# АВТОМАТИЗИРОВАННОЕ ПРОЕКТИРОВАНИЕ ЭЛЕКТРОННЫХ УСТРОЙСТВ

## **1 Цели, задачи и проблемы автоматизации проектирования сложных систем**

Автоматизированным называют проектирование, осуществляемое человеком при взаимодействии с ЭВМ. Степень автоматизации может быть различной, и оценивается долей проектных работ, выполняемых на ЭВМ без участия человека. При=0 проектирование называется неавтоматизированным, при =1 – автоматическим.

Система автоматизированного проектирования – организационно-техническая система, состоящая из комплекса средств автоматизации проектирования, взаимодействующего с подразделениями проектной организации и выполняющая автоматизированное проектирование.

Разработка средств автоматизации проектирования сложных электронных систем преследует следующие цели:

сокращение сроков и снижение стоимости разработки и внедрения изделий;

уменьшение количества ошибок при проектировании;

обеспечение возможности изменения проектных решений и сокращения сроков проверки и тестирования изделий.

Задачи, решаемые на различных этапах проектирования, можно укрупненно разделить на три группы: синтез и анализ. Задача анализа заключается в изучении поведения и свойств системы при заданных характеристиках внешней среды, ее компонентов и структуре системы (или ее модели). Согласно общей теории систем, синтез - это процесс порождения функций и структур, необходимых и достаточных для получения определенных результатов. Выявляя функции, реализуемые системой, определяют некоторую систему, о которой известно только то, что она будет делать.

В связи с этим, этап синтеза функций называется абстрактным синтезом. Существуют еще этапы структурного и параметрического синтеза. При структурном синтезе определяется структура объекта - множество составляющих его элементов и способы их связи между собой(в составе объекта и с внешней средой). Параметрический синтез заключается в определении числовых значений параметров элементов при заданных структуре и условиях работоспособности (т.е.необходимо найти точку или область в пространстве внутренних параметров, в которых выполняются те или иные условия).

Разработка САПР представляет собой крупную научно-техническую проблему. Несмотря на большие трудозатраты (50-200 квалифицированных специалистов), создание интегрированных САРП в различных областях техники - необходимость, вызванная ростом сложности объектов проектирования. С учетом изложенного можно сформулировать основные требования, которым должны удовлетворять САПР:

1. Иметь универсальную структуру, реализующую принципы декомпозиции и иерархичности (блочно-иерархический подход). Причем системы проектирования различных уровней иерархии должны быть информационно согласованы. Информационная согласованность означает, что для последовательного идущих проектных процедур, выходные данные одной из них могут быть входными для другой и при этом не требуется никаких преобразований.

2. Иметь высокую степень интеграции. Степень интеграции должна быть такова, чтобы обеспечить реализацию всего пути проектирования: от выдвижения идеи вплоть до реализации проекта. Важную роль для обеспечения интеграции инструментальных средств проектирования играют так называемые инфраструктуры (frameworks), САПР, обеспечивающие как интегрирование различных средств проектирования и данных, так и выполнение функций управления при помощи единого интерфейса пользователя.

3. Осуществлять проектирование в реальном масштабе времени. Уменьшение времени, необходимого для взаимодействия САПР с пользователем обеспечивается наличием оперативных технических средств взаимодействия разработчика с системой, эффективность процедур проектирования и т.п.

4. Структура САПР должна быть открытой, т.е. обладать свойством удобства расширения подсистем при ее совершенствовании.

5. Иметь средства контроля входной и выходной информации.

6. Иметь средства автоматического внесения изменений в проект.

2. Структура комплекса аппаратно-программных средств САПР

Все аппаратно-программные средства, составляющие базовое обеспечение САПР, могут быть классифицированы по выполняемым функция:

математическое обеспечение (МО);

лингвистическое обеспечение (ЛО);

программное обеспечение (ПО);

техническое обеспечение (ТО);

информационное обеспечение (ИО);

организационное обеспечение (ОО);

В МО входят: теория, методы, математические модели, алгоритмы, используемые при автоматизированном проектировании.

ЛО представлено совокупностью языков, применяемых при автоматизированном проектировании. Основная часть ЛО - языки общения человека с ЭВМ.

ПО - это совокупность машинных программ и соответствующая документация. Оно делится на общесистемное и прикладное. Компонентами общесистемного ПО являются, например, операционные системы, компиляторы и т.п. Эти программные средства предназначены для организации функционирования технических средств, т.е. для планирования и управления вычислительным процессом.

Прикладное ПО создается для нужд САПР. Оно обычно представлено в форме пакетов прикладных программ (ППП), каждый из которых обслуживает определенный этап процесса проектирования.

Компоненты ТО представляют собой совокупность взаимосвязанных и взаимодействующих технических средств (например, ЭВМ, средства передачи, ввода, отображения и документирования данных), предназначенных для автоматизированного проектирования.

ИО объединяет данные, необходимые для автоматизированного проектирования. Они могут быть представлены в виде тех или иных документов на различных носителях, содержащих сведения справочного характера о параметрах объекта проектирования, промежуточных результатах и т. д.

Основная часть ИО САПР - это банк данных (БНД), представляющий собой совокупность средств для централизованного накопления и коллективного использования данных в САПР. БНД состоит из базы данных (БД) и системы управления базой данных (СУБД). БД - сами данные, находящиеся в ЗУ ЭВМ и структурированные в соответствии с принятыми в данном БНД правилами. СУБД - совокупность программных средств, обеспечивающих функционирование БНД. С помощью СУБД осуществляется запись данных в БНД, их выборка по запросам пользователя и прикладных программ, и т.д.

Процесс автоматизированного проектирования представляет собой последовательное взаимодействия большого числа программных модулей. Взаимодействие модулей проявляется в основном в связях по управлению (упорядоченные переходы от исполнения одного программного модуля к исполнению другого), и по информации (использование одних и тех же данных в различных модулях) (см. рис. 1 и 2).

При проектировании сложных систем значительной является именно проблема информационного согласования различных программных модулей. Существует три основных способа реализации связей по информации:

через передачу параметров из вызывающей программы в вызываемую программу;

через общие области (обменные зоны) взаимодействующих модулей;

через банк данных.

Реализация информационных связей через передачу параметров означает, что передаются либо параметры, либо их адреса. Применяется при сравнительно небольшом объеме передаваемых данных и их простой структуре.

Реализация информационных связей через обменную зону, каждый модуль должен направлять данные в обменную зону, представляя их в форме, допустимой с позиции требования любого из остальных модулей. Так как требования к структуре данных каждого модуля - потребителя данных могут оказаться различными, то способ связи через обменные зоны сравнительно легко реализуется только при малом и стабильном числе информационных связей. Применяются для программных модулей внутри определенного ППП.

Если же одни и те же модули могут входить в различные проектные процедуры, взаимодействовать со многими модулями, то целесообразно унифицировать средства информационного обмена. Такая унификация осуществляется с помощью концепции БНД. Главная особенность информации, хранимой в БНД, заключается в ее структурированности. Основные преимущества информационного взаимодействия БНД заключаются в следующем:

- снимаются ограничения на число обслуживаемых проектных процедур;

- возможно развития и модификация программной системы;

- возможна модификация модернизация технических средств для хранения данных без изменения ППП;

- обеспечивается целостность данных.

Однако реализация информационных связей через БНД данных имеет и свои недостатки, связанные главным образом со значительными затратами времени на поиск данных в БД.

Рис. 1. Граф, отражающий связи по управлению.

Рис. 2. Граф, отражающий связи по информации.

Рис. 3. Реализация информационных связей через СУБД.

## **3. Состав САПР электронных систем**

Современная САПР представляет собой сложный программно-аппаратный комплекс, именуемый в научно-технической литературе как "рабочая станция" (PC).

Рис. 3. Структура рабочей станции проектирования электронных систем.

Рис. 4. Структура ПО САПР.

## **4. Иерархические уровни представления электронных устройств**

Основным методом проектирования с применением САПР является блочно-иерархический метод или метод декомпозиции сложного объекта на подсистемы (блоки, узлы, компоненты). В этом случае описание сложной системы разделяется на иерархические уровни (уровни абстрагирования) по степени подробности отражения свойств системы. На каждом уровне представления проекта существует свое понятие системы, подсистемы, элемента системы, закона функционирования элементов системы в целом и внешних воздействий.

Именно эти понятия определяют тот либо иной уровень иерархии представления устройства. Подсистема - это часть системы, которая представляет собой совокупность некоторых ее элементов, выделенных по определенному функциональному признаку, и подчиняется по своей цели функционирования единой цели функционирования всей системы. Под элементом системы понимают ее часть, выполняющую определенную функцию (функции) и не подлежащую декомпозиции при данном уровне рассмотрения. Неделимость элемента - это понятие, но не физическое свойство этого элемента. Оперируя понятием элемент проектировщик оставляет за собой право перейти на другой уровень на основании части или объединив несколько элементов в один.

На верхнем иерархическом уровне рассматривается весь сложный объект как совокупность взаимодействующих подсистем. На следующем иерархическом уровне подсистемы рассматриваются отдельно как системы, состоящие из некоторых составных частей (элементов), и имеют большую подробность описания. Данный иерархический уровень является уровнем подсистем. Количество уровней иерархии всегда ограничено. Уровни характеризуются тем, что множество типов элементов, из которых может быть составлена подсистема проектирования, ограничено. Такое множество называется базисом уровня.

Метод декомпозиции порождает серьезные проблемы при создании САПР:

определение уровней иерархии и базисов для них;

разработка математического обеспечения;

отображение из одного базиса в другой и др.

Метод иерархического представления проектируемого объекта, используемый разработчиками электронных схем и систем, может базироваться на двух способах представления (описания) элементов: структурном и поведенческом.

Структурный способ предусматривает описание элемента системы как совокупности взаимосвязанных элементов более низкого уровня, тем самым определяя базис этого уровня. Структурная форма иерархии проекта подразумевает процесс декомпозиции или разбиения проекта так, что на любом уровне, который выбирается для моделирования, модель системы строится как совокупность взаимосвязанных элементов, определенных для данного уровня. Здесь сразу же возникает вопрос: каким образом определяются эти элементы? Чаще всего они формируются с использованием элементов следующего, более низкого уровня. Таким образом, как показано на рис. 5, проект может быть представлен в виде дерева, причем различным уровням иерархии абстракций соответствуют свои уровни этого дерева. На уровне листьев дерева определяется поведение элементов проекта самого низкого уровня. Поведенческий способ предусматривает описание элемента системы по зависимостям вход/выход при помощи некоторой процедуры. Причем это описание определяется некоторой собственной процедурой, а не описывается с использованием других элементов. Поэтому поведенческая модель используется для описания элементов уровня листьев дерева проекта. Поскольку поведенческая модель некоторого проекта может существовать на любом уровне, различные части проекта могут иметь поведенческие описания на разных уровнях.

Рис. 5. Проект, представленный в виде полного (а) и неполного (б) дерева.

На рис. 5 (а) показано "полное" дерево проекта, где все поведенческое описание формируется на одном и том же уровне. На рис 5 (б) показан проект, представленный в форме неполного дерева, где поведенческие описания относятся к различным уровням. Подобная ситуация возникает потому, что разработчику зачастую желательно строить и анализировать взаимосвязи между системными компонентами еще до завершения проектирования. Таким образом, не обязательно иметь спецификации всех системных компонентов, например, на уровне логических вентилей, чтобы получить возможность контроля проекта в целом на отсутствие ошибок. Такой контроль осуществляется при помощи многоуровневого моделирования, то есть моделирования, при котором поведенческие описания моделей компонентов относятся к различным уровням иерархии. Важным дополнительным достоинством подобного подхода является то, что он способствует повышению эффективности моделирования.

C точки зрения разработчика аппаратных средств можно выделить шесть основных уровней иерархии, представленных на рис. 6.

Рис. 6. Уровни иерархии представления электронных систем.

Это системным, микросхемный (или ИС), регистровый , вентильный, схемный и топологический уровни. Из рисунка видно, что иерархия уровней представления имеет форму усеченной пирамиды. Расширение пирамиды книзу отображает увеличение степени детализации, т.е. количества элементов, которые необходимо учитывать при описании проектируемого устройства на этом уровне.

В табл. 1 приведена характеристика уровней - указываются элементы структуры и поведенческое представление для каждого уровня.

Таблица 1. Иерархия моделей

|  |  |  |
| --- | --- | --- |
| Уровень | Структурные примитивы | Формальный аппарат для поведенческого представления |
| Системный | Центральные процессоры, коммутаторы, каналы, шины, запоминающие устройства и др. | Системный анализ, теория игр, теория массового обслуживания и др. |
| Микросхемный | Микропроцессоры, ЗУПВ, ПЗУ, УАПП, и др. | Входные-выходные зависимости, ГСА |
| Регистровый | Регистры, АЛУ, счетчики, мультиплексоры, дешифраторы | Теория цифровых автоматов, таблицы истинности, ГСА |
| Вентильный |  Логические вентили, триггеры | Алгебра логики, системы логических уравнений |
| Схемный | Транзисторы, диоды, резисторы, конденсаторы | Теория электрических цепей, системы линейных, нелинейных, дифференциальных уравнений |
| Кремниевый | Геометрические объекты | нет |

На самом нижнем уровне, кремниевом, в качестве базовых примитивов используются геометрические формы, которые представляют области диффузии, поликремния и металлизации на поверхности кремниевого кристалла. Соединение этих форм как бы имитирует процесс изготовления кристалла с точки зрения разработчика. Здесь представление только чисто структурное(не поведенческое).

На следующем, более высоком уровне, схемном, представление проекта формируется с использованием межсоединений традиционных активных и пассивных элементов электрической схемы: резисторов, конденсаторов и биполярных и МОП-транзисторов. Соединение этих компонентов используется для моделирования поведения электрической схемы, выражаемого с помощью зависимостей между напряжениями и токами.Для поведенческого описания на этом уровне можно использовать дифференциальные уравнения.

Третий уровень-уровень логических вентилей, традиционно играет основную роль при проектировании цифровых схем и систем. Здесь используются такие базовые элементы, как логические вентили И, ИЛИ и НЕ и различные типы триггеров. Соединение этих примитивов позволяет обрабатывать комбинационные и последовательностные логические схемы. Формальный аппарат для поведенческого описания на этом уровне- булева алгебра.

Выше вентильного уровня в иерархии находится регистровый уровень. Здесь базовые элементы - это такие компоненты, как регистры, счетчики, мультиплексоры и арифметико-логические устройства (АЛУ). Поведенческое представление проекта на регистровом уровне возможно с использованием таблиц истинности, таблиц состояний и языков регистровых передач.

Над регистровым уровнем находится уровень микросхем (или ИС). На микросхемном уровне в качестве элементов выступают такие компоненты, как микропроцессоры, устройства основной памяти, последовательные и параллельные порты и контроллеры прерываний. Хотя границы микросхем являются и границами моделей элементов, возможны и другие ситуации. Так, набор микросхем, которые совместно образуют одно функциональное устройство, можно представить как один элемент. Показательным примером здесь может служить моделирование разрядно - модульного процессора. Возможен и альтернативный вариант - когда элементы представляют отдельные секции одной микросхемы, например, на этапе анализа технического задания и декомпозиции. Главной особенностью здесь является то, что элементом представляется большой блок логики, где для длинных и зачастую сходящихся трактов обработки данных необходимо представлять зависимости выходов от входов. Как и в случае элементов нижележащих уровней, элементы микросхемного уровня не строятся иерархически из более простых примитивов, а представляют собой единые объекты-модели. Так, если нужно моделировать последовательный порт ввода-вывода (универсальный асинхронный приемопередатчик, УАПП), соответствующую модель не строят путем соединения более простых функциональных моделей таких блоков, как регистры и счетчики, здесь сам УАПП становится базовой моделью. Модели такого типа важны для изготовителей комплексного оборудования, которые приобретают микросхемы у других фирм-изготовителей, но не знают их внутренней структуры уровня логических вентилей, поскольку это является обычно секретом фирмы. Поведенческое описание модели микросхемного уровня строится на основе входной-выходной зависимости каждой конкретной ИС-алгоритма,реализуемого данной ИС. Верхний уровень - это системный уровень. В качестве элементов этого уровня используются процессор, память и коммутатор (шина) и др. Поведенческое описание на этом уровне включает такие основные данные и характеристики, как, например, показатель быстродействия процессора в миллионах команд в секунду (мегофлопсы) или пропускная способность тракта обработки данных (бит/с). Из табл. 1 и вышеизложенного видно, что структурные или поведенческие характеристики соседних уровней в определенной степени перекрываются. Например, и на регистровом и на микросхемном уровне может использоваться представление при помощи ГСА. Однако структурное представление для обоих уровней совершенно различно, поэтому они и разделяются. Микросхемный и системный уровень имеют по сути одни и те же элементы, однако они абсолютно различны по своим поведенческим характеристикам. Так, поведенческие модели уровня ИС позволяют вычислять детальные отдельные реакции в виде значений целых чисел и битов. А поведенческому представлению системного уровня свойственно серьезное ограничение - оно служит преимущественно для моделирования пропускной способности системы или определения стохастических параметров системы. На практике представление проекта на системном уровне используется главным образом для сравнительной оценки различных архитектур. В общем, следует использовать модели разного уровня, если требования либо поведенческого, либо структурного характера различны.

Последнее понятие, связанное с иерархическим представлением проекта, - это так называемое окно проекта.

Этим термином обозначается группа уровней дерева проекта, с которыми работает каждый конкретный разработчик. Так, окно проекта для разработки СБИС охватывает кремниевый, схемный, вентильный, регистровый и микросхемный уровни. Разработчика вычислительной машины, с другой стороны, обычно интересует окно, охватывающее вентильный, регистровый, микросхемный и системный уровни. Именно концепция окна проекта является основой для многоуровневого проектирования. С ростом сложности СБИС станет нецелесообразно включать вентильный уровень в окно проекта, поскольку на одном кристалле можно будет разместить сотни тысяч логических вентилей. Регистровый уровень, хотя он безусловно имеет меньшую сложность, чем вентильный, может также содержать необязательные подробности для тех, кого интересуют только сигналы ввода-вывода СБИС.

Таким образом, с точки зрения разработчика машины сама СБИС будет становиться элементом проекта.

Рис. 7. Пример реализации уровней представления мультипроцессорной системы.