

3. Glassner A. S. Space subdivision for fast ray-tracing // IEEE Comp. Graph. and Appl.— 1984.— 4, N 10.— P. 15.  
4. Kaplan M. The uses of spatial coherence in ray-tracing // ACM SIGGRAPH'85, Course Notes.— 1985.— 11, p. 22.

Поступила в редакцию 17 января 1990 г.

УДК 681.3.068

Р. Н. ВИЛЬДАНОВ, А. М. МАЦОКИН  
(Новосибирск)

### ГРАФИЧЕСКИЙ ПАКЕТ ДЛЯ ДВК-3

Использование средств машинной графики на ПЭВМ в значительной степени повышает эффективность труда исследователей. В нашей стране на отечественных мини-ЭВМ и ЕС ЭВМ широко используется графическая система СМОГ [1], предоставляющая необходимый набор средств по формированию и выводу графиков, карт изолиний, аксонометрических проекций поверхностей, векторных полей, чертежно-конструкторской документации.

В настоящей работе предлагается вариант этой системы для ДВК-3 [2]. Напомним, что в состав ДВК-3 входят 2 МГМД и видеомонитор «Электроника МС 6105», обеспечивающий вывод графической информации на экран и ее совмещение с имеющейся алфавитно-цифровой информацией, а также отдельный вывод графической и алфавитно-цифровой информации. Размер графического изображения — 286 строк по вертикали и 400 точек в каждой строке (50 байт). Графическая информация хранится в ОЗУ контроллера графического дисплея (КГД), имеющем информационную емкость 16 Кбайт. Логической единице в памяти соответствует светящаяся точка на экране монитора.

При адаптации графической системы СМОГ на ДВК-3 к операционной системе РАФОС основная задача — математическое описание логического графического экрана и обеспечение вывода вектора. Естественно, что эта задача является простой и при ее решении на графическом экране вводим декартову систему координат, начало которой соответствует левому нижнему углу экрана, размер по горизонтали и вертикали экрана — 399 и 285, координаты  $(i, j)$  каждой точки экрана являются целочисленными. Поскольку ОЗУ КГД представляет собой линейный байтовый массив, точке  $(i, j)$  при ограничениях  $0 \leq i \leq 399$  и  $0 \leq j \leq 285$  соответствует  $n$ -й байт  $m$ -го бита, где

$$n = (285 - j)50 + k;$$

$$m = i - 8k$$

( $k$  — целая часть от  $i/8$ ).

Так как ни в аппаратном, ни в программном обеспечении ДВК-3 генератор векторов не реализован, целесообразно использовать один из эффективных алгоритмов — алгоритм Брезенхэйма. Отметим, что алгоритм Брезенхэйма реализован таким образом, что генерация точек вектора происходит упорядоченно по горизонтали слева направо. Такая реализация позволяет с помощью повторного вывода вектора стереть его без появления лишних точек («мусора») при произвольном порядке задания концов отрезка.

Прежде чем перенести пакет СМОГ на ДВК-3, было разработано более простое, но, с нашей точки зрения, минимально необходимое графическое ядро (ГЯ), на основе которого можно строить то или иное программное обеспечение по решению графических задач, в том числе и

адаптацию графического пакета СМОГ. В качестве базовых графических примитивов выбраны точка, вектор, границы прямоугольника и прямоугольник. Заметим, что в число базовых примитивов не входит символ, поскольку он является базовым примитивом пакета СМОГ.

ГЯ предоставляет средства включения и выключения символьного или графического экранов, инверсию графического изображения, стирание алфавитно-цифровой или (и) графической информации.

Вывод базовых графических примитивов осуществляется по следующему правилу: каждому примитиву соответствует совокупность точек экрана, каждой из которых условно присваивается нулевое логическое значение, а вывод осуществляется под управлением целочисленного параметра  $IVID$  ( $0 \leq IVID \leq 5$ ). При  $IVID = 0$  условное значение каждой точки присваивается соответствующей точке КГД, при  $IVID = 1$  соответствующей точке ОЗУ КГД присваивается отрицание значения каждой точки, при  $IVID = 2$  значение бита ОЗУ КГД логически складывается со значением точки, при  $IVID = 3$  значение бита ОЗУ КГД логически складывается с инвертированным значением точки, при  $IVID = 4$  под значением бита ОЗУ КГД и значением точки выполняется операция XOR («исключающее или»), а при  $IVID = 5$  операция XOR выполняется над значением бита ОЗУ КГД и инвертированным значением точки. Разумеется, в ГЯ реализована операция клипирования при выходе координат точек за пределы экрана.

Кроме того, ГЯ предоставляет возможность запоминания части изображения (прямоугольной) в отведенной программистом памяти и последующего ее вывода со сдвигом на экран. Это средство позволяет расширить набор графических примитивов, компоновать изображение из заранее подготовленных фрагментов и т. д.

К сожалению, в ДВК-3 отсутствуют аппаратные средства ввода графической информации. Поэтому в ГЯ программно реализован «локатор» в виде управляемого клавиатурой перекрестья (большого и малого), предоставляющего программе по ее запросу координаты его центра. Изображение «локатора» на экране при выводе графических примитивов или запоминании части экрана отключается автоматически с тем, чтобы не исказить построенную «картинку».

ГЯ реализован на Ассемблере в виде единого модуля и занимает менее 6 Кбайт.

Адаптация графического пакета СМОГ на ДВК-3 была реализована на основе разработанного ГЯ и обеспечивает хранение графической информации на внешнем запоминающем устройстве, вывод сформированных рисунков в любом месте экрана и в любом масштабе.

Необходимо отметить, что ДВК-3 удобен только для запуска готовых программ в виде загрузочных модулей. Трансляция программ, компоновка загрузочных модулей из-за медленной работы накопителей магнитных дисков, малой их емкости требует большого времени.

Одним из авторов была проделана работа по установке электронного диска емкостью 1 Мбайт в ДВК-3, что резко повысило производительность труда программистов при подготовке программ. Еще более можно повысить производительность при подключении ДВК-3 к более мощной ЭВМ типа СМ-4, которая имеет разнообразное периферийное оборудование (магнитные ленты, диски большой емкости, графопостроители и т. д.). Подключение ДВК-3 к СМ-4 было произведено с помощью линии связи DL—KI/SI. Программное обеспечение межмашинной связи, созданное авторами, базируется на протоколе, описанном в [3]. При запуске программ на ДВК-3, включающих каналную часть программ, ДВК-3 по отношению к СМ-4 становится обычным терминалом. Программы на СМ-4, включающие уже свою каналную часть, могут обмениваться с ДВК-3 символьной или любой двоичной информацией, т. е. СМ-4 использует графические возможности ДВК-3. Графический пакет СМОГ может быть в данном случае расположен на СМ-4, а функции ГЯ — на ДВК-3.

## СПИСОК ЛИТЕРАТУРЫ

1. Математическое обеспечение графопостроителей. I уровень: СМОГ: Инструкция по программированию/Под ред. Ю. А. Кузнецова.— Новосибирск: ВЦ СО АН СССР, 1976.
2. Вильданов Р. Н., Дебелов В. А., Упольников С. А. СМОГ-85. Графический пакет для ЕС и СМ ЭВМ // Тез. докл. Всесоюз. конф. по проблемам машинной графики и цифровой обработки изображений.— Владивосток: ИАПУ ДВНЦ АН СССР, 1985.
3. Дебелов В. А., Мацокин А. М., Чубарев А. И. Подход к разработке распределенных систем для двухмашинных комплексов // Машинная графика и ее приложения.— Новосибирск: ВЦ СО АН СССР, 1983.

Поступила в редакцию 17 января 1990 г.

УДК 681.3.082.5

И. И. ЛИСИЦЫНА, Т. И. ПЕТРУШИНА, Н. Ф. ТРУБИНА

(Одесса)

### МОДЕЛИРОВАНИЕ АРХИТЕКТУРЫ МНОГОПРОЦЕССОРНОЙ РАСТРОВОЙ ГРАФИЧЕСКОЙ СИСТЕМЫ И ОЦЕНКА ЕЕ ПРОИЗВОДИТЕЛЬНОСТИ

Создание высокопроизводительной графической системы реального времени требует обоснования и предварительного исследования предполагаемой архитектуры [1]. Авторы используют для этой цели разработанную ими программную модель архитектуры многопроцессорной растровой графической системы получения трехмерных реалистических изображений. Программная модель позволяет охарактеризовать и проанализировать связи в графической системе, а на основе собираемой статистики получить оценки производительности системы и отдельных ее компонент. Анализ получаемой статистики дает также возможность выявить «узкие» места архитектуры, что является основанием для изменения конфигурации системы и (или) модификации набора команд единичного процессора. Модульная организация программной модели обеспечивает не только гибкость и удобство ее разработки, но и возможность подбирать и испытывать различные конфигурации, соответствующие вариантам архитектуры графической системы.

Графическая система предназначена для получения изображения сцен, содержащих трехмерные объекты, представленные в системе в виде визуализируемых оболочек (в-оболочек). В-оболочка получается в результате триангуляции реального объекта и содержит массив координат точек объекта (вершин), их яркостей и координат нормали в вершине, массив треугольных граней, заданных ссылками на вершины, и некоторые дополнительные характеристики. В-оболочки объектов строятся автономными средствами и накапливаются в базе данных объектов. Программная модель обрабатывает описание сцены, состоящее из набора в-оболочек, извлекаемых из базы данных.

Программная модель имитирует обработку описания сцены следующими компонентами графической системы.

**Магистраль геометрических преобразований (МГП)**, отвечающая требованиям развитых графических стандартов (типа GKS и PHIGS) [2]. Предусматривается модельное преобразование координат вершин каждой в-оболочки из модельной системы координат, в которой задан объект, в мировую систему координат. Далее применяются *видовое преобразование* (геометрическое преобразование для приведения к норма-