

8. Из ЭВМ «А» по каналу прямого доступа передается 514 байт (2 байт контрольной суммы (КС)).

9. ЭВМ «В» возвращает символ «\*» (подтверждение успешного приема информации) или символ «#» (завершение приема), или «С» (ошибка КС), или «Т» (тайм-аут при приеме буфера). В двух последних случаях передача повторяется до 3 раз.

10. При завершении передачи файла (успешном или неуспешном) программа XFR готова к передаче следующего файла.

Принем данных в ЭВМ «А» из ЭВМ «В». 1 и 2. Аналогичны протоколу передачи.

3. Пользователь на ЭВМ «А» задает принимаемый файл командой «XFR > R <спецификация файла> <BK>».

4. Из ЭВМ «А» передается строка «XSL<BK>», что приводит к запуску на ЭВМ «В» подчиненной программы обмена XSL.

5. ЭВМ «А» получает подтверждение успешного запуска XSL.

6. Из ЭВМ «А» передается символ «R» (прием), а затем спецификация файла и открывается файл для приема данных.

7. ЭВМ «А» получает подтверждение, что файл данных открыт на подчиненной ЭВМ и готов для передачи. В качестве подтверждения ЭВМ «В» передает спецификацию файла и символ «\*». Если файл с данной спецификацией не может быть открыт, передается символ «F», а если файл не найден, — символ «E».

8. Из ЭВМ «А» передается символ «\*» (готовность к приему данных).

9. Из ЭВМ «В» по каналу прямого доступа передается 514 байт данных с контрольной суммой.

10. Из ЭВМ «А» передается символ «\*» (подтверждение успешного приема информации) или символ «#» (завершение приема), или символ «С» (ошибка КС), или символ «Т» (тайм-аут при приеме буфера). В двух последних случаях передача повторяется до 3 раз.

11. При завершении передачи файла (успешного или неуспешного) программа XFR закрывает принятый файл и готова к передаче следующего файла в том либо другом направлении.

Рассмотренные программные средства являются достаточно эффективными для управления потоками информации при решении широкого класса задач проектирования изделий электронной техники.

#### ЛИТЕРАТУРА

1. Власов А. М., Дыбой В. А., Межов В. Е. и др. Аппаратные средства связи в многопроцессорных унифицированных интерактивных графических системах типа «Кулон». — Автометрия, 1986, № 4.
2. Толстых Б. Л., Талов И. Л., Харин В. Н. и др. Унифицированные интерактивные средства проектирования изделий электронной техники. — М.: Радио и связь, 1984.

Поступило в редакцию 15 июля 1985 г.

УДК 621.3.049.77.001.2

Е. Г. ЮРАШАНСКИЙ, К. Э. ЮРИН  
(Новосибирск)

#### СИСТЕМА ИНТЕРАКТИВНОГО РЕДАКТИРОВАНИЯ ТОПОЛОГИИ СБИС

В статье рассматривается разработанная в ИАиЭ СО АН СССР система интерактивного редактирования топологии СБИС, реализованная на основе мини-ЭВМ «Север» [1] и цветного графического дисплея [2].

Высокая производительность и большой объем оперативной памяти ЭВМ «Север» позволяют осуществить одновременную работу многих пользователей. Графический дисплей обеспечивает отображение одновременно до 256 цветов с разрешением по полю памяти до  $2048 \times 2048$  точек. Реализация дисплея на базе микропроцессора позволяет значительно повысить автономность выполнения графических функций.

Пользователю предоставляется два режима работы с системой — директивный и экранный.

В директивном режиме пользователь имеет возможность изменять состояние системы (устанавливать масштаб, определять положение окна, число и порядок отображения слоев, дискретность рабочих сеток и т. п.), читать информацию о топологии из внешнего файла, заносить ее в рабочий буфер и записывать информацию из рабочего буфера во внешний файл, осуществлять редактирующие действия, подключать внешние библиотеки элементов, определять новые библиотеки и т. д.

В директивном режиме управление системой производится путем подачи команд, представленных именем директивы с возможным параметрами. Значения опущенных параметров при необходимости запрашиваются системой у оператора. Синтаксически команда может подразделяться на несколько отдельных слов, разделяемых символом «—». Допускается сокращенная запись имени директивы. В процессе работы оператор имеет возможность определять новые команды, представляющие собой произвольные комбинации базовых команд системы.

Основная работа по редактированию и созданию топологии осуществляется в экранном режиме. При переходе в экранный режим в качестве необязательного параметра могут указываться координаты левого нижнего угла экрана. При этом на экране дисплея отображается соответствующий фрагмент топологии СВИС. В экранном режиме управление системой производится с помощью функциональных клавиш и курсора. Результат применения редактирующих действий немедленно отражается на экране.

Наряду с точным указанием координат курсором, возможно округление координаты до ближайшего узла рабочей сетки. Для упрощения работы оператора применяется набор из трех сеток. Набор сеток и номер рабочей сетки оператор может менять, не выходя из экранного режима. Возможно одновременное отображение на экран информации о всех используемых слоях топологии или о группе активных слоев (слоев, над которыми разрешено осуществлять редактирующие действия), набор которых оператор может менять. Возможен просмотр всей топологии путем перемещения окна.

Элементы топологии представляют различными геометрическими фигурами (многоугольниками, шинами, прямоугольниками, кругами и т. д.) и их комбинациями. Ввод новых элементов топологии осуществляется путем определения типа вводимого элемента и последующего указания курсором характерных точек элемента, например, для многоугольника — координаты вершин, для круга — центр и точка на окружности, для прямоугольника — сторонами, не параллельными осям координат, — вершины боковой стороны и ширина, для шины — точки перегиба и т. д. После указания всех необходимых параметров элемента происходит его отображение на экране. Вводимый элемент размещается в рабочем слое (слое, где происходит генерация новых элементов), номер которого оператор может устанавливать, исходя из набора активных слоев.

Редактирующие действия можно осуществлять только над предварительно отобранными элементами, т. е. элементами, которые предварительно выделены как разрешенные к редактированию.

Отбор одиночного элемента осуществляется путем указания курсором произвольной точки, принадлежащей этому элементу. Отобранный таким образом элемент может подвергаться редактирующим действиям, заключающимся в удалении, перемещении, мультиплицировании, модификации и т. д.; в случае если точка, указанная курсором, принадлежит одновременно нескольким элементам, то по каждой команде отбора будет выделяться очередной элемент из числа указанных курсором. При этом отобранный элемент помечается на экране особым образом (миганием или специальным цветом). Возможен отбор элемента в активных слоях или только в рабочем слое.

Для отбора группы элементов используются три способа: первый — отбор точкой (при этом отбирается группа элементов, каждый из которых содержит точку отбора); второй — отбор областью (отбираются элементы, целиком попадающие в прямоугольную область отбора, указываемую двумя «диагональными» вершинами); третий — отбор секущей областью (отбираются элементы, пересекаемые прямоугольной секущей областью). Отбор группы может осуществляться как во всех используемых слоях, так и только в активных или рабочем.

Система реализована на языке высокого уровня Планк, обладает модульной структурой и подразделяется на восемь модулей: для работы со структурами данных, отображения графической информации на экран дисплея, описания дисплея, пользовательского интерфейса, виртуальной памяти, для работы с динамическими структурами данных, с внешними файлами — и главный модуль, в котором осуществляется вызов всех основных процедур. Система выполнена таким образом, чтобы обеспечить возможность простой адаптации к другому типу дисплея, которая осуществляется заменой модулей пользовательского интерфейса и описания дисплея.

В связи с ограниченными размерами логического адресного пространства 16-рядной мини-ЭВМ для размещения достаточного числа элементов топологии используется программно реализованная виртуальная память, позволяющая увеличить объем доступного пространства примерно до 30 Мслов. Виртуальная память с точки зрения пользователя представлена массивами размером до 64 Кслов (короткая память) и до 10 Мслов (длинная память), число которых задается при инициализации системы. Адресация виртуальной памяти производится путем указания номера памяти и номера слова (индекса массива) в ней. Каждой памяти однозначно соответствует поле из таблицы памяти, указывающее на голову списка блоков дескрипторов. Блок состоит из 32 дескрипторов, каждый из которых соответствует странице памяти размером 1 Кслов с определенным номером и содержит адрес страницы в физической памяти, признак действия над страницей и адрес, по которому производится сброс вытесняемой страницы на диск или чтение содержимого данной страницы с диска (рис. 1). Виртуальная память реализована в виде отдельного модуля. Для работы с ней используется ряд процедур: инициализация памяти, чтение и

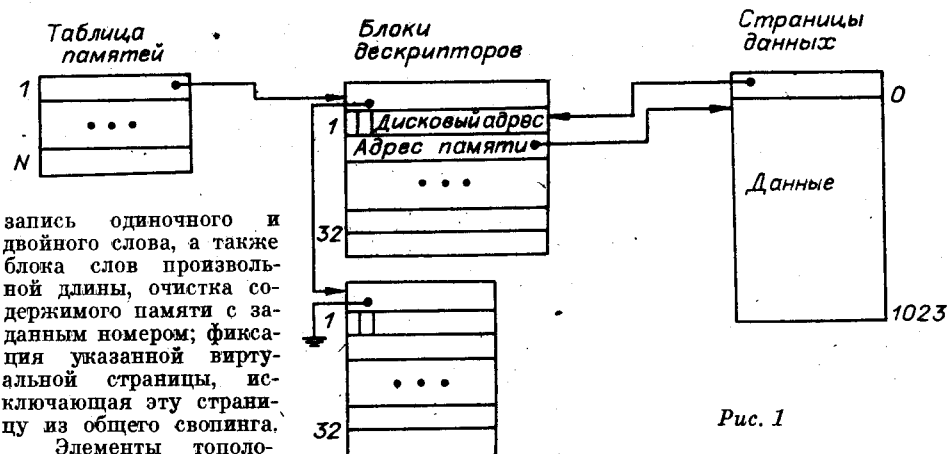


Рис. 1

запись одиночного и двойного слова, а также блока слов произвольной длины, очистка содержимого памяти с заданным номером; фиксация указанной виртуальной страницы, исключая эту страницу из общего свопинга.

Элементы топологии СВИС в программе представляются вариантными записями, постоянная часть которых содержит информацию о типе элемента, слове, в котором он определен, точке привязки элемента, его ориентации и размере записи. Переменная часть записи зависит от типа элемента и включает информацию, специфичную для конкретного типа элемента.

Записи формируются с помощью специальных процедур во время выполнения оператором действий по определению и (или) редактированию элементов.

Записи с описаниями элементов топологии связаны в однонаправленные списки, отсортированные по координатам точек привязки элементов. Для ускорения поиска элементов (в процессе отбора, клиппирования и т. п.) в программе используется ряд независимых списков, каждый из которых ставится в соответствие определенному диапазону координат по оси X.

Первый шаг при поиске элемента — определение номеров диапазонов, внутри которых может находиться X-координата точки привязки, и индексирование этими номерами в массивы диапазонов. Элементом массива диапазонов является указатель на список элементов, точки привязки которых принадлежат данному диапазону значений координаты X. Далее поиск осуществляется только внутри списков, соответствующих заданным диапазонам. Это позволяет, несмотря на большой объем базы данных, достаточно быстро осуществлять поиск элементов в процессе редактирования (рис. 2).

Все действия оператора с уже созданными элементами (удаление, модификация, перемещение, копирование и т. д.) сопровождаются модификацией соответствующих записей с описаниями, зависящей от вида действий, и (или) изменением положения записей в рабочих списках.

Списки элементов и массив диапазонов располагаются в длинной памяти. Массив диапазонов размещен в начале длинной памяти и занимает две страницы; для ускорения доступа к массиву диапазонов его страницы фиксированы в памяти.

Для представления топологии внутренняя структура может быть сведена к четырем базовым типам элементов: прямоугольник, многоугольник, круг, текстовая строка. Набор типов элементов, предоставляемых пользователю, может быть значительно шире, однако все производные типы элементов сводятся к данным базовым.

При необходимости некоторая совокупность базовых элементов может быть выделена в библиотечный элемент (макрос), который содержит, кроме базовых элементов, ссылки на другие библиотечные элементы без ограничения уровня вложенности. Он может создаваться в процессе работы или храниться в заранее под-

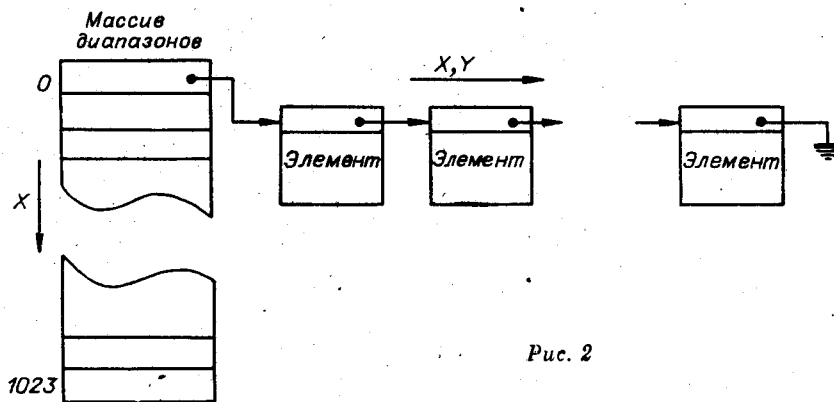


Рис. 2

готовленном библиотечном файле. Работа с библиотечным элементом осуществляется как с единым целым аналогично работе с базовыми элементами. Определение библиотечных элементов позволяет осуществлять иерархический подход к проектированию.

Библиотечные элементы в списках задаются в виде ссылки на тело элемента и точки привязки элемента. Структура ссылки аналогична структуре базовых типов элементов. Помимо обеспечения возможности определения уровней иерархии, библиотечные элементы позволяют существенно сократить объем базы данных. Его дальнейшее сокращение достигается благодаря применению итераций библиотечных элементов, что позволяет весьма кратко описывать регулярные структуры. Итерации представляются ссылкой на библиотечный элемент, заданием точки привязки, шагов по осям  $X$  и  $Y$ , а также количества библиотечных элементов по осям  $X$  и  $Y$ .

Описания тел элементов представляют собой сортированные списки, аналогичные рабочим спискам, хранящиеся в отдельной длинной памяти. Поскольку с телами библиотечных элементов не работают непосредственно, а только через ссылки на тело элемента, то разбиение списков на диапазоны не является необходимым и поэтому не применяется.

Результат работы системы — файл с полным описанием топологии, на основе которого в дальнейшем может быть получена программа для управления фото-построителем.

## ЛИТЕРАТУРА

1. Рассохин В. А., Трубкин В. А. Универсальная мини-ЭВМ на базе процессора с разрядно-модульной организацией. — Автометрия, 1986, № 4.
2. Остапенко А. М., Шеметов С. А. Цветной графический дисплей. — Автометрия, 1984, № 4.

Поступило в редакцию 14 февраля 1986 г.

УДК 681.3.06

Б. Х. ЗИНГЕР  
(Новосибирск)

### ОБ АЛГОРИТМАХ ПОСТРОЕНИЯ ПЛОСКОСТИ, РАЗДЕЛЯЮЩЕЙ КОНЕЧНЫЕ МНОЖЕСТВА ТОЧЕК

В системах трехмерной машинной графики, использующих для удаления невидимых поверхностей алгоритмы приоритетного типа, плоскость, разделяющая модель сцены на две части, позволяет глобально решить вопрос о взаимных приоритетах объектов, находящихся в разных полупространствах, образуемых этой плоскостью, а именно объекты, располагающиеся в одном полупространстве с наблюдателем, не могут закрываться объектами из другого полупространства. Такую технику можно считать классической [1—3]. В данном сообщении предлагаются соображения, которые могут составить основу алгоритмов построения разделяющих плоскостей.

Определение. Плоскость  $Ax + By + Cz + D = 0$  называется разделяющей множества  $U$  и  $V$ , если для любых  $u \in U$  и  $v \in V$   $d(u) \leq 0 \leq d(v)$  либо для любых  $u \in U$  и  $v \in V$   $d(v) \leq 0 \leq d(u)$ . Здесь  $d(p) = Ax + By + Cz + D$  при  $p = (x, y, z)$ .

Пусть имеется два конечных множества в  $R^3$   $U = \{u_1, \dots, u_n\}$  и  $V = \{v_1, \dots, v_m\}$ . Для простоты дальнейших рассуждений допустим, что каждое из множеств содержит четыре точки, не лежащие в одной плоскости.

Утверждение 1. Если существует плоскость, разделяющая  $U$  и  $V$ , то существует разделяющая плоскость, проходящая через три точки множества  $U \cup V$ , не лежащие на одной прямой.

Утверждение 1 позволяет строить алгоритм, основанный на переборе всех линейно независимых троек точек из  $U$  и  $V$  и попытке построить на них разделяющую плоскость. Учитывая, что проверка разделяемости множеств  $U$  и  $V$  данной плоскости может привести еще к одному перебору всех точек, сложность данного алгоритма будет пропорциональна  $(n + m)^4$ . Можно предложить ряд приемов сокращения перебора.

Если в исходных множествах выделены плоские грани, то из вершин, принадлежащих одной грани, достаточно проверить одну тройку.

При проверке разделяемости множеств плоскостью можно выбирать точки поочередно из одного и из другого. Таким способом можно раньше обнаружить факт неразделяемости и перейти к проверке следующей тройки.

Если предварительно построить выпуклые оболочки обоих множеств (с использованием, например, алгоритма [4]), то достаточно рассматривать только точки,