

ИНТЕРКОМПЬЮТЕР

INTERCOMPUTER

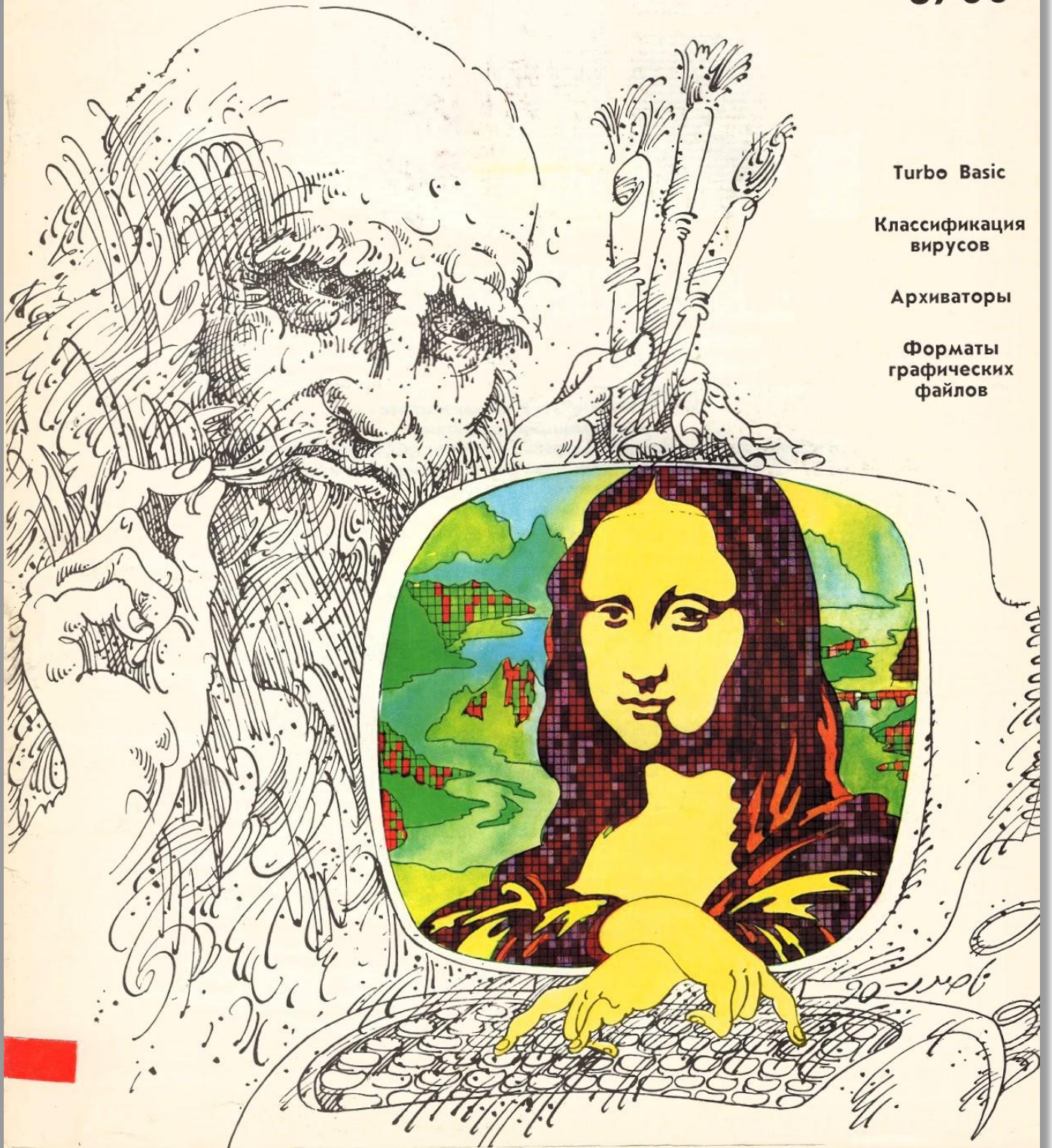
3/90

Turbo Basic

Классификация
вирусов

Архиваторы

Форматы
графических
файлов



ВОКРУГ САПР



А.И. МАСАЛОВИЧ

ОБМЕН ДАННЫМИ
МЕЖДУ

САПР

Едва справившись с проблемой выбора наиболее подходящей системы автоматизированного проектирования (САПР), ее пользователь обнаруживает новую проблему: как разобраться в обилии форматов хранения проектных данных и организовать эффективный обмен данными с другими САПР и автоматизированным производством? Технология изготовления электронной аппаратуры развивается так стремительно, что стандарты в этой области устаревают, едва родившись, и начинающие пользователи буквально теряются в море предлагаемых различными фирмами текстовых, графических, табличных и иных форматов хранения и передачи данных. В статье рассматриваются три островка стабильности в этом неспокойном море - форматы IGES, EDIF и язык VHDL, ведущая роль которых в организации взаимодействия между САПР признается в настоящее время большинством фирм-разработчиков программного обеспечения.

Проблема совместимости САПР

Освоение любой САПР печатных плат и БИС требует организации обмена данными с другими САПР и автоматизированным производством. Легкость использования, широта возможностей, эффективность реализации - все эти достоинства, которыми может обладать та или иная новая система, будут сведены на нет неуклюжим интерфейсом ввода-вывода проектных данных. Если Ваша новая система совместима по форматам хранения и передачи данных с уже освоенной САПР и Вы избежали проблем их стыковки, считайте, что Вам повезло. Чтобы решить проблему совместимости, каждая фирма-разработчик САПР обеспечивает некоторые общие форматы хранения и передачи данных для своих изделий. Между группами фирм существуют также соглашения о совместных разработках в данной области. Кроме того, многие фирмы поддерживают определенные стандарты, к числу которых и относятся рассматриваемые в данной статье стандарты фор-

матов IGES и EDIF, а также стандарт языка VHDL, высокие качества которых подтверждаются многолетним опытом их применения.

IGES

В сентябре 1979 г. Национальное бюро стандартов США, а также фирмы General Electric и Boeing в рамках программы ICAM (Integrated Computer-Aided Manufacturing), финансируемой Министерством обороны США и Национальным управлением по авионавигации и исследованию космического пространства, объединили усилия для разработки способов обмена данными между САПР. Результатом этих усилий явилось создание формата IGES (Initial Graphics Exchange Specification). Первоначально он был представлен в отчете Национального бюро стандартов, однако его характеристики заинтересовали Американский национальный институт стандартов (ANSI), и в мае 1980 г. подкомитет Y14.26 ANSI опубликовал версию 1.0 стандарта этого формата, снабдив его вводной частью и разделом, посвященным

математическому описанию трехмерных объектов. Первая крупномасштабная попытка стандартизации представления графических данных вызвала широкий резонанс в научных и промышленных кругах. В течение следующего года в ANSI было внесено 384 предложения по усовершенствованию нового стандарта, 280 из которых были учтены. Модернизированная версия, наиболее широко известная в настоящее время, была опубликована в сентябре 1981 г. (ANSI Y14.26M Stand. for Geometry Interchange/American National Standards Institute. - New York, 1981).

Стандарт пришелся "ко двору", и уже через год 32 фирмы представили трансляторы, обеспечивающие возможность использования формата IGES для САПР, объем продаж которых составлял 81% от общего объема продаж на рынке САПР.

Какие же принципы положены в основу формата IGES?

Файл данных в формате IGES представляет собой последовательность 80-байтовых текстовых записей, содержащих информацию о геометрических элементах проектируемого изделия. Выбор такой длины записи, сделанный в свое время в расчете на операционную систему System/360, в настоящее время является атавизмом, хотя и находит понимание у пользователей ЕС ЭВМ.

Базовая единица представления информации в формате IGES - элемент. Элементы делятся на следующие основные типы:

- геометрические (точки, линии, дуги, плоскости, параметрические поверхности и т.д.);
- символьные (описания текстов, размеров, символов оформления и т.д.);
- структурирующие (описания отношений, существующих между компонентами проектируемого изделия).

Элемент любого типа состоит из каталога и параметризованных данных. Каталог имеет фиксированную длину, длина данных переменная. Определение структурирующих эле-

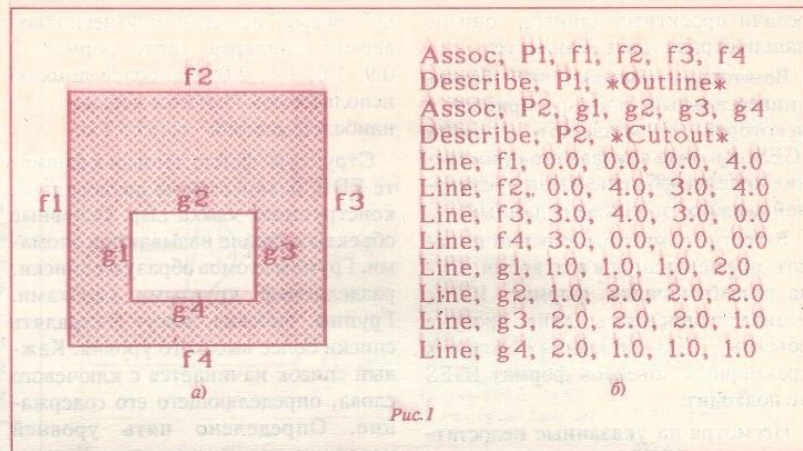


Рис. 1

ментов осуществляется с помощью макроопераций. Для описания элементов предусмотрен набор инструкций, в который входят определение (SET), присваивание (LET), повторение (REPEAT) и ссылка на другие операции (MREF).

Каждая 80-байтовая запись файла данных в формате IGES содержит в 73-й позиции символ, указывающий тип записи. Например, символ D означает, что запись относится к каталогу.

Для задания операндов предусмотрены следующие типы данных: целые числа, вещественные числа, цепочки литер, указатели. Кроме того, в файле могут встречаться инструкции языка описания, а также может быть использован свободный формат.

Файл данных в формате IGES состоит из пяти разделов:

1. Начальный раздел (S в 73-й позиции) содержит вводную информацию для пользователя о проектируемом изделии.

2. Глобальный раздел (G в 73-й позиции) предназначен для настройки препроцессора. В нем указаны такие параметры, как число бит в порядке и мантиссе действительного числа, номер версии транслятора, номер версии стандарта и т.д.

3. Каталог (D в 73-й позиции) содержит общую для всех элементов файла информацию и указатели на разделы данных и другие каталоги.

4. Раздел данных (P в 73-й позиции) содержит описание основных

элементов, составляющих проектируемое изделие.

5. Завершающий раздел (T в 73-й позиции) представляет собой заключительную строку в файле.

В Разделе данных могут использоваться записи о проектных данных четырех видов:

1. Геометрические данные. Определяют тип элемента (дуга, сплайн и т.д.) и его геометрию (координаты конечных точек, центров дуг и т.д.), задают параметры отображения (матрицу преобразования, толщину линий и т.д.), а также содержат информацию о составных частях элемента.

2. Данные о визуализации. Указывают, какой из двух типов отображения выбран для данного элемента.

3. Надписи.

4. Данные о структурировании.

На рис. 1 показаны контур несложной детали (a) и соответствующий фрагмент его описания в формате IGES (b).

Столь объемная работа, какой явилась разработка стандарта формата IGES, предпринималась впервые, поэтому наряду с несомненными достоинствами, обусловившими широкое распространение этого стандарта, он обладает и рядом недостатков.

Во-первых, как всякий стандарт, IGES с трудом поддается расширению и модификации. Несмотря на то, что за время, прошедшее со дня его разработки, появились новые решения в организации хранения и пе-

редачи проектных данных, они не нашли отражения в стандарте.

Во-вторых, попытка разработать универсальный стандарт привела к некоторой громоздкости формата IGES, и в ряде случаев его применение нецелесообразно из-за чрезмерной сложности описания данных.

В-третьих, математический аппарат, применявшийся в то время, когда разрабатывался формат IGES, позволял описывать лишь простые объекты. Для описания сложных трехмерных объектов формат IGES не подходит.

Несмотря на указанные недостатки, формат IGES сейчас чрезвычайно распространен и интенсивно используется в системах конструкторского проектирования. Достаточно сказать, что система AutoCAD, признанный лидер в этой области, располагает возможностью обмена данными в формате IGES.

EDIF

Формат IGES разрабатывался в основном для графического описания механических конструкций с использованием двух- и трехмерных геометрических построений. Область его применения - конструкторское проектирование. Специфические требования разработчиков электронной аппаратуры вызвали необходимость создания формата, более подходящего для решения стоящих перед ними задач. Различными фирмами было разработано несколько форматов для описания электронной аппаратуры: IDF, CIDF, GAIL, TIDAL и др. Многие из этих форматов успешно используются и в настоящее время. В начале 80-х гг. у сотрудников трех американских фирм-изготовителей БИС - Motorola, Texas Instruments и National Semiconductor - появилась идея создания формата, объединяющего в себе достоинства этих форматов, которые к тому времени уже достаточно хорошо себя зарекомендовали. В 1983 г. к разработке формата, получившего название EDIF (Electronic Design Interchange Format), подключились фирмы Tektronix, Mentor Graphics и Daisy Systems. В настоя-

щее время используется несколько версий стандарта этого формата - 0.9, 1.0, 1.1, 2.0. Наиболее широко используемой является версия 1.0, а наиболее мощной - версия 2.0.

Структура файла данных в формате EDIF базируется на древовидных конструкциях языка Lisp. Основные объекты в файле называются атомами. Группы атомов образуют списки, разделяемые круглыми скобками. Группы списков могут составлять списки более высокого уровня. Каждый список начинается с ключевого слова, определяющего его содержание. Определено пять уровней иерархии ключевых слов. Ключевое слово EDIF, находящееся на верхнем уровне, служит для идентификации файла. На следующем уровне находятся слова status (состояние), design (проект), library (библиотека) и userdate (данные пользователя), определяющие содержание разделов файла данных в формате EDIF. Следующий уровень задается ключевым словом cell (ячейка), предвещающим описание проектируемых элементов. Внутри описания ячейки содержится раздел view (вид), который, в свою очередь, делится на подразделы interface (интерфейс) и contents (содержимое).

Для каждого из уровней иерархии определен набор ключевых слов, позволяющий конкретизировать их содержание. Кроме того, регламентированы синтаксис описания каждой конструкции и допустимые типы данных.

Такое построение файла данных в формате EDIF требует наличия средств обработки списков для трансляторов. Многие трансляторы для формата EDIF имеют структуру, заимствованную у компиляторов языка Lisp.

Для файлов данных в формате EDIF установлены три уровня сложности, определяющих разные требования к программам обработки данных. В файлах уровня 0 допускается использование только простых констант и примитивов. Файлы уровня 1 содержат константы, переменные и выражения. Наконец, в файлах уровня 2 допустимо также использова-

ние процедур и сложных типов данных. Такое разделение делает формат EDIF пригодным как для сложных интегрированных САПР, так и для простых систем на персональных компьютерах.

Широкое использование версии 1.0 стандарта позволило сформулировать ряд предложений по его улучшению. На их основе были созданы версии 1.1 и 2.0.

Последняя версия стандарта - 2.0, помимо возможностей, которыми располагают ранние версии стандар-

ФАКТЫ

- Специалисты фирмы AT&T успешно продемонстрировали макет первого в мире оптического процессора - устройства, которое выполняет вычисления с помощью оптических призм, линз, зеркал, светочувствительных микросхем и лазерных диодов. Устройство смонтировано на квадратном шасси со стороной около 70 см. но разработчики надеются "втиснуть" процессор на подложку площадью 18 см². Теоретически процессор, собранный из оптических компонентов, может обеспечить быстроедействие до миллиарда операций в секунду. Многие специалисты скептически относятся к возможности практической реализации такого процессора, признавая, однако, целесообразность использования света для передачи сигналов между электронными компонентами. Сотрудники фирмы AT&T утверждают, что в 1995 г. суперкомпьютеры будут на 20-30 % состоять из оптических элементов, а к 2000 г. доля таких компонентов возрастет до 50 %.

та, имеет средства описания имитационных моделей и печатных плат. Кроме того, в ней устранены некоторые недостатки, присущие ранним версиям. В частности, исключено контекстно-зависимое употребление ключевых слов, а описание входных сигналов синтаксически отделено от описания портов. В этой версии введены новые ключевые слова Pcblayout (аналогично Masklayout) и Graphics, а также изменена семантика ключевых слов Netlist и Schematic. Специальные меры были приняты для уменьшения частоты использования конструкции userdata, наиболее сложной для компиляции. Определенные изменения, внесенные в формат EDIF, позволяют описывать устройства более компактно. На рис.2 приведены схема печатной платы (а) и пример ее описания в формате EDIF 2.0 (б) (Electronic Design Interchange Format // Computer Design. - 1987. - Vol.26, N 3. - P.24 - 28).

С 1987 г. крупнейшие разработчики БИС - фирмы Daisy Systems, Mentor Graphics, Valid Logic, Hewlett Packard - уже используют в своих САПР версию 2.0 стандарта, а многие другие стремятся к этому.

Версия 2.0, позволившая заметно улучшить стандарт, тем не менее не устранила всех проблем в его использовании. Так, большие сложности связаны с формированием библиотек компонентов. Отсутствие стандартного набора символов для описания имитационных моделей элементов уменьшает совместимость между различными системами. Решение этой проблемы может потребовать совместных усилий фирм-разработчиков электронной аппаратуры.

VHDL

Хотя формат EDIF и содержит конструкции для описания имитационных моделей, он не ориентирован на использование в качестве высокоуровневого языка проектирования в САПР заказных БИС. Долгое время разработчики крупных заказных кристаллов испытывали потребность в средствах, которые бы позволяли при проектировании оперировать

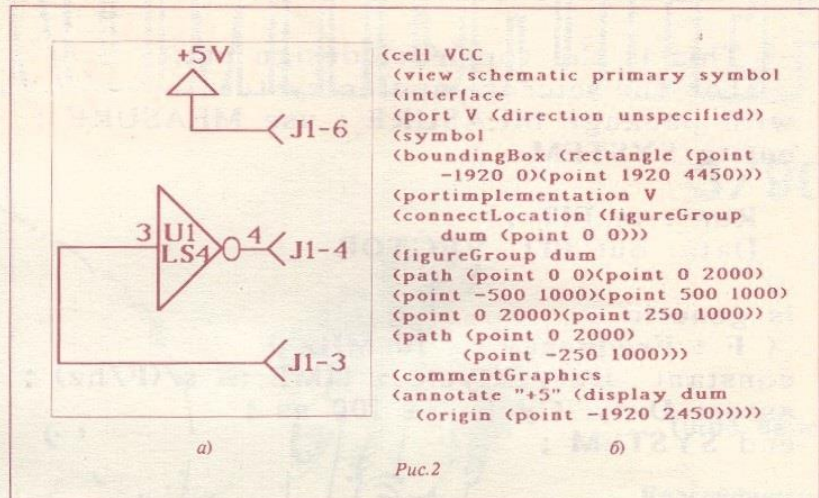


Рис.2

описаниями сложных поведенческих моделей электронных устройств. Блестательная работа К. Мида и Л. Конвей (С. Mead, L. Conway. Introduction to VLSI Systems. - Addison-Wesley, 1980), в которой были изложены теоретические основы метода "кремниевой компиляции", положила начало интенсивным разработкам языков подобного назначения. Министерство обороны США, развернувшее в марте 1980 г. крупные работы по созданию сверхбыстродействующих БИС (Проект VHSIC), также испытывало потребность в языке, который бы содержал средства описания архитектуры, а также поведенческих и структурных моделей сложных БИС. Летом 1981 г. в Институте анализа проблем обороны США была создана лаборатория для определения требований к такому языку. Институтом были также заключены контракты на двухэтапную разработку языка с фирмами Intermetrics, IBM и Texas Instruments. Создаваемый язык получил название VHDL (VHSIC Hardware Description Language). К нему предъявлялись следующие основные требования:

- возможность иерархического описания устройств на всех уровнях - от системы до логических вентилей;
- независимость от технологии;
- наличие средств документирования процесса разработки;

- возможность использования для управления моделированием;
- расширяемость.

Начало разработки языка VHDL по времени совпало с периодом широкого распространения языка программирования Ада, также созданного по заказу Министерства обороны США. Выразительные средства языка Ада, ранее не встречавшиеся в языках такого класса, - абстрактные типы данных, пакеты, средства управления видимостью переменных и т.д. - обусловили выбор его в качестве основы для разработки языка VHDL. И действительно, язык VHDL вообрал в себя все основные языковые конструкции и особенности синтаксиса языка Ада.

Основное понятие в языке VHDL - проектный элемент (design entity). Описание проектного элемента делится на подразделы interface (интерфейс) и тело (body). Интерфейс определяет порты ввода-вывода проектируемого электронного устройства и их характеристики, в теле описывается возможный вариант реализации устройства. Описание проектного элемента может содержать информацию о модели устройства любой сложности - от логического вентилля, схемы управления до компьютерной системы.

Такой подход к представлению проектных элементов облегчает иерархическую декомпозицию проектируемого устройства. Пример

```
-- This is the top-level design entity
   for the microcomputer system
   with package MEASURE ; use MEASURE ;
   entity SYSTEM
   (
     Run : in BIT ;
     Data: out BIT_VECTOR
   )
   is generic
     ( F : Frequency := 10 MHz )
   constant Duty_Cycle : TIME := s/(F/hz) ;
   assert Duty_Cycle >= 100 ns ;
   end SYSTEM ;
```

Рис.3

обобщенного описания микропроцессорного устройства на языке VHDL приведен на рис.3 (R. Lipsett et al. VHDL - The Language //IEEE Design & Test. - 1986. -N 4. - P.28 - 41).

Интерфейс может содержать как информацию, общую для всех возможных вариантов реализации элемента (количество и тип входных и выходных контактов и т.п.), так и конкретные значения, определяющие требования к его реализации (точное время задержки, нагрузочную способность и др.).

В теле содержится описание поведения (behavioral) либо архитектуры (architectural) проектируемого устройства.

Описание поведения определяет входные характеристики элемента и алгоритм его работы (алгоритм задается с помощью операторов присваивания, ветвления и цикла). Операторы выполняются последовательно. В описании поведения отсутствует информация о внутренней структуре элемента.

Описание архитектуры задает процесс преобразования данных внутри устройства. Операторы описания выполняются параллельно под управлением условных булевых выражений. Оператор выполняется, когда значение его управляющего выражения становится истинным либо

когда изменяется один из входных параметров.

В теле элемента можно использовать описание как архитектуры, так и поведения проектируемого устройства, однако если требуется конкретизировать его внутреннюю структуру, должно быть выбрано первое.

В языке VHDL предусмотрены средства для описания регулярных частей кристалла и генерации макрорасширений внутри описания архитектуры.

Способ управления моделированием в языке VHDL во многом схож

принятым в языке Conlan, например, в нем поддерживаются две системы единиц измерения времени (микро- и макро- шкалы).

В языке VHDL предусмотрены средства для порождения новых типов данных и группирования описаний данных в пакеты.

Язык VHDL, разрабатываемый как средство для описания электронной аппаратуры, не предназначался для организации обмена данными между САПР. Тем не менее в настоящее время именно VHDL становится ведущим языком для исходного описания моделей в интегрированных САПР БИС, и ему уготовлена роль фактического стандарта описания данных при проектировании крупных заказных БИС. Ведущие фирмы-разработчики БИС уже в настоящее время требуют представления входных данных, предназначенных для проектирования кристаллов, на языке VHDL.

Переняв многие достоинства языка Ада, VHDL унаследовал и некоторые присущие ему недостатки. Обилие сложных синтаксических конструкций в языке требует разработки громоздких трансляторов. Для больших ЭВМ эта проблема преодолима, однако она является тормозом при использовании VHDL на персональных компьютерах. В поисках выхода из этой ситуации некоторые фирмы предлагают САПР, в которых используется то или иное подмножество языка VHDL. Специалисты считают, что с развитием САПР на базе 32-разрядных персональных компьютеров проблема трансляторов для языка VHDL будет решена.

Использование персональных компьютеров в САПР приобретает все большие масштабы. В настоящее время число таких САПР измеряется сотнями, а число форматов передачи данных - десятками. Наряду с рассмотренными выше форматами широко используются также формат описания топологии БИС CIF, язык описания печатных плат и процесса их изготовления IPC, формат описания графических данных в растровой форме TIFF и многие другие форматы. Однако описанные в данной статье форматы играют ведущую роль в решении проблемы взаимодействия между САПР, и поэтому при организации хранения и обмена проектными данными целесообразно обратить на эти форматы самое пристальное внимание.

■ ФАКТЫ

■ Фирма IBM объявила, что будет устанавливать разработанную фирмой NeXT операционную среду NextStep на своих рабочих станциях и ПК семейства PS/2, работающих под управлением ОС AIX (версии ОС UNIX, разработанной фирмой IBM). В настоящее время на компьютерах фирмы NeXT эксплуатируется ОС Mach - версия ОС UNIX, разработанная в Университете Карнеги-Меллона. Операционные системы AIX и Mach несовместимы на уровне двоичных кодов, поэтому программы, разработанные под управлением одной из них, до использования под управлением другой необходимо перекомпилировать.