

Алгоритм PBSP, как и BSP, оперирует произвольными данными. Но есть небольшие отличия. Алгоритм BSP-деревьев является общим алгоритмом удаления невидимых поверхностей. А алгоритм PBSP является частью алгоритмов теории твердого тела, т.е. исходит из того предположения, что вся сцена высекает собой часть пространства, а не как в BSP – заключает в себе часть пространства. Эти принципы очень близки по восприятию, но существенно влияют на процесс обработки поступающей графической информации. Как правило, на практике гораздо удобнее пользоваться концепцией твердого тела при построения сцен для PBSP-деревьев.

Таким образом, алгоритм PBSP является преемником алгоритма BSP, лишенным практически всех его недостатков – та же элегантность представления графической информации и отсутствие необходимости рассчитывать PVS - данные.

Литература:

1. <http://www.sgi.com>,
2. <http://www.cs.unc.edu/~geom/SSV>,
3. <ftp://ftp.sgi.com/other/bspfaq/>,
4. <http://www.flipcode.com>.

РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ ОБЕСПЕЧЕНИЯ ЖИВУЧЕСТИ ОДНОРОДНЫХ ПРОГРАММИРУЕМЫХ ДРЕВОВИДНЫХ СТРУКТУР НА СБИС ПЛМ

М.М. Высоцкий

(БГУ, г. Минск)

В последнее время, разрабатывая новые системы, стали уделять повышенное внимание обеспечению таких свойств, как надежность, отказоустойчивость, живучесть.

Программируемые логические матрицы (ПЛМ) – это один хорошо известный класс структурированных логических схем, которые нашли свое широкое использование в системах СБИС. Задача обеспечения живучести СБИС ПЛМ состоит в том, чтобы в любой момент времени в СБИС ПЛМ существовала подсистема аппаратурно-программных средств (функциональная подсистема) для выполнения цели функционирования. Создание соответствующей

щей подсистемы при нарушении условий функционирования обеспечивается средствами самой СБИС ПЛМ.

Структура, построенная на СБИС ПЛМ, должна нормально реагировать на такие типы неисправностей, как:

- константные;
- неисправности пересечения линий;
- короткие замыкания;

Восстановление функционирования после возникновения неисправности — одна из наиболее трудно реализуемых функций вычислительной системы, так как при эксплуатации могут возникнуть непредвиденные ситуации, а в существующих системах ограничена способность программных и аппаратных средств к обнаружению, диагностике, локализации неисправностей, к созданию работоспособных конфигураций.

Восстановление определяется как совокупность действий, которые осуществляются системой по получению сигнала о неисправности (при наличии системы восстановления). Так, например, восстановление — это реконфигурация структуры системы при элементарных отказах ЭВМ или процессоров с целью перераспределения решаемых системой задач между работоспособными модулями.

Различают восстановление после трех видов неисправностей (ошибок): физических; ошибок программного обеспечения; при управлении вычислительным процессом. Соответственно выделяют три принципиально различных уровня восстановления: аппаратный (физический), программный и уровень управления. Имеющийся опыт проектирования позволяет сформулировать ряд общих положений /1/, лежащих в основе построения систем, обладающих свойством живучести:

- 1) выбор базовой архитектуры, допускающей и облегчающей применение средств повышения живучести;
- 2) последовательное применение принципа параллельности (параллелизм алгоритмов, программ, устройств, данных, маршрутов передачи информации);
- 3) использование и разумное сочетание всех видов избыточности (временной, структурной, энергетической, информационной);
- 4) многофункциональность программного и аппаратного обеспечения;
- 5) автономность и модульность построения устройств, блоков, подсистем;
- 6) частичная децентрализация процессов управления;

7) наличие и обеспечение работы процедур контроля, диагностики и восстановления;

8) введение специальных механизмов и подсистем обеспечения живучести (реконфигурации, перераспределения ресурсов, постепенной функциональной деградации).

Как правило, живучесть системы достигается при использовании всех или части перечисленных принципов, реализованных в виде средств аппаратно-программной поддержки процессов в однородных программируемых древовидных структурах [2].

Для обеспечения живучести однородных древовидных структур на СБИС ПЛМ предлагаются следующие алгоритмы:

- Алгоритм оценки параметров;
- Алгоритм реконфигурации;
- Алгоритм восстановления;

В данной работе используются все три алгоритма, реализованные на языке Delphi 4.0. Программный продукт, представляет собой проект, состоящий из нескольких модулей. Каждый модуль имеет свой набор функций и процедур и предназначен для выполнения конкретных задач.

При помощи программного продукта, реализующего алгоритмы оценки параметров, реконфигурации и восстановления, приведенные в данной работе, были проведены некоторые эксперименты с двумя видами древовидной структуры (бинарное дерево с одной резервной вершиной на каждом уровне и бинарное дерево с дополнительным резервом на каждом уровне), причем каждая из них тестировалась по несколько раз с различными параметрами живучести узлов, связей, количеством резервных узлов и резервных связей.

Анализы проводились на древовидных структурах с 4 уровнями. Созданный удобный и простой интерфейс позволяет легко создавать и редактировать любую древовидную структуру. Быстро произвести анализ и расчёт живучести, удобно и доступно отобразить расчётные характеристики, как в цифровом, так и графическом формате. Все результаты представлены в виде таблиц и графиков, что позволяет более наглядно изучить полученные результаты исследования.

Литература.

1. IEEE Trans. Comput. Отказоустойчивость в архитектуре двоичного дерева, 1984, V. 33, №6.
2. IEEE Trans. Comput. Перестраиваемые программируемые логические матрицы с высоким выходом годных схем, 1990, V. 39, №4, p. 480-489.