

информацией с ними сводятся к директивам переписи файлов. Такие директивы подаются непосредственно с терминала с использованием дополнительного протокола.

Программа доступа к ВУ ЭВМ КР выполняет следующие функции:

1) прием и анализ запроса на обмен с ВУПД, ввод/вывод данных с ИМД, прием/передача данных по линии связи (основной протокол);

2) после синхронизации обмена между ЭВМ «Электроника-60» и ЭВМ КР производится через резидентный драйвер; при этом пользователь работает в среде операционной системы RT-11 и может располагать всеми ее возможностями: редакторами, трансляторами, вспомогательными программами, драйверами и т. д. В частности, пользователь может написать драйвер для любого устройства (в том числе резидентного ВУПД), создать другой аппаратный загрузчик.

Заслуживает внимания то обстоятельство, что ЭВМ «Электроника-60» может работать в данной системе под управлением RT-11 при минимальном наборе устройств (центральный процессор, ОЗУ 16 К слов, интерфейс терминала и аппаратный загрузчик).

Описанный комплекс в составе пяти рабочих мест экспериментатора эксплуатируется уже более года и показал высокую эффективность, гибкость и удобство в работе.

В заключение авторы выражают благодарность сотрудникам лаборатории терминальных систем НГУ А. Б. Большевскому и К. А. Гилеву за весьма плодотворные дискуссии и предоставление некоторых программных средств RT-11 и программы аппаратного загрузчика, без чего выполнить данную работу было бы невозможно.

ЛИТЕРАТУРА

1. RT-11. Software support manual.—Waynard, Massachusetts, DEC, 1980.
2. Бажан А. И. и др. Многомашинный комплекс автоматизации физического эксперимента (технические и системные средства).— В кн.: Тр. Второго всесоюз. семинара по автоматизации научных исследований в ядерной физике и смежных областях. Новосибирск: ИЯФ СО АН СССР, 1982.

Поступило в редакцию 12 января 1984 г.

УДК 681.3.06

Э. А. ТАЛНЫКИН

(Новосибирск)

БАЗОВОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ РАСТРОВОГО ДИСПЛЕЯ

В работе* описан графический дисплей, разработанный в Институте автоматизации и электрометрии СО АН СССР. Цель настоящей статьи — дать более подробное представление о его программном обеспечении, точнее, речь будет идти о программных средствах графической станции на базе микро-ЭВМ «Электроника-60». Микро-ЭВМ управляет графическим контроллером и может иметь линию связи с другой, более оснащенной ЭВМ, называемой далее ведущей. Рассматриваются только варианты, когда обе ЭВМ имеют совместимую архитектуру.

Организация программного обеспечения. Основу для организации программного обеспечения составляет модуль базовых графических примитивов (БГП), который представляет графический дисплей по отношению к другим программным компонентам. Этот модуль всегда размещается в микро-ЭВМ. Если прикладная программа (ПП), которой требуются графические возможности, также размещается в микро-ЭВМ, то все проблемы межмодульного взаимодействия решаются на уровне редактирования внешних связей (рисунк, а).

* Ковалев А. М., Талныкин Э. А. Графический дисплей растрового типа для систем двухкоординатного проектирования.— Автотметрия, 1984, № 4.

в оформлении запроса в виде пакета и передаче его через модуль связи (МС) в микро-ЭВМ. Модуль связи в микро-ЭВМ после достоверного приема пакета разворачивает его и организует запрос к реальному модулю базовых графических примитивов.

Базовые графические примитивы. Графические примитивы приводятся в форме функциональных запросов с возможными параметрами. Не будем структурировать отдельные примитивы по группам, хотя их можно подразделить на средства управления графической станцией, построения элементов изображения, управления режимами отображения и другие.

DINIT

Привести в рабочее состояние аппаратуру дисплея. Данный запрос достаточно выдать один раз при работе резидентной прикладной программы.

TMODE /GMODE /OFFMODE

Установить соответственно текстовый, графический или автономный режимы работы графической станции.

FAST /SLOW

Установить ускоренный протокол обмена по линиям связи либо менее быстрый, но более надежный.

FIELD (n, m)

Установить габариты виртуального поля так, чтобы строка содержала, как минимум, n элементов изображения. Установить расслоение изображения на m карт.

MAP (m)

Установить номер текущей карты равным m .

PENSET (p)

Установить значение «пера». Рисование на дисплее сводится к заполнению памяти единицами либо нулями. Значение $p \neq 0$ задает режим рисования единицами, а $p = 0$ — нулями.

PENFF

Инvertировать состояние пера.

CLMEM

Стереть изображение. Если значение пера установлено в единицу, то память заполняется нулями, а иначе — единицами.

CLMAP

Стереть изображение с текущей карты памяти.

POINT (x, y)

Нарисовать точку в координатах x, y текущей карты памяти. Это означает, что один бит памяти устанавливается в единицу или нуль в зависимости от состояния пера.

LINE (x_1, y_1, x_2, y_2)

Соединить две точки текущей карты изображения отрезком прямой линии.

BOX (x_1, y_1, x_2, y_2)

Нарисовать на текущей карте прямоугольник со сторонами, параллельными осям, и двумя заданными точками в качестве противоположных вершин.

CIRCLE (x, y, d)

Нарисовать на текущей карте окружность с центром в точке x, y и диаметром d .

CBODY (x, y, d)

Нарисовать на текущей карте круг с центром в точке x, y и диаметром d .

CONNCT (x_1, y_1, x_2, y_2, d)

Соединить две точки текущей карты изображения отрезком прямой линии толщиной d .

GRID (x, y, Cx, Cy, s)

Нарисовать на текущей карте сетку, размещая в каждом узле сетки точку. Здесь x, y — координаты левого нижнего узла сетки, Cx — число столбцов, Cy — число строк и s — шаг сетки.

GETFUN (ch, x, y)

Вести графическую информацию. После исполнения запроса переменной ch присваивается код функциональной кнопки, а переменным x, y — координаты локатора на поле изображения. Ввод информации возможен только в графическом режиме. Данный запрос переводит прикладную программу в состояние ожидания ввода от оператора.

- LOCXP /LOCXM /LOCYP /LOCYM**
Переместить локатор вправо (влево, вверх, вниз) на один элемент изображения.
- LOCSET (*x, y*)**
Установить локатор в позицию с координатами *x, y*.
- LOCOFF /LOCON**
Выключить (включить) локатор. Локатор может не отображаться на экране. Например, допускается включение локатора только перед запросом на ввод графической информации.
- SCRXP /SCRXM /SCRYP /SCRYM**
Переместить экран вправо (влево, вверх, вниз) на минимально возможное смещение. Квант перемещения экрана по оси *Y* составляет один элемент изображения, а по *X* — шестнадцать.
- SCRSET (*x, y*)**
Установить экран так, чтобы точка виртуального поля с координатами *x, y* была в верхнем левом углу экрана.
- ZOOM (*p*)**
Установить экранный масштаб равным *p*.
- ZOOMP /ZOOMM**
Увеличить (уменьшить) экранный масштаб, если это возможно.
- PRIOR**
Изменить приоритет карт памяти на изображении. Последовательные исполнения данного запроса вызывают циклическую смену приоритетов.
- MIXER**
Установить режим смеси цветов на пересечении изображений с разных карт памяти.
- DISABLE (*m*) /ENABLE (*m*)**
Отключить (включить) отображение карты памяти *m*. Изображение отключенной карты отсутствует на экране, но не стирается из памяти, и, более того, на невидимой карте можно рисовать.
- COLOR**
Произвести смену типов раскраски карт. Последовательные исполнения данного запроса вызывают циклическую смену раскраски.
- SETPRI (*p*)**
Установить значение регистра приоритета равным *p*. Все запросы по заданию приоритетов в смеси цветов, а также включению, отключению карт и смены типов раскраски сводятся к установке значения регистра приоритета.
- MIRROR /NORMAL**
Установить зеркальную (нормальную) систему координат, в которой начало находится в верхнем левом углу, ось *x* направлена вправо, а ось *y* — вниз.

Модули связи. Модуль связи ведущей ЭВМ получает от модуля виртуальных графических примитивов пакет, и его задача состоит в достоверной передаче пакета получателю. Линия связи для программы является собственным терминалом, и протокол связи реализуется на уровне запросов чтения/печати строки.

Основная функция модуля связи микро-ЭВМ — осуществление прозрачной коммутации терминала. Символы, вводимые с клавиатуры, передаются в линию (к ведущей ЭВМ), а поступающие с линии печатаются на экране алфавитно-цифрового дисплея. Если в потоке символов из ведущей ЭВМ обнаруживается правильно оформленный пакет, то последний обрабатывается через модуль графических примитивов. Такой режим называется текстовым, поскольку со стороны терминала поступает только текстовая информация.

В графическом режиме среди символов, вводимых с клавиатуры, выделяются символы локального управления, которые обрабатываются через базовые примитивы. Ввод любого другого символа вызывает передачу в ведущую ЭВМ пакета, содержащего код символа и текущие координаты локатора на поле изображения. Таким образом, со стороны терминала поступает графическая информация.

Автономный режим отличается от графического тем, что не производится никаких передач по линии в сторону ведущей ЭВМ. Все символы, вводимые с клавиатуры и не являющиеся символами локального управления, просто отражаются на экране алфавитно-цифрового дисплея.

Локальное управление. Символы локального управления действуют как функциональные кнопки и инициируют выполнение следующих функций: перемещение локатора по экрану, перемещение экрана по изображению, управление приоритетами и раскраской, экранное масштабирование.

Перемещения (локатора или экрана) управляются позиционно или по скорости. Позиционное управление состоит в перемещении объекта на один дискрет в ответ на нажатие кнопки (стрелки четырех направлений). При управлении по скорости нажатие кнопки со стрелкой определенного направления вызывает дискретное приращение скорости, которая может быть, в частности, нулевой или отрицательной (движение в обратную сторону).

Заключение. Определим некоторые направления развития рассмотренной базовой системы. Адаптацией модуля виртуальных примитивов можно обеспечить интер-

фейс к какому-либо графическому пакету или стандарту. Изменением протокольной части в микро-ЭВМ возможна эмуляция определенного типа графического терминала с целью использования имеющегося для него прикладного программного обеспечения. Достаточно легко адаптировать базовый графический интерфейс к различным языкам высокого уровня.

Поступило в редакцию 10 мая 1984 г.

УДК 681.3.06

В. Х. ЗИНГЕР
(Новосибирск)

ГЕНЕРАЦИЯ ТЕНЕЙ В СИСТЕМЕ СИНТЕЗА ТРЕХМЕРНЫХ ВИЗУАЛЬНЫХ СЦЕН

1. При создании систем синтеза визуальных сцен одной из принципиальных проблем является повышение реальности и наглядности изображений. Основной эффект достигается удалением невидимых поверхностей. Однако привнесение теней в синтезируемую сцену позволяет резко усилить иллюзию глубины изображения. Тени обеспечивают значительную информацию о положении наблюдателя и более детальное понимание окружающей обстановки.

Предлагаемый в данной статье метод генерации теней в трехмерном пространстве использует в качестве исходной информации базу данных, состоящую из многоугольников, задаваемых упорядоченным набором координат вершин и нормалью. Результатом работы алгоритма генерации является база данных, дополненная многоугольниками, описывающими тени. Метод основан на клипировании трехмерного пространства пирамидами, опирающимися на отбрасывающие тени грани.

Необходимо отметить, что генератор теней — предпроцессор в системе отображения реального времени с программно-аппаратной схемой удаления невидимых поверхностей, основанной на механизме приоритетов [1, 2]. Механизм заключается в том, что база данных, которую необходимо отображать, предварительно подвергается программной статической сортировке по приоритетам, а удаление невидимых поверхностей проводится аппаратно во время визуализации на основе поступления объектов в кадр в порядке приоритета.

Следует отметить, что для экономии вычислительных ресурсов, требуемых для отображения визуальных сцен, расчет теней и добавление граней, описывающих тени, к исходной базе данных необходимо производить заранее. Так как тень является многоугольником, совпадающим с некоторым многоугольником (или его частью), принадлежащим исходной базе данных и отличающимся только цветом, то генератор теней не изменяет организации исходной базы и не влияет на последующие алгоритмы обработки. В том случае когда тени генерируются над базой данных, отсортированной по приоритетам, грани, описывающие тени, не нарушают приоритетного порядка, т. е. дополнительной сортировки не требуется.

Применение генератора теней может существенно увеличить количество граней в базе данных. Однако во время построения изображения затраты на обработку дополнительных граней незначительны и никак не могут сравниться с затратами, необходимыми на расчет теней.

2. Последующее изложение относится к ситуации, когда сцена освещается точечным источником, находящимся от нее на конечном или бесконечном (параллельный пучок) расстоянии, причем положение источника света предполагается фиксированным и неизменным. Для случая протянутого источника применимы соображения, аналогичные приводимым ниже. Однако трудоемкость вычислений, связанная с необходимостью расчета полутеней, заметно возрастает.

Задача генерации теней тесно связана с задачей удаления невидимых поверхностей в трехмерном пространстве. В самом деле, если положение наблюдателя зафиксировать в точке, где расположен источник света, и задать ориентацию, совпадающую с направлением пучка света, то будут видны только освещенные грани, а все невидимые из этой точки грани будут лежать в тени. Тени становятся видимыми по мере удаления наблюдателя от точки, в которой расположен источник света. Таким образом, для того чтобы определить все тени, необходимо установить точку наблюдения в источнике света и выбрать ориентацию наблюдателя, совпадающую с направлением пучка света. Затем для этого фиксированного положения и ориентации наблюдателя требуется исключить из рассмотрения «задние» грани (т. е. грани, направленные нормали к которым совпадают с направлением луча света); определить все невидимые грани. Далее следует удалить невидимые из этой точки поверхности окрасить в цвета, соответствующие их окраске в тени, и добавить их в исходную базу данных.

3. Излагаемый ниже способ генерации теней основан на анализе пирамид видимости, построенных из плоскостей, проходящих через источник света и стороны многоугольников, принадлежащих базе данных [3, 4].