

Концепция конвейерного блока отсечения синтезирующей системы

Хачумов Михаил Вячеславович, к.ф.-м.н.,с.н.с.,
Хачумов Вячеслав Михайлович, д.т.н., проф.
ФИЦ ИУ РАН, РУДН, ИПС им. А.К.Айламазяна РАН

Аннотация. Рассматриваются вопросы структурной организации блока отсечения изображений прямоугольным окном для параллельной системы графики реального времени (ПС ГРВ), синтезирующей сцены окружающего мира в составе тренажера летчиков. В основу построения модулей ПС ГРВ положены конвейерные алгоритмы с совмещением циклов обработки данных.

Введение

В настоящее время широкое распространение получают синтезирующие системы визуализации, построенные по конвейерному принципу для выполнения требований реального времени. Конвейерный режим работы накладывает определенные ограничения на разработку алгоритмов отсечения.

В ряде публикаций авторов были рассмотрены оптимизационные задачи реализации конвейерных (периодических) расписаний [1-4] применительно к алгоритмам машинной графики и решению задач управления и навигации в бортовых вычислительных комплексах беспилотных летательных аппаратов [5-7]. В рамках этих работ были решены задачи минимизации числа процессорных элементов и межпроцессорных связей, выбора структуры геометрического процессора на основе графических процессорных элементов, разработано программное обеспечение для автоматизации проектирования. В настоящей работе рассматривается концепция конвейерного устройства реального времени на примере блока отсечения прямоугольным окном.

1. Структура ПС ГРВ

Синтезирующая система содержит геометрический процессор, базу данных, блок предварительного отсечения (БПО), загрузчик ребер, интерполятор ребер, память кадра. Система представляет собой конвейер, структура которого может меняться за счет изменения числа подключаемых последовательно специализированных блоков. БПО осуществляет реализацию комплекса геометрических операций, включая тестирование на видимость граней, отсечение объектов по экранной плоскости, перевод из мировой в экранную систему координат, проецирование объектов и выполнение аффинных преобразований. База данных хранит описание граней трехмерных объектов, загрузчик ребер - производит отсечение ребер по границам экрана. Конвейерный режим работы системы с асинхронным механизмом информации между блоками накладывает определенные ограничения на организацию структур вы-

числительных блоков. Рассмотрим далее особенности структурной организации БПО.

2. Блок предварительного отсечения

БПО предназначен для выполнения операций загрузки атрибутов графического объекта, проецирования трехмерных объектов на плоскость наблюдателя с отсечением части объекта, расположенной до экранной плоскости; передачи атрибутов графических объектов без предварительной обработки соседнему звену и отсечения по прямоугольному окну. На базе БПО [8] могут быть реализованы более сложные конвейерные и параллельные структуры геометрического процессора. Проектирование структуры данного модуля сопровождалось исследованием следующих подходов:

1) последовательная организация вычислений в одном процессорном элементе, ориентированном на реализацию отсечения прямоугольным окном сцен, наблюдаемых средствами технического зрения. Часть алгоритмов его функционирования, связанных с множительно-делительными операциями, разрабатывается с применением разрядно-параллельных форм CORDIC [9];

2) организация многопроцессорной системы, включающей четыре процессорных элемента с параллельно-последовательной или конвейерной организацией.

Широкое распространение получил способ отсечения, основанный на процедуре "прогонки" всех точек через каждую отсекающую полуплоскость в отдельности [10]. Предполагается, что графические объекты представлены наборами видимых плоских граней, где в свою очередь грань есть выпуклый многоугольник, задаваемый перечислением своих вершин (рис.1). В каждый данный момент времени имеется информация только об одном ребре многоугольника, представленном соответственно координатами текущей (X_2, Y_2) и предшествующей (X_1, Y_1) вершины. Имеется рабочий регистр для хранения точки пересечения и буфер первой (она же последняя) точки усеченного многоугольника (БПТ).

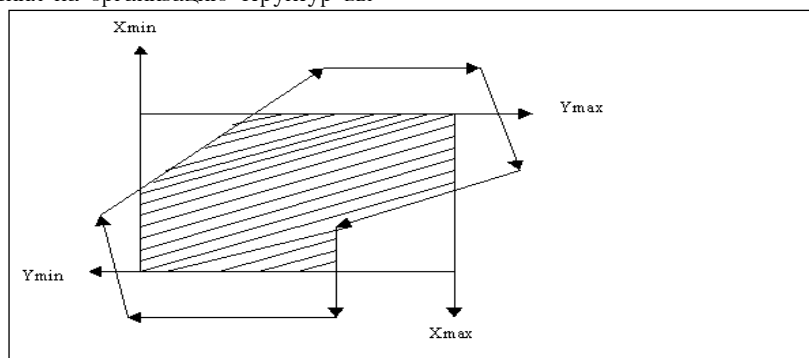


Рисунок 1. Направление границ и результат отсечения

Блок-схема алгоритма "прогонки" через отсекающую полуплоскость $X=X_{\min}$, представлена на рис.2. Алгоритм функционирует подобно цифровому автомату с памятью, текущее состояние которого описывается системой однобитовых флагов $F_1F_2F_3F_4$. Переход автомата в новое состояние инициируется проверкой расположения текущей

вершины относительно границы плоскости, причем $F_2=0$, если $X_2 < X_{\min}$, иначе $F_2=1$. Переход определяется текущим состоянием, содержащим входной информации и сопровождается, в общем случае, выполнением набора вычислительных операций с последующим изменением комбинации флагов.

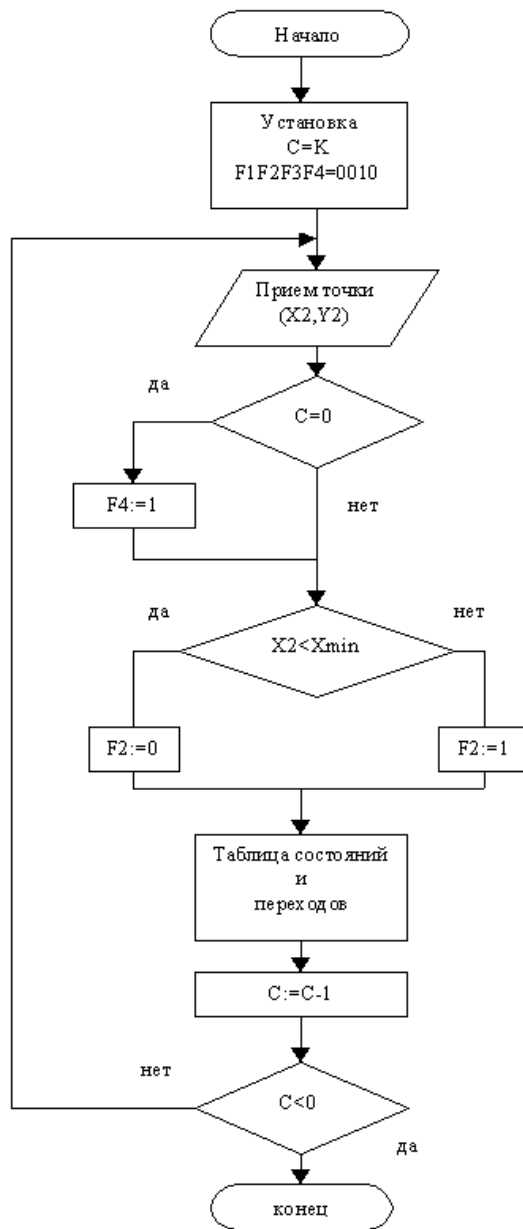


Рисунок 2. Алгоритм отсекающего по границе окна

Все состояния автомата и переходы определяются табл.1. В таблице используются следующие операции, оформленные в виде независимых модулей: P_1 - выдача текущей точки; P_2 - определение точки пересечения ребра многоугольника с плоскостью по формуле $Y = (X_{\min} - X_1) / (X_2 - X_1) (Y_2 - Y_1) + Y_1$; P_3 - запись координат текущей точки или точки пересечения в БПТ усеченного многоугольника; P_4 - выдача координат точки из БПТ; P_5 - выдача координат точки результирующего многоугольника из рабочего регистра.

Начальному состоянию автомата соответствует комбинация флагов $F_1F_2F_3F_4=0010$ и состояние счетчика числа вершин грани $C = K$. Алгоритм завершает свою работу после обработки последней вершины многоугольника, совпадающей с первой вершиной. Это соответствует установке флага $F_4 = 1$ (пришла последняя вершина многоугольника). Хранение координат первой точки усеченного многоугольника позволяет производить замыкание результирующей фигуры без потерь времени на дополнительные вычисления.

Таблица 1. Состояния и переходы

N	Состояние флагов F1 F2 F3 F4	Выполняемые операции
1	0 0 0 0	Выход из таблицы
2	0 0 0 1	P_4
3	0 0 1 0	$F_4 = 1$
4	0 0 1 1	Выход из таблицы
5	0 1 0 0	$P_2, P_5, P_1, F_i = 1$
6	0 1 0 1	P_2, P_5, P_4
7	0 1 1 0	$P_1, P_3, F_1 = 1, F_3 = 0$
8	0 1 1 1	$P_2, P_5, P_3, P_1, F_1 = 1, F_3 = 0, F_4 = 0$
9	1 0 0 0	$P_2, P_5, F_1 = 0$
10	1 0 0 1	P_2, P_5, P_4
11	1 0 1 0	Запрещенное состояние
12	1 0 1 1	Запрещенное состояние
13	1 1 0 0	P_1
14	1 1 0 1	P_4
15	1 1 1 0	Запрещенное состояние
16	1 1 1 1	Запрещенное состояние

Рассмотренные предложения по отсечению отрезков положены в основу алгоритма отсечения многоугольника, в котором проверка осуществляется одновременно по всем границам окна (рис.3). Здесь: P_6 – кодирование расположения предшествующей и текущей точек многоугольника относительно каждой отсекающей полуплоскости по системе «вне-внутри» с помощью наборов однобитовых флагов соответственно; P_7 – определение точек пересечения ребра многоугольника с полуплоскостями и их кодирование относительно границ окна по системе «внутри», «выше», «ниже». Для этой цели используются флаги N_{11} , N_{12} , IP_i , где i – номер полуплоскости (границы окна), причем перебор границ производится в жестко заданном порядке, как это показано на рис.4; P_8 – определение расстояния S от текущей точки до каждой точки пересечения с границами; P_9 – упорядочение точек пересечения по величинам вычисленных расстояний. Данная процедура необходима для определения истинного порядка расположения точек на ориентированном ребре, поскольку, как это видно из рисунка, установленный порядок перебора полуплоскостей может привести к нарушению очередности вычисления точек пересечения; P_{10} – принятие возможного решения, заключающегося в выполнении одной из нескольких операций типа: выдача координат точки усеченного объекта следующему звену конвейера, запись в буфер первой точки (БПТ) усеченного многоугольника или выдача его содержимого, занесение вершин окна в буфер вершин окна (БВО) и т.д.

Точка пересечения, оказавшаяся на внутренней границе окна, передается следующему звену обработки. Пересечение ребра с полуплоскостью вне области окна приводит к установке в единичное состояние флага N_{i1} или N_{i2} . Процедурой производится проверка на наличие совпадений флагов, примыкающих к соответствующим вершинам

окна. Так для первой полуплоскости проверяются флаги N_{11} , N_{22} , N_{12} и N_{42} . Одновременная установка соответствующих флагов в единичное состояние свидетельствует о возможном охвате данной вершины окна. При этом содержимое счетчика охвата k -ой вершины C_k увеличивается на единицу. Если содержимое счетчика становится равным двум, то производится исключение соответствующей вершины из БВО и сброс C_k , что свидетельствует об отсутствии охвата. Выгрузка вершин из буфера и включение их в описание усеченного многоугольника осуществляется перед каждой выдачей точки пересечения с внутренней границей окна. Выгрузка производится по принципу «первым пришел – последним вышел»; P_{11} – выдача текущей точки, лежащей внутри окна; P_{12} – запись первой точки БВО в БПТ, если они не пусты.

2. Программное обеспечение БПО

В структуру программного обеспечения БПО входят программы, реализующие выполнение соответствующих команд. Загрузка атрибутов объекта заключается в обращении к базе данных графического спецпроцессора и выборе из нее количества граней, цвета обрабатываемой грани, количества вершин в грани и т.д., необходимых для различных этапов визуализации объекта. Этот этап характеризуется большим количеством обращений к памяти и выполнением простых арифметико-логических операций. Программа формирования кода и анализа объекта формирует код для передачи в загрузчик ребер и по значению класса объекта определяет способ его обработки. Программа обработки объектов 1-го класса выбирает координаты точек из базы данных и передает их в загрузчик ребер. Отсечение по экранной плоскости (объекты 3-го класса) связано с выполнением таких процедур как нахождение центральных проекций точек объекта, лежащих за экранной плоскостью, вычисление точек пересечения ребер с экранной плоскостью. Отсечение по окну (объекты 2-го

класса) заключается в отсечении граней объекта окном прямоугольной формы и, в общей постановке, сводится к задаче пересечения плоских многоугольников. Реализация требует выполнения операций умножения и деления, а также проверки расположения ребер многоугольника от-

носительно экранной плоскости и окна. Несмотря на кажущуюся простоту указанных операций, требуется, тем не менее, значительная работа по созданию алгоритмов, способных успешно функционировать в ПС ГРВ.

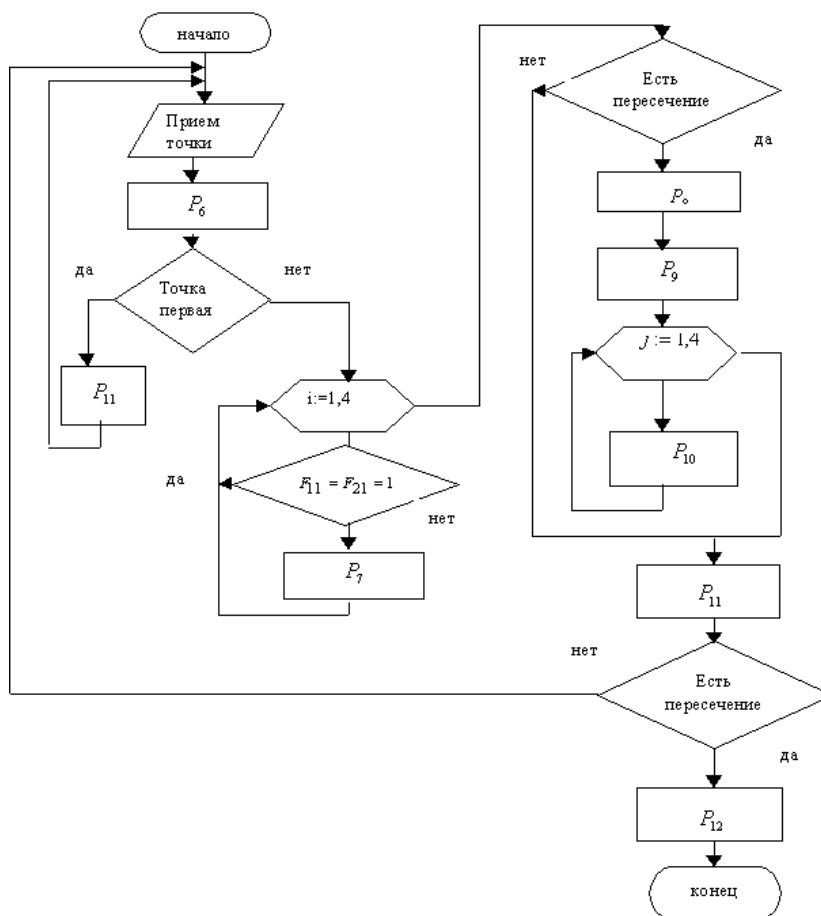


Рисунок 3. Алгоритм отсечения многоугольника

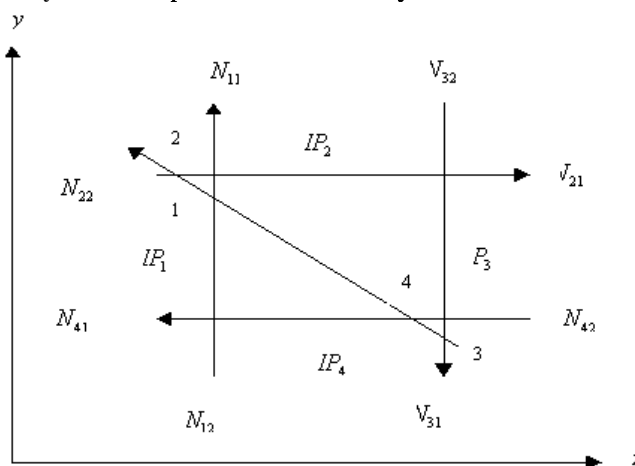


Рисунок 4. Порядок перебора границ

Заключение

В рамках концепция конвейерного геометрического процессора выполнено исследование БПО. Специализация достигается настройкой процессора на решение конкретной задачи путем загрузки в его память соответствующей управляющей микропрограммы. Для этой цели блоки ПЗУ заменяются блоками оперативной памяти, в которые производится загрузка микропрограмм в соответствии с

выполняемой функцией. Такое решение позволяет относительно просто наращивать функциональные возможности для новых приложений и обеспечивает удобство работы пользователей.

Работа выполнена при финансовой поддержке проектов РФФИ № 16-29-12839 офи_м, № 18-07-00025 а, № 16-07-00096 а.

Литература:

1. Хачумов В.М. Оптимизация периодической обработки информации в специализированных устройствах. Часть 1. // Информационные технологии и вычислительные системы. 2017. № 1. С. 62...76.
2. Хачумов В.М. Оптимизация периодической обработки информации в специализированных устройствах. Часть 2. // Информационные технологии и вычислительные системы. 2017. № 4. С. 29...42.
3. Хачумов В.М., Хачумов М.В. Технология автоматического синтеза структур геометрических процессоров // Приборы и системы. Управление, контроль, диагностика. 2017. № 11. С. 1...8.
4. Кочина Л.В., Хачумов М.В., Шустова М.В. Синтез структур специализированных вычислительных устройств периодической обработки информации // Труды XVI международной конференции «Системы проектирования, технологической подготовки производства и управления этапами жизненного цикла промышленного продукта (CAD/CAM/PDM-2016)». 17–19 октября 2016 г. Москва. С. 90...93.
5. Хачумов В.М. Аппаратные платформы и математическое обеспечение для бортовых систем управления малых летательных аппаратов // Научное обозрение. 2016. № 7. С. 114...122.
6. Хачумов М.В. Реализация алгоритмов навигации и управления в бортовых вычислительных комплексах летательных аппаратов // Программные системы: теория и приложения. 2016. № 2. С. 35...59.
7. Хачумов В.М. Вычисление математических функций на основе разрядно-параллельных схем. // Информационные технологии и вычислительные системы. 2016. № 3. С. 26...44.
8. Фомин М.Б., Хачумов М.В. Основы компьютерной графики и обработки изображений. Учебное пособие. – М.: РУДН, 2018 (в печати)
9. Хачумов М.В. Разрядно-параллельные вычислительные схемы для бортовых систем управления летательных аппаратов // Авиакосмическое приборостроение. 2017. № 10. С. 12...17.
10. Фоли Дж, вэн Дэм А. Основы интерактивной машинной графики: В 2-х книгах. - М.: Мир, 1985.