

А. М. ВЛАСОВ, В. А. ДЫБОЙ, В. Е. МЕЖОВ,
В. В. ПЛОТНИКОВ, В. Н. ХАРИН, Ю. А. ЧЕВЫЧЕЛОВ

(Воронеж)

ИНТЕРАКТИВНАЯ СИСТЕМА ЛОГИЧЕСКОГО МОДЕЛИРОВАНИЯ ЦИФРОВЫХ СХЕМ

В настоящее время унифицированные интерактивные графические системы (ИГС) типа «Кулон» на основе мини- и микроЭВМ типа «Электроника» находят широкое применение в интегрированных САПР для решения в диалоговом режиме задач по обработке текстовой и графической информации о схемах и документированию результатов логического моделирования. В то же время собственно процесс логического моделирования в таких системах выполняется в пакетном режиме, при котором невозможно проводить оперативный анализ результатов расчета и вносить необходимые коррективы в ходе вычислительного процесса, что значительно снижает эффективность использования САПР, особенно при логическом моделировании сложных цифровых блоков. Кроме того, при разработке таких блоков образуется разрыв между этапами их проектирования, построения тестов и испытания. Используемые разработчиками инженерные приемы проектирования ориентированы в основном на обеспечение выполнения заданных функций и достижения заданных параметров блока. В то же время сложный проект зачастую является «необозримым» для разработчика с точки зрения возможности и трудоемкости проверки его работоспособности.

В статье рассмотрен прикладной пакет программ логического анализа цифровых блоков ПРАЦИС-2, предназначенный для использования на всех унифицированных ИГС типа «Кулон» и вычислительных комплексах на основе мини- и микроЭВМ «Электроника» с емкостью ОЗУ не менее 64 Кслов. Наряду со средствами логического моделирования, пакет включает также блок экспресс-анализа цифровых схем на управляемость, тестируемость, наблюдаемость.

Разработанный пакет программ позволяет проводить логический анализ цифровых блоков большой размерности как на вентиляльном, так и на функциональном уровнях. Он реализован на основе следующих программных модулей: обработки и документирования информации о схеме; экспресс-анализа схем на управляемость, наблюдаемость, тестируемость; фрагментирования; системной библиотеки; моделиатора; логического анализатора схем с одиночными ошибками.

Лингвистические средства и блок обработки и документирования достаточно подробно рассмотрены в работе [1].

Для логического анализа используется событийный интерактивный алгоритм, который обеспечивает высокое быстродействие и достаточную точность результатов моделирования цифровых блоков большой размерности, простоту учета и обработку верменных состояний.

Техника событийного моделирования, достаточно отработанная в программах для больших ЭВМ [3], была взята за основу при реализации рассматриваемого пакета. При этом проводилась оптимизация организации процедуры планирования событий с целью увеличения быстродействия и эффективного использования емкости ОЗУ.

На рис. 1 изображена файловая структура комплекса ПРАЦИС-2. Из рисунка видно, что основные программные модули комплекса — программы ВВОД, МОД, ТЕСТ и УПРАВ, назначение которых станет ясно из дальнейшего изложения.

Программа ВВОД. Программа ВВОД (ее блок-схема приведена на рис. 2) является препроцессором комплекса ПРАЦИС-2, который пере-

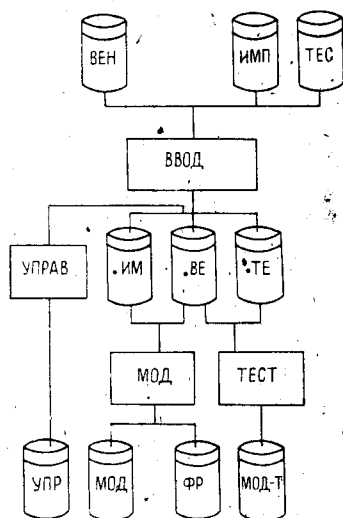


Рис. 1

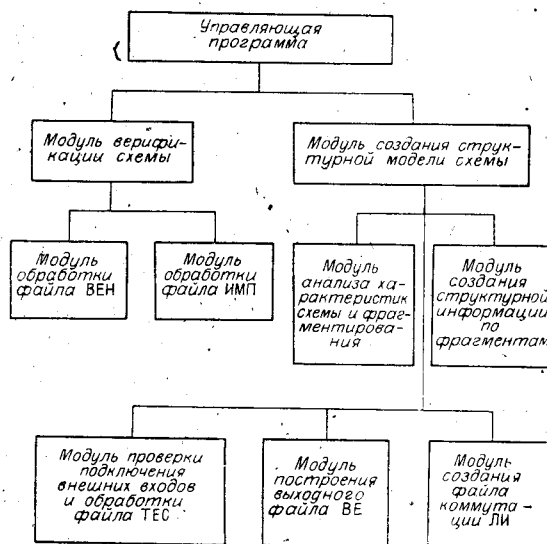


Рис. 2

водит исходную (буквенно-цифровую) информацию в цифровую определенных форматов. Все многообразие исходной информации представлено в файлах форматов ВЕН и ИМП (или ТЕС). Файл формата ВЕН служит для описания типов элементов, их характеристик и связей между элементами; файл формата ИМП — для задания входного воздействия как функции времени; файл формата ТЕС — для векторного задания входных последовательностей и ожидаемой ответной реакции. Основные функции программы ВВОД — верификация программы и создание структурной модели схемы.

Верификация информации необходима, поскольку описание логической схемы может содержать ошибки, возникающие на различных стадиях подготовки, кодирования и ввода исходных данных. Своевременное обнаружение этих ошибок позволяет сократить вычислительные затраты и цикл проектирования. Программа осуществляет проверку синтаксиса и семантики описания логической схемы и обеспечивает формирование рабочих массивов (файлы .ВЕ, .ИМ и .ТЕ) в символьно-цифровом и цифровом виде, в которых информация систематизирована по спискам и однозначно представляет структуру схемы. Заметим, что файлы ВЕН, .ИМП и .ТЕ создаются пользователем в системном редакторе и при наличии сообщения об ошибке они корректируются в процессе создания этих файлов и на этапе выполнения рабочих массивов, из которых формируется структурная модель схемы в соответствии с внутренним форматом. При этом нужно выделить две задачи: перевод символьной информации в цифровую и фрагментирование схемы. Принципиальное значение имеет выбор метода поиска информации в файлах, который в значительной степени определяет быстродействие программы (особенно с увеличением числа компонентов в схеме). Для ускорения процедуры поиска соответствующих идентификаторов в упорядоченном массиве (упорядочение проводилось по группе разрядов) был применен дихотомический алгоритм [4] (бинарный поиск). Число операций, необходимых для реализации такого поиска, пропорционально $cN \log N$, где N — число линий схемы; c — константа.

Необходимо более подробно остановиться на вопросе фрагментирования логических схем, так как с ростом объема и сложности цифровых схем возникают объективные трудности реализации основных параметров программы логического моделирования, вызванные стремлением обеспечить приемлемое время ее работы в условиях ограниченных размеров оперативной памяти (ОП). Размещение большей части рабочих

массивов на внешних носителях, таких как магнитные диски, в значительной степени уменьшает потребность в ОП, но общее время моделирования катастрофически возрастает. Использование специальных директив управления ОП дает выигрыш как в пространстве ОП, так и во времени (сокращает вычислительные затраты), но существенно ограничивает класс применяемых ЭВМ. Таким образом, эти методы не удовлетворяют требованиям, предъявляемым к программному обеспечению логического моделирования: быстродействию и унифицированности.

В программе реализован принцип фрагментирования схемы, позволяющий анализировать схемы большой размерности при постоянном объеме ОП, который заключается в следующем (блок-схема алгоритма приведена на рис. 3): схема ранжируется с использованием модификации алгоритма [5], элементам присваивается определенный ранг и выделяются контуры обратной связи. Особенностью алгоритма является то, что элементы схемы, не входящие в один и тот же контур, не могут иметь одинаковый ранг, а всем элементам одного контура присваивается один и тот же ранг, что позволяет объединить их в некоторый «псевдоэлемент» и исходную схему, содержащую «псевдоэлементы» и элементы, не входящие ни в один контур обратной связи, представить на комбинационном уровне. Таким образом, не допускается нахождение элементов одного контура в разных фрагментах. Фрагменты обрабатываются один за другим на всю глубину моделирования, и нет необходимости обращаться к обработанным фрагментам, что в значительной мере сокращает накладные расходы на считывание и запись промежуточных результатов на диск. При формировании фрагмента выбирается элемент с рангом, большим, чем ранг предыдущего элемента, уже включенного во фрагмент, и оцениваются известные характеристики: общее число элементов, липий, внешних входов и выходов у фрагмента. В случае превышения одного из параметров очередной элемент не включается во фрагмент, который считается сформированным, и процедура формирования переносится на оставшиеся элементы. Процедура фрагментирования схемы завершается формированием списков коммуникаций, последователей, идентификаторов, типов для каждого фрагмента.

Завершающим этапом работы программы. ВВОД является построение выходного файла .BE, файла рабочих массивов, организация которых в значительной степени определяет скорости работы алгоритма моделирования. В разработанной структуре данных каждый логический элемент представляется совокупностью информационных единиц нескольких массивов. Массивы делятся на основные и дополнительный. К основным массивам относятся массивы типов, задержек, коммутаций, последователей и идентификаторов.

В списке типов хранятся в закодированном виде логические имена типов элементов; в списке задержек — временные характеристики типов; в списке коммутаций — выходные и входные номера линий элемента; в списке последователей каждой линии ставятся в соответствие номера элементов приемников сигнала этой линии; в списке идентификаторов — символические имена линии схемы, необходимые для выдачи выходной документации.

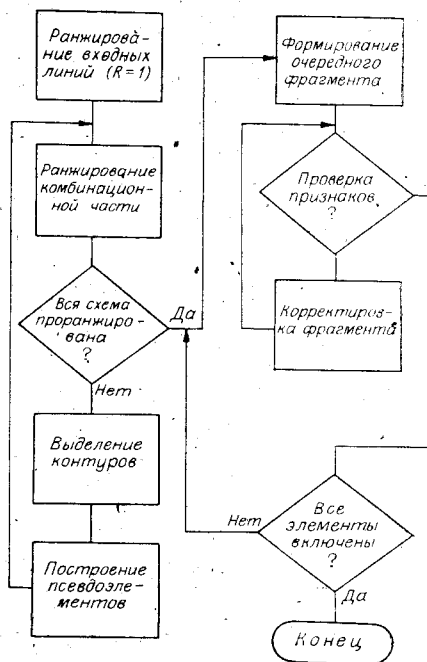


Рис. 3

Список дополнительных характеристик введен для элементов в библиотеке комплекса ПРАЦИС-2 и записи дополнительных характеристик элемента (например, для регистра сдвига — это направление сдвига, для ПЗУ — содержимое ячеек, для счетчика — число синхровходов и т. д.), а также для упрощения включения новых элементов в библиотеку ПРАЦИС-2.

Каждому элементу поставлены в соответствие два указателя. Один указатель определяет слово в списке типов (и соответственно в списке задержек), а другой — в дополнительном списке, если он используется. С помощью такой системы указателей несложно найти все параметры логического элемента схемы.

Организация массивов в структуре данных — последовательная и последовательная с указателями. Принятый способ организации данных (по сравнению с использованием фиксированных массивов) позволяет не ограничивать число параметров у каждого элемента. Это хотя и потребовало определенных расходов на конструирование и программирование, но в конечном счете обеспечило сокращение памяти, времени поиска, выборки и обработки. Поскольку такая экономия имеет место при многократной обработке, общий эффект довольно значителен.

Адекватность моделирования зависит от модели цифровой схемы, моделей элементов (примитивов), сигналов и способа учета временных соотношений между сигналами. Модели цифровой схемы определяются совокупностью примитивов и связей между ними. При реализации программы моделирования разработана библиотека примитивов, позволяющая анализировать работу цифровых устройств широкого класса. Модель элемента схемы представляет собой совокупность соотношений, определяющих логику его работы. В [3] рассмотрены возможные методы построения моделей, которые с достаточной для практических целей точностью позволяют создать примитивы элементов малой и средней степени интеграции. Сложность элементной базы ЭВМ постоянно возрастает, и стремление сохранить точность при реализации примитивов, наряду с быстродействием моделирующих программ, приводит к использованию комбинированных структурно-функциональных методов. В частности, такой подход был применен при разработке программного примитива микропроцессора, реализованного блочной структурой, развернутой на уровне регистровых передач. Выбор такой стратегии произведен по следующим соображениям. Во-первых, представление такой сложной микросхемы, как микропроцессор, в виде функционального элемента и описание его крайне затруднительно из-за огромного числа возможных комбинаций входных сигналов; кроме того, структурная модель позволяет более полно и точно показать все многообразие процессов, происходящих внутри микросхемы, и, следовательно, более точно отобразить выходную информацию. Во-вторых, моделирование структурных единиц микропроцессора на функциональном уровне достаточно просто из-за их относительно низкой функциональной сложности и потому, что их поведение полностью описывается приведенными функциональными таблицами.

Важно отметить следующее. Хотя модель микропроцессора и реализована как структурная, фактическое моделирование связано с последовательной обработкой потока информации и его трансформацией в процессе прохождения по структурным единицам через ряд промежуточных состояний. Это значит, что процесс моделирования не требует знания внутренних коммутаций между структурными единицами, так как данные на входных линиях однозначно определяют путь вычислений и, следовательно, значения на выходных линиях. Отсутствие вычислительных затрат на анализ внутренних коммутаций в процессе моделирования позволяет, с одной стороны, существенно упростить процесс моделирования, с другой — увеличить быстродействие модели.

Одним из основных факторов, определяющих быстродействие программы, является алгоритм моделирования. В силу того, что у схем большой размерности для обеспечения высокого быстродействия число

связей на вентиль ограничено, существенна временная разреженность, среднее число логически активных элементов невелико (1—3%) и при этом анализируются схемы с задержками в асинхронном режиме, в основу программы моделирования ПРАЦИС-2 положен интерпретативный событийный алгоритм. Ниже рассмотрены вопросы организации, учета и обработки основной категории логического моделирования — события.

Наличие связи элемент — последователь дает возможность программе управления идентифицировать и планировать те элементы, которые потенциально могут быть активными. Адекватность моделирования во времени определяется учетом величин задержек элементов, которые используются для нахождения момента времени моделирования сигнала. Предсказанное событие, т. е. переключение сигнала на каком-либо выходе элемента, записывается в упорядоченную во времени очередь будущих событий. По мере изменения времени события обрабатываются, т. е. моделируются, как только они становятся текущими.

Программа, моделирующая отсчет времени, формирует текущее время T с дискретным регулярным приращением ΔT , называемым тактом моделирования. Событие планируется в момент изменения сигнала на линии и приема его элементами — последователями данной линии. Процесс планирования использует тот факт, что задержки элементов кратны ΔT , и заключается в следующем: обработка текущего события производится, когда установлено текущее время, событие кодируется номером линии и хранится в связанном списке вместе с указателем следующего события, относящегося к тому же моменту времени, а указатель первого события хранится во временной ячейке временного круга. По извлечении события из списка моделируются последователи активной линии, и если выходной последовательный сигнал должен изменяться, то планируется соответствующее событие на такте $T + TЗ$, где $TЗ$ — время задержки.

Управляющие и информационные сигналы задаются посредством файлов ИМЯ.ИМП или ИМЯ.ТЕС, представляющих модель сигнала в желаемом формате: временного изменения уровней (чередование высокого и низкого уровней в определенные моменты времени) либо одномерного вектора (набора единиц и нулей). Форма задания входных воздействий определяет режим моделирования (нахождение функциональных возможностей объекта, работа его в динамическом режиме, отработка временных соотношений, выявление ложных выбросов и гонок сигналов, функционирование в статическом режиме — статическое тестирование).

Поиск, построение и отработка входных тестовых последовательностей для отладки и проверки работоспособности объекта — одна из возможностей, предоставляемых пользователю программой при исследовании цифровой схемы в условиях наличия одиночных константных дефектов, т. е. разработчик в процессе создания проекта создает проверочный тест на схему. Программа проведения анализа схемы на тестируемость позволяет при этом получить ответы на вопросы, связанные с возможностью построения теста и необходимыми для этого вычислительными затратами.

При создании блока УПРАВ экспресс-анализа цифровых схем на управляемость, наблюдаемость, тестируемость использовался подход, описанный в работе [2]. Он предоставляет разработчику количественные показатели, на основании которых можно вносить изменения в проект схемы для обеспечения простоты ее проверки. Предусмотрен расчет семи количественных показателей для каждой линии схемы: 0 и 1 — комбинационные и последовательные управляемости; комбинационная и последовательная наблюдаемости; тестируемость.

Алгоритм расчета показателей достаточно прост и обладает высоким быстродействием, что дает возможность разработчику использовать данный программный блок многократно в процессе изменения проекта цифрового узла. Например, расчет показателей для цифровой БИС на основе

Схема		ЭВМ				
Имя	Характеристика	«Электроника 82»	«Электроника 79»	«Электроника 100-25»	«Электроника ВМС 11200.1»	«Электроника 60-1»
PG (N0) (мс)	107 вентиляей 563 линии 6 входов	51	229,42	314,2	422,9	339,5
MAX (N0)	800 вентиляей 806 линий 6 входов	113,25	334,39	728,8	1019,3	780,9
PRW (LB)	392 вентиля 752 линии 7 входов	23,7	59,83	81,4	111,6	87,3

нескоммутированной логической матрицы сложностью 453 вентиля произведен за 1 мин на мини-ЭВМ «Электроника 82».

Программа логического моделирования ПРАЦИС-2 написана на языке высокого уровня Фортран-IV и работает в операционной среде МДОС РВ на ЭВМ с оперативным объемом памяти не менее 64 К. Объем программы составляет 20 000 операторов. Программа используется в процессе разработки цифровых схем и верификации тестов их проверки. В таблице приведены некоторые результаты моделирования цифровых схем на различных ЭВМ семейства «Электроника».

ЛИТЕРАТУРА

1. Толстых Б. Л. и др. Унифицированные интерактивные средства проектирования изделий электронной техники.— М.: Радио и связь, 1984.
2. Бодулин С. С. и др. Автоматизированное проектирование цифровых устройств.— М.: Радио и связь, 1981.
3. Goldstein L. H. Controllability/observability analysis of digital circuits.— IEEE Trans. on Circuits and System, 1979, v. SAC-26, N 9.
4. Папернов А. А., Подъемов В. Я. Методы упорядочения информации в цифровых системах.— М.: Наука, 1973.
5. Абрайтис Л. Б. и др. Автоматизация проектирования ЭВМ.— М.: Сов. радио, 1978.

Поступила в редакцию 12 марта 1986 г.