

Министерство образования и науки РФ

Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
Владимирский государственный университет имени  
Александра Григорьевича и Николая Григорьевича Столетовых

Кафедра приборостроения и  
информационно измерительных технологий

## **САПР ИНФОРМАЦИОННО ИЗМЕРИТЕЛЬНОЙ ТЕХНИКИ**

(конспект лекций)

Составитель  
Л.К.Генералов

Владимир 2013

УДК 681.31 (075.8)

ББК 32.97

Рецензент:

Кандидат технических наук, доцент кафедры  
мехатроника и электронные системы автомобилей

*В.П. Умнов*

Методическое пособие «САПР информационно измерительной техники» (конспект лекций) / сост.: Л.К.Генералов; Владим. гос. ун – т. – Владимир : Электронный ресурс. 2013. – 143 с.

В пособии даны сведения по различным аспектам и видам обеспечения систем автоматического проектирования приборов и приборных систем, необходимые магистрам в практической деятельности. Значительное внимание уделено вопросам проектирования, анализа и синтеза динамических систем с использованием интегрированных пакетов прикладных программ. Освящены вопросы разработки печатных плат в системах P-CAD и Micro-CAP.

Содержание лекций соответствует программе подготовки магистров по дисциплине «САПР информационно измерительной техники», который автор читает в ВлГУ на кафедре Приборостроения и информационно – измерительных технологий.

Конспект лекций может быть полезен магистрантам машиностроительных и радиотехнических специальностей

УДК 681.31 (075.8)  
ББК 32.97

## **Введение**

В государственном стандарте направления 200100 — «Приборостроение» предусмотрено изучение магистрантами очной формы обучения дисциплины «САПР информационно – измерительной техники». В процессе изучения дисциплины магистранты осваивают наиболее общие положения, модели и методики автоматизированного проектирования, входящие в программу курса, детально изучают методы и программы, посвященные САПР приборостроения. Данный конспект лекций ориентирован на базовую подготовку магистрантов в области САПР. Лекции 1,2,3 посвящены общим вопросам САПР, применительно к приборостроению, в лекциях 4, 5 и 6 рассматриваются вопросы конструирования и проектирования приборов и приборных систем с использованием графических пакетов программ и программ исследования динамики. Лекции 7, 8 и 9 являются основополагающими при изучении программных продуктов схемотехнического моделирования и проектирования печатных плат. Дано описание типовой последовательности проектных процедур и использования поддерживающих программ различного назначения.

## **Лекция №1**

### **САПР информационно – измерительной техники**

## **Введение**

Автоматизация проектирования занимает особое место среди информационных технологий. Во первых, автоматизация проектирования — синтетическая дисциплина, ее составными частями являются многие другие современные информационные технологии. Так, техническое обеспечение систем автоматизированного проектирования (САПР) основано на использовании вычислительных сетей и телекоммуникационных технологий, в САПР используются персональные компьютеры и рабочие станции.

Математическое обеспечение САПР отличается богатством и разнообразием используемых методов вычислительной математики, статистики, математического программирования, дискретной математики, искусственного интеллекта. Программные комплексы САПР относятся к числу наиболее сложных современных программных систем, основанных на опера-

ционных системах Unix, Windows 95/NT, языках программирования. С, С++, Java и других, современных CASE технологиях, реляционных и объектно-ориентированных системах управления базами данных (СУБД), стандартах открытых систем и обмена данными в компьютерных средах.

Во вторых, знание основ автоматизации проектирования и умение работать со средствами САПР требуется практически любому инженеру разработчику. Компьютерами насыщены проектные подразделения, конструкторские бюро и офисы. Работа конструктора за обычным кульманом, расчеты с помощью логарифмической линейки или оформление отчета на пишущей машинке стали анахронизмом. Предприятия, ведущие разработки без САПР или лишь с малой степенью их использования, оказываются неконкурентоспособными как из-за больших материальных и временных затрат на проектирование, так и из-за невысокого качества проектов. Появление первых программ для автоматизации проектирования за рубежом и в СССР относится к началу 60-х гг. Тогда были созданы программы для решения задач строительной механики, анализа электронных схем, проектирования печатных плат.

Дальнейшее развитие САПР шло по пути создания аппаратных и программных средств машинной графики, повышения вычислительной эффективности программ моделирования и анализа, расширения областей применения САПР, упрощения пользовательского интерфейса, внедрения в САПР элементов искусственного интеллекта.

К настоящему времени создано большое число программно-методических комплексов для САПР с различной степенью специализации и прикладной ориентацией. В результате автоматизация проектирования стала необходимой составной частью подготовки инженеров разных специальностей; инженер, не владеющий знаниями и не умеющий работать в САПР, не может считаться полноценным специалистом.

Подготовка инженеров разных специальностей в области САПР включает базовую и специальную компоненты. Наиболее общие положения, модели и методики автоматизированного проектирования входят в программу курса, посвященного основам САПР, более детальное изучение тех методов и программ, которые специфичны для конкретных специальностей, предусматривается в профильных дисциплинах.

## **2. Основные принципы построения САПР**

Разработка САПР представляет собой крупную научно-техническую проблему, а ее внедрение требует значительных капиталовложений. Накопленный опыт позволяет выделить следующие основные принципы построения САПР.

1. САПР — человеко-машинная система. Все созданные и создаваемые системы проектирования с помощью ЭВМ являются автоматизированными, важную роль в них играет человек — инженер, разрабатывающий проект технического средства.

В настоящее время и по крайней мере в ближайшие годы создание систем автоматического проектирования не предвидится, и ничто не угрожает монополии человека при принятии узловых решений в процессе проектирования. Человек в САПР должен решать, во-первых, все задачи, которые не формализованы, во-вторых, задачи, решение которых человек осуществляет на основе своих эвристических способностей более эффективно, чем современная ЭВМ на основе своих вычислительных возможностей. Тесное взаимодействие человека и ЭВМ в процессе проектирования — один из принципов построения и эксплуатации САПР.

2. САПР — иерархическая система, реализующая комплексный подход к автоматизации всех уровней проектирования. Иерархия уровней проектирования отражается в структуре специального программного обеспечения САПР в виде иерархии подсистем.

Следует особо подчеркнуть целесообразность обеспечения комплексного характера САПР, так как автоматизация проектирования лишь на одном из уровней оказывается значительно менее эффективной, чем полная автоматизация всех уровней. Иерархическое построение относится не только к специальному программному обеспечению, но и к техническим средствам САПР, разделяемых на центральный вычислительный комплекс и автоматизированные рабочие места проектировщиков.

3. САПР — совокупность информационно-согласованных подсистем. Этот очень важный принцип должен относиться не только к связям между крупными подсистемами, но и к связям между более мелкими частями подсистем. Информационная согласованность означает, что все или

большинство возможных последовательностей задач проектирования обслуживаются информационно согласованными программами. Две программы являются информационно согласованными, если все те данные, которые представляют собой объект переработки в обеих программах, входят в числовые массивы, не требующие изменений при переходе от одной программы к другой. Так, информационные связи могут проявляться в том, что результаты решения одной задачи будут исходными данными для другой задачи. Если для согласования программ требуется существенная переработка общего массива с участием человека, который добавляет недостающие параметры, вручную перекомпоновывает массив или изменяет числовые значения отдельных параметров, то программы информационно не согласованы. Ручная перекомпоновка массива ведет к существенным временным задержкам, росту числа ошибок и поэтому уменьшает спрос на услуги САПР. Информационная несогласованность превращает САПР в совокупность автономных программ, при этом из-за неучета в подсистемах многих факторов, оцениваемых в других подсистемах, снижается качество проектных решений.

4. САПР — открытая и развивающаяся система. Существует, по крайней мере, две веские причины, по которым САПР должна быть изменяющейся во времени системой. Во-первых, разработка столь сложного объекта, как САПР, занимает продолжительное время, и экономически выгодно вводить в эксплуатацию части системы по мере их готовности. Введенный в эксплуатацию базовый вариант системы в дальнейшем расширяется. Во-вторых, постоянный прогресс техники, проектируемых объектов, вычислительной техники и вычислительной математики приводит к появлению новых, более совершенных математических моделей и программ, которые должны заменять старые, менее удачные аналоги. Поэтому САПР должна быть открытой системой, т. е. обладать свойством удобства использования новых методов и средств.

5. САПР — специализированная система с максимальным использованием унифицированных модулей. Требования высокой эффективности и универсальности, как правило, противоречивы. Применительно к САПР это положение сохраняет свою силу. Высокой эффективности САПР, выражаемой прежде всего малыми временными и материальными затратами

при решении проектных задач, добиваются за счет специализации систем. Очевидно, что при этом растет число различных САПР. Чтобы снизить расходы на разработку многих специализированных САПР, целесообразно строить их на основе максимального использования унифицированных составных частей. Необходимым условием унификации является поиск общих черт и положений в моделировании, анализе и синтезе разнородных технических объектов. Безусловно, может быть сформулирован и ряд других принципов, что подчеркивает многосторонность и сложность проблемы САПР.

### **3. Системный подход к проектированию**

Основные идеи и принципы проектирования сложных систем выражены в системном подходе. Для специалиста в области системотехники они являются очевидными и естественными, однако, их соблюдение и реализация зачастую сопряжены с определенными трудностями, обусловливаемыми особенностями проектирования. Как и большинство взрослых образованных людей, правильно использующих родной язык без привлечения правил грамматики, инженеры используют системный подход без обращения к пособиям по системному анализу. Однако интуитивный подход без применения правил системного анализа может оказаться недостаточным для решения все более усложняющихся задач инженерной деятельности.

Основной общий принцип системного подхода заключается в рассмотрении частей явления или сложной системы с учетом их взаимодействия. Системный подход выявляет структуру системы ее внутренние и внешние связи.

Системы автоматизированного проектирования и управления относятся к числу наиболее сложных современных искусственных систем. Их проектирование и сопровождение невозможны без системного подхода. Поэтому идеи и положения системотехники входят составной частью в дисциплины, посвященные изучению современных автоматизированных систем и технологий их применения.

### **4. Структура САПР**

Как и любая сложная система, САПР состоит из подсистем. Различают подсистемы проектирующие и обслуживающие.

Проектирующие подсистемы непосредственно выполняют проектные процедуры. Примерами проектирующих подсистем могут служить подсистемы геометрического трехмерного моделирования механических объектов, изготовления конструкторской документации, схемотехнического анализа, трассировки соединений в печатных платах.

Обслуживающие подсистемы обеспечивают функционирование проектирующих подсистем, их совокупность часто называют системной средой (или оболочкой) САПР. Типичными обслуживающими подсистемами являются подсистемы управления проектными данными, подсистемы разработки и сопровождения программного обеспечения CASE (Computer Aided Software Engineering), обучающие подсистемы для освоения пользователями технологий, реализованных в САПР.

## **5. Виды обеспечения САПР**

Структурирование САПР по различным аспектам обуславливает появление видов обеспечения САПР. Принято выделять семь видов обеспечения САПР:

техническое (ТО), включающее различные аппаратные средства (ЭВМ, периферийные устройства, сетевое коммутационное оборудование, линии связи, измерительные средства);

математическое (МО), объединяющее математические методы, модели и алгоритмы для выполнения проектирования;

программное (ПО), представляемое компьютерными программами САПР;

информационное (ИО), состоящее из базы данных, СУБД, а также включающее другие данные, которые используются при проектировании; отметим, что вся совокупность используемых при проектировании данных называется информационным фондом САПР, база данных вместе с СУБД носит название банка данных;

лингвистическое (ЛО), выражаемое языками общения между проектировщиками и ЭВМ, языками программирования и языками обмена данными между техническими средствами САПР;

методическое (МетО), включающее различные методики проектирования; иногда к нему относят также математическое обеспечение;

организационное (ОО), представляемое штатными расписаниями, должностными инструкциями и другими документами, которые регламентируют работу проектного предприятия.

## **6. Разновидности САПР**

Классификацию САПР осуществляют по ряду признаков, например по приложению, целевому назначению, масштабам (комплексности решаемых задач), характеру базовой подсистемы — ядра САПР.

По приложениям наиболее представительными и широко используемыми являются следующие группы САПР:

САПР для применения в отраслях общего машиностроения. Их часто называют машиностроительными САПР или системами МСAD (Mechanical CAD);

САПР для радиоэлектроники: системы ЕСAD (Electronic CAD) или EDA (Electronic Design Automation);

САПР в области архитектуры и строительства.

Кроме того, известно большое число специализированных САПР, или выделяемых в указанных группах, или представляющих самостоятельную ветвь классификации. Примерами таких систем являются САПР больших интегральных схем (БИС); САПР летательных аппаратов; САПР электрических машин и т. п.

По целевому назначению различают САПР или подсистемы САПР, обеспечивающие разные аспекты (страты) проектирования. Так, в составе МСAD появляются рассмотренные выше САЕ/CAD/CAM-системы.

По масштабам различают отдельные программно-методические комплексы (ПМК) САПР, например: комплекс анализа прочности механических изделий в соответствии с методом конечных элементов (МКЭ) или комплекс анализа электронных схем; системы ПМК; системы с уникальными архитектурами не только программного (software), но и технического (hardware) обеспечений.

По характеру базовой подсистемы различают следующие разновидности САПР:

1. САПР на базе подсистемы машинной графики и геометрического моделирования. Эти САПР ориентированы на приложения, где основной процедурой проектирования является конструирование, т. е. определение пространственных форм и взаимного расположения объектов. К этой группе систем относится большинство САПР в области машиностроения, построенных на базе графических ядер.

В настоящее время широко используют унифицированные графические ядра, применяемые более чем в одной САПР (ядра Parasolid фирмы EDS Urographics и ACIS фирмы Intergraph).

2. САПР на базе СУБД. Они ориентированы на приложения, в которых при сравнительно несложных математических расчетах перерабатывается большой объем данных. Такие САПР преимущественно встречаются в технико-экономических приложениях, например при проектировании бизнес-планов, но они имеются также при проектировании объектов, подобных щитам управления в системах автоматики.

3. САПР на базе конкретного прикладного пакета. Фактически это автономно используемые ПМК, например имитационного моделирования производственных процессов, расчета прочности по МКЭ, синтеза и анализа систем автоматического управления и т. п. Часто такие САПР относятся к системам САЕ. Примерами могут служить программы логического проектирования на базе языка VHDL, математические пакеты типа MathCAD.

4. Комплексные (интегрированные) САПР, состоящие из совокупности подсистем предыдущих видов. Характерными примерами комплексных САПР являются САЕ/CAD/CAM-системы в машиностроении или САПР БИС. Так, САПР БИС включает в себя СУБД и подсистемы проектирования компонентов, принципиальных, логических и функциональных схем, топологии кристаллов, тестов для проверки годности изделий. Для

управления столь сложными системами применяют специализированные системные среды.

## Лекция № 2

### Обзор современных методов проектирования

#### Введение

Современные предприятия не смогут выжить во всемирной конкуренции, если не будут выпускать новые продукты лучшего качества; 2), более низкой стоимости и за меньшее время. Поэтому они стремятся использовать огромные возможности памяти компьютеров, их высокое быстродействие и возможности удобного графического интерфейса для того, чтобы автоматизировать и связать друг с другом задачи проектирования и производства, которые раньше были весьма утомительными и совершенно не связанными друг с другом. Таким образом сокращается время и стоимость разработки и выпуска продукта. Для этой цели используются технологии автоматизированного проектирования (**computer aided design — CAD**), автоматизированного производства (**computer aided manufacturing — CAM**) и автоматизированной разработки или конструирования (**computer aided engineering — CAE**). Чтобы понять значение систем CAD/CAM/CAE, мы должны изучить различные задачи и операции, которые приходится решать и выполнять в процессе разработки и производства продукта. Все эти задачи, взятые вместе, называются *жизненным циклом продукта (product cycle)*. Пример жизненного цикла продукта, описанного Зейдом, с незначительными усовершенствованиями приведен на рис. 2.1. Прямоугольники, нарисованные сплошными линиями, представляют два главных процесса, составляющих жизненный цикл продукта: процесс разработки и процесс производства. Процесс разработки начинается с запросов потребителей, которые обслуживаются отделом маркетинга, и заканчивается полным описанием продукта, обычно выполняемым в форме рисунка. Процесс производства начинается с технических требований и заканчивается поставкой готовых изделий. Операции, относящиеся к процессу разработки, можно разделить на аналитические и синтетические. Как следует из рис., 2.1, первичные операции разработки, такие как опреде-

ление необходимости разработки, формулирование технических требований, анализ осуществимости и сбор важной информации, а также концептуализация разработки, относятся к подпроцессу синтеза. Результатом подпроцесса синтеза является концептуальный проект предполагаемого продукта в форме эскиза или топологического чертежа, отражающего связи различных компонентов продукта. В этой части цикла делаются основные финансовые вложения, необходимые для реализации идеи продукта, а также определяется его функциональность. Большая часть информации, порождаемой и обрабатываемой в рамках подпроцесса синтеза является качественной, а следовательно неудобной для компьютерной обработки.

Готовый концептуальный проект анализируется и оптимизируется — это уже подпроцесс анализа. Прежде всего вырабатывается аналитическая модель, поскольку анализируется именно модель, а не сам проект. Несмотря на быстрый рост количества и качества компьютеров, используемых в конструировании, в обозримом будущем отказаться от использования абстракции аналитической модели мы не сможем. Аналитическая модель получается, если из проекта удалить маловажные детали, редуцировать размерности и учесть имеющуюся симметрию. Редукция размерностей, например, подразумевает замену тонкого листа из какого-либо материала на эквивалентную плоскость с атрибутом толщины или длинного и тонкого участка на линию с определенными параметрами, характеризующими поперечное сечение. Симметричность геометрии тела и нагрузки, приложенной к нему, позволяет рассматривать в модели лишь часть этого тела. Вообще говоря, вам уже приходилось заниматься подобным абстрагированием при анализе структур в курсе основ механики. Вспомните, что вы всегда начинали анализ с построения упрощенного эскиза структуры. Типичные примеры анализа: анализ напряжений, позволяющий проверить прочность конструкции, контроль столкновений, позволяющий обнаружить возможность столкновений движущихся частей, составляющих механизм, а также кинематический анализ, показывающий, что проектируемое устройство будет совершать ожидаемые движения. Качество результатов, которые могут быть получены в результате анализа, непосредственно связано с качеством выбранной аналитической модели, которым оно ограничивается.

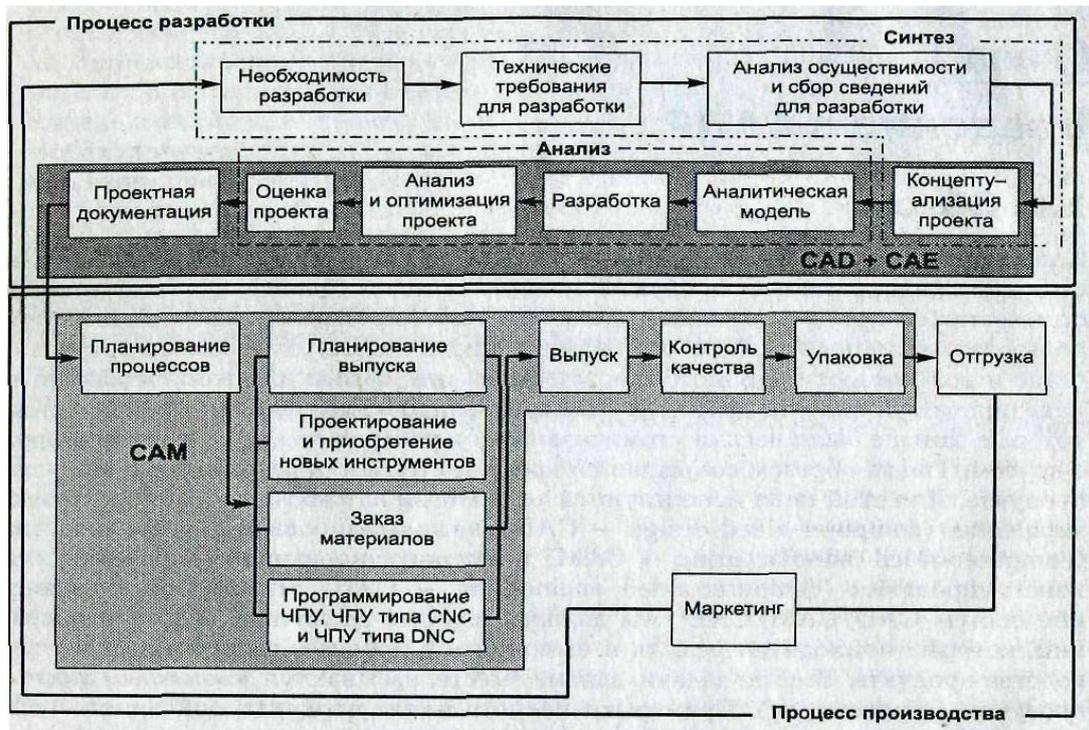


Рис.2.1. Жизненный цикл продукта

После завершения проектирования и выбора оптимальных параметров начинается этап оценки проекта. Для этой цели могут изготавливаться прототипы. В конструировании прототипов все большую популярность приобретает новая технология, названная *быстрым прототипированием*. Эта технология позволяет конструировать прототип снизу вверх, то есть непосредственно из проекта, поскольку фактически требует только лишь данных о поперечном сечении конструкции. Если оценка проекта на основании прототипа показывает, что проект не удовлетворяет требованиям, описанный выше процесс разработки повторяется снова.

Если же результат оценки проекта оказывается удовлетворительным, начинается подготовка проектной документации. К ней относятся чертежи, отчеты и списки материалов. Чертежи обычно копируются, а копии передаются на производство.

Как видно по рис. 2.1, процесс производства начинается с планирования, которое выполняется на основании полученных на этапе проектирования чертежей, а заканчивается готовым продуктом. Технологическая подготовка производства — это операция, устанавливающая список технологических процессов по изготовлению продукта и задающая их параметры.

Одновременно выбирается оборудование, на котором будут производиться технологические операции, такие как получение детали нужной формы из заготовки. В результате подготовки производства составляются план выпуска, списки материалов и программы для оборудования. На этом же этапе обрабатываются прочие специфические требования, в частности рассматриваются конструкции зажимов и креплений. Подготовка занимает в процессе производства примерно такое же место, как подпроцесс синтеза в процессе проектирования, требуя значительного человеческого опыта и принятия качественных решений. Такая характеристика подразумевает сложность компьютеризации данного этапа. После завершения технологической подготовки начинается выпуск готового продукта и его проверка на соответствие требованиям. Детали, успешно проходящие контроль качества, собираются вместе, проходят тестирование функциональности, упаковываются, маркируются и отгружаются заказчикам.

Выше мы описали типичный жизненный цикл продукта. Посмотрим теперь, каким образом на этапах этого цикла могут быть применены технологии CAD, CAM и CAE. Как уже говорилось, компьютеры не могут широко использоваться в подпроцессе синтеза, поскольку они не обладают способностью хорошо обрабатывать качественную информацию. Однако даже на этом этапе разработчик может, например, при помощи коммерческих баз данных успешно собирать важную для анализа осуществимости информацию, а также пользоваться данными из каталогов.

Непросто представить себе использование компьютера и в процессе концептуализации проекта, потому что компьютер пока еще не стал мощным средством для интеллектуального творчества. На этом этапе компьютер может сделать свой вклад, обеспечивая эффективность создания различных концептуальных проектов. Полезными могут оказаться средства параметрического и геометрического моделирования, а также макропрограммы в *системах автоматизированной разработки чертежей*. Все это типичные примеры систем CAD. *Система геометрического моделирования* — это трехмерный эквивалент системы автоматизированной разработки чертежей, то есть программный пакет, работающий с трехмерными, а не с плоскими объектами.

В аналитической фазе проектирования ценность компьютеров проявляется по-настоящему. Программных пакетов для анализа напряжений, кон-

троля столкновений и кинематического анализа существует столько, что приводить какие-либо названия смысла не имеет. Эти программные пакеты относятся к средствам автоматизированного конструирования (САЕ). Главная проблема, связанная с их использованием, заключается в необходимости формирования аналитической модели. Проблемы не существовало бы вовсе, если бы аналитическая модель автоматически выводилась из концептуального проекта. Однако, как уже отмечалось, аналитическая модель не идентична концептуальному проекту — она выводится из него путем исключения несущественных деталей и редукции размерностей. Необходимый уровень абстракции зависит от типа анализа и желаемой точности решения. Следовательно, автоматизировать процесс абстрагирования достаточно сложно, поэтому аналитическую модель часто создают отдельно. Обычно абстрактная модель проекта создается в системе разработки рабочих чертежей или в системе геометрического моделирования, а иногда с помощью встроенных средств аналитического пакета. Аналитические пакеты обычно требуют, чтобы исследуемая структура была представлена в виде объединения связанных сеток, разделяющих объект на отдельные участки, удобные для компьютерной обработки. Если аналитический пакет может генерировать сетку автоматически, человеку остается задать только границы абстрактного объекта. В противном случае сетка также создается пользователем либо в интерактивном режиме, либо автоматически, но в другой программе. Процесс создания сетки называется *моделированием методом конечных элементов*. Моделирование этим методом включает в себя также задание граничных условий и внешних нагрузок.

Подпроцесс анализа может выполняться в цикле оптимизации проекта по каким-либо параметрам. Разработано множество алгоритмов поиска оптимальных решений, а на их основе построены коммерчески доступные программы. Процедура оптимизации может считаться компонентом системы автоматизированного проектирования, но более естественно рассматривать эту процедуру отдельно.

Фаза оценки проекта также выигрывает от использования компьютера. Если для оценки проекта нужен прототип, мы можем быстро сконструировать его по заданному проекту при помощи программных пакетов, генерирующих код для машины быстрого прототипирования. Такие пакеты считаются программами для автоматизированной подготовки производства

(САМ). Разумеется, форма прототипа должна быть определена заранее в наборе входных данных. Данные, определяющие форму, получаются в результате геометрического моделирования. Быстрое прототипирование — удобный способ конструирования прототипа, однако еще удобнее пользоваться виртуальным прототипом, который часто называется «цифровой копией» и позволяет получить столь же полезные сведения.

Когда аналитические средства для работы с цифровыми копиями станут достаточно мощными, чтобы давать столь же точные результаты, что и эквивалентные эксперименты на реальных прототипах, цифровые копии начнут вытеснение обычных прототипов. Эта тенденция будет усиливаться по мере совершенствования технологий виртуальной реальности<sup>1</sup>, позволяющих нам ощущать цифровую копию так же, как реальный прототип. Построение цифровой копии называется *виртуальным прототипированием*. Виртуальный прототип может быть создан и в специализированной программе геометрического моделирования

Последняя фаза процесса разработки — подготовка проектной документации. На этом этапе чрезвычайно полезным оказывается использование систем подготовки рабочих чертежей. Способность подобных систем работать с файлами позволяет систематизировать хранение и обеспечить удобство поиска документов. Компьютерные технологии используются и на стадии производства. Процесс производства включает в себя планирование выпуска, проектирование и приобретение новых инструментов, заказ материалов, программирование машин с ЧПУ, контроль качества и упаковку. Компьютерные системы, используемые в этих операциях, могут быть классифицированы как системы автоматизированного производства. Например, программа автоматизированной технологической подготовки (computer aided process planning — САРР) используется на этапе подготовки производства и относится к системам автоматизированного производства (САМ). Как отмечалось выше, подготовка производства с трудом поддается автоматизации, поэтому полностью автоматических систем технологической подготовки в настоящий момент не существует. Однако существует множество хороших программных пакетов, генерирующих код для станков с числовым программным управлением. Станки этого класса позволяют получить деталь нужной формы по данным, хранящимся в компьютере. Они аналогичны машинам для быстрого прототипирования. К системам автоматизированного производства

относят также программные пакеты, управляющие движением роботов при сборке компонентов и перемещении их между операциями, а также пакеты, позволяющие программировать координатно измерительную машину, используемую для проверки продукта.

Итак, вы получили представление о том, каким образом компьютерные технологии используются в операциях, составляющих жизненный цикл продукта, и какие задачи решаются при помощи систем автоматизированного проектирования. Технологии CAD, CAM и CAE будут определены в следующем разделе.

Виртуальная реальность — это технология создания изображений, выглядящих как настоящие объекты. Эта технология позволяет оператору ощущать цифровые объекты и манипулировать ими так же, как настоящими.

## **2. Определение CAD, CAM и CAE**

Согласно предыдущему разделу, *автоматизированное проектирование CAD* представляет собой технологию, состоящую в использовании компьютерных систем для облегчения создания, изменения, анализа и оптимизации проектов. Таким образом, любая программа, работающая с компьютерной графикой, так же как и любое приложение, используемое в инженерных расчетах, относится к системам автоматизированного проектирования. Другими словами, множество средств CAD простирается от геометрических программ для работы с формами до специализированных приложений для анализа и оптимизации. Между этими крайностями уместятся программы для анализа допусков, расчета масс инерционных свойств, моделирования методом конечных элементов и визуализации результатов анализа. Самая основная функция CAD — определение геометрии конструкции (детали механизма, архитектурные элементы, электронные схемы, планы зданий и т. п.), поскольку геометрия определяет все последующие этапы жизненного цикла продукта. Для этой цели обычно используются системы разработки рабочих чертежей и геометрического моделирования. Вот почему эти системы обычно и считаются системами автоматизированного проектирования. Более того, геометрия, определенная в этих системах, может использоваться в качестве основы для дальнейших операций в системах CAE и CAM. Это одно из наиболее значительных преимуществ CAD, позволяющее эко-

номить время и сокращать количество ошибок, связанных с необходимостью определять геометрию конструкции с нуля каждый раз, когда она требуется в расчетах. Можно, следовательно, утверждать, что системы автоматизированной разработки рабочих чертежей и системы геометрического моделирования являются наиболее важными компонентами автоматизированного проектирования.

*Автоматизированное производство — САМ* — это технология, состоящая в использовании компьютерных систем для планирования, управления и контроля операций производства через прямой или косвенный интерфейс с производственными ресурсами предприятия. Одним из наиболее зрелых подходов к автоматизации производства является числовое программное управление (ЧПУ, numerical control — NC). ЧПУ заключается в использовании запрограммированных команд для управления станком, который может шлифовать, резать, фрезеровать, штамповать, изгибать и иными способами превращать заготовки в готовые детали. В наше время компьютеры способны генерировать большие программы для станков с ЧПУ на основании геометрических параметров изделий из базы данных САО и дополнительных сведений, предоставляемых оператором. Исследования в этой области концентрируются а сокращении необходимости вмешательства оператора.

Еще одна важная функция систем автоматизированного производства — программирование роботов, которые могут работать на гибких автоматизированных участках, выбирая и устанавливая инструменты и обрабатываемые детали на станах с ЧПУ. Роботы могут также выполнять свои собственные задачи, например заниматься сваркой, сборкой и переносом оборудования и деталей по цеху. лакирование процессов также постепенно автоматизируется. План процессов может определять последовательность операций по изготовлению устройства от начала и до конца на всем необходимом оборудовании. Хотя полностью автоматизированное планирование процессов, как уже отмечалось, практически невозможно, план обработки конкретной детали вполне может быть сформирован автоматически, если уже имеются планы обработки аналогичных деталей. Для этого была разработана технология группировки, позволяющая объединять схожие детали в семейства. Детали считаются подобными, если они имеют общие производственные особенности (гнезда, пазы, фаски, отверстия и т. д.)- Для ав-

томатического обнаружения схожести деталей необходимо, чтобы база данных САП содержала сведения о таких особенностях. Эта задача осуществляется при помощи объектно-ориентированного моделирования или распознавания элементов.

Вдобавок, компьютер может использоваться для того, чтобы выявлять необходимость заказа исходных материалов и покупных деталей, а также определять их количество исходя из графика производства. Называется такая деятельность планированием технических требований к материалу (*material requirements planning* — MRP). Компьютер может также использоваться для контроля состояния станков в цехе и отправки им соответствующих заданий.

*Автоматизированное конструирование* — CAE) — это технология, состоящая в использовании компьютерных систем для анализа геометрии САД, моделирования и изучения поведения продукта для усовершенствования и оптимизации его конструкции. Средства CAE могут осуществлять множество различных вариантов анализа. Программы для кинематических расчетов, например, способны определять траектории движения и скорости звеньев в механизмах. Программы динамического анализа с большими смещениями могут использоваться для определения нагрузок и смещений в сложных составных устройствах типа автомобилей. Программы верификации и анализа логики и синхронизации имитируют работу сложных электронных цепей.

По всей видимости, из всех методов компьютерного анализа наиболее широко в конструировании используется *метод конечных элементов* (*finite element method FEM*). С его помощью рассчитываются напряжения, деформации, теплообмен, распределение магнитного поля, потоки жидкостей и другие задачи с непрерывными средами, решать которые каким-либо иным методом оказывается просто непрактично. В методе конечных элементов аналитическая модель структуры представляет собой соединение элементов, благодаря чему она разбивается на отдельные части, которые уже могут обрабатываться компьютером.

Как отмечалось ранее, для использования метода конечных элементов нужна абстрактная модель подходящего уровня, а не сама конструкция. Абстрактная модель отличается от конструкции тем, что она формируется путем исключения несущественных деталей и редуцирования размерностей.

Например, трехмерный объект небольшой толщины может быть представлен в виде двумерной оболочки. Модель создается либо в интерактивном режиме, либо автоматически. Готовая абстрактная модель разбивается на конечные элементы, образующие аналитическую модель. Программные средства, позволяющие конструировать абстрактную модель и разбивать ее на конечные элементы, называются *препроцессорами*. Проанализировав каждый элемент, компьютер собирает результаты воедино и представляет их в визуальном формате. Например, области с высоким напряжением могут быть выделены красным цветом. Программные средства, обеспечивающие визуализацию, называются *постпроцессорами*.

Существует множество программных средств для оптимизации конструкций. Хотя средства оптимизации могут быть отнесены к классу CAE, обычно их рассматривают отдельно. Ведутся исследования возможности автоматического определения формы конструкции путем объединения оптимизации и анализа. В этих подходах исходная форма конструкции предполагается простой, как, например, у прямоугольного двумерного объекта, состоящего из небольших элементов различной плотности. Затем выполняется процедура оптимизации, позволяющая определить конкретные значения плотности, позволяющие достичь определенной цели с учетом ограничений на напряжения. Целью часто является минимизация веса. После определения оптимальных значений плотности рассчитывается оптимальная форма объекта. Она получается отбрасыванием элементов с низкими значениями плотности.

Замечательное достоинство методов анализа и оптимизации конструкций заключается в том, что они позволяют конструктору увидеть поведение конечного продукта и выявить возможные ошибки до создания и тестирования реальных прототипов, избежав определенных затрат. Поскольку стоимость конструирования на последних стадиях разработки и производства продукта экспоненциально возрастает, ранняя оптимизация и усовершенствование (возможные только благодаря аналитическим средствам CAE) окупаются значительным снижением сроков и стоимости разработки.

Таким образом, технологии *CAD*, *CAM* и *CAE* заключаются в автоматизации и повышении эффективности конкретных стадий жизненного цикла продукта. Развиваясь независимо, эти системы еще не до конца реа-

лизовали потенциал интеграции проектирования и производства. Для решения этой проблемы была предложена новая технология, получившая название *компьютеризованного интегрированного производства (computer integrated manufacturing — CIM)*. CIM пытается соединить «островки автоматизации» вместе и превратить их в бесперебойно и эффективно работающую систему. CIM подразумевает использование компьютерной базы данных для более эффективного управления всем предприятием, в частности бухгалтерией, планированием, доставкой и другими задачами, а не только проектированием и производством, которые охватывались системами CAD, CAM и CAE. CIM часто называют философией бизнеса, а не компьютерной системой.

### **3. Сценарий интеграции проектирования и производства посредством общей базы данных**

Приведенный ниже сценарий демонстрирует использование систем САП/САМ/ САЕ в рамках всего жизненного цикла продукта для достижения упоминавшихся целей: повышения качества (Q), снижения стоимости (C) и ускорения отгрузки (B). Этот сценарий может показаться несколько упрощенным на фоне современных передовых компьютерных технологий, однако он иллюстрирует направления развития техники. Рассмотрим проектирование и производство шкафа для аудиосистемы, (рис. 2.2).

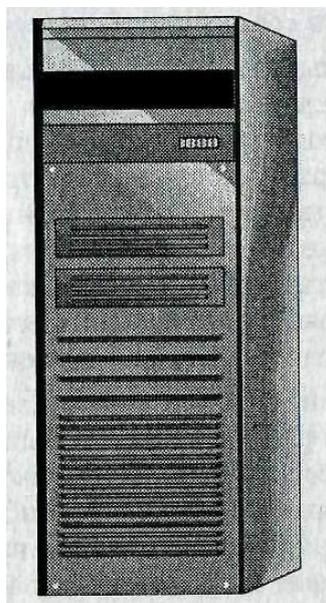


Рис. 2.2. Шкаф для видеосистемы

Предположим, что в технических требованиях для разработчика указано, что шкаф должен иметь четыре полки: одну для проигрывателя компакт-дисков, одну для проигрывателя аудиокассет, одну для радиоприемника и одну для хранения компакт-дисков. Вероятно, разработчик сделает множество набросков конструкции, прежде чем придет к варианту, изображенному на рис. 2. На данном этапе он может пользоваться автоматизированной системой разработки рабочих чертежей (если задача решается в двух измерениях) или системой геометрического моделирования (в случае трех измерений). Концептуальный проект может быть отправлен в отдел маркетинга по электронной почте для получения отзыва. Взаимодействие разработчика с отделом маркетинга может происходить и в реальном времени через объединенные в сеть компьютеры. При наличии подходящего оборудования подобное взаимодействие может быть удобным и продуктивным. Информация о готовом концептуальном проекте сохраняется в базе данных. Туда попадают сведения о конфигурации мебели (в нашем случае — вертикальное хранение компонентов аудиосистемы друг над другом), количестве полок, распределении полок по компонентам и тому подобные данные. Другими словами, все особенности проекта, которые можно увидеть, взглянув на рис. 2, упорядочиваются и помещаются в базу данных с возможностью считывания и изменения в любой последующий момент.

Следующий шаг — определение размеров шкафа. Его габариты должны быть выбраны таким образом, чтобы на каждую полку можно было поставить одну из множества имеющихся на рынке моделей аудиотехники соответствующего класса. Значит, нужно получить сведения об их размерах. Эти сведения можно взять в каталоге или в базе данных производителей или поставщиков. Доступ к базе данных осуществляется аналогично доступу к книгам и их содержанию при подключении к электронной библиотеке. Разработчик может даже скопировать сведения в свою собственную базу данных, если он планирует часто пользоваться ими. Накопление сведений о проекте подобно накоплению форм и файлов при работе с текстовыми процессорами. Форма конструкции, изображенной на рис. 2, должна изменяться в соответствии с полученными сведениями. Затем разработчик должен выбрать материал для шкафа. Он может взять натуральный дуб, сосну,

ДСП, сталь или что-нибудь еще. В нашем случае выбор осуществляется интуитивно или исходя из имеющегося у разработчика опыта. Однако в случае продуктов, рассчитанных на работу в жестких условиях, в частности механических устройств, разработчик обязательно учитывает свойства материалов. На этом этапе также полезна база данных, потому что в ней могут быть сохранены свойства множества материалов. Можно воспользоваться даже экспертной системой, которая выберет материал по свойствам, хранящимся в базе данных. Информация о выбранном материале также помещается в базу. Следующий шаг — определение толщины полок, дверец и боковых стенок. В простейшем случае, который мы рассматриваем, толщина может определяться главным образом эстетическими соображениями. Однако она должна быть по крайней мере достаточной для того, чтобы избежать прогиба под воздействием установленной в шкаф техники. В механических устройствах высокой точности и структурах, рассчитанных на большие нагрузки, такие параметры, как толщина, должны определяться точным расчетом, чтобы избежать деформации. Для расчета деформации структур широко используется метод конечных элементов. Как уже было объяснено, метод конечных элементов применяется к аналитической модели конструкции. В нашем случае аналитическая модель состоит из каркасных сеток, на которые разбивается шкаф, рассматриваемый в приближении листов. Переход к приближению листов может быть выполнен автоматически при помощи *алгоритма преобразования к средним осям*. Элементы оболочки приближения листов также могут генерироваться автоматически. Параметры нагрузки, которые в нашем случае есть просто веса соответствующих устройств, считываются из базы данных точно так же, как и сведения о размерах. Определяя зависимость прогиба полок от их толщины, разработчик может выбрать подходящее значение этого параметра и сохранить его в базе данных. Этот процесс может быть автоматизирован путем интеграции метода конечных элементов с процедурой оптимизации. Аналогичным образом можно определить толщину боковых стенок и дверец, однако сделать это можно и просто из эстетических соображений.

Затем разработчик выбирает метод сборки полок и боковых стенок. В идеале метод также может быть определен из расчета прочности структуры в целом или при помощи экспертной системы, имеющей сведения о методах сборки. После завершения этапов концептуализации проекта, его анализа и оптимизации разработчик переходит к работе над проектной документацией,

описывающей шкаф с точностью до мельчайших подробностей. Чертежи отдельных деталей (полок, дверец и боковых стенок) изготавливаются в системе разработки рабочих

чертежей. На этом этапе разработчик может добавить некоторые эстетические детали, например декоративные элементы на дверцах и боковых стенках. Детальные чертежи помещаются в базу данных для использования в процессе производства. Изготовление шкафа осуществляется в следующем порядке. Форма каждой детали наносится на необработанный материал (в нашем случае дерево) и вырезается пилой. Количество отходов можно снизить, располагая детали на кусках дерева оптимально. Разработчик может испытывать разные варианты размещения на экране компьютера до тех пор, пока не будет найдена конфигурация с минимальным количеством отходов. Компьютерная программа может помочь в этой работе, рассчитывая количество отходов для каждой конфигурации. Программа более высокого уровня может самостоятельно определить наиболее экономичное размещение деталей на заготовке. В любом случае конечная конфигурация сохраняется в компьютере и используется для расчета траектории движения пилы станка с числовым программным управлением. Более того, программные средства позволяют разработать зажимы и крепления для процедуры выпиливания, а также запрограммировать системы передачи материала. Эти системы могут быть как простыми конвейерами, так и сложными роботами, передающими необработанный материал на распилку и забирающими готовые детали. Подготовленные детали должны быть собраны вместе. Процесс сборки также может выполняться роботами, которые программируются автоматически на основании описания конечного продукта и его деталей, хранящегося в базе данных. Одновременно проектируются зажимы и крепления для автоматизированной сборки. Наконец, робот может быть запрограммирован на покраску шкафа после сборки. В настоящее время зажимы и крепления для сборки проектируются или выбираются планировщиком процессов, а программирование роботов осуществляется в интерактивном режиме путем перемещения рабочего органа робота вручную.

Общий вид получившегося сценария показан на рис. 2.3, из которого видно, каким образом база данных позволяет интегрировать системы CAD, CAE и CAM, что и является конечной целью CIM



Рис. 2.3. Интеграция CAD, CAM и CAE через базу данных

### Лекция № 3

#### Компоненты САПР

Для реализации компьютерно-ориентированного подхода к проектированию и производству, описанному с главе 1, нужно специальное аппаратное и программное обеспечение. Ключевым аспектом является интерактивное управление формой, поэтому неудивительно, что аппаратное и программное обеспечение для интерактивного манипулирования формами относится к числу основных компонентов, составляющих системы САВ/САМ/САЕ. Графические устройства и периферийные устройства ввода-вывода вместе с обычным вычислительным модулем составляют аппаратное обеспечение систем САВ/САМ/САЕ (рис. 3.1). Ключевыми программными компонентами являются пакеты, манипулирующие формами или анализирующие их под управлением пользователя в двух или в трех измерениях, одновременно обновляя базу данных. В последующих разделах эти аппаратные и программные компоненты рассматриваются подробно.



Рис.3.1 Компоненты системы CAD/CAM/CAE

## 1. Аппаратное обеспечение

Графическое устройство состоит из дисплейного процессора, устройства отображения, или дисплейного устройства (называемого монитором), и одного или нескольких устройств ввода. Дисплей (монитор) представляет собой экран, на который выводится графическое изображение, однако вывод конкретного изо-

Изображения на экран выполняется дисплейным, процессором, другими словами, дисплейный процессор получает сигналы, которыми кодируются графические команды, генерирует электронные пучки и направляет их в нужное место монитора, порождая желаемое изображение.

В состав графического устройства обычно входит одно или несколько устройств ввода. Помимо клавиатуры к ним относятся мышь, сиейсбол<sup>1</sup> и цифровой планшет с пером и роликом (рис. 3.2). Эти устройства ввода призваны способствовать интерактивному манипулированию формами, давая пользователю возможность вводить графические данные в компьютер непосредственно. Каждое графическое устройство обычно подключается к устройствам вывода, например к плоттеру или цветному лазерному принтеру .

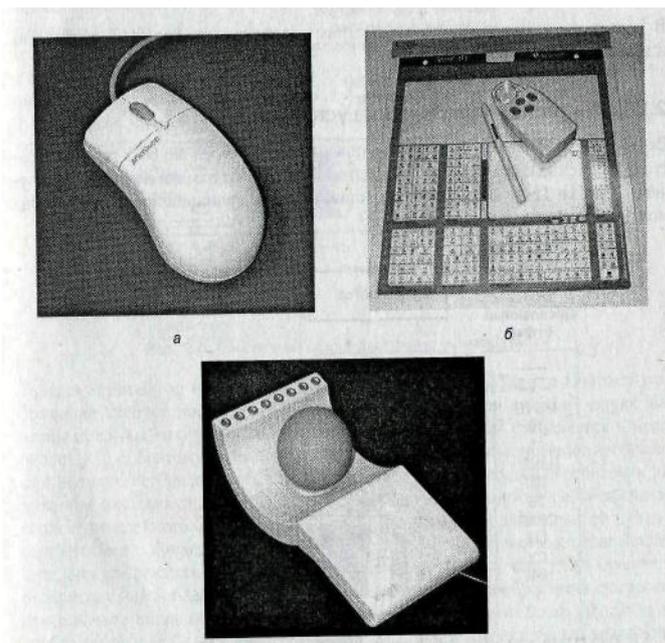


Рис.3.2. Примеры устройств ввода

Эти устройства могут использоваться несколькими графическими устройствами совместно. Устройства вывода позволяют вывести любое изображение на бумагу

### ***1.1. Векторные графические устройства***

Векторные графические устройства, появившиеся в середине 60-х гг. XX в., состоят из дисплейного процессора, дисплейного буфера памяти и электронно-лучевой трубки (рис. 3.3). Основные принципы их функционирования вкратце можно описать следующим образом

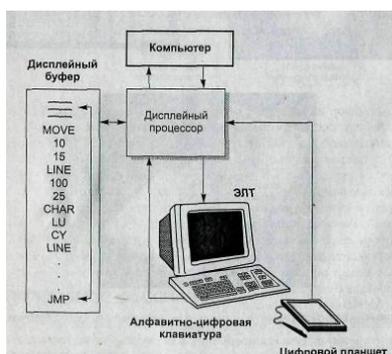


Рис. 3.3. Компоненты векторно графического устройства

Дисплейный процессор считывает *дисплейный файл* (*display III*), который представляет собой последовательность передаваемых приложением кодов, соответствующих графическим командам. Дисплейный файл хранится в разделе памяти, который называется *дисплейным буфером* (*сИзрШу Би//ег*). Дисплейный процессор осуществляет также загрузку дисплейного файла в дисплейный буфер. После этого дисплейный процессор формирует необходимые напряжения на вертикальных и горизонтальных парах отклоняющих пластин таким образом, что электрон, вылетающий с катода, попадает в нужное место внутренней поверхности передней стенки электронно-лучевой трубки (рис. 3.4). Это место соответствует одной точке создаваемого изображения. Внутренняя поверхность трубки покрыта люминофором, поэтому в той точке, куда попал электрон, возникает короткая вспышка света. Электронный пучок движется по экрану и формирует изображение в соответствии с командами дисплейного файла.

Однако люминофор излучает свет очень недолго, а затем гаснет. Поэтому изображение требует постоянного перерисовывания с очень высокой скоростью, чтобы пользователю не досаждало мерцание. Изображение сохраняется в мозге около  $1/30$  с, поэтому пользователь не будет замечать исчезновения изображения, если его перерисовывать с периодом менее  $1/30$  с. Процесс перерисовки называется *обновлением* (*ге/гезИ*), которое обеспечивается повторяющимся считыванием дисплейного буфера сверху вниз и направлением электронного пучка в соответствии с командами из буфера. Дисплейный буфер используется исключительно для реализации обновления. В зависимости от сложности изображения перерисовка может занимать и более  $1/30$  с. Тогда часть изображения, построенная в начале цикла обновления, успеет исчезнуть, в то время как другая часть изображения еще будет прорисовываться

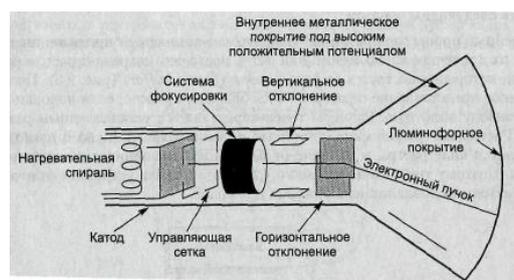


Рис. 3.4. Поперечный разрез электронно - лучевой трубки

Возникнет эффект мерцания картинка в целом. Мерцание изображения и высокая стоимость — основные недостатки устройств векторной графики.

Есть у них и преимущества. Напряжения, подаваемые на вертикальные и горизонтальные отклоняющие пластины, можно контролировать с любой заданной точностью, что обеспечивает любое желаемое разрешение. Поэтому дисплейное

устройство может иметь высокое разрешение (например, 4096x4096), а прямые линии получаются действительно прямыми, а не ступенчатыми. К тому же векторные устройства позволяют отображать динамическую анимацию. Динамика обеспечивается благодаря изменению содержимого дисплейного буфера по мере „ того, как дисплейный процессор занимается обновлением. Содержимое буфера изменяется графическими командами, передаваемыми управляющим приложением, в данном случае — программой анимации.

## ***1.2. Растровые графические устройства***

Растровые графические устройства появились в середине 70-х гг. XX в. в результате широкого распространения телевизионных технологий. С тех пор они стали основным видом графических устройств благодаря высокому соотношению «качество — цена». Основные принципы их функционирования коротко можно описать следующим образом.

Дисплейный процессор принимает графические команды от приложения, преобразует их в точечное изображение, или растр, после чего сохраняет растр в разделе памяти, который называется *буфером кадра* (*/gate Бу//ег*) (рис. 3.5). Получить наглядное представление о растровом изображении можно, если подойти к телевизору вплотную. Размеры точек определяются установленным разрешением. Растровые графические устройства должны хранить в своей памяти изображение в виде растра, в отличие от векторных, хранящих лишь дисплейные файлы. Поэтому требования к памяти у этих двух видов устройств отличаются, как и методы обновления изображения на экране.



Рис. 3.5. Компоненты растрового графического устройства

Когда дисплейный процессор генерирует растровое изображение и сохраняет его в буфере кадра, он параллельно считывает содержимое всего буфера и направляет электронные пучки на дисплей, воспроизводя на нем картинку, хранящуюся в буфере. На внутренней поверхности электронно-лучевой трубки может быть столько точек люминофора, сколько точек описывается буфером кадра. Электронный пучок направляется на точки, соответствующие точкам растрового изображения. Время свечения люминофора в растровых устройствах так же коротко, как и в векторных, поэтому необходимо регулярное обновление изображения. Единственное отличие — порядок движения электронного пучка при обновлении. Пучок пробегает по экрану слева направо, переходя со строки на строку в направлении сверху вниз, как показано на рис. 3.6

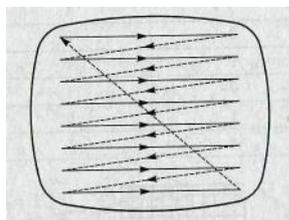


Рис. 3.6. Развертка растрового изображения

Когда электронный пучок направляется на точку люминофора, соответствующую точке изображения, он включается, возбуждая свечение лю-

минофора. Время обновления остается постоянным независимо от сложности воспроизводимого изображения. Время обновления определяется, таким образом, временем сканирования всех строк развертки от верхней до нижней, и, как правило, составляет  $1/30$  с для обычных телевизоров или  $1/60$  с для качественных растровых графических устройств.

Однако буфер кадра в растровых устройствах требует гораздо больше памяти, чем дисплейный буфер в векторных графических устройствах. Растровое изображение в буфере кадра может содержать сведения о цвете, если каждой точке (пикселу) будет соответствовать не один, бит; а несколько. Рассмотрим пример с тремя битами на каждый пиксел. Буфер кадра может быть представлен тремя плоскостями (рис. 2.8), каждая из которых содержит по одному биту для каждого пиксела. Говорят, что в таком случае буфер кадра содержит 3-битовые плоскости. В случае трехбитового представления цвета первый разряд может использоваться для включения или выключения красного, второй — зеленого, а третий — синего цвета. Так получается восемь цветов, показанных в таблице, которые могут быть одновременно выведены на экран графического устройства. ЦАПы, показанные на рис. 3.7, — это цифроаналоговые преобразователи, выдающие аналоговый сигнал, управляющий электронной пушкой определенного цвета на основании значений битов соответствующей плоскости. Цена на микросхемы памяти падает, поэтому на рынке в настоящее время доминируют графические устройства с 24-битовыми плоскостями (по восемь битов на каждый из основных цветов). В таких устройствах каждый цвет может иметь 256 градаций ( $2^8$ ), а всего возможно одновременно отобразить  $16777216$  ( $2^{24}$ ) цветов. Хранение конкретного цвета в буфере кадра иллюстрирует рис.9. ЦАПы на рис. 9 решают ту же задачу, что и на рис. 3.8, но работают с 8 разрядами вместо одного.

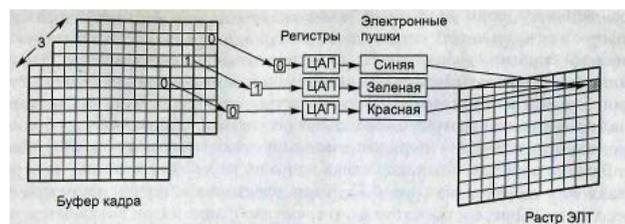


Рис. 3.7. Пример трехбитовых плоскостей

	Красный	Зеленый	Синий
Черный	0	0	0
Красный	1	0	0
Зеленый	0	1	0
Синий	0	0	1
Желтый	1	1	0
Голубой	0	1	1
Малиновый	1	0	1
Белый	1	1	1

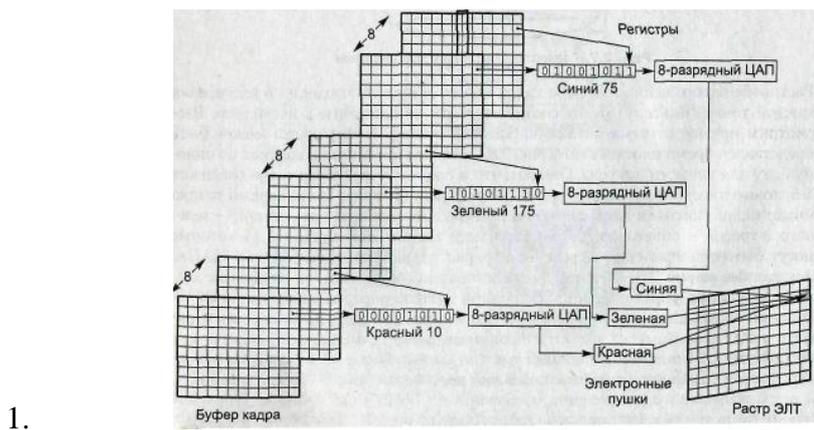


Рис. 3.8. Задание цвета 24 – битовых плоскостей

Точка монитора, соответствующая точке буфера кадра, будет светиться определенным цветом, как в обычном цветном телевизоре. Формирование цвета происходит следующим образом. Внутренняя поверхность трубки монитора покрыта люминофором трех разных типов. Люминофор первого типа светится красным, второго — синим, а третьего — зеленым. Точки люминофора расположены в вершинах равносторонних треугольников (рис. 3.9). Каждый треугольник соответствует одной точке в буфере кадров. Энергия электронного пучка, падающего на люминофор из пушки соответствующего цвета, пропорциональна аналоговому сигналу от ЦАП (см. рис. 3.7 и 3.8). Следовательно, интенсивность излучения определенного цвета также пропорциональна сигналу от ЦАП. Точки разных цветов расположены в вершинах треугольника, но для человека они сливаются

воедино и образуют цвет, задаваемый значением, хранящимся в .буфере кадра. Электронный пучок может попадать и на люминофор другого цвета. Например, пушка, предназначенная для красного люминофора, может попасть пучком на зеленый люминофор. Для предотвращения этого эффекта перед слоем люминофора помещается *тневая маска* (рис. 3.10). Тневая маска гарантирует, что электронные пучки попадают только на люминофор соответствующего цвета.

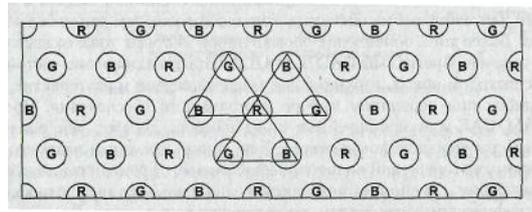


Рис. 3.9. Распределение точек люминофора

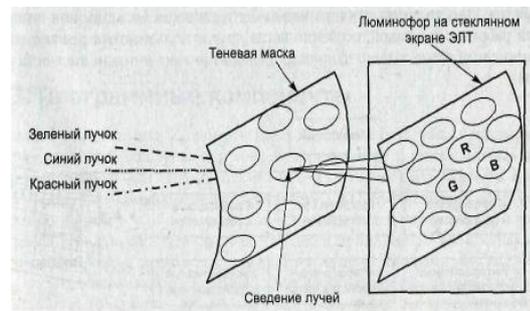


Рис. 3.10 Тневая маска

## 2. Конфигурация аппаратных средств

Графические устройства, описанные, в предыдущем разделе, редко используются поодиночке. Чаще всего они объединяются в кластер того или иного рода, рассчитанный на обслуживание множества пользователей. Существует три основных варианта конфигурации такого кластера.

Первая конфигурация состоит из мейнфрейма (татггате) и множества графических устройств (рис. 3.11). Графические устройства подключаются к мейнфрейму точно так же, как алфавитно-цифровые терминалы в обыч-

ных вычислительных центрах. К нему же подключаются и устройства вывода, такие как принтеры и плоттеры. Поскольку такая конфигурация может рассматриваться как естественное расширение существующей вычислительной среды, она с готовностью принималась большинством крупных компаний, у которых уже были мейнфреймы. Этот подход до сих пор используется производителями автомобилей и кораблей, у которых имеются огромные базы данных, обрабатываемые централизованно. Однако он обладает некоторыми недостатками. Он требует больших начальных вложений в аппаратное и программное обеспечение, да и обслуживание эксплуатируемой системы тоже стоит недешево. Обслуживание мейнфрейма всегда включает в себя расширение системной памяти и жесткого диска, что для мейнфрейма обходится гораздо дороже, чем для небольших компьютеров. Более того, обновление операционной системы тоже оказывается непростой задачей. Программы САВ/САМ/САЕ требуют довольно частой замены в связи с выходом новых, гораздо более мощных версий и альтернатив, а также из-за ошибок при первичном выборе программного обеспечения. Программы САП/САМ/САЕ для мейнфреймов стоят намного дороже, чем аналогичные программы для меньших компьютеров. Еще одним серьезным недостатком централизованных вычислений является нестабильность времени отклика системы. В конфигурации с мейнфреймом приложения пользователей, относящиеся к разным графическим устройствам, конкурируют друг с другом за вычислительные ресурсы мейнфрейма. Поэтому время отклика для любого конкретного графического устройства зависит от того, какие задачи были запущены с другого устройства. Иногда время отклика может быть слишком большим для интерактивной работы с графикой, особенно когда другие пользователи решают сложные вычислительные задачи, например проводят анализ методом конечных элементов.



Рис. 3.11. Интерфейсы с графическими устройствами

Вторая конфигурация состоит из автоматизированных рабочих мест проектировщика (рабочих станций — workstations), объединенных в сеть (рис. 3.12). К той же сети подключаются устройства вывода — плоттеры и принтеры. Рабочая станция — это графическое устройство с собственными вычислительными ресурсами. Такой подход в настоящее время используется очень широко, потому что в области технологий изготовления рабочих станций прогресс идет огромными темпами, да и вообще имеется тенденция к распределению вычислений. Производительность рабочих станций удваивается каждый год при сохранении I цены. Данный подход обладает и другими преимуществами. Пользователь может работать с любой станцией сети, выбирая ее в соответствии со своей задачей, причем системные ресурсы не будут зависеть от задач других пользователей. Еще одно преимущество — отсутствие необходимости в крупных первичных вложениях. Количество рабочих станций и программных пакетов может увеличиваясь постепенно, по мере роста потребности в ресурсах CAD/CAM/CAE. Это очень выгодно, потому что стоимость оборудования постоянно падает.



Рис. 3.12. Рабочие станции объединенные в сеть

Третья конфигурация аналогична второй за тем исключением, что вместо рабочих станций используются персональные компьютеры с операционными системами Windows 95 и NT. Конфигурации на базе персональных компьютеров популярны в небольших компаниях, особенно если выпускаемые продукты состоят из небольшого количества деталей ограниченной сложности, а также в компаниях, использующих системы CAD/CAM/CAE главным образом для построения чертежей. По мере того как различие между персональными компьютерами и рабочими станциями сглаживается, стирается и различие между вторым и третьим типами конфигурации.

### 3. Инструментальные средства проектирования

#### 3.1 Системы проектирования

Весь спектр проблем, связанных с проектной деятельностью (графических, аналитических, экономических, эргономических, эстетических и др.), решается в настоящее время с использованием эффективных компьютерных технологий и систем автоматизированного проектирования (САПР).

Использование САПР позволяет в значительной мере сократить продолжительность проектирования, обеспечивая [1]:

быстрое выполнение чертежей;

высокую точность и качество выполнения чертежей, т.е. на чертеже, построенном с помощью программных средств, любая точка определена точно, а для более детального просмотра элементов чертежа можно увеличить любую его часть. Кроме того, в САПР имеются различные специальные средства выполнения чертежей, например программные средства любой САПР позволяют быстро стереть лишние линии без каких-либо последствий для конечного чертежа;

возможность многократного использования чертежа, т.е. построение всего чертежа или его части можно сохранить и использовать для последующего проектирования;

ускорение расчетов и анализа, требуемых при проектировании. Существующее разнообразное программное обеспечение позволяет выполнять практически все проектные расчеты;

сокращение затрат на исследование и усовершенствование прототипов объектов. Средства имитации и анализа, включенные в САПР, позволяют резко снизить затраты времени и средств на эти дорогостоящие этапы процесса проектирования;

интеграцию проектирования с другими видами деятельности. Интегрированная вычислительная сеть с высококачественными средствами коммуникации обеспечивает САПР более тесное взаимодействие с другими инженерными подразделениями.

Конструктор должен досконально знать правила оформления чертежно-графической документации (ЕСКД), свободно владеть программ-

ными средствами необходимыми для работы, и иметь представление о составе и возможностях своего автоматизированного рабочего места (АРМ).

Различают системы проектирования трех уровней: высокого, среднего и

твердотельных моделей с привлечением мощных конструкторско-технологических библиотек и использованием современного математического аппарат; для проведения необходимых расчетов. Кроме того, эти системы позволяют (помощью средств анимации имитировать перемещение в пространстве рабочих органов изделия (например, манипуляторов робота), а также отслеживают траекторию движения инструмента при разработке и контроле технологического процесса изготовления спроектированного изделия. Все это дела трехмерное моделирование неотъемлемой частью совместной работы АСТПП. К системам высокого уровня относятся Pro/ENGINEER, (США) EUCLID (Франция), T-FLEX CAD, СПРУТ (Россия); к систем среднего уровня **Mechanical Desktop** (фирмы **Auto-DESK**), **SolidWorks96** (фирмы SolidWorks) и др.

Системы низкого уровня — это графические редакторы, предназначены для автоматизации инженерно-графических работ. Совместно с компьютером и монитором они представляют собой «электронный кульман», т.е. хороший инструмент для выполнения конструкторской документации. Системы низкого уровня называют двухмерными. К ним относятся AutoCAD, MiniCAD Microsoft Visio (США), КОМПАС (фирмы АСКОН, Россия), ElectricS (ООО РОЗМЫСЕЛ, Россия), ERLAN (Германия).

Системы двухмерного проектирования распознают геометрические формы, определяемые точками, прямыми или кривыми линиями, только на плоскости. Каждый вид некоторого объекта (главный, сверху и т.д.) такая система может выполнить лишь как отдельную фигуру, рассматриваемую вне связи с любыми другими видами.

Двухмерные (проекционные) системы более примитивны, чем трехмерные, однако они довольно широко распространены, и их сравнительно не большая стоимость является существенным фактором при выборе систем проектирования. С помощью двухмерных систем создается большинство конструкторских документов. Все команды любой двухмерной системы (или графического редактора) подразделяются на три вида: команды черчения; команды редактирования; команды нанесения размеров, условных обозначения и текста (оформления чертежа).

### 3.2. Основные методы выполнения инженерно-графических работ

Процесс проектирования можно разделить на ряд этапов или видов деятельности, причем порядок их описания не имеет значения, поскольку на практике постоянно происходит переход от одного вида деятельности к другому без очевидных приоритетов. В основном можно выделить следующие виды деятельности:

*создание* — выполнение проекционных чертежей новых изделий, которые пока не существуют;

*редактирование* — внесение изменений в разрабатываемые чертежи изделия по мере их возникновения;

данных (например, чертежей прототипов изделий, расчетов и т.п.);

*поиск* — работа с архивами (знакомство с уже существующими решениями, историей видоизменения изделия и т.д.), круг которых, как правило ограничивается прототипами изделий конкретной отрасли.

Перечисленные виды деятельности поддаются автоматизации благодаря современным программным продуктам, разрабатываемым различными фирмами.

Системы Microsoft Visio 2002, AutoCAD, ElectricCS предназначены для создания и редактирования электрических схем и текстовой документации соблюдением требований стандартов. В их состав, как правило, входят графический редактор схем, модуль логической обработки, система управления базой электрических устройств, базы данных, библиотеки расчетов, генератор отчетов и система управления проектом. Системы проектирования позволяют вводить и редактировать принципиальные электрические схемы, создавать редактировать и использовать базу электрических устройств, осуществлять трассировку проводов, определять внешние трассы, подбирать клеммные блоки создавать схемы подключений и соединений, настраивать и создавать форм! отчетов и выполнять по ним конструкторскую документацию. С помощью любого из перечисленных программных средств автоматизации чертеж строится на основе базовых графических элементов (графических примитивов): точек отрезков, окружностей и кривых. Метод построения каждого отдельного чертежа в большинстве случаев зависит от требуемой точности.

### 3.3. Анализ программного обеспечения

Проведем анализ программного обеспечения систем Microsoft Visio AutoCAD, ElectricCS и покажем их основные достоинства и недостатки.

*Требования к вычислительной среде.* Для нормального функционирования рабочих мест, на которых устанавливаются Microsoft Visio и AutoCAD), должны удовлетворять следующим требованиям:

наличие операционной системы Windows 98, 2000, NT, XP;

соблюдение необходимых минимальных требований конфигурации компьютера (процессор Pentium с частотой не меньше 166 МГц, оперативная память 64 Мбайт жесткий диск 1 Гбайт).

Для установки ElectricCS требуется наличие графического пакета AutoCAD и редактор MS Word для подготовки отчетов. Требования к рабочему месту соответствуют требованиям AutoCAD

При использовании систем AutoCAD 2002 и ElectricCS перечисленные характеристики компьютера должны быть увеличены примерно в два раза.

*Установка систем.* Установка систем Microsoft Visio и AutoCAD выполняется специальной программой — мастером. По умолчанию предлагается устанавливать систему в папку c:\Program File\. Microsoft Visio или AutoCAD

При установке системы ElectricCS копируются файлы программы в указанный каталог; устанавливаются BDE, программа просмотра документа!] Acrobat Reader и драйвер электронного ключа HASP HINSTALL.EXE;

После установки этих систем на рабочем столе появляются ярлыки, хорошая настройка которых обеспечивает дополнительные возможности.

*Возможности конструирования.* Программы Microsoft Visio, AutoCAD, ElectricCS предоставляют удобную объектную технологию, позволяющую быстро и легко создавать технические рисунки и диаграммы, а следовательно, широко используются. Каждые несколько лет появляются их новые версии. Описания программного обеспечения Microsoft Visio и AutoCAD приводятся, для английских версий, однако пользователи, имеющие русифицированные версии, легко смогут в них разобраться.

Русификация программ Microsoft Visio и AutoCAD касается только команд меню, поэтому русификация версии обеспечивает очень незначительные преимущества.

Microsoft Visio позволяет создавать схематические чертежи, используемые при разработке технологических процессов в электротехнике и машиностроении. Элементы схем соответствуют ГОСТам

Каждая очередная модифицированная версия совершеннее предшествующей версии. При этом от версии к версии сохраняется возможность запуска команд из диалоговой строки. Программа Microsoft Visio 2002 способна открывать файлы, созданные в более ранних ее версиях, и сохраняет файл в формате, который совместим с любыми приложениями.

Усовершенствованная версия Microsoft Visio 2002 автоматизации проектирования обладает следующими возможностями:

- импорт и экспорт .dwg - файлов, вставка и конвертация их с использованием специального дизайн-конвертера;

- чтение и запись dwg - файлов. АШойевК АЩоСАО и dwg - файлов . из ВепОеу 8у81ет8' М1сго5>Шюп;

- легкое и быстрое размещение форм, так как они фиксируются в основной В\УО-геометрии;

- быстрое изменение размера окна вокруг рисунка в формате .й\у\$ или ,д\$п;

- управление видимостью, цветом, толщиной линий изображений объектов и конвертация уровней в формы.

В подразд. 4.2 приведены только самые необходимые и часто используемые команды для РгоГежюпа!.

В М1сго8ой У18Ю возможно создание многостраничных документов. Если схема, создается в формате А1, а принтер печатает только А4, то Мюгозой У1зю сама разобьет изображение на требуемые части, которые следует распечатать и склеить.

Также к плюсам данного редактора можно отнести большое число форматов, в которых можно сохранять изображения, — от Ътр до АйоЪе ШизггаЮг.

Система АитоСАО свои документы сохраняет в файлах формата О\УС, который различен для разных ее версий, но каждая последующая версия совместима по формату с предыдущими. Файл формата АшоСАВ 2000 Вгадагщ не может напрямую читаться более ранними версиями системы (Ю4, К13 и Т.Д.)>4 поэтому для передачи в другую версию данной системы следует использовать команду 5ауеа5 (Сохранить как).

ды \^ЫюсК (Пблок) и 1пкеП (Вставить). Возможен также экспорт восьми фс матов: Me1aП1e (\*.»тС), АС18 (\*.ка1), и\*Ьо§гарЬу (\*.з11), Епсар\$и1а1ес! Р8 (\*.е{ рХХ Ех1гас1 (\*.йхх), ВИтар (\*.Ътр), ЗО Зшёю (\*.ЗЙ5), В1юсК (\*.а»§) — и и порт трех форматов: Me(aП1e (\*.»уптГ), АС18 (\*.5а1), ЗВ 81идю (\*.3с)5).

В АШоСАП можно создавать схемы в форматах от А4 до А]. При этом предусмотрена возможность распечатывать чертежи как на принтере, так и на плоттере

Так как система Е1ес1пС5 устанавливается на базе Аи1юСАВ ее возможности идентичны рассмотренным.

Во всех описанных программных средствах имеется готовая библиотека электротехнических элементов.

Так как М1сго8ой и АиюСАО — универсальные инструменты для проектирования, набор электротехнических элементов в них ограничен, однако пользователи легко могут пополнить библиотеку недостающими элементами и сохранить их как шаблоны для дальнейшего использования.

Так как Е1ес1пС3 предназначена для автоматизированного проектирования электротехнических схем, набор электротехнических элементов в пакете программ максимальный. При этом не рекомендуется использовать папку б] библиотеки условно-графических обозначений (УГО), созданной по умолчанию при установке программы, так как разрабатываемые файлы УГО могут 51 утеряны. Необходимо создать свою папку для библиотеки и скопировать в 1 библиотеку УГО.

## **Лекция №4**

### **Компьютеризация конструкторского проектирования**

#### **1. Функциональная схема проектирования**

Согласно наиболее обобщенной модели системы, модели «черного ящика», система автоматизированного конструирования изделий представляет собой средство преобразования входной информации в выходную (рис. 4.1).

Процесс разработки проекта изделия складывается из двух основных этапов: этапа проектирования, на котором изделие представляется как формальная система с оформлением соответствующих схем и эскизной проектной документации, и собственно конструирования с формированием данных общих видов и рабочей документации. Входной информацией этого процесса является дескриптивное описание проектируемого объекта, которое обычно содержится в техническом задании. Выходная информация в соответствии с функциональным назначением системы определяет конструктивное описание

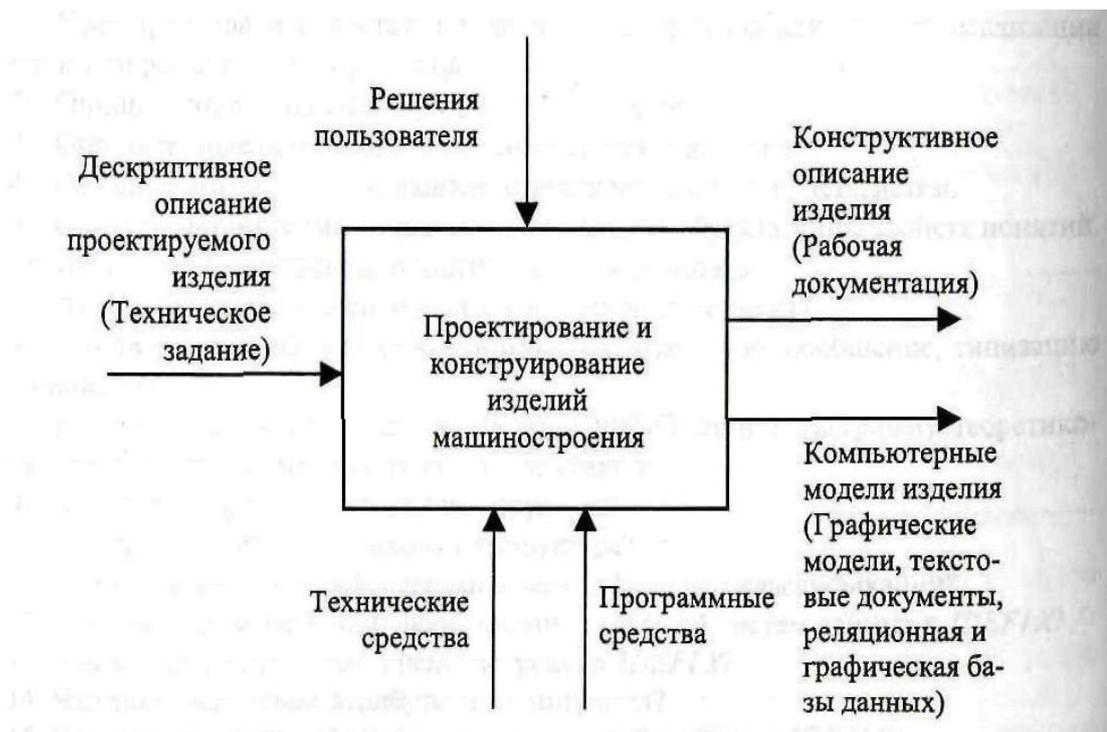


Рис. 4.1. Функциональная модель системы автоматизированного конструирования

проектируемого объекта, общепринятой формой представления которого является проектная и конструкторская документация, а компьютерной формой — графические модели, текстовые документы, реляционная и графическая базы данных. Сама система автоматизированного конструирования реализует с помощью технических и программных средств вычислительной техники процесс преобразования входной информации в выходную. Управляет процессом преобразования информации пользователь. Отсюда следует, что первыми шагами системного анализа данной проблемной ситуации

является системологическое исследование двух основных компонент: проектируемых объектов и процессов проектирования. Исследование первых двух компонент относится к проблематике анализа проектируемых объектов и форм их представления, а последняя — к проблематике анализа собственно проектных действий.

## 2. Принципы методологии проектирования

Любая научная дисциплина должна достаточно четко определить предмет своих исследований, основные понятия и принципы.

Методология проектирования объединяет два понятия: методология и проектирование. *Методология* вообще учение о структуре логической организации, методах и средствах деятельности. Под *проектированием*, как уже отмечалось, понимается процесс составления описания, необходимого для создания еще не существующего объекта. На этой основе методологию проектирования можно определить как учение о структуре, логической организации, методах и средствах составления описания, необходимого для создания еще не существующего объекта.

Уточним приведенное определение в части понятия проектирования. ГОСТ 22487 — 77 указывает, что описание составляется путем преобразования первичного описания, оптимизации заданных характеристик объекта или алгоритма его функционирования, устранения некорректности первичного описания и последовательного его представления (при необходимости) на 1 различных языках. В определении нужно дополнительно отразить целенаправленность проектирования на удовлетворение определенной потребности и особые его процедуры, предшествующие составлению описания и относящиеся к творческим актам поиска и принятия решений.

С учетом дополнений под *методологией проектирования* будем понимать учение о структуре, логической организации, методах и средствах поиска и принятия решений о принципе действия и составе еще не существующего объекта, наилучшим образом удовлетворяющего определенным потребностям, а также составление описания, необходимого для его создания в заданных условиях.

Остановимся на некоторых основных понятиях в соответствии с ГОСТ 22487—77.

**Проектное решение** — это промежуточное или конечное описание объекта проектирования, необходимое и достаточное для рассмотрения и определения дальнейшего направления или окончания проектирования.

**Алгоритм проектирования** — совокупность предписаний, необходимых для выполнения проектирования.

**Язык проектирования** — язык, предназначенный для представления и преобразования описаний при проектировании.

**Проектная процедура** — совокупность действий, выполнение которых оканчивается проектным решением.

**Проектная операция** — действие или совокупность действий, составляющих часть проектной процедуры, алгоритм которых остается неизменным для ряда проектных процедур.

Охарактеризуем особенность современных методов проектирования.

Сложность технических объектов, состоящих из многочисленных, по-разному взаимодействующих друг с другом и окружающей средой подсистем и объектов, требует системного подхода, т. е., по выражению В. И. Ленина, охвата всех их сторон, всех связей и «опосредований». При этом вся человеческая практика должна войти в полное «определение» объекта проектирования и как критерий истины, и как практический определитель связи с тем, что нужно человеку.

Сложность современных объектов вызывает и сложность задачи проектирования. Они не могут быть решены сразу прямым замыканием входной информации на постоянную концептуальную модель действительности, а требуют развернутого во времени сложного информационного поиска. В этих условиях общая задача распадается на подзадачи, т. е. происходит ее декомпозиция, о чем говорилось в предыдущем разделе.

Множественность путей достижения целей проектирования требует рассмотрения не одного, а многих вариантов технического решения, к каждому из которых применяются определенные методы анализа и оценки. Повторное применение методов или алгоритмов проектирования характеризует еще одну его особенность, названную итеративностью.

Современные методы проектирования должны быть ориентированы на широкое использование ЭВМ, не исключая человека при решении наиболее сложных и творческих задач. Такую особенность называют эрга-

тичностью, подразумевая разумное сочетание формализованных (машинных) и неформализованных (человеческих) процедур в процессе проектирования.

Сформулируем основные задачи методологии проектирования с учетом приведенных особенностей изучаемых ею методов.

Декомпозиция требует логической схемы последовательности действий, наилучшим образом организующей процесс проектирования.

Построение такой схемы будем считать первой задачей методологии проектирования.

Стремление к широкому использованию ЭВМ требует формализации процедур, а это, в свою очередь, требует составления, математической модели как процесса, так и объекта проектирования.

Разработка математических моделей составляет вторую, а методы и алгоритмы выполнения проектных процедур и операций — третью задачу методологии.

И еще об одной задаче. Она не вытекает непосредственно из всего ранее изложенного, стоит как бы в стороне, но не теряет от этого своей важности. Речь идет о выборе стадий разработки объекта проектирования.

Известно, что ЕСКД предусматривает следующие стадии: техническое задание; техническое предложение; эскизный проект; технический проект; рабочая документация. ГОСТ не обязывает выполнение всех стадий. В практике каждой проектной организации установилась традиционная стадийность. Однако она зачастую не отвечает возросшей сложности объектов проектирования. Выбор стадий разработки должен быть более гибким и устанавливаться обоснованно, а не традиционно.

Чаще всего выполняют техническое задание, технический проект и рабочую документацию, опуская техническое предложение и эскизный проект. К чему это может привести? Невыполнение работ, предусмотренных техническим предложением, <sup>a</sup> именно технического и технико-экономического обоснования Целесообразности проектирования, выбора различных вариантов возможных решений, сравнительной оценки решений может привести к тому, что в основу разработки ляжет не лучшее техническое решение. Невыполнение эскизного проекта может привести к выбору неоптимальных параметров объекта.

Стадии разработки необходимо выбирать с учетом сложности объекта, степени его новизны, последствий возможных ошибок при проектировании.

Для образности представим: все, что нужно человеку, уже создано и находится на складе. Содержащиеся в нем объекты **классифицированы** по признакам:

функциональным, конструктивным, качественным и, др. Обращаясь на склад, человек должен осознать свои цели, сопоставить их с признаками и по ним отыскать необходимый ему объект. Можно предположить, что таких объектов окажется более одного (отсутствие объекта исключается по начальным условиям). В этом случае человек отбирает такой из них, который в наибольшей степени отвечает поставленным целям. При этом он руководствуется некоторой шкалой оценок.

Распространив такой поведенческий акт на проектирование, определим его основные компоненты:

- $A$  — множество целей;
- $P$  — множество признаков;
- $X$  — множество технических решений;
- $V$  — множество оценок.

На рис. 4.2 показана схема основных компонентов проектирования. Она представляет собой граф с вершинами, означающими элементы множеств целей, признаков, технических решений и оценок, и с ребрами, отображающими отношения между элементами.

Пользуясь языком теории множеств, проектирование технической системы можно связать с отображением на множество оценок срезом произведения бинарных отношений множества целей и множества признаков; множества признаков и множества технических решений.

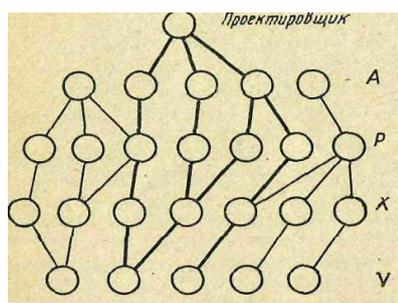


Рис. 4.2 Схема основных компонентов проектирования

### 3. Процедурная модель проектирования

Логическая схема проектирования представлена процедурной моделью. Она реализует системный подход к этому процессу и является результатом обобщения целого ряда работ. В основе процедурной модели проектирования лежат этапы, характерные для трудовой деятельности с отсроченной реализацией. Она дает наглядное представление об основных процедурах и операциях проектирования, задачах и методах их решения, указывает и источники информации. Модель (рис. 4.3) согласуется со стадиями разработки согласно ЕСКД, а выпуск тех или иных видов технической документации представлен как результат соответствующих проектных процедур.

Проектирование начинается с определения потребности создании нового изделия, которая чаще всего диктуется состоянием общественного производства и отражается в перспективных планах партии и правительства. Внутри предприятия она может быть вызвана стремлением к повышению производительности труда или к устранению ручных операций. Инженер, постоянно наблюдающий за состоянием производства, своевременно ощущает ситуацию, препятствующую повышению производительности труда. Это вызывает у него стремление к устранению препятствия. Если возникшая ситуация хорошо знакома, то он может сразу принять решение и перейти к его реализации. Однако значительно чаще инженер не может сразу найти решение, лучшим образом удовлетворяющее возникшую потребность. Тогда он прибегает к развернутому во времени сложному информационному процессу, т. е. проектированию. Поиск может быть тогда удачным, когда имеется ясное представление о его цели.

Определение цели проектирования — весьма ответственная процедура. Во многих случаях результат разработки объектов новой техники оказывается неудовлетворительным из-за неправильной или неточной формулировки целей.

Основная задача процедуры выбора целей — распознать в общих чертах объект проектирования и его окружение. Какой-либо четкой методики решения нет. Определенным образом организует решение задачи составление сценария и построение графа целей. Маточником информации для выполнения процедура являются прогнозы развития самого объекта

проектирования и его окружения. Весьма удобный аппарат для анализа и синтеза информации представляет инженерное прогнозирование.

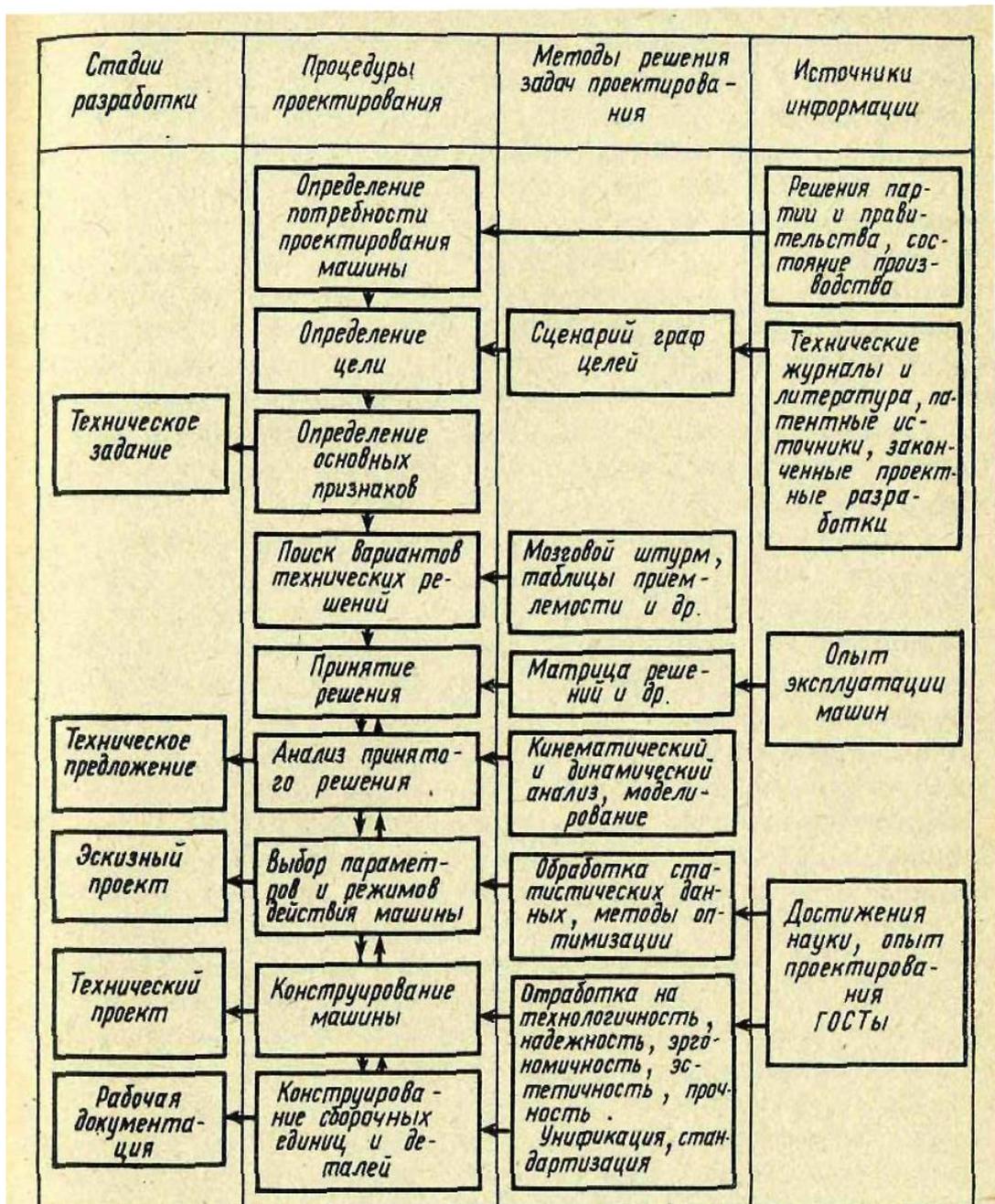


Рис. 4.3 Процедурная модель проектирования

Оно способно ответить на следующие вопросы: какие инженерные направления займут лидирующее положение в технике; каковы возможные

пропорции внедрения в практику конкурирующих направлений; какова вероятность использования в будущем тех или иных направлений; какова предполагаемая эффективность реализации технических направлений и, кроме того, когда можно ожидать внедрение в производство отдельных технических решений и целых направлений. Прогнозирование становится особенно важным сейчас, в эпоху бурного развития науки и техники. Выбирая, например, тот или иной способ воздействия на среду, как основу для создания новой машины, конструктор должен помнить, что на проектирование и внедрение в производство его разработки уйдет не менее 5 лет (речь идет о серийном изготовлении). За этот срок могут произойти существенные изменения в науке и технике, может оказаться, что выбранное направление перестанет отвечать научно-техническому прогрессу и созданная машина с первых дней своей жизни морально устареет. Чтобы избежать такого положения, нужно предвидеть развитие тех или иных технических направлений. На стадии определения целей уже может возникнуть то или иное техническое решение. Однако опытный конструктор не торопится с его реализацией. Он знает, что это решение далеко не единственное.

После выбора целей проектирования можно приступить к процедуре определения основных признаков объекта. Согласно основной концепции методологии эта процедура состоит в построении среза бинарных отношений между элементами множеств имеются множества признаков по выбранному подмножеству целей и признаков включаются в техническое

Процедура поиска возможных решений напоминает формирование оперативных моделей в сознании человека. Она в наибольшей степени опирается на творческие начала и выполняется чаще неформальными методами, однако уже сейчас имеются попытки [23] разрабатывать варианты технического решения с помощью ЭВМ. В дальнейшем будут показаны приемы, организующие творческий поиск. Основные источники информации: техническая литература и журналы, авторские свидетельства и патенты.

На следующем этапе проектирования выполняется процедура принятия решения. Из множества вариантов необходимо выбрать лучший по показателю или показателям, устанавливающим соответствие технического решения ранее определенным целям. Принятие решения уже сейчас

формализовало в значительно большей степени, чем предыдущие процедуры, хотя и содержит ряд задач, решаемых эвристическим методом. Для сравнения вариантов, не содержащих параметрическую информацию, можно применять матрицу решений и генеральную определительную таблицу. На окончательном этапе принятия решения используется экономический расчет.

Основным источником информации для сравнения вариантов служит опыт использования существующих однотипных изделий. Весьма полезную информацию для выполнения процедур несет развивающаяся теория принятия решений.

Отобрав из всех возможных вариантов один, конструктор должен тщательно проверить его на работоспособность и возможность технического воплощения. Эта процедура может быть названа анализом принятого решения. Методами решения задач на данном этапе проектирования выступают: кинематический и динамический анализ, моделирование. Может случиться, что-выбранный вариант не удовлетворяет условиям работоспособности или не может найти в современных условиях технического воплощения. В таком случае нужно вновь вернуться к этапу принятия решения, отобрать другой вариант и произвести его анализ. На схеме процедурной модели это отражено стрелкой, идущей вверх. Окончательным оформлением принятого решения является техническое предложение.

Любое, даже самое передовое техническое решение окажется бесплодным, если не получат технического воплощения. Прогресс техники направлен на повышение производительности труда, поэтому каждая новая машина оказывается, как травило, более производительной, но не (Всякая новая машина оказывается более надежной. Вызывается это неудачным конструированием, неправильным выбором параметров. Современная машина выступает как единый агрегат, отдельные узлы которого находятся во взаимодействии. Так, например, вибрационное воздействие рабочего органа передается не только на обрабатываемую среду, но и на раму машины, на ходовое и силовое оборудование, на систему управления, ухудшая условия работы этих узлов. Стремление к полному устранению вредного воздействия на все узлы может, однако, привести к существенному удорожанию машины. Удачная машина представляет собой оптимальное сочетание параметров всех ее узлов. Выбор параметров, связанный с компромиссом внутри противоречивых факторов, протекает в усло-

вия\* действия всемирного закона марксистско-ленинской диалектики — закона единства и борьбы противоположностей. Выбор параметров можно отнести к классу задач, которые носят название экстремальных. Тот или иной критерии качества, улучшение которого 'составляет цель проектирования, выбирается главным и представляется в виде функции, подлежащей максимизации >или минимизации. Аргументами этой функции служат параметры машины. Область их допустимых значений ограничивается некоторым подмножеством. Решить поставленную таким образом задачу значит найти такие значения аргументов из заданного подмножества, при которых целевая функция получает экстремальное значение. По результатам процедуры выбора параметров может быть составлен эскизный проект. Он представляет собой совокупность конструктивных документов, дающих представление в общих чертах о принципе работы прибора

#### **4. Техническое задание**

Техническое задание (ТЗ) на проектирование — первый и весьма важный технический документ. Он разрабатывается всегда независимо от дальнейшей стадии разработки.

Содержание ТЗ определено ГОСТ 15.001—88. Основное назначение технического задания — определить цели проектирования, обосновать направление поиска.

Составляется этот документ заказчиком с участием конструктора. Особо нужно подчеркнуть, что в современных условиях выполнить проектирование качественно только инженеру-конструктору, как правило, не удастся. Необходимо его постоянное сотрудничество с инженером-технологом и художником-конструктором.

На стадии разработки ТЗ инженер-технолог устанавливает базовые показатели технологичности изделия на основе аналогов с учетом корректирующих коэффициентов. Художник-конструктор в рамках составления сценария проводит предварительный анализ проектной ситуации с точки зрения технической эстетики и эргономики.

В ТЗ отражаются и результаты латентных исследований, указывается перспективное направление технических решений, а иногда и конкретное изобретение (группа изобретений), рекомендованное к внедрению.

## 5. Техническое предложение

В результате поиска вариантов технического решения и выбора наиболее рационального из них составляется техническое предложение. Согласно ГОСТ 2.103—68 оно должно содержать техническое и технико-экономическое обоснование целесообразности разработки объекта, т. е. основные результаты процедур <sup>11</sup> исков вариантов принятия решений, анализа принятого решения.

Технологи, участвуя вместе с конструкторами в выборе ванта, заботятся о лучших предпосылках для использования "стандартных и унифицированных узлов, типовых технологических процессов, ограничения номенклатуры конструкционных материалов.

Художник-конструктор формирует требования технической эстетики и эргономики, разрабатывает варианты художественно-конструкторского решения.

В техническом предложении отражаются результаты исследований по проверке патентной чистоты выбранного варианта технического решения как в СССР, так и в странах, предполагаемых для экспорта.

В число обязательных документов технического предложения входят пояснительная записка и ведомость технического предложения. В зависимости от характера, назначения или условий производства объекта дополнительно могут быть выполнены: чертеж общего вида или габаритный чертеж, схемы, таблицы, расчеты, патентный формуляр, карта технического уровня и качества продукции.

## 6. Эскизный проект

По выбранным основным параметрам может быть разработан эскизный проект. В число его обязательных документов входят пояснительная записка и ведомость эскизного проекта. Дополнительно могут составляться: чертеж общего вида, габаритный чертеж, теоретический чертеж, схемы, ведомость покупных изделий, ведомость согласования применения покупных изделий, программа и методика испытаний, расчеты, таблицы, патентный формуляр, карта технического уровня и качества продукции

В пояснительной записке отражается выбор параметров. Кроме того, в нее включаются расчеты ожидаемого экономического эффекта. При эскизном проектировании в отличие от стадии технических предложений они выполняются по более уточненным данным. Если расчеты подтверждают экономический эффект, определенный на стадии технического предложения, то принимается решение о продолжении разработки. В противном случае нужно в него внести изменение или обратиться к другому.

В ходе эскизного проектирования инженер-технолог совместно с конструктором продолжают отработку на технологичность, заключающуюся в [41]: окончательном выборе рациональной конструктивной схемы; принципиальной оценке технологичности основных составных частей; выявлении составных частей, которые могут быть стандартными или унифицированными; выявлении составных частей, которые могут быть позаимствованы; выявлении условий сборки изделия и составных частей; выявлении номенклатуры используемых конструкционных материалов; выявлении условий технического обслуживания изделия; выявлении условий контроля, регулировки и подготовки изделия к функционированию; выявлении условий подготовки производства и определении основных укрупненных данных для организации технологической подготовки производства; выявлении новых технологических процессов, требующих технического оснащения и освоения.

Расчет показателей технологичности производится на основе базовых данных, установленных в техническом задании.

Художественная проработка изделия на этапе эскизного проекта включает изучение конструкции, материалов и технологии изготовления; разработку вариантов композиции графически ляли на макетах; выбор окончательного варианта композиции.

Особо следует остановиться на роли художественной проработки на этапе эскизного проекта. Для объектов новой техники, не имеющих аналогов, это, пожалуй, пока единственный наиболее успешный метод общей компоновки. Интуиция дизайнера подсказывает наиболее рациональное размещение основных узлов с точки зрения эстетики. Оно же, как правило, оказывается я наиболее удачным в техническом отношении. Такая закономерность высказывается многими конструкторами. В практике работы

студенческого конструкторского бюро ЛИИЖТа она проверена в ходе совместных работ с учащимися Ленинградского высшего художественно-графического училища им. В. Н. Мухиной. На стадии эскизного проекта продолжаются работы по выявлению патентоспособных решений, которые могут появиться в ходе компоновки объекта. Оформляются заявки на изобретение как по устройству, так и по промышленному образцу. Выявляются страны или фирмы-потребители объекта, разрабатываются предложения о патентовании изобретений за границей

## **7. Технический проект**

Процедура конструирования объекта завершается составлением технического проекта, содержащего окончательные технические решения, дающие полное представление об устройстве разрабатываемого изделия, и исходные данные для разработки рабочей документации. Обязательными документами технического проекта являются: чертеж общего вида, пояснительная записка, ведомость технического проекта. Дополнительно в зависимости от характера, назначения или условий производства изделия могут быть теоретический и габаритные чертежи, расчеты, таблицы, схемы, ведомость покупных изделий, ведомость согласования применения покупных изделий, технические условия, программа и методика испытаний, патентный формуляр, карта технического уровня и качества продукции.

Инженер-технолог, участвуя в разработке технического проекта, отрабатывает конструкцию на технологичность, добиваясь наилучших значений ее показателей.

Инженерно-психологической и художественной проработкой объекта определяются: окончательная компоновка и конструкция рабочих мест, средства обеспечения условий обитаемости, конкретные задачи и функции, выполняемые оператором, техническая форма объекта и его составных частей.

Патентными исследованиями обосновывается возможность использования технических решений, защищенных авторскими

## Лекция №5

### Компьютерное проектирование приборных систем с использованием интегрированного пакета Simulink

Систему **Simulink** можно рассматривать как совокупность методов и средств автоматизации процесса разработки современных систем управления. Широкое применение пакет нашел в промышленности, экономике, образовании и других сферах деятельности. Отличительной особенностью пакета Simulink является *быстрая разработка моделей опытных образцов и ускоренное прикладное проектирование*. Для этого используются проектные процедуры создания среды моделирования.

#### 1. Назначение и особенности Simulink

Система Simulink является расширением системы инженерных и научных расчетов MATLAB и предназначена для решения задач сокращения сроков проектирования, повышения качества разработки моделей физических систем и моделирования процессов в этих системах. Развитие этих подходов дает возможность улучшить документирование работ и использовать средства анимации.

При традиционном подходе, который использовался до настоящего времени, на первом этапе производилось формирование математической модели из описания в виде структурной схемы. Затем программа записывалась на универсальном языке программирования. При этом возникало дублирование описания элементов и связей элементов системы управления, что, как правило, приводило к синтаксическим ошибкам записи алгоритма на языке программирования. Это требовало внесения исправлений и повторного выполнения процесса, что было связано с большими потерями времени. В системе Simulink принципиально изменен характер требований к математическому обеспечению и для управления всем ходом вычислительного процесса разработано графическое описание модели в виде структурной схемы системы.

Представление в виде структурных схем в большинстве случаев не требует написания кода программы из-за наличия библиотеки готовых программ, используемых для решения отдельных прикладных задач (биб-

лиотеки блоков). Рассмотрим механическую систему, которая показана на рис. 5.1. В таблице 5.1 приводятся сравнительные результаты выполнения программы моделирования этой системы: на языке ассемблера процессора 8086, используя массив данных с форматом 16 бит на элемент и интегрирование по методу Эйлера; на языке Фортран с описанием данных в форме с плавающей точкой; в системе МАТЪАВ с применением векторно-матричного описания

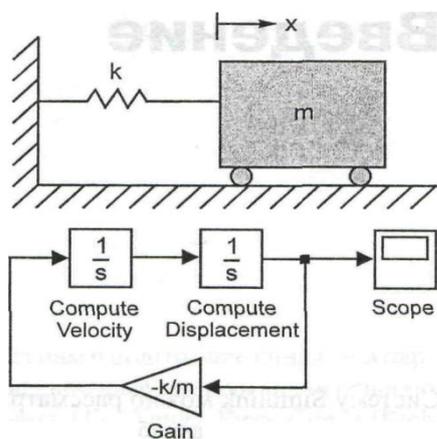


Рис. 5.1. Механическая система и Simulink модель механической системы

и с использованием систем Simulink. Для системы Simulink в таблице приведено число заданных блоков и количество нажатий кнопки мыши. Эти данные можно сравнить с соответствующим числом операторов программы и операций по нажатию клавиш клавиатуры. Учитывая прямо пропорциональную зависимость между числом строк прикладной программы и временными затратами, очевидно, что система Simulink значительно повышает эффективность труда проектировщика.

Сравнение размеров программного кода

Таблица 5.1.

Язык программирования	Число строк кода	Количество нажатий клавиш
Ассемблер 8086	921540	
Fortran	14 240	
Matlab	3 90	
Simulink	4 25	

## **2. Реализация результатов проектирования в системе Simulink**

В этом разделе рассматриваются две основные процедуры, осуществляемые проектировщиками при использовании системы Simulink: *быстрая разработка моделей опытных образцов* и *ускоренное прикладное проектирование (RAD)*. Быстрое создание моделей опытных образцов служит для реализации результатов конструкторского проектирования, разработки и испытания опытного образца или пробной партии систем управления в сжатые сроки. Методы ускоренного прикладного проектирования оптимизированы для сокращения сроков разработки. RAD реализует дополнительные возможности методологии быстрой разработки опытных образцов, в которой программа для ЭВМ является моделью системы Simulink, либо автоматически генерируется из модели системы Simulink

## **3. Разработка опытных образцов**

Дискретные системы управления в качестве корректирующих устройств включают персональные компьютеры или цифровые регуляторы. Разработка и отладка программного обеспечения для управления конкретной физической системой требует существенных затрат. При создании алгоритмов для летательных аппаратов, в целом ряде других научно-технических направлений использование модели является единственным способом анализа и синтеза систем управления из-за невозможности экспериментальной настройки и тестирования результатов в реальных условиях. Первым этапом при создании программного обеспечения для регуляторов является интерпретация математической модели объекта управления, которая доступна для реализации на ЭВМ, и использование данной модели для моделирования системы управления. После составления математических моделей объекта, датчиков и исполнительных устройств выполняется проектирование регулятора на основе разработанных моделей в виде структурных схем. Затем записываются уравнения цифрового регулятора и корректирующее устройство включается в структурную схему разрабатываемой системы. По результатам моделирования вносятся необходимые коррективы в модель регулятора. Цикл повторяется для адекватной оценки

результатов синтеза и испытаний альтернативных вариантов регуляторов. Разработка моделей опытных образцов и последующая трансляция модели контроллера из структурной схемы в программный код завершает этап синтеза регулятора.

Разработка моделей опытных образцов рассматривается во взаимосвязи с построением соответствующей математической модели для физической системы (объекта), которой надо управлять надлежащим образом. Математическая модель может быть представлена как модель системы Simulink, модель, построенная в системе MATLAB или написанная на языке С. Затем дискретный регулятор разрабатывается в системе Simulink, или с помощью пакета прикладных программ (ППП) Optimization Toolbox. Эта стадия разработки представляет собой совместную проработку частей проекта, связанную с учетом определенных требований и положений, определяющих функционирование системы. При создании моделей опытных образцов технические решения конкретизируются и детализируются.

Выполнение этапа проектирования, связанного с созданием и испытанием опытного образца, позволяет получить программное обеспечение дискретного регулятора, способное функционировать на персональных компьютерах. Программа загружается в компьютер и проверяется конкретный регулятор, который управляет дискретной моделью физического объекта. Это позволяет оценить результаты программной реализации закона регулирования.

Процедура синтеза закона управления, оценка результатов синтеза и моделирование системы выполняются в системе. Затем реализуется этап испытания физической системы. Следует отметить, что операция создания специализированного программного обеспечения на языке высокого уровня для регулятора выполняется однократно. Во-вторых, данный подход обеспечивает возможность легко вводить изменения, дополнения и исправления в проектные процедуры этапа программирования.

#### **4. Прикладное проектирование**

Ускоренное прикладное проектирование (КАО) не требует написания программ на языке высокого уровня и тем самым использует подходы, реализованные на этапе проектирования модели опытного образца. Для

цифровых регуляторов, которые программно совместимы с системой Simulink, обеспечивается возможность создания соответствующего программного кода. Для проекта прокатного стана, обсужденного ранее в разделе 1.2.1, модель, созданная в системе Simulink, используется инженерами как средство освоения, моделирования и эксплуатации физической системы.

В ряде случаев модель цифрового регулятора системы Simulink, не может быть использована непосредственно в конкретном проекте, или выбранный регулятор не обеспечивает достаточное быстродействие, либо конкретный промышленный регулятор не совместим с системой Simulink, Эти и другие ограничения могут быть сняты путем использования операции автоматической генерации программного кода программным средством, которое из описания модели в виде структурной схемы формирует представление модели на языке C. Программное средство RTW (Real Time Workshop) автоматически конвертирует структурные схемы системы в код на языке C, который может быть скомпилирован для использования на целевом компьютере. Автоматически сгенерированный код регулятора может быть скомпилирован со стандартными исполняемыми модулями (например, в специальных операционных системах реального времени) и кодом интерфейса для включения в состав систем управления со встроенными цифровыми регуляторами. В тех случаях, когда модель системы выполняется недостаточно быстро, транслирование в код на языке C с помощью программного средства RTW обеспечивает высокое быстродействие, особенно при использовании специальных аппаратных средств моделирования, предлагаемых dSPACE, Applied Dynamics и другими разработчиками.

### ***5. Выбор программных средств***

Рассматривая общие принципы построения программного обеспечения (ПО) автоматизированных систем разработки, можно отметить, что система Simulink является инструментом моделирования систем управления, ориентированным на выполнение определенного класса проектных процедур вне зависимости от объектов проектирования, и представляет собой проблемно-ориентированный программный комплекс. Например, система Simulink может использоваться для моделирования электронных схем, однако существуют специализированные пакеты прикладных про-

грамм, непосредственно предназначенные для этих целей. Значительное число задач в сфере программирования предпочтительно решать с применением языков программирования общего назначения. При выборе инструментария для решения задачи инженерного проектирования необходимо ответить на три главных вопроса:

1. Действительно ли инструментарий соответствует решаемой задаче? Система 8нгшПпк разработана для математических моделей систем, представленных в виде структурных схем, которые целесообразно использовать, если существенной частью проектных работ является моделирование.

2. Каково значение графических представлений? Временные зависимости являются важными компонентами технических проектов в сфере образования и в промышленной сфере. Если система Simulink ориентирована для решения текущих задач, данный подход может сэкономить значительное время.

3. Как выбирать средства разработки, ориентируясь на общую стоимость проектных работ? Проведенные оценки показали, что действия по исправлению ошибки на стадии рабочего проекта, испытаний и внедрений обходятся в 10-100 раз дороже, чем исправление исходной ошибки в программном обеспечении на стадии эскизного проекта [2]. При определенном допущении математическую модель системы в системе 81тиНпк можно представить как программную форму проектной документации. Таким образом, использование системы Simulink для оперативной разработки опытных образцов или прикладного проектирования является средством экономии и эффективного использования выделенных средств.

## **6. Начало работы**

Система Simulink обеспечивает пользователю понятную и удобную форму взаимодействия с ПК, как с помощью языка MATLAB, так и графического интерфейса пользователя.

Диалоговое окно в системе Simulink служит для настройки параметров блоков и задания параметров моделирования

Главное окно MATLAB

Библиотека блоков Simulink

Построение модели  
Редактирование модели

## **7. Задание параметров моделирования**

Создание математической модели физической системы включает в себя ряд этапов: математическую формулировку задачи, построение дискретной модели и реализацию численного метода на ЭВМ. Математический аппарат в большинстве случаев использует описание в виде дифференциальных и разностных уравнений.

В системе Simulink модель строится в виде структурной схемы, затем выполняются этапы анализа, синтеза и моделирования системы, используя выбранный численный метод. Для функции интегрирования задаются входные параметры: интервал интегрирования, шаг интегрирования, допустимые погрешности и др. Кроме этого, можно загружать переменные из рабочей области или размещать переменные в рабочей области системы MATLAB.

## **Лекция № 6**

### **Технология проектирования динамических процессов в Simulink**

#### **1. Исследование на компьютерной модели гипотезы влияния спроса и предложения на динамику рыночного равновесия**

##### ***1.1 Постановка задачи исследований***

Поставка товаров на рынок увеличивается в случае повышения спроса и увеличения цены на товар. При этом также увеличивается число поставщиков. Потребители покупают товар и чем меньше цена, тем больше его раскупают и больше покупателей появляется на рынке. Товар на рынке характеризуется двумя параметрами: количеством и ценой.

Основоположник ценовой теории Альфред Маршалл полагал, что большинство экономических процессов можно объяснить в терминах рав-

новесной рыночной цены. Цена устанавливается при взаимодействии спроса и предложения. Но, регулируя предложение, не всегда удается установить приемлемые потребителям и производителям цены.

Обычно на листе бумаги чертят пересечение линий спроса и предложения в зависимости от цены товара. Смещают линии, меняют их крутизну, наблюдают точки новых равновесий. Объясняют ножницы дефицита, инфляцию, перепроизводство и др. Наиболее удобно все это можно сделать с помощью компьютера в системе Matlab – Simulink.

## ***1.2 Математическая модель***

Для первоначального изучения выбирается линейная модель, без запасов, случайностей, прогнозов и прочих возмущающих факторов.

Функция зависимости спроса от цены имеет вид:

$$D_{md} = D_0 - K_d * P_{ic}$$

где:  $D_{md}$  – спрос за текущий интервал времени;

$D_0$  – спрос при нулевой цене;

$K_d$  - крутизна линий спроса;

$P_{ic}$  – цена товара.

Линия зависимости предложения от цены:

$$S_{hl} = S_0 + K_s * P_{ic}$$

где :  $S_{hl}$  – предложение за текущий интервал времени;

$S_0$  - предложение при нулевой цене;

$K_s$  – крутизна линий спроса;  $P_{ic}$  – цена товара

.

## ***1.3 Компьютерная модель в Simulink и управление экспериментами***

Блок схема компьютерной имитационной модели представлена на рис. 5.2.

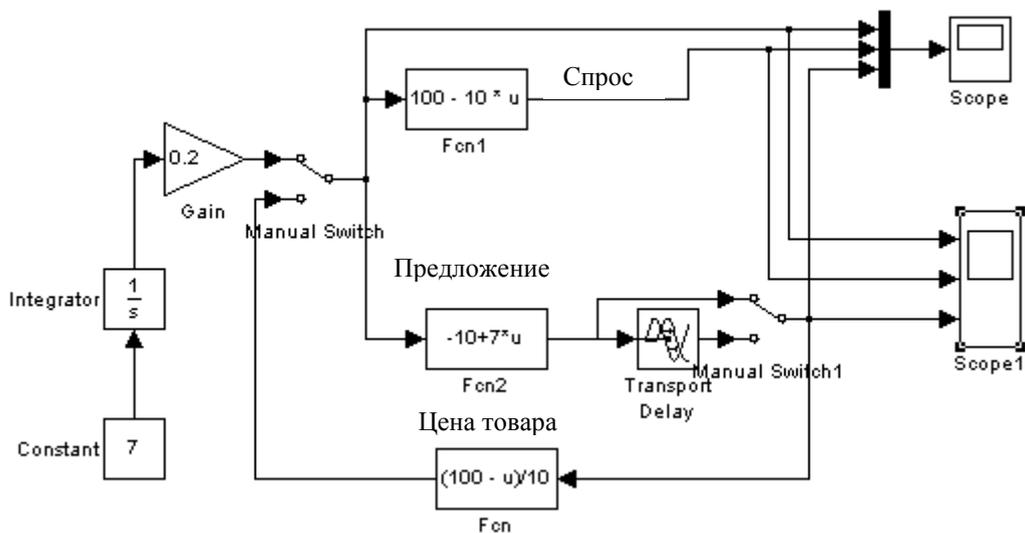


Рис.5.2. Блок схема имитационной модели рыночного равновесия

**Предложение** представлено стандартным блоком **Fcn2**. В нем вычисляется значение предложения в зависимости от цены, подаваемой на вход блока. Обозначения параметров в блоке и их значения следующие:

$u = P_{ic}$  - цена товара;  $S_0 = 10$  шт. - предложение при нулевой цене;

$K_s = 7$  – крутизна линий спроса.

Блок **Transport Delay** имитирует запаздывание поставщика на рынке. Это означает, что продавец поставляет товар в количестве определенном на основе цен прошлого интервала времени.

Блок **Fcn** имитирует решение поставщика согласится с ценой текущего спроса и принимает ее. Он соглашается продать весь товар по цене, которую диктует линия **Спрос**, формируемая в блоке **Fcn1**.

Блок **Fcn** реализует функцию обратную функции спроса и вычисляет цену, по которой покупатель сможет купить весь товар поставщика. Так последовательно происходит выравнивание цен предложения и спроса и определяется цена, при которой предложение становится равно спросу. Параметры блоков **Fcn** и **Fcn1** в расчетах принимаются одинаковыми.

### 1.4 Задание параметров моделирования

Методы численного решения задач Коши для обыкновенных диф. уравнений реализуется в специальном решателе ОДУ. В окне **Simulation**

выбирается команда **Simulation Parameters** . Данное окно имеет пять вкладок **Solver**, **Workspace**, **Diagnostics**, **Advanced** (Экспертные настройки), **Real-Time Workspace** (Мастерская реального времени)

### 1.5 Внутримодельное управление

В модели кроме экономической части присутствуют блоки управления экспериментом.

Отображение результатов моделирования осуществляется блоками **Scope**. На схеме модели рис. 5.3 они расположены справа. Блоки строят графики изменения во времени проса, предложения и цены.

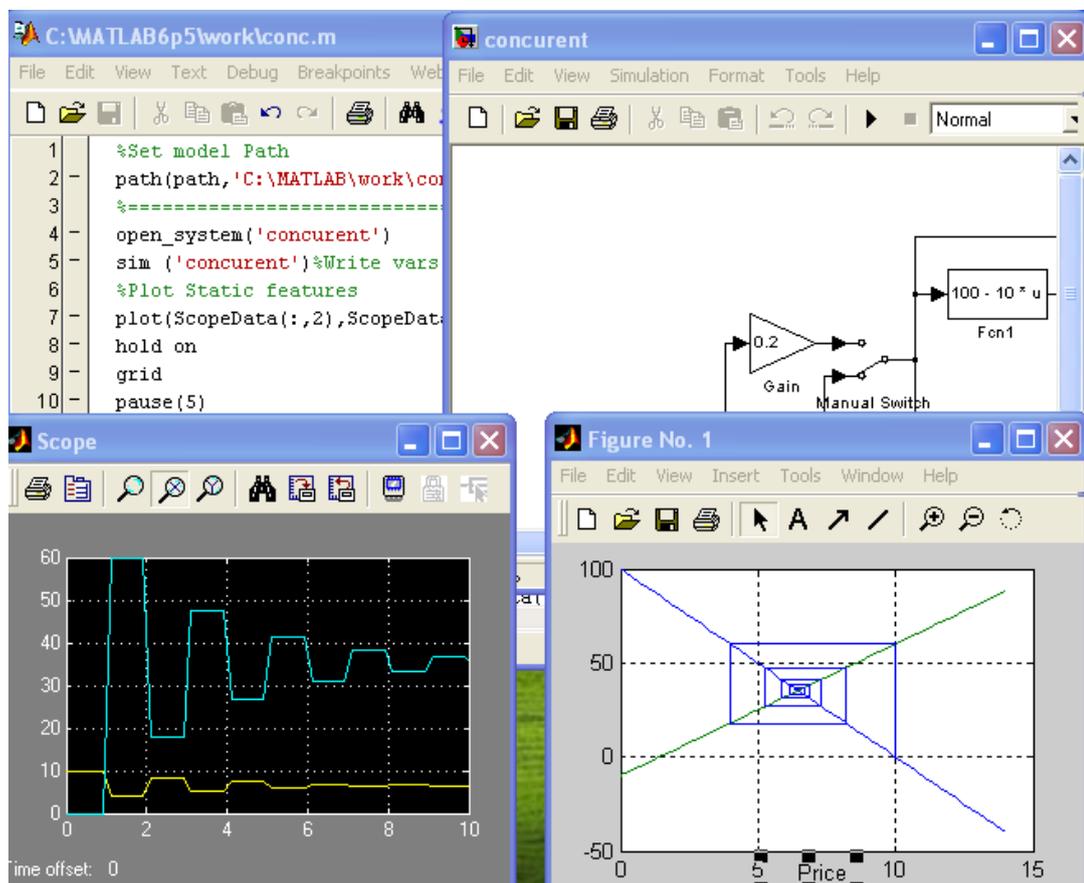


Рис. 5.3. Вид экрана при выполнении лабораторных работ по анализу процессов рыночного равновесия

Слева блоки константы, интегратора и усилителя задают значение цены для построения графиков статических характеристик функций спроса и предложения.

Ключи предназначены для переключения режимов моделирования. Для первого режима моделирования, чтобы построить функции спроса и предложения от цены мы ставим ключи в верхнее положение. Вычисляются значения функций и отправляются в рабочее пространство Matlab для построения крестообразного графика спроса и предложения.

Затем, для выбора второго режима устанавливаем стрелку мыши на один из ключей и двойным щелчком левой кнопки мыши переводим ключи в нижнее положение. Процесс имитационного моделирования в это время продолжается и формируется переходный процесс рынка в равновесное состояние. Графическое изображение переходного процесса, построенного редактором Matlab, показано на рис. 5.4

### 1.6 Программа управления экспериментом

```
1 | %Set model Path
2 | path(path, 'C:\MATLAB\work\conc')
3 | %=====
4 | open_system('concurrent')
5 | sim('concurrent')%Write vars into WS from scope
6 | %Plot Static features
7 | plot(ScopeData(:,2),ScopeData(:,3:4))
8 | hold on
9 | grid
10 | pause(5)
11 | %=====
12 | %2.simulate prise dynamic
13 | sim('concurrent')
14 | %=====
15 | %3.plot price Web graphics
16 | for i=2:57
17 |     plot([ScopeData(i-1,2) ScopeData(i,2)],[ScopeData(i,4) ScopeData(i,4)])
18 |     plot([ScopeData(i,2) ScopeData(i,2)],[ScopeData(i,4) ScopeData(i+1,4)])
19 | end
20 | hold off
```

Рис. 5.4 Программа управления экспериментом

Строки со знаком % представляют комментарий к программе и не выполняются компьютером. Командой path устанавливается путь к модели

в файловой системе. Оператор **open\_system** загружает блок-модель под названием **concurrent**, показанную на рис 5.2, в Simulink.

Функция **sim** запускает модель **concurrent**. **Plot** чертит крест графиков спроса и предложения. **Hold on** разрешает дополнять рисунок новыми графиками. **Grid** рисует для графиков масштабную сетку. На этом первый этап заканчивается. Функция **pause** останавливает процесс моделирования на 5 секунд для установки ключей двойным щелчком мыши в нижнее положение.

Затем начинается второй этап – моделирование переходного процесса к рыночному

равновесию. Оператор **Sim** повторно запускает модель. Идет имитация. **Scopes** в своих окнах чертит графики показателей и пишут их значения в рабочее пространство Matlab **Workspace**.

Цикл **for**, используя данные памяти, чертит линии паутины переходного процесса цены к цене рыночного равновесия.

Общий вид окон эксперимента представлен на рис. 5.3.

Слева вверху расположено окно редактора - отладчика Matlab с m-файлом управления экспериментом. Справа вверху окно модели Simulink. Внизу слева окно Scope, справа график паутины переходного процесса к равновесию на конкурентном рынке, построенный программой Matlab.

### ***1.7. Изучение переходного процесса к рыночному равновесию***

Из командного окна Matlab открыть программу управления экспериментом, это m-файл с именем «**conc**». Файл отображается в окне редактора. Запускаем файл из меню **Debug > Ran**. Программа загружается и стартует модель. Сразу после старта в окне Simulink на схеме переводим ключи в нижнее положение. Наблюдаем графики изменения показателей во времени в окнах Scope и паутину движения цены в окне Figure. Графики Scope представлены на рис 5.5. Для более наглядного просмотра графиков используется пиктограмма в виде бинокля, находящаяся в середине инструментальной строки, с помощью которой устанавливается автоматическое масштабирование.

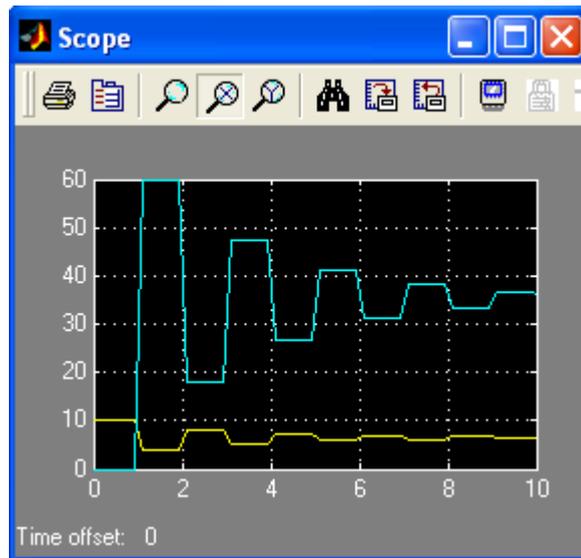


Рис. 5.5. Изменение цены, спроса - предложения в окне Score.

Верхний график показывает изменение цены, а нижний изменение спроса и предложения.

Графики Score1 представлены на рис. 5.6.

График паутинообразной модели движения к рыночному равновесию представлен на рис. 5.7. По оси абсцисс находится цена товара, а по оси ординат количество. При первоначальной цене в 10 у.е. спрос на товары нулевой, однако при движении к равновесию в точку пересечения линий

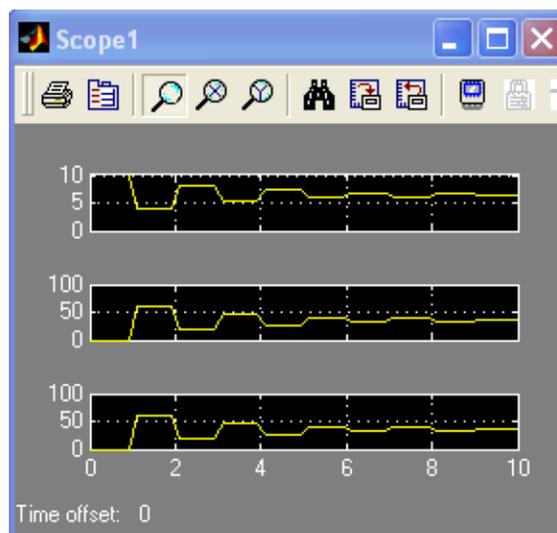


Рис. 5.6. Изменение цены, спроса и предложения во времени в окне Score1.

спроса и предложения становится очевидным, что оптимальная цена для данного товара составляет 6,3 у.е. и продать можно 35 единиц товара, получив при этом максимальную прибыль.

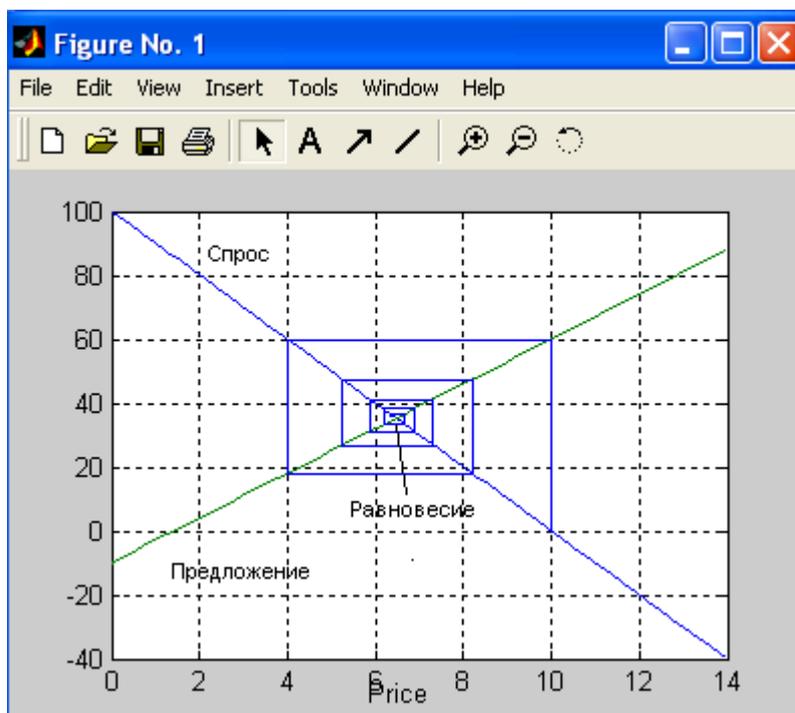


Рис. 5.7. График паутины движения цены к равновесию на рынке.

### ***1.8 Изучение влияния смещения линий спроса и предложения на рыночное равновесие***

Для смещения спроса изменим значения параметров  $D_0$  в блоках Fcп и Fcп1 последовательно на 60; 80; 100; 120; 140 и регистрируем движение цены к новому равновесию. Построить в Excel график движения цены к новому равновесию.

Для смещения предложения изменим значения параметров  $S_0$  в блоке Fcп1 последовательно на 5; 8; 11; 14; 17 и наблюдать движение к новому равновесию. Построить в Excel график движения цены к новому равновесию.

### **1.9. Изучение влияния крутизны линий спроса и предложения на рыночное равновесие**

Для изменения крутизны спроса меняем значения параметров  $K_d$  в блоках Fcn и Fcn1 последовательно на 12; 14; 16; 18; 20 и наблюдаем движение цены к новому равновесию.

Для изменения крутизны предложения меняем значения параметров  $K_s$  в блоке Fcn1 последовательно на 6; 8; 9; 10; 12 и наблюдаем движение к новому равновесию.

При увеличении  $K_s$  колебания показателей в системе возрастают. При  $K_s = K_d$  параметры колебаний не изменяются во времени. Устойчивое равновесие не достигается. На экране паутина превращается в прямоугольную рамку, это значит система находится в режиме устойчивых колебаний. При  $K_s > K_d$  система не устойчива, процесс расходится, паутина расширяется и уходит из окна. Это не соответствует действительности и подтверждается предположение, что модель груба.

## **Лекция № 7**

### **Общие сведения о системе проектирования печатных плат P-CAD**

Система **P-CAD** предназначена для проектирования многослойных печатных плат (ПП) вычислительных и радиоэлектронных устройств. В состав **P-CAD** входят четыре основных модуля - **P-CAD Schematic**, **P-CAD PCB**, **P-CAD Library Executive**, **P-CAD Autorouters** и ряд других вспомогательных программ (рис. 7.1).

**P-CAD Schematic** и **P-CAD PCB** - соответственно графические редакторы принципиальных электрических схем и ПП. Редакторы имеют системы всплывающих меню в стиле **Windows**, а наиболее часто применяемым командам назначены пиктограммы. В поставляемых вместе с системой библиотеках *зарубежных цифровых* ИМС имеются три варианта графики: **Normal** - нормальный (в стандарте США), **DeMorgan** — обозна-

чение логических функций, **IEEE** — в стандарте Института инженеров по электротехнике (наиболее близкий к российским стандартам).

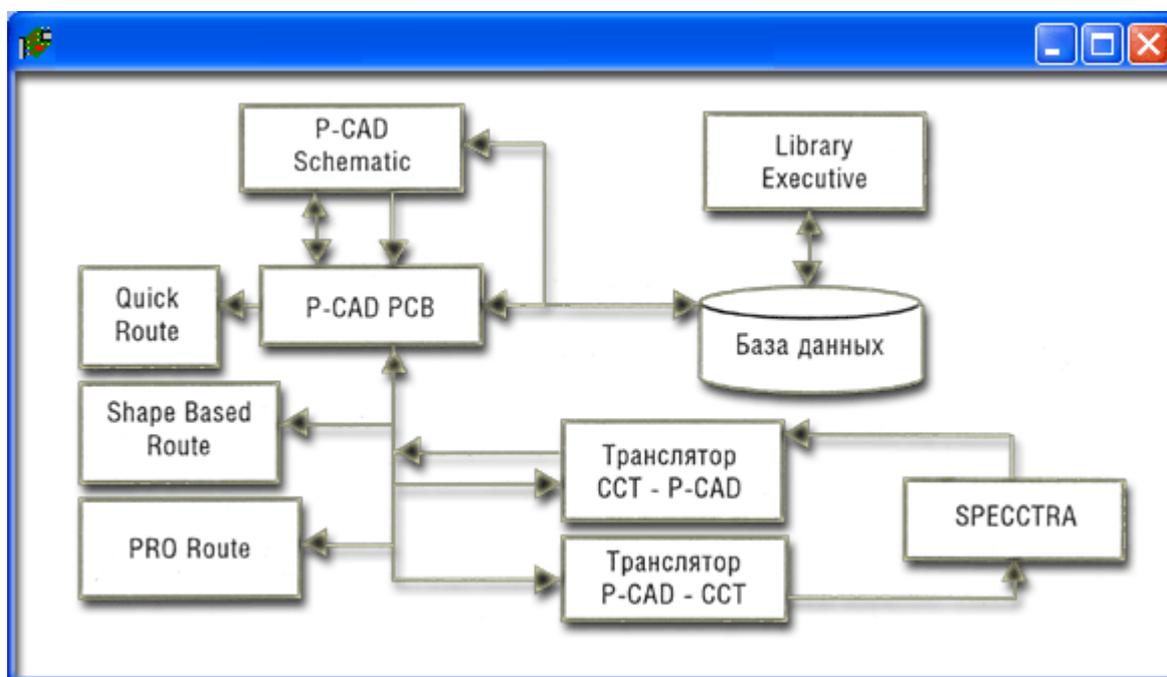


Рис. 7.1. Структура системы проектирования **P-CAD**

Редактор **P-CAD PCB** может запускаться автономно и позволяет разместить модуль на выбранном монтажно—коммутационном поле и проводить ручную, полуавтоматическую и автоматическую трассировку проводников. Если **P-CAD PCB** вызывается из редактора **P-CAD Schematic**, то *автоматически* составляется список соединений схемы и на поле ПП переносятся изображения корпусов компонентов с указанием линий электрических соединений между их выводами. Эта операция называется **упаковкой схемы на печатную плату**. Затем вычерчивается контур ПП, на нем размещаются компоненты и, наконец, производится трассировка проводников.

Применение шрифтов **True Type** позволяет использовать на схеме и ПП надписи на русском языке.

**Автотрассировщики** вызываются из управляющей оболочки **P-CAD PCB**, где и производится настройка стратегии трассировки. Информацию об особенностях трассировки отдельных цепей можно с помощью стандартных атрибутов ввести на этапах создания принципиальной схемы

или ПП. Первый трассировщик **QuickRoute** относится к трассировщикам лабиринтного типа и предназначен для трассировки простейших ПП. Второй автоматический трассировщик **PRO Route** трассирует ПП с числом сигнальных слоев до 32. Трассировщик **Shape-Based Autorouter** - бессеточная программа автотрассировки ПП. Программа предназначена для автоматической разводки многослойных печатных плат с высокой плотностью размещения элементов. Эффективна при поверхностном монтаже корпусов элементов, выполненных в различных системах координат. Имеется возможность размещения проводников под различными углами на разных слоях платы, оптимизации их длины и числа переходных отверстий.

**Document Toolbox** - дополнительная опция **P-CAD PCB** и **P-CAD Schematic** для размещения на чертежах схем или ПП различных диаграмм и таблиц, составления различных списков и отчетов, которые динамически обновляются, таблиц сверловки, данных о структуре платы, технологической и учетной информации, размещения на чертежах схем списков соединений, выводов подключения питания и другой текстовой информации. Программа предназначена для расширения возможностей выпуска технической документации без использования чертежных программ типа **AutoCAD**. **Document Toolbox** позволяет автоматизировать создание конструкторской документации, необходимой для производства проектируемых ПП.

**SPECSTRA** - программа ручного, полуавтоматического и автоматического размещения компонентов и трассировки проводников. Трассирует ПП большой сложности с числом слоев до 256. В программе используется так называемая бессеточная **Shape-Based** - технология трассировки. За счет этого повышается эффективность трассировки ПП с высокой плотностью размещения компонентов, а также обеспечивается трассировка одной и той же цепи трассами различной ширины. Программа **SPECSTRA** имеет модуль **AutoPlace**, предназначенный для автоматического размещения компонентов на ПП. Вызов программы производится автономно из среды **Windows** или из программы **P-CAD PCB**.

**P-CAD Library Executive** - менеджер библиотек. *Интегрированные библиотеки P-CAD* содержат как *графическую* информацию о символах и типовых корпусах компонентов, так и *текстовую* информацию (число секций в корпусе компонента, номера и имена выводов, коды логической

эквивалентности выводов и т.д.). Программа имеет встроенные модули: **Symbol Editor** — для создания и редактирования символов компонентов и **Pattern Editor** — для создания и редактирования посадочного места и корпуса компонента. Упаковка вентиля компонента, ведение и контроль библиотек осуществляются модулем **Library Executive**. Модуль имеет средства просмотра библиотечных файлов, поиска компонентов, символов и корпусов компонентов по всем возможным атрибутам.

**Вспомогательные утилиты**, образующие интерфейс **DBX (Data Base Exchange)**, в частности, производят перенумерацию компонентов, создают отчеты в требуемом формате, автоматически создают компоненты, выводы которых расположены на окружности или образуют массив, рассчитывают паразитные параметры ПП и т. п.

*Основные характеристики* системы проектирования **P-CAD** приведены ниже.

#### Общие характеристики:

- 32-разрядная база данных;
- разрешающая способность **P-CAD PCB** и других программ равна **0,001мм**;
- до 100 открытых одновременно библиотек;
- число компонентов в одной библиотеке - неограничено;
- до 64 000 электрических цепей в одном проекте;
- до 10 000 выводов в одном компоненте;
- до 5000 секций (вентилей) в одном компоненте;
- до 2000 символов в атрибуте компонента;
- до 2000 символов в текстовой строке;
- до 20 символов в имени вывода, имени цепи, позиционном обозначении вывода (пробелы, знаки табуляции, точки и скобки не допускаются);
- до 16 символов в имени типа компонента (пробелы и знаки табуляции не допускаются);
- до 30 символов в позиционном обозначении компонента (двоеточие, пробелы, знаки табуляции, точка и точка с запятой не допускаются);

- до 8 символов в имени файла (в том числе и при работе в среде Windows);
- многошаговый «откат» вперед и назад. По умолчанию количество запоминаемых шагов установлено равным 10, но эту величину можно при необходимости изменить, редактируя файл конфигурации **\*.ini**.
- минимальный шаг сетки 0,1 mil в английской системе и 0,001 мм в метрической системе (**1 mil = 0,001 дюйма = 0,0254 мм, 1 мм = 40 mil**). Систему единиц можно изменять в любой фазе проекта.

#### ***Графический редактор принципиальных схем P-CAD Schematic:***

- до 99 листов схем в одном проекте, максимальный размер листа 60 x 60 дюймов;
- поддержка стандартных форматов листов от A до E, A0-A4 и др. форматов;
- дискретность угла поворота компонента 90°;
- работает утилита **ERC** для просмотра и сортировки ошибок в принципиальных схемах;
- перекрестные связи между **P-CAD Schematic** и **P-CAD PCB** позволяют для выбранной на схеме цепи высветить на ПП соответствующий ей проводник и наоборот;
- возможна передача данных в программу моделирования **Dr. Spice A/D**.

#### ***Графический редактор печатных плат, P-CAD PCB:***

- до 99 слоев в ПП, из них 11 слоев предварительно определены;
- максимальный размер ПП 60 x 60 дюймов;
- автоматическая коррекция принципиальных схем по изменениям в печатной плате и наоборот (коррекция «назад» и «вперед»);
- до 64 000 типов контактных площадок в проекте;
- ширина проводника на ПП до 10 мм;
- до 64 000 стилей стеков контактных площадок в проекте;

- контактные площадки различных форм: эллипс, овал, прямоугольник, скругленный прямоугольник, сквозное переходное отверстие, перекрестье для сверления (target), непосредственное соединение, тепловой барьер с 2 или 4 перемычками;
- контроль соблюдения зазоров и полноты разводки ПП;
- минимальный дискрет угла поворота текста и графических объектов — 0,1 град;
- поддержка управляющих файлов фотоплоттеров **Gerber** и сверлильных станков с ЧПУ типа **Excellon**.

### 1.1. Интерфейс пользователя

Графические редакторы **P-CAD** имеют похожие интерфейсы и системы меню команд, общие сведения о которых излагаются ниже.

На рис.7.2 представлен экран графического редактора **P-CAD Schematic**.

Горизонтальная панель инструментов содержит пиктограммы *системных команд*, а вертикальная панель - *команды размещения* объектов на рабочем поле экрана.

В поле *рабочего окна* располагают символы принципиальных схем и собственно схемы, составленные из символов, электрических соединений, шин и т. п.

Вторая строка снизу на экране — строка *сообщений*.

Самая нижняя строка - *строка состояний*. Значения лелей строки состояния перечисляются ниже.

*Координаты X и Y*. Числа в полях указывают текущие координаты курсора. Перемещение курсора в заданную пользователем точку производится следующим образом. Если активизирован режим выбора объекта (команда Edit/Select), то нажатие клавиши J передает управление полю X. На клавиатуре можно набрать значение координаты X, затем нажать клавишу **Tab**, набрать значение координаты Y и нажать клавишу **Enter**. В результате указанных действий курсор переместится в заданную точку. Если выбрана одна из команд размещения Place, то можно указанными выше операциями разместить объект в заданную точку.

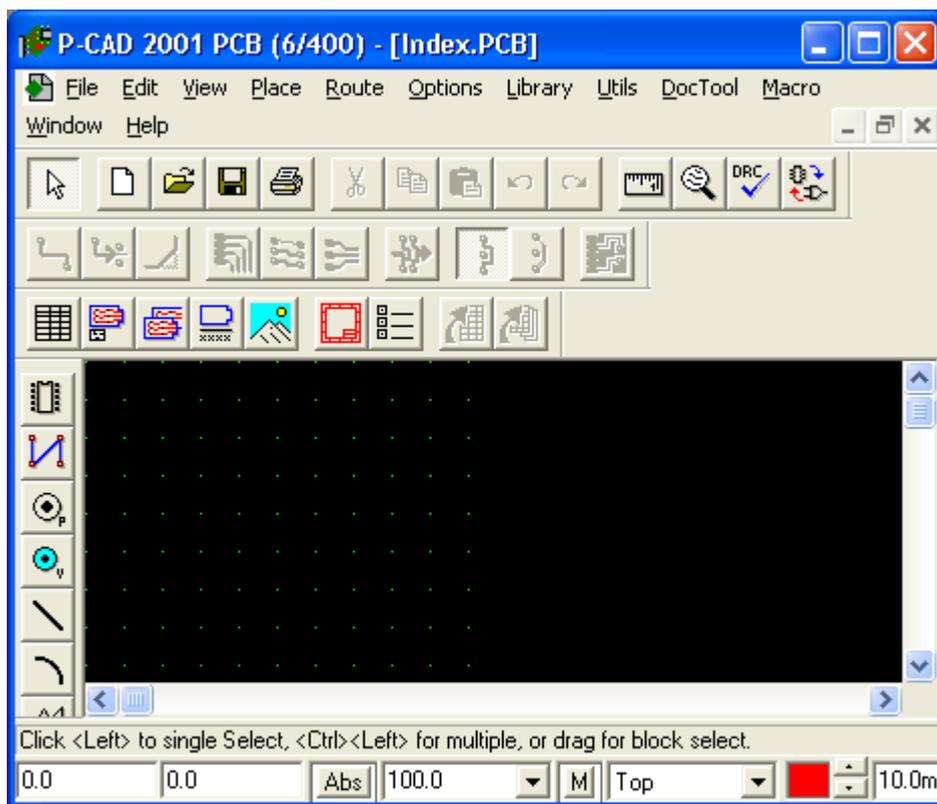


Рис. 7.2. Рабочий экран P-CAD Schematic

**Значения координат** вводят в милах (**mil**), миллиметрах (**mm**) или в дюймах (**inch**). Выбор системы единиц измерения производится при выполнении команды **Options/Configure/Units**. Если координаты точки заданы в **mil**, то точность — один десятичный знак, а если в **mm** - три десятичных знака после запятой.

**Кнопки переключения типа сетки ABS и Rel.** Абсолютная сетка **ABS** имеет начало координат в нижнем левом углу рабочей области экрана. Относительная сетка **Rel** имеет начало координат в точке, указанной пользователем. Сетка **Rel** включается в том случае, если в окне команды **Options/Grids** активен режим **Prompt for Origin**. Значение **шага сетки** устанавливается при нажатии на кнопку **выбора** (стрелка,), находящуюся справа от поля шага сетки. А набор шагов сеток устанавливается в поле **Grid Spacing** после выполнения команды **Options/Grids**.

При активизации **кнопки записи макрокоманд M** (или клавиши M) начинается запись во временный файл всех выполняемых команд. Повторное нажатие кнопки M (или клавиши M) прекращает запись файла с именем **\_default.mac**. Этот файл доступен только в течение *текущего сеанса*.

Поля **текущего имени схемы** и кнопка выбора **имени листа** отражают установки, проведенные по команде **Options/Sheets** в закладке **Sheets**. Все листы схемы одного проекта содержатся в одном файле с расширением **.sch**. Добавление листов в проект осуществляется командой **Options/Sheets/ Sheets/Add**.

Поля **ширина линии и выбор ширины линии** дублируют команду **Options/CurrentLine**. Для добавления в список новой толщины линии необходимо щелкнуть по кнопке **Line Width** и ввести новое значение толщины линии. *Тип линии* устанавливается командой **Options/ Current Line** в области **Style** диалогового окна.

**В строке сообщений** (справа от кнопки выбора ширины линий) отображается следующая *текущая* информация:

- тип, позиционное обозначение или общее количество выбранных объектов;
- значения приращений по осям X и Y при перемещении выбранных объектов;
- имя выбранной цепи;
- расстояние между выбранными точками и их проекции на оси X и Y при выполнении команды Edit/Measure.

## **1.2. Команды обзора**

Команды обзора рабочего окна сгруппированы в меню **View** и изменяют вид, масштаб изображения объекта и его положение в пределах рабочего окна.

Команда **View/Redraw** перечерчивает экран с целью удаления «следов», оставшихся после редактирования изображения.

Команда **View/Extent** масштабирует изображение так, что на экран выводится все введенные на данный момент фрагменты проекта.

Команда **View/Last** выводит предыдущее изображение экрана.

Команда **View/All** выводит на экран все содержимое активного окна вместе с рамкой.

Команда **View/Center** (или нажатие на клавишу **C**) центрирует изображение относительно текущего положения курсора. Эта же клавиша используется для панорамирования изображения при установке курсора на край экрана. Если курсор расположить на границе экрана, то смещение изображения (панорамирование) выполняется при нажатии одной из клавиш со стрелками ( **<—**, **—>** )

Команды **View/Zoom In** или **View/Zoom Out** увеличивают или уменьшают изображение относительно координаты, указанной курсором, на величину параметра **Zoom Factor**, указанного в меню **Options/Configure**. Изменение масштаба изображения удобнее производить при установке курсора в точку, относительно которой будет произведено изменение, и последующем нажатии на клавиши *серый +* или *серый -*. Команда **View/Jump Location** перемещает курсор в точку с координатами, которые указываются в окнах диалогового окна, появляющегося после выполнения указанной команды.

Размещение объектов в рабочем окне производится с помощью команд меню **Place**. Эти команды дублируются соответствующими пиктограммами на панели инструментов **Placement Toolbar**.

Команда **View/Snap to Grid** перемещает курсор только по узлам координатной сетки экрана. Это позволяет точно позиционировать точки привязки объектов.

При *размещении* (или рисовании) объекта вначале щелчком мыши выбирается соответствующая пиктограмма, а затем курсором указывается местоположение объекта (или начальная точка рисования объекта). Перед размещением некоторых объектов возможна операция ввода дополнительной информации (появляется соответствующее диалоговое окно). Точка *привязки* объекта устанавливается в узле сетки рабочего окна. Объекты в процессе размещения можно *перемещать*. Для этого, после указания начальной точки размещения (не отпуская кнопку мыши), необходимо протащить контур объекта в нужную позицию экрана и отпустить кнопку мыши.

Нажатие клавиши **B** в процессе размещения позволяет некоторые объекты развернуть против часовой стрелки на угол, кратный  $90^\circ$ .

Нажатие клавиши **F** в процессе размещения позволяет некоторые объекты *отобразить зеркально относительно оси Y*. Например, для редактора **P-CAD PCB** эта операция эквивалентна переносу компонента на противоположную сторону платы.

#### 1.4. Выбор и редактирование объектов

*Режим выбор объектов* активизируется при нажатии клавиши **S** или щелчком мыши по пиктограмме **Select**. Объект выбирается щелчком мы-

ши, при этом имя выбранного объекта и его данные выводятся в строку информации. Если один объект закрывает другой, то выбор невидимого объекта осуществляется повторным щелчком мыши или повторным нажатием на клавишу <Пробел> (при этом курсор должен находиться на выделенном объекте). Для добавления выбранных объектов к уже выделенным перед щелчком мыши нажимается и удерживается клавиша Ctrl. Щелчком мыши в свободной части рабочего окна выбор объектов отменяется.

Двойной щелчок левой клавишей по объекту позволяет редактировать все его *атрибуты*. Щелчок правой клавишей мыши вызывает *контекстно-зависимое меню* команд.

При *перемещении* объекта его можно сдвигать на один или несколько шагов сетки (не отпуская кнопку мыши нажимать соответствующую клавишу со стрелкой). Для изменения положения *точки привязки* после выбора объекта в меню команд редактирования компонента выбирают команду Selection Point и щелчком мыши устанавливают новое положение точки привязки.

После выбора объекты можно вращать (R), отображать зеркально (F), выравнивать (Align) по горизонтали и вертикали, копировать в буфер обмена (Copy) или в файл (Copy to File), вставлять из буфера или передвигать в нужное место рабочего поля.

В **P-CAD** можно выбрать отдельный элемент сложного компонента, пример вывод символа, схемное имя или номер контакта компонента. Для этого перед щелчком мыши необходимо нажать и удерживать клавишу Shift. После выделения объекта и последующего щелчка правой кнопкой мыши по объекту вызывается контекстное меню, пример которого показан на рис. 7.3.

Полный перечень команд контекстного меню таков:

- **Properties...** - просмотр и редактирование характеристик выбранного объекта;
- **Copy** - копирование объекта в буфер обмена;
- **Copy Matrix...** - множественное копирование объекта;
- **Cut** - удаление выбранного объекта с сохранением его копии в буфере обмена;
- **Delete** - удаление выбранного объекта;
- **Edit Nets...** - редактирование атрибутов цепи;
- **Select Contiguous** - выбор соприкасающихся элементов цепи;

- **Select Net** - выбор всей цепи (включая фрагменты цепи, связанные с общей шиной);
- **Net Info...** - вывод информации о цепи;

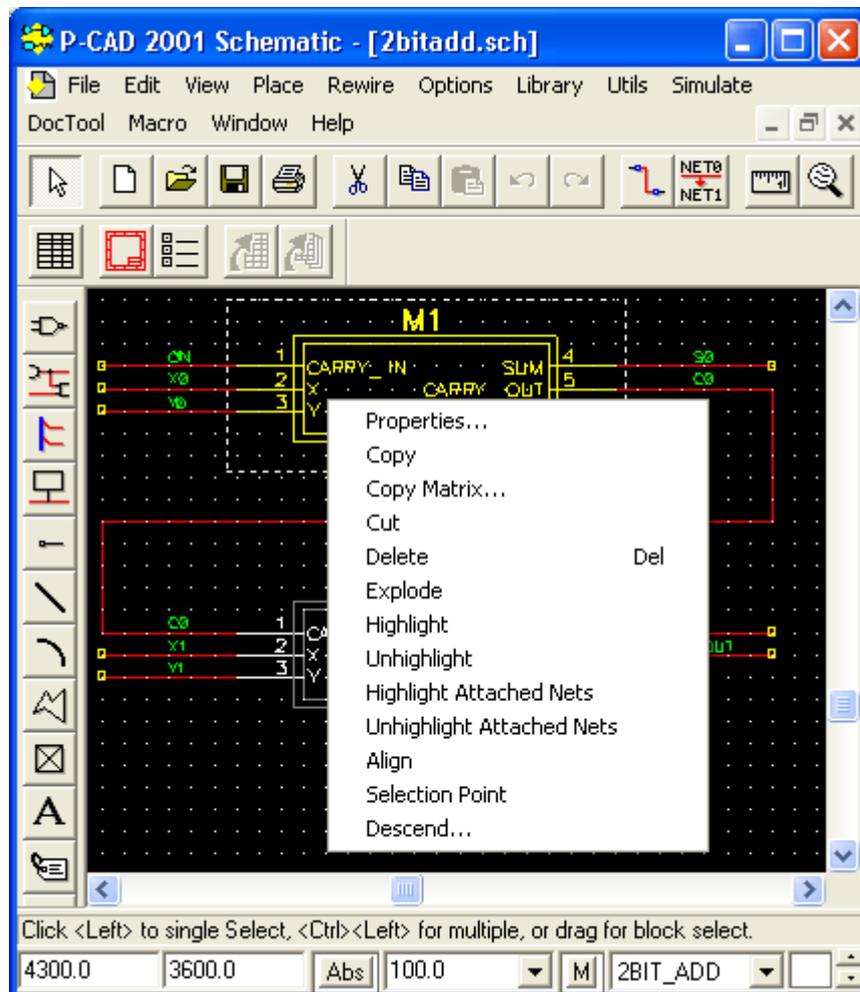


Рис. 7.3. Контекстное меню команд редактирования компонента

- **Highlight** - окрашивание выбранного объекта;
- **Unhighlight** - отмена предыдущей команды;
- **Highlight Attached Nets** - окрашивание цепей, подсоединенных к выбранным объектам;
- **Unhighlight Attached Nets** - отмена предыдущей команды;
- **Align** - выравнивание компонентов на рабочем поле;
- **Selection Point** - изменение положения точки привязки выбранного объекта или группы объектов.

Дополнительные возможности по выбору и редактированию параметров выбранных объектов предоставляют команда **Options/Block Selection** для редактора **P-CAD Schematic**, и команда **Options/Selection Mask** для редактора **P-CAD PCB**, которые настраивает фильтры и режимы выбора параметров. Диалоговые окна указанных команд для соответствующих редакторов представлены на рис. 7.4 и рис. 7.5.

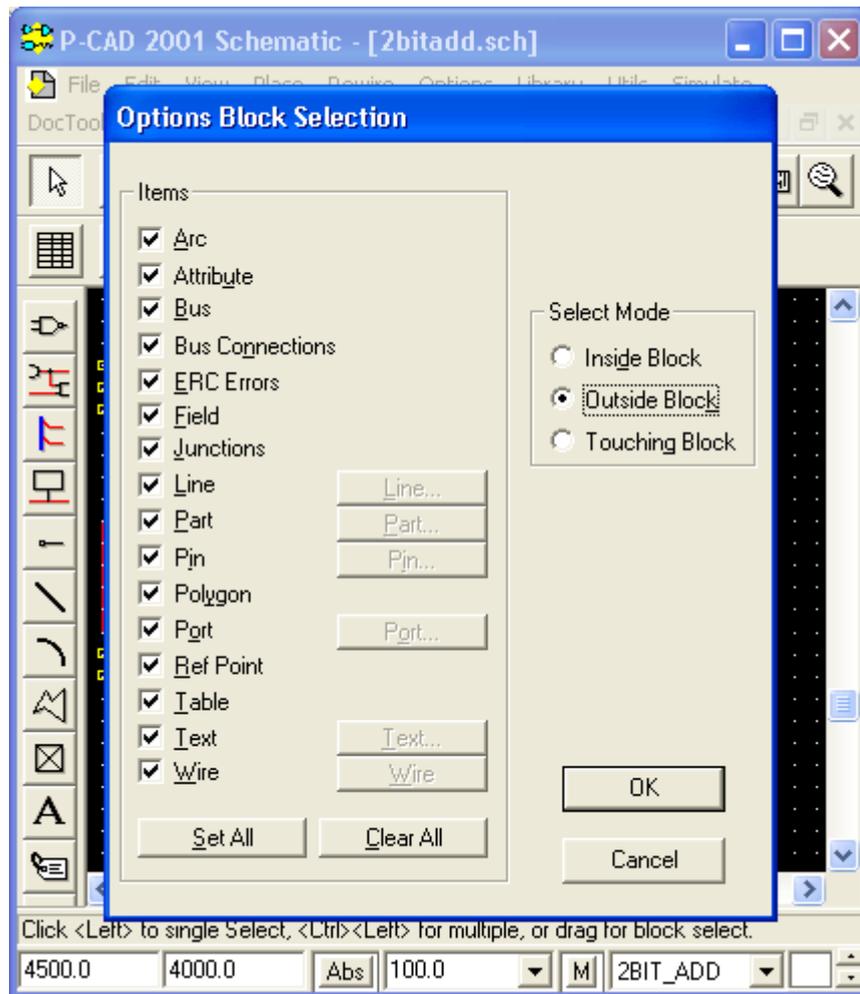


Рис. 7.4. Настройка параметров блока выбора редактора **P-CAD Schematic**

В секции **Items** (Элемент) диалогового окна указывают элементы, которые должны быть выбраны после выполнения операции блочного выбора. Некоторые элементы левого столбца имеют кнопки для задания дополнительных параметров.

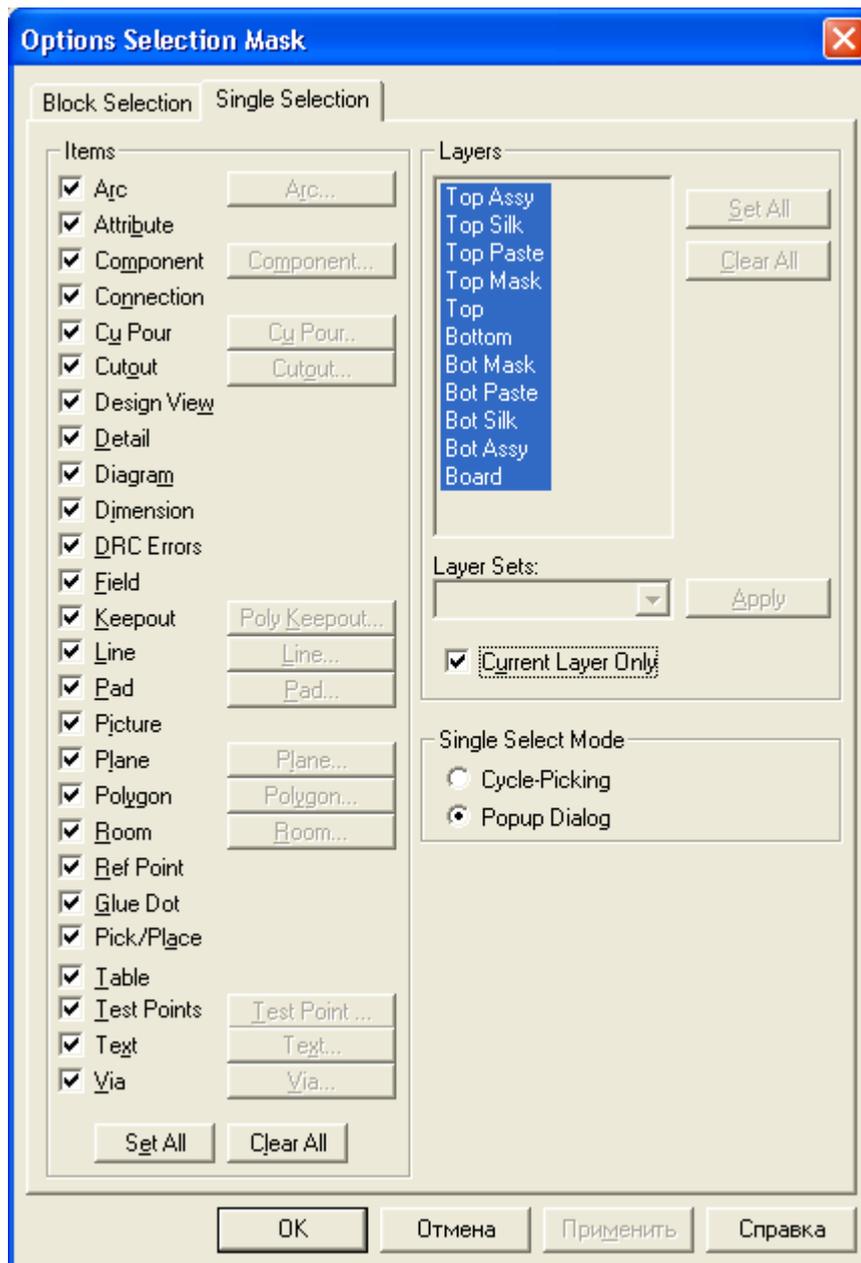


Рис. 7.5. Настройка параметров блока выбора редактора P-CAD PCB

В секции **Select Mode** можно задать способ выбора: **Inside Block** - выбор элементов, находящихся внутри блока, **Outside Block** — выбор всех элементов, находящихся вне блока, **Touching Block** — выбор всех элементов, находящихся внутри блока и касающихся его контура. Выбор объектов, удовлетворяющих заданным условиям, производится заданием на рисунке прямоугольного контура с помощью мыши. Все выбранные объекты

подсвечиваются, и в дальнейшем их можно копировать, перемещать и редактировать.

**При копировании** графические данные из буфера обмена можно передавать в другие программы, работающие под управлением **Windows: MS Word, Paintbrush** и др. Эти же данные не *передаются* из графического редактора **P-CAD PCB** в редактор **P-CAD SCHEMATIC** и наоборот. Для ускорения копирования в пределах одного листа необходимо выбрать нужный объект, нажать клавишу **Ctrl**, и, не отпуская левую кнопку мыши, переместить копию объекта на новое место.

**Изменение размеров объекта** производится после его выделения, удерживанием курсора на одной из узловых точек объекта и последующим перемещением курсора в нужном направлении. Изменяются, таким образом, размеры линий, проводников, шин питания и полигоны при захвате концов или изломов объектов.

При изучении системы **P-CAD** для проектирования ПП будем придерживаться следующей последовательности изложения материала:

- создание символов компонентов (программа **SYMBOL EDITOR**, исполняемый файл **SYMED.EXE**);
- создание посадочных мест компонентов (программа **PATTERN EDITOR**, исполняемый файл **PATED.EXE**);
- правила ведения библиотек (программа **LIBRARY EXECUTIVE**, исполняемый файл **СMP.EXE**);
- создание принципиальных электрических схем, их верификация, создание отчетов и вывод результатов на печать (графический редактор **P-CAD SCHEMATIC**);
- размещение компонентов на монтажно-коммутационном поле, ручная трассировка электрических цепей, верификация ПП, оформление чертежей, генерация отчетов и вывод на печать (графический редактор **P-CAD PCB**);
- использование программ автоматической трассировки **QUICKROUTE, PROROUTE** и **SHAPE-BASED ROUTING**.

## 1.5. Структура библиотек P-CAD

Библиотеки в системе **P-CAD** называются *интегрированными* по той причине, что библиотека (**\*.lib**), помимо информации о рисунке сим-

вола компонента на электрической схеме и посадочном месте компонента на печатной плате, содержит также и текстовую информацию о внутренней структуре и функциях отдельных составляющих компонента. Заметим, что символ компонента и его посадочное место можно записывать не только в библиотеку, но и в отдельные файлы с расширениями \*.sym и \*.pat соответственно.

Каждый компонент состоит из одной или нескольких (**одинаковых или различных**) логических секций (вентилей), которые упаковываются в один корпус. Компоненты с разными именами могут иметь одну и ту же графику корпуса или символа. Корпусы и символы в этом случае должны находиться в одной и той же библиотеке.

При создании компонентов введены следующие обозначения:

- **Pad Numbers** - номер вывода (контактной площадки) компонента;
- **Pin Designator** - позиционное обозначение вывода компонента на схеме (может отличаться от **Pad Numbers**);
- **Symbol Pin Numbers** - номер вывода в секции (вентиле) компонента;
- **Pin Names** — имя вывода в секции (вентиле) компонента.

Отметим, что в систему **P-CAD** включены несколько десятков интегрированных библиотек компонентов (увы, не отвечающих требованиям российских ГОСТов), компоненты которых в принципе можно отредактировать до параметров, нужных для текущего проекта. В частности, библиотека **PCBMAIN.LIB** содержит информацию о корпусах дискретных компонентов и интегральных схем со *штыревыми* выводами, **PCBSMT.LIB** - корпуса дискретных компонентов и интегральных схем с *планарными* выводами, **PCBCON.LIB** - разъемы.

Для создания библиотечных элементов при проектировании электронных систем необходимо реализовать следующие шаги:

- создание схемного (символьного) образа компонента и запись его в библиотеку;
- создание стеков контактных площадок установочных мест компонентов;
- создание посадочных мест элементов со штыревыми и планарными выводами для дальнейшего размещения их на монтажно-коммутационном поле;

- запись посадочных мест компонентов и стеков контактных площадок в библиотеки;
- создание взаимосвязи между символами элементов и их посадочными местами.

## Лекция № 8

### Система схемотехнического моделирования Micro-CAP

#### Введение

Электронные устройства (схемы) применяются повсеместно - в быту, науке и технике, в промышленности, связи и транспорте. По богатству и разнообразию их типов они сравнимы разве что с имеющими многовековую историю механическими устройствами. Физические процессы, протекающие в схемах сложны и недоступны для обычных органов чувств человека. Лишь их математический анализ и исследования с помощью специальных приборов позволяет понять сложные физические закономерности работы электронных устройств.

Еще совсем недавно инженер-схемотехник представлялся в образе "колдуна с паяльником", сидящего за столом, заваленным горками деталей и многочисленными заумными электронными измерительными приборами. Время от времени он извергал проклятия по поводу сгоревших при его экспериментах электронных ламп, транзисторов, микросхем или резисторов и делился с друзьями мыслями о том, где и как добыть новый скоростной осциллограф, сверхчувствительный вольтметр или высокочастотный частотомер.

С возникновением микроэлектроники стало ясно, что инженеры-практики схемотехнического профиля выходят из моды, поскольку отмакетировать микроэлектронное устройство традиционными способами оказывается попросту невозможно. Слишком дорого стали обходятся и эксперименты с новыми типами мощных биполярных и полевых транзисторов. Достаточно кратковременных перегрузок (порою в тысячные доли секунды) чтобы эти дорогие и дефицитные приборы пополнили горку металлолома, из коей полезны лишь тысячные доли грамма драгоценных металлов.

А исследование схем на множестве макетов с целью прогнозирования серийного выпуска (метод Монте-Карло) обходится, как говорится, "в копейку".

Существует немало схем (например, активные фильтры) проектирование которых невозможно без привлечения сложных (и как правило недоступных практику) математических расчетов. Отладка таких схем на макетах не позволяет получить оптимальные результаты.

Актуальным всегда был и вопрос о доводке схемы до нужного уровня надежности и долговечности, обеспечения нечувствительности к изменению параметров различных компонентов. Все это было несовместимо с обычной эмпирической разработкой электронных схем, которая все больше отдалялась от серьезной науки и превращалась скорее в "искусство схемотехники", чем в хорошо прогнозируемое научное или техническое ремесло (в лучшем смысле этого слова).

Современное схемотехническое проектирование уже невозможно без машинного расчета и проектирования электронных схем. Отечественная литература по этому направлению представлена широким спектром книг - от расчетов отдельных схем на Бейсике до описания универсальных алгоритмов, машинных методов расчета и систем автоматического проектирования электронных схем.

К сожалению, из всей этой литературы выпала поистине "золотая середина" - описание конкретных систем схемотехнического проектирования, ориентированных на современные персональные компьютеры класса IBM PC.

Не спасает положение и разработка дружественных по отношению к пользователю программ. Любая серьезная интегрированная программная система требует от пользователя значительных усилий по ее изучению и практическому применению, знания множества команд и профессиональных навыков применения. Поэтому пользователь постоянно нуждается в руководствах по применению программных средств и в другой технической литературе.

Рассмотрим семейство интегрированных систем моделирования и автоматизированного анализа электронных схем фирмы Spectrum Software Micro-CAP и Micro-LOGIC, позволяющих моделировать широкий класс аналоговых, логических и цифровых схем. Эти программы имеют превосходный графический интерфейс связи с пользователем и ориентированны

на наиболее массовые и достаточно мощные профессиональные персональные компьютеры класса IBM PC.

Системы Micro-CAP II различных версий способны работать даже на простых ПК серии IBM PC (например IBM PC XT, Amstrad PC 512, Euro PC и т.д.). Их отличает крайняя простота и удобство графического диалога. Освоение таких систем при наличии толкового описания (авторы надеются, что к таковым относится эта книга) занимает буквально несколько часов. Одна из последних версий таких систем (4.0) имеет обширную библиотеку компонентов (по 100 диодов, биполярных транзисторов, полевых транзисторов и операционных усилителей) и позволяет моделировать широкий круг электронных схем.

Новые большие возможности для проектирования, изучения и анализа электронных схем предоставляют последние версии Micro-CAP III, графический редактор которых ориентирован на работу с мышкой (хотя и сохраняет возможность работы с клавиатурой). Помимо значительно большего быстродействия, система Micro-CAP III представляет пользователю множество дополнительных возможностей: современный многооконный интерфейс, выпадающие меню и каталоги файлов и компонентов, встроенный калькулятор, удобную систему подсказок, возможность нанесения на рисунки схем сопроводительных надписей и т.д. Предусмотрена трансляция библиотек компонентов и схемных данных из более ранних версий а также подготовка файлов схем для работы с другой популярной системой моделирования PSPICE (практически аналогичной широко известной в эру применения больших ЭВМ программе SPICE [4]).

Для настройки моделей компонентов в системы семейства Micro-CAP введены специальные программы расчета параметров компонентов PER (Parameter Estimation Program), которые по справочным данным на прибор автоматически составляют таблицу параметров его модели и записывают их в специальную библиотеку. Пользователь может не только проанализировать полученные значения параметров, но и изменять их для получения желаемого сходства статических характеристик прибора с характеристиками модели.

Дискуссия о практической пользе машинного моделирования электронных схем не затихает и сейчас. Одни считают, что такое моделирование полностью заменяет натурные испытания схем и даже считают его достаточным для обучения студентов вузов основам электронной схемотехники.

техники. Другие резко отвергают полезность моделирования, считая (не без серьезных оснований), что оно не учитывает массу нюансов практической работы схем. Но с тем же успехом можно сказать, что наблюдаемые экспериментально результаты всегда относятся отнюдь не к той схеме, которую инженер якобы исследует, а к гораздо более сложной и подчас ему неизвестной, ибо невозможно учесть все паразитные параметры, присущие каждому компоненту схемы. И только моделирование позволяет отсеять эти паразитные параметры либо учесть их с той или иной степенью достоверности

## **1. Интегрированная система Micro-CAP III**

### **1.1. Основные сведения о системе Micro-CAP III**

Система Micro-CAP III позволяет быстро вводить графическое изображение схем и работать с меню либо с помощью мышки, маркер которой перемещается по экрану и позициям меню, либо с помощью клавиатуры. Micro-CAP III имеет довольно широкий набор средств редактирования. Например, имеется возможность переносить отдельные элементы схемы по экрану, выделять фрагмент схемы в "ящик", с которым можно проводить операции редактирования отдельно, плавно перемещать все изображение в любом направлении и др..

Позиционирование элементов схем возможно с высокой точностью, а не с точностью их большого "элементоместа", как в системе Micro-CAP II. Это, наряду с более совершенными обозначениями элементов схем (кстати допускающими редактирование), позволяет получать качественные стандартные изображения принципиальных схем на экране дисплея и на распечатках принтером. Намного расширены возможности редактирования схем и параметров моделей.

В системе предусмотрена возможность нанесения на схему поясняющих надписей (в том числе на русском языке), что делает информацию о схеме более доступной для других пользователей. Однако следует отметить, что встроенный клавиатурный драйвер системы не всегда позволяет вводить малую букву "р". Впрочем, ее легко вставить с помощью любого текстового редактора прямо в схемный файл с расширением .cir.

Micro-CAP III позволяет вести отдельную "записную книжку" для каждой рассчитываемой схемы и при необходимости пользоваться комментариями для пояснения ее особенностей. Встроенный калькулятор обеспечивает проведение простых математических вычислений по ходу анализа схемы, что является весьма полезным при анализе результатов моделирования или при прикидках данных к нему.

Имеется возможность не только выводить данные моделирования на экран дисплея, принтер или на диск, но и проводить оперативную математическую обработку данных - например для их представления в логарифмическом или ином масштабе.

В систему входят программы для задания параметров моделей и создания графических изображений элементов. Имеются возможности переноса библиотек компонентов и схем из системы Micro-CAP III в другую профессиональную программу анализа электронных схем PSPICE.

Наиболее важным достоинством Micro-CAP III является возможность задания практически любых аналитических зависимостей (например временных или ВАХ) с помощью функций пользователя, что необходимо для создания собственных математических моделей новых приборов. В результате система стала расширяемой и в ней можно моделировать схемы с принципиально новыми приборами и их моделями. Наконец, математический аппарат Micro-CAP III существенно усилен по сравнению с предыдущими версиями, что позволяет использовать более точные математические модели основных полупроводниковых приборов (аналогичные применяемым в системе PSPICE).

Таким образом, Micro-CAP III можно отнести к новому поколению систем автоматизированного анализа и моделирования электронных схем, которое по производительности и удобству приближаются, а иногда и превосходят широко известные схемотехнические системы автоматического проектирования класса NAP и PSPICE. На базе системы Micro-CAP III возможно создание современного автоматизированного рабочего места (АРМ) схемотехника. При этом разработанные схемные решения могут передаваться в САПР проектирования печатных плат и изготовления. Тем самым возможна реализация сквозного цикла проектирования и выпуска устройств.

## 1.2. Работа с меню системы

Режимы конструирования и анализа схем задаются с помощью современной многооконной системы представления информации и вывода меню, которая при начальной загрузке имеет вид " Рис. 3.1. Система меню Micro-CAP III " (исполнена команда Micro-CAP III позиции "мю"):

Основное меню расположено в верхней строке экрана. Каждый пункт основного меню имеет свое выпадающее подменю, которое раскрывается после указания соответствующего пункта. Для этого достаточно нажать клавишу с заглавной буквой одной из позиций основного меню (позиция "мю" активизируется одновременным нажатием клавиш Alt и U). Теперь клавишами перемещения курсора по горизонтали можно выбирать нужную позицию основного меню. Нажатие клавиши ENTER приводит к появлению вспомогательного подменю. Клавиши перемещения курсора вверх и вниз позволяют перемещаться по позициям дополнительных подменю и включать их команды. На " Рис.3.2. Результат исполнения команды Load позиции Files " показан вид переднего панно системы после нажатия клавиш F (Files) и L (Load).

В данном случае в центре экрана появляется дополнительное окно с указанием наименований дисков, их директорий и перечнем файлов активной директории. Стоит мышкой выделить нужный файл, как произойдет его загрузка. Так можно просмотреть тестовые примеры системы. Можно отказаться от текущей команды нажав клавишу Esc или подведя маркер мышки в пустой прямоугольничек в верхней части окна и нажав левую клавишу мышки.

Итак, в центре экрана при загруженном драйвере мышки виден ее маркер в виде жирной стрелки. Его можно перемещать перемещением мышки. Подобным образом нетрудно быстро найти нужную позицию меню и (нажав левую клавишу мышки) вызвать появление подменю. Не отпуская клавишу можно перемещать маркер по позициям подменю. Отпускание клавиши делает текущую позицию подменю активной и ведет к выполнению соответствующей команды (см. далее).

Кроме того, на экране присутствуют два дополнительных окна: окно с меню режимов редактирования Mode (левое вертикальное окно) и окно с перечнем компонентов Components (правое вертикальное окно). Схема конструируется в центральном окне. Окна можно перемещать по экрану с помо-

щью левой клавиши мышки, окно схем можно распахнуть на весь экран. Открытие и закрытие окон осуществляется пунктом Windows основного меню.

В среднем большом окне под строкой основного меню находится еще одна служебная строка. В середине ее имеется название текущего файла (по умолчанию UNTITLED), а по краям (если окно активно) четыре небольших прямоугольника. С левого края это пустой прямоугольник и прямоугольник с буквой H. Они служат для управления работой Micro-CAP III с помощью мышки (для этой цели такие прямоугольники есть в большинстве других окон). Если поместить маркер мышки в прямоугольник с буквой H (от Help - помощь) и нажать левую клавишу мышки, то будет выведена подсказка системы о назначении окна. А для выхода из окна или режима подсказки достаточно подвести маркер к пустому прямоугольнику.

Прямоугольник с сеткой позволяет передвигать изображение по экрану в направлении, указанном положением маленького квадратика, выбранного маркером мышки в этом прямоугольнике. При этом перемещения возможны как по вертикали и горизонтали, так и по диагонали. Наконец, поле выбора (вложенные маленькие прямоугольники в крайнем правом квадратике) позволяет осуществить модификацию режима вывода информации на экран путем двукратного нажатия любой из клавиш мышки (например, распахнуть окно схемы на полный экран или вновь поместить схему в центре экрана и т.д.). С помощью мышки можно "захватить" окно и переместить его в другое место экрана.

Подобная идеология управления применяется и при работе с другими окнами. Это облегчает практическое знакомство с управлением системой.

При работе без мышки указанные операции выполняются при использовании клавиш функциональной клавиатуры:

Выключение окон осуществляется нажатием ESC, повторное включение - через пункт меню Windows. Для перемещения и изменения размеров окон с клавишного пульта можно использовать клавиши Alt+F7 и Alt+F8 совместно с клавишами перемещения курсора. Помните что, если вы случайно закрыли окно и оно исчезло с экрана дисплея то включение окна в пункте Windows вернет окно на место.

***Рассмотрим пункты основного меню Micro-CAP III и вводимые ими команды управления системы.***

Пункт меню "**мю**" вызывается одновременным нажатием клавиш Alt и U (или его указанием маркером мышки) и раскрывает подменю режимов работы Micro-CAP III. Оно содержит следующие пункты:

- 1: Micro-CAP III - сообщение об авторах программы, дате создания, информация о текущем времени и дате (см. окно в центре экрана на рис.3.1).
- 2: Notes - заметки, которые пользователь может сделать в специальном окне для описания рассматриваемой схемы. Заметки сохраняются вместе с файлом схемы. Для записи заметок открывается специальное окно и в него в общепринятом порядке вводится необходимый текст. Имеются типовые возможности его редактирования.
- 3: Calculator - калькулятор, который позволяет пользователю по ходу моделирования схемы сделать простейшие арифметические и алгебраические вычисления.
- 4: Help - вызов подсказки на английском языке.
- 5: Quit F3 - выход из Micro-CAP III, который осуществляется и нажатием функциональной клавиши F3. Если в окне схемы были изменения, то перед выходом система запросит о необходимости записи схемы или ее отсутствии. Установки системы, именуемые опциями (например цветовая гамма экрана), в момент выхода запоминаются, поэтому последующее включение системы уже не требует их новой установки.

Для отмены действующего пункта меню следует либо нажать клавишу ESC, либо воспользоваться мышкой. Выбор пункта подменю, как отмечалось выше, осуществляется или нажатием цифровой клавиши, или протяжкой маркера мышки от пункта основного меню к пункту подменю и отпусканьем левой клавиши мышки.

Подменю позиции **Files** <файлы> основного меню служит для работы с файлами. Оно содержит указанные ниже позиции.

- 1: Create new circuit - создание новой схемы. При этом окно схемы становится активным, и в нем можно конструировать схему. Обычно этот пункт используется при вводе новой схемы после включения системы.
- 2: Load circuit - загрузка схемного файла с диска. Имя файла либо задается пользователем, либо выбирается из списка имен, имеющихся на диске в указанном директории. Список файлов выводится на экран в зависимости от установки опции File dialog box пункта меню Options (см. ниже).

- 3: Unload circuit - удаление схемы из оперативной памяти (при этом очищается окно схемы).

- 4: Save circuit - сохранение схемного файла на диске. Пользователь получает запрос об указании имени файла. По умолчанию задается имя UNTITLED или имя схемы, которая была загружена по позиции 2.

- 5: Erase circuit - удаление схемного файла с диска, пользователь либо указывает имя удаляемого файла схемы, либо выбирает его из списка (см. пункт Load circuit). Эта команда незаменима, если вы намерены привести в порядок свою библиотеку схемных файлов.

- 6: Merge circuit - объединение схемных файлов, пользователь получает возможность добавить к уже существующей на экране схеме новую, записанную ранее на диск. Как мы увидим далее, система позволяет моделировать сразу несколько схем, имеющихся на экране дисплея, и даже одновременно выводить результаты их работы для сравнения.

- 7: Load library - загрузка библиотеки с диска, необходимая в случае, если происходит обращение к новой библиотеке. Опыт показывает, что удобно хранить библиотечные файлы для определенного класса схем вместе с их схемными файлами. Это минимизирует необходимость в использовании данной команды.

- 8: Save library as - сохранение библиотеки под новым именем.

- 9: Change data path - изменение пути данных для схемных файлов.

Редактирование библиотечных данных системы Micro-CAP III осуществляется с помощью пункта меню **Editors**. Он порождает подменю с рядом позиций.

- 1: Library devises - библиотечные компоненты (транзисторы, операционные усилители, диоды, программируемые источники и т.п.). Пользователю предоставляется список типов компонентов и таблица параметров выбранного типа компонента. Ввиду большого числа относящихся к компонентам данных нет возможности описать все нюансы их задания и корректировки. Однако, они проводятся по той же идеологии, что и работа с системой, так что задание новых параметров компонентов или их модификация легко осуществляются уже при небольшом навыке работы с системой. Оно особенно просто при использовании мышки. При ее отсутствии для переключения библиотек используется одновременное нажатие клавиш Shift и Tab. Далее клавишами Tab и перемещения курсора можно пе-

решать маркер по позициям меню и активизировать их нажатием клавиши ENTER.

- 2: Library labels - пассивные компоненты (резисторы, независимые источники тока и напряжения, конденсаторы, индуктивности, линии задержки, трансформаторы, ключи), формат записи параметров соответствует формату Micro- CAP II. Переход с поля имени компонента на поле параметров осуществляется клавишей табуляции.

- 3: Library functions - библиотечные функции (математические выражения, описывающие временные зависимости в формате языка BASIC). Список допустимых математических и логических операторов, а также операторов сравнения дается в Help.

Позиция **Print** основного меню служит для поддержки работы с принтером. Если она активна, то появляется подменю с указанными ниже позициями.

- 1: Graphic: circuit - печать схемы.
- 2: Graphic: front window - печать содержимого активного открытого окна, которым может быть изображение схемы, ее топологическая структура, таблица подменю и т.п.;
- 3: Graphic: screen - печать полной копии экрана.
- 4: Text: netlist - вывод на экран таблицы топологической структуры схемы для последующей ее распечатки. Эта таблица показывает какие компоненты используются в схеме и между какими узлами они включаются.

Пункт основного меню **View** управляет обзором информации, которую можно наблюдать на экране дисплея. При его активности появляется подменю с рядом позиций.

- 1: Capacity used - вывод информации о количестве элементов и текстовых надписей в схеме.
- 2: Scale 1:1 - установка масштаба вывода схемы 1:1.
- 
- 4: Scale 4:1 - установка масштаба вывода схемы 4:1.
- 5: Show comp text - вывод на экран обозначений (имен) компонентов.
- 6: Show grid text - вывод на экран поясняющего текста.
- 7: Show node numbers - вывод номеров узлов схемы.
- 8: Search - поиск компонента в библиотеке.

- 9: Repeat search - повторный поиск.

Обратите особое внимание на пункт 7 подменю View, поскольку по умолчанию номера узлов схем не проставляются. Система автоматически назначает номера узлов и они могут меняться при добавлении в схему новых компонентов. В дальнейшем при организации вывода данных всегда необходимо знать к каким узлам относятся токи и напряжения. Поэтому вывод номеров узлов необходим и приводится для всех описанных далее схем.

Включенная позиция подменю View помечается символом "птичка" напротив соответствующей надписи (" Рис.3.3. Вид переднего панно с подменю позиции View ").

Пункт меню **Options** обеспечивает установку основных опций работы Micro- CAP III. Оно содержит перечисленные далее позиции.

- 1: Palette - палитра цветов (Micro-CAP III позволяет хранить 3 набора цветов): планшета (Desktop), заднего плана окон (Background), переднего плана окон (Foreground), надписей на окнах (Static), элементов и соединений на схеме (Shapes), поясняющего текста на схемах (Text). Выбор желаемого набора осуществляется мышкой или табулятором с помощью поля Select, фиксация набора - клавишей "пробел", сохранение набора - с помощью поля Save.

- 2: Preferences - установка желательных режимов диалога: звуковое сообщение об ошибках, вывод списка файлов и т.д.

- 3: Global setting - установка общих параметров анализа (число итераций, погрешности и т.п.).

- 4: Plotter - установка параметров графического плоттера.

- 5: Models - типы моделей биполярного и полевого транзисторов (Эберса - Молла, Гуммеля - Пуна, модели полевых транзисторов разного уровня сложности).

- 6: Output - опции режима вывода выходной информации: на дисплей, в файл или принтер. Соответствующее подменю позволяет выбрать подходящий принтер (включая лазерные принтеры и плоттеры) и установить порт ввода-вывода информации.

При выводе информации в файл Micro-CAP III автоматически формирует составное имя файла, тип которого определяется расширением и зави-

сит от режима анализа и характера выводимых величин. **При выводе числовых значений результатов** полное имя файла имеет вид:

**(имя схемы).(расширение).**

Имя схемы определяется пользователем при конструировании схемы, расширениями могут быть:

- .tpo - численные значения результатов анализа переходных процессов.
- .apo - численные значения результатов малосигнального анализа.
- .dpo - численные значения результатов анализа по постоянному току.
- .fpo - численные значения результатов Фурье-анализа.
- .tmc - численные значения результатов статистического анализа переходных процессов.
- .amc - численные значения результатов статистического малосигнального анализа.
- .dmc - численные значения результатов статистического анализа по постоянному току.

В файл может быть выведена таблица параметров режима анализа, при этом расширения имеют вид:

- .tlm - таблица параметров анализа переходных процессов.
- .alm - таблица параметров малосигнального анализа.
- .dlm - таблица параметров анализа по постоянному току.

Файл схемы имеет расширение .cir, структура схемы Netlist (список компонентов, их параметры и номера узлов) записывается в файл с расширением .doc.

Для сохранения начальных значений при анализе переходных процессов формируется файл с расширением .top.

Для обеспечения режима анализа Фурье данные предварительно записываются в файл с расширением .usr.

**При выводе графиков составное имя файла** имеет вид:

**(имя схемы)(номер расчета).(расширение).**

Номер серии расчетов - четырехзначный номер, две первые цифры которого определяются номером серии расчетов при статистическом анализе (см. режимы анализа), а две вторые - номером шага изменения параметра схемы (Stepping). Расширениями могут быть:

- .tsa - графики анализа переходных процессов.

- .asa - графики малосигнального анализа.
- .dsa - графики анализа по постоянному току.

Пункт основного меню Windows служит для включения и выключения главных окон системы. Он порождает подменю со следующими позициями.

- 1: Mode - включение или выключение окна режимов редактирования.
- 2: Components - включение или выключение окна компонентов.
- A: "имя схемы" - включение или выключение окна для конструирования схемы с заданным именем.

Включенное окно помечается знаком "птичка"

Запуск процедур анализа схемы осуществляется пунктом Run основного меню. Оно порождает подменю с указанными ниже позициями.

- 1: Transient - режим анализа переходных процессов.
- 2: AC - режима малосигнального анализа.
- 3: DC - режима анализа по постоянному току.
- 4: Fourier - Фурье-анализ.
- 5: Quit Analysis F3 - выход из анализа - F3.

### 1.3. Конструирование и редактирование схем

Создание (конструирование) схем выполняется в центральном окне системы путем выбора изображений нужных компонентов и размещения их в нужном месте экрана. При этом приходится выбирать компоненты из библиотек, перемещать изображения компонентов, поворачивать их вокруг базовой точки, указывать их имена и номиналы. Все это можно делать с клавишного пульта или с помощью мышки. Последнее намного удобнее.

Выбор режимов редактирования осуществляется командами, содержащимися в пунктах окна режимов редактирования **Mode** (левое вертикальное окно). Оно включает в себя пять опций и четыре указателя типа элемента, над которым осуществляется операция. Опции, перечисленные в верхней части окна, имеют следующее назначение:

Add - добавление нового компонента, линии, поясняющего текста или содержимого предварительно определенного фрагмента схемы ("ящика").

Zip - стирание компонента, линии, текста или содержимого "ящика".

Def - определение (переопределение) параметров компонента, текста или границ "ящика".

Move - перемещение компонента, линии, текста или "ящика".

Step - установка шага увеличения размеров "ящика" (по горизонтали, вертикали или одновременно по обеим координатам), а также копирование содержимого "ящика".

Info - информация о библиотечном компоненте схемы (вывод параметров компонента из библиотеки). Эта команда особенно удобна при редактировании схем, когда надо, например, изменить параметры модели некоторого компонента (диода, транзистора, операционного усилителя и др.).

Правое вертикальное окно (компонентов) используется для выбора компонента, достраиваемого в схему, а также для редактирования изображения компонента.

Например, для внесения в схему транзистора NPN типа, достаточно выбрать левой клавишей мышки в правом окне компонент NPN, перевести курсор мышки в соответствующую точку схемы и вновь нажать левую клавишу.

При вводе других компонентов (например, резисторов), будет выведен список пассивных компонентов из библиотеки. Если требуется ввести лишь номинал элемента, то необходимо указать метку Own (пользователь) и в появившемся поле ввести номинал.

Правое окно компонентов может также использоваться для редактирования начертания компонентов и изменения положения текста, обозначающего его имя. Для определения компонента имеется 8 основных полей:

Name - имя компонента, которое присутствует в правом окне компонентов и используется для вызова компонента при конструировании схемы. Имя указывается при редактировании начертания компонента;

Definitions - электрическое определение связей между токами и напряжениями в элементе. В Micro-CAP III имеются стандартные описания (модели) следующих активных и пассивных компонентов:

- NULL - нулевой элемент (отсутствие связи),
- BATTERY - источник э.д.с.,
- RESISTOR - резистор,
- CAPASITOR - конденсатор,
- INDUCTOR - индуктивность,

- DIODE - диод,
- NPN TRANSISTOR - транзистор npn - типа,
- PNP TRANSISTOR - транзистор pnp - типа,
- PMOS TRANSISTOR - МОП-транзистор с каналом P-типа,
- NMOS TRANSISTOR - МОП-транзистор с каналом N-типа,
- TRANSFORMER - трансформатор,
- POLYSOURCE - полиномиально-зависимый источник,
- SWITCH - ключ,
- PULSE SOURCE - программируемый источник,
- SHORT - переключатель,
- LINE - линия задержки,
- USER SOURCE - источник заданный пользователем,
- SINE SOURCE - синусоидальный источник,
- GROUND - "земля",
- OPAMP - операционный усилитель,
- JUMPER - пересечение без соединения,
- ISOURCE - источник тока,
- MACRO - подплата (макроопределение пламы),
- TIE - соединитель,
- PJFET - полевой транзистор с управляющим переходом P-канальный,
- NJFET - полевой транзистор с управляющим переходом N-канальный,
- IOFV - зависимый источник тока, управляемый напряжением,
- IOFI - зависимый источник тока, управляемый током,
- VOFV - зависимый источник напряжения, управляемый напряжением,
- VOFI - зависимый источник напряжения, управляемый током,
- USER FUNCTION - функция пользователя.

Пользователь имеет также возможность формировать собственные макроплады для нестандартных компонентов (MACRO). Для определения макроплады необходимо предварительно создать ее пламу и сохранить под именем, соответствующем имени нестандартного компонента. Большая часть компонентов, показанных на рис.3.5 и представляет такие компоненты (подплаты или макроопределения).

**Shape names** - список имен изображений компонентов, которые имеются в библиотеке. Имеются стандартные изображения всех перечисленных выше компонентов. Кроме того, для удобства работы имеются изображения соединительных линий двойной, тройной и т.д. длины, обозначенных SHORT2, SHORT3 и т.д., соответственно; пересекающихся линий без соединения (CROSS., JUMPER.), обобщенного изображения блока (BLOCK), инвертора (INV), логической схемы ИЛИ (OR), И (AND), полиномиальных источников, управляемых напряжением (POLY V) и током (POLY I). При необходимости библиотеку можно пополнять путем изображения новых элементов с помощью редактора изображений sh.exe, входящего в состав Micro-CAP III

Для коррекции расположения текста можно использовать левую клавишу мыши для элемента ориентации A и правую клавишу - для элемента ориентации B.

Допустим, что вы только загрузили систему и получили изображение трех окон и основного меню - см. рис. 3.1 (без малого рекламного окна). Проверьте, что в окне слева установлены позиции Add (добавление) и Comp (компонентов). Это означает, что Вы можете включать в свою схему компоненты.

Теперь с помощью мышки в правом окне найдите позицию ОРАМР (операционный усилитель) и сделайте ее активной, нажав левую клавишу мышки. Далее, отпустив клавишу, переведите маркер мышки на середину среднего окна и нажмите левую клавишу мышки. На экране появится изображение операционного усилителя. Оно сразу занимает правильное положение, но вы можете покрутить его если будете нажимать правую клавишу не отпуская левую. Как только положение усилителя вас устроит, отпустите левую клавишу.

В этот момент появится меню с перечислением типов операционных усилителей - см. " Рис.3.6. Переднее панно с меню типов операционных усилителей ".

С помощью мышки установите ее маркер на желаемый тип операционного усилителя и нажмите левую клавишу мышки. Меню типов операционных усилителей исчезнет, а рядом с изображением усилителя появится надпись о его типе. На этом ввод первого компонента (операционного усилителя) завершается.

Аналогичным образом введите другие компоненты схемы - " Рис.3.7. Полностью введенная схема усилительного каскада ".

При построении схемы вам придется неоднократно соединять компоненты отрезками проводов. Удобно делать это с помощью команды **Line** в левом окне. Установите маркер мышки на начало отрезка и нажмите левую клавишу мышки. Затем переместите маркер на конец отрезка и нажмите правую клавишу мышки - будет построен нужный отрезок.

И последнее полезное замечание по конструированию схемы. Если Вы случайно выполнили неправильную команду (например, стерли нужный компонент), не отчаивайтесь. Достаточно нажать Shift + Backspace, и результат последней команды будет отменен, стертый элемент появится в прежнем месте экрана.

При вводе текстов в окно Notes используется простейший текстовый редактор.

Возможно вы не сразу построите нужную схему. Однако если у вас есть хотя бы начальный опыт работы ПК, то через полчаса экспериментов вы не только справитесь с этой задачей, но и получите ценные практические навыки по вводу схем. Для этого стоит вернуться к началу этого раздела и более внимательно прочитать все приемы конструирования схем.

#### 1.4. Анализ переходных процессов

Выбор режима анализа осуществляется пунктом **Run** основного меню. Режим анализа переходных процессов **Transient** определяется рядом дополнительных подменю. При выборе этого режима на экране возникает панель параметров **Analysis limits**, которая включает в себя следующие параметры:

**Simulation time** (Tmax/Tmin/Timestep) - время моделирования, причем параметр Timestep определяет максимально допустимый шаг расчета переходных процессов.

**Display time** (Dmax/Dmin/Printstep) - временной интервал, выводимый на дисплей. Параметр **Printstep** определяет временной интервал вывода соседних точек на экран дисплея.

**Maximum change** - максимальное процентное изменение величин в течение одного шага расчета, типичное значение - 0.5 - 5%.

**Convergence criteria** - критерий сходимости процессов на каждом шаге расчета. Эквивалентен параметру RELTOL для SPICE. Имеет порядок  $1E-3 \dots 1E-6$ .

**Temperature** - температура, при которой производится анализ. Имеет формат (High/Low/Step).

**Minimum Timestep** - минимальный шаг расчета, имеет порядок  $1E-12 \dots 1E-15$ .

**Maximum voltage error** - максимальная ошибка при расчете напряжения (обычно  $1E-6$ ).

**Maximum current error** - максимальная ошибка при расчете токов (обычно  $1E-6$ ).

Таблица параметров **Analysis limits** вызывается также из меню Transient пунктом Limits - F9. Следует отметить что от трех последних установок существенно зависит скорость выполнения моделирования. Чем меньше заданы ошибки, тем длительнее идет моделирование (но его результат будет более достоверным). Чаще всего можно согласиться с заданными по умолчанию значениями ошибок.

Таблица параметров задается с некоторыми предложенными системой параметрами (или теми, которые были установлены ранее, если схема введена с дискового файла). Нужно скорректировать параметры таблицы. Для перемещения по ее строкам используйте клавишу ENTER или мышку. Новые значения параметров набираются с помощью клавишного пульта ПК в обычном порядке. " Рис.3.8. Таблица параметров режима Transient " показывает как выглядит таблица параметров для нашего примера (расчета каскада на операционном усилителе).

Большинство опций режима Transient имеют нужные установки по умолчанию. Однако полезно знать, что эти опции вызываются пунктом меню Optinos (F10) и имеют следующий смысл:

**Numeric output** - включение и выключение режима вывода численных значений величин, указанных в пункте Monitor (см. ниже), на дисплей, принтер или в файл. Устройство вывода указывается в пункте Options основного меню.

**Run options** - опции режима расчета: Normal - расчет с начальных значений, указанных в опции Initial values без сохранения результатов в файле; Save - расчет с сохранением в файле выходных значений для последующего

использования; **Retrive** - загрузка сохраненных ранее значений для последующей работы с ними.

**Initial values** - начальные значения для расчета переходных процессов. **Zero** - нулевые начальные условия, **Read** - начальные условия, предварительно сохраненные путем использования пункта **Initial values** меню **Transient** (см. ниже) в файле с расширением **.top**. **Leave** - покинутые начальные условия, соответствующие конечной точке предыдущего расчета.

**Run option** - опция расчета переходного процесса. Опция **Iterate** определяет наличие итеративного процесса при расчете каждой точки временного интервала. Число итераций на каждом шаге расчета обычно составляет 2 ... 4. Опция **Operating point** предполагает расчет схемы по постоянному току до расчета переходных процессов. Для большинства нелинейных электронных схем это приводит к лучшей сходимости результатов.

Более часто приходится пользоваться опциями дисплея, поскольку заранее неизвестны номера узлов схемы, задаваемых как выходные, желаемый тип графиков, цветовая палитра их и т.д.

**Display options** определяют вид результирующих графиков.

Пункт **Scope (F8)** подменю **Transient** позволяет использовать "лупу координат" после вывода графиков на экран. При этом пользователь имеет возможность выставить левую и правую границы временного интервала, а также верхнюю и нижнюю границы сигнала и просмотреть указанный фрагмент графиков в увеличенном масштабе, попутно прочитав интересные его координаты.

Пункт **Stepping (F11)** позволяет изменить любой параметр заданного компонента схемы с заданным шагом и провести при этом серию расчетов. Компонент обозначается в позиции **Step what**, начальное, конечное значения и шаг - в позициях **From**, **To**, **Step value**, соответственно. Графики переходных процессов при различных значениях избранного параметра совмещаются на одном окне. Для пользования этой опцией подлежащий изменению компонент должен иметь общее имя - например **RG** для выходного сопротивления генератора сигналов.

**Palette** - палитра цветов режима анализа позволяет сохранить три набора цветов графиков, заднего плана и сетки. Установка палитры столь наглядна, что не стоит на ней останавливаться подробно.

**Monitor (F12)** - пункт меню, определяющий совокупность величин, выводимых в той или иной форме для последующего анализа. Таблица мо-

нитора имеет 10 строк, соответствующих 10 выводимым величинам. На графики может выводиться до 8 величин: узловых потенциалов, напряжений между узлами, токов в элементах между узлами, мощностей и т.д. Столбцы таблицы указывают на форму вывода той или иной величины.

Опция Monitor одна из самых важных, поскольку именно она определяет, что именно вы выводите на график и сколько графиков задаете. Поэтому рассмотрим ее команды более детально.

USER - вывод в файл типа USERFILE для последующего анализа Фурье.

PLOT - вывод графика величины.

OUT - вывод значений величины в числовом виде. MS - вывод указанной величины в режиме статистического анализа.

T - тип выводимой величины:

V - узловой потенциал,

I - ток,

U - функция пользователя,

P - мощность,

E - энергия.

Waveform - описание узлов схемы, определяющих выводимую величину. Например, пометив узлы схемы OUT1 и OUT2 и записав функцию OUT1-OUT2, получим график дифференциального сигнала на выходе устройства. Вы можете использовать достаточно сложные функции, например вычислить логарифм разности напряжений между двумя узлами, или извлечь из нее квадратный корень (перечень функций соответствует применяемому в языке Бейсик).

Scale range (high/low) - масштаб соответствующего графика. Например, если задано 10/-10, то кривые графика будут расположены в интервале от -10 до 10 В (или А при токе).

Format - формат вывода численных значений величин в виде: (число знаков слева от точки).(число знаков справа). Наиболее часто используется формат 5.2.

Initial values - редактор начальных значений для расчета переходных процессов. Начальные значения могут быть отредактированы непосредственно после вывода их в виде таблицы, обнулены, считаны с файла или записаны в него. Для выбора значений, подлежащих редакции, используется "мышь" или клавиши Tab, стрелки. Выводимые в таблицу значения соответ-

ствуют последней точке расчета. Режимы обнуления, записи в файл и считывания с файла обеспечиваются командами Alt+1, Alt+2, Alt+3, соответственно.

### 1.5. Малосигнальный анализ в частотной области

Малосигнальный анализ задается пунктом АС меню **Run**. Как и в случае анализа переходных процессов вызывается дополнительное подменю, содержащее набор опций и параметров анализа в частотной области.

Limits (F9) - таблица параметров анализа. Она существенно расширена по сравнению с аналогичной таблицей Micro-CAP II и имеет следующие установочные опции:

Frequency range - частотный диапазон анализа, задаваемый в формате (высшая частота)/(низшая частота).

Gain range - диапазон коэффициента передачи в дБ: (наибольшее)/(наименьшее) значения.

Noise range - коэффициент шума в децибелах (дБ).

Zin range - диапазон изменения входного комплексного сопротивления (Ом). Yin range - обратная величина - комплексная проводимость в См.

Zout range - диапазон изменения комплексного выходного сопротивления в Ом.

Yout range - комплексная выходная проводимость.

Phase range - диапазон изменения фазового сдвига выходного сигнала относительно входного в град.

Delay range - диапазон групповой задержки, с.

Input - спецификация узла или узлов, являющихся входными. Формат задания: (номер узла) или (номер узла +)/(номер узла -). В качестве номера узла может выступать его метка, по аналогии с описанным в режиме Transient.

Output - аналогичная спецификация выходного узла.

Maximum change - максимальное процентное изменение величин в течение одного шага расчета, типичное значение - 0.5 - 5%.

Temperature - температура, при которой производится анализ. Формат задания аналогичен формату режима Transient.

На " Рис.3.11. Установочная таблица при малосигнальном анализе схемы рис.3.7 " приведена установочная таблица для малосигнального ана-

лиза схемы " Рис.3.7. Полностью введенная схема усилительного каскада" в частотной области.

Опции режима анализа АС задаются пунктом меню Options. Опции Numeric output, Run options и Display options режима АС совпадают с рассмотренным выше режимом Transient.

Monte Carlo Variable - переменные статистического анализа задают в качестве параметра для анализа влияния случайных величин главную переменную (Main variable), фазовый сдвиг (Phase angle) или групповую задержку (Group delay). Главная переменная анализа определяется отдельной опцией Main variable, ею может быть:

- Gain - коэффициент передачи,
- Zin - входное сопротивление,
- Yin - входная проводимость,
- Zout - выходное сопротивление,
- Yout - выходная проводимость,
- Noise - коэффициент шума.

Опции печати выходных данных Optional plots дают возможность вывести на экран кроме главной переменной ее фазочастотную характеристику (Phase angle) и групповую задержку (Group delay).

Наконец, опция шага анализа Frequency step позволяет строить характеристики в логарифмическом масштабе частоты (Auto) или с постоянным шагом Fixed.

Пункты меню Scope, Stepping и Palette аналогичны рассмотренным в режиме Transient.

На " Рис. 3.12. Результаты малосигнального анализа схемы рис.3.7 " показан результат малосигнального анализа схемы " Рис.3.7. Полностью введенная схема усилительного каскада" Построены частотные зависимости коэффициента передачи по напряжению, фазового сдвига и времени групповой задержки. На экране дисплея они выделены различными цветами.

Нетрудно заметить, что упомянутые выше (при анализе переходных процессов) разделительные RC-цепи порождают спад усиления на низких частотах. Хорошо виден и спад усиления на высоких частотах, связанный с инерционностью интегрального операционного усилителя.

## 1.6. Анализ по постоянному току

Режимы анализа по постоянному току в Micro-CAP III существенно дополнены по сравнению с предыдущей версией. Это относится прежде всего к наличию двух независимых источников сигнала Input 1 и Input 2, второй из которых, изменяясь с некоторым шагом, может являться параметром, определяющим вид передаточной характеристики по постоянному току. С его помощью удобно, например, снимать семейство статических характеристик транзисторов, задавая в качестве параметра ток базы. Режим анализа по постоянному току DC задается пунктом Run основного меню Micro-CAP III.

Таблица параметров анализа по постоянному току задается пунктом подменю Limits (F9). Она имеет вид:

Input 2 range - диапазон изменения параметров второго (дополнительного) источника, задаваемый в формате High/Low;

Input 2 step - шаг изменения величины дополнительного источника;

Input 2 - узлы подключения дополнительного источника в формате: (узел +)/(узел -)/(I или V).

Дополнительный источник подключается к узлам схемы + и -, может являться источником тока или напряжения (I или V). Дополнительный источник изменяет свою величину на один шаг после каждого анализа по постоянному току с изменяющимся источником Input 1. Если дополнительный источник не используется (выполняется обычный режим анализа по постоянному току), то вместо параметров, определяющих дополнительный источник, указывается NONE.

- Input 1 range - диапазон изменения параметров основного источника.
- Maximum input 1 step - максимальный шаг изменения параметров основного источника.
- Input 1 - узлы подключения основного источника.
- Формат описания всех параметров основного источника аналогичен формату дополнительного источника.
- Output range - диапазон изменения выходных параметров.
- Output - узлы, определяющие выход схемы. В качестве описания узлов могут служить их номера, метки, а также выражения вида (узел

1)/(узел 2)/(I или V). При указании I выходным параметром является ток в резисторе, соединяющем узлы 1 и 2.

- Maximum change - максимальное изменение выходного параметра за 1 шаг расчета, типичное значение 0.5 - 5 %.
- Temperature - температура, при которой осуществляется анализ.

Опции анализа по постоянному току Numeric options, Display options и Run options совпадают с соответствующим опциями режима Transient.

Дополнительные функции "лупа координат" Scope, ступенчатое изменение параметров элемента Stepping и палитра цветов Palette аналогичны выше описанным режимам.

При анализе по постоянному току надо сменить номера входного и выходного узлов, поскольку подавать сигнал постоянного тока на вход разделительной RC-цепи бессмысленно, равно как и снимать выходной сигнал после такой цепи. Поэтому в установочной таблице входным узлом взят неинвертирующий вход операционного усилителя, а выходным - его выход.

После пуска моделирования командой Run меню DC можно получить зависимость выходного напряжения усилителя от его входного напряжения. В данном случае результат тривиален - просто наклонная прямая. Поэтому рассмотрим более интересный случай.

При работе с полевыми и биполярными транзисторами важнейшее значение имеют семейства их вольт-амперных характеристик - ВАХ. Их наблюдают с помощью специальных характериографов. На " Рис.3.13. Схема, моделирующая работу характериографа " в правом нижнем углу показана очень простая схема, моделирующая работу характериографа для получения семейства выходных ВАХ мощного МДП-транзистора. К затвору его подключен источник постоянного напряжения  $U_{in}$ , а к стоку через малый по величине резистор (0.001 Ом) источник импульсов линейно нарастающего (от 0 до 100 В) напряжения.

В центре экрана на рис.3.13 видно окно с установочной таблицей режима DC. Как отмечалось, напряжение на входе (затворе) можно задать как бы от двух источников. В данном случае задано напряжение первого источника от 0 до 16 В с шагом 2 В, а напряжение второго источника 0 (для входного узла). В этом случае будет построено семейство ВАХ (т.е. зависимостей тока стока от напряжения на затворе) при напряжениях на затворе, равных 0, 2, 4, ..., 16 В. Именно в таком виде семейства ВАХ обычно приводятся

в справочниках по транзисторам. Смотри " Рис.3.14. Семейство выходных ВАХ мощного МДП-транзистора "

Хотя подобное семейство можно построить с помощью описанной далее программы PEP, во многих случаях удобно проводить такие построения не выходя из среды системы. Это позволяет легко модернизировать и корректировать статические модели транзисторов и других активных компонентов.

### **1.7. Анализ Фурье**

Гармонический анализ производится путем разложения в ряд Фурье сигнала, записанного в файле (имя схемы).usr. Для формирования указанного файла необходимо при расчете переходных процессов установить опцию User в меню Monitor для интересующего выходного сигнала (см. режим анализа Transient).

Режим анализа Fourier задается пунктом Run основного меню Micro-CAP III. При этом на экран выводится список файлов типа USER или предоставляется возможность набрать имя файла. Для проведения анализа указывается число гармоник (до 128). Вывод результатов в числовой форме осуществляется с помощью пункта Text Output дополнительного меню Hardcopy. Режим анализа Фурье позволяет также хранить три палитры цветов. При выводе числовых результатов в файл ему присваивается расширение .fno.

Методика проведения гармонического анализа в системе Micro-CAP III совершенно аналогична описанной ранее для системы Micro-CAP II.

### **1.8. Статистическое моделирование схем**

Издавна карточные фокусники пользовались законами случайных чисел. По имени человека, во многом разгадавшего эти трюки, назван метод анализа - Монте-Карло. Коротко анализ этим методом электронных схем означает многовариантный их расчет при условии, что параметры компонентов меняются по случайному закону (обычно с нормальным распределением вероятностей). Такой расчет очень напоминает исследование множества макетов одной и той же схемы, собранных на компонентах всегда имеющих разброс параметров.

Метод Монте-Карло в системе Micro-CAP реализован заданием определенного разброса у всех параметров компонентов. Параметры меняются от случая (Case) к случаю. Разумеется, реализация этого метода требует на много больше времени чем обычный одновариантный расчет схемы. Зато она дает представление о работе схемы в условиях, близких к реальным. Можно установить допустимые нормы разброса выходных параметров. Все это остро необходимо при создании высоконадежных и технологичных устройств, способных функционировать при любом возможном разбросе параметров входящих в них деталей (компонентов).

Для реализации метода Монте-Карло используется позиция основного меню Monte Carlo. Ее можно использовать при любом виде анализа. Режим анализа задается позицией подменю Options:

Number of runs - число запусков расчета при вероятностном характере распределения значений переменных. Статистический анализ стартует при числе запусков более одного. В режиме статистического анализа Micro-CAP III запоминает до 10 графиков результатов запуска.

Independent variable lower limits,

Independent variable upper limits - нижний и верхний пороги независимой переменной. В режиме переходных процессов указанная переменная - время, в малосигнальном анализе - частота, в анализе по постоянному току - входной сигнал. Разброс параметров компонентов учитывается лишь тогда, когда независимая переменная находится в заданном интервале.

Dependent variable lower limits,

Dependent variable upper limits - нижний и верхний пороги зависимой переменной. Разброс параметров компонентов учитывается лишь тогда, когда зависимая переменная, указанная в таблице монитора, находится в заданном интервале.

Numeric output - включение и выключение режима вывода информации о статистическом расчете в числовом виде.

Distribution to use - закон распределения случайной величины:

Normal - нормальное Гауссовское распределение,

Linear - линейное распределение,

Worst case - случайная выборка параметров из диапазона минимальное ... максимальное значение.

Statistic - выводит на дисплей обобщенные данные о результатах статистического анализа, включая красочную трехмерную гистограмму распределения выбранных параметров.

Режимы статистического анализа Monte Carlo в частотной области устанавливаются по аналогии с режимом временного анализа.

В качестве примера рассмотрим статистический анализ переходного процесса в LCR-цепи " Рис.3.15. Колебательный LRC-контур, возбуждаемый перепадом " компоненты которой имеют разброс номиналов в пределах 10%.

На " Рис.3.16. Результат моделирования цепи рис.3.15 методом Монте-Карло " представлено входное воздействие (перепад напряжения) и семейство графиков выходных временных зависимостей.

С первого взгляда наложение многих временных зависимостей друг на друга просто портит рисунок. Однако специалист из этого месива может выделить весьма ценную информацию - например определить максимальную и минимальную амплитуду колебаний на выходе схемы. Более того, можно получить статистические данные о процессе и даже красочные трехмерные гистограммы распределения параметров.

С этой целью необходимо (после многовариантного просчета схемы) перейти к команде Monte Carlo основного меню и использовать команду Statistics. Это вызовет появление на экране окна с результатами статистической обработки результатов моделирования - см. " Рис.3.17. Окно с результатами статистической обработки ".

В данном случае представлены результаты обработки и гистограмма распределения амплитуды импульсов на выходе анализируемой цепи (максимального напряжения). При необходимости можно получить и иные данные, например распределение отрицательного выброса (минимального напряжения).

## **1.9. Описание компонентов и работа с библиотеками**

Параметры стандартных компонентов Micro-CAP III содержатся в библиотеке, которая размещена в нескольких файлах. Для создания новых библиотек пользователь может присваивать им уникальные имена (не более 8 символов). Расширения файлов указывают на тип компонента:

- .als - список имен всех компонентов,
- utx - функции пользователя,
- sin - синусоидальные источники,
- rlc - пассивные компоненты,
- pul - программируемые источники,
- pol - полиномиально-зависимые источники,
- opa - операционные усилители,
- mos - МОП или МДП-транзисторы,
- jft - полевые транзисторы,
- dio - полупроводниковые диоды,
- bjt - биполярные транзисторы.

Редактирование параметров компонентов осуществляется с помощью пункта Editors основного меню Micro-CAP III. Выбор пункта 1:Library devices выводит на экран меню библиотек параметров компонентов.

Отметим, что войти в меню библиотек можно и с помощью пункта Info меню режимов работы. Установив курсор мышки в поле желаемого компонента, вы сразу же попадаете в библиотеку, содержащую выбранный компонент, и получаете возможность редактировать его параметры.

### **1.10. Задание и использование макроопределений (подсхем)**

Система Micro-CAP III позволяет задавать из фрагментов схем не только четырехполюсники, но и многополюсники. Для этого лишь нужно составить фрагмент схемы, обозначить надписями все свободные входы и выходы схемы и записать схему в виде файла с расширением .cir. При этом схема должна быть определена как макрорасширение системы (MACROSS).

В расширенном варианте системы Micro-CAP III представлен полный набор макрорасширений - логических микросхем AND, OR и INV (в том числе схемы AND и OR с инверторами). Это позволяет моделировать довольно сложные логические и цифровые схемы, хотя для этих целей больше удобств дает специальная программа Micro-LOGIC

### **1.11. Задание и применение функций пользователя**

Наиболее интересной возможностью Micro-CAP III является создание произвольных моделей новых компонентов с помощью функций пользова-

теля. Функция пользователя программируется с помощью пункта **Editors** основного меню (подменю 3:Library functions). При этом в качестве аргумента функции могут быть использованы метки узлов (т.е. узловые потенциалы различных точек схемы). В описываемой версии Micro-CAP III функции пользователя определены лишь для режима анализа переходных процессов Transient.

### 1.12. Преобразование библиотек и схемных файлов

Существенные различия в описании активных компонентов электронных схем и в графическом начертании рассчитываемой схемы вызывают естественную проблему преемственности поколений Micro-CAP. Для снятия указанной проблемы в комплексе программ Micro-CAP III существуют исполняемые файлы конвертирования библиотек и схем Micro-CAP II в формат Micro-CAP III. Эти файлы называются `convertl.exe` и `convertf.exe`, соответственно.

Для конвертирования библиотеки Micro-CAP II она переносится в тот же директорию, что и файл `convertl.exe` и выполняется командная строка `convertl.exe` (имя библиотеки Micro-CAP II)

Например, если компоненты Micro-CAP II содержались в библиотеке `library.std`, то после выполнения командной строки

```
convertl.exe library.std
```

будут созданы файлы библиотечных компонентов Micro-CAP III с именами `std.als`, `std.opa`, `std.bjt` и т.д.

Для конвертирования схемных файлов запускается командная строка **convertf.exe** (имя схемного файла Micro-CAP II).

При этом необходимо обеспечить присутствие в одной директории файлов `convertf.exe`, `mc3.dat`, исходного схемного файла и файлов библиотек компонентов.

### 1.13. Подготовка параметров компонентов с помощью программы PER

Параметры моделей библиотечных активных компонентов: биполярных и полевых транзисторов, операционных усилителей и диодов для интегрированного пакета Micro-CAP III могут быть определены по их спра-

вочным данным с помощью программы PER. Библиотека параметров PER содержит до 50 компонентов каждого типа. Библиотека PER физически представлена в двух файлах с именами \*\*\*\*.sur и \*\*\*\*.per, где \*\*\*\* - имя библиотеки, совпадающее с именем соответствующей библиотеки Micro-CAP III. Также как и в Micro-CAP II запуск PER осуществляется исполняемым модулем per.exe, который формирует на экране основное меню следующего вида:

Parameter Estimation Program  
for MICRO-CAP III

### **1.14. Редактирование изображений компонентов и дополнительные сервисные функции Micro-CAP III**

Любому пользователю хотелось бы, чтобы изображение схемы, которую он конструирует с помощью Micro-CAP, выглядело как можно более естественно и соответствовало требованиям постоянно меняющихся стандартов. Micro-CAP III позволяет реализовать даже эту возможность с помощью редактора изображений компонентов, который входит в состав интегрированной системы. Редактор запускается исполняемым файлом sh.exe, и позволяет формировать до 75 различных изображений. Действия по редактированию обеспечиваются системным меню, работа с которыми осуществляются с помощью мышки или путем одновременного нажатия: Alt + первая буква имени пункта меню.

## **Лекция № 9**

### **Интерактивная среда моделирования FEMLAB**

#### **Введение**

FEMLAB – Toolbox MATLAB, предназначенный для решения мультифизических проблем, формулируемых в терминах дифференциальных уравнений в частных производных (PDE). Программное и алгоритмическое обеспечение FEMLAB наполнено большим количеством обучающих программ и примеров применения достаточно мощных функций, а также уз-

нать, как установить модель в графическом интерфейсе пользователя (GUI) FEMLAB – среде, позволяющей запускать щелчком мыши инструментальные средства, поддерживающие самые современные конечноэлементные методы математического моделирования.

Для успешной работы следует изучить справочник, который содержит ряд содержательных обучающих программ, объясняющих следующие вопросы:

- Инициализация новых моделей и определение их параметров с помощью Навигатора Моделей, а также загрузка ранее записанных моделей из Библиотеки Моделей.
- Работа с графическим интерфейсом пользователя (GUI) FEMLAB.
- Эффективное выполнение моделирования в пределах графического интерфейса.
- Расширение возможностей графического интерфейса при работе с FEMLAB-функциями командной строки.
- Выполнение моделирования, основанного непосредственно на дифференциальных уравнениях в частных производных (PDE).

Изучив материал этого справочника, Вы сможете создавать готовые к запуску модели (в том числе и заказные) для Ваших спецпроектов. Работа над таким уровнем сложности требует, однако, чтобы Вы были знакомы с тонкостью работы многих компонентов FEMLAB, поэтому документация пакета включает в себя несколько вспомогательных справочников. Данное справочное пособие детально описывает базовые концепции пакета и показывает, как максимально полно использовать все его преимущества (структуры данных и функции командной строки MATLAB). Кроме того, библиотека моделей содержит много примеров из науки и инженерных разработок, которые поясняют, как использовать FEMLAB, чтобы решить реальные проблемы.

## **1. Назначение и области применения FEMLAB**

FEMLAB – мощная интерактивная среда для моделирования и решения научных и технических проблем, основанных на дифференциальных уравнениях в частных производных (PDE) (см. S. Littmarck, et. al., Math, models, motion and more, PT Design Magazine, Penton Media (Cleveland,

ОН), May 2000.). В этой среде можно легко расширять модели, находящиеся в составе пакета, переходя от одного физического поля к мультифизике и разыгрывая различные состояния модели, которые могут иметь отношение к различным отраслям науки и техники. Мощный интерфейс FEMLAB не требует, чтобы Вы имели глубокое знание математики или численного анализа. Вы можете формировать много полезных моделей, определяя участвующие в них физические величины, не рассматривая уравнений, которые описывают их. Однако FEMLAB позволяет Вам создавать модели, основанные на явно указанных уравнениях (в дальнейшем мы будем называть их “уравнение-основанные” модели). Помимо обеспечения этих многообразных подходов моделирования, программное обеспечение предлагает различные способы их использования: или через GUI-приложение femlab или из командной строки MATLAB.

Основная математическая структура, с которой работает FEMLAB – система дифференциальных уравнений в частных производных. В FEMLAB Вы можете представлять системы PDE двумя способами: в коэффицентной форме (для линейных или почти линейных задач) или в общей форме (для нелинейных задач). Возможно создание стационарных (не зависящих от времени) или нестационарных (зависящих от времени), а также линейных или нелинейных моделей, соответствующих либо скалярной, либо многокомпонентной краевой задаче. Пакет поддерживает также задачи на собственные значения и частоты.

При решении PDE, которые описывают модель, в FEMLAB применяются конечноэлементные методы (FEM). Программное обеспечение пакета поддерживает конечноэлементную технологию вместе с адаптивным построением сетки и контролем ошибок при работе с различными численными решателями. Более подробное описание математической и численной основы этой технологии дано в “Кратком обзоре PDE-моделей в FEMLAB”, см. также “Функции командной строки”.

Дифференциальные уравнения в частных производных – фундаментальная основа почти всех научных законов, значит, они могут и должны использоваться для моделирования научных явлений. FEMLAB имеет чрезвычайно широкую область применения, что позволяет моделировать большое количество материальных явлений во многих дисциплинах, таких как:

- акустика

- химические реакции
- диффузия
- электромагнитное поле
- гидроаэродинамика
- общие физические задачи
- геофизика
- теплопередача
- потоки пористых сред
- квантовая механика (и вообще, квантовая физика)
- полупроводниковые устройства
- сопротивление материалов (моделирование напряжённо-деформированного состояния вещества)
- распространение волн.

Чтобы показать, как FEMLAB решает известные или интересующие проблемы во многих из этих областей, эта документация включает “Библиотеку моделей”. Этому посвящён отдельный том, содержащий обширный набор законченных, готовых к запуску моделей. Исследование этих моделей – превосходный способ изучить, как работать с FEMLAB, и посмотреть, как всё это применяется к различным прикладным областям знаний. Далее, Вы можете приспособлять, расширять, или иначе изменять эти модели, чтобы удовлетворить ваши собственные требования. Таким образом, эти модели представляют отправную точку для дальнейшей работы, что позволит сэкономить значительное время во многих случаях.

Много реальных приложений включают совместное применение PDE из нескольких областей науки или техники. Исследователи назвали этот тип анализа мультифизическим моделированием. Например, электрическое сопротивление проводника часто зависит от температуры; таким образом, модель проводника с током включает термоэлектрические эффекты. Этот справочник вводит Вас в курс уникальных возможностей системы FEMLAB в работе с мультифизическими моделями (см. “Мультифизика: термоэлектрические эффекты”, “Мультифизическое и Уравнение-основанное моделирование”). Далее, “Библиотека моделей” посвящает отдельную главу изучению нескольких интересных мультифизических примеров.

Даже в базовой конфигурации FEMLAB имеются огромные возможности моделирования и анализа для многих дисциплин. Данный про-

граммный продукт показал себя особенно полезным для решения задач в рамках этих дисциплин. В связи с этим созданы необязательные модули прикладных программ по каждой из этих дисциплин. В данную версию включены модули Строительной механики (SME), Электромагнетизма (SEM), Химической технологии (CHEM), Гидроаэродинамики.

Несмотря на всё вышесказанное, надо признать, что невозможно создать специализированные модули для каждой возможной области применения, удовлетворив потенциальные требования каждого пользователя. Гибкость FEMLAB заключается в том, что этот Toolbox работает в системе MATLAB. В среде FEMLAB доступны все средства MATLAB и, наоборот, в среде MATLAB доступны все средства FEMLAB. Это даёт возможность комбинировать конечноэлементное моделирование с симуляцией динамических систем, а также с многочисленными другими аналитическими и численными методами анализа в науке и технике. Например, возможно создать конечноэлементную модель в FEMLAB и затем экспортировать её в Simulink или в Control System Toolbox, где модель становится неотъемлемой частью анализа динамической системы. Следует также обратить внимание на мультидисциплинарные прикладные программы FEMLAB.

## 2. Среда FEMLAB

Для понимания материала этого руководства и другой документации по FEMLAB нужно быть знакомым с основной терминологией. По крупному среда FEMLAB состоит из пяти главных частей:

**Графический интерфейс пользователя (GUI).** В этой среде Вы создаёте и исследуете модели двумя способами: 1) используя одну из нескольких predefined физических моделей, где Вы работаете со знакомыми научными законами и зависимостями, или 2) в одном из predefined PDE режимов, где Вы работаете прямо с основополагающими дифференциальными уравнениями в частных производных модели. Вы можете также выполнять мультифизическое моделирование, комбинируя любой из этих физических режимов и PDE режимов.

**Физические режимы,** предназначенные, чтобы выполняться в графическом интерфейсе FEMLAB, охватывают общие прикладные области типа теплопроводности, электромагнетизма, гидроаэродинамики, и строительной механики. Каждый режим, в свою очередь, реализует конкретную

модель PDE, для которой Вы определяете обычно используемые параметры, переменные и их значения. Из-за predetermined природы этих физических режимов, Вы можете устанавливать модель без потребности явного задания PDE.

**GUI FEMLAB** содержит **набор геометрических инструментальных средств (CAD)** для одномерного, двумерного и трёхмерного моделирования. В этом интерфейсе имеется средство автоматической генерации конечноэлементной сетки для любой геометрии.

В **FEMLAB** имеется простое, быстрое и удобное **средство визуального отображения** любой физической величины или параметра. Наиболее распространённые способы визуализации – двумерные поверхностные графики, сечения, изоповерхности, контурные графики.

В **GUI FEMLAB** имеются **средства экспорта различных данных в рабочую область MATLAB**. Это даёт возможность для обработки этих данных применять функции командной строки, обеспечивающие доступ ко всем средствам FEMLAB и MATLAB. В диалоговых окнах имеются кнопки и переключатели, обеспечивающие экспорт моделей непосредственно в другие инструментальные средства MATLAB, такие как Simulink и Control System Toolbox.

**Библиотека Моделей.** Этот набор predetermined моделей, к которым Вы можете обращаться через диалоговое окно Model Navigator, является важным по нескольким причинам. Это средство предоставляет быстрый способ изучить, как использовать возможности FEMLAB: загружая и запуская модели из библиотеки, Вы можете быстро увидеть результаты их работы. Вы можете взаимодействовать с моделью, изменяя ее геометрию, ключевые параметры, типы значений переменных, входящих в уравнения. Модели из Библиотеки можно использовать как базовые, сохраняя их модификации под разными именами.

**m-файлы моделей.** Возможно экспортировать (сохранять) любую модель, созданную в GUI-приложении femlab, в виде m-файла (т.е. в виде сценария или функции MATLAB). Представление модели в этой форме удобно для документирования работы или для её продолжения вне GUI-приложения femlab с помощью функций командной строки.

**Функции командной строки.** FEMLAB-функции командной строки позволяют создавать модели и работать с ними прямо из командного окна MATLAB. Здесь можно работать непосредственно с основной структурой

данных FEMLAB (структурный тип fem). Такое объединение функций FEMLAB и MATLAB даёт возможность заниматься существенно-нестандартным моделированием.

**Программный интерфейс приложений (API).** Эта библиотека функций и методов MATLAB позволяет Вам создавать компоненты GUI и таким образом настраивать графический интерфейс FEMLAB для конкретных приложений. Строя заказной интерфейс, оптимизированный для конкретной задачи, Вы можете создавать законченные приложения, которые ограждают случайных пользователей от ненужной сложности, но всё же позволяют им быстро получить результаты сложных исследований. Вы можете также использовать API для формирования параметризованных моделей в GUI-приложениях.

### **3 Состав документации по FEMLAB**

Чтобы обеспечить пользователям указанные выше возможности, FEMLAB поставляется с обширным набором документации. Документация включает в себя справочную систему по функциям командной строки и по интерактивной работе, в состав документации входят также печатные версии справочников и руководств. Полный набор печатной документации состоит из следующих документов:

- FEMLAB 2.3. Руководство по инсталляции и Замечания по версии
- FEMLAB 2.3. Руководство пользователя
- FEMLAB 2.3. Системы автоматического управления
- FEMLAB 2.3. Химические технологии
- FEMLAB 2.3. Руководство по Программному Интерфейсу Приложений

#### **Необязательные модули для конкретных дисциплин**

В дополнение к ядру пакета FEMLAB группа разработчиков написала необязательные модули расширения, которые содержат функции и обобщённые модели, предназначенные для конкретных дисциплин. Эти модули поддерживают знакомую для пользователя рабочую среду со стандартизированной и специализированной терминологией, а также со специализированными решателями и средствами визуализации. Как и ядро, эти модули поставляются с обширным набором предопределённых моделей,

которые не только поясняют, как воспользоваться преимуществом обширных возможностей FEMLAB в этих конкретных дисциплинах, но и содержат базовые модели для Ваших собственных разработок.

Модули конкретных дисциплин теперь применимы к следующим физическим полям:

**Сопrotивление материалов.** Модуль моделирования напряжённо-деформированного состояния вещества (SME) решает статические и динамические задачи теории упругости. Последние решаются в пространственно-частотной области. Имеется расширяемая библиотека материалов, в которой параметры свойств могут быть стандартизованными или определяемыми пользователем. Возможно решение линейных и нелинейных задач теории упругости. Кроме того, можно включать FEMLAB-модели в блоки структурных схем Simulink.

**Электромагнитные поля.** Модуль электромагнитных расчётов (CEM) поддерживает моделирование нескольких наиболее важных частных случаев этих полей (электростатических, магнитостатических, гармонических, нестационарных и др.) Подобно всем специализированным модулям, в нём содержится библиотека предопределённых моделей. Этот модуль особенно полезен для пользователей, которые имеют твёрдые познания в электромагнетизме, но не знакомы с современными вычислительными методами. Модуль является также превосходным инструментом для образовательных целей. Модели этого модуля хорошо встраиваются в мультифизическое моделирование. Кроме того, возможно преобразование моделей, которые Вы разрабатываете в CEM модуле в “уравнение-основанные” модели. Это позволяет разбираться в уравнениях электромагнитного поля. И как с любой моделью FEMLAB, Вы можете экспортировать модели в рабочую область MATLAB, сохранять в виде m-файлов и затем подключать их к любым приложениям MATLAB.

**Химические технологии (CHEM).** Модуль предназначен для специалистов по химическим технологиям и может быть полезен при выполнении исследований, инженерных проектов, а также в образовании. Модуль содержит дополнительную библиотеку моделей. Возможна загрузка ряда примеров моделей непосредственно через Навигатор Моделей, который можно приспособлять к Вашей собственной задаче. Модели, разработанные в среде данного модуля, могут легко интегрироваться как с базовыми функциями FEMLAB, так и с другими модулями, что позволяет мо-

дели химических технологий легко встраивать в мультифизические модели. Модели химических технологий могут легко экспортироваться в рабочую область MATLAB и сохраняться в виде m-файлов. Это позволяет встраивать модели в любые инструментальные средства MATLAB, включая Simulink и Control System Toolbox.

#### 4. Порядок решения задач

1. Выбираем размерность модели, определяем физический раздел в Model Navigator [Навигаторе моделей] (каждому разделу соответствует определенное дифференциальное уравнение) и определяем стационарный или нестационарный анализ температурного поля.
2. Определяем рабочую область и задаем геометрию
3. Задаём исходные данные, зависимости переменных от координат и времени
4. Указываем теплофизические свойства и начальные условия
5. Указываем граничные условия
6. Задаём параметры и строим сетку
7. Определяем параметры решающего устройства и запускаем расчет.
8. Настраиваем режим отображения
9. Получаем результаты

##### 4.1. Навигатор моделей и главное меню

- Выбирая размерность модели, следует помнить, что просто задание сетки в трехмерной модели может занимать десятки минут даже на мощном компьютере. Для большинства трехмерных задач имеет смысл сначала задать и рассчитать двумерную модель, а уже потом при необходимости повторить расчет для трехмерной модели. Тем более, что если вы не импортируете геометрию из внешней CAD системы, а задаете непосредственно в FEMLAB, то гораздо удобнее получить трехмерную модель, преобразованием соответствующей двумерной. Размерность модели выбирается в **Model Navigator [Навигаторе моделей]** на первой вкладке **New** в **Space Dimension [размерность пространства]**, кроме **1D**, **2D** и **3D** там есть **Axial Symmetry (1D)** и **(2D)** для осесимметричных моделей.

- Теперь выбираем физический раздел, в рамках данного пособия рассмотрены три раздела **Heat Transfer [Теплоперенос]** , **Fluid Dynamics [Гидродинамика]** и **Diffusion [Диффузия]** и модуль расширения **General Heat Transfer [Общий теплоперенос]** . Раздел **Fluid Dynamics** использует уравнение Навье-Стокса для несжимаемой жидкости. Раздел **Heat Transfer** включает в себя, кроме подраздела **Conduction [Кондукция]** , подраздел **Convection and Conduction [Конвекция и кондукция]** , который отличается тем, что кроме теплофизических свойств, можно задать поле скоростей теплоносителя. Раздел **Diffusion** тоже состоит из двух подразделов: **Diffusion [Диффузия]** и **Convection and Diffusion [Конвекция и диффузия]** . Один из модулей расширения **General Heat Transfer [Общий теплоперенос]** включает в себя объединенные с излучением, конвекцию и кондукцию, а так же разделы, **Thin Conductive Layer** – тонкий теплопроводный слой, **Bioheat equation** – режим расчета тепломассопереноса в живых тканях, **Non - Isothermal Flow** – неизотермический поток.

Чтобы создать мультифизические модели, например, поток жидкости в канале: надо нажать кнопку **Multiphysics** , затем нажимается кнопка **Add geometry [Добавить геометрию]** , в открывшемся окне выбирается размерность и названия осей. После выбора геометрии, нажимается кнопка **Add... [Добавить]** и сначала выбирается один физический раздел (**Heat Transfer > Convection and Conduction**), а потом в модель добавляется второй раздел (**Fluid Dynamics > Incompressible Navier - Stokes**). Между собой они взаимодействуют как раз через поле скоростей.

- Для каждого из подразделов можно выбрать **Steady-state analysis [Стационарный анализ]** или **Transient analysis [Переходный анализ]** , впрочем, вид анализа можно потом изменить.

- Так же на вкладке **New** в **Model Navigator** можно выбрать вид конечных элементов, по умолчанию стоит **Lagrange - Quadratic [Лагранжевы-квадратичные]** , предлагаются Лагранжевы элементы вплоть до пятой степени, в некоторых разделах доступны элементы Эйлера.

Кроме **New** , в **Model Navigator** , есть еще три вкладки. **Model Library [Библиотека моделей]** , в ней расположены примеры моделей для всех физических разделов, некоторые из этих моделей присутствуют в данном пособии. **User Models [Пользовательские модели]** , представляет собой отображение папки, в которой хранятся модели созданные. **Settings [Настройки]** позволяют установить язык (при установленном русификаторе).

ре, даже русский). И изменить фон рабочей области с белого на черный. В версии COMSOL 3.2 там же устанавливается система единиц, можно выбрать из десяти вариантов, включая СИ. Кроме того в этой версии появилась вкладка **Open** которая так же как и **User Models** позволяет работать с файлами.

После нажатия клавиши **ОК** в **Model Navigator** открывается окно основной программы с рабочей областью, инструментальными панелями и главным меню. Кнопки на инструментальных панелях повторяют пункты главного меню, поэтому мы рассмотрим пункты меню по порядку.

## 4.2 Главное меню

**File** – содержит команды создания, открытия и сохранения файлов, печати, а также импорта геометрии из внешних CAD систем и экспорта полученных данных в текстовый файл.

**Edit** – содержит команды отмены и повторения операций, работы с буфером обмена и команды выделения.

**Options** – содержит команды задания рабочей области **Axes/Grid settings**, констант **Constants**, выражений **Expression**, функций **Function**, связанных переменных **Coupling Variables** и различные настройки отображения геометрических элементов и масштаба.

**Draw** – содержит команды построения и преобразования геометрических объектов, а так же команды превращения двумерных объектов в трехмерные.

**Physics** – содержит команды задания физических свойств подобластей **Subdomain**, граничных условий **Boundary**, в том числе периодических ГУ (Граничных Условий) **Periodic Condition** и изменения системы дифференциальных уравнений **Equation system**.

**Mesh** – содержит команды управления конечноэлементной сеткой.

**Solve** – содержит очень важные команды управления решающим устройством, можно выбрать зависимость от времени, линейность или нелинейность, а так же указать множество других параметров решателя.

**Postprocessing** – содержит команды для отображения результатов вычислений, во всех возможных видах: от векторов и поверхностей уровня, до графиков и интегралов по границе.

**Multiphysics** – открывает **Model Navigator** , и позволяет переключаться между физическими режимами в мультифизических моделях.

**Help** – содержит обширную справочную систему.

### 4.3. Рабочая область и задание геометрии

Итак, мы открыли окно основной программы: если мы выбрали размерность пространства 1D , то мы видим рабочую область с координатной осью и кнопками для рисования геометрии. В одномерном режиме это кнопки **point** [точка], **line** [линия], **mirror** (отображает объект зеркально), **move** (перемещает объект) и **scale** (изменяет размер объекта). Также в верхней части экрана есть стандартные кнопки для выполнения операций с файлом и буфером обмена и кнопки повторяющие основные команды из разделов **Mesh** , **Solve** и **Postprocessing** главного меню. В режиме 2D добавляются кнопки создания кривых Безье, прямоугольников и овалов, кнопка **Array** [массив] создающая из одного объекта, матрицу объектов любого размера. Кнопка **Rotate** [вращение] позволяет повернуть созданный объект на любой угол. В трехмерном режиме с помощью кнопок можно создавать параллелепипеды, эллипсоиды, конусы, цилиндры и шары, а также управлять расположением координатных осей и освещением фигуры.

В первую очередь рекомендуется задать границы отображаемой рабочей области **Options > Axes/Grid settings** [Опции>Установки оси/решетки] как показано на рис. 9.1. Предположим, что мы работаем в двумерном режиме и хотим задать модель объекта с размерами метр на метр. В открывшемся окне выбираем вкладку **Axis** [Оси], галочка **Axis equal** означает, что оси будут равны, то есть один метр по оси X будет визуально такого же размера как по оси Y . Для протяженных объектов эту галочку можно снять, и тогда на экране оси могут быть не равны. Это удобно, когда в каком-нибудь из измерений объект непропорционально большой.

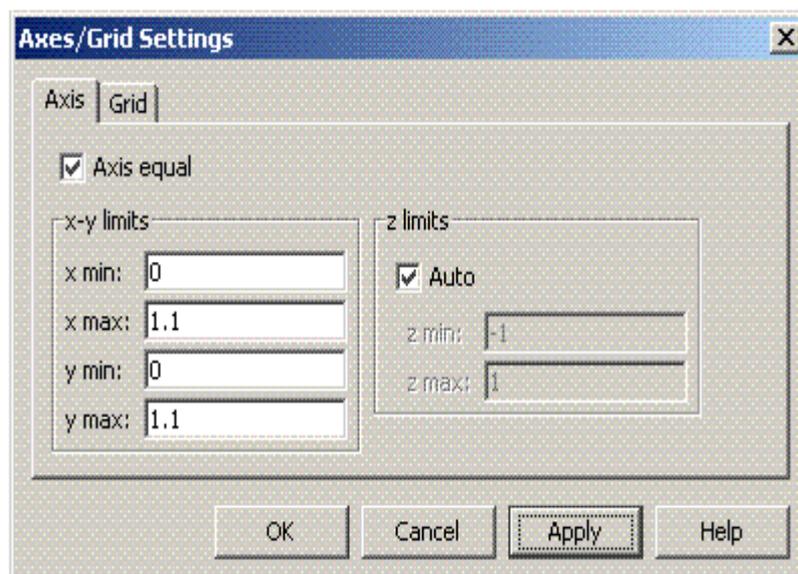


Рис. 9.1 Установка границ отображаемой области

В разделе x - y limits нужно задать пределы отображения осей, в нашем случае это 0 и 1.1 для минимума и максимума соответствующих осей. На вкладке **Grid [Решетка]** можно снять галочку **Auto** и самим установить интервал решетки. Вообще, при построении модели можно задавать просто координаты соответствующих фигур (например координаты центра окружности и ее радиус), но часто удобнее задать фигуру просто отметив эти координаты с помощью мыши, и тогда очень важно чтобы узлы решетки совпадали с ключевыми точками фигуры. Поэтому если толщина минимального элемента составляет скажем один миллиметр, то целесообразно задать именно такой интервал сетки. Галочка **Visible** позволяет выключить режим отображения. Внизу рабочей области можно выключить и привязку мыши к сетке **SNAP**, но тогда в случае ввода объекта с помощью мыши ключевые точки можно будет задать лишь примерно. В области x - y grid можно задать в полях x и y spacing интервал решетки по соответствующим осям. Поля Extra x и Extra y позволяют добавить любое количество дополнительных линий решетки.

**Внимание:** Не стоит путать понятия Grid – решетка и Mesh – сетка конечных элементов. Grid служит только для удобства задания геометрии модели и никак не влияет на расчеты. От сетки же конечных элементов Mesh напрямую зависит точность расчетов.

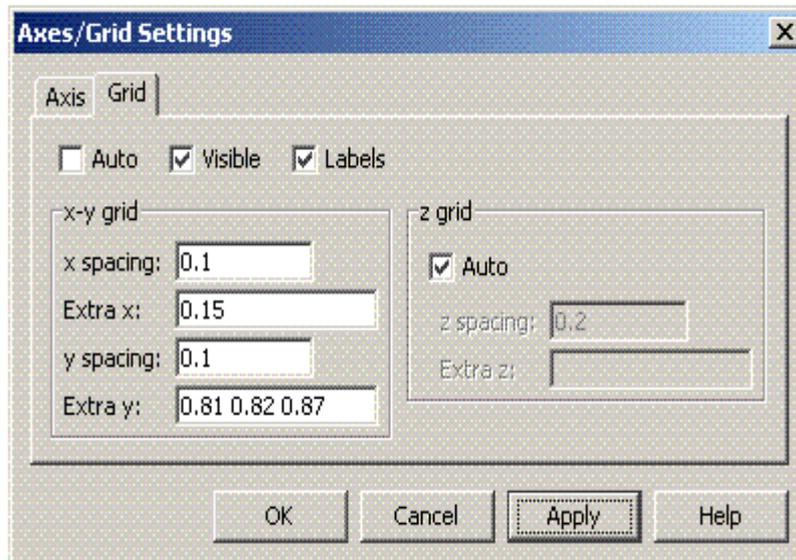


Рис. 9.2 Задание геометрии фигуры

После задания решетки можно приступить к определению геометрии. Если она у нас заранее не создана во внешней CAD программе или не задана в MATLAB (в этих случаях она просто импортируется через **File > Import**), то придется ее задавать внутренними средствами, как показано на рис. 9.2.

Предположим, надо нарисовать прямоугольник. Можно воспользоваться соответствующими кнопками **Rectangle/Square** [прямоугольник/квадрат] и **Rectangle/Square (Centered)** [прямоугольник/квадрат (центрированный)], первым щелчком отмечается расположение угла или центра, а потом прямоугольник растягивается до нужного размера и закрепляется вторым щелчком. При нажатой клавише Ctrl создается квадрат. Если нажать на клавишу Shift и щелкнуть на кнопке, то откроется окно со всеми параметрами фигуры. Это же окно можно открыть через главное меню: **Draw > Specify objects**.

Size задает размер объекта с помощью полей **Width** [ширина] и **Height** [высота]. **Rotation angle** задаёт угол поворота прямоугольника в градусах.

Область **Position** определяет место расположения объекта. Раскрывающийся список **Base** позволяет определить к чему относятся координаты x и y. **Corner** означает, что мы задаем расположение угла прямоугольника

(если мы рисуем эллипс, то надо задать координаты описанного прямоугольника). **Center** – мы задаем координаты центра объекта, рис. 9.3.

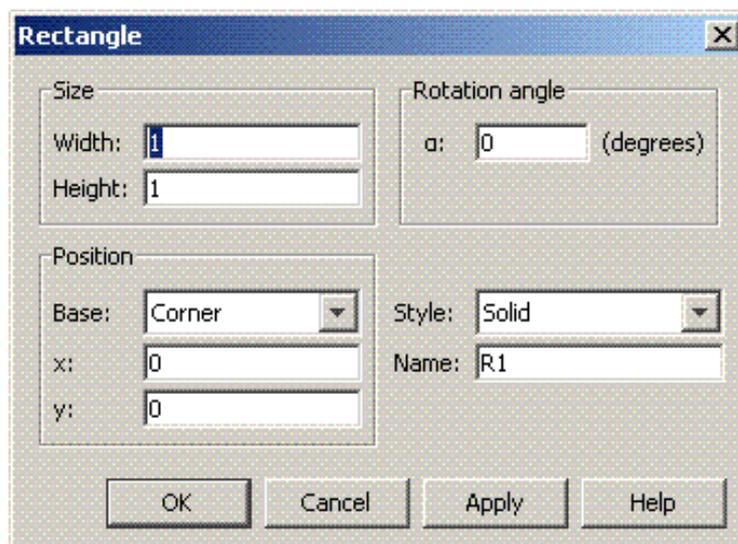


Рис. 9.3 Задание координаты центра объекта

Раскрывающийся список **Style**, предлагает варианты **Solid** – будет создана целая фигура – и **Curve** – будет создана кривая-контур фигуры. Кривая необходима для создания сложной фигуры: сначала задаются кривые, границы объекта, а потом командой **Coerce to solid** выделенные кривые делаются цельной фигурой. В режиме 3D вместо **Curve** – кривая есть понятие **Face** – оболочка. В поле **Name** можно вписать имя объекта. Несмотря на то, что в окнах **Specify objects** можно задать точные координаты и размеры объектов, часто легче задавать их с помощью мыши, а кривые Безье можно задать только с помощью мыши. Вот почему важно заранее определить период решетки.

При задании сложных фигур приходится задавать десятки элементарных объектов (овалы, прямоугольники, кривые Безье, линии, точки) потом их надо объединить или разделить. Обычно это делается по физическим признакам, с помощью кнопок **Union [объединить]**, **Difference [разность]**, и **Intersection [пересечение]** или команды **Draw > Create Composite Object ...** Эта команда открывает окно, где можно указать из каких элементов создается фигура. После создания фигуры с помощью кнопки **Fillet / Chamfer** или одноименного пункта меню **Draw** можно задать фаски или скругления углов. Так же можно размножить фигуру с по-

мощью кнопки **Array** , отразить с помощью **Mirror** , и изменить размер с помощью **Scale** . Кнопки **Rotate** и **Move** поворачивают и сдвигают выделенную фигуру соответственно. Все эти кнопки повторяются в виде пунктов меню **Draw > Modify**, как показано на рис. 9.4 .

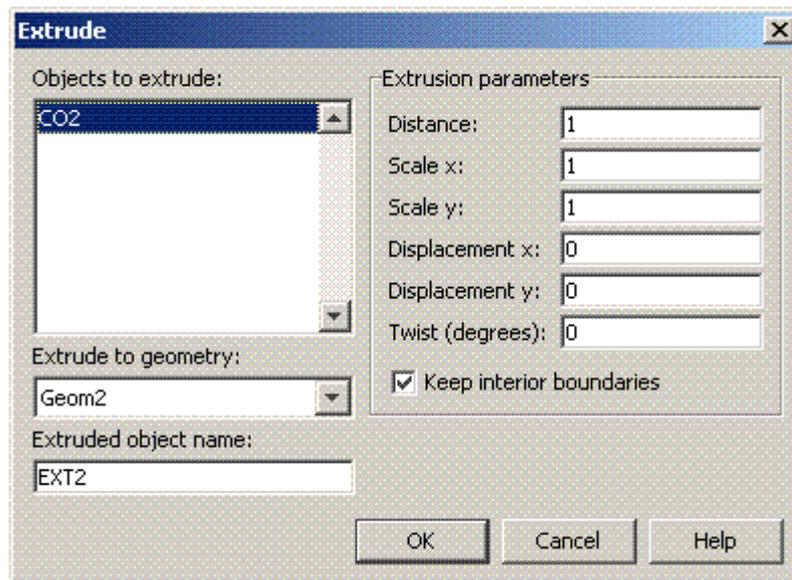


Рис. 9.4 Окно редактирования изображения фигуры

Для создания трехмерных моделей удобно элементарные фигуры задавать в режиме 3D , а более сложные задать сначала в режиме 2D , а потом перенести в трехмерную область.

Например, мы создали прямоугольник 1 x 0.5 метра. Если его выделить и нажать кнопку **Draw > Extrude** то откроется окно **Extrude [Экструзия]** где можно задать объект подвергающийся операции, название рабочей области (для одной модели можно задать несколько рабочих областей, обычно это несколько 2D геометрий и одна составная 3D ). Поле **Distance** определяет на сколько будет растянуто сечение. Если у нас был нарисован круг, то после Экструзии будет цилиндр, если сечение рельсы, то будет модель рельсы. **Scale x** и **y** задают во сколько раз будет изменяться сечение по длине объекта. Если задать в эти поля две двойки, то после экструзии (если сечение было круглым) появится усеченный конус. **Displacement** определяет сдвиг верхней плоскости фигуры относительно основания. **Twist** закручивает фигуру вокруг своей оси.

**Draw > Embed** скопирует наш двумерный прямоугольник в трехмерную рабочую область. По умолчанию в плоскость  $z = 0$ . Плоскость задается через **Draw > work plane settings...**

**Draw > Revolve**, создаст фигуру вращения. То есть из прямоугольника можно создать кольцо с прямоугольным сечением. В открывшемся окне можно задать градусы вращения по двум осям и координаты точек вокруг которых будет создаваться фигура вращения.

Для наглядности, с помощью команды **Scene Light** можно установить «освещение объекта», кнопка **Zoom extents** поместит фигуру на весь экран.

#### 4.4. Функции, константы и выражения

Теплофизические параметры могут зависеть от температуры и координат, внешние воздействия могут описываться различными функциями, зависящими от времени и множества других параметров. В расчетных уравнениях могут присутствовать различные константы, при чем значения этих констант могут различаться для разных частей системы (например: коэффициент теплоотдачи или коэффициент черноты поверхности).

В пакете Comsol ( Femlab ) существует мощный инструментарий для работы с константами и функциями. Большинство этих команд находится в меню Options. Рассмотрим некоторые из них.

- **Constants [Константы]** рекомендуется все применяемые в системе константы вынести в эту таблицу, а во всех формулах задавать только буквенное обозначение. Например, задать температуру окружающего воздуха **T<sub>air</sub> = 30**, а во всех граничных условиях вместо цифры задавать **T<sub>air</sub>**. Тогда в случае необходимости можно будет поменять одну цифру, а не искать по всему меню. Так же список всех часто употребляемых констант можно сохранить в отдельный файл и переносить из модели в модель. В последних версиях программы появилась возможность – к каждой константе написать **Description [Примечание]**, при работе нескольких человек с одной моделью не стоит игнорировать эту графу.

- **Expression [Выражения]** содержит **Scalar expression [Скалярные выражения]**, **Subdomain, Boundary, Edge** (только в трехмерном режиме) и **Point expression**. Можно задать зависимость теплофизического параметра от времени  $t$ ; координат  $x, y, z$ ; от безразмерной координаты  $s$  (из-

меняется от 0 до 1 по длине каждой границы) или от любых других вычисляемых параметров. Например  $Q_{ist} = 100 * \exp(t)$ . У различных элементов системы очень часто одни и те же параметры определяются по разным законам. Есть возможность задать одно имя у переменной, например **alfa**. И открыв **Boundary expression [Граничные выражения]** задать для разных границ, различные формулы вычисления **alfa**. Тогда для всех границ можно будет задать коэффициент теплообмена **alfa**, а программа сама подставит для каждой границы соответствующую формулу.

- **Couples Variables [Связанные переменные]** с помощью этих пунктов меню можно задавать очень сложные зависимости между частями системы. Например связать граничные условия с интегралом по объему.

- **Functions [Функция]** можно задать свою функцию, причем не только используя математические выражения. Если выбрать **Interpolation function**, то можно задать массив параметров и массив значений функций, и по ним построить интерполяционную функцию. Можно задать метод интерполяции, есть возможность импортировать данные из внешнего файла.

- **Coordinate systems [Системы координат]** можно создать произвольную систему координат, с любым расположением осей относительно друг друга.

- **Material / Coefficients Library [Библиотека материалов]** в ней есть возможность задать любые физические свойства веществ, и даже их зависимость от параметров (температуры и давления).

- **Visualization / Selection settings [Установки визуализации]** позволяют управлять отображением объектов, освещением и выделением.

- **Suppress [Скрытие]** В сложных объектах для наглядности иногда нужно сделать невидимым какой либо элемент системы. Например, если рассматривается процесс конвективного охлаждения, нагляднее, если воздух будет не виден, а будет видна только охлаждаемая поверхность с распределением температуры.

#### 4.5 Задание теплофизических свойств материалов и начальных условий

Когда геометрия задана и все константы определены, можно приступить к заданию теплофизических свойств. Для начала открываем меню **Physics > Subdomain Settings** или **F8**: откроется окно ввода коэффициен-

тов соответствующих дифференциальных уравнений. Для каждого из физических режимов это окно имеет свой вид, и все поля будут рассмотрены в соответствующих главах. Здесь же мы рассмотрим только общие для всех режимов поля. На рисунке показано это окно для режима **Heat Transfer by Conduction** в трехмерном режиме. Наверху в поле **Equation**, указано текущее уравнение. В поле **Subdomain selection**, необходимо выбрать область, для которой определяются физические свойства. Если областей много, то нужно выделить все созданные из одного материала. Чтобы выделить все области надо нажать **Ctrl +A**.

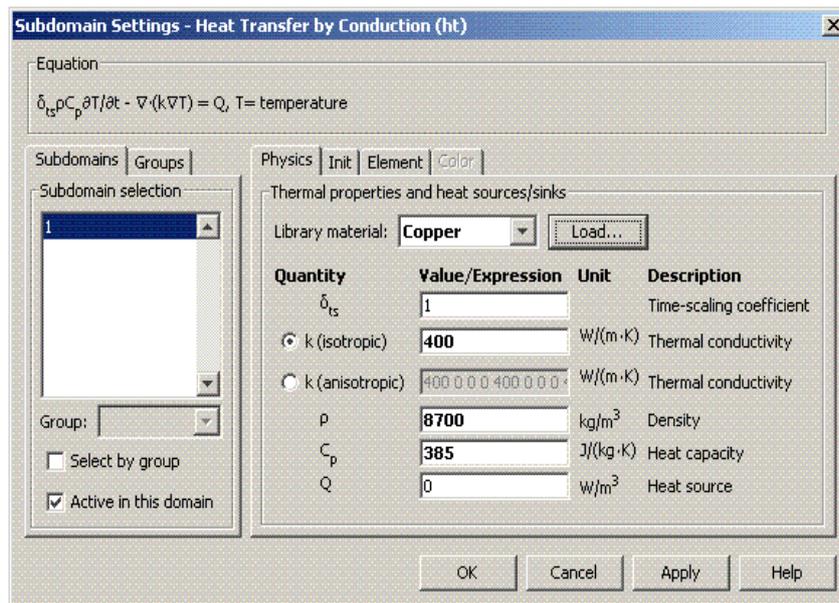


Рис. 9.5. Задание свойств материалов

Во вкладке **Physics** надо задать свойства материала, в данном случае теплофизические, для распространенных материалов можно воспользоваться встроенной библиотекой. В нестационарном режиме присутствует поле **Time - scaling coefficient [Временной коэффициент]**. Значение **1** соответствует заданию времени в секундах. Если мы рассматриваем процессы идущие значительно больше или меньше чем одна секунда, то удобнее задать другой коэффициент и указывать внешние воздействия в соответствующих единицах. Для того что бы работать с временем в минутах в это поле надо записать **1/60**, если в часах то **1/3600** и т.д.

Кроме этого можно задать дополнительные параметры объекта. В режиме теплопереноса это объемная мощность. В режиме уравнения Навье-Стокса и Конвекции это скорость потока по всем осям. В каждом из полей можно вводить математические выражения, синтаксис такой же как у MATLAB, но удобнее вводить в поля не формулы, а названия переменных, и определять их через **Options > Expression**.

В режимах связанных с конвекциями или потоком жидкости во вкладке появляется кнопка **Artificial Diffusion ... [Искусственная диффузия]** она позволяет стабилизировать решение. Метод конечных элементов подразумевает некоторую дискретизацию пространства. При моделировании явлений переноса в сложных объектах это может привести к неустойчивости решения. Неустойчивость можно обнаружить увидев большие колебания значений в областях с большими градиентами. Из-за больших колебаний решение может не сходиться. Нажимая **Solve > Restart [Решать>Заново]** новое решение будет отличаться от предыдущего. Методы стабилизации решения основаны на добавлении малых слагаемых в дифференциальные уравнения.

Вкладка **Init** предназначена для задания начальных условий, в данном случае это температура. Вкладка **Element** позволяет выбрать вид конечных элементов и их коэффициенты.

В последней версии программы COMSOL 3.2 появились вкладки **Groups** и **Color**. Вкладка **Group** позволяет распределить различные элементы по группам и менять свойства сразу у всей группы. По умолчанию, они раскрашиваются в разные цвета, изменить этот режим можно с помощью вкладки **Color**.

#### 4.6 Задание граничных условий и изменение дифференциальных уравнений

Граничные условия задаются через **Physics > Boundary Settings** или **F7**. В этом окне надо выбрать необходимые границы в поле **Boundary selection**. Для того, чтобы задать температурный перепад на границе двух тел надо сначала включить галочку **Interior boundaries** иначе внутренние границы будут недоступны. Во вкладке **Coefficients** надо выбрать вид граничных условий и указать в соответствующих полях коэффициенты ГУ.

На рисунке 9.6 показан пример задания граничного условия третьего рода. В поле Equation выводится соответствующее уравнение.

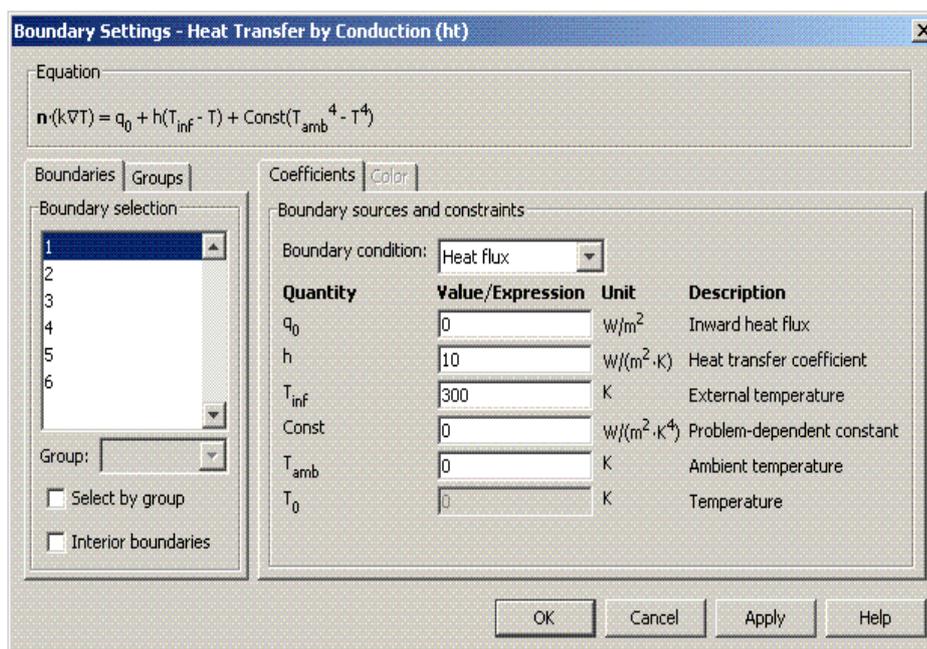


Рис. 9.6. Пример задания граничных условий

Как и в окне **Subdomain Settings** в окне **Boundary Settings** по последней версии программы - C OMSOL 3.2 появились вкладки **Groups** и **Color** присваивающие границам с различными ГУ разные цвета. В режиме расчета теплопереноса излучением в модуле **General Heat Transfer**, кроме группировки по свойствам теплообмена, для экономии вычислительных ресурсов, появляется группировка по взаимодействующим границам. То есть принято, что обмениваются излучением («видят друг друга») только границы принадлежащие к одной группе. Эта группа задается отдельным полем **Member of Group(s)** во вкладке **Boundary Condition**.

В некоторых моделях (в основном связанных с электричеством) кроме свойств на границе необходимо задавать свойства ребер и точек. В режиме **Fluid Flow>Incompressible Navier-Stokes** так же необходимо задавать давление в точке.

Часто при моделировании сложных систем, например радиоэлектронных аппаратов кассетного типа, выделяют элементарный объем и проводят расчет для этого элементарного объема. Для корректного расчета необходимо задать особый вид граничных условий – периодические гранич-

ные условия. В программе они задаются с помощью команды меню **Physics > Periodic Condition** . Кроме условий на границе **Boundary** требуется задать периодические свойства для точек **Point** и в некоторых режимах для ребер **Edge** . Настройки этих окон позволяют задать зависимость противоположных границ элементарного объема.

Для некоторых классов мультифизических задач, где надо связать два объекта с разным типом сетки (например, прямоугольную сетку в одной части системы с треугольной в другой) и граничными условиями Неймана можно применять условия идентичности **Physics > Identity Conditions** .

В COMSOL 3.2 есть много возможностей гибкой настройки системы под каждую конкретную задачу. Можно изменять систему Дифференциальных Уравнений в Частных Производных ( PDE ). Для этого служит группа команд **Physics > Equation system** . Эти команды позволяют в широких пределах изменять исходные PDE , способы задания начальных и граничных условий, а также параметры конечных элементов.

Рассмотрим на примере уравнения теплопереноса. В общем виде нестационарный процесс переноса выражается в коэффициентной форме данным уравнением:

$$c_a \frac{\partial^2 u}{\partial t^2} + d_a \frac{\partial u}{\partial t} + \nabla \cdot (-c \nabla u - \alpha u + \gamma) + \beta \cdot \nabla u + \alpha u = f$$

В прикладных режимах это уравнение представляется в упрощенном виде без лишних членов. Например, нестационарное уравнение теплопроводности (режим **Heat Transfer > Conduction** ) выглядит так:

$$\rho C \frac{\partial T}{\partial t} - \nabla \cdot (k \nabla T) = Q$$

То есть  $C_a$  ,  $\alpha$  ,  $\beta$  ,  $\gamma$  и  $a$  равны нулю. С учетом конвекции в режиме **Heat Transfer > Convection and Conduction** добавляется коэффициент  $\alpha = \rho \cdot C_p \cdot u$  (где  $u$  поле скоростей) и уравнение выглядит так:

$$\rho C_p \frac{\partial T}{\partial t} + \nabla \cdot (-k \nabla T + \rho C_p T u) = Q$$

Для быстропротекающих тепловых процессов, где закон Фурье уже не работает, добавляется коэффициент  $C_a$  со второй производной по времени от температуры, стандартный прикладной режим теплопереноса его учитывать не будет.

Чтобы самостоятельно добавить значения необходимых коэффициентов открываем окно **Physics > Equation system > Subdomain settings** или **Ctrl + F8** , здесь можно для каждой области ( **Subdomain** ) задать в прямом виде значение любого коэффициента PDE или же изменить формулу вычисления коэффициентов. У каждой области для которой мы примем изменения, появится значок замка. Он будет виден и из окна **Physics > Subdomain settings** . Приоритет всех значений у окна **Physics > Equation system > Subdomain settings** , поэтому надо проверять какие коэффициенты уже здесь указаны, а лишь потом задавать материальные свойства в **Physics > Subdomain settings** . Сбросить все изменения системы уравнений можно кнопкой **Reset Equation** .

Для изменения формул ГУ служит окно **Physics>Equation system>Boundary settings** или **Ctrl+F7** . Работа с ним аналогична предыдущему.

С помощью команды **Physics > ODE settings** в COMSOL 3.2 можно добавить любые Обыкновенные Дифференциальные Уравнения.

Эта глава методического руководства описывает общие принципы задания начальных условий и функций в COMSOL 3.2 (FEMLAB).

#### 4.7 Построение сетки

После задания всех свойств и граничных условий наступает очередь построения сетки. Для простейших моделей, на первом этапе оценочного расчета можно задать сетку по умолчанию **Mesh > Initialize Mesh** (или кнопка с изображением треугольника) и несколько раз нажать **Mesh > Refine mesh** и получив достаточно мелкую сетку приступить к решению. Для моделей чистой кондукции, не связанных с потоком массы ,можно этим и ограничиться: для более мелких элементов сетки система автоматически произведет сгущение, а если необходимо в какой-либо части системы еще более сгустить сетку, то можно нажать кнопку **Refine selection** и указать нужную область. В одномерном и двумерном стационарном режиме можно посоветовать просто делать наиболее мелкую сетку – скорость вычисления на современных компьютерах все равно будет приемлемой. Вообще, рекомендуется строить сетку такой густоты, чтобы между любыми двумя границами было не меньше десяти конечных элементов. В моделях учитывающих конвекцию надо всегда помнить, что размер конечного

элемента должен быть в несколько раз меньше толщины пограничного слоя иначе решение может не сойтись и будет в любом случае нестабильно.

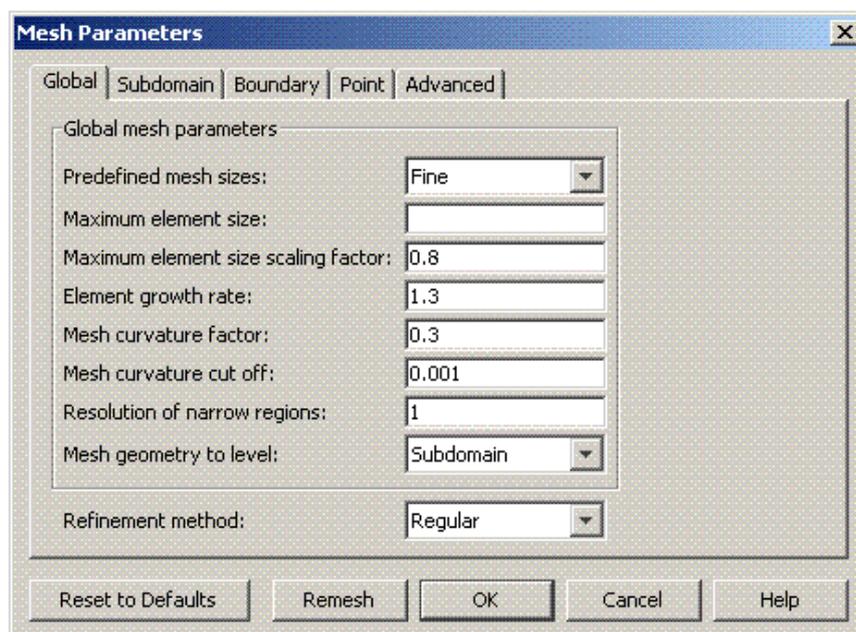


Рис. 9.7. Задание параметров сетки

По умолчанию, COMSOL строит в двумерном режиме треугольную, а в трехмерном тетраэдрическую сетку. Для задания параметров сетки надо выбрать **Mesh > Mesh parameters** или нажать **F 9** . Откроется окно настроек, на вкладке **Global** можно выбрать один из предустановленных режимов. В списке **Predefined mash sizes** девять режимов от **Extremely fine** [Чрезвычайно точный] до **Extremely coarse** [ Чрезвычайно грубый] , остальные расположены между этими крайними режимами. В полях можно задать собственные значения параметров сетки. **Maximum element size** задает максимальный размер элемента. По умолчанию равен **1/15** максимальной стороны, задавать его необязательно. **Maximum element size scaling factor** если ничего не задавать в предыдущее поле, то значение этого поля будет определять размер элемента (если задать **0.5** , то размер элемента будет равняться **1/30** максимальной стороны, если **0.1** то **1/150**). **Element growth rate** [ Темп роста элемента] отвечает за степень сгущения, принимает значения от единицы до бесконечности, чем ближе значение к единице тем более равномерная сетка. **Mesh curvature factor** и **Mesh**

**curvature cut off** чем меньше эти значения, тем более точно задана криволинейность границы: при больших значениях этих параметров вместо кривой будет считаться ломанная линия. **Resolution of narrow regions** задает минимальное количество элементов по самой короткой границе, для точных вычислений рекомендуется устанавливать значения этого параметра не меньше десяти.

Поле **Mesh geometry to level** позволяет выбрать отдельно создание сетки для областей, поверхностей и ребер. Для большинства задач тепло-массопереноса это значение должно быть областью **Subdomain**. **Refinement method** принимает два значения **Regular** и **Longest** и отвечает за режим работы команды **Refine mesh**. Если установлено значение **Regular** при нажатии этой команды каждый элемент делится на четыре части в 2 D режиме, и на восемь в 3 D. Значение **Longest**, делит каждый элемент на две части по самой длинной стороне. Вкладки **Subdomain**, **Boundary**, **Edge** и **Point** позволяют задать размер элемента для соответствующих элементов модели. Вкладка **Advanced** позволяет задать анизотропию сетки.

Кнопка **Remesh** перестраивает сетку с новыми параметрами.

В двумерном режиме для объектов близких к прямоугольным, можно задать четырехугольную сетку с помощью пункта меню Mesh>Map mesh или **Ctrl+F9**.

В разделе Рабочая область и задание геометрии мы рассмотрели способы преобразования двумерных моделей в трехмерные с помощью команд **Draw > Extrude** и **Draw > Revolve**. Тогда после задания трехмерной геометрии придется заново строить сетку из тетраэдров, что может занимать значительное время. Иногда целесообразно сначала построить сетку в 2 D режиме (треугольную или четырехугольную), а потом с помощью команд **Mesh > Extrude Mesh ...** растянуть фигуру с сеткой или **Mesh > Revolve Mesh ...** раскрутить фигуру с сеткой. Тогда элементы будут не тетраэдрические, а в виде параллелепипедов или призм. Время построения такой сетки меньше, чем строить тетраэдрическую сетку заново, но на скорость расчета задачи вид сетки кардинально не влияет.

#### 4.8. Решающее устройство

Выбор решающего устройства и его параметров очень важен, так как в основном от него зависит достоверность вычислений. Неправильная настройка может привести к грубым ошибкам решения, которые очень труд-

но выявить. Так же очень важно правильно оптимизировать решение, так как, к примеру, даже не очень сложная трехмерная модель элемента касетной конструкции рассчитывается десятки минут на компьютере с процессором Athlon 64 3000+ и 1 Gb оперативной памяти, а некоторые нелинейные нестационарные модели могут рассчитываться многие часы даже на очень мощном компьютере.

Кнопка **Solve** или пункт меню **Solve > Solve problem** запускает решающее устройство с текущими настройками. Кнопка **Restart** или пункт меню **Solve > Restart** перезапускает решающее устройство используя текущие значения (поле температур или скоростей) как начальные. Если мы рассматриваем стационарную задачу, то нажатие этой кнопки не должно менять решение. Колебания значений в этом случае говорят о неустойчивости решения. Эту команду целесообразно применять для сложных расчетов, когда можно получить приближенный вариант решения на грубой сетке и для линейного или стационарного решателя, а потом сделав более мелкую сетку и при необходимости изменив решатель на нелинейный или переходный пересчитать задачу. Зачастую это позволяет получить решение быстрее, чем сразу производя сложный расчет.

Для изменения параметров нажмем **F 11 Solve > Solver parameters ...** или соответствующую кнопку. Откроется окно, показанное на рис. 9.8:

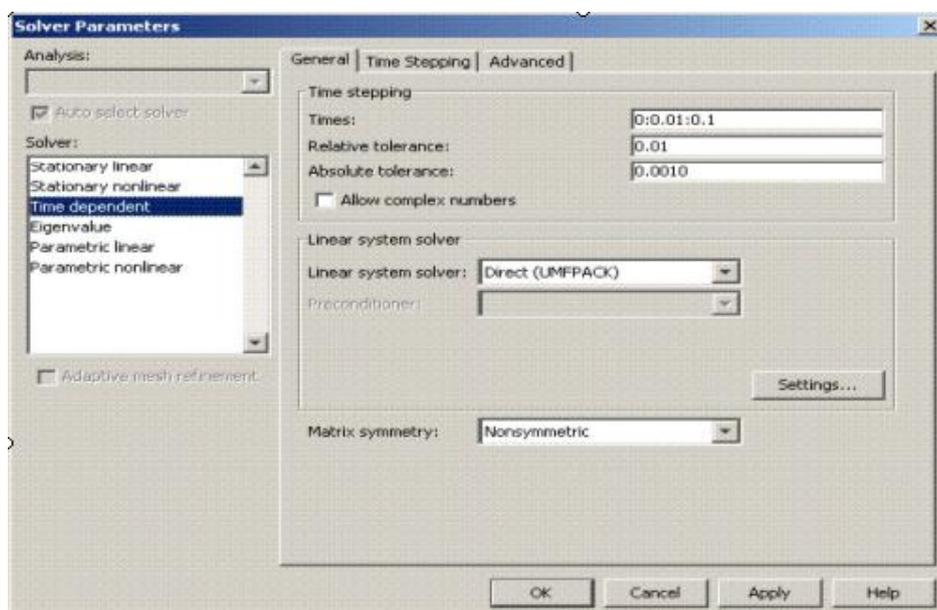


Рис. 9.8. Окно решателя Solver parameters

Если стоит галочка **Auto select solver** , значит программа в зависимости от прикладного режима подобрала наиболее подходящее устройство, для несложных вычислений чаще всего менять его не надо.

Выбирая решающее устройство, надо в первую очередь определиться стационарный или переходный процесс мы изучаем. Если процесс нестационарный, то в подавляющем большинстве случаев подходит решатель **Time Dependent** . Для очень редких задач, связанных с нахождением собственных чисел дифференциальных уравнений, например волнового уравнения теплопроводности, надо выбирать решатель **Eigenvalue** .

Если проблема стационарна, то надо определить линейна она или нелинейна. Если есть сомнения в линейности системы, то рекомендуется сразу устанавливать нелинейный решатель: если для линейной модели установить нелинейный решатель, то ответ будет корректен, но на вычисления затратится больше времени; а если для нелинейной задачи установить линейный решатель, то наверняка будут грубые ошибки.

**=> Внимание.** Если в заданных коэффициентах присутствуют переменные (например, теплоемкость или теплопроводность) для которых мы задали зависимость от температуры или от других переменных, то задача нелинейная. Все задачи в которых учитывается теплообмен излучением, то есть где  $C$  или  $C_{trans}$  (константа Стефана-Больцмана) не равна нулю – нелинейные. Модели, которые задают в прикладном режиме **Incompressible Navier-Stokes** практически всегда нелинейные.

Для линейных и нелинейных стационарных задач можно выбрать параметрический решатель, в котором надо указать параметры, для которых задаётся несколько значений. Например, задать ряд разных теплопроводностей или мощностей для какого-либо объекта, а потом сравнить распределение температур при всех значениях параметра.

После выбора решателя в поле **Solver** , задаем основные свойства.

Для **Time Dependent** на вкладке **General** это **Time stepping [шаги по времени]** в поле **Times** в формате **0:10: 60** задаются временные слои. Здесь, от **0** до **60** секунд с промежутком **10** секунд. Единица измерения времени секунда, она устанавливается в **Physics > Subdomain Settings** в поле **Time scaling coefficient** . Если вместо **1** задать **1/60** , то единица измерения времени будет равна 1 минуте. Поля **tolerance** определяют ошибку на каждом шаге итерации. Галочка **Allow complex number** позволяет при-

менять в расчетах комплексные числа это нужно в случаях, если мы задали коэффициенты PDE в комплексном виде. Вкладка **Time Stepping** определяет шаги по времени . Пункт **Times to store in output** определяет какие временные шаги будут выводиться для постобработки вычислений. По умолчанию стоит **Specified Times** , то есть времена определённые на вкладке **General** . Для того чтобы получить значения всех шагов решающего устройства надо выбрать **Time steps from solver** . Вообще, решающее устройство выбирает шаги произвольно, в зависимости от динамики системы, то есть игнорирует заданные значения **Times** на вкладке **General** . Чтобы решающее устройство учитывало этот список (например, если внешние воздействия импульсные и решатель может «проскочить мимо них») надо установить **Time steps taken by solver** в значение **Strict** или **Intermediate** вместо стоящего по умолчанию **Free** . Если надо принудительно задать равномерный шаг по времени, то это делается в поле **Manual Tuning of step size** .

Для **Eigenvalue** необходимо задать количество собственных чисел уравнения в поле **Desired number of eigenvalues** и около каких значений искать собственные числа в поле **Search for eigenvalues around** .

Для параметрических решателей надо установить имя параметра, который будет изменяться в поле **Name of parameter** и значения которые он будет принимать в поле **List of parameter values**. Значения можно перечислить через пробел, задать в виде **0:10: 100** или задать как функцию **linspace(1,100,25)** , то есть двадцать пять значений равномерно расположенных от **1** до **100**.

Для нелинейных решателей появляется вкладка **Nonlinear** где можно указать количество итераций, (по умолчанию стоит **25** ). Для сильно нелинейных проблем рекомендуется поставить галочку **Highly nonlinear problem** и увеличить количество итераций.

Для всех режимов кроме **Time Dependent** можно установить галочку **Adaptive Mesh Refinement** , тогда в процессе решения сетка будет уточнена по сложному алгоритму. Если физика и геометрия достаточно сложна, и не очень ясно как задавать параметры сетки, рекомендуется установить эту галочку. Но это увеличит время вычисления.

Кроме того, можно установить **Matrix symmetry** в значение **Symmetric** если наша матрица симметрична. Чаще всего это бывает, если мы рассматриваем процессы чистой кондукции или диффузии. При выборе

соответствующего линейного решателя это ускорит расчет. Матрицы заведомо несимметричны в мультифизических моделях, моделях связанных с конвекцией или уравнением Навье-Стокса.

Большая часть времени расчета занята решением систем линейных уравнений, отвечает за их решение **Linear system solver**. По умолчанию стоит **Direct (UMFPACK)** этот решатель отнимает много ресурсов компьютера и для моделей требующих длительного расчета можно подобрать более подходящий. Если предыдущий решатель не работает, можно попробовать **Direct (SPOOLES)** он требует меньше памяти, но работает нестабильно. В крайнем случае выбирается итеративный решатель **GMRES**. Для положительно определенных систем с симметричными матрицами

[Способ определения положительно определенных систем с симметричными матрицами [symmetric, positive-definite systems] и методы настройки итеративных решателей подробно описаны в справке программы.] в выбирается **Direct Cholesky (TAUCS)** или итеративный **Conjugate Gradients**. Итеративные решатели потребляют меньше памяти, но надо следить за тем, что они сходятся и в случае необходимости увеличивать количество итераций.

Уравнения чистой кондукции и чистой диффузии в стандартных режимах являются в стационарном состоянии эллиптическими, а в нестационарном параболическими. Для них очень эффективно использовать решатель **Geometric multigrid**.

После задания свойств нажимаем кнопку Solve или команду **Solve > Solve Problem**. Часто после получения решения, модель и её параметры (физические свойства и граничные условия) немного изменяются. И если эти изменения не очень велики, то можно использовать команду **Solve > Update model**, тогда задача пересчитываться не будет, а новые значения будут получены методом интерполяции. Так же можно нажать кнопку **Restart**, тогда задача пересчитается, но за начальные Init значения будут заданы те, которые были получены на прошлом этапе. Это может немного ускорить вычисления. Так же с помощью этой команды можно выявить нестабильность решения: если нажимая эту кнопку без изменения параметров модели мы получаем разные решения (осцилляции численного решения), то это говорит о нестабильности. В этом случае нужно уменьшить сетку или, если эта задача связана с конвекцией, применить Диффузионную стабилизацию в настройках физических свойств модели.

## Оглавление

1. САПР информационно – измерительной техники.....	3
2. Обзор современных методов проектирования .....	10
3. Компоненты САПР .....	25
4. Компьютеризация конструкторского проектирования .....	42
5. Компьютерное проектирование приборных систем с использованием интегрированного пакета Simulink .....	56
6. Технология проектирования динамических процессов в Simulink .....	62
7. Общие сведения о системе проектирования печатных плат P-CAD .....	70
8. Система схемотехнического моделирования Micro-CAP .....	85
9. Система схемотехнического моделирования Micro-CAP .....	114