированного проектирования сложных объектов, как правило, невозможно организовать в рамках одного программного модуля. Как отмечалось в гл. 1, при создании САПР сложных объектов необходимо использовать блочно-иерархический подход, в основу которого положена декомпозиция (разбиение) сложной задачи на ряд более простых. Похожий принцип заложен и в организацию специального программного обеспечения САПР. Модульный принцип программирования предполагает разбиение проекта на отдельные программные модули в начальной стадии проектирования и объединение независимо разработанных программных модулей на конечной стадии.

Программа, предназначенная для решения задачи, делится на части или модули, исходя из удобства разработки и использования программ. Каждая из них может разрабатываться с использованием различных языков и средств операционной системы. После подготовки исходного текста программы каждый модуль обрабатывается транслятором, который вырабатывает объектные модули, а затем компоновщиком, который из объектных модулей формирует загрузочные.

Удобство модульного построения состоит еще и в том, что в настоящее время существует достаточно большое количество стандартных программ, реализующих различные математические методы и типовые проектные процедуры. Такие программы, как правило, существуют в виде библиотек объектных модулей, и их ис-

пользование существенно сокращает время и уменьшает стоимость разработки программного обеспечения.

Структура загрузочного модуля может быть простой и динамической.

В простых структурах один загрузочный модуль содержит все коды, необходимые для выполнения задачи, т.е. этот модуль не передает управление никакому другому модулю. Он загружается в память ЭВМ как единое целое. При этом несущественно, был ли модуль единым или получен объединением нескольких модулей. Загрузочный модуль простой структуры не использует динамически другие загрузочные модули. Однако программы простой структуры можно динамически вызывать из других (динамических) загрузочных модулей. Программы простой структуры эффективны по скорости выполнения, так как не требуют дополнительной загрузки других программ. В случае, когда задача слишком велика и все ее программы нельзя загрузить в оперативную память ЭВМ одновременно, вместо простой структуры следует использовать динамическую.

Динамические структуры не требуют одновременного присутствия в памяти всех модулей программы при выполнении. Модули динамически загружаются в память, когда это требуется. Различают динамические последовательные и динамические параллельные структуры. В динамических последовательных структурах вызываемый динамически модуль выполняется последовательно; это значит, что вызывающий модуль не выполняется до завершения выполнения вызванного модуля. В динамических параллельных структурах вызываемый и вызывающий модули выполняются параллельно.

Любая САПР может иметь различные принципы организации от простейшей, в виде пакета типа библиотеки прикладных программ, до сложной, использующей общесистемные программные средства управления базами данных, проблемно-ориентированные языки и т.д.

В любом случае создание и использование программного обеспечения САПР требует соблюдения определенных основополагающих принципов, одним из которых является принцип модульности системы.

Модуль специального программного обеспечения САПР можно рассматривать как совокупность определенных программных и операционных компонентов, которыми являются (при иерархическом разложении, начиная снизу):

операция ЭВМ;

макрооперация, представляющая типовую совокупность нескольких операций;

стандартная программа;

подпрограмма решения функциональной подзадачи;

функциональная программа;  
набор функциональных программ.

Преимущества модульного построения системы рассмотрены выше. Следует отметить, что при модульном построении комплекса программ длительность их исполнения может возрасти.

На рис. 3.7 представлена возможная структура САПР объекта машиностроения, работающая под управлением базовой операционной системы ЭВМ.

Программное обеспечение в этом случае представлено в форме проблемно-ориентированного пакета прикладных программ, работающего под управлением базовой операционной системы. Пакет программ состоит из управляющей программы, лингвистического процессора, прикладных и обслуживающих программ. В памяти ЭВМ постоянно находится основная часть управляющей программы — корневой сегмент (монитор). Все другие программы загружаются в память и выполняются последовательно по мере необходимости.

Управляющая программа предназначена для управления общим ходом работы и после ввода пароля вырабатывает сообщение системе о допуске к работе с теми или иными программами. Корневой сегмент управляющей программы формирует данные, необходимые базовой операционной системе для вызова той или иной части пакета и передачи ей управления. Например, пользователю-проектировщику разрешается заносить данные только в определенные файлы, управление которыми осуществляется соответствующими программами. Программа организации базы данных может пользоваться только службой администрирования системы. Информация пользователя сначала обрабатывается анализатором программ, который выявляет наличие в базе данных программ, справочных или определенных ранее данных, необходимых для работы. После этого формируется пакет заданий, который через управляющую программу передается базовой операционной системе для вызова обслуживающих или прикладных программ пакета.

Лингвистический процессор предназначен для обработки сообщений, написанных на внешних языках системы и состоит из препроцессора, осуществляющего синтаксический и семантический контроль входной информации; транслятора (интерпретатора) с внешних языков системы; программ диагностики, осуществляющих логический контроль входной информации и выдачу сообщений об ошибках; программ управления файлом лингвистического процессора, обеспечивающих ввод-вывод сообщений САПР, запросов пользователя и сопоставление базовой и входной информации.

Рассматриваемая САПР состоит из комплексов программ и отдельных модулей, предназначенных для выполнения процедур

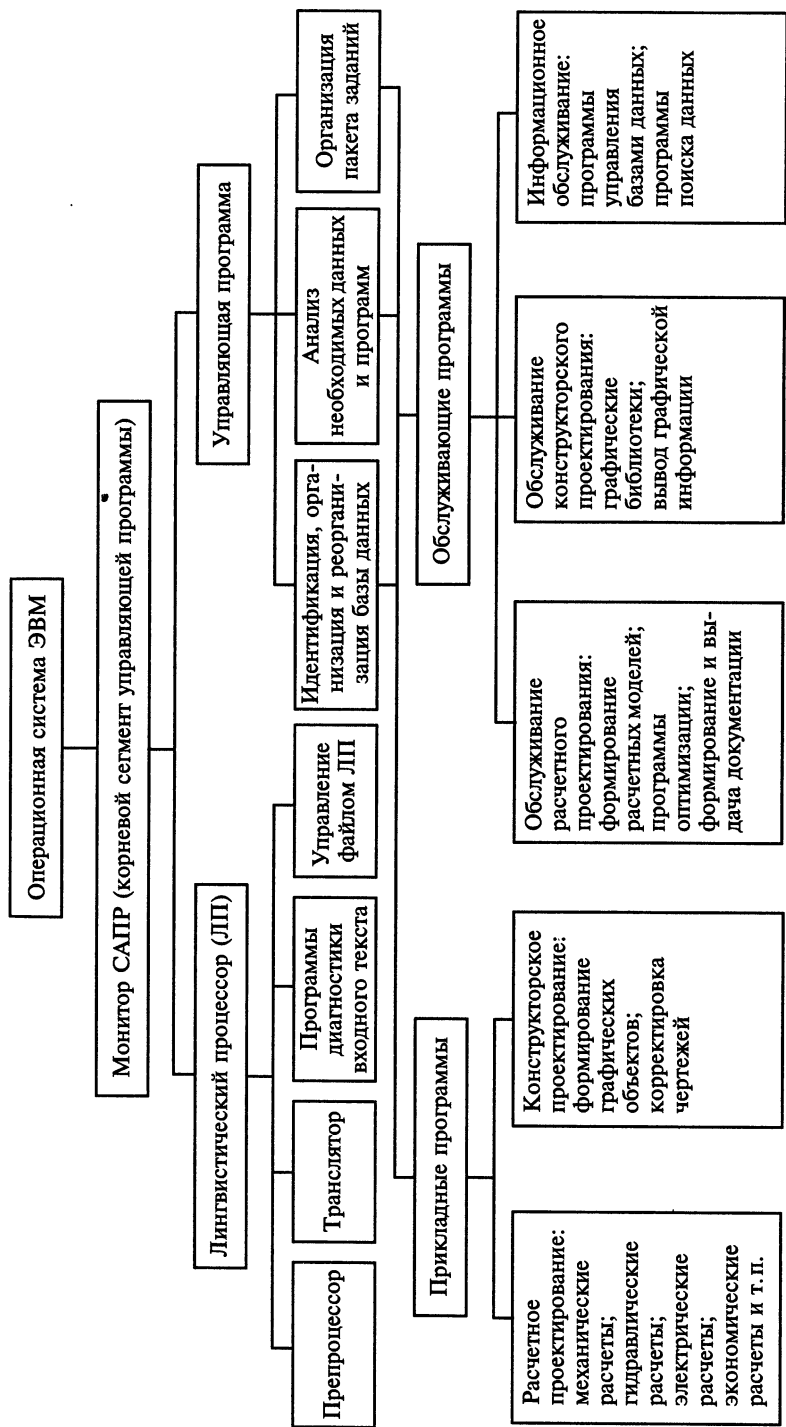


Рис. 3.7. Пример структуры программного обеспечения САПР

расчетного и конструкторского проектирования, т.е. отдельные комплексы пакета сами могут являться самостоятельными программными продуктами.

В САПР комплексы программ прочностных, тепловых, гидравлических расчетов являются пакетами взаимосвязанных программ простой структуры с управляющими программами; программы остальных пакетов могут вызываться независимо друг от друга. Управляющие программы могут работать по-разному в зависимости от значений управляющих параметров, заданных пользователем. Управляющие параметры определяют тип применяемой математической модели, форму входной и выходной информации, признак режима работы (режим диалогового проектирования, автоматического проектирования или тестирования). Настройку управляющих параметров и организацию оптимизационного процесса осуществляют обслуживающие программы, которые работают под управлением администратора САПР.

При создании специального программного обеспечения САПР большое значение в последнее время приобретают комплексные технологии проектирования, программирования, отладки и сопровождения программ. Они заметно отличаются от так называемого традиционного программирования, базирующегося исключительно на использовании свойств и особенностей соответствующего языка программирования, но тем не менее используют модульный принцип построения программных систем. Каждая технология ограничивает поле деятельности программиста, дисциплинируя его усилия в направлении создания наиболее эффективной и надежной программы.

### 3.3. Классификация и использование языков в САПР

Лингвистическое обеспечение конструкторского и технологического проектирования должно учитывать, кроме общих требований, возможность комплексного использования конструкторской и технологической информации (текстовой и графической) для обеспечения диалогового режима проектирования и автоматизации оформления проектной документации.

В соответствии с классификацией языков САПР, представленной на рис. 3.8, различают языки программирования и проектирования.

**Языки программирования.** Для написания программного обеспечения предназначены языки программирования; они являются средством разработчика САПР.

К языкам программирования предъявляют требования удобства использования, универсальности и эффективности объектных программ (программ, полученных после трансляции на ма-

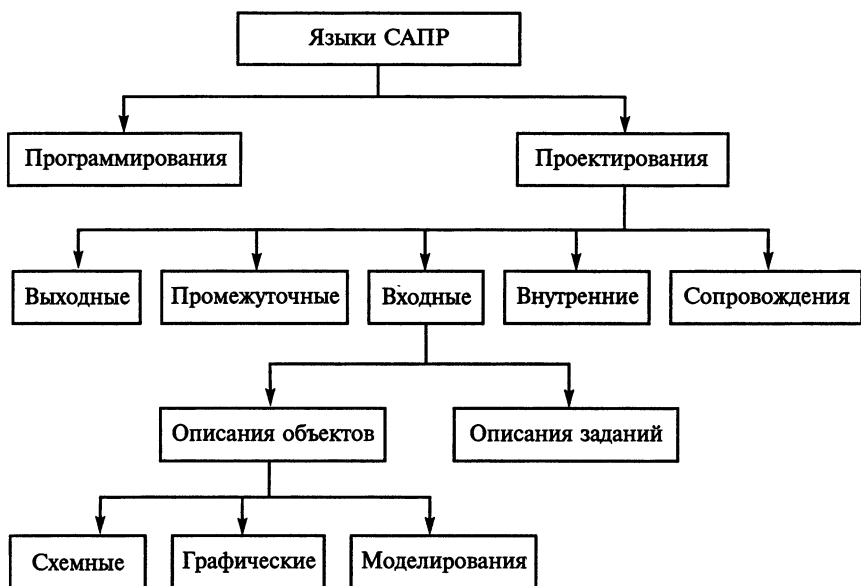


Рис. 3.8. Классификация языков САПР

шинный язык). Удобство использования выражается в затратах времени программиста на освоение языка и главным образом на написание программ на этом языке. Универсальность определяется возможностями языка для описания разнообразных алгоритмов, характерных для программного обеспечения САПР, а эффективность объектных программ — свойствами используемого транслятора, которые, в свою очередь, зависят от свойств языка. Эффективность оценивается затратами машинного времени и памяти на исполнение программ.

С позиций универсальности и эффективности объектных программ наилучшими свойствами обладают машинно-ориентированные языки, которые называют языками ассемблера, или автокодами. Однако языки ассемблера неудобны для человека, так как их использование снижает производительность труда программистов. Они применяются для разработки лишь тех модулей программного обеспечения САПР, которые требуют для своего исполнения больших вычислительных ресурсов, существенно влияющих на общие затраты машинного времени и памяти.

Среди алгоритмических языков высокого уровня, созданных на ранних этапах развития вычислительной техники, наибольшее распространение получил язык Fortran. Его сравнительная простота обуславливает легкость освоения и достаточно высокую эффективность объектных программ при решении задач численного анализа. Программное обеспечение многих ранее разработанных и



существующих САПР создано в основном с использованием этого языка программирования.

Однако язык Fortran имеет ограниченные возможности для описания сложных алгоритмов логического характера, поэтому при создании таких программ, как мониторы или языковые процессоры, используют либо языки ассемблера, либо языки высокого уровня с более развитыми возможностями описания невычислительных процедур (Pascal, PL/1, ADA, C, C++).

**Языки проектирования.** Для описания информации об объектах и задачах проектирования предназначены языки проектирования. Большинство из них относится к средствам пользователя САПР. Среди языков проектирования выделяют языки входные, выходные, сопровождения, управления, промежуточные и внутренние.

Входные языки служат для задания исходной информации об объектах и задачах проектирования и включают в себя языки описания объектов и языки описания заданий. Первые предназначены для описания свойств проектируемых объектов, вторые — для описания заданий на выполнение проектных операций и процедур.

Из рис. 3.8 видно, что языки описания объектов делятся в свою очередь на языки схемные, графические и моделирования. Они используются для описания исходной информации, представленной в виде некоторой схемы, конструкторского чертежа или алгоритма функционирования. Схемные языки широко применяют при описании принципиальных электрических схем в подсистемах проектирования электронных устройств и функциональных схем — в подсистемах функционально-логического проектирования ЭВМ. Графические языки являются основой лингвистического обеспечения в подсистемах геометрического моделирования и машинной графики. Языки моделирования широко используются в подсистемах имитационного моделирования систем массового обслуживания.

Выходные языки используются для выражения результатов выполнения проектных процедур на ЭВМ.

Языки сопровождения применяются для корректировки и редактирования данных при выполнении проектных процедур.

Языки управления служат для представления управляющей информации для программно-управляемого исполнительного оборудования, например устройств документирования и технологических автоматов.

Промежуточные внутренние языки предназначены для представления информации на определенных стадиях ее переработки в ЭВМ. Достоинство этих языков состоит в том, что в отличие от входных языков, характеризующихся большим разнообразием, узкой проблемной ориентацией и изменчивостью при адаптации САПР к изменяющимся условиям, они являются унифицирован-

ными и более универсальными. Недостаток узкоспециализированных языков состоит в необходимости существенной перестройки связанной с ними программной системы при изменении условий проектирования. Недостаток универсальных языков связан с их громоздкостью и, следовательно, с неудобствами применения конечным пользователем.

Устранение указанных выше недостатков выполняется в программных системах, структура которых поясняется рис. 3.9. При этом вводятся универсальный промежуточный язык, отражающий особенности широкого класса проектируемых объектов, и узкоспециализированные входные языки, ориентированные на разработчиков объектов из некоторых подклассов проектируемых объектов.

Пользователь составляет описание на входном языке, это описание с помощью специальной транслирующей программы, называемой конвертором, переводится на промежуточный язык. Далее работает основной транслятор, переводящий описание задачи с промежуточного языка в объектную рабочую программу. Преимущество двухуровневого лингвистического обеспечения, показанного на этой схеме, заключается в том, что программная система сравнительно легко настраивается на новые подклассы объектов. Для включения в систему нового входного языка достаточно разработать только конвертор с этого языка на промежуточный язык. Наиболее сложная часть системы — основной языковый процессор — остается без изменений.

Внутренние языки появляются в результате стремления унифицировать формы представления информации внутри ЭВМ, что

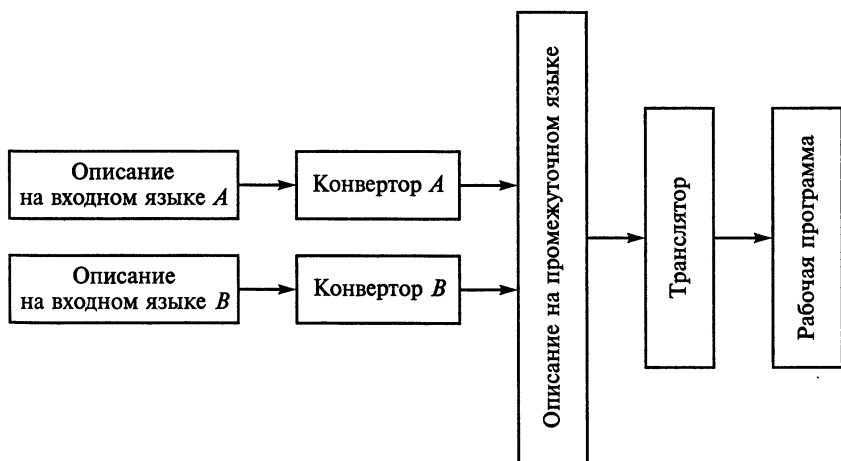


Рис. 3.9. Схема двухуровневого лингвистического обеспечения

облегчает решение проблем информационного согласования различных программ в САПР.

**Языки процедурные и непроцедурные.** Языки проектирования, предназначенные для описания развивающихся во времени процессов, обычно оказываются близкими к языкам описания алгоритмов и называются процедурными языками. Языки проектирования, предназначенные для описания структур проектируемых объектов, называются непроцедурными языками. Как правило, для пользователя САПР непроцедурные языки более удобны. С их помощью непосредственно описывается исходная схема или чертеж, пользователю нужно лишь соблюдать правила языка, не заботясь о разработке моделирующего алгоритма. Формальный характер перевода исходного рисунка схемы в текст на непроцедурном языке облегчает разработку программного обеспечения интерактивных графических систем, в которых исходная информация об объекте формируется пользователем непосредственно в виде рисунка на экране дисплея. Кодирование изображения в соответствии с правилами непроцедурного языка осуществляется автоматически.

**Диалоговые языки.** Важное значение для САПР имеют диалоговые режимы работы пользователя с ЭВМ. Лингвистическое обеспечение диалоговых режимов представляется диалоговыми языками. Фактически диалоговый язык объединяет в себе средства языков входного, выходного и сопровождения и служит для оперативного обмена информацией между человеком и ЭВМ. Различают пассивный и активный диалоговые режимы и соответственно активные и пассивные диалоговые языки.

В пассивном диалоговом режиме инициатива диалога принадлежит ЭВМ. Прерывания вычислительного процесса в нужных местах выполнения процесса проектирования и обращения к пользователю осуществляются с помощью диалоговых программных средств, воплощенных в мониторной системе САПР. Обращения ЭВМ к пользователю могут быть следующих типов: запрос, информационное сообщение, подсказка.

Запросы предусматриваются в тех случаях, когда от человека требуется либо задание исходных данных, либо выбор между ограниченным множеством возможных продолжений проектирования. При запросе исходных данных ЭВМ высвечивает на экране дисплея шаблон, состоящий из вопроса и пустых позиций, в которые пользователь должен поместить запрошенные числа или фразы. При запросе варианта дальнейшего проектирования на экране высвечивается меню — вопросы и несколько поименованных вариантов ответа. Пользователю достаточно указать имя выбранного из меню ответа.

Информационные сообщения используются для передачи пользователю промежуточных и окончательных результатов реше-

ния, сведений о состоянии его задания. На эти сообщения не требуется реакция пользователя.

Подсказки применяются в тех случаях, когда действия пользователя ошибочны, например нарушаются формальные правила языка.

В активном диалоговом режиме инициатива начала диалога может быть двусторонней — возможности прерываний вычислительного процесса имеются и у ЭВМ, и у пользователя. Активные диалоговые языки могут быть близкими к естественному языку человека, но с ограниченным набором возможных слов и фраз. Для активного диалога требуется существенно более сложное программное обеспечение, чем для пассивного.

### **3.4. Языковые средства машинной графики**

Лингвистическое обеспечение процесса геометрического проектирования предназначено для обеспечения проектировщика мощными и удобными средствами формирования, хранения, преобразования и ввода-вывода графической информации.

Комплекс программ отображения геометрической информации включает в себя программы преобразования входной информации в команды управления устройствами отображения.

В него входят:

программы-трансляторы, преобразующие исходную информацию в форму, удобную для машинной обработки;

программы графических операций, к которым обращаются оттранслированные проблемные программы; примерами графических операций являются перевод чертежа или его части в другой масштаб, нанесение штриховки в разрезе или сечении, формирование размера с нанесением выносных и размерных линий, стрелок и т. д.;

программы создания архива графических документов, которые позволяют использовать типовые графические фрагменты в процессе работы проблемных программ с целью включения их в изображение объекта; полученный чертеж объекта в целом может быть помещен в графический архив для последующего неоднократного использования;

проблемные программы, обеспечивающие воспроизведение чертежа или его фрагмента;

управляющая программа, осуществляющая связи между программами и управление чертежно-графической подсистемой.

Сформированная проблемными программами информация содержит все необходимые для автоматического отображения данные, которые можно непосредственно преобразовать в программы управления устройствами отображения, но делать это нецелесообразно по следующим причинам:

- в решении различных задач одни и те же программы графических операций реализуются в различной последовательности, и только после выполнения последней программы информацию необходимо преобразовать в команды графопостроителя или дисплея;

- результат может отображаться на нескольких графических устройствах, имеющих различные системы команд управления; например, сформированный чертеж можно сначала отобразить на экране дисплея, визуально проверить, а затем дать указание вычертить рабочий чертеж на графопостроителе;

- отображение может производиться по частям или полностью, непосредственно после формирования графического элемента или с любой задержкой по времени;

- набор программ графических операций и системы команд устройств отображения могут со временем изменяться, при этом комплекс проблемных программ машинной графики не должен подвергаться коренным изменениям;

- взаимодействие ЭВМ с различными графическими устройствами является сложным процессом, в котором наряду со специфическими для каждого устройства действиями есть много общих черт.

В связи с этими соображениями результаты работы проблемных программ записываются на некотором промежуточном языке, который еще не является языком команд устройств отображения, но позволяет при необходимости быстро преобразовать данные в команды управления графопостроителя или дисплея. Такое преобразование осуществляется программным блоком постпроцессоров.

Кроме программы постпроцессора, существует программный блок связи ЭВМ с устройствами отображения. Именно эти программы приходится в большей или меньшей степени дорабатывать при включении в подсистему графических устройств новых средств отображения. Все остальные программные средства являются инвариантными по отношению к комплексу технических средств машинной графики. Такая структура программного обеспечения делает ее гибкой и быстро приспособляемой к составу конкретной САПР.

Следует отметить, что общесистемное программное обеспечение машинной графики базируется на общесистемном программном обеспечении САПР (см. подразд. 3.1) и должно включать в себя либо специальные пакеты программ, либо специализированные языки для описания и ввода графической информации с соответствующими трансляторами.

Языковые средства машинной графики удобно рассматривать в тесной связи с историей их возникновения.

Первые попытки создания и использования программного обеспечения для формирования и ввода-вывода графической инфор-

мации относятся к периоду возникновения устройств отображения графической информации. В это же время созданы программные средства машинной графики первого поколения. Они представляют собой пакеты подпрограмм, вызываемых из основной программы, написанной на одном из алгоритмических языков высокого уровня и предназначенных для формирования графического изображения какого-либо элемента (графического примитива) или выполнения заданного графического преобразования. Объектные модули этих подпрограмм сведены в специальный библиотечный файл, откуда могут быть вызваны при компоновке основной программы. Элементами языка в этом случае являются наименования указанных подпрограмм. Языковые средства машинной графики первого поколения реализуют двумерную (плоскую) графику.

Наибольшую известность среди отечественных программных средств первого поколения получил пакет ГРАФОР. Он позволяет вычерчивать графические примитивы (вектор, дуга, текст, различного вида маркеры). На подпрограммах формирования примитивов основаны программные средства, предназначенные для отображения плоских изображений (программы разметки числовых осей, аффинных преобразований на плоскости, штриховки и т. д.). В пакете ГРАФОР, несмотря на его ориентацию на плоскую графику, реализованы алгоритмы проекционной машинной графики (аффинные преобразования в трехмерном пространстве, вычерчивание пространственных кривых и т. п.).

Использование ГРАФОРА и других языковых средств первого поколения в целях автоматизированного проектирования не удобно, поскольку для получения и редактирования изображения каждый раз необходимо либо писать, либо редактировать программу на языке высокого уровня, транслировать ее, компоновать с графической библиотекой и только после запуска и окончания ее работы получать требуемое графическое изображение.

К языковым (программным) средствам машинной графики второго поколения относятся также пакеты подпрограмм, реализующих формирование графических изображений. Принципиальная разница по сравнению с программными средствами машинной графики первого поколения заключается в том, что основой для пакетов программ второго поколения является пространственная (трехмерная) графика. В остальном языковые средства этих поколений схожи. В настоящее время языковые средства машинной графики первого и второго поколений с целью автоматизации проектирования не применяются. Область их использования ограничивается в основном обработкой и графическим представлением результатов расчетов и экспериментов.

К современным средствам машинной графики, составляющим основу большинства САПР, относятся языковые средства тре-

твого поколения, реализующие интерактивную машинную графику. Процесс формирования и преобразований графического изображения ведется на экране графического дисплея в режиме диалога проектировщика с графической системой. При этом формируются графические изображения объектов любой сложности в трехмерном пространстве. Элементами языка в этом случае являются проблемно-ориентированные меню, управляемые перемещением курсора. Подробно эти программные средства рассмотрены в гл. 6.

# ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ САПР

### 4.1. Общие сведения

Одним из условий успешного функционирования САПР является наличие необходимой информации.

Под информационным обеспечением САПР понимаются документы, содержащие описания стандартных проектных процедур, типовых проектных решений, типовых элементов, комплектующих изделий, материалов и другие данные, а также файлы и блоки данных на машинных носителях с записью указанных документов.

Основной задачей информационного обеспечения процессов проектирования является своевременная выдача источнику запроса полной и достоверной информации, приводящей к выполнению определенной части процесса проектирования.

Исходя из этого к информационному обеспечению САПР предъявляются следующие требования:

- обеспечение информацией как автоматизированных, так и ручных процессов проектирования;
- хранение и поиск информации, представляющей результат ручных и автоматизированных процессов проектирования;
- достаточный объем хранилищ информации. Структура системы должна допускать возможность наращивания емкости памяти вместе с ростом объема информации, подлежащей хранению, одновременно должны обеспечиваться компактность хранимой информации и минимальный износ носителей информации;
- быстрдействие системы информационного обеспечения, благодаря которой время получения информации проектировщиками гораздо меньше времени, необходимого для получения этой же информации традиционными средствами;
- возможность быстрого внесения изменений и корректировки информации, доведения этих изменений до потребителя и получения твердой копии документа.

При создании информационного обеспечения основная проблема заключается в преобразовании информации, необходимой для выполнения проектно-конструкторских работ над данным классом объектов, в приемлемую и наиболее рациональную для машинной обработки форму.



Основу информационного обеспечения САПР составляет совокупность всевозможных данных, которые используются в процессе проектирования. При проектировании сложного объекта, в котором участвует более одного проектировщика, данные, необходимые каждому из них, должны быть легко доступны одновременно.

*Данные* — это сведения о некоторых фактах, позволяющие делать определенные выводы. Взаимосвязанные данные часто называют системой данных, а хранимые данные — информационным фондом. Основное назначение информационного обеспечения — предоставлять пользователям САПР достоверную информацию в необходимом им виде.

На заре развития вычислительной техники данные хранились на перфоленте или перфокартах, которые могли считываться электронным оборудованием. Для обработки данных существовали специальные прикладные программы, представлявшие собой независимые программные единицы и не являвшиеся частью более сложной системы, которую в современных условиях представляет САПР.

Со временем оборудование и используемые для эксплуатации системы электронной обработки данных процедуры существенно усложнились. В качестве информационных носителей в настоящее время преимущественно используются магнитные и оптические носители, отличающиеся большей на несколько порядков плотностью записи. Стало возможным использовать как независимые информационные системы, так и являющиеся подсистемами более сложных систем.

Главная цель информационного обеспечения осталась прежней — предоставление по запросу заинтересованного пользователя достоверной информации в определенное время.

Чтобы понять процесс электронной обработки данных, необходимо знать ряд терминов, которые применяются при описании и представлении данных.

*Предметная область* может относиться к любому типу организации (банк, университет, больница или завод) или автоматизированной системы. Для предметной области САПР может потребоваться информация, характерная для данной области проектирования.

*Объектом* может быть любой предмет, событие, понятие и т. п., о котором приводятся данные.

Каждый объект характеризуется рядом основных *атрибутов*. Например, конструкционный материал характеризуется плотностью, прочностными параметрами, тепло- и электропроводностью и т. п., автомобиль можно характеризовать такими атрибутами, как максимальная скорость, динамика разгона, расход топлива, масса, габариты и др.

Атрибут называют также элементом данных, полем данных или просто полем.

Сведения, содержащиеся в каждом атрибуте, называют значениями данных. *Значения данных* представляют действительные данные, содержащиеся в каждом элементе данных; ими могут быть конкретные величины упомянутых выше характеристик конструкционного материала.

Среди атрибутов имеются такие, по значениям которых возможна идентификация объекта. Атрибуты, по значениям которых определяют значения других атрибутов, называют идентификаторами объекта, или *ключевыми элементами данных*. Отметим, что один и тот же объект могут идентифицировать несколько элементов данных. Их тогда считают кандидатами в идентификаторы. Проблему выбора идентификатора из нескольких кандидатов решает пользователь САПР. Например, зная какую-либо из характеристик конструкционного материала, можно определить вид этого материала, его свойства.

Совокупность значений связанных элементов данных образует *запись данных*. В приведенном выше примере с конструкционным материалом такими элементами данных являются конкретное его обозначение и численные значения его свойств.

Упорядоченную совокупность записей данных называют *файлом данных*, или *набором данных*.

## 4.2. Банки данных

Наиболее высокой формой организации информационного обеспечения для больших систем являются банки данных, представляющие собой совокупность средств для централизованного накопления и коллективного использования данных в САПР. Банк данных является проблемно-ориентированной информационно-справочной системой, которая обеспечивает ввод необходимой информации, автономное от конкретных задач ведение и сохранение информационных массивов и выдачу необходимой информации по запросу пользователя или программы. Банк данных может быть определен как система программных, языковых, организационных и технических средств, предназначенных для хранения и многоцелевого использования информации.

Банки данных должны обеспечивать:

- сокращение времени поиска данных;
- многократность использования данных;
- простоту и удобство обращения к данным пользователей;
- достоверность хранения данных.

Основными частями банка данных является база данных (БД), представляющая собой систематизированные взаимосвязанные

совокупности данных, и система управления базами данных (СУБД), обеспечивающая необходимые манипуляции с информационными массивами.

Множество данных, которые потенциально могут использоваться при функционировании САПР или являются запоминаемым результатом ее работы, образует БД системы.

*База данных* — сами данные, находящиеся в запоминающих устройствах ЭВМ и структурированные в соответствии с принятыми в данном банке данных правилами.

*Система управления базами данных* — совокупность программных средств, обеспечивающих функционирование банка данных. С помощью СУБД производятся запись данных в БД, их выборка по запросам пользователей и прикладных программ, обеспечивается защита данных от искажений и несанкционированного доступа и т. п.

Файлы данных чаще всего специально создаются для использования конкретными программами (подпрограммами), которые реализуют ввод данных из файла в строго определенной последовательности.

К недостаткам размещения данных, необходимых предметной САПР, в файлах данных можно отнести следующие.

*Избыточность данных.* Некоторые элементы данных, необходимых САПР, неизбежно используются во многих прикладных программах. Поскольку данные требуются нескольким прикладным программам, они часто записываются в несколько файлов, при этом одни и те же данные хранятся в разных местах. Такое положение называют избыточностью данных. Оно делает проблематичным обеспечение непротиворечивости данных и обуславливает еще один недостаток — *сложность в управлении*.

*Проблемы непротиворечивости данных.* Одной из причин нарушения непротиворечивости данных является их избыточность, что связано с хранением одной и той же информации в нескольких местах. При необходимости обновления информации ее нужно изменить во всех файлах, что бывает затруднительно. В результате об одном и том же объекте предметной области в разных местах хранится различная информация.

*Ограничения по доступности данных.* В современных условиях лицо с соответствующими правами доступа должно иметь возможность получить данные за приемлемый отрезок времени. Если же данные содержатся в разных файлах, доступность данных, комбинируемых из этих файлов, ограничена.

Для решения вышеуказанных проблем были разработаны системы с базами данных, представляющими собой совокупность специально организованных данных, рассчитанных на применение в большом количестве прикладных программ конкретной предметной области, работа с которыми обеспечивается специальным па-

кетом прикладных программ — системой управления базой данных с целью создания массивов данных, их обновления и получения справок.

Основное отличие БД от файла данных состоит в том, что файл данных может иметь несколько назначений, но соответствует одному представлению о хранимых данных; БД также имеет несколько назначений, но соответствует различным представлениям о хранимых данных.

Программный модуль, входящий в состав САПР, при своей работе обращается за необходимой информацией не к какому-то массиву информации, как это имело место в автономных программах, а к СУБД. Последняя организует поиск необходимой информации в сложной информационной структуре — БД, упорядочивание и представление этой информации в необходимом объеме соответствующему модулю.

Различие в методах использования файлов данных и БД проиллюстрировано на рис. 4.1 и 4.2.

Основные требования к БД следующие:

- целостность данных — их непротиворечивость и достоверность;
- универсальность, т. е. наличие в БД всех необходимых данных и возможность доступа к ним в процессе решения проектной задачи;
- открытость БД для внесения в нее новой информации;
- наличие языков высокого уровня взаимодействия пользователей с БД;
- секретность, т. е. невозможность несанкционированного доступа к информации и ее изменений;
- оптимизация организации БД — минимизация избыточности данных.



Рис. 4.1. Взаимно-однозначное соответствие между прикладными программами и файлами данных



Рис. 4.2. Использование универсальных методов доступа

Одним из принципов построения САПР является информационная согласованность частей ее программного обеспечения, т. е. пригодность результатов выполнения одной проектной процедуры для использования другой проектной процедурой без их трудоемкого ручного преобразования пользователем. Отсюда вытекают следующие условия информационной согласованности:

- использование программами одной и той же подсистемы САПР единой БД;
- использование единого внутреннего языка для представления данных.

Комплексная автоматизация процесса проектирования объекта предполагает информационную согласованность не только отдельных программ подсистем САПР, но и самих подсистем между собой. Способом достижения этой согласованности является единство информационного обеспечения.

Основные способы информационного согласования подсистем САПР достигаются либо созданием единой БД, либо сопряжением нескольких БД с помощью специальных программ, которые перекодируют информацию, приводя ее к требуемому виду.

Сложность разработки БД определяется тем, что формирование структуры БД возможно только после разработки алгоритмов, реализуемых при проектировании. При этом необходимо, чтобы степень разработки алгоритмов была доведена до машинной реализации, так как структура БД должна учитывать специфику автоматизированного преобразования информации при решении проектных задач с целью эффективного использования вычислительной техники. Однако для разработки программ необходимы сведения о структуре БД. Следовательно, информационное обеспечение САПР и специальное программное обеспечение должны создаваться параллельно.

Информация, имеющая место при проектировании, может быть разделена на статическую и динамическую (рис. 4.3). Статическая информация характеризуется сравнительно редкими изменениями и используется только в режиме чтения. К этой информации следует отнести данные технического задания на проектирование и справочные данные, которые характеризуются большим объемом. Формирование, загрузка и корректировка справочных данных осуществляются исключительно администратором базы данных, т. е. программистом системного профиля, формирующим БД. Объем данных технического задания на проектируемый объект значительно меньше объема справочных данных. Круг лиц, имеющих право вносить изменения в данные технического задания, более ограничен, чем круг лиц, имеющих право производить корректировку справочных данных.

Динамическая информация состоит из данных, накапливаемых для выполнения определенных процессов проектирования (промежуточные данные), и данных, представляющих собой ре-

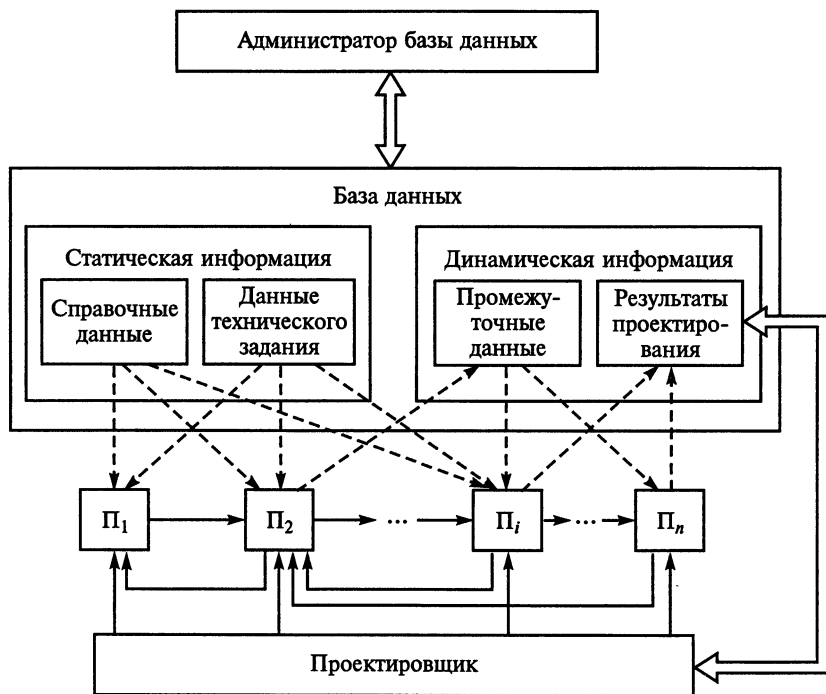


Рис. 4.3. Информационная схема проектирования:

$\Pi_i$  — элементарный процесс проектирования;  $\longrightarrow$  — управляющие связи;  $\dashrightarrow$  — информационные связи;  $\longleftrightarrow$  — наполнение и корректировка базы данных, анализ результатов

зультат проектирования при выполнении этих процессов. Промежуточные данные меняются при функционировании САПР. Вносить же коррективы в проектные результаты имеет право только главный конструктор проекта при соответствующем обосновании.

Широко используется в САПР фактографическая информация, составляющая основу БД. Эта информация представляет собой числовые и буквенные справочные данные о материалах, ценах, комплектующих изделиях, спроектированных в САПР объектах, их зарубежных и отечественных аналогах и т. п. Сюда же относятся данные, необходимые для выполнения расчетов: коэффициенты, таблицы, аппроксимированные графические зависимости и т. д.

При формировании БД в первую очередь надлежит исследовать информацию, необходимую для решения проектно-конструкторских задач. К этому исследованию следует подходить с двух позиций: с точки зрения полезности информации и с позиции эффективности обработки информации и пропускной способности вычислительной техники и человека.

Как уже отмечалось выше, части программного обеспечения и методы, осуществляющие управление базой данных, составляют СУБД, которая допускает множество различных представлений о хранимых данных, а также позволяет прикладным программам работать с БД без знания конкретного способа размещения данных в памяти ЭВМ. СУБД выступает как совокупность программных средств, предназначенных для создания, ведения и совместного использования БД многими пользователями и должна обеспечивать:

эффективное выполнение различных функций предметной области;

простоту физической реализации БД;

возможность централизованного и децентрализованного управления БД;

минимизацию избыточности хранимых данных;

предоставление пользователю по запросам непротиворечивой информации;

простоту разработки, ведения и совершенствования прикладных программ.

СУБД реализует два интерфейса:

- между логическими структурами данных в программах и БД;
- между логической и физической структурами БД.

Структурная схема СУБД приведена на рис. 4.4.

Манипулятор является ядром СУБД. В его функции входит управление компонентами системы, организация их взаимодействия и осуществление связи с операционной системой ЭВМ и администратором банка данных, выполнение основных операций над БД, контроль и защита целостности и секретности данных, ре-



Рис. 4.4. Структурная схема СУБД

дактирование вывода, кодирование (декодирование) и сжатие (расширение) данных, сбор статистики и ведение протокола.

Администратор банка данных осуществляет внешнюю координацию всей работы банка данных и выполняет действия, пока не поддающиеся формализации. Прежде всего он отвечает за разработку концептуальной модели предметной области, описывающей все представляющие интерес объекты и взаимосвязи. Концептуальная модель должна быть трансформирована в модель данных, поддерживаемую конкретной СУБД. Следовательно, администратору необходимо спроектировать логическую модель. Наконец, исходя из логической модели, он должен спроектировать физическую (внутреннюю) модель, которая будет обеспечивать требуемые операционные характеристики. Таким образом, в его функции входит создание БД, согласование требований пользователей, управление загрузкой баз, распределение паролей; общее управление работой банка данных (наблюдение за ходом работы, подключение новых пользователей, управление восстановлением при сбоях, принятие решений в случае нарушений секретности, анализ статистики, оценка и обеспечение эффективности работы БД); реорганизация БД (изменение схем, реорганизация баз); генерация и развитие СУБД.

Администратор и манипулятор выполняют свои функции, взаимодействуя с сервисными программами. Ряд сервисных программ допускает двойной запуск: автоматически со стороны манипулятора БД и вручную со стороны администратора, другие запускаются только администратором БД (например, программы моделирования БД). Администратор БД может пополнять набор новыми сервисными программами (контроля данных, измерения производительности и т. д.).

Сервисные программы осуществляют основные операции над БД, в частности сортировку, выборку данных, слияние, дополнение и изменение БД, редактирование ответов.

При функционировании банка данных различают следующие основные виды запросов к банку со стороны пользователей:



на выдачу справок в виде показателей и документов для пользователей системы;

формирование и выдачу рабочих массивов для прикладных программ САПР;

выдачу рабочих массивов со стороны других банков данных;

пополнение или изменение БД;

ввод входных массивов.

Поступивший в банк данных запрос проходит этап предварительной обработки, на котором осуществляется синтаксический и логический контроль, включающий проверку пароля абонентов, недопустимых сочетаний признаков и т. д. При обнаружении ошибок запрос к дальнейшей обработке не принимается, а на печать выдается информация об ошибках.

Следующий этап — интерпретация запроса — состоит в распознавании вида запроса: на выдачу показателя и документов, на формирование рабочих массивов, на изменение или пополнение БД. На этом этапе запрос с языка запросов переводится на язык манипулирования данными.

В соответствии с поисковым предписанием происходят обращение к рабочей области в памяти и выборка искомых данных или корректировка данных в базе. Найденные в БД данные тщательно контролируются и анализируются, затем редактируются, а на печать выдается информация об ошибках.

Разработка банков данных и его основного инвариантного компонента, т. е. СУБД, представляет собой сложную задачу, требующую значительных затрат времени и предполагающую высокую квалификацию разработчиков в области системного программирования. В связи с этим при построении информационного обеспечения САПР конкретных технических объектов на основе банков данных необходимо сделать анализ существующих банков данных и принять для САПР один из уже разработанных. Однако и в этом случае необходимость использования такого сложного компонента, каким является банк данных, должна быть всесторонне обоснована. Следует иметь в виду, что информационное обеспечение САПР может быть организовано и на основе более простой, но специально разработанной информационной структуры, учитывающей специфику проектируемых технических объектов. Подобные ориентированные информационные структуры при относительной простоте и невысоких требованиях к конфигурации технических средств могут реализовать набор необходимых функций по обработке данных, для чего должно быть разработано необходимое программное обеспечение.

Такая альтернатива к использованию банков данных объясняется тем, что универсальные информационные системы в виде банков данных рассчитаны на решение задач с большей номенклатурой различных параметров и характеристик объектов, поэто-

му применение их при создании САПР с относительно небольшой номенклатурой данных иногда оказывается просто нецелесообразным, а порой и невозможным.

Одной из важнейших характеристик СУБД является модель данных, которая поддерживается СУБД. Как отмечалось, существуют три модели данных: иерархическая, сетевая и реляционная. Выбор той или иной модели данных для построения информационного обеспечения САПР зависит от требований, которые предъявляются к информационной базе создаваемой САПР.

Большое значение при выборе СУБД имеют средства взаимодействия пользователей с БД. Непрограммисты, а ими являются большинство пользователей САПР, взаимодействуют с БД с помощью языков манипулирования данными с использованием средств телеобработки. Тогда СУБД САПР должна включать в себя язык манипулирования данными, доступный непрограммисту, и средства доступа к данным через терминал.

Кроме того, при выборе для САПР той или иной СУБД следует учитывать следующие требования:

- возможность обеспечения мультипрограммного режима;
- независимость данных от конкретных пакетов программ;
- наличие средств, позволяющих сократить дублирование данных в информационной базе; возможность восстановления БД;
- совместимость СУБД с операционной системой ЭВМ и т. д.

Рациональная организация и функционирование банка данных возможны лишь при максимальном учете специфики информации, заносимой в БД. Если специфические особенности такой информации не учитываются, это приводит к значительному перерасходу машинной памяти и увеличивает затраты времени на обработку данных.

База данных организуется на основе принципов системного подхода. При этом предполагаются:

- неизбыточность данных;
- независимость данных от программ решения задач;
- выбор структур данных, ориентированных на все задачи, решаемые системой;
- возможность дополнения, развития и обновления данных;
- типизация алгоритмов обработки данных.

Организация, структура и состав БД зависят от информационных моделей проектируемых объектов, от методов получения проектных решений и от используемых в САПР технических и программных средств. С другой стороны, все перечисленные факторы зависят от структуры БД. Основной особенностью БД по сравнению с массивами записей является наличие связей между структурными единицами данных.

Базы данных иногда определяют как избыточную совокупность элементов данных. Однако в действительности для умень-

шения времени доступа к данным во многих БД избыточность в незначительной степени присутствует. Некоторые записи повторяются для того, чтобы обеспечить возможность восстановления данных при их случайной потере. В этом случае говорят о минимальной избыточности БД.

Таким образом, БД можно определить как совокупность взаимосвязанных, совместно хранящихся данных при наличии такой минимальной избыточности, при которой допускается их использование оптимальным образом для одного или нескольких приложений. Для этого данные запоминаются так, чтобы они были независимы от прикладных программ, использующих эти данные. Для добавления новых или модификации существующих данных и для поиска данных в БД применяется стандартный набор управляющих алгоритмов.

База данных представляет собой постоянно развивающийся объект (к ней добавляются новые записи, а в существующие включаются новые элементы данных). С целью повышения эффективности функционирования БД изменяется и ее структура. Используемые на практике способы построения БД реализуются в виде иерархических (древовидных), сетевых или реляционных моделей (моделей отношений) данных.

### 4.3. Модели представления данных

Современные СУБД основываются на использовании моделей данных, позволяющих описывать объекты предметных областей и взаимосвязи между ними.

Существуют три основные модели данных, на которых основываются СУБД: реляционная, сетевая и иерархическая; возможны также комбинации указанных моделей. Наиболее простая модель — это представление данных в виде определенной иерархии. Универсальной моделью является представление данных в виде отношений, или реляций. Сетевая модель занимает промежуточное положение между иерархической и реляционной моделями данных. В ней, с одной стороны, применяется достаточно универсальное представление данных, имеющее широкое применение, а с другой — учитывается возможность прямого доступа к данным.

Рассмотрим особенности построения различных моделей БД.

*Иерархическая модель* основана на понятии деревьев, состоящих из узлов и ветвей, задающих отношения (связи) между данными. Узлы дерева ставятся в соответствие совокупности данных, характеризующих некоторый объект. Узлы и ветви дерева образуют иерархическую древовидную структуру, состоящую из подуровней. Она всегда начинается с корневых узлов, которые в этой

модели находятся на первом уровне. На последующих уровнях могут добавляться зависимые элементы, которые имеют только одну точку входа (рис. 4.5). Узел и его зависимые элементы называются группой. Наименование корневого узла служит названием группы. Группа называется простой, если ни один из зависимых элементов не имеет своих зависимых элементов, в противном случае группа называется составной. Группа, являющаяся элементом составной группы и имеющая более одной реализации, называется повторяющейся.

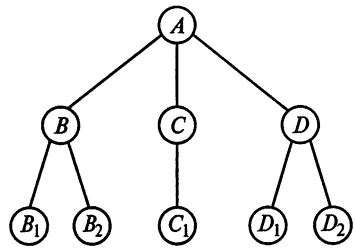


Рис. 4.5. Иерархическая модель базы данных

Иерархическая структура удовлетворяет следующим условиям:

- иерархия начинается с корневого узла;
- на уровнях с большим номером (нижележащих) находятся зависимые узлы;
- узел предшествующего уровня является начальным для новых зависимых узлов;
- каждый узел, находящийся на нижележащем уровне, соединен с одним и только одним узлом предшествующего уровня, за исключением корневого узла;
- корневой узел может быть связан с одним или несколькими зависимыми узлами;
- доступ к каждому узлу происходит только через корневой узел по единственному пути;
- существует произвольное количество узлов каждого уровня.

Основное свойство иерархической модели данных — это возможность доступа к любым данным только через корневой узел по единственному пути, что увеличивает время доступа. Из этого свойства вытекают и другие недостатки иерархической модели, которые заключаются в следующем:

- невозможно получить данные об элементе *B* без осуществления доступа к данным об элементе *A* (см. рис. 4.5);
- если два узла содержат одинаковые элементы, эти элементы в БД повторяются (например,  $C_1 = D_1$ ), т.е. имеет место избыточность данных, что особенно нежелательно на физическом уровне (избыточность на логическом уровне, напротив, желательна, поскольку упрощает работу с данными);
- иерархическая упорядоченность усложняет операции удаления и включения данных;
- если теряются данные об узле *B*, удаляются данные об узлах  $B_1$  и  $B_2$ , т.е. удаление объекта влечет за собой удаление подчиненных объектов (поэтому выполнение операции удаления требует особой осторожности);

• легко получить ответ на вопрос «Какие элементы входят в узел В?», но трудно получить ответ на вопрос «Какие узлы содержат одинаковые элементы?», т.е. возникают трудности с составлением выборок информации по заданному признаку.

Достоинства иерархической модели следующие:

- наличие хорошо зарекомендовавших себя систем управления БД, основанных на ее применении;
- простота понимания и использования (пользователи систем обработки данных хорошо знакомы с иерархической структурой);
- обеспечение определенного уровня независимости данных;
- простота оценки операционных характеристик благодаря заранее заданным взаимосвязям.

*Сетевая модель* может быть описана в виде графа (рис. 4.6). Она представляет собой совокупность взаимосвязанных элементов и не имеет элементов со специальными функциями, подобно корневым узлам в иерархической модели. Помимо самих данных, в каждом элементе сетевой модели содержится строка справочника, в которой указываются все связи данного элемента. В отличие от иерархической модели все данные, содержащиеся в сетевой модели, равноправны, т.е. отсутствует система подчинения одних данных другим и в узлах сети могут быть расположены данные, тематически не связанные друг с другом.

Другим важным свойством сетевой модели является то, что возможно установление отношений между любыми узлами сети, однако следует иметь в виду, что объем справочника, в котором хранится информация об отношениях (связях) данного узла, с увеличением количества связей возрастает. Таким образом, излишнее количество связей может отрицательно сказаться на размерах БД.

С точки зрения количества полезной информации сетевая модель может быть наиболее компактной, поскольку возможность установления любых связей между узлами дает возможность избежать в БД узлов с одинаковой информацией (отсутствие избыточности данных).

Вход в сетевую модель в принципе возможен через любой из узловых элементов.

Главными достоинствами сетевой модели данных являются:

- наличие успешных реализаций систем управления БД, обеспечивающих эту сетевую модель (как в иерархической модели);

- простота реализации часто встречающихся в реальном мире взаимосвязей «многие ко многим».

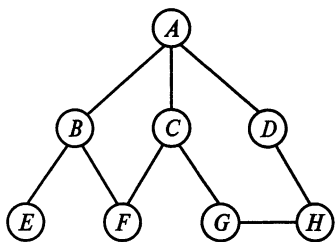


Рис. 4.6. Сетевая модель базы данных

Основной недостаток сетевой модели состоит в ее сложности. Прикладной программист должен детально знать логическую структуру БД, поскольку ему необходимо осуществлять навигацию среди различных по сути записей данных. Иначе говоря, программист должен представлять «свое» текущее положение в БД при «продвижении» по ней.

Недостатком является также возможная потеря независимости данных при реорганизации БД. Кроме того, в сетевой модели данных представление, используемое прикладной программой, сложнее, чем в иерархической модели, поэтому и составление прикладных программ может оказаться сложнее.

Сетевая структура может быть получена из иерархической структуры путем добавления понятия ссылки, которое может быть определено как логический указатель. Так можно связать две древовидные структуры без установления между ними иерархических отношений.

Сетевая модель не имеет перечисленных выше недостатков иерархической модели. Связи в ней могут быть устранены без удаления всей группы, на которую указывает ссылка. Однако эта модель не обеспечивает независимости данных.

Полной независимостью данных и операций над ними от физического представления и возможностью формального подхода к данным и операциям обладает *реляционная модель*.

Реляционный подход к построению моделей данных основывается на использовании для их описания произвольных отношений (англ. relation, отсюда название). Любое подобное отношение может быть представлено двумерной таблицей.

В табл. 4.1 представлено отношение с именем  $A$  над элементарными данными с именами  $a, b, c, \dots, x$ . Каждая строка таблицы

Таблица 4.1

Отношение  $A$

$a$	$b$	$c$	...	$x$
$a_1$	$b_1$	$c_1$	...	$x_1$
$a_2$	$b_2$	$c_2$	...	$x_2$
$a_3$	$b_3$	$c_3$	...	$x_3$
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
$a_m$	$b_m$	$c_m$	...	$x_m$

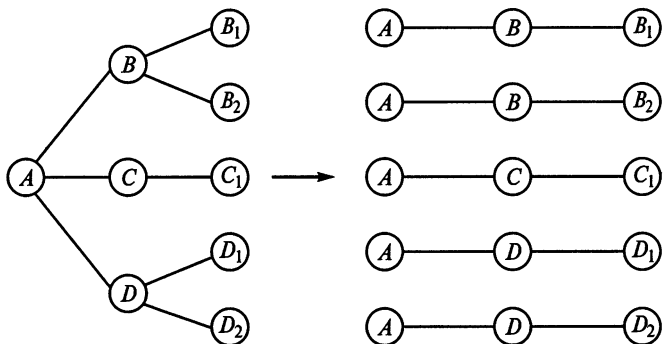


Рис. 4.7. Нормализация иерархической модели базы данных

представляет набор значений этих данных, находящихся между собой в данном отношении. Строки таблицы обычно называют кортежами, т.е. упорядоченными множествами, а столбцы — доменами (полями). Каждый столбец в таблице является атрибутом. Значения в столбце определяют из множества значений, которые принимает атрибут. Столбцы таблицы — это элементы данных, а строки — записи.

Если количество столбцов равно  $n$ , то говорят, что отношение  $A$  является  $n$ -местным отношением.

Считая кортежи отдельными записями, легко представить такое отношение (таблицу) в виде простейшего последовательного файла.

Главная идея реляционного подхода состоит в том, чтобы представлять произвольные структуры данных в виде совокупностей описанных таблиц. Процесс такого представления называется нормализацией, а само представление — реляционной структурой. В качестве примера на рис. 4.7 приведена нормализация иерархической модели данных. Путем «размножения» узлов иерархической модели она представляется в виде таблицы, которой достаточно лишь присвоить имя корневого узла соответствующей иерархической модели и пронумеровать домены. В случае сетевых структур тем же приемом «размножения» узлов они приводятся сначала к иерархической, а затем и к реляционной форме (рис. 4.8).

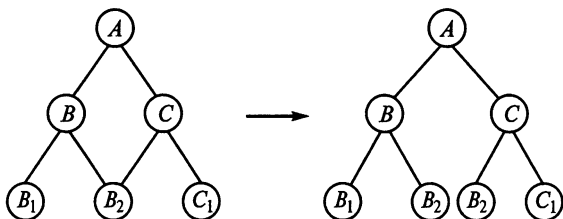


Рис. 4.8. Приведение сетевой модели базы данных к иерархической

Работая с реляционной моделью, пользователь не интересуется точной физической структурой данных, что является важным достоинством реляционного подхода. Кроме того, здесь обеспечивается простота понимания структуры БД, поскольку многие пользователи имеют практику работы с двумерными таблицами. Наконец, реляционную БД проще развивать и дополнять.

Важным свойством реляционной модели является простота составления выборок по заданным признакам, что становится возможным исходя из самой структуры модели, где каждый столбец таблицы соответствует определенному атрибуту объекта и, таким образом, является его идентификатором. Система в процессе поиска необходимой информации анализирует атрибуты и, в случае удовлетворения требованиям информационного запроса, может поставить в соответствие найденным значениям атрибута любые связанные с ним данные из соответствующего ему кортежа.



**ТЕХНИЧЕСКИЕ СРЕДСТВА САПР****5.1. Общие сведения**

Техническое обеспечение САПР совместно с программным обеспечением является инструментальной базой САПР, в среде которой реализуются другие виды обеспечения САПР.

Компоненты технического обеспечения САПР включают в себя средства вычислительной и организационной техники, средства передачи данных, измерительные и другие устройства или их сочетания. Требования к составу и структура технических средств формируются исходя из общих требований к структуре САПР, эффективного решения заданного класса задач проектирования, активного включения пользователя в процесс проектирования, возможности работы с графическим материалом, включая процессы как ввода и обработки, так и вывода информации.

При выборе технических средств САПР, обеспечивающих интерактивное взаимодействие конструктора с процессом решения, следует руководствоваться следующим:

- используемая ЭВМ должна позволять организовать работу определенного числа пользователей (т.е. удовлетворять требованиям по производительности) в режиме коллективного пользования;
- используемое терминальное оборудование должно отвечать эргономическим, эстетическим требованиям и требованиям эффективности работы.

Конкретное оборудование определяется классом решаемых в САПР задач, т.е. структурный состав комплекса технических средств определяется только в условиях конкретной разработки на основе анализа состава решаемых задач.

Эволюция развития комплекса технических средств САПР характеризуется созданием территориально рассредоточенных многомашинных систем сбора, хранения и обработки информации, реализованных в виде вычислительных сетей. Последние, рассредоточенные на небольших территориях предприятий и объединяющие в единую информационную систему АРМ пользователей, ЭВМ и микроЭВМ, графопостроители, терминальные станции и другую специализированную аппаратуру, называют локальными вычислительными сетями (ЛВС). Локальные вычислительные

сети имеют открытую архитектуру, обеспечивающую возможность подключения к сети любых других ЛВС, в том числе крупных сетей ЭВМ. Основное достоинство ЛВС — низкая стоимость системы передачи данных.

Локальные вычислительные сети САПР должны обеспечивать: использование режимов пакетной и диалоговой обработки, разделения времени и виртуальной памяти;

экономичную обработку информации по принципу «наиболее важные процессы САПР выполняются техническими средствами с развитым программным обеспечением и высокой производительностью, наименее ответственные — на дешевых мини- и микроЭВМ»;

высокую надежность и достоверность функционирования, высокую производительность;

применение разнообразного проблемно-ориентированного программного обеспечения, централизованных и локальных БД с необходимым объемом памяти;

работу с автоматизированными рабочими местами различного назначения и другим специализированным оборудованием;

централизованную и децентрализованную обработку информации.

Использование ЛВС позволяет создать САПР нового поколения, объединяющие контрольно-измерительные комплексы и места сбора информации с автоматизированными рабочими местами конструкторов, механиков и т. д.

Основное назначение ЛВС — распределение ресурсов ЭВМ (программ, совокупности периферийных устройств, терминалов, памяти) для эффективного решения задач автоматизированного проектирования.

Локальные вычислительные сети должны иметь надежную, быструю и дешевую систему передачи данных, а стоимость передачи единицы информации должна быть значительно ниже стоимости обработки единицы информации.

## **5.2. Электронные вычислительные машины в САПР**

Основой комплексов технических средств САПР являются разнообразные ЭВМ от простейших микроЭВМ до сложнейших суперЭВМ.

При определении возможности использования той или иной ЭВМ в составе технических средств САПР их оценивают по совокупности различных показателей, главные из которых технические характеристики, стоимость приобретения и эксплуатации.

История создания и развития САПР неразрывно связана с историей возникновения ЭВМ. Эра электронных вычислительных машин началась в 1945 г., когда в США была построена первая

действующая ЭВМ — ENIAC. Она содержала 18 000 электронных ламп, потребляла 150 кВт электроэнергии и обладала следующими характеристиками: операция сложения занимала 0,2 мс, а операция умножения — 2,8 мс. Объем памяти для чисел составлял 20 ячеек, память для программ отсутствовала, а программирование осуществлялось с помощью штекеров и переключателей.

Первая ЭВМ с хранимой в памяти программой появилась в Англии в 1949 г. По быстродействию она существенно не отличалась от ENIAC, но имела память из 512 ячеек. В последующие годы появились и другие ЭВМ, отличающиеся от первых более высоким быстродействием, увеличенным объемом памяти и более развитыми средствами ввода-вывода программ и данных.

В СССР к разработке ЭВМ приступили в конце 1940-х гг. В 1950 г. в Институте электротехники АН УССР в Киеве была испытана первая отечественная ЭВМ на электронных лампах — малая электронная счетная машина (МЭСМ). В 1952 г. была создана большая электронная счетная машина (БЭСМ), которая после модернизации в 1954 г. имела высокое для того времени быстродействие — 10 000 операций/с. Серийный выпуск ЭВМ начался в 1953 г. (ЭВМ «Стрела», М-2, М-3 и др.).

Историю ЭВМ принято делить на поколения. К первому поколению относятся ламповые машины. В 1948 г. был изобретен транзистор, а через несколько лет, примерно с 1955 г., появились транзисторные ЭВМ, относящиеся ко второму поколению. Они были компактнее, экономичнее и надежнее ламповых ЭВМ. В СССР первая серийная транзисторная ЭВМ «Раздан-2» была создана в 1961 г., а в последующие годы начался выпуск транзисторных ЭВМ «Минск», «Урал», БЭСМ-4 и др.

К третьему поколению, возникшему примерно с 1964 г., принадлежат вычислительные машины на интегральных схемах. Применение интегральных схем позволило резко уменьшить размеры создаваемых ЭВМ, повысить их надежность и сократить потребление энергии. В СССР наиболее популярными вычислительными машинами третьего поколения были ЭВМ Единой системы (ЕС), представленные целым рядом типов или моделей, совместимых между собой (например, ЕС-1020, ЕС-1040, ЕС-1050 и др.). К этому поколению относятся также БЭСМ-6 и малые вычислительные машины, или мини-ЭВМ: СМ-2, СМ-4, «Электроника-100» и др.

В 1971 г. в разных странах были созданы принципиально новые интегральные схемы, в которых с помощью оригинальных технологических методов резко, в десятки и сотни, а позднее и в тысячи раз повысилось количество транзисторов, входящих в схему, без увеличения ее габаритных размеров. Такие интегральные схемы называли большими интегральными схемами (БИС). Появление БИС стало началом рождения миниатюрных вычислительных машин четвертого поколения — микроЭВМ, или микрокомпьюте-

ров. «Сердце» микрокомпьютера — микропроцессор. Первый микропроцессор был создан в США в 1971 г. В последующие годы были разработаны микрокомпьютеры, которые, уступая большим и мини-ЭВМ по объему памяти и некоторым другим показателям, имели несравненное преимущество в стоимости, компактности, экономичности, надежности, простоте производства и эксплуатации; эти достоинства предопределили широкое распространение микрокомпьютеров и их внедрение во все сферы деятельности человека.

Появились и стали привычными для людей персональные микрокомпьютеры, предназначенные для автоматизации деятельности, связанной с умственным трудом и рутинными вычислениями и подсчетами (в середине 1970-х гг. — микрокомпьютер LSI-11 фирмы DEC, а в 1981 г. — первый персональный компьютер фирмы IBM).

В СССР микроЭВМ появились в середине 1970-х гг. Один из первых отечественных микрокомпьютеров — 16-разрядная микромашинка «Электроника-60» широко применялась в системах автоматического и автоматизированного управления технологическими процессами, станками, роботами, в контрольно-измерительном оборудовании, для решения различных вычислительных задач. В конце 1970-х гг. был создан микрокомпьютер СМ-1800, положивший начало 8-разрядным микромашинкам в СССР. В 1981 г. появился одноплатный микропроцессорный вычислитель «Электроника НЦ-8001Д», на основе которого в 1982 г. была разработана первая модель микрокомпьютеров семейства диалоговых вычислительных комплексов (ДВК), за которой последовали новые, более совершенные модели. Следует упомянуть также микрокомпьютеры «Электроника-85», ЕС-1840, ЕС-1841 и «Искра-1030». На смену 8- и 16-разрядным пришли 32-разрядные микрокомпьютеры, которые по своему быстродействию, объему оперативной памяти и другим техническим характеристикам приблизились к большим ЭВМ.

Все аппаратные средства ЭВМ делятся на две группы устройств: центральные и периферийные.

К *центральному устройству*, осуществляющим непосредственно обработку данных, относятся центральный процессор и оперативное запоминающее устройство (ОЗУ).

*Центральный процессор* предназначен для преобразования информации в соответствии с выполняемой программой, управления вычислительным процессом и устройствами, работающими совместно с процессором.

*Оперативное запоминающее устройство* выполняет функции хранения, приема и выдачи данных и программ.

К *периферийным устройствам* относятся устройства, выполняющие функции ввода, вывода, подготовки данных и хранения

больших объемов информации. Общим для всех периферийных устройств является то, что они преобразуют данные из одной формы представления в другую, не изменяя их содержания.

Одновременная работа нескольких периферийных устройств обеспечивается тем, что их скорость работы обычно существенно ниже быстродействия канала ввода-вывода, который может осуществлять обмен с несколькими периферийными устройствами, распределяя между ними время использования общих средств.

Функции управления процессом обмена данными распределяются между каналом и устройством управления периферийными устройствами (контроллером периферийных устройств): в канале реализуются функции, общие для всех периферийных устройств, а в контроллере — специфические для данного периферийного устройства.

Периферийные устройства, используемые в комплексе технических средств САПР, можно разделить на следующие группы:

- внешние запоминающие устройства, предназначенные для хранения больших объемов информации, организации БД, содержащих различную справочную информацию, типовые проектные решения и т. п.;

- устройства ввода-вывода и документирования символьной информации, осуществляющие ввод-вывод информации с различных носителей и документирование полученных проектных решений;

- устройства оперативного взаимодействия с ЭВМ, позволяющие инженеру обращаться к САПР в произвольные моменты времени удобными для него способами (обычно это различные дисплеи и устройства речевого ввода-вывода);

- устройства машинной графики, предназначенные для формирования, ввода-вывода и документирования информации, представленной в графической форме;

- технические средства теледоступа и организации сетей ЭВМ, осуществляющие связь ЭВМ с удаленными пользователями САПР, между всеми ЭВМ, входящими в комплекс технических средств САПР при организации сети проектирования, и между комплексом технических средств САПР и вычислительной сетью;

- устройства связи с технологическим оборудованием, позволяющие, минуя промежуточные носители, непосредственно передавать данные о спроектированных объектах системам автоматизации производства, станкам с числовым программным управлением и другому оборудованию;

- устройства подготовки данных, предназначенные для подготовки данных и программ на различных носителях автономно без участия ЭВМ.

К основным техническим параметрам ЭВМ относят разрядность машинного слова, производительность, емкость ОЗУ, про-

пускную способность подсистемы ввода-вывода информации, надежность функционирования и др.

*Производительность* — один из важнейших показателей ЭВМ, измеряемый количеством операций, выполняемых в единицу времени (в секунду). Для разных типов ЭВМ этот показатель колеблется от нескольких сотен до сотен миллионов операций в секунду и более. Производительность зависит не только от свойств самой ЭВМ, но и от особенностей обрабатываемой информации (разрядности обрабатываемых слов, формы представления чисел — с плавающей или фиксированной точкой, частоты повторения различных операций в общем потоке выполняемых программ и др.) и определяется с помощью тестовых наборов задач (предварительно выявляется процентное содержание команд разного типа). Здесь необходимо отметить, что значения производительности ЭВМ, приводимые в различной справочной литературе (в том числе и в данном издании), могут использоваться только для ориентировочной оценки реальной производительности при решении конкретных задач автоматизированного проектирования. Для большинства проектных процедур в САПР наиболее приемлемы оценки производительности, полученные для класса научно-технических задач.

*Емкость ОЗУ* определяет возможности ЭВМ по выполнению сложных программ с обработкой больших объемов информации. Емкость ОЗУ может выражаться в битах, байтах, словах, килобайтах, мегабайтах и т. п. Наиболее распространена оценка емкости ОЗУ в байтах, килобайтах (1 Кбайт = 1024 байт), мегабайтах (1 Мбайт = 1024 Кбайт), гигабайтах (1 Гбайт = 1024 Мбайт). Емкость ОЗУ для ЭВМ, используемых в САПР, достигает гигабайт.

*Пропускная способность подсистемы ввода-вывода* ЭВМ позволяет определить возможности ЭВМ при обмене информацией с различными периферийными устройствами или другими ЭВМ. Она измеряется максимальным количеством единиц информации, переданных через подсистему ввода-вывода за единицу времени. Чаще всего пропускная способность измеряется количеством переданных в секунду байт, килобайт, мегабайт и изменяется от сотен байт в секунду до десятков и сотен мегабайт в секунду.

По производительности и характеру использования ЭВМ можно условно подразделить:

- на микроЭВМ, в том числе персональные ЭВМ (ПЭВМ);
- мини-ЭВМ;
- суперЭВМ.

*МикроЭВМ* — это компьютеры, в которых центральный процессор выполнен в виде микропроцессора.

Лучшие модели микрокомпьютеров могут иметь несколько микропроцессоров. Производительность компьютера определяет-

ся не только характеристиками применяемого микропроцессора, но и емкостью оперативной памяти, типами периферийных устройств, качеством конструктивных решений и др.

Микрокомпьютеры представляют собой инструменты для решения разнообразных сложных задач. Их микропроцессоры с каждым годом увеличивают мощность, а периферийные устройства — эффективность. Быстродействие таких ЭВМ порядка 1—10 миллионов операций в секунду, объем оперативного запоминающего устройства — сотни и тысячи мегабайт.

Персональные компьютеры (ПК) — это микрокомпьютеры универсального назначения, рассчитанные на одного пользователя и управляемые одним человеком. В класс персональных компьютеров входят различные машины — от дешевых домашних и игровых с небольшой оперативной памятью, с памятью программы на кассетной ленте и обычным телевизором в качестве дисплея (1980-е гг.) до сверхсложных машин с мощным процессором, накопителями на магнитных дисках емкостью в десятки гигабайт, с цветными графическими устройствами высокого разрешения, средствами мультимедиа и другими дополнительными устройствами.

Персональный компьютер должен удовлетворять следующим требованиям:

- наличие внешних запоминающих устройств на магнитных дисках;
- объем оперативной памяти не менее 32 Мбайт;
- наличие операционной системы;
- способность работать с программами на языках высокого уровня;
- ориентация на пользователя-непрофессионала (в простых моделях).

МикроЭВМ — не единственное современное направление развития вычислительной техники.

Продолжается совершенствование и больших ЭВМ. На их основе, а также с использованием мини- и микрокомпьютеров создаются вычислительные сети, охватывающие предприятия, города, целые страны и группы стран. Достигнутое к настоящему времени на некоторых типах машин быстродействие (миллиарды операций в секунду) и объем оперативной памяти (сотни гигабайт) не являются предельными.

*Миникомпьютерами и суперминикомпьютерами* называют машины объемом порядка  $0,5 \text{ м}^3$ , конструктивно выполненные в одной стойке. По своим возможностям они занимают промежуточное положение между микрокомпьютерами и суперкомпьютерами.

Для выполнения сверхсложных, ресурсоемких расчетов приходится использовать очень мощные вычислительные средства. Самые производительные компьютеры называют суперкомпьютерами, которые представляют собой комплекс программно-аппарат-

ных средств, обслуживающего оборудования и инфраструктуры. Чтобы оптимально загрузить вычислительные мощности суперкомпьютера, требуется иметь целую сеть вспомогательных компьютеров, сверхскоростные каналы передачи информации и специальные управляющие алгоритмы.

*Суперкомпьютеры* — это очень мощные компьютеры производительностью свыше 100 Мфлоп (1 Мфлоп — миллион операций с плавающей точкой в секунду), представляют собой многопроцессорные и (или) многомашинные комплексы, работающие на общую память и общее поле внешних устройств.

Архитектура суперкомпьютеров основана на идеях параллелизма и конвейеризации вычислений. Все суперкомпьютеры имеют несколько независимых процессорных блоков (от сотни до нескольких тысяч). Память может быть доступна целиком каждому процессору или может быть поделена на зоны ответственности каждого процессора. В то же время каждый процессор может иметь свою собственную быструю память для ускорения локальных расчетов. Несмотря на то, что максимальная производительность суперкомпьютера может превосходить в тысячи раз производительность обычных компьютеров, скорость вычислений при запуске на суперкомпьютере обычной программы окажется всего в несколько раз выше, поэтому при написании программ для суперкомпьютеров необходимо учитывать большое количество одновременно работающих процессоров и оптимально распределять задачи между ними.

В этих машинах параллельно, т. е. одновременно, выполняется множество похожих операций, что называется мультипроцессорной обработкой. Сверхвысокое быстродействие на суперкомпьютере обеспечивается только для задач, поддающихся распараллеливанию (конвейерной обработке).

Что такое конвейерная обработка? Приведем сравнение. На каждом рабочем месте конвейера выполняется одна операция производственного процесса, а на всех рабочих местах в одно и то же время обрабатываются различные изделия на всевозможных стадиях. По такому же принципу устроено арифметико-логическое устройство суперкомпьютера.

Отличительной особенностью суперкомпьютеров являются векторные процессоры, оснащенные аппаратурой для параллельного выполнения операций с многомерными цифровыми объектами — векторами и матрицами. В них встроены векторные регистры и параллельный конвейерный механизм обработки. Если на обычном процессоре программист выполняет операции над каждым компонентом вектора по очереди, то на векторном выдает сразу векторные команды. Векторная аппаратура очень дорога, в частности потому, что требуется много сверхбыстродействующей памяти под векторные регистры.



## Технические характеристики суперкомпьютеров

Наименование	Фирма	Количество процессоров	Производительность, Тфлоп	Пиковая производительность, Тфлоп
Earth-Simulator	NEC	5120	35,860	40,960
ASCI Q-AlphaServer	HP	8192	13,880	20,480
Seaborg	IBM	6656	7,304	9,984
PRIMEPOWER HPC2500	Fujitsu	2304	5,406	11,980

Наиболее распространенные суперкомпьютеры — массово-параллельные компьютерные системы. Они имеют десятки тысяч процессоров, взаимодействующих через сложную, иерархически организованную систему памяти.

В качестве примера рассмотрим характеристики многоцелевого массово-параллельного суперкомпьютера среднего класса Intel Pentium Pro 200. Этот компьютер содержит 9200 процессоров Pentium Pro на 200 МГц, в сумме (теоретически) обеспечивающих производительность 1,34 Тфлоп (1 Тфлоп равен  $10^{12}$  операций с плавающей точкой в секунду), имеет 537 Гбайт памяти и диски емкостью 2,25 Тбайт. Масса системы 44 т, а кондиционеров для нее — 300 т; потребляемая мощность 850 кВт.

Суперкомпьютеры используют для решения сложных и ресурсоемких научных задач (метеорология, гидродинамика и т.п.), в управлении, разведке, в качестве централизованных хранилищ информации и т.д. Среди типичных задач для суперкомпьютера можно перечислить следующие: моделирование физических процессов, расшифровка кодов (в том числе генетических), обработка больших массивов информации, управление распределенной сетью абонентских компьютеров.

Некоторые технические характеристики современных суперкомпьютеров приведены в табл. 5.1.

В настоящее время АРМ проектировщиков, как правило, построены на базе микроЭВМ, которые могут обращаться к ЭВМ более высокого уровня, в случаях, когда ресурсов микроЭВМ (особенно памяти) недостаточно. Иерархическая структура такого вычислительного комплекса представлена на рис. 5.1.

СуперЭВМ для автоматизации проектирования обычно не применяются, однако они используются для решения сопутствующих

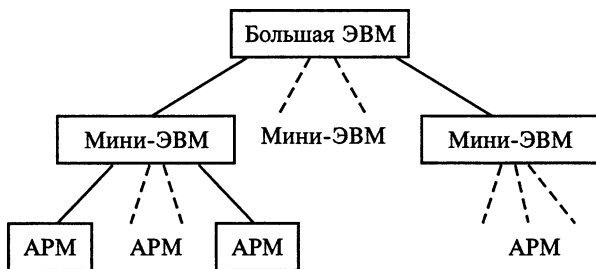


Рис. 5.1. Иерархическая структура вычислительного комплекса

щих задач, связанных, например, с физикой процессов, происходящих в различных технических системах.

### 5.3. Периферийные устройства ЭВМ

Для выполнения автоматизированного проектирования рабочее место проектировщика должно быть оборудовано всеми необходимыми техническими средствами. Комплекс технических средств, предназначенный для использования конкретным проектировщиком в САПР, называется автоматизированным рабочим местом (АРМ).

Впервые это понятие в нашей стране появилось в связи с тем, что на заре возникновения САПР существовали ЭВМ, пригодные для автоматизации проектирования, и различные периферийные устройства, необходимые в процессе автоматизированного проектирования: графопостроители, кодировщики графической информации, графические дисплеи и др. Однако зачастую не существовало устройств сопряжения этой техники с ЭВМ, программного обеспечения, которое, с одной стороны, обеспечивало бы «взаимопонимание» между устройствами, входящими в систему, а с другой — позволяло бы выполнить некоторые этапы автоматизированного проектирования. Поэтому была осуществлена идея комплексирования ЭВМ с периферийными техническими устройствами с соответствующим программным обеспечением. Именно тогда и появилось понятие АРМ; они разделялись по назначению (АРМ-М для машиностроения, АРМ-Р для радиоэлектроники и др.) и в зависимости от этого комплектовались различными устройствами и программными средствами.

В настоящее время вопросы стандартизации аппаратного и программного сопряжения ЭВМ и любых устройств, работающих под ее управлением, решены и практически нет проблем в формировании любого рабочего места проектировщика, которое по тра-

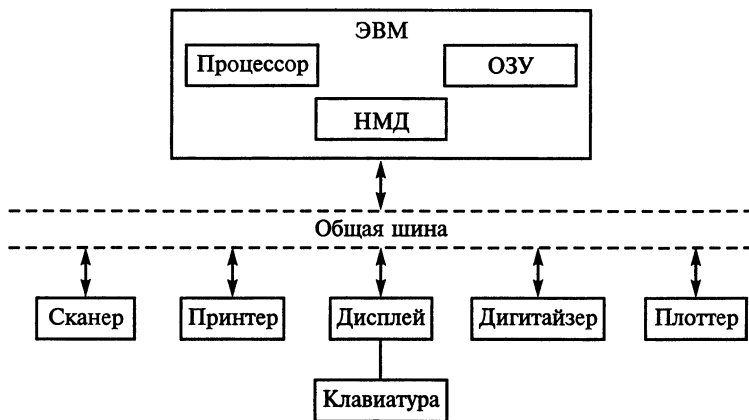


Рис. 5.2. Схема автоматизированного рабочего места

диции носит название АРМ. Возможная схема АРМ для выполнения автоматизированного проектирования машиностроительных конструкций представлена на рис. 5.2.

Периферийные устройства — устройства ЭВМ, используемые для ввода, вывода, подготовки данных и запоминания больших объемов информации. Отличительная особенность периферийных устройств состоит в том, что они в процессе работы преобразуют форму представления информации, не изменяя ее содержания. Быстрое совершенствование центральных устройств ЭВМ, уменьшение их размеров, постоянное снижение стоимости привели к повышению роли периферийных устройств. Так, уже сейчас стоимость периферийных устройств составляет большую часть стоимости ЭВМ, а их габаритные размеры определяют размеры помещения для установки ЭВМ. В значительной мере это объясняется тем, что периферийные устройства в основном являются электромеханическими, быстродействие, надежность, габаритные размеры и другие характеристики которых ограничены.

Ниже будут рассмотрены периферийные устройства, которые обычно входят в типовой комплект устройств ЭВМ (внешние запоминающие устройства, устройства ввода-вывода информации, устройства подготовки данных).

**Внешние запоминающие устройства.** В иерархической структуре памяти ЭВМ в качестве запоминающего устройства большой емкости применяются устройства с записью информации на подвижный магнитный носитель (обычно это накопители на магнитных дисках — НМД); эти запоминающие устройства являются внешними по отношению к ОЗУ и поэтому называются внешними запоминающими устройствами (ВЗУ). Они позволяют увеличить емкость памяти ЭВМ до десятков и сотен гигабайт, что не-

обходимо для САПР, оперирующих с большими объемами справочной и проектной информации. Данные, хранящиеся в ВЗУ, непосредственно центральным процессором не обрабатываются.

Накопители на магнитных дисках используются для оперативного хранения больших массивов информации. Как правило, на магнитных дисках (МД) хранятся многократно используемые программы, справочные данные и т. п. Накопители на магнитном диске имеют большую емкость и малое время поиска при сравнительно невысокой стоимости хранения бита информации.

Накопитель на магнитном диске содержит пакет МД и его привод, блок магнитных головок и механизм их позиционирования, электронные схемы, обеспечивающие запись и воспроизведение информации, коммутацию магнитных головок и др.

Поверхности МД используются для хранения информации, записываемой по концентрическим дорожкам. Дорожки одного и того же диаметра на всех рабочих поверхностях образуют цилиндр. Понятие «цилиндр» используется при размещении и поиске информации в НМД.

Наборы данных, состоящие из записей постоянной или переменной длины (блоков), записываются вдоль магнитных дорожек. Для записи и считывания информации используются плавающие магнитные головки, расположенные на подвижных рычагах.

Накопители на магнитных дисках классифицируются по ряду признаков.

По материалу основы диска НМД разделяют на накопители на жестких и накопители на гибких магнитных дисках (ГМД).

По числу дисков различают НМД однодисковые и с пакетом дисков.

Жесткие МД изготавливаются из алюминиевого сплава. На поверхность дисков наносится магнитное покрытие. Для увеличения скорости передачи данных требуется увеличивать скорость движения носителя. В современных НМД частота вращения дисков составляет тысячи оборотов в минуту. Во избежание интенсивного износа магнитных головок и носителя используется бесконтактный способ записи с плавающими головками. При этом между магнитными головками и поверхностью носителя создается зазор в 3...5 мкм за счет подъемной силы, действующей на специальный башмак, удерживающий головки. Современные накопители на жестких магнитных дисках имеют емкость тысячи гигабайт.

До последнего времени в качестве ВЗУ мини- и микроЭВМ широко использовались накопители на сменных ГМД. Такие диски выполняются на лавсановой основе толщиной 0,12 мм, на которую наносится магнитное покрытие. Гибкий магнитный диск постоянно находится в квадратной кассете (дискете), имеющей прорезь, через которую осуществляется взаимодействие магнитной головки с магнитной поверхностью диска. Накопители на ГМД

компактны и дешевы, однако их емкость (как правило, 1,44 Мбайт) зачастую недостаточна для хранения больших массивов информации, характерных для САПР.

В настоящее время ГМД вытесняются устройствами чтения-записи оптических (лазерных) дисков (CD-ROM). Приводы CD-ROM подразделяются на устройства, предназначенные только для чтения компакт-дисков, и устройства, позволяющие производить как чтение, так и запись информации. Для этих устройств используются два типа компакт-дисков: CD-R, позволяющие однократно записать информацию, и CD-RW, на которых возможна перезапись. Емкость одного диска обычно составляет до 700 Мбайт, т.е. на нем может уместиться информация, содержащаяся на 500 ГМД.

Дальнейшим развитием оптических способов хранения информации являются приводы DVD-ROM, в которых информация также содержится на специальных компакт-дисках, но ее объем на порядок больше, чем на CD-ROM.

Применявшиеся ранее накопители информации на магнитных лентах, магнитных барабанах и на бумажных носителях (перфокартах и перфолентах) либо ушли в прошлое, либо доживают последние дни в устаревших вычислительных системах.

**Дисплеи.** Дисплеи предназначены для визуального отображения информации и представляют собой наиболее перспективную группу терминалов. Они дают возможность пользователю оперативно вводить и корректировать вводимую информацию, компоновать ее в соответствии с требуемым форматом и видоизменять по своему усмотрению. Данные выводятся на экран дисплея в цифровой, буквенно-цифровой, графической и комбинированной формах.

Дисплеи пришли на смену печатающим устройствам ввода-вывода информации и в настоящее время полностью их вытеснили.

По сравнению с печатающими терминалами дисплеи имеют следующие достоинства:

- большая скорость приема сообщений;
- возможность редактирования вводимой информации;
- бесшумность в работе;
- ориентация на непрофессионального пользователя;
- отсутствие бумажного носителя для отображения вводимой и выводимой информации.

Последние достижения науки и техники позволили создать различные по назначению, структуре и характеристикам видео-терминальные устройства, которые широко используются для связи человека с ЭВМ при интерактивных режимах проектирования.

Структурная схема дисплея представлена на рис. 5.3.

Устройство сопряжения дисплея предназначено для организации приема и передачи информации и обеспечивает требуемый для работы протокол обмена информацией. Память дисплея играет существенную роль в функционировании всего устройства. За-

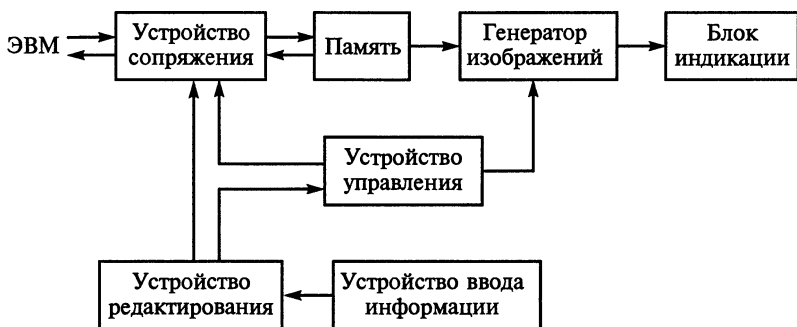


Рис. 5.3. Структурная схема дисплея

помятая принятую от оператора или ЭВМ информацию, блок памяти, как правило, имеет объем, превышающий объем кадра изображения и может колебаться в достаточно больших пределах.

Так как в качестве блока индикации в большинстве современных дисплеев используется электронно-лучевая трубка (ЭЛТ) с малым временем послесвечения, то для надежного восприятия человеческим глазом отображенной информации требуется ее циклическое повторение (регенерация) на экране ЭЛТ с частотой не менее 50 Гц. Современные дисплеи на ЭЛТ, как правило, имеют частоту до 100 Гц и выше. Выполнение функций регенерации информации в подобных дисплеях также лежит на блоке памяти.

Принцип действия монитора на базе ЭЛТ заключается в том, что испускаемый электродом (электронной пушкой) пучок электронов, попадая на экран, покрытый специальным составом — люминофором, вызывает его свечение. Направление пучка электронов задают также дополнительные электроды: отклоняющая система, позволяющая изменять направление пучка, и модулятор, регулирующий яркость получаемого изображения. Электронный луч периодически сканирует экран, образуя на нем строки развертки. По мере движения луча по строкам видеосигнал, подаваемый на модулятор, изменяет яркость определенных пикселей, образуя некоторое видимое изображение. Разрешающая способность монитора определяется числом пикселей, из которых и формируется изображение на экране.

В цветном мониторе имеются не одна, а три электронные пушки с отдельными схемами управления, а на поверхность экрана нанесен люминофор трех основных цветов: красного, зеленого, синего. Таким образом, каждая пушка «стреляет» по своей цели. Для этого в каждом мониторе имеется либо теневая маска, либо апертурная решетка; теневая маска имеет систему отверстий, а апертурная решетка — систему щелей.

В основном применяются кинескопы двух типов: плоско-прямоугольные (наиболее распространены) и вертикально-плоские. В плоско-прямоугольных кинескопах экраны на самом деле не плоские и не прямоугольные, они выглядят такими по сравнению с обычными кинескопами за счет большого радиуса кривизны. В вертикально-плоских кинескопах экран действительно плоский по вертикали, а по горизонтали имеет кривизну (как вырезанный из цилиндра). Наиболее известны вертикально-плоские кинескопы Trinitron фирмы Sony.

Четкость изображения определяется так называемым шагом триад — расстоянием (в мм) между двумя ближайшими точками люминофора одного цвета свечения. Чем меньше шаг триад, тем выше потенциальная четкость изображения. Типичные значения для шага триад для современных дисплеев составляют 0,28; 0,26; 0,24 мм.

Важным параметром является разрешение монитора экрана — количество точек, которое может отображать монитор. Чем больше точек он может отобразить, тем более высокого качества изображения можно достичь. Существуют стандартные режимы:  $800 \times 600$ ,  $1024 \times 768$ ,  $1152 \times 864$  и т. д. Первая цифра — это количество точек по горизонтали, вторая — количество точек по вертикали; чем выше указанные числа, тем четче изображение.

Другим важным параметром дисплея является его размер, определяемый размером экрана кинескопа по диагонали. В настоящее время стандартными являются следующие размеры: 15, 17, 19 и 21 дюйм. Пятнадцатидюймовый монитор постепенно выходит из употребления, так как для использования в целях САПР он неудобен. Наибольшее распространение получили мониторы с размером диагонали 17 дюймов, как оптимальные с точки зрения соотношения цена — качество.

Дисплей по своим функциям представляет собой преобразователь двоичного кода, поступающего на его вход из ЭВМ, в символьную или графическую информацию на экране. Непосредственное преобразование осуществляет генератор изображения совместно с блоком индикации и устройством управления.

В общем случае генератор изображения преобразует двоичный код символа либо фрагмента графического изображения (точки, вектора, части окружности) в код, который, поступая в блок индикации, заставляет его вычертить требуемый символ или графическую фигуру на экране. Блок индикации во многом определяет общую структуру дисплея и предназначен для формирования и отображения информации на экране.

По методу формирования изображения на экране дисплея можно подразделить на точечные, векторные и телевизионные (растровые).

В точечных дисплеях изображения формируются путем последовательной передачи координат каждой точки в блок управле-

ния, который вырабатывает необходимые управляющие сигналы для вычерчивания. При близком расположении точек создается впечатление непрерывной линии.

В векторных дисплеях имеется специальный блок — генератор векторов, предназначенный для быстрого вычерчивания отрезков линий.

Преимущественное применение получили растровые методы формирования изображений на экране.

По назначению и виду отображаемой информации дисплеи делятся на алфавитно-цифровые и графические.

Современные растровые дисплеи сочетают возможности алфавитно-цифровых и графических дисплеев. Именно они и используются для организации рабочего места проектировщика в САПР.

В нише профессиональной графики, в том числе используемой САПР, т. е. там, где требуется безупречное качество изображения и цветопередача, ЭЛТ-мониторам на сегодняшний день альтернативы нет. Мониторы с ЭЛТ превалируют над всеми остальными, поскольку являются наиболее дешевыми из моделей с плоским экраном, размеры и качество изображения которого неуклонно растут; но постепенно их популярность начинает падать, и на смену им приходят плоские дисплеи, занимающие значительно меньше места, легкие и потребляющие меньше энергии. Здесь, однако, следует сделать оговорку: речь идет только о рынке домашних и офисных компьютеров.

В последнее время все более широкое распространение получают дисплеи с жидкокристаллическим (ЖК) экраном. Экран ЖК-монитора представляет собой массив пикселей (элементов изображения), которыми можно управлять для отображения информации. Дисплей создается из двух стеклянных пластин, расположенных на расстоянии нескольких микрон друг от друга и содержащих тонкий слой жидких кристаллов между собой. На верхнюю стеклянную пластину наносится трафарет, который играет роль фильтра цвета (как и в ЭЛТ, каждый пиксель матрицы строится из трех компонентов цвета — красного, зеленого и синего). Затем на обе стеклянные пластины наносятся поляризационные фильтры, которые пропускают только тот компонент светового пучка, ось поляризации которого соответствует заданной; так создается ячейка ЖК-дисплея.

Общий принцип действия ЖК-мониторов следующий. Свет от неоновой лампы проходит через систему отражателей, направляется через первый поляризационный фильтр и попадает в слой жидких кристаллов, контролируемый транзистором; затем свет проходит через цветные фильтры. Транзистор создает электрическое поле, задающее пространственную ориентацию жидких кристаллов. Свет, проходя через такую упорядоченную молекулярную структуру, меняет свою поляризацию, и в зависимости от



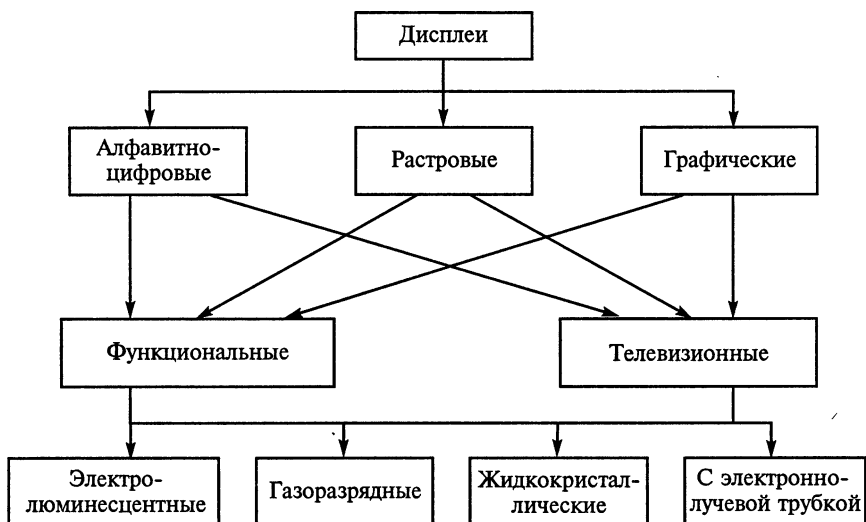


Рис. 5.4. Классификация дисплеев

нее будет либо полностью поглощен вторым поляризационным фильтром на выходе (образуя черный пиксель), либо не будет поглощаться или поглотится частично (образуя различные цветовые оттенки, вплоть до чистого белого).

Поляризация, лежащая в основе ЖК-технологии, имеет и свои минусы, главный из которых сокращение угла обзора ЖК-дисплея. Производители ЖК-панелей это учитывают; в настоящее время созданы технологии, позволяющие если не искоренить, то хотя бы значительно уменьшить этот недостаток.

Классификация дисплеев приведена на рис. 5.4.

В настоящее время наибольшее распространение получили дисплеи фирм Sony, NEC, Mitsubishi.

**Устройства ввода и редактирования информации.** Данные устройства предназначены для ввода и редактирования как символьной, так и графической информации. В их состав могут входить всевозможные клавиатуры, джойстики, планшеты, манипуляторы (мышь), световое перо.

**Печатающие устройства.** Среди разнообразной номенклатуры периферийных устройств печатающие устройства (ПУ) занимают ведущее место. Они используются в составе ЭВМ всех типов, от персональных ЭВМ до суперЭВМ.

Выпуск ПУ непрерывно возрастает; он составляет более 50 % общего производства периферийного и терминального оборудования.

В зависимости от порядка вывода информации на носитель записи различают посимвольные, построчные и постраничные ПУ.

Посимвольные ПУ выводят алфавитно-цифровую информацию на носитель записи последовательно символ за символом. При этом за один цикл печати формируется только один знак.

Построчные ПУ формируют и выводят за один цикл печати всю строку, делая ее доступной для восприятия пользователем сразу же после завершения процесса.

Постраничные ПУ формируют и выводят за один цикл печати целиком страницу. Основное отличие постраничных устройств состоит в том, что информация поступает пользователю сразу же отпечатанной на отдельной или соединенной с другими странице.

По принципу формирования изображений символов на носителе записи различают литерные (полносимвольные) и матричные (знакосинтезирующие) ПУ. В литерных ПУ изображение формируют одновременно по всей поверхности символа при единичном воздействии на носитель записи печатающей головки, например молоточка. В матричных ПУ изображение символов формируют из отдельных элементов — точек последовательно или последовательно-параллельно при многократном воздействии на носитель записи.

Вначале наибольшей популярностью пользовались литерные ПУ. По мере развития техники печати и расширения функциональных возможностей как самих ПУ, так и управляющих ими ЭВМ, большое распространение получили матричные ПУ; это было обусловлено возможностью последних практически неограниченно расширять номенклатуру и число наборов символов, печатаемых ПУ, возможностью вывода не только алфавитно-цифровой, но и графической информации, возможностью вывода многоцветных и полутонных изображений. При этом качество получаемого изображения благодаря возможности повышения плотности печати во многих случаях не уступает полиграфическому.

В качестве носителя записи в ПУ используют, как правило, бумагу в рулонах или пачках с краевой перфорацией и без нее. Наряду с этим в постраничных устройствах преимущественно применяют бумагу в виде листов, причем во многих даже низкопроизводительных ПУ предусмотрена автоматическая подача листов на печать.

Наибольшее распространение получили ПУ, использующие в качестве носителя записи обычную бумагу. Следует отметить, что под обычной часто понимают бумагу широко распространенных марок, таких, как писчая, типографская и др. Однако в ПУ рекомендуют использовать бумагу вполне определенных марок, обеспечивающих наилучшее качество печати, при этом учитывают ее гладкость, белизну, степень проклейки, удельную плотность и многие другие показатели.

Преимущественное применение обычной бумаги объясняется ее меньшей стоимостью, большей доступностью и, как правило,

более высоким качеством печати по сравнению с печатью на специальной бумаге. Специальной называют бумагу, которая содержит дополнительные слои или добавки в бумажную массу, благодаря чему обеспечиваются требуемые термочувствительность, ударочувствительность, диэлектрические или электропроводные свойства и др.

Большое влияние на технические показатели, конструктивные особенности, эксплуатационные и другие характеристики ПУ оказывает физический принцип — способ регистрации, на основе которого оно реализовано. При этом различают ПУ ударного и безударного принципов действия.

При ударном действии получают изображения символьной и графической информации в результате удара по носителю записи органом записи — печатающим знаком, молоточком, стержнем.

При безударном действии изображение на носителе записи получают в результате физико-химического или другого вида воздействия на окончательный носитель записи, поступающий к пользователю, или на некоторый промежуточный носитель записи, входящий, как правило, неотъемлемой частью в состав ПУ. С промежуточного носителя записи или с его помощью информацию переносят на окончательный носитель записи.

*Печатающие устройства ударного принципа действия* имеют богатую предысторию, поскольку многие технические решения на первых этапах развития были заимствованы из техники пишущих (печатающих) машинок с ручным приводом.

При ударном принципе изображения на бумаге получают механическим воздействием (ударом) печатающего элемента на бумагу, как правило, через красящую ленту, из которой выдавливается краситель.

В ряде устройств используется прямая печать, при которой краситель (краска) наносится (накатывается) непосредственно на поверхность литеры и далее при ударе переносится на бумагу. При использовании ударочувствительной бумаги ее цвет изменяется без дополнительного красносителя.

Печатающие устройства ударного принципа действия разработаны двух классов — посимвольные и построчные.

Посимвольные ПУ первоначально создавались рычажного типа, в которых литера — рельефное изображение каждого знака закреплена на отдельном рычаге. В дальнейшем в качестве литеро(шрифто)носителя использовали шаровые и цилиндрические головки с литерами на их поверхности. В последние годы получили распространение лепестковые шрифтоносители, представляющие собой диск с радиально расположенными по его периферии лепестками, на концах которых имеются литеры.

Во всех рассмотренных конструкциях шрифтоносителя выборка знаков производится путем механического перемещения опре-

деленной литеры на ударную позицию. Удар литеры по бумаге обеспечивается, как правило, электромагнитным приводом.

В настоящее время наибольшее распространение получают многоэлементные матричные печатающие головки, каждый печатающий элемент которых при воздействии на носитель записи создает отдельную точку, а комбинация точек формирует изображение знака.

Печатающий элемент головки представляет собой тонкий стержень, соединенный с автономным быстродействующим электроприводом (электромагнитным или пьезоэлектрическим). Печатающие элементы располагаются в один или несколько рядов и, как правило, вдоль движения носителя. При перемещении такой головки поперек движения носителя и возбуждении определенных элементов образуются отдельные знаки и вся печатная строка.

Построчные ПУ реализуются обычно одним из следующих вариантов исполнения шрифтоносителя.

В литерных ПУ шрифтоноситель выполняют в виде цилиндра или замкнутой ленты. В первом случае образующая шрифтоносителя поперечна перемещению носителя записи. По окружности цилиндра для каждой печатной позиции имеются рельефные изображения знаков. Выборка каждого знака производится при вращении цилиндрического шрифтоносителя, а печать — ударом по выбранной литере через красконоситель и бумагу печатающим молоточком без остановки шрифтоносителя — так называемая «печать на лету». Недостатками ПУ с цилиндрическим шрифтоносителем являются большая масса последнего, его громоздкость и практическая невозможность замены шрифтов. Во втором случае шрифтоноситель имеет форму эластичной бесконечной ленты, перемещаемой поперек движения носителя записи. Печать осуществляется без остановки шрифтоносителя на печатной позиции, так же как и для цилиндрического шрифтоносителя. Смена шрифта сравнительно просто производится путем замены ленточного шрифтоносителя.

В построчных матричных ПУ в отличие от посимвольных матричных ПУ элементы формируются одновременно для всех знаков и последовательно элемент за элементом для каждого знака, а не последовательно-параллельно. Печатающие элементы перемещаются поперек носителя записи, причем на каждую печатную позицию приходится, как правило, по одному печатающему элементу и по одному электромагниту.

Как посимвольные, так и построчные ПУ ударного принципа действия обеспечивают одно- и многоцветную печать. Последнюю получают, используя красящую ленту, у которой отдельные зоны пропитаны красителями разных цветов. Печать разными цветами осуществляется автоматической сменой зон красящей ленты.

Печатающие устройства ударного действия сравнительно просты по конструкции, но имеют много элементов, работающих в режиме больших динамических нагрузок; этим обусловлен ограниченный ресурс элементов и повышенный уровень шума при работе ПУ.

*Печатающие устройства безударного принципа действия* (безударные печатающие устройства — БПУ) свободны от ряда недостатков ПУ ударного действия.

Применяются три класса БПУ — посимвольные, построчные и постраничные. Во всех классах БПУ изображения символьной и графической информации синтезируют из отдельных точек — элементов изображения с плотностью записи (печати) 3—32 точек/мм. По скорости печати выпускаемые многочисленные модели этих устройств значительно отличаются друг от друга — от десятков символов в секунду в БПУ малой производительности до сотен тысяч символов в секунду в наиболее быстродействующих моделях.

Наибольшее распространение при реализации БПУ получили следующие способы регистрации: электрографический (электрофотографический); феррографический (магнитографический); электростатический и его разновидность — электроионный; термический; струйный. Ожидается, что эти же способы регистрации будут в основном использоваться при создании БПУ и в дальнейшем.

*Электрографические БПУ.* В основе всех электрографических БПУ лежит электрофотографический способ регистрации, при котором создают скрытое электрическое изображение на промежуточном носителе записи с фотопроводниковым слоем на поверхности, визуализируют это изображение мелкодисперсным красящим порошком — тонером, получая при этом порошковое изображение, которое затем переносят на оконечный носитель записи — бумагу, где и закрепляют — фиксируют, например термическим способом. В электрографических БПУ для создания скрытого электрического изображения на фотопроводниковый носитель записи воздействуют электромагнитным излучением, источниками которого могут быть лазеры, светодиоды, светоклапанные системы или электронно-лучевые трубки.

В ряде случаев электрографические БПУ различают по виду используемого в них источника излучения (лазерные, светодиодные, светоклапанные или электронно-лучевые).

В высокоскоростных лазерных БПУ в качестве источника излучения применяют газовые He—Ne лазеры; в средне- и низкоскоростных лазерных БПУ — полупроводниковые лазеры. Во всех лазерных БПУ развертку лазерного луча вдоль строки производят электромеханическим путем с помощью вращающегося зеркального многогранника или призмы. Светодиодные и светоклапанные системы записи применяют в средне- и низкоскоростных БПУ.

Светодиодные системы записи представляют собой светодиодную линейку, излучение которой проецируется с помощью объективов или самофокусирующих световодов на промежуточный фотопроводниковый носитель записи.

В светоклапанных системах записи электромагнитное излучение, создаваемое, например, лампой накаливания, проецируется на фотопроводниковый носитель записи через световые затворы — магнитооптические или жидкокристаллические.

Сопоставление рассмотренных оптических систем записи показывает, что по многим параметрам они идентичны.

Электронно-лучевые системы записи в электрографических БПУ в настоящее время применения не находят из-за больших габаритных размеров, низкой надежности и ограниченного срока службы ЭЛТ.

В качестве промежуточного носителя электрографических устройств используют, как правило, барабаны с цилиндрической поверхностью, покрытой фотопроводниковым слоем, — электрофотографические цилиндры. В некоторых высокопроизводительных устройствах в качестве промежуточного носителя используют гибкий электрофотографический носитель, основой которого является пластмассовая пленка, покрытая фотопроводниковым слоем с металлическим подслоем. Применение гибкого носителя усложняет устройство, однако по сравнению с устройством с электрофотографическим цилиндром стоимость его эксплуатационных расходов меньше.

Основными требованиями, предъявляемыми к электрофотографическим слоям, являются:

- электрооптическая чувствительность к электромагнитному излучению определенного спектрального состава, воздействующему на слой в процессе записи;
- высокий электростатический контраст скрытого электрического изображения;
- высокая износостойкость (тиражеустойчивость) и др.

В качестве фотопроводникового слоя применяют неорганические вещества типа триселенида мышьяка, селен-теллура или органические фотопроводниковые вещества, повышая их чувствительность — сенсибилизируя их так, чтобы обеспечить наибольшую чувствительность в диапазоне длин волн электромагнитного излучения источника света.

Скрытое электрическое изображение во всех современных моделях электрографических БПУ визуализируют с помощью так называемой магнитной кисти одно- или двухкомпонентным проявителем. В однокомпонентном проявителе в состав частиц тонера наряду с красящим веществом входит магнитная составляющая. В двухкомпонентном проявителе частицы тонера, содержащие красящее вещество, смешаны с более крупными магнитными части-

цами носителя тонера и удерживаются на них в результате трибоэлектрического взаимодействия.

При работе узла проявления с двухкомпонентным проявителем из магнитной кисти постоянно удаляется тонер. Для поддержания его постоянной концентрации БПУ имеет средства контроля за концентрацией тонера в проявителе, средства подачи тонера в узел проявления и равномерного распределения тонера в проявителе.

Перенесенное с промежуточного носителя записи на окончательный (бумажный) порошковое изображение закрепляют — фиксируют. В подавляющем числе моделей электрографических БПУ для закрепления изображения используют термический или термосиловый способы фиксации. Первый применяется только в отдельных моделях низко- и среднескоростных БПУ. Вторым способом, при котором носитель с порошковым изображением прокатывают между двух нагретых валов, применяется в большинстве низко-, средне- и высокоскоростных устройств, так как он обеспечивает высокое качество закрепления, более низкое по сравнению с другими способами закрепления потребление энергии и практически полную противопожарную безопасность (даже при аварийных ситуациях бумага, застрявшая между закрепляющими валами, не загорается, так как температура валов не превосходит для разных моделей устройств 120... 200 °С).

Во всех высокоскоростных и некоторых среднескоростных БПУ в качестве окончательного носителя записи используют рулонную бумажную ленту и ленту, сложенную в пачку с поперечными насечками и краевой перфорацией. В ряде моделей БПУ предусмотрена рубка ленты на отдельные листы и обрезка краевой перфорации с последующей сортировкой отпечатанных материалов, что резко упрощает подготовку отпечатанных материалов для пользователя.

Во всех низкоскоростных устройствах печать ведется на листовый бумажный носитель форматов А4, А3 и др. Листы поступают из подающих кассет емкостью 250—500 листов. В ряде устройств имеется по две подающие кассеты: одна для формата А3, другая для формата А4.

В некоторых БПУ наряду с выводом информации, поступающей от ЭВМ, предусмотрено также копирование документов. При этом используются две системы формирования изображений — лазерная система, управляемая сигналами от ЭВМ, и оптическая проекционная система, выполненная по традиционной схеме.

В зависимости от назначения БПУ, его быстродействия и особенностей использования в широких пределах изменяются структурная схема блока управления и объем его внутренней памяти. Высокоскоростные БПУ, обслуживающие мощные ЭВМ, осна-

щаются буферной памятью объемом в несколько страниц, растровой памятью, в которой формируется электрический образ регистрируемого изображения, генератором символов, преобразователями векторной формы описания графики в растровую, а также мощной системой контроля и диагностики состояния устройства. В средне- и низкоскоростных БПУ часть функций перекладывается на ЭВМ и ПЭВМ, в частности по преобразованию вектор—растр, и упрощается обмен информацией с ЭВМ.

В большинстве устройств предусматриваются интерфейсы Centronics.

Практически все электрографические БПУ предназначены для одноцветной (черно-белой) печати, хотя имеются и многоцветные устройства.

*Феррографические БПУ.* Структурные схемы феррографических БПУ аналогичны схемам электрографических БПУ. Скрытое изображение также создают на промежуточном носителе записи, затем визуализируют мелкодисперсным порошком — тонером, переносят на бумажный носитель и закрепляют. Основное отличие между электрографическими и феррографическими устройствами состоит в том, что в феррографических БПУ создают скрытое магнитное, а не электрическое изображение. Для этого в качестве промежуточного носителя записи в феррографических БПУ используют магнитные барабаны или ленты; для записи информации применяют многодорожечные блоки магнитных головок; тонер обязательно должен быть магниточувствительным.

Поскольку в качестве промежуточного носителя, как правило, используют магнитные барабаны с металлическим рабочим слоем, их износостойчивость на много порядков выше, чем у фотопроводниковых носителей, применяемых в электрографии.

В связи с тем, что в феррографических БПУ не нужно защищать рабочую зону от воздействия света, конструкция этих устройств получается более простой, при этом улучшаются условия их регулировки и обслуживания.

Отмеченные достоинства феррографии обусловили интерес к ней разработчиков БПУ. На коммерческий рынок выпущено несколько моделей феррографических БПУ среднего быстродействия.

Основная проблема феррографических устройств практически всех разновидностей заключается в обеспечении эффективного взаимодействия магнитных головок с магнитным носителем. Для этого необходимо сократить до минимума зазор между рабочей поверхностью магнитной головки и носителя (до 0,01...0,02 мм). Однако наличие мелкодисперсного тонера, также взаимодействующего с магнитным носителем, приводит к тому, что промежуток между магнитной головкой и носителем постепенно заполняется тонером; при этом либо увеличивается зазор, либо усиленно изнашивается магнитная головка, либо происходит и то и другое.



Диапазон распространения феррографических БПУ будет определяться тем, насколько эффективной окажется конструкция магнитной головки и удачными решения по предотвращению засорения зазора магнитная головка — магнитный носитель записи.

*Электростатические БПУ.* Электростатическая регистрация состоит в создании скрытого электрического изображения на диэлектрической поверхности оконечного или промежуточного носителя.

Электростатические БПУ без промежуточного носителя получили незначительное распространение. В них запись ведется на специальную электростатическую бумагу, рабочая поверхность которой имеет тонкий диэлектрический слой, а основа бумаги пропитана гидрофильными солями, позволяющими получить требуемую для нее влажность и электропроводность. Для записи информации на такой носитель используют одно-, двух- и многорядные записывающие головки, представляющие собой блок тонких (диаметром около 0,1 мм) электродов, расположенных соответственно в один-два ряда и более. При подаче высоковольтного импульса напряжения между записывающим электродом и противоэлектродом на диэлектрической поверхности бумаги образуется элемент скрытого электрического изображения. Затем электростатическая бумага протягивается через узел проявления, в котором диспергированные в жидкой органической среде частицы красителя визуализируют скрытое изображение. Плотность записи в таких устройствах составляет 4—8 точек/мм и более. Скорость регистрации определяется скоростью записи, проявления и высыхания растворителя. Обычно электростатическая бумага движется в устройстве со скоростью 10...50 мм/с, хотя известны отдельные устройства, в которых она перемещается со значительно большей скоростью (до 250 мм/с). Современная технология производства электростатической бумаги позволила снизить ее стоимость до уровня высококачественной типографской бумаги.

Основная область применения электростатических БПУ без промежуточного носителя — вывод графической информации, включая схемную и конструкторскую документацию. Возможность вывода информации на широкий (свыше 1000 мм) носитель делает электростатические устройства особенно полезными в системах автоматизированного проектирования.

Наибольший интерес представляют устройства с многоцветным (4—7 цветовых оттенков и более) выводом информации. Для уменьшения габаритных размеров, массы и стоимости устройств многоцветную печать, как правило, получают при многократном проходе бумажного носителя мимо одних и тех же записывающих головок, подключенных к единому блоку управления. Изменение цвета изображения достигается включением при каждом проходе носителя соответствующего узла проявления.

Более широкое применение находят электростатические устройства с промежуточным носителем. Их особенность состоит в том, что скрытое электрическое изображение получают на цилиндре с диэлектрическим покрытием, которое проявляют мелкодисперсным тонером, аналогичным используемому в электрографических устройствах.

*Термические БПУ.* Способ термопечати основан на двух принципиально различных схемах — без промежуточного (донорного) носителя и с промежуточным носителем. Для реализации каждой из них локально воздействуют на окончательный или промежуточный носитель теплотой, выделяемой записывающей (термопечатающей) головкой, которая содержит от нескольких единиц до нескольких тысяч отдельных элементов.

В термопечатающих БПУ, выполненных по первой схеме (без донорного носителя), окончательный носитель — бумага покрыта или пропитана теплочувствительным веществом, которое при локальном нагреве в результате термохимической реакции изменяет цвет, образуя визуально наблюдаемое изображение точки. Достоинством устройств, выполненных по схеме с термочувствительной бумагой, является их простота. Кроме механизма перемещения бумаги в контакте с термопечатающими головками и в некоторых случаях механизма перемещения головки, в таких устройствах практически отсутствуют какие-либо подвижные элементы. По этой причине, а также в силу малой энергоемкости процесса записи устройства этого типа нашли преимущественное применение в первую очередь в малоразрядных калькуляторах и персональных ЭВМ для посимвольного или построчного вывода информации.

К недостаткам устройств, выполненных без промежуточного носителя, относятся сравнительно невысокое качество изображения (плотность записи обычно не более 3—5 точек/мм), повышенная в 2—3 раза по сравнению с обычными потребительскими сортами бумаги стоимость термобумаги, ограниченность ресурса термопечатающих головок, чувствительность бумаги к изменениям температуры окружающей среды, особенно при длительном хранении. Правда, надо отметить, что последний недостаток для большинства современных марок бумаги преодолен. Кроме того, устранен серый или цветной фон бумаги, повышена контрастность получаемого изображения. Тем не менее число моделей БПУ, предназначенных для печати на термочувствительную бумагу, постоянно уменьшается.

Термопечатающие БПУ, выполненные с промежуточным носителем, находят все большее распространение. По этой схеме между термопечатающей головкой и окончательным носителем, в качестве которого используют бумагу, пластмассовую пленку (часто прозрачную) и т. п., размещают донорный носитель — копируемую пленку. Последняя представляет собой тонкую (толщи-

ной 5...10 мкм) пластмассовую планку (например, лавсановую), покрытую со стороны, обращенной к оконечному носителю, красящим слоем. Особенностью этого слоя является низкая (менее 100 °С) температура оплавления связующего вещества красящего слоя, в котором диспергирован краситель.

При контакте оконечного носителя с красящим слоем донорного носителя и при кратковременном прогреве его термопечатающей головкой красящий слой локально оплавляется и практически полностью переходит на оконечный носитель, создавая на нем элемент изображения. Его цвет определяется цветом красящего вещества. При последовательном переносе элементов на один и тот же оконечный носитель с донорных носителей разных цветов получают многоцветные изображения. Допускается наложение одних отпечатков на другие, что расширяет цветовую гамму изображения.

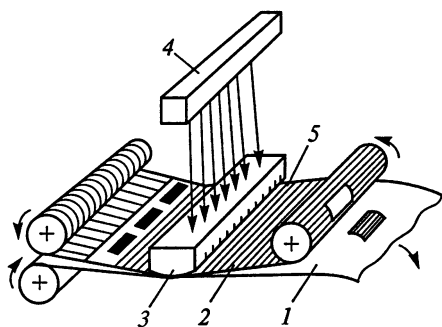
Преимуществами данной схемы термопечати являются высокое качество получаемых изображений (плотность записи в отдельных устройствах достигает 6—12 элементов изображения/мм), высокая контрастность, практически любая цветовая гамма получаемых изображений, возможность печати практически на все известные носители информации, отсутствие фона на пробельных местах изображения и др. Ввиду того, что термопечатающая головка контактирует с поверхностью химически нейтральной пластмассовой пленки, имеющей высокую гладкость, срок службы головки на много порядков больше, чем срок службы головок, используемых в устройствах, выполненных по первой схеме.

Недостатками термопечати с промежуточным носителем являются большой расход донорного носителя для одноцветных устройств, равный, а для многоцветных — по крайней мере в три раза превосходящий расход оконечного носителя, и повышенная сложность устройства. Последнее объясняется тем, что помимо механизма перемещения оконечного носителя печатающее устройство должно содержать еще и механизм перемещения донорного носителя. Для многоцветных устройств должны быть предусмотрены также средства возврата оконечного носителя и его точного позиционирования в исходное перед записью состояние, что необходимо для обеспечения точного совмещения цветов, и механизм отвода термопечатающей головки и донорного носителя от оконечного при его возвратном движении. В ряде устройств такого типа имеется механизм автоматической отрезки листов после завершения цикла многоцветной печати.

В возможной схеме реализации такого БПУ (рис. 5.5) оконечный носитель 1 поступательно перемещается в процессе печати совместно с донорным носителем 2 мимо многоканальной термопечатающей головки 3. При возбуждении ее от блока управления 4 отдельные резистивные элементы головки нагреваются, расплав-

Рис. 5.5. Схема действия термического печатающего устройства:

1 — оконечный носитель; 2 — донорный носитель; 3 — термопечатающая головка; 4 — блок управления; 5 — красящий слой донорного носителя



ляют связующее вещество красящего слоя 5 донорного носителя, в результате чего на оконечном носителе образуется изображение.

**Струйные БПУ.** Длительное время высказывались сомнения по поводу возможности практической реализации струйных БПУ. Причиной этому было характерное для них засорение сопел, из которых выбрасываются капли чернил — краски, посторонними включениями или засыхание чернил в соплах при длительных паузах в работе; при этом работа струйной головки либо полностью прекращалась, либо становилась нестабильной. В настоящее время перечисленные недостатки практически преодолены.

Структурное отличие струйных БПУ от ПУ ударного действия заключается в конструкции печатающей головки, используемом красконосителе и способе его подачи. В настоящее время струйные ПУ выпускаются только последовательного действия (посимвольные или растровые).

В струйных ПУ используют два принципиально различных подхода к формированию чернильных (красящих) капель.

При первом подходе формируют непрерывную капельную струю, причем каждой из капель при вылете из сопла сообщается определенный электрический заряд. Пролетая через одну или две пары отклоняющих электродов, каждая капля в соответствии с управляющими сигналами изменяет траекторию полета и попадает на требуемую позицию оконечного носителя, формируя отдельный элемент изображения, или попадает в улавливатель, откуда собранные чернила перекачиваются насосом в резервуар (емкость с чернилами) и вновь подаются к соплу.

В различных системах формирования капельной струи диаметр сопла составляет 30... 100 мкм. Давление, необходимое для образования струи, в зависимости от диаметра сопла составляет 0,3... 3,0 МПа. Для повышения разрешающей способности и получения полутоновых изображений иногда создают изображения не основными каплями, а следующими за ними сателлитами — микрокаплями, диаметр которых намного меньше диаметра выходного отверстия сопла.

Реализация струйной печати в соответствии с первым подходом достаточно сложна, так как требует введения в состав устройства гидросистемы, поддерживающей давление у соплового отверстия с точностью до нескольких процентов, системы регулирования температуры чернил для устранения влияния изменения вязкости на скорость каплеобразования, источников постоянного и управляемого высоковольтного напряжения и т.д.

При втором подходе к формированию капель, используемом в подавляющем большинстве современных струйных ПУ, капля генерируется по запросу, т.е. с поступлением одиночного управляющего импульса из отверстия сопла вылетает только одна капля. Частота генерации капель по запросу ниже, чем при генерации непрерывной капельной струи, и не превышает 2,5...3 кГц, поэтому для достижения приемлемой скорости печати используют многосопловые струйные головки. Число сопел в головках чаще всего равно 9—12, однако для получения высококачественной печати их число увеличивается до 30, 50 и более. Плотность печати в струйных БПУ составляет 4—40 точек/мм.

Использование многоканальной струйной головки с генерацией капель по вызову исключает необходимость в их отклонении. Для генерации капель по вызову в канале с чернилами, сопряженном с выходными отверстиями сопел, возбуждают ударную волну, которая, дойдя до отверстия сопла, выбрасывает каплю. В струйных головках наиболее широко распространены два способа создания ударной волны — возбуждение пьезоэлемента и нагревание микрорезистора.

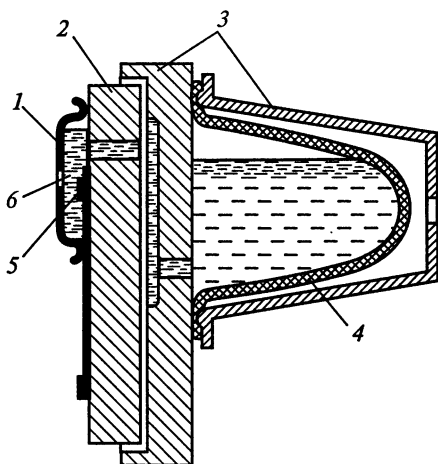
Головка, работающая по способу возбуждения пьезоэлемента, содержит цилиндрический пьезоэлемент, охватывающий капиллярную трубку, которая заполнена чернилами. С одного конца капиллярная трубка оканчивается соплом, к другому концу через фильтр подключена емкость с чернилами. Внутренняя и внешняя цилиндрические поверхности пьезоэлемента имеют токопроводящие покрытия, к которым подключают источник импульсного управляющего напряжения.

Для получения требуемой плотности печати концы капиллярных трубок с сопловыми отверстиями располагают в плоскости по одной, двум или нескольким параллельным линиям. В двух последних случаях сопла каждого ряда располагаются между соплами других рядов.

Достоинством головок с пьезоприводом (пьезоструйных головок) является практически неограниченный срок их службы. По мере расходования чернил заменяют баллончик, который выполняют герметичным, чтобы предотвратить растворение газов из воздуха. В противном случае растворенный газ демпфирует распространение ударной волны и нарушает работоспособность головки.

Рис. 5.6. Термоструйная печатающая головка:

1 — сопловая пластина; 2 — тонкопленочная плата; 3 — корпус; 4 — резиновый колпачок; 5 — микрорезистор; 6 — сопловое отверстие



Недостатками пьезоструйной головки являются ее сложность и повышенная трудоемкость при изготовлении.

Существенно проще по конструкции головка с нагреваемым терморезистором, изготавливаемая по современной технологии.

Такая головка (рис. 5.6) состоит из сопловой пластины 1, тонкопленочной платы 2, корпуса 3 и резинового колпачка 4 — резервуара для чернил. Сопловая пластина 1 имеет сопловые отверстия 6, расположенные над микрорезисторами 5 тонкопленочной платы. Пространство между платой 2 и пластиной 1 постоянно заполняется чернилами из резинового колпачка.

Головка работает следующим образом. При пропускании тока через микрорезистор 5 его температура за 1...2 мкс повышается примерно до 300 °С, чернила вскипают, образуя над резистором газовый пузырек, который выбрасывает каплю чернил из соплового отверстия 6. В зависимости от качества поверхности соплового отверстия траектория полета капли различна и не совпадает с осевой линией сопла, поэтому окончательный носитель располагают на расстоянии 0,3...0,5 мм от сопловой пластины 1 во избежание разброса капель, генерируемых разными соплами.

Разновидностью рассмотренной головки является головка, работающая с твердым красящим веществом, которое при нагревании плавится, переходит в жидкое состояние и в виде капли выбрасывается на бумагу, образуя изображение. При плотности записи 20 точек/мм достигается скорость печати 200—400 знаков/с.

Струйный способ позволяет реализовать не только одноцветную, но и многоцветную печать. Для этого в блоке головок располагают, как правило, четыре группы сопел, каждая из которых связана с емкостью, заполненной чернилами одного из четырех цветов: черного, синего, пурпурного и желтого, что позволяет получать семицветные изображения. При тонкой дозировке и фор-

мировании одного элемента изображения из большого количества микрокапель получают изображения, состоящие из еще большего числа цветов.

Сравнительно недавно предполагалось, что выпуск ПУ, особенно ударного принципа действия, будет сокращаться, однако массовое внедрение ПЭВМ привело к росту производства ПУ. Так, японские фирмы во второй половине 1980-х гг. удвоили выпуск ПУ. Наибольшее развитие в настоящее время получили БПУ постраничные электрографические (лазерные и светодиодные) и посимвольные струйные с термоприводом, в том числе для многоцветной печати.

**Устройства вывода графической информации.** Задача вывода информации из ЭВМ в графической форме возникла одновременно с появлением первых вычислительных машин, так как графическое представление информации является более наглядным, а в ряде случаев, например при автоматизированном способе получения чертежей и графиков, — одной из основных целей функционирования вычислительных комплексов. Устройства вывода графической информации широко применяются в составе технических средств автоматизированного проектирования.

Устройства вывода графической информации можно разделить на два основных класса:

- электромеханические графопостроители векторного типа, в которых пишущее устройство (например, чернильный записывающий элемент) перемещается по двум (планшетные графопостроители) или по одной координате; в последнем случае по второй координате производится перемещение бумажного носителя (барабанные графопостроители);
- растровые устройства вывода графической информации, изображение в которых получается за счет использования различных физических принципов (электростатика, электрография, тепловые процессы и др.).

Классификация устройств вывода графической информации приведена на рис. 5.7.

По принципу действия электромеханические векторные графопостроители делятся на устройства с неподвижным носителем информации и устройства с перемещаемым носителем информации.

Для устройств с неподвижным носителем носитель информации укрепляется на плоской рабочей поверхности планшета. Перемещение пишущего элемента осуществляется электромеханической координатной системой по двум осям; этот тип графопостроителей принято именовать *планшетами*.

Для устройств с перемещаемым носителем информации характерно наличие механизма перемещения пишущего элемента только по одной координате, запись информации по другой оси осуществляется путем перемещения самого носителя.



Рис. 5.7. Классификация устройств вывода графической информации

В зависимости от способа перемещения носителя такие устройства делятся на *устройства с перфорированным носителем*, в которых носитель перемещается транспортным валом за краевую перфорацию, и *устройства с фрикционным перемещением неперфорированного носителя*, в которых перемещение носителя осуществляется за счет частичного или полного носителя валом с фрикционным покрытием.

По производительности устройства делятся на графопостроители с высокой, средней и малой производительностью. Производительность электромеханических графопостроителей определяется динамическими параметрами устройства: максимальной скоростью и ускорением пишущего элемента.

По точности устройства делятся на прецизионные, предназначенные для изготовления подлинников конструкторской документации, шаблонов, карт и т. д., средней точности — для контрольных прорисовок чертежей и схем и малой точности — для эскизной прорисовки в основном с экранов графических дисплеев.

По области применения различают автономные; работающие в составе больших ЭВМ и систем; работающие в составе рабочих станций и ПЭВМ.

Электромеханические графопостроители предназначены для вывода на носители (обычно на бумажные) графической и текстовой информации.

Создание и развитие ЭВМ вызвало появление регистрирующих устройств, работающих на основе цифровых данных, подготавливаемых в ЭВМ. Независимо от структуры эти устройства преобразовывают информацию из цифровой формы в графическую в виде различных документов и чертежей.



Основное преимущество графопостроителей состоит в обеспечении высокой точности черчения.

Графопостроители могут работать автономно, воспринимая исходные данные с промежуточного носителя — магнитной ленты или диска, а также непосредственно с ЭВМ, используя интерфейсы различных типов.

Графопостроители состоят из трех основных частей:

- блок механизма;
- блок управления исполнительными каналами устройства;
- система управления.

Блок механизма представляет собой планшетный или барабанный механизм, предназначенный для перемещения в плоскости чертежа пишущих элементов, их подъема и опускания.

Блок управления исполнительными каналами по взаимно-перпендикулярным направлениям (координатам  $x$  и  $y$ ) строится как по замкнутому (с использованием обратной связи), так и по разомкнутому принципу. В первом случае для привода применяются малоинерционные двигатели постоянного тока с датчиками обратной связи по положению и скорости, во втором случае — шаговые двигатели.

В последние годы широкое распространение получили графопостроители барабанного типа с перемещением носителя вдоль одной оси с помощью фрикционного вала, что позволило значительно снизить энергоемкость и материалоемкость изделия. В этом случае по сравнению с планшетными графопостроителями масса устройства снижается в 3—5 раз, хотя точность построения графопостроителей, как правило, ниже, чем у планшетных. Однако для получения чертежей в САПР точность барабанных графопостроителей вполне достаточна, а компактность предопределяет их преимущественное использование для оборудования большинства рабочих мест проектировщиков. Возможная кинематическая схема барабанного графопостроителя приведена на рис. 5.8.

Система управления предназначена:

для организации логической связи с источником информации; организации контроля состояния и диагностики устройства; подготовки исходных данных и выполнения интерполяционных процессов;

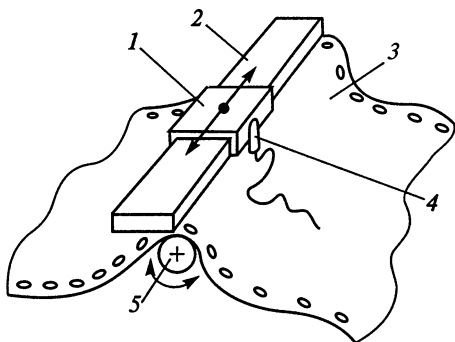
обеспечения вычерчивания символов и различных типов линий;

учета конструктивных особенностей устройств и динамических характеристик исполнительных каналов.

Процесс формирования данных для вычерчивания выполняется цифровыми интерполяторами, предназначенными для определения координат промежуточных точек, лежащих между заданными узловыми точками. Принципы построения интерполяторов во многом определяют эффективность работы графопостроите-

Рис. 5.8. Схема графопостроителя барабанного типа:

1 — каретка; 2 — направляющая пишущего узла; 3 — бумажный носитель; 4 — пишущий узел; 5 — приводной барабан



лей, их надежность и качество вычерчивания различных изображений. Цифровые интерполяторы разделяются на линейные, линейно-круговые и параболические.

В процессе вычерчивания различных чертежей для задания символов (букв, цифр и различных знаков) необходим значительный объем исходной информации по отдельным отрезкам, из которых они формируются. Вычерчивание символов по кодам существенно уменьшает объем исходной информации (практически на порядок) и упрощает процесс составления программы для графического устройства.

В настоящее время все более широкое применение в комплексах технических средств САПР находят растровые графопостроители (плоттеры). По принципу действия эти устройства аналогичны принтерам безударного действия. Наибольшее распространение получили лазерные и струйные плоттеры.

Лазерные плоттеры базируются на электрографической технологии, в основу которой положены физические процессы внутреннего фотоэффекта в светочувствительных полупроводниковых слоях селенсодержащих материалов и силовое действие электростатического поля. Селен в темноте может быть заряжен до потенциала в сотни вольт. Луч света снимает этот заряд, создавая скрытое электростатическое изображение, которое визуализируется намагниченным мелкодисперсным тонером, а затем переносится на бумагу.

В качестве промежуточного носителя в лазерных плоттерах используется вращающийся селеновый барабан. Заряженные области барабана притягивают сухой тонер, который затем переносится на проходящую под барабаном бумагу. После этого бумага с нанесенным тонером проходит через нагреватель, в результате чего частички тонера запекаются, создавая изображение.

Создание скрытого изображения на барабане осуществляется при помощи лазера. Для управления перемещением лазерного луча используется сложная система вращающихся зеркальных много-

гранников или призм и линз. Вследствие этого плоттеры (и принтеры), использующие лазеры, чувствительны к вибрациям и ударам, которые могут сбить настройку.

Избежать сложностей с оптикой позволило применение точечных полупроводниковых светодиодов (англ. light-emitting diode — LED), которые и дали имя новому типу устройств (LED-плоттеры). Общий принцип создания изображения сохранился, однако вместо зеркал используется линейка светоизлучающих диодов. LED-плоттеры относятся к классу растровых, каждой точке строки изображения соответствует свой светодиод (например, при разрешении 400 точек/дюйм линейка для формата A1 состоит из 9600 (24×400) диодов). Отказ от оптического управления сделал систему проще, легче и надежнее, так как все диоды жестко закреплены.

Лазерные и LED-плоттеры, ввиду высокого быстродействия (лист формата A1 выводится менее чем за полминуты), в первую очередь интересны пользователям с большими объемами работ. Для повышения эффективности такие плоттеры чаще всего используются как сетевые устройства. К числу их преимуществ относится то, что они могут работать на обычной бумаге, а это сокращает удельные затраты при эксплуатации.

Струйные плоттеры устроены по тому же принципу, что и струйные принтеры. Струйные плоттеры так же, как и лазерные, выводят изображения, по точности воспроизведения не уступающие лазерным, на обычные бумажные носители, но работают в несколько раз медленнее. Их важными преимуществами являются сравнительная простота получения цветных изображений и невысокая стоимость.

На сегодняшний день наибольшей известностью пользуются плоттеры фирм Hewlett Packard, ENCAD Inc., Mutoh, Ose Technologies, Mimaki и др.

Современные плоттеры имеют разрешение 150—1440 точек/дюйм, ширину печати до 2500 мм, скорость печати листа формата A1 (в зависимости от заданного разрешения) 0,5...10 мин.

**Устройства ввода графической информации.** Значительная часть конструкторской документации на изделие, как правило, представляется в графическом виде, поэтому для эффективного использования средств вычислительной техники в процессе проектирования требуется обеспечить возможность преобразования графической информации к цифровому виду, удобному для ее дальнейшей обработки в ЭВМ.

Процесс преобразования осуществляется с помощью устройств ввода графической информации (УВГИ). Условно его можно разделить на четыре этапа: поиск изображения на носителе информации; выделение элементов изображения, подлежащих кодированию; преобразование координат точек кодируемого изобра-

жения в цифровую форму; передача цифрового описания элементов изображения по каналу связи в ЭВМ для дальнейшей обработки.

В отечественной и зарубежной практике используется множество типов устройств ввода графической информации, которые могут быть классифицированы по степени автоматизации поиска и выделения элементов изображения на два больших класса: автоматические и полуавтоматические.

В полуавтоматических устройствах ввода графической информации функции поиска и выделения элементов изображения возлагаются на человека-оператора, а преобразование координат считываемых точек выполняется автоматически.

В автоматических устройствах процесс поиска и выделения элементов изображения осуществляется без участия человека. Такие устройства строятся либо по принципу сканирования всего изображения с последующей его обработкой и переводом из растровой формы представления в векторную, либо по принципу слежения за линией, обеспечивающей считывание графической информации, представленной в виде графиков, диаграмм, контурных изображений. Возможности последних устройств ограничены числом пересечений, типом линий и характером распределения элементов изображения на поле чертежа.

*Полуавтоматические устройства ввода графической информации.* В зависимости от физического принципа, положенного в основу определения местоположения указателя координат на рабочем поле устройства, полуавтоматические УВГИ можно разделить на группы: фотоэлектрические; акустические; электромагнитные; магнитострикционные и др.

С точки зрения конструктивного исполнения УВГИ можно разделить на два типа:

- электромеханические с жестким креплением указателя координат;
- устройства со свободно перемещаемым указателем координат.

Основой электромеханических устройств является механическая координатная система, представляющая собой стол с перемещающейся по одной из координат траверсой. По траверсе перемещается каретка с указателем координат. Координатная система механически связана с датчиками линейного или кругового перемещения, выполняемыми чаще всего на основе электромагнитного (линейные индуктисины, сельсины, синусно-косинусные трансформаторы) или оптико-механического (кодовые диски, «муровые линейки», интерферометрические датчики) принципа действия. Точность электромеханических устройств определяется в основном качеством изготовления механических узлов, а разрешающая способность — чувствительностью применяемых датчиков перемещения.

Электромеханические устройства отличает повышенная точность измерения координат, что позволяет использовать их в качестве поверочного (технологического) оборудования. Эти устройства, построенные, например, на основе интерферометрических линейных датчиков и прецизионных механических узлов, обеспечивают точность измерения координат до 0,01 мм. Кроме того, некоторые устройства снабжаются телевизионной передающей камерой, установленной на координатной системе, обеспечивающей 10-кратное увеличение кодируемого участка изображения.

К недостаткам устройств этого типа следует отнести необходимость ручного перемещения достаточно тяжелой механической координатной системы, что требует определенных усилий со стороны оператора и ухудшает эргономические характеристики устройства.

Развитием устройств данного типа явились полуавтоматические УВГИ с управляемым механизмом перемещения координатной системы. При этом управление перемещением координатной системой осуществляется либо за счет слежения за положением каретки, индуктивно связанной со свободно перемещаемым визиром, либо с помощью рукоятки (джойстика), управляющей перемещением координатной системы посредством шаговых двигателей (обратимый графопостроитель). Применение механического привода позволяет освободить оператора от необходимости перемещения координатной системы, но ухудшает точностные характеристики таких устройств примерно до 0,2 мм.

УВГИ со свободно перемещаемым указателем координат (планшеты) отличаются значительно большим удобством эксплуатации, но обычно имеют меньшую точность и разрешающую способность, чем электромеханические. Однако сравнительно невысокая стоимость и хорошие эксплуатационные показатели качества этих устройств привели к тому, что их доля в последние годы составила примерно 80 % общего числа выпущенных устройств ввода графической информации. Точность определения координат в этих устройствах сравнительно невысока и составляет примерно 0,3... 0,4 мм при разрешающей способности 0,125... 0,25 мм. Основное назначение этих устройств состоит в управлении перемещением маркера на экране графического дисплея и вводе смысловой информации с бланков меню, которые размещаются на рабочей поверхности планшета. Дисплейные планшеты оказались значительно более удобными в работе, чем используемые ранее рычажный (джойстик) и шаровой (трекболл) указатели перемещений, а также световое перо (в режиме указания места на экране), так как при работе с планшетом оператор производит действия, аналогичные тем, которые он привык делать при рисовании или письме на обычной бумаге обычной ручкой, и ему не требуется вырабатывать специальные навыки для работы с таким устройством.

Требования по точности и разрешающей способности, предъявляемые к дисплейным планшетам, определяются разрешающей способностью дисплеев, которая составляет для лучших образцов  $0,125 \dots 0,2$  мм, что вполне соответствует точностным характеристикам этих устройств.

Наиболее простыми с точки зрения используемых физических принципов устройствами ввода графической информации являются звуковые планшеты, определение координат в которых производится путем измерения времени распространения звуковой волны, возбуждаемой искровым разрядником, установленном на кончике указателя координат и выполненном в виде карандаша, до микрофонов, расположенных по краям планшета.

В ряде зарубежных магнитострикционных устройств используются магнитострикционные свойства материала, из которого изготовлено рабочее поле планшета. При возбуждении ультразвуковой волны внутри предварительно намагниченного магнитострикционного материала в месте прохождения фронта волны намагниченность изменяется. Изменение напряженности магнитного поля улавливается катушкой индуктивности, расположенной в указателе координат, и преобразуется в электрический сигнал, свидетельствующий о том, что фронт ультразвуковой волны находится под катушкой.

Определение координат в этих устройствах производится так же, как и в звуковых, — путем измерения времени распространения фронта ультразвуковой волны от края планшета до указателя координат.

Точностные характеристики рассматриваемых устройств относительно невысоки вследствие сильной зависимости скорости распространения волны от внешних факторов, в частности от температуры, давления, влажности, неізотропности структуры материала, движения воздушной среды в помещении.

Важное место среди устройств ввода графической информации со свободно перемещаемым указателем координат занимают устройства, называемые сеточными планшетами. По своим точностным характеристикам они занимают промежуточное положение между электромеханическими и акустическими устройствами, а некоторые образцы сеточных планшетов имеют характеристики, близкие к характеристикам лучших образцов электромеханических устройств.

Рабочее поле таких устройств представляет собой координатную сетку взаимно ортогональных изолированных проводников, закрепленных на рабочей поверхности. При протекании переменного электрического тока по проводникам координатной сетки вокруг них возникает переменное электромагнитное поле, преобразуемое кольцевым индукционным датчиком в измерительные сигналы, по которым судят о местоположении датчика по отно-

шению к координатным шинам планшета. В зависимости от того, какая из составляющих электромагнитного поля участвует в формировании измерительного сигнала, эти устройства делятся на емкостные, в которых преобразуется электрическая составляющая электромагнитного поля, и индукционные, в которых преобразуется магнитная составляющая.

В емкостных сеточных планшетах указатель координат выполнен в виде штыревого зонда, соединенного с колебательным контуром, имеющим высокое резонансное сопротивление на частоте измерительного сигнала. На практике емкостные сеточные планшеты не получили широкого распространения из-за сложности обеспечения высокой помехозащищенности (для обеспечения достоверности результатов измерения в этих устройствах используются специальные приемы кодирования, в частности код Грея). Кроме того, на этих устройствах нельзя кодировать документы, выполненные карандашом, поскольку графит как токопроводящий материал вносит погрешность в измерение электрической составляющей поля.

В индукционных сеточных планшетах преобразование магнитной составляющей поля в электрический измерительный сигнал происходит в индукционном кольцевом датчике, имеющем малое выходное сопротивление. Помехозащищенность такого датчика выше, чем у емкостного, что позволяет получить более высокую разрешающую способность устройства без использования специальных приемов кодирования сигналов, возбуждающих координатные шины планшета. Считывание информации на таких устройствах может производиться с любых немагнитных носителей, что расширяет их область применения по сравнению с емкостными сеточными планшетами.

Для устройств, соответствующих современному уровню развития техники, характерны высокие эксплуатационные показатели при сравнительно малых габаритных размерах и массе. Это достигается путем встраивания внутрь устройства одноплатных или даже однокристалльных микроЭВМ, позволяющих реализовать в устройстве несколько режимов работы, например режим поточечного (однократного) вывода координат точек, режим непрерывного считывания координат, непрерывный инкрементальный режим, при котором выдача координат происходит каждый раз по достижении любой из считываемых координат величины наперед заданного значения инкремента (0,1; 0,2; 0,3 мм и т. д.), режим выдачи координат местонахождения координатного указателя по запросу из ЭВМ, диагностический режим и др.

МикроЭВМ позволяет также осуществлять выдачу значения координат как в двоичном, так и в символическом формате в зависимости от характера решаемых задач и типа используемой аппаратуры.

В последних разработках дисплейных планшетов наряду с перечисленными чисто планшетными режимами выдачи абсолютных значений координат точек появляются режимы, имитирующие работу так называемых координатных манипуляторов типа «мышь», используемых только как средство перемещения маркера на экране графического дисплея.

Реализация новых возможностей позволила существенно расширить область использования дисплейных планшетов совместно с ПЭВМ.

Основные требования, предъявляемые к техническим параметрам, функциональным характеристикам и условиям эксплуатации полуавтоматических устройств, регламентированы ГОСТ 24593—87 «Устройства ввода графические для электронных вычислительных машин».

К перспективным направлениям, в которых ведется в настоящее время и будет проводиться в дальнейшем совершенствование полуавтоматических устройств ввода графической информации, относится уменьшение объема электронной аппаратуры за счет более широкого использования специальных заказных больших интегральных схем, сочетающих в себе функции аппаратного и программного управления устройством. В свою очередь, снижение объема электронного оборудования уменьшает массу устройств, снижает энергопотребление, повышает надежность и ремонтпригодность устройств, снижает стоимость и, как следствие, расширяет области их применения в различных сферах хозяйственной деятельности.

Одним из направлений использования кодирующего планшета связывают с управлением командами различных графических программных систем, например AutoCAD, при помощи накладных меню. Команды меню расположены в разных местах на поверхности планшета. При выборе курсором одной из команд специальный программный драйвер интерпретирует координаты указанного места, посылая соответствующую команду на выполнение.

Хотя область применения полуавтоматических устройств ввода графической информации достаточно широка, в настоящее время все большее распространение получают автоматические устройства ввода графической информации.

*Устройства автоматического ввода графической информации.* Полуавтоматические устройства ввода графической информации позволяют кодировать практически любые виды графических изображений, и в этом смысле они универсальны. Однако процесс кодирования изображений сложных, нестандартных объектов, таких, как данные аэрофотосъемки, топографические карты, данные сейсморазведки, лекальные профили, сложные, насыщенные информацией машиностроительные чертежи и другая подобная информация, на полуавтоматических устройствах отличается



высокой трудоемкостью, требует исключительного внимания и постоянного контроля со стороны оператора и поэтому становится малоэффективным. При этом ввод в ЭВМ машинописной или рукописной текстовой информации непосредственно с носителя вообще возможен только с использованием автоматических методов ввода и распознавания изображений.

Автоматические УВГИ делятся на следящие и сканирующие.

Для УВГИ следящего принципа действия характерно наличие оптико-электронного узла слежения за линией изображения. Как правило, такой узел устанавливается либо на графопостроитель, органы управления которого соединяются с блоком управления оптической головкой, либо на специально разработанную для этой цели управляемую с помощью сервопривода координатную систему с единым блоком управления, который осуществляет операции управления перемещением оптической головки по направлению линии графического изображения. Область применения таких устройств в основном ограничивается вводом в ЭВМ координат точек различных непрерывных непересекающихся кривых, например различных графиков, сейсмограмм, топографических карт, шаблонов, плазов и пр. При вводе более сложных объектов требуется постоянное вмешательство оператора, что сводит на нет преимущества следящих устройств перед полуавтоматическими.

Для УВГИ сканирующего принципа действия (сканеров) ввод графических изображений происходит практически без ограничений на характер кодируемых объектов. Считывание информации в таких устройствах осуществляется, как правило, с помощью полупроводниковой запоминающей линейки или матрицы. В зависимости от размера считываемого документа и разрешения линейки или матрицы считывание производится либо одновременно по всему документу, либо последовательно фрагмент за фрагментом. Полученное в результате сканирования растровое представление документа записывается в память машины и может быть при необходимости доработано средствами растрового графического редактирования и выведено вновь на носитель растровым печатающим устройством.

Растровое представление документа формата А1 даже при монохромном изображении и шаге 0,1 мм (чуть более грубом, чем стандартное значение 300 точек/дюйм) занимает объем около 15 Мбайт, поэтому с точки зрения уменьшения объема хранимой информации имеет смысл перевести ее в векторную форму. Операция перевода производится с помощью специального программного-аппаратного блока преобразования растр — вектор.

На первом этапе происходит выделение так называемого скелетного изображения объекта (совокупность линий толщиной в одну растровую единицу, образующих контур выделяемого объекта).

На втором этапе происходит замена этого контура совокупностью векторов. Количество векторов определяется степенью кривизны контура и заданным максимальным отклонением векторов от реального контура объекта. Совокупность полученных таким образом отрезков, последовательно записанных в память машины, позволяет в зависимости от степени насыщенности чертежа уменьшить объем хранимой информации на один-два порядка.

На следующем этапе обработки растровых изображений происходит распознавание и замена совокупностей векторов, образующих стандартные изображения, на их графические описания или коды, однозначно их характеризующие. Например, изображение алфавитно-цифровых символов распознается и заменяется стандартными кодами из таблиц КОИ-7, КОИ-8 или ДКОИ, изображения стандартных электрорадиоэлементов заменяются их идентификатором (номером) в БД, в которой хранятся параметры и графические описания объектов. Такая обработка растровых изображений позволяет не только в еще большей степени сократить объем хранимой информации, но и, что особенно важно, идентифицировать ее с информацией, получаемой от других средств ввода, таких, как дисплей и полуавтоматические УВГИ.

Необходимо отметить, что задачи распознавания изображения объектов достаточно сложны и требуют значительных вычислительных мощностей, поэтому поначалу они были реализованы на базе больших и средних вычислительных машин. При этом время распознавания составляло от нескольких минут до нескольких десятков минут на страницу текста. Задачи распознавания текстовой информации были решены с созданием аппаратно-программных средств на базе ПЭВМ.

Для идентификации изображений графических объектов вычислительные мощности современных микроЭВМ вполне достаточны. Время распознавания страницы текста составляет всего несколько секунд.

По исполнению сканеры подразделяют на ручные и настольные.

В ручных устройствах проблема перемещения считывающей головки относительно бумаги целиком ложится на пользователя. Равномерность перемещения сканера существенно сказывается на качестве вводимого в компьютер изображения, что отражается на точности считывания изображения и, таким образом, делает ручные сканеры неудобными для использования в аппаратных комплексах САПР.

Обычно настольные (планшетные, барабанные, страничные) сканеры позволяют вводить изображения формата А3 и А4. В настоящее время существуют устройства, вполне пригодные для сканирования машиностроительных чертежей. Их использование в

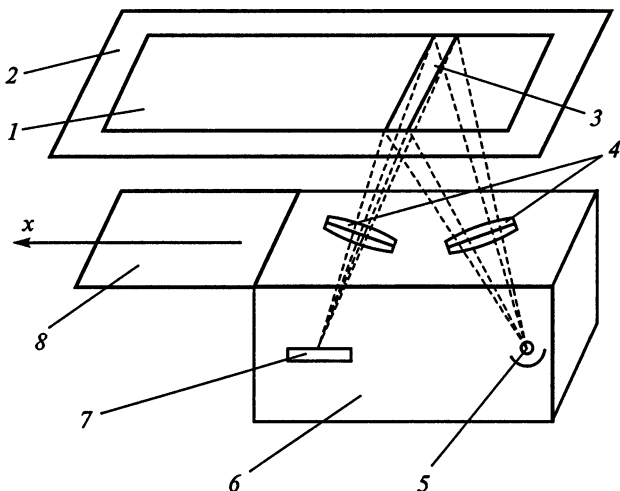


Рис. 5.9. Принцип действия автоматического кодировщика графической информации:

1 — лист с графической информацией; 2 — стеклянный планшет; 3 — полоса сканирования; 4 — оптическая система; 5 — источник света; 6 — сканирующая головка; 7 — считывающее устройство; 8 — направляющая

САПР предпочтительнее полуавтоматических кодировщиков, поскольку существует ряд программных систем для векторизации изображений.

Основным отличием планшетных сканеров является то, что их сканирующая головка перемещается относительно бумаги с помощью шагового двигателя.

Принцип действия таких устройств показан на рис. 5.9. Лист с графической информацией 1, подлежащей кодированию, обычно располагается на стеклянном планшете 2. Сканирующая головка 6 перемещается вдоль оси  $x$  по направляющей 8. Излучаемый источником света 5 поток с помощью оптической системы 4 преобразуется в отраженный от полосы сканирования 3 световой сигнал и поступает на вход считывающего устройства 7. Затем полученная информация передается по каналам связи в ЭВМ, где происходит ее необходимая обработка.

Работа барабанных сканеров напоминает работу барабанных плоттеров. Отдельные листы документов протягиваются через такое устройство, при этом и осуществляется их сканирование. Сканирующая головка остается на месте, а относительно нее перемещается бумага.

В этом случае копирование страниц книг и журналов невозможно, однако для использования в составе технических средств САПР такое устройство вполне приемлемо.

Сканеры практически не применяются непосредственно в процессе проектирования, но их использование для создания информационного фонда конструкций, выполненных в «докомпьютерную эпоху», трудно переоценить.

Преобразование документов в электронный вид делает их практически вечными и доступными широкому кругу прикладных программ. До недавнего времени такое преобразование было очень дорого, но с разработкой современных технологий процесс стал экономически оправданным.

Разрешающая способность современных сканеров находится в диапазоне 150—1200 точек/дюйм, что делает возможным кодирование графической информации практически любой сложности.

## СОВРЕМЕННЫЕ САПР АГРЕГАТОВ, УЗЛОВ И ДЕТАЛЕЙ

### 6.1. Сведения о некоторых САПР зарубежной разработки

До недавнего времени сложность решения задач автоматизированного проектирования конструкций в автомобиле- и тракторостроении, практически всегда связанных с трехмерным моделированием, требовала применения дорогостоящих систем (к ним относятся такие системы, как CADD5, NASTRAN, EUCLID, ProEngineer, UNIGRAPHICS, CATIA, CIMATRON, DUCT и др.) стоимостью в десятки тысяч долларов за одно рабочее место. На основе перечисленных программных продуктов строились САПР в основном всех ведущих мировых производителей автомобилей и тракторов. Приведем общие характеристики некоторых из этих программных систем.

**NASTRAN.** Пакет NASTRAN разработан специалистами фирмы MacNeal Schwendler Corporation (США) на базе операционной системы UNIX и представляет собой систему инженерных расчетов, основанную на методе конечных элементов.

Геометрические модели для NASTRAN можно как формировать с помощью внутреннего препроцессора системы, так и импортировать из любой другой системы автоматизированного проектирования. NASTRAN имеет прямой интерфейс с системой твердотельного параметрического моделирования SolidWorks. Кроме того, система может работать с готовыми конечно-элементными моделями, которые были созданы с помощью других систем инженерных расчетов.

Система имеет модульную архитектуру, и конкретная конфигурация может быть определена отдельно для каждого пользователя. Она обеспечивает полный набор расчетов, включая расчет напряженно-деформированного состояния, собственных частот и форм колебаний, анализ устойчивости, решение задач теплопередачи, исследование установившихся и неустойчивых процессов, акустических явлений, нелинейных статических процессов, нелинейных динамических переходных процессов, расчет критических частот и вибраций роторных машин, анализ частотных характеристик при воздействии случайных нагрузок, спектральный анализ и исследование аэроупругости. Предусмотрена возможность моделирования практически всех типов материалов,

включая композитные и гиперупругие. Расширенные функции включают технологию суперэлементов (подконструкций).

Возможности автоматического рестарта в NASTRAN позволяют проводить сложные поэтапные расчеты конструкции, соответствующие изменениям в условиях нагружения, граничных условиях в любых других параметрах конструкции, а также переходу от одного вида анализа к другому.

Наряду с расчетом конструкций NASTRAN может использоваться и для их оптимизации, которая проводится путем вариации параметров формы, размеров и свойств конструкции. Эффективные алгоритмы позволяют обрабатывать любое количество проектных параметров и ограничений. Вес, напряжения, перемещения, собственные частоты и многие другие характеристики могут рассматриваться либо в качестве целевых функций (в этом случае их можно минимизировать или максимизировать), либо в качестве ограничений. Алгоритмы анализа чувствительности позволяют исследовать влияние различных параметров на поведение целевой функции и управлять процессом поиска оптимального решения. Методы оптимизации NASTRAN дают возможность проводить согласование расчетной модели и эксперимента. Целевая функция определяется в виде минимизации отклонений результатов расчета от данных эксперимента, в качестве варьируемых параметров выбираются наименее достоверные расчетные параметры конструкции. Результатом является новая расчетная модель, полностью соответствующая физической. NASTRAN — единственная конечно-элементная программа, способная выполнять это в автоматическом режиме.

NASTRAN включает в себя уникальную функцию оптимизации конструкции, допускающую изменение геометрической топологии изделия, и используется для автоматического проектирования силовых схем конструкций, когда из исходной массивной заготовки в результате оптимизации (минимизируется вес при выполнении условий по прочности) получается оптимальная в весовом отношении конструкция.

Кроме того, NASTRAN применяется для планирования экспериментов (к примеру, определение мест расположения датчиков) и оценки полноты полученных экспериментальных данных. С помощью этого продукта решаются задачи моделирования систем управления, систем терморегулирования с учетом их воздействия на конструкцию.

Численные методы разреженных матриц, используемые при любом типе расчетов, резко увеличивают скорость вычислений и минимизируют объем необходимой дисковой памяти, что повышает эффективность обработки данных.

Система позволяет создавать (на основе библиотеки агрегатов) полные расчетные модели автомобиля, включая корпус, двига-

тель, подвеску, другие узлы и агрегаты. Данные по каждому узлу (агрегату) хранятся в текстовом формате и содержат конечно-элементные сетки деталей и узлов. NASTRAN работает на ПК, рабочих станциях и суперкомпьютерах, предусматривает возможность векторной и параллельной обработки данных на машинах, которые поддерживают эти функции.

**Комплекс EUCLID.** Система обладает наиболее полным набором функциональных возможностей по проектированию и дизайну изделий, имитации работы механизмов, подготовке управляющих программ и моделированию процессов изготовления изделий при относительно невысокой стоимости подсистем с документацией. В России наиболее популярными являются следующие подсистемы EUCLID: промышленного дизайна; проектирования узлов сложной формы с динамическим контролем процесса сборки; конструирования деталей из пластмасс и металлического листа с заданными свойствами материалов и контролем технологичности; моделирования механизмов с учетом инерционных нагрузок и характеристик приводов; проектирования пресс-форм с анализом процесса впрыска расплава пластмассы и охлаждения, литейных форм с моделированием заливки и отверждения металлов с контролем возможных дефектов; подготовки программ сложной ЧПУ-обработки.

Следует отметить, что первоначально эти подсистемы требовали применения высокопроизводительных ЭВМ, обладающих огромными ресурсами памяти и высоким быстродействием. Для большинства отечественных производителей автомобильной и тракторной техники возможность приобретения столь совершенных ЭВМ была практически исключена.

Вследствие этого применение мощных и дорогостоящих систем (класса CATIA, EUCLID, CADD5, ProEngineer, Unigraphics) было весьма проблематично. Чаще всего автоматизация проектирования базировалась на применении более простых программных продуктов, в основном обеспечивающих выпуск чертежно-конструкторской документации и имеющих некоторые возможности твердотельного моделирования. Появившиеся на рынке новейшие системы конструкторского моделирования заполняют этот вакуум и предлагают мощные решения среднего уровня в приемлемом ценовом диапазоне.

На микроЭВМ (в том числе ПЭВМ) первоначально были реализованы графические системы, обеспечивающие решение только двумерных задач. О комплексном подходе к проектированию, включающему расчетно-оптимизационные компоненты, не могло быть и речи.

**AutoCAD.** Прорыв в использовании ПЭВМ для проектирования сложных конструкций связан для большинства отечественных конструкторов с системой AutoCAD, которая первоначально

была только графической и предназначалась для ЭВМ сравнительно малой производительности.

Разработчик системы AutoCAD — фирма Autodesk уже не один год работает на рынке программных продуктов для автоматизации проектирования. Она привлекает пользователей к активному участию в своей деятельности, предлагая им высказывать свое мнение о выпущенных на рынок продуктах, тестировать пробные версии новых программ, присылать пожелания о включении новых средств в разрабатываемые приложения. Основная концепция деятельности Autodesk — максимально возможное удовлетворение потребностей потенциальных пользователей; это отчетливо просматривается в истории развития приложений для автоматизации конструкторских работ.

Уже в начальный период использования ЭВМ на рынке появилось довольно много программ, которые позволяли проводить на экране прямые линии, окружности и т. п., но большинство из них были разработаны людьми, которые очень слабо представляли себе, в чем заключается работа инженера-конструктора. Программы работали медленно, требовалось много подготовительных операций и настроек, и многие конструкторы на спор брались вычертить за доской чертеж детали быстрее, чем опытный программист за пультом «автоматической системы».

Наиболее удачной из первых систем автоматизации графических работ была система AutoCAD. Ее разработчики первыми поняли, на чем именно нужно в первую очередь сосредоточить внимание, чтобы компьютер стал действительно помощником конструктора, а не дорогой игрушкой для презентаций.

Первоначально скромные возможности, в сочетании с доступностью программного и аппаратного обеспечения, сделали AutoCAD одной из наиболее популярных графических систем в отечественном машиностроении. Именно с этой системой у многих связаны первые опыты применения ЭВМ в процессе проектирования конструкций. Возможность работы без традиционных кульмана, карандаша и ластика, вкупе с возможностью создания безбумажного архива конструкторской документации, позволили системе AutoCAD быстро завоевать симпатии многих (особенно молодых) создателей машиностроительных конструкций. В данном случае речь шла только о замене чертежного инструмента на более высокопроизводительный, остальные элементы процесса проектирования оставались прежними.

Высокая производительность проектирования с использованием AutoCAD достигалась за счет автоматизации как ряда чертежных функций, так и функций управления вычерчиванием. Отличительной особенностью AutoCAD являлось широкое использование различных средств для управления изображением. В разное время в добавление к обычной командной строке, вводимой с клавиату-



ры, и к экранному меню, существовавшему с самых ранних версий AutoCAD, были добавлены такие способы управления изображением, как:

выпадающие меню, которые активизируются при указании одного из пунктов главного меню и содержат возможность вызова абсолютного большинства команд AutoCAD;

графические меню, в которых в качестве отдельных элементов используются графические образы, наглядно представляющие результаты выбора той или иной опции (или команды), например выбор стиля штриховки;

панели инструментов, на каждой из которых помещены пиктограммы группы подобных команд (команды вычерчивания элементарных объектов, редактирования, запроса параметров объектов и т. п.).

Существуют и другие способы управления изображением, которые взаимно дополняют друг друга, и, таким образом, для каждого пользователя создается возможность использования наиболее удобной для него системы работы с изображением.

Изначально в системе AutoCAD существовала возможность вычерчивания графических примитивов с применением любых типов линий, в том числе существовала возможность создания новых типов линий, не имеющих в стандартной библиотеке системы. Важным преимуществом системы стала автоматизация штриховки, нанесение которой занимает много времени при «ручном» черчении. Возможность автоматизированного нанесения размеров также существенно сократила время создания чертежа.

Здесь необходимо отметить, что «по умолчанию» размерные линии и размерный текст наносились в соответствии со стандартами ISO, а не ЕСКД, что являлось недостатком для использования AutoCAD в отечественном машиностроении. Однако создателями системы была предусмотрена возможность путем изменения установочных величин размерных переменных приспособить систему автоматизированного нанесения размеров к требованиям других стандартов, в том числе ЕСКД. Кроме того, в AutoCAD была предусмотрена возможность создания новых шрифтов, что также позволяло создавать и использовать при проектировании шрифты ЕСКД.

Поскольку в чертежах часто встречаются повторяющиеся группы объектов, в AutoCAD для вычерчивания таких групп предусмотрено использование блоков. *Блоки* — это группы объектов, которые сохраняются под определенным именем и при необходимости вставляются в чертеж. Блок — это один объект, независимо от количества элементов, использованных для его создания, его можно легко перемещать, копировать, масштабировать, вращать. Главное преимущество использования блоков — уменьшение размеров файла чертежа, поскольку состав блока сохраняется только

один раз, а остальные экземпляры блока представлены в файле чертежа в виде ссылок на него.

С блоком можно работать так же, как с любым объектом чертежа. Во многих САПР используются библиотеки деталей, состоящие из тысяч элементов. Для создания и поддержки таких библиотек в AutoCAD существует возможность сохранения и вставки блока. Блок можно сохранить в чертеже или в отдельном файле и в дальнейшем вставлять в любой чертеж.

На основе такого подхода в машиностроении возможно создание библиотек стандартных элементов чертежа: библиотеки подшипников, крепежных деталей и т.п. Таким образом, в понятии блока системы AutoCAD в полной мере реализован структурно-символический способ представления графической информации (см. подразд. 1.5).

Очень важным достижением создателей системы AutoCAD явилась концепция послойных изображений, основанная на получении изображений в отдельных слоях, которые можно представить себе в виде прозрачных пленок.

При совмещении слоев из частичных изображений, имеющих-ся в каждом слое, можно получить законченный чертеж; это очень удобно при создании сложных чертежей машиностроительных объектов. При этом в разных слоях могут быть изображены разные детали, для каждой из которых могут существовать слои с элементами оформления (размеры, технологические требования, штриховки и т.п.). Изображения в любом слое можно сделать невидимыми, что дает возможность при выводе сборочного чертежа исключить изображения элементов оформления чертежей отдельных деталей. Таким образом, существует возможность хранения в одном файле чертежа любой сложности подробной информации обо всех его элементах.

Важной отличительной особенностью AutoCAD является возможность программирования необходимых функций адаптации AutoCAD к конкретной области применения, что реализуется с помощью языка AutoLISP.

На современном уровне использование ЭВМ и специального программного обеспечения только для создания чертежей недостаточно, необходимо создание трехмерных геометрических образов проектируемых объектов, совмещение геометрического моделирования с расчетами проектируемых конструкций и обеспечение связи проектирования объекта с технологической проработкой его будущего производства. Для многих отечественных предприятий автомобиле- и тракторостроения первые попытки решения проблем дальнейшего развития САПР на производстве связаны с системой AutoCAD, например появление возможности проектирования на ЭВМ пространственных (в том числе твердотельных) объектов.

Последние версии этой системы AutoCAD 2000, AutoCAD 2002 имеют возможности, позволяющие использовать ее для проектирования достаточно сложных технических объектов:

- возможность открыть сразу несколько чертежей и организовать обмен графической информацией между ними методом перетаскивания, вырезания и копирования. В каждом окне ведется своя независимая сессия редактирования. Специальное средство Design Center организует легкий доступ к чертежам и объектам, в них содержащимся (блокам, слоям, типам линий, формам, внешним ссылкам и т.д.) с возможностью их копирования в другие чертежи; имеется возможность частичной загрузки больших файлов;

- простота использования (имеются контекстно чувствительные всплывающие меню для всех областей экрана); новый менеджер управления слоями и новые свойства слоев: толщина линий, стиль печати, запрет вывода на печать; автотрассировка по заданным углам наклона и характерным точкам графических примитивов облегчает сложные геометрические построения; новые улучшенные средства изменения свойств объектов и образмеривания;

- новые средства управления выводом на печать (толщина линий задается напрямую через свойства слоев или графических примитивов). По одной графической модели можно создать несколько моделей пространства листа, например для компоновочного, габаритного или сборочного чертежа, содержащие необходимые виды и установки для печати. Для более удобного управления выводом на плоттер введено понятие стиля печати. Объекты можно масштабировать в единицах чертежа, процентах от текущего размера или в единицах размера шрифта;

- средства совместной работы и доступа к базам данных (редактирование блоков и внешних ссылок возможно прямо на месте вставки); доступ к чертежам через Интернет для открытия, вставки, ссылки или сохранения; организация гиперссылок между документами проекта; связывание объектов чертежа с записями в базах данных, организация запросов;

- расширенные возможности трехмерного моделирования (базируются на твердотельном ядре ACIS 4.0). Можно редактировать твердотельные объекты, динамически вращать с сохранением тонирования, в каждом видовом экране иметь свою систему координат;

- расширяемость (Visual LISP является новой интегрированной средой визуального программирования); сохраняется совместимость с AutoLISP.

Отличительной особенностью AutoCAD всегда были сравнительно невысокие требования к оборудованию. Так, для реализации версии AutoCAD 2000 достаточно возможностей персональ-

ной ЭВМ уровня Pentium 133 и выше с емкостью оперативной памяти не ниже 32 Мбайт.

В настоящее время на рынке программных продуктов имеется довольно много приложений, предназначенных для объемного моделирования изделий.

Первоначальные опыты разработчиков AutoCAD по включению в эту программу некоторых средств трехмерного моделирования не позволяли перевести ее в другой класс систем и бросить вызов признанным фаворитам в области трехмерной графики, таким, как ProEngineer и Unigraphics.

Первым шагом Autodesk в мир трехмерной графики был пакет Advanced Modeling Extension (AME), который прилагался к AutoCAD для расширения его возможностей в этом направлении. В нем уже использовались булевы операции логического конструирования, но программа не позволяла создавать поверхности сложной формы.

За этой программой последовал пакет AutoCAD Designer, в котором уже была заложена концепция параметрического конструирования и реализован более широкий набор средств моделирования объемных объектов, но для его полноценного использования требовалось подключение дополнительных программ, в частности для формирования файлов для стереолитографии в формате STL. Однако это уже был шаг в правильном направлении, который привел к появлению на рынке программы Mechanical Desktop.

**Mechanical Desktop.** Новый продукт фирмы Autodesk Mechanical Desktop является, во-первых, достаточно мощной системой трехмерного моделирования, а во-вторых, включает в свой состав весь комплекс средств двумерной машинной графики, чем совершенно напрасно пренебрегают разработчики более мощных и дорогих систем.

В настоящее время появился новый продукт Autodesk, который является продолжением той же стратегической линии, что и Mechanical Desktop, — программа Autodesk Inventor. По сравнению с Mechanical Desktop в ней усовершенствованы средства управления большим проектом, включающим сотни деталей; кроме того, эта программа позволяет импортировать файлы описания геометрических объектов, созданные в таких мощных САПР, как ProEngineer, SDRC, CATIA и SolidWorks. С появлением на рынке программы Inventor фирма Autodesk сделала новый шаг навстречу потребностям многочисленной армии конструкторов-машиностроителей.

Программа Mechanical Desktop является развитием функциональных возможностей программы AutoCAD 2000 в плане формирования среды разработки сложных машиностроительных изделий. Ее уникальной возможностью является не только способность

создавать трехмерные геометрические модели отдельных деталей, но и возможность сконструировать полноценное изделие из множества деталей и подготовить комплект соответствующих конструкторских документов.

Mechanical Desktop представляет собой полнофункциональную систему объемного моделирования и автоматизации проектирования машиностроительных изделий, которая призвана помочь конструктору на всех стадиях от концепции нового изделия к законченной конструкции. Используя функции управления иерархией деталей в сложной конструкции, эта система помогает отслеживать все изменения, вносимые в отдельные детали, и формировать согласованный комплект конструкторских документов.

Как правило, первым этапом работы над новой конструкцией является создание двумерных эскизных чертежей в среде AutoCAD. Следующая стадия — трехмерное моделирование конструкции и построение ее адекватной объемной (твердотельной) модели (выполняется в среде Mechanical Desktop).

Назначение приложения Mechanical Desktop можно сформулировать следующим образом:

- дать конструктору полноценное объемное представление о форме и функциях проектируемого изделия;
- помочь согласовать форму и размеры всех деталей собранного изделия;
- предоставить в распоряжение специалистов по маркетингу и работе с потребителями полноценную информацию о предполагаемом изделии, обеспечив им возможность продвигать его на рынок. Никакие двумерные чертежи не дают такого полного представления о проектируемом изделии, как объемное изображение, сформированное после тонирования созданной в среде Mechanical Desktop твердотельной модели изделия;
- подготовить данные для анализа качества проектируемого изделия современными инженерными методами (анализ напряжений, аэро- или гидродинамики формы и т. п.);
- предоставить исходные данные для подготовки специализированного инструмента и оснастки, необходимых для производства изделия (пресс-форм, литейных форм, программ для металлорежущих станков с ЧПУ).

С помощью приложения Mechanical Desktop форму сконструированного изделия можно оценить либо по тонированному изображению его объемной модели, либо по созданному с помощью технологии стереолитографии объемному макету изделия. В последнем случае используется создаваемый программой Mechanical Desktop файл трехмерного описания формы изделия, который передается непосредственно на технологическую установку для стереолитографии.

Встроенные в Mechanical Desktop средства анализа и оценки качества проектируемого изделия позволяют избежать затрат на изготовление реальных образцов и проведение соответствующих натуральных испытаний по крайней мере на стадии предварительного отбора вариантов конструкции. Программы анализа распределения напряжений и деформаций позволяют еще на стадии моделирования прогнозировать поведение конструкции при разных условиях, в том числе температурных, подбирать материал для изготовления.

Результаты моделирования можно преобразовать в документы разного вида:

- двумерные конструкторские чертежи деталей;
- сборочные чертежи;
- спецификации (при желании в формате листа электронной таблицы Microsoft Excel).

Таким образом, можно сформировать полный комплект конструкторской документации, необходимый для внедрения спроектированного изделия в производство.

Одно из главных преимуществ сформированных таким способом конструкторских документов — их полная совместимость с AutoCAD, что позволяет в дальнейшем организовать их поддержку и сопровождение в этой распространенной среде автоматизированного проектирования.

Если чертежи деталей и сборочных единиц связаны гиперсвязями с файлом трехмерной модели, можно на основе одного комплекта параметризированной документации создавать комплекты вариантов изделия заданного параметрического ряда.

Термин «параметрическая модель» означает модель изделия, в которой используются не жестко установленные размеры, а символические параметры. В результате на основе такой параметрической модели можно формировать параметрический ряд изделий, подставляя конкретные значения базовых параметров.

*Формирование объемной модели.* В среде Mechanical Desktop формирование объемной модели основано на создании замкнутого профиля базовой формы изделия и дальнейшего его преобразования различными способами:

*Extrusion* (Выдавливание) — исходный профиль выдавливается в направлении, перпендикулярном плоскости профиля;

*Lofted Feature* (Использование формообразующих сечений) — на исходный профиль и линии формообразующих сечений, расположенные в плоскостях, перпендикулярных плоскости базового профиля, «натягивается» упругая оболочка, а полученный объем «заливается» материалом;

*Revolve* (Построение тела вращения) — объемное тело образуется в результате «заполнения» пространства при вращении исходного профиля вокруг заданной оси;

*Face Split* (Отсечение грани) — в качестве отсекающей поверхности может использоваться либо плоскость, либо проекция какой-либо кривой на преобразуемый объект;

*Boolean* (Логическое конструирование) — метод предполагает использование функций объединения, вычитания и пересечения объектов, полученных другими способами.

После того как, используя один из этих методов или их комбинацию, создана форма изделия, можно приступить к ее доводке (снятие фасок, скругление острых ребер и т. п.).

*Сборка изделия.* Существует большой набор средств, позволяющих объединить отдельные детали в конструктивный узел, а узлы в законченное изделие. При этом возможна проверка соответствия размеров деталей, их подгонка друг к другу, формирование документации на сборку (перечня деталей и комплектующих). При сборке деталей пользователь задает ограничения на размеры каждой детали или узла. После этого программа проверяет соответствие деталей заданным ограничениям и друг другу. Если будет обнаружено какое-либо несоответствие, программа выдает предупреждающее сообщение пользователю. Вводимые пользователем ограничения на сборку влияют на степени свободы отдельных деталей и «заставляют» их подгоняться друг под друга и под требования к сборке в целом.

*Визуализация.* Набор функций визуализации дает возможность в полной мере использовать современные технологии получения изображений трехмерных объектов, близких к реальным. У заказчика или клиентов вашей фирмы при просмотре изображений, созданных в среде *Mechanical Desktop* вполне может сложиться впечатление, что рассматривается не созданное компьютером изображение модели, а фотографии реального изделия. Большой набор материалов, текстур, структур микрорельефа и прочих атрибутов, необходимых для формирования реалистичного изображения, входит в состав библиотек вспомогательных компонентов *Mechanical Desktop*. Полученное в результате изображение может быть выведено на печать или экспортировано в другие программы машинной графики, например *Corel* или *PhotoShop*. Таким образом, *Mechanical Desktop* располагает всем комплексом средств для просмотра модели с любого направления.

Следующий этап работы над проектом нового изделия — изготовление прототипа или полноразмерного макета изделия.

*Изготовление объемного макета изделия.* После завершения компьютерного проектирования и моделирования изделия во многих случаях желательно получить еще и макет изделия в натуральную величину. Макет — это воспроизведение формы изделия без воспроизведения его функций. В последнее время все большую популярность завоевывает на рынке использование для этого технологии стереолитографии. В общих чертах процесс получения макета

по этой технологии выглядит следующим образом. Под воздействием луча ультрафиолетового лазера специальное вещество, напоминающее полужидкую резину, отвердевает. Луч управляется компьютером, в результате чего формируются близко расположенные поперечные сечения сколь угодно сложной формы. Процесс этот выполняется с достаточно высокой скоростью, так что создание макета средней сложности занимает не более двух часов. Программа Mechanical Desktop способна преобразовать описание объемной модели изделия в файл формата STL, который может быть передан непосредственно на компьютер, управляющий лазером в установке для стереолитографии.

*Интегрированная система автоматизации производства.* Электронная модель изделия, разработанная с помощью Mechanical Desktop, может не только использоваться для подготовки конструкторской документации и изготовления макета изделия, но и передаваться непосредственно в систему управления изготовлением изделий (так называемые системы CAM — Computer-Aided Manufacturing). Таким образом, симбиоз Mechanical Desktop с системой CAM образует интегрированную систему CAD/CAM. В рамках этой системы созданное описание изделия преобразуется в программы управления станками с ЧПУ, сборочными и сварочными работами, которые распределяются между имеющимся на производстве оборудованием. В результате получается реальное воплощение высшей из существующих на сегодняшний день стадий автоматизации производства — гибкая производственная система изготовления изделия в целом.

Таким образом, AutoCAD — основа реализации высокоэффективных технологий в проектировании, позволяющих получать идеальный результат с меньшими затратами. Функциональные возможности AutoCAD, открытая архитектура, широкие возможности по программированию, связь с базами данных, большой выбор совместимых периферийных графических устройств, наличие сотен постоянно развивающихся прикладных программ на его базе сделали AutoCAD мировым промышленным стандартом во всех областях автоматизации проектирования.

## **6.2. Отечественные САПР, используемые в автомобиле- и тракторостроении**

Состояние отечественного рынка САПР в начале XXI в. характеризуется тем, что за последние годы промышленными предприятиями накоплен немалый опыт автоматизации локальных служб конструкторских и технологических подразделений. Несмотря на ограниченное применение средств САПР в реальной работе, результат очевиден — высокий уровень использования новых тех-



нологий, различных прикладных систем, приобретенный реальный опыт работы плюс огромное количество выполненных конструкций, моделей и т.п. Практически на каждом предприятии используются компьютерные сети, ширится применение телекоммуникационных технологий (электронная почта, Интернет). Предприятия реально подошли к интегрированным решениям, способным увязать отдельные задачи в единый комплекс. Появилось понимание необходимости массового применения компьютерных технологий для сокращения сроков разработки и реализации проекта, что диктуется жесткой конкуренцией и, вследствие этого, ограниченными сроками разработки и реализации проектов, изменившимся уровнем ответственности за результат. На первый план выходит качество сервисного обслуживания применяемых САПР и технической поддержки, что возможно лишь только при эксплуатации лицензионного программного обеспечения.

Процент использования на предприятиях лицензионного программного обеспечения неуклонно растет, а причина этого — боязнь риска, что нелегальное программное обеспечение без документации в необходимом объеме приведет к ненужным проблемам и потерям времени на устранение неполадок в системе.

Автоматизированные системы проектирования постепенно, но все же становятся обычным и привычным инструментом конструктора, технолога, расчетчика. Обеспечить конкурентоспособность проектируемого объекта в современных условиях можно только при высоком качестве проектирования в предельно сжатые сроки.

Наметилось явное изменение структуры рынка САПР. Приобретение мощных дорогостоящих систем, требующих высокого уровня подготовки персонала, не решает всех проблем конструкторских и технологических служб. Появление в последнее время новой генерации систем среднего класса типа SolidWorks, тесно интегрированными с чертежной графикой, существующими технологическими и расчетными приложениями, позволяет говорить о том, что 50...80 % задач можно решить при качественно меньших затратах. В настоящее время происходит «захват» определенной части рынка CAD/CAM, принадлежавшей ранее исключительно «тяжелым» системам, построенным на базе дорогостоящих высокопроизводительных ЭВМ, и потеснение балансирующего между легким и средним классом средств автоматизации проектирования AutoCAD программными продуктами САПР отечественной разработки.

Появление новых решений и наметившийся крен в сторону услуг заметно оживляет рынок систем CAD/CAM и увеличивает число желающих заняться их внедрением. Это неудивительно, так как зарубежная статистика показывает, что удельный вес стоимости услуг зачастую выше стоимости самого программного обеспечения. Чистая продажа САПР уже мало кого интересует. Конечной

целью потребителя является комплексное решение своих задач, а оно требует определения оптимальной спецификации программно-аппаратных средств, их увязки и интеграции друг с другом, подбор и тестирование оборудования, внедрение, обучение персонала, запуск программно-аппаратного комплекса и его техническая поддержка.

Одной из наиболее динамично развивающейся в последнее время системой является САПР КОМПАС 5.x, разработкой которой в течение ряда лет занимается российская компания «АСКОН».

Работы по проекту КОМПАС 5.x для операционной системы Windows начались в 1993 г. и основывались на существующих заделах в области конструкторской графики, инструментальных средств для разработки приложений, прикладных проблемно-ориентированных САПР, трехмерного твердотельного моделирования и программирования оборудования с ЧПУ, представленных в виде коммерчески распространяемых программных продуктов с торговой маркой КОМПАС 4.x.

Создание новой версии КОМПАС явилось естественным следствием бурных изменений, происходящих в сфере высоких технологий. Персональные компьютеры поколений 486, Pentium и Pentium Pro, операционные системы Windows различных уровней определили ключевые направления развития индустрии массового программного обеспечения. Разработки КОМПАС 5.x в полной мере учитывали эти тенденции, чтобы предоставить пользователям мощный и современный программный продукт с прекрасными перспективами развития и совершенствования.

Уже с появлением компьютеров IBM PC 386 операционная система MS-DOS превратилась в сдерживающий фактор для прикладного программного обеспечения и существующей вычислительной техники, что обусловило новый виток в развитии операционных систем. Стремительный выход на рынок Windows 95 и Windows NT 4.0, достигнутый ранее успех таких операционных сред, как Windows NT 3.5 и Windows 3.1/3.11, обеспечили подавляющее преимущество данного направления по сравнению с конкурирующей разработкой IBM OS/2 (хотя многие, возможно, придерживаются противоположной точки зрения).

Многозадачный режим работы, использование в полной мере возможностей компьютера, стандартизованный оконный интерфейс и многое другое преобразило мир программного обеспечения, определив на годы вперед основные технологические приемы пользователя любого программного продукта. Разработка каждого прикладного пакета должна проводиться в данном направлении для сохранения конкурентоспособности. КОМПАС 5.x разрабатывался специально для Windows и его отличают как более мощные функциональные возможности, так и современный унифицированный настраиваемый интерфейс.

Непрерывное повышение уровня решаемых задач является законом развития любого программного продукта. Таким уровнем, к которому естественным образом стремятся все чертежные пакеты, является поддержка трехмерного моделирования (недаром в составе всех отечественных САД-систем появились ограниченные возможности моделирования твердых тел). Реализуемые в составе КОМПАС 5.x средства трехмерного моделирования обеспечивают не усеченное, а полномасштабное моделирование поверхностей и твердых тел любой сложности на базе самых современных технологий.

Главной целью проекта КОМПАС 5.x для Windows является создание массовой интегрированной системы, относящейся к средней категории и сочетающей в себе не только высокоэффективный чертежно-графический комплекс со средствами интерактивной параметризации и широким набором параметрических библиотек и прикладных САПР, но и полномасштабные средства поверхностного и твердотельного моделирования.

Все вышеперечисленные факторы позволяют предположить, что КОМПАС 5.x сможет заполнить существующий вакуум между «тяжелыми» и «легкими» системами; это позволит использовать в качестве окружения дорогостоящих программно-аппаратных комплексов не только чертежную систему (а именно в таком качестве в основном применяется AutoCAD), но и мощные средства моделирования.

В течение 1995—1996 гг. компания «АСКОН» завершила разработку и тестирование математического аппарата интерактивной параметризации, создание комплексной модели и базового математического аппарата для твердотельного и поверхностного трехмерного моделирования. Аппарат параметризации включен как в чертежный комплекс, так и в систему трехмерного моделирования.

Впервые КОМПАС 5.x для Windows поступил в продажу в сентябре 1996 г. после тщательной доводки и скрупулезного тестирования. В его состав вошли чертежно-графический редактор, инструментальные средства разработки приложений, готовые библиотеки типовых конструктивных элементов и различные чертежные приложения.

В системе КОМПАС 5.x нового поколения сочетаются высокая эффективность, удобство и интуитивная понятность пользовательского интерфейса. Совершенные инструменты для конструкторского проектирования удовлетворяют самым высоким современным требованиям. По своим техническим характеристикам чертежно-конструкторский модуль КОМПАС-ГРАФИК 5.x на сегодняшний момент является одним из самых мощных в мире.

**Состав и возможности КОМПАС 5.x.** Система КОМПАС 5.x представляет собой современный программный продукт, функционирующий под управлением операционной системы Windows раз-

личных уровней, имеет настраиваемый оконный интерфейс, соответствующий стандартам Windows, и управляется с помощью команд текстового меню, панелей кнопок, контекстно-зависимых (динамических) меню. Оформление экрана, состав кнопочных панелей и любые параметры системы могут быть настроены непосредственно во время сеанса работы. Пользователь может формировать собственные кнопочные панели, в том числе подключая функции прикладных библиотек в качестве команд. Поддерживается одновременная работа с несколькими документами, а также отображение каждого документа в нескольких окнах; реализованы различные режимы резервного копирования загруженных документов.

Печать разработанных документов может выполняться на любых устройствах (принтерах или плоттерах), поддерживаемых Windows. Реалистичное изображение документов в режиме предварительного просмотра позволяет скомпоновать на поле вывода и распечатать одновременно несколько документов; обеспечена гибкая настройка всех параметров печати. В состав системы входит утилита для разработки собственных драйверов перьевого устройств вывода.

Для пользователей, работающих над проектами вместе со специалистами других предприятий, полезной окажется бесплатная утилита просмотра КОМПАС-Viewer, с помощью которой можно ознакомиться с документами и распечатать их даже на тех компьютерах, где не установлен КОМПАС 5.x.

Система КОМПАС 5.x поддерживает технологию OLE (Object Linking and Embedding — компоновка и внедрение объектов), что позволяет вставить документы КОМПАС в любой документ, являющийся OLE-контейнером (например, в документ Microsoft Word). Созданный таким образом OLE-объект в дальнейшем можно просматривать при помощи КОМПАС-Viewer или редактировать средствами КОМПАС. Если при вставке OLE-объекта сохранена связь с источником, все вносимые в источник изменения будут отражаться в документе-контейнере.

Наличие подробного Руководства пользователя, Практического руководства и обширной Справки облегчает и ускоряет освоение системы КОМПАС.

Основными компонентами КОМПАС 5.x являются КОМПАС-ГРАФИК — редактор конструкторской документации и КОМПАС-3D — система трехмерного твердотельного моделирования.

Минимальные требования к конфигурации компьютера для установки КОМПАС 5.x: IBM PC 486DX2-66, 16 Мбайт RAM, Windows 95, не менее 100 Мбайт свободного места на жестком диске. Рекомендуемая конфигурация для высокопроизводительной работы: Pentium 100 и выше, 32 Мбайт RAM, монитор 17 дюймов и более, Windows 95/98 или Windows NT 4.0.

**Система КОМПАС-ГРАФИК.** Данная система предназначена для автоматизации проектно-конструкторских работ в различных отраслях деятельности. Она успешно используется в машиностроении, архитектуре, строительстве, составлении планов и схем — везде, где необходимо разрабатывать и выпускать графические и текстовые документы.

*Графический редактор КОМПАС-ГРАФИК* позволяет разрабатывать и выпускать различные документы (эскизы, чертежи, схемы, плакаты и т. д.). В системе предусмотрены два вида графических документов — чертежи и фрагменты. Чертеж имеет рамку и основную надпись, в нем можно создавать до 255 видов (проекций, разрезов, сечений) с разным масштабом изображения. На листе чертежа могут быть размещены спецификация, технические требования, знак неуказанной шероховатости. Фрагмент содержит изображение в натуральную величину без элементов оформления (рамки, технических требований и т. п.). Любой вид чертежа или фрагмент может содержать до 255 слоев, каждый из которых можно делать текущим, или недоступным для редактирования, или невидимым.

Редактор КОМПАС-ГРАФИК позволяет работать со всеми типами графических примитивов, необходимыми для выполнения любого построения; к ним относятся точки, прямые, отрезки, окружности, эллипсы, дуги окружностей и эллипсов, многоугольники, ломаные линии, кривые NURBS (в том числе кривые Безье). Разнообразные способы и режимы построения этих примитивов (например, команды создания фасок, скруглений, эквидистант, построения отрезков и окружностей, касательных к объектам, и т. п.) избавляют пользователя от необходимости производить сложные вспомогательные построения. Для ускорения построений можно использовать локальные системы координат, разномасштабную сетку и механизм объектных привязок.

Одной из самых сильных сторон КОМПАС-ГРАФИК по-прежнему является полное соответствие ЕСКД; поддерживаются стандартные (соответствующие ЕСКД) и пользовательские стили линий и штриховок; реализованы все типы линейных, угловых, радиальных и диаметральных размеров (включая наклонные размеры, размеры высоты и размеры дуги); автоматически выполняются простановка допусков и подбор качества по заданным предельным отклонениям; среди объектов оформления — все типы шероховатостей, линий-выносок, обозначения баз, допусков формы и расположения поверхностей, линии разреза и сечения, стрелки направления взгляда, штриховки, тексты, таблицы.

В графический документ КОМПАС-ГРАФИК может быть вставлено растровое изображение формата BMP, PCX, DCX, JPEG, TIFF. При вставке растрового объекта возможно задание его масштаба и угла поворота.

Редактор КОМПАС-ГРАФИК обеспечивает пользователя всеми инструментами, необходимыми для редактирования чертежа. Выполняются операции сдвига, копирования, поворота, масштабирования, симметричного отображения, деформации, удаления, выравнивания; поддерживается перенос и копирование объектов через буфер обмена; перетаскивание мышью характерных точек любых (как векторных, так и растровых) объектов позволяет быстро менять их размер и положение.

Возможно создание макроэлементов и именованных групп объектов. При формировании и изменении чертежа можно использовать ссылки на связанные с ним внешние фрагменты, которые могут храниться как в отдельных файлах, так и в специальных библиотеках фрагментов.

Любому графическому объекту можно поставить в соответствие неграфическую информацию, называемую атрибутом. Атрибутом может быть число, строка, запись или таблица; объект может иметь любое количество атрибутов. Атрибуты объекта могут быть просмотрены и отредактированы в любой момент работы над документом; они используются и для поиска графических объектов.

Система содержит большой набор команд для измерения длин, расстояний, углов в графическом документе и вычисления массы и инерционных характеристик плоских фигур, тел выдавливания и вращения.

Режим реалистичного заполнения граф основной надписи и текста технических требований облегчает оформление документа. В комплект поставки КОМПАС-ГРАФИК входит библиотека стандартных основных надписей графических документов; возможно создание пользовательских основных надписей.

В графическом редакторе КОМПАС-ГРАФИК могут создаваться параметрические модели. Отличие параметрической модели от обычной состоит в том, что в ней существуют взаимосвязи между объектами.

Примерами взаимосвязей могут служить параллельность, перпендикулярность, симметрия, равенство радиусов, касание объектов, совпадение их характерных точек и т. п. Взаимосвязи формируются как при вводе объектов (автоматически), так и путем вызова специальных команд. Автоматическое формирование связей может быть запрещено, а любая существующая связь может быть удалена. Возможно также создание ассоциативных объектов оформления (размеров, штриховок, обозначений шероховатости и т. д.). Ассоциативные объекты «отслеживают» изменение положения своих базовых примитивов и автоматически перестраиваются в соответствии с ним.

Параметрам графических объектов (например, длинам, углам, радиусам) могут быть поставлены в соответствие буквенные переменные. Возможно задание аналитических зависимостей (урав-

нений и неравенств) между этими переменными и, следовательно, между параметрами объектов. В результате редактирования любого параметрического объекта остальные объекты перестраиваются так, чтобы заданные пользователем взаимосвязи не нарушались. Благодаря этому свойству параметрической модели она идеально подходит для создания однотипных изображений, различающихся параметрами элементов.

*Текстовый редактор КОМПАС-ГРАФИК* позволяет выпускать различные текстовые документы (расчетно-пояснительные записки, технические условия, инструкции и т.д.); текстовый документ является отдельным типом документа КОМПАС.

При работе с текстовым документом доступны все основные возможности, являющиеся стандартом де-факто для современных текстовых редакторов: работа с растровыми и векторными шрифтами Windows (в том числе в формате UNICODE); выбор параметров шрифта (размер, наклон, начертание, цвет и т.д.); выбор параметров абзаца (отступы, межстрочный интервал, выравнивание и т.д.); ввод специальных знаков и символов, надстрочных и подстрочных символов, индексов, дробей; вставка рисунков (графических файлов КОМПАС); автоматическая нумерация списков (в том числе с различными уровнями вложенности) и страниц; поиск и замена текста; формирование таблиц. Возможно создание стилей текста и стилей оформления текстового документа и быстрое форматирование документа с использованием этих стилей. Часто встречающиеся фрагменты текста могут быть сохранены для последующего быстрого ввода. Предусмотрена возможность автоматической замены ошибочно введенных латинских символов на кириллические и наоборот.

Все функции текстового редактора КОМПАС-ГРАФИК доступны не только при создании отдельных текстовых документов, но и при вводе любого текста в графическом документе (при создании технических требований, таблиц, технологических обозначений всех графических объектов, содержащих текст).

*Модуль проектирования спецификаций КОМПАС-ГРАФИК* позволяет выпускать разнообразные спецификации, ведомости и другие табличные документы. Спецификация является отдельным типом документа КОМПАС-ГРАФИК. Многие функциональные возможности модуля разработки спецификаций КОМПАС-ГРАФИК заимствованы из логики и технологии разработки «бумажных» спецификаций. При заполнении документа на экране пользователь видит стандартную таблицу спецификации и может вводить данные в ее графы.

В конструкторской практике спецификация, составляемая на изделие, всегда соответствует сборочному чертежу этого изделия. Спецификация КОМПАС-ГРАФИК также может быть связана со сборочным чертежом (одним или несколькими его листами) и

другими электронными документами; эта связь является двунаправленной и ассоциативной. Находясь в окне спецификации, можно быстро открыть подключенные к ней чертежи и, наоборот, при работе с чертежом можно вызвать подключенную к нему спецификацию. Возможна передача данных из чертежа в спецификацию или из спецификации в чертеж; из спецификации в чертеж передаются номера позиций компонентов сборки (стандартных изделий, деталей и т. д.); из сборочного чертежа в спецификацию передаются номера зон, в которых расположено изображение соответствующих компонентов сборки. Из чертежей деталей и сборочных единиц в спецификацию передаются наименование, обозначение, масса, материал изделия, формат его чертежа и другие данные. Если в сборочный чертеж вставлены изображения стандартных элементов из Конструкторской библиотеки КОМПАС-ГРАФИК, информация о них передается в спецификацию.

Спецификация может содержать сведения, дополняющие информацию, включаемую в стандартный бланк. Эти сведения хранятся в так называемых «дополнительных колонках», они могут быть просмотрены или отредактированы в любой момент, однако в бланке спецификации они не видны и на печать не выводятся. Примером информации в дополнительных колонках могут служить масса и стоимость объекта. Сервисные команды позволяют сложить числовые значения дополнительных параметров. При этом может учитываться количество одинаковых объектов в сборке и даже количество объектов в ее различных исполнениях (таким способом можно подсчитать массу или стоимость специфицируемого изделия). В дополнительные колонки вводят и любую другую информацию об объекте (код ОКУД, материал, текстовый комментарий и т. д.); количество и состав колонок определяются потребностями пользователя.

Строки спецификации могут быть связаны с графическими объектами в сборочном чертеже. При наличии таких связей в спецификации можно включить режим, в котором система автоматически выделяет в чертеже геометрию, относящуюся к выделенной строке спецификации. Спецификацию можно настроить таким образом, чтобы при удалении ее строки происходило и автоматическое удаление соответствующей геометрии из сборочного чертежа.

Если в момент создания спецификации рабочие чертежи деталей и узлов еще не готовы, графические объекты, подключенные к строке спецификации, можно передать в новый графический документ, получив таким образом заготовку чертежа. В его основную надпись будут автоматически переданы обозначение и наименование, присвоенные изделию в спецификации.

Модуль проектирования спецификаций КОМПАС-ГРАФИК поддерживает заполнение разделов и предусмотренную стандартом



сортировку строк внутри них. Замечательной особенностью спецификации КОМПАС-ГРАФИК является возможность создавать и заполнять разделы в произвольной последовательности. Пользователь может сначала ввести стандартные изделия, затем создать и заполнить раздел Документация, перейти к вводу деталей, а потом — сборочных единиц. Система автоматически расположит получившиеся разделы в стандартной последовательности; каждый новый раздел будет размещаться в строго определенном по отношению к существующим разделам месте, раздвигая при необходимости уже заполненные строки.

Зачастую стандарт предписывает группировать объекты в разделе по видам, а внутри этих групп сортировать по наименованию или обозначению; и этот механизм поддерживает спецификация КОМПАС-ГРАФИК: при вводе нового объекта можно указать не только раздел, но и подраздел для его размещения.

Строки спецификации можно заполнять в произвольном порядке, выбирая разделы и подразделы, к которым они относятся. Спецификация автоматически располагает строки в предписанной стандартом последовательности (например, детали сортируются по возрастанию их буквенно-числового обозначения). Пользователь может самостоятельно выбрать колонку, по которой должна производиться сортировка, и указать одно из правил сортировки — по возрастанию числового значения, по убыванию числового значения, по алфавиту, смешанную (и по алфавиту, и по значению числа в порядке следования символов текста) и для раздела документации (при таком типе сортировки коды документов располагаются не по алфавиту, а в порядке, предписанном ГОСТ 2.102—68\*). Возможно также формирование более сложных правил сортировки (например, вначале по номеру стандарта, затем по названию изделия, потом по его типоразмеру). Такие правила используются, например, при сортировке раздела «Стандартные изделия».

После редактирования обозначения детали, документа или другой строки спецификации эта строка меняет свое положение в спецификации в соответствии с правилами сортировки. После любого изменения порядка строк в спецификации нарушается нумерация позиций, что можно устранить автоматически (команда простановки позиций присваивает строкам спецификации номера в порядке их следования, учитывая при этом наличие резервных строк в разделах).

Совокупность параметров и настроек объединяется в стиль спецификации. В поставку входят стили, позволяющие оформлять документы по ГОСТ 2.108—68 и ГОСТ 2.113—75\*. В них уже включены стандартные разделы, заданы правила заполнения колонок в каждом из них, настроена сортировка объектов и т. д. Возможно создание пользовательских стилей спецификации, объединяющих

разнообразные настройки и параметры (вид бланка, названия и оформление разделов, правила сортировки и т.д.).

Разнообразие параметров и настроек, особенно возможность использовать пользовательский бланк, позволяет создавать не только спецификации в соответствии со стандартом. Механизмы модуля разработки спецификаций подходят для работы с различными ведомостями, перечнями, каталогами и списками: их строки можно нумеровать, сортировать, связывать с документами и графическими объектами и т.д. Комбинируя различные настройки спецификации, можно создавать ведомости спецификаций, ведомости ссылочных документов, ведомости покупных изделий, таблицы соединений, листы регистрации изменений и пр.

*Обмен информацией с другими системами КОМПАС 5.x* предоставляет пользователю благодаря различным конверторам. Функции импорта данных из большинства форматов предоставляются пользователям системы КОМПАС 5.x бесплатно; к ним относятся:

- чтение графических файлов форматов DXF, DWG и IGES;
- чтение текстовых файлов форматов ASCII (DOS), ANSI (Windows) и RTF;
- чтение файлов трехмерных моделей форматов IGES и SAT;
- чтение файлов документов КОМПАС версии 4.

Некоторые конверторы для экспорта и импорта данных являются отдельно оплачиваемыми компонентами системы. Используя их, можно выполнить:

- запись графических файлов форматов DXF и IGES;
- запись файлов трехмерных моделей форматов IGES, SAT и STL;
- запись данных спецификации в форматы DBF и Microsoft Excel;
- чтение файлов формата PDF (P-CAD 4.5).

*Средства разработки приложений КОМПАС-МАСТЕР* — это набор динамически подключаемых библиотек (DLL — Dynamic Link Library), которые можно использовать из любой стандартной системы программирования для Windows на языках C, C++. КОМПАС 5.x является открытой системой, что позволяет создавать дополнительные программные модули (пользовательские библиотеки), а затем применять их во время работы над документами. Таким образом, стандартные возможности чертежно-графического редактора и редактора деталей могут быть дополнены исходя из тех специальных задач, которые приходится решать пользователю. В библиотеке включено свыше 300 специальных функций для доступа к ядру КОМПАС. Функции обеспечивают построение и обработку документов, работу с геометрической моделью, организацию интерфейса между прикладной библиотекой и основной системой. Для доступа к данным различных стандартных СУБД можно использовать интерфейс ODBC (Open DataBase Connectivity). Типичный пример приложения — библиотека стандартных машиностроительных элементов.

*Машиностроительная библиотека* включает в себя более 200 параметрических изображений различных типовых машиностроительных элементов (болты, винты, гайки, заклепки и другой крепеж, подшипники, профили, конструктивные места, элементы соединения трубопроводов, манжеты и т. д.). Выбор параметров из стандартного ряда значительно упрощает простановку элемента на чертеже и практически исключает ошибки конструктора. В дальнейшем внесенный в чертеж элемент хранится как единое целое, и конструктор может легко отредактировать его, дважды щелкнув мышью на изображении. При простановке стандартного элемента в чертеж вносится дополнительная информация, необходимая для последующего построения спецификации. Библиотека существенно сокращает затраты времени конструктора при разработке сборочных и детализовочных машиностроительных чертежей и обеспечивает высокое качество документации.

*Интегрированная система проектирования тел вращения КОМПАС-SHAFT Plus* предназначена для параметрического проектирования деталей типа тела вращения (вал, втулка, цилиндрические и конические шестерни, червячные колеса и червяки, шкивы ременных передач); обеспечивается построение шлицевых, резьбовых и шпоночных участков на ступенях валов. Сложность моделей валов не ограничена, количество ступеней — любое.

Параметрические модели валов сохраняются непосредственно в чертеже и доступны для последующего редактирования средствами системы КОМПАС-SHAFT Plus. При создании и редактировании может быть изменен порядок ступеней вала (методом простого перетаскивания мышью Drag&Drop), изменен любой параметр ступени либо выполнено ее удаление. Система включает в себя модуль расчетов механических передач КОМПАС-GEARS 5 (геометрические и прочностные расчеты цилиндрических и конических зубчатых, цепных, червячных, ременных передач).

По результатам расчетов, помимо формирования контура ступени, могут быть автоматически сформированы таблицы параметров и выносные элементы с профилями зубьев. При изменении расчетных параметров передач они также автоматически корректируются.

Интуитивно понятный интерфейс и развитая система Помощи обеспечивают быстрое освоение системы. Система КОМПАС-SHAFT Plus позволяет в десятки раз увеличить скорость проектирования деталей класса тел вращения и выпуска конструкторской документации на них.

*Модуль КОМПАС-SPRING* обеспечивает выполнение проектного или проверочного расчетов цилиндрической винтовой пружины растяжения или сжатия с одновременным автоматическим формированием чертежа на пружину. Расчет выполняется при минимальном количестве исходных данных и гарантирует получение необходи-

мых конструктору параметров пружины при ее минимальной массе. В ходе расчета конструктор может варьировать различными параметрами пружины для получения наилучшего результата. Как показывает практика пользователей, КОМПАС-SPRING позволяет в 15—20 раз повысить скорость проектирования витых пружин и выпуска документации на них.

*Библиотека элементов гидравлических и пневматических схем* рекомендуется для использования при разработке различных технических схем средствами КОМПАС-ГРАФИК 5. В библиотеку включено большое количество типовых изображений блоков, вентилях, гидрозамков, дросселей, емкостей, клапанов давления, кондиционеров, насосов, обратных клапанов, распределителей, средств измерений, цилиндров и прочих стандартизованных элементов.

Кроме перечисленных, в состав системы входят десятки различных библиотек, в частности:

- библиотека элементов электрических схем;
- библиотека элементов станочных приспособлений;
- библиотека элементов кинематических схем;
- архитектурно-строительная библиотека;
- система проектирования металлоконструкций;
- пакет библиотек «Элементы инженерных коммуникаций»;
- пакет библиотек «Элементы химических производств»;
- пакет библиотек «Электроснабжение»;
- пакет библиотек «Автоматизация технологических процессов»;
- пакет библиотек «Коммутационные устройства»;
- библиотека трубопроводной арматуры;
- библиотека фрагментов «Строительные конструкции. Профили» и др.

Хотелось бы особо отметить комплексы справочных данных, предназначенных для использования в машиностроении.

*Справочник конструкционных материалов для КОМПАС-ГРАФИК* предоставляет конструктору и технологу следующую информацию:

- базы данных (обозначения и документы на поставку) по черным и цветным металлам, сплавам, неметаллическим материалам (более 200 наименований);
- физико-механические и технологические свойства конструкционных материалов, их назначение и области применения;
- более 40 видов сортамента (фасонного, листового, профильного и др.), изготавливаемого из этих материалов, включая перечни типоразмеров, выпускаемых промышленностью.

Библиотека имеет широкий набор сервисных функций, включающий в себя:

поиск материала по нескольким критериям (по назначению, физико-механическим свойствам, марке);

добавление новых марок материалов, а также редактирование имеющихся (можно добавлять физико-механические свойства для других состояний материала; например, если в базе уже есть характеристики материала в состоянии поставки, пользователь может ввести такие параметры для этого же материала после термообработки);

формирование обозначений как непосредственно материалов (включая документы на химический состав), так и сортов (в соответствии с ЕСКД и российскими стандартами (сформированная запись обозначения может быть вставлена в соответствующую графу штампа чертежа, в спецификацию, в ведомость материалов или другие конструкторско-технологические документы);

контроль применимости материалов, их сортов и типоразмеров на конкретном предприятии (установка применимости выполняется администратором системы управления производством или службой материально-технического снабжения).

Библиотека интегрирована с системой КОМПАС-ГРАФИК 5. Параметры и обозначения материалов и сортов могут быть использованы в других проектно-конструкторских системах, таких, как SolidWorks, в системах управления проектами и документооборотом (например, КОМПАС-МЕНЕДЖЕР), в системах проектирования технологических проектов (например, АВТОПРОЕКТ). Требования к аппаратным средствам не отличаются от аналогичных требований для работы системы КОМПАС-ГРАФИК.

*Электронный справочник по подшипникам качения* предназначен для инженерно-технических работников всех отраслей промышленности.

При его разработке использованы государственные стандарты по подшипникам качения, данные монографий, общепринятых справочников и фирменных (заводских) каталогов. По согласованию с заказчиками в него могут быть введены недостающие сведения и данные.

Справочник включает в себя:

данные по стандартным подшипникам (информация по более чем 5000 типоразмерам стандартных подшипников 100 наиболее распространенных типоразмеров);

характеристики и рекомендации о применении свыше 100 наиболее распространенных типоразмеров стандартных подшипников;

возможность идентификации российских подшипников по номеру знаков в соответствии с ГОСТ 3189—89;

возможность расчета реакций опор двухопорных валов, нагруженных в двух плоскостях в пяти сечениях радиальными силами и (или) сосредоточенными изгибающими моментами;

информацию по посадкам, точности и шероховатости сопряженных с подшипниками деталей в соответствии с ГОСТ 3325—85.

При назначении посадок колец дополнительно учитывается вид нагружения колец. Предусмотрена ручная корректировка полей допусков посадочных поверхностей вала и корпуса;

словарь, в который включены более 500 понятий, сведений с необходимыми иллюстрациями, групп данных, список литературы (около 200 наименований), перечень ГОСТов и соответствующих им стандартов ISO, перечень фирм — производителей подшипников с характеристикой их продукции, а также основные сведения по уплотнениям и смазочным материалам.

Отличительной особенностью КОМПАС 5.x является исключительно развитая система трехмерного твердотельного моделирования, включающая в себя как возможности построения твердотельных образов отдельных деталей, так и составление сборок из них.

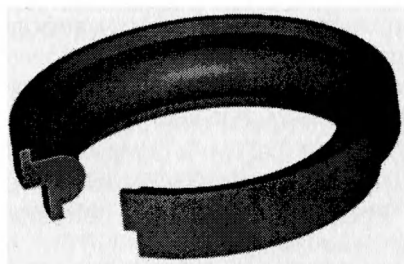
**Система КОМПАС-3D.** Данная система предназначена для создания трехмерных параметрических моделей деталей и сборочных единиц, содержащих как типичные, так и нестандартные, уникальные конструктивные элементы. Параметрическая технология позволяет быстро получать модели типовых изделий на основе однажды спроектированного прототипа.

Ключевой особенностью системы КОМПАС-3D является использование собственного математического ядра и параметрических технологий, разработанных специалистами фирмы «АСКОН».

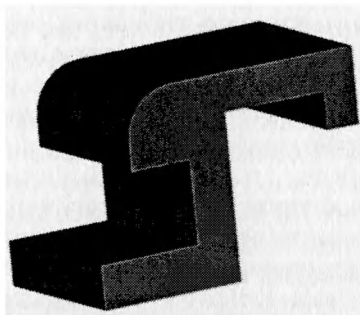
Основные задачи, решаемые системой:

- моделирование изделий с целью создания конструкторской и технологической документации, необходимой для их выпуска (сборочные чертежи, спецификации, деталировки и т.д.);
- моделирование изделий с целью расчета их геометрических и инерционных характеристик;
- моделирование изделий для передачи геометрических параметров в расчетные пакеты;
- моделирование деталей для передачи геометрических параметров в пакеты разработки управляющих программ для оборудования с ЧПУ;
- создание изометрических изображений изделий (например, для составления каталогов, создания иллюстраций к технической документации и т.д.).

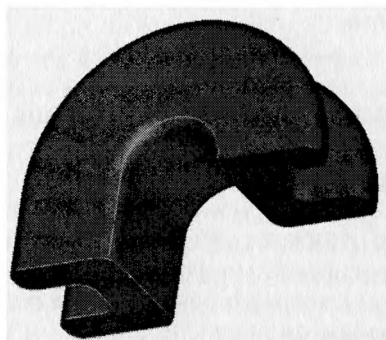
*Детали.* Модель детали в КОМПАС-3D создается путем выполнения булевых операций над объемными элементами, которые образуются путем заданного пользователем перемещения плоской фигуры (эскиза) в пространстве. Эскиз изображается на плоскости и создается стандартными средствами чертежно-графического редактора КОМПАС-ГРАФИК. В него можно перенести изображение из ранее подготовленного графического документа, что позволяет при создании трехмерной модели опираться на существующую чертежно-конструкторскую документацию. При этом



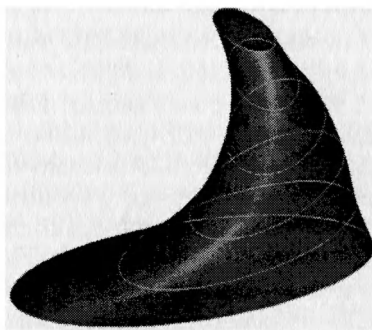
*a*



*б*



*в*



*г*

Рис. 6.1. Способы построения объемных моделей в КОМПАС-3D:  
*a* — вращением; *б* — выдавливанием; *в* — перемещением вдоль направляющей;  
*г* — по нескольким сечениям

доступны следующие типы операций, примеры которых показаны на рис. 6.1:

- вращение эскиза вокруг оси, лежащей в плоскости эскиза (рис. 6.1, *a*);
- выдавливание эскиза в направлении, перпендикулярном плоскости эскиза (рис. 6.1, *б*);
- кинематическая операция — перемещение эскиза вдоль указанной направляющей (рис. 6.1, *в*);
- построение тела по нескольким сечениям-эскизам (рис. 6.1, *г*).

Нетрудно заметить, что способы вращения и выдавливания являются частными случаями кинематической операции, при которых в качестве направляющей используются дуга окружности и отрезок прямой линии соответственно. Тем не менее выделение способов вращения и выдавливания в самостоятельные способы представляется оправданным, поскольку в большинстве

случаев проектирования деталей машиностроения их возможно-стей достаточно, а простота использования делает их наиболее удобными.

Дополнительные операции упрощают задание параметров пространственных конструктивных элементов (фасок, скруглений, круглых отверстий, литейных уклонов, ребер жесткости). На любом этапе работы можно сформировать тонкостенную оболочку, удалить часть тела по границе, представляющей собой плоскость или криволинейную поверхность.

В КОМПАС-3D доступны разнообразные способы копирования элементов: копирование по сетке, по окружности, вдоль кривой, «зеркальное» копирование, а также создание «зеркальных» деталей.

Кроме твердотельных объектов, в КОМПАС-3D могут быть построены пространственные кривые:

- цилиндрические спирали;
- конические спирали;
- ломаные по точкам и координатам (в том числе с заданием радиусов скруглений в углах);
- сплайны по точкам и координатам.

Эти объекты могут использоваться, например, в качестве направляющих при моделировании пружин, резьб и подобных объектов. Поверхности, импортированные из форматов IGES и SAT, могут использоваться для отсечения части модели или в качестве объекта, до которого производится выдавливание.

Если существующих в модели ортогональных плоскостей, граней и ребер недостаточно для выполнения построений, пользователь может создавать вспомогательные плоскости, оси и пространственные кривые, задавая их положение различными способами. Применение вспомогательных конструктивных элементов значительно расширяет возможности построения модели.

После создания основания детали одним из указанных методов можно приклеивать к нему или вырезать из него следующие формообразующие элементы. Они, как и основание, могут представлять собой элементы:

- выдавливания;
- вращения;
- кинематические;
- по сечениям.

Основные правила построения этих элементов аналогичны правилам построения оснований соответствующей формы.

При вводе параметров операции вырезания или приклеивания доступно несколько больше опций, чем при построении основания. Дополнительные опции позволяют упростить задание параметров элементов, а также связать их друг с другом. Например, при создании сквозного отверстия можно не рассчитывать его длину, а указать, что оно должно быть построено через всю деталь;



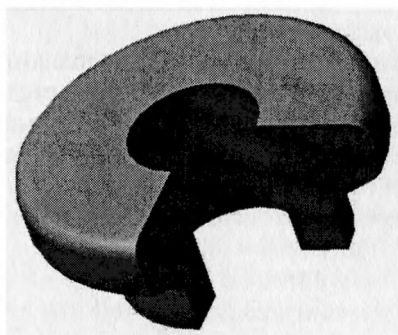
при создании бобышки указать, что она должна быть построена до определенной поверхности.

Приклеивание или вырезание формообразующего элемента начинается с создания его эскиза. Примеры операций приклеивания и вырезания выдавливанием приведены на рис. 6.2.

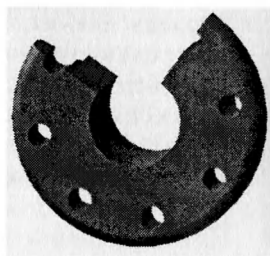
На рис. 6.2, *а* показана базовая деталь, полученная способом вращения. Вращение эскиза проведено вдоль дуги окружности величиной  $270^\circ$ . На рис. 6.2, *б* показана та же деталь с отверстиями, вырезанными выдавливанием. На рис. 6.2, *в* к той же детали приклеены выдавливанием бобышки, причем они выполнены коническими (система позволяет вырезать и выдавливать элементы с учетом уклонов), а их наружные поверхности скруглены.

В системе КОМПАС-3D предусмотрено несколько возможностей визуализации трехмерных объектов: каркас, отображение без невидимых линий или с тонкими невидимыми линиями и полупрозрачное отображение.

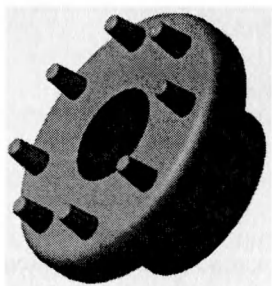
Последняя опция позволяет представить проектируемую деталь практически с фотографической точностью, причем предусмотрена возможность изменения цвета детали, что обеспечивает, с



*а*



*б*



*в*

Рис. 6.2. Примеры операций приклеивания и вырезания выдавливанием:

*а* — базовая деталь; *б* — та же деталь с отверстиями, полученными выдавливанием; *в* — та же деталь с коническими бобышками, приклеенными выдавливанием

одной стороны, удобство при создании сборок, а с другой — возможность имитации цвета материала, из которого изготовлена деталь.

Приведенные на рис. 6.2 детали выполнены с помощью полутонового изображения.

Важнейшей возможностью с точки зрения использования системы твердотельного моделирования является создание сборок из пространственных элементов.

Модель сборки в системе КОМПАС-3D состоит из набора компонентов — деталей, подборок и стандартных изделий (рис. 6.3). На рис. 6.3, *а* показан редуктор в сборе, а на рис. 6.3, *б* — этот же редуктор без корпуса и вала.

Существующий компонент (например, модель детали, хранящаяся в файле) может быть многократно вставлен в сборку. Вставка производится путем выбора файла компонента. Возможно также перетаскивание компонента мышью из «дерева» построения или из окна редактирования этого компонента.

Компонент сборки может создаваться и непосредственно в сборке «на месте». В этом случае при построении компонента используется его окружение. Например, эскиз основания новой детали создается на грани существующей детали и повторяет ее контур, а траекторией этого эскиза при выполнении кинематической операции становится ребро другой детали.

Модель любого компонента сборки (в том числе созданного «на месте») хранится в отдельном файле на диске. В модели сборки находятся лишь ссылки на файлы всех компонентов и информация об их положении в сборке. Благодаря этому становятся доступны:

быстрая замена компонентов;

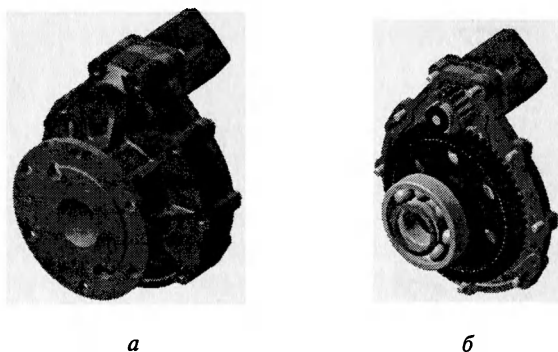


Рис. 6.3. Объемная модель редуктора, спроектированного с помощью КОМПАС-3D:

*а* — редуктор в сборе; *б* — редуктор без корпуса и вала

применение одной детали (сборки) в разных изделиях; передача изменений геометрии компонента во все сборки, где он используется.

Управление структурой (иерархией) сборки возможно в любой момент работы с ее моделью. Например, пользователь может объединить несколько компонентов в новую подсборку, или внести новые компоненты в существующую подсборку, или удалить любой компонент. При удалении компонента из сборки уничтожается только ссылка на него, а сам файл с моделью компонента остается на диске в неизменном виде.

В составе системы поставляется обширная библиотека стандартных элементов (крепежных изделий, опор валов и т. д.). Применение библиотечных элементов избавляет пользователя от необходимости самостоятельно создавать стандартные модели, высвобождая тем самым время для решения творческих задач проектирования.

*Операции, выполняемые с компонентами.* Сразу после вставки компонента в сборку его положение таково, что ориентация его системы координат совпадает с ориентацией системы координат сборки. Пользователю доступны разнообразные команды, позволяющие позиционировать компонент в сборке и связать его с другими компонентами.

Для перемещения компонента пользователь свободно передвигает его мышью в плоскости экрана или поворачивает его мышью вокруг указанной оси или вершины.

Для точного указания взаимного положения компонентов пользователь определяет (задает) сопряжения между гранями, ребрами и вершинами компонентов. В КОМПАС-3D предусмотрены следующие типы сопряжений:

- совпадение;
- параллельность;
- перпендикулярность;
- «на расстоянии»;
- «под углом»;
- концентричность;
- касательность.

При установлении сопряжения можно выбрать ориентацию компонентов. Например, при совпадении плоских граней деталей они (детали) могут быть установлены по одну или по разные стороны от плоскости граней, при размещении вершины на расстоянии от грани это расстояние может быть отложено «внутри» или «наружу» грани и т. д.

Процесс задания сопряжений как бы повторяет действия слесаря-сборщика: каждая деталь (подсборка) последовательными действиями приставляется к соседним деталям. Каждое сопряжение лишает компонент одной или нескольких степеней свободы.

Следовательно, компонент, участвующий в сопряжении, невозможно произвольно перемещать в пространстве (сдвигать, поворачивать). Например, если крышка установлена концентрично люку, ее можно поворачивать только вокруг оси и сдвигать вдоль этой же оси (поворот и сдвиг в иных направлениях не допускается). Если компонент участвует в сопряжениях, полностью определяющих его положение, задание для него новых сопряжений невозможно (как и свободное изменение его положения). Положение компонента можно сделать постоянным, не прибегая к наложению сопряжений. Команда фиксации компонента «замораживает» его в текущем положении. Сопряжения компонентов могут быть в любой момент отредактированы или удалены.

*Формообразующие операции и вспомогательные построения.* При формировании модели сборки возможно не только взаимное перемещение компонентов, но и выполнение формообразующих операций. В сборке может выполняться удаление материала одновременно нескольких компонентов путем:

- вырезания элемента выдавливания;
- вырезания элемента вращения;
- вырезания кинематического элемента;
- вырезания элемента по сечениям;
- создания круглого отверстия;
- сечения плоскостью;
- сечения по эскизу.

Данные операции выполняются теми же способами, что и при моделировании отдельной детали. Аналогией таких операций, выполняемых в модели сборки, является совместная обработка нескольких деталей в сборе.

Вообще говоря, при редактировании сборки возможно «приклеивание» формообразующих элементов, создание уклонов, тонкостенных оболочек и т. д., но эти операции, в отличие от операций, выполняемых в сборе, применимы не к сборке в целом, а к конкретным деталям в ее составе.

При моделировании сборки доступны те же вспомогательные построения, что и при моделировании детали (создание вспомогательных осей, плоскостей, пространственных кривых). При этом эскизы создаются на вспомогательных плоскостях и плоских гранях любых компонентов.

*Массивы компонентов.* Подобно тому как при моделировании детали создаются упорядоченные массивы одинаковых формообразующих элементов, при моделировании сборки создаются массивы ее компонентов. Компоненты могут располагаться вдоль кривой, по параллелограммной или концентрической сетке. Если массив компонентов должен совпадать с каким-либо существующим массивом копий (например, положение винтов должно повторять положение отверстий под них), вместо задания парамет-

ров копирования можно указать образец массива. Например, если в крышке есть шесть отверстий, расположенных по окружности, этот массив может стать основой массива винтов, вставляемых в отверстия.

*«Дерево» построения, иерархия компонентов модели.* При работе в КОМПАС-3D вся последовательность построения сборки отображается в отдельном окне в виде «дерева» построения. В нем перечислены все существующие в модели вспомогательные элементы, компоненты, сопряжения и формообразующие операции в порядке их создания, причем ветви, соответствующие компонентам, показывают, в свою очередь, последовательность создания этих компонентов.

Компоненты для выполнения операций можно указывать не только в окне редактирования модели, но и в «дереве» построения. При выделении любого компонента в «дереве» соответствующая ему часть модели подсвечивается в окне модели.

Помимо «дерева», отражающего «историю» создания сборки, система запоминает иерархию компонентов модели. В любой момент возможен просмотр иерархии в специальном диалоге, в котором отображаются все топологические отношения между элементами модели. Например, сопряжение какой-либо детали располагается в иерархической ветви, соответствующей этой детали.

*Параметрические свойства сборки.* Параметрические свойства модели складываются из вариационной параметризации эскизов компонентов и иерархической параметризации отношений между компонентами. В модели хранится информация не только об абсолютных координатах и размерах компонентов, но и о способах их построения и взаимном положении. Наличие параметрических связей и ограничений в модели накладывает отпечаток на принципы ее редактирования. Модель сборки редактируется путем изменения сопряжений, параметров и положения компонентов, при этом сохраняются все существующие в ней параметрические связи.

*Редактирование сопряжений.* При редактировании сопряжений можно изменить их параметры (например, указать другое расстояние между параллельными гранями) и заменить участвующие в сопряжении элементы (к примеру, вместо параллельности грани плоскости сделать ее параллельной другой грани). Любое сопряжение можно удалить из модели.

*Редактирование компонента в контексте сборки.* Можно перейти в специальный режим, в котором работают все команды редактирования отдельного компонента (например, детали), при этом остальные компоненты сборки доступны для выбора и указания. Элементы этих компонентов можно проецировать в эскиз редактируемого компонента и использовать для задания гранич-

ных условий (например, выдавливать элемент до грани ближайшей детали).

*Редактирование компонента в отдельном окне.* При этом способе редактирования в окне компонента отсутствует его окружение. После сохранения измененного компонента содержащая его сборка перестраивается так, чтобы существующие в модели связи и ограничения не были нарушены. Следует особо подчеркнуть, что после редактирования компонента, занимающего любое место в иерархии построений, не требуется заново задавать последовательность построения подчиненных ему компонентов и их параметры. Вся эта информация хранится в модели и не изменяется при редактировании отдельных ее частей.

*Дополнительные сервисные возможности.* Возможности системы КОМПАС-3D не ограничиваются только операциями моделирования изделий, система обладает множеством сервисных функций, позволяющих облегчить использование модели спроектированного изделия. Система автоматически обнаруживает области пересечения и касания указанных пользователем компонентов (или всех компонентов сборки), что позволяет проконтролировать их наличие в конструкции и избежать их.

Иногда при формировании изображения изделия (например, для каталога) требуется показать его не в сборе, а в виде разнесенных в пространстве компонентов. Пользователю КОМПАС-3D достаточно указать направление и величину перемещения компонентов, система разнесет их автоматически; обратная операция — совмещение компонентов тоже выполняется автоматически.

Одной из целей построения трехмерной модели является получение комплекта документации для изготовления соответствующего изделия. В КОМПАС-3D генерируются чертежи трехмерных моделей (деталей и сборок) в указанных пользователем проекциях, с выбранными разрезами и сечениями. Кроме того, спецификации на созданные в КОМПАС-3D сборки заполняются автоматически с учетом состава и структуры изделия (обозначения, наименования и количества деталей записываются в раздел «Детали», информация о подсборках — в раздел «Сборочные единицы» и т. д.). Полученные таким образом документы представляют собой чертежи и спецификации формата КОМПАС-ГРАФИК (графического редактора системы КОМПАС) и могут быть при необходимости доработаны его средствами.

Для упрощения и ускорения разработки чертежей, содержащих стандартные типовые детали, в КОМПАС-ГРАФИК созданы различные библиотеки.

Библиотека — это приложение, созданное для расширения стандартных возможностей КОМПАС-ГРАФИК, работающее в его среде и представляющее собой сложную подсистему автоматизированного проектирования, которая после выполнения

проектных расчетов формирует готовые конструкторские документы.

Библиотека КОМПАС-SHAFT 5 Plus позволяет не просто вставлять в чертежи изображения различных валов, но и производить проектные и проверочные расчеты этих конструктивных элементов, автоматически создавать для них таблицы параметров и технические требования.

КОМПАС-SHAFT 5 Plus — это стандартное приложение Windows, поэтому экран, который появляется после запуска системы, имеет многие стандартные атрибуты приложений Windows (заголовок окна с системным меню и пиктограммами минимизации, максимизации и закрытия окна, рамку для изменения размеров окна).

Внешний вид рабочего окна КОМПАС-SHAFT 5 Plus показан на рис. 6.4.

В верхней части окна диалога КОМПАС-SHAFT 5 Plus находится панель управления, на которой расположены кнопки с пиктограммами, предназначенные для быстрого вызова команд.

Окно диалога разделено на две области: в верхней отображается «дерево» построения внешнего контура, а в нижней — «дерево» построения внутреннего контура разрабатываемого вала. В области внешнего и внутреннего контуров по мере построения бу-

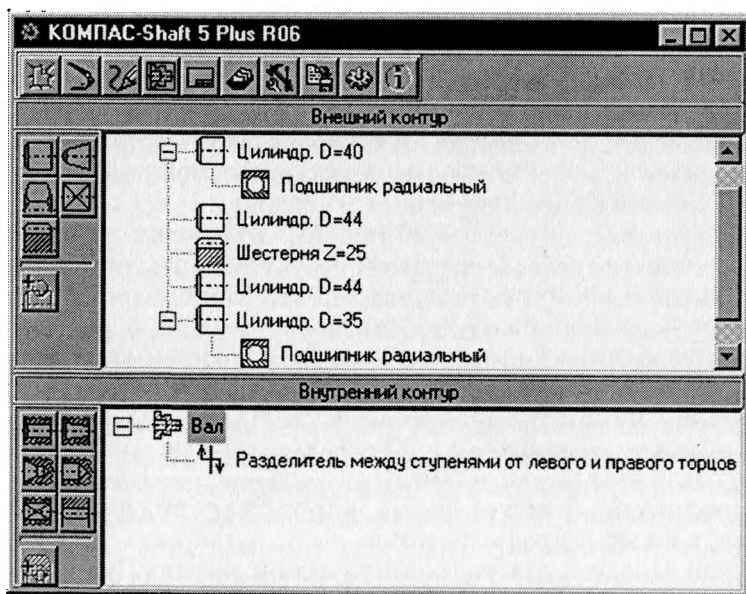


Рис. 6.4. Внешний вид рабочего окна КОМПАС-SHAFT 5 Plus

дуг отображаться пиктограммы построенных элементов. Вал, разработанный в КОМПАС-SHAFT 5 Plus, является макроэлементом.

В процессе работы можно отредактировать любой элемент вала. При этом после задания новых параметров все связи внутри вала сохраняются.

Команды построения внешнего контура следующие:

- цилиндрическая ступень внешнего контура;
- коническая ступень внешнего контура;
- шестигранник;
- элементы механических передач.

Первые три команды достаточно просты. В диалоговом режиме на экране задаются параметры ступеней вала, после чего ступень вала вычерчивается на экране монитора.

Помимо построения самих элементов вала, имеется возможность дополнительных построений. Например, дополнительно на цилиндрической ступени внешнего контура могут быть построены:

- канавка (под стопорное кольцо или для выхода инструмента);
- подшипники;
- кольцевые пазы;
- резьба (метрическая, дюймовая, трубная и др.);
- шлицы (прямобочные, эвольвентные, треугольные);
- шпоночный паз (под цилиндрические и сегментные шпонки).

Важным обстоятельством является то, что по желанию проектировщика любые элементы могут быть вынесены и образмерены отдельно. В частности, можно построить профили любых шлицов с автоматическим указанием их размеров. Шпонки могут быть включены в состав чертежа.

Наибольший интерес вызывает команда «Элементы механических передач». Возможен расчет и построение следующих элементов механических передач:

- шестерня цилиндрической зубчатой передачи;
- шестерня конической передачи с круговыми зубьями;
- шестерня конической передачи с прямыми зубьями;
- червяк цилиндрической червячной передачи;
- червячное колесо цилиндрической червячной передачи;
- звездочка цепной передачи с роликовой цепью;
- шкив клиноременной передачи.

Эти детали могут быть построены как отдельно, так и включены в состав вала (вал-шестерня).

Дополнительно могут быть построены таблицы параметров зубчатых колес и профили зубьев с автоматическим нанесением размеров. Предусмотрены геометрический расчет зубчатых колес, расчеты на прочность и долговечность.

Для цилиндрических передач предусмотрено три варианта расчета:



по межосевому расстоянию;  
коэффициентам смещения;  
диаметрам вершин колес.

Отличительными особенностями рассматриваемой программы являются наличие большого количества баз данных, необходимых в процессе проектирования, в частности базы данных по подшипникам качения и конструкционным материалам, возможность обращения к базам данных в процессе проектирования вала.

Для построения внутреннего контура могут быть использованы следующие команды:

- цилиндрическая ступень внутреннего контура;
- коническая ступень внутреннего контура;
- центровое отверстие;
- глухое отверстие;
- шестерня внутреннего зацепления.

Дополнительные элементы цилиндрической ступени внутреннего контура те же, что и для наружного.

Пример вала, построенного с помощью программы КОМПАС-SHAFT Plus, приведен на рис. 6.5.

Важной особенностью САПР КОМПАС 5.x является комплексность решений конструкторских и технологических задач, что дает возможность использования этой системы для сопровождения проекта начиная эскизным проектированием и заканчивая технологической подготовкой производства.

Новые возможности, предоставляемые КОМПАС-3D, позволяют существенно повысить эффективность проектирования.

Основные задачи, которые решает система КОМПАС-3D, — формирование трехмерной модели детали с целью передачи геометрии в различные расчетные пакеты или пакеты разработки управляющих программ для оборудования с ЧПУ, а также создание конструкторской документации на разработанные детали. Некоторые расчеты (масса, объем, площадь поверхности, инерционные характеристики детали) можно произвести непосредственно в КОМПАС-3D.

В качестве примера удачного взаимодействия КОМПАС-3D с технологическими пакетами можно привести опыт формирова-

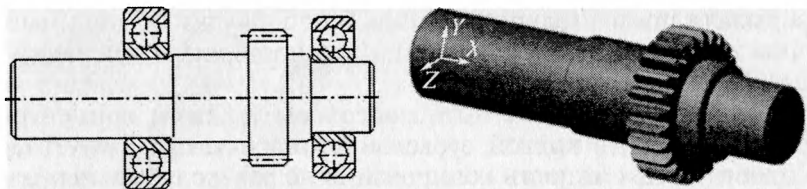


Рис. 6.5. Пример вала, построенного в системе КОМПАС-SHAFT 5 Plus

ния в системе ГЕММА-3D управляющих программ для механической обработки деталей, спроектированных в КОМПАС-3D.

Детали, смоделированные в КОМПАС-3D, можно передать для создания сборочной единицы в другие системы трехмерного моделирования. Например, возможен такой вариант оснащения конструкторского подразделения: на одном рабочем месте в системе SolidWorks производится сборка изделия из деталей, спроектированных на нескольких рабочих местах в КОМПАС-3D.

Передача данных из КОМПАС-3D в другие системы производится через стандартные форматы обмена — IGES и SAT. Для передачи геометрии детали (точнее, информации о ее поверхности) на стереолитографическое оборудование используется формат STL.

Некоторые функции КОМПАС-3D разработаны специально для быстрой реализации стандартных приемов проектирования (например, использования существующих наработок, создания зеркально-симметричных деталей и т.д.).

Удобный прием моделирования изделий, которые отличаются лишь некоторыми конструктивными элементами, — использование в качестве основания (первого объемного элемента) детали ранее подготовленной модели (она называется деталью-заготовкой). Деталь-заготовку можно вставить в модель, сохранив ссылку на содержащий ее файл (иначе говоря, сохранив связь с файлом-источником). В этом случае любые изменения модели в файле-источнике будут передаваться во все модели, содержащие данную заготовку. Использование детали-заготовки в некоторых случаях позволяет значительно ускорить работу системы при моделировании деталей высокой сложности.

В модель можно вставлять не только деталь-заготовку, но и ее зеркальную копию. Благодаря этому модели зеркально-симметричных деталей можно создать за несколько секунд, причем получаемая деталь может отслеживать все изменения, вносимые в ее прототип, и самостоятельно перестраиваться в соответствии с изменениями, сохраняя свойство симметрии.

Часто при построении тела требуется произвести несколько одинаковых операций так, чтобы образовавшиеся элементы были определенным образом упорядочены (например, отверстия образовывали прямоугольный массив, или бобышки были симметричны относительно плоскости). Для повторения операций в КОМПАС-3D предусмотрены различные варианты команд копирования: копирование по параллелограммной и концентрической сетке, вдоль кривой, зеркальное копирование. Следует также отметить возможность копирования не только отдельных элементов, но и наборов элементов, а также массивов копий (копия копии) и возможность удаления отдельных экземпляров из массива копий.

Система КОМПАС-3D предоставляет пользователю разнообразные средства получения плоских изображений модели — от печати каркасного или полутонного изображения, видимого на экране, до автоматического создания чертежа КОМПАС-ГРАФИК, содержащего выбранные виды детали. В таком чертеже доступны все команды редактирования изображения, простановки размеров и технологических обозначений, знакомые пользователям по работе с графическими документами КОМПАС-ГРАФИК.

При создании плоских изображений детали можно выбирать не только ее стандартные проекции (вид слева, вид сверху и др.), но и любую пользовательскую ориентацию детали; это позволяет быстро получать качественные изображения деталей для каталогов, текстовых документов и т. д.

Обычно при использовании системы трехмерного моделирования вначале создается модель изделия, а затем (при необходимости) его плоские изображения (например, рабочие чертежи деталей). Однако иногда требуется построить трехмерную модель детали, документация на которую уже выпущена (это происходит, как правило, при использовании разработанных ранее деталей во вновь проектируемых изделиях). И очень кстати оказывается полная интеграция компонентов системы КОМПАС — КОМПАС-3D и КОМПАС-ГРАФИК: изображения из любых графических документов КОМПАС можно использовать при построении трехмерной модели. Особенно эффективно построение таким способом моделей, сформированных небольшим количеством операций над сложными контурами.

В настоящее время система КОМПАС широко применяется на предприятиях отечественного автомобиле- и тракторостроения; в частности, гигант отечественного машиностроения АвтоВАЗ, Челябинский тракторный завод, Ярославское объединение «Автодизель» активно используют эту систему в своей работе.

## СПИСОК ЛИТЕРАТУРЫ

1. *Атре Ш.* Структурный подход к организации баз данных. — М.: Финансы и статистика, 1983.
2. *Бородулин Ю. Б., Гусев В. А., Попов Г. В.* Автоматизированное проектирование силовых трансформаторов. — М.: Энергоатомиздат, 1987.
3. *Дементьев Ю. В., Пешкилев А. Г.* Технические средства САПР: Учеб. пособие. — М.: МАМИ, 1985.
4. *Дорн Т.* КОМПАС-SHAFT Plus: Новые возможности, или Как облегчить работу конструктора // САПР и Графика. № 8. 2002. — С. 66—69.
5. *Корячко В. П., Курейчик В. М., Норенков И. П.* Теоретические основы САПР. — М.: Энергоатомиздат, 1987.
6. *Майоров С. А., Кириллов В. В., Приблуда А. А.* Введение в микроЭВМ. — Л.: Машиностроение, 1988.
7. *Норенков И. П.* Введение в автоматизированное проектирование технических устройств и систем. — М.: Высш. шк., 1986.
8. *Норенков И. П.* Принципы построения и структуры САПР. — М., 1986.
9. *Панченко В., Николаева И.* КОМПАС-3D 5.11. Новый уровень возможностей для заказчиков компании АСКОН // САПР и Графика. № 9. 2001. — С. 26—29.
10. Перспективы развития вычислительной техники: Справ. пособие: В 11 кн. Кн. 8: Периферийное и терминальное оборудование ЭВМ / Л. В. Семенов, В. Н. Абрамов, М. Г. Арутюнов и др.; Под ред. Ю. М. Смирнова. — М.: Высш. шк., 1990.
11. Перспективы развития вычислительной техники: Справ. пособие: В 11 кн. Кн. 11: Программное обеспечение ЭВМ / А. С. Марков, М. П. Миллов, Г. В. Пеледов и др.; Под ред. Ю. М. Смирнова. — М.: Высш. шк., 1990.
12. Программное обеспечение микроЭВМ: Учеб. пособие: В 11 кн. Кн. 1: *Костин А. Е.* Структура и функционирование микроЭВМ / Под ред. В. Ф. Шаньгина. — М.: Высш. шк., 1991.
13. Разработка САПР: Практ. пособие: В 10 кн. Кн. 1: *Петров А. В., Черненький В. М.* Проблемы и принципы создания САПР / Под ред. А. В. Петрова. — М.: Высш. шк., 1990.
14. Разработка САПР: Практ. пособие: В 10 кн. Кн. 4: *Вейнеров О. М., Самохвалов Э. Н.* Проектирование баз данных САПР / Под ред. А. В. Петрова. — М.: Высш. шк., 1990.
15. САПР: Системы автоматизированного проектирования: Учеб. пособие: В 9 кн. Кн. 1: *Норенков И. П.* Принципы построения и структура. — Минск: Вышэйш. шк., 1987.

16. САПР: Системы автоматизированного проектирования: Учеб. пособие: В 9 кн. Кн. 2: Технические средства и операционные системы / Д. М. Жук, В. А. Мартынюк, П. А. Сомов и др.; Под ред. И. П. Норенкова. — Минск: Вышэйш. шк., 1988.

17. САПР: Системы автоматизированного проектирования: Учеб. пособие: В 9 кн. Кн. 3: *Федорук В. Г., Черненко В. М.* Информационное и прикладное программное обеспечение / Под ред. И. П. Норенкова. — Минск: Вышэйш. шк., 1988.

18. САПР: Системы автоматизированного проектирования: Учеб. пособие: В 9 кн. Кн. 4: *Трудоношин В. А., Пивоварова Н. В.* Математические модели технических объектов / Под ред. И. П. Норенкова. — Минск: Вышэйш. шк., 1987.

19. САПР: Системы автоматизированного проектирования: Учеб. пособие: В 9 кн. Кн. 5: *Кузьмин П. К., Маничев В. Б.* Автоматизация функционального проектирования / Под ред. И. П. Норенкова. — Минск: Вышэйш. шк., 1988.

20. САПР: Системы автоматизированного проектирования: Учеб. пособие: В 9 кн. Кн. 6: *Капустин Н. М., Васильев Г. Н.* Автоматизация конструкторского и технологического проектирования / Под ред. И. П. Норенкова. — Минск: Вышэйш. шк., 1988.

21. *Хауз Р.* Использование AutoCAD 2000. — М.: Издательский дом «Вильямс», 2001.

22. *Эндерле Г., Кэнси К., Пфафф Г.* Программные средства машинной графики. — М.: Радио и связь, 1988.

# ОГЛАВЛЕНИЕ

Предисловие .....	3
Сокращения .....	5
<b>Глава 1. ОБЩИЕ СВЕДЕНИЯ О СИСТЕМАХ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ .....</b>	<b>6</b>
1.1. Применение ЭВМ для автоматизации проектирования и технологической подготовки производства автомобилей и тракторов .....	6
1.2. САПР и роль проектировщика в автоматизированном проектировании .....	11
1.3. Структурная схема и классификация САПР .....	15
1.4. Подходы и методы проектирования в САПР .....	20
1.5. Способы представления графической информации в ЭВМ .....	24
1.6. Задачи синтеза и анализа. Оптимальное проектирование конструкций .....	29
1.7. Методы решения задач оптимизации .....	35
<b>Глава 2. МАТЕМАТИЧЕСКИЕ МОДЕЛИ ОБЪЕКТОВ ПРОЕКТИРОВАНИЯ .....</b>	<b>44</b>
2.1. Общие сведения .....	44
2.2. Преобразование математических моделей в процессе получения рабочих программ анализа .....	57
2.3. Математические модели объектов на макроуровне .....	60
2.4. Формальное представление структуры объекта на макроуровне .....	67
2.5. Примеры составления эквивалентных схем технических объектов .....	72
2.6. Моделирование работы технических объектов на макроуровне .....	84
<b>Глава 3. ПРОГРАММНОЕ И ЛИНГВИСТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САПР .....</b>	<b>87</b>
3.1. Общее программное обеспечение .....	87
3.2. Специальное программное обеспечение .....	101
3.3. Классификация и использование языков в САПР .....	106
3.4. Языковые средства машинной графики .....	111

<b>Глава 4. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ САПР</b> .....	115
4.1. Общие сведения .....	115
4.2. Банки данных .....	117
4.3. Модели представления данных .....	126
<b>Глава 5. ТЕХНИЧЕСКИЕ СРЕДСТВА САПР</b> .....	132
5.1. Общие сведения .....	132
5.2. Электронные вычислительные машины в САПР .....	133
5.3. Периферийные устройства ЭВМ .....	141
<b>Глава 6. СОВРЕМЕННЫЕ САПР АГРЕГАТОВ, УЗЛОВ И ДЕТАЛЕЙ</b> .....	176
6.1. Сведения о некоторых САПР зарубежной разработки .....	176
6.2. Отечественные САПР, используемые в автомобиле- и тракторостроении .....	187
Список литературы .....	215

*Учебное издание*

**Дементьев Юрий Витальевич,  
Щетинин Юрий Сергеевич**

**САПР в автомобиле- и тракторостроении**  
**Учебник**

Редактор *Е. А. Зубкова*

Технический редактор *Н. И. Горбачева*

Компьютерная верстка: *В. А. Крыжко*

Корректоры *Л. М. Зинченко, С. Ю. Свиридова*

Диапозитивы предоставлены издательством.

Изд. № А-1131-1. Подписано в печать 28.04.2004. Формат 60×90/16.

Бумага тип. № 2. Гарнитура «Таймс». Печать офсетная. Усл. печ. л. 14,0.

Тираж 5100 экз. Заказ № 13201.

Лицензия ИД № 02025 от 13.06.2000. Издательский центр «Академия».

Санитарно-эпидемиологическое заключение № 77.99.02.953.Д.003903.06.03 от 05.06.2003.

117342, Москва, ул. Бултерова, 17-Б, к. 223. Тел./факс: (095) 330-1092, 334-8337.

Отпечатано на Саратовском полиграфическом комбинате.

410004, г. Саратов, ул. Чернышевского, 59.