

СХЕМОТЕХНИКА И ПРОЕКТИРОВАНИЕ

УДК 621.3.049.771.14

Методы логико-временного анализа заказных блоков СБИС

А.Л.Стемпковский, С.В.Гаврилов

Институт проблем проектирования в микроэлектронике РАН

Э.Р.Каграманян

**Московский государственный институт электронной техники
(технический университет)**

Рассмотрены проблемы логического и временного анализа, возникающие на этапе характеристики сложных заказных блоков КМОП СБИС. Предложена обобщенная логико-временная модель функционального блока, объединяющая в себе как логическую функцию на основе SP-DAG-графа, так и иерархическую структуру схемы, раскрытую до транзисторного уровня. Предложен модифицированный метод исключения Гаусса для экстракции обобщенной модели блока из транзисторного описания схемы. На основе полученной модели схемы проведены оценка проводимостей и емкостей, а также анализ задержек. Разработан алгоритм на основе метода ветвей и границ, позволяющий анализировать задержки для большого числа различных комбинаций входных воздействий, не применяя полного моделирования.

Процесс временной верификации цифровых КМОП СБИС подразумевает проверку проекта на предмет его соответствия различным временным ограничениям. Одним из основных ограничений является максимально допустимая задержка распространения сигнала в схеме от первичных входов до первичных выходов. Среди прочих причин, влияющих на задержку схемы вообще и на максимальную задержку в частности, можно отметить ее зависимость от входного набора сигналов. Иными словами, для разных пар входных наборов, переход между которыми приводит к переключению на выходе схемы, получаются различные задержки. В простейшем случае переход между двумя входными наборами осуществляется при переключении только на одном из входов схемы. Именно такой случай и представляет особый интерес, так как обычно задержку критического пути в схеме рассчитывают от одного конкретного входа до выхода при неизменных значениях на остальных входах. Очевидно, что данная задержка будет зависеть не только от параметров и количества элементов на пути распространения сигнала, но и от значений на остальных входах схемы. Следовательно, существует такая комбинация значений на остальных входах, при которой задержка от рассматриваемого входа до выхода будет максимальной.

Определение указанного набора значений является довольно непростой и трудоемкой задачей. Большие трудности в решении этой задачи возникают в тех случаях, когда проект представляет собой полностью заказную схему, не состоящую из стандартных библиотечных элементов, т.е. когда временной анализ не может быть проведен на основании результатов предварительной характеристики библиотеки цифровых вентиляей. Размер такой заказной схемы может достигать до нескольких десятков тысяч транзи-

стором. Моделирование схемы на транзисторном уровне (SPICE-моделирование) неэффективно, так как требуется перебирать все комбинации для так называемых «пассивных» (не переключающихся) входов. Значительно более эффективными в этой связи представляются методы редукции схемы для сокращения времени моделирования (при сохранении эквивалентности с исходной схемой с точки зрения максимальной задержки) при определении наихудшей комбинации значений входов. При этом последующая SPICE-верификация требуется только для точного определения максимальной задержки лишь для отобранного наихудшего варианта.

Некоторые крупные фирмы-разработчики современных СБИС, например Intel, имеют свои внутренние программные продукты, позволяющие на порядки быстрее SPICE-симуляторов определять наихудший набор входных сигналов для полностью заказных схем с точки зрения максимальной задержки. Однако при росте количества логических ограничений в схеме неполный их учет может привести к неточным результатам работы таких программ.

В настоящей работе разработаны алгоритм и методы быстрого определения наихудшей комбинации входных сигналов для характеристики полностью заказных блоков СБИС с точки зрения максимальной задержки. В основе предлагаемого подхода лежит использование комплексного логико-электрического представления заказного блока. Основными этапами реализации предлагаемого подхода являются:

- экстракция SP-DAG-графа (последовательно-параллельный направленный ациклический граф) из транзисторного описания схемы;
- преобразование произвольной структуры в последовательно-параллельную структуру посредством гауссовского метода исключения. Схема, полученная в результате преобразования, эквивалентна исходной схеме с точки зрения элморовской задержки и реализуемой функции;
- преобразование последовательно-параллельной структуры RC -цепи в эквивалентную (с точки зрения элморовской задержки) π -модель C_0-R-C_1 ;
- реализация расчета элморовской задержки для SP-DAG-графа;
- реализация алгоритма ветвей и границ для поиска критических путей и входных наборов.

Формирование иерархического представления цифровых схем. Одним из распространенных вариантов представления цифровой схемы является BDD (binary decision diagram) – диаграмма двоичных решений [1], которая довольно эффективна для анализа логических состояний, однако не отражает структуру схемы на транзисторном уровне и, как следствие, не может быть использована для точного анализа задержки. Альтернативным подходом в решении данной задачи является использование вместо BDD иерархического SP-DAG-графа для логико-временного анализа. SP-DAG-граф является удобным иерархическим представлением схемы [2, 3]. Как показано в работе [2], для большинства практических случаев SP-DAG-граф схемы имеет почти линейную сложность от числа ее элементов, а когда схема представляет собой последовательно-параллельную (SP) структуру – то строго линейную сложность. Иными словами, схема из (s) транзисторов может быть представлена с помощью SP-DAG-графа, имеющего $O(s)$ вершин. В данной работе использован метод экстракции SP-DAG-графа из транзисторного описания схемы, описанный в работах [2, 3]. При этом SP-DAG-граф схемы предлагается использовать как для представления ее логики, так и для анализа задержек. В отличие от [2, 3] рассматриваемый SP-DAG-граф кроме описания логики схемы также представляет модель ее транзисторного описания. Для этой цели используется декомпозиция схемы на подсхемы элементов, связанных по постоянному току (DCCC-компоненты – Direct Current Connected Component). Таким образом, схема может быть представлена как совокупность DCCC. В одну DCCC попадают все МОП-транзисторы, соединенные через стоки/истоки. Например, схема логического элемента «И» с тремя входами (AND3, рис.1) делится на две DCCC – элементы NAND3 (функция «И-НЕ») и INV (инвертор).

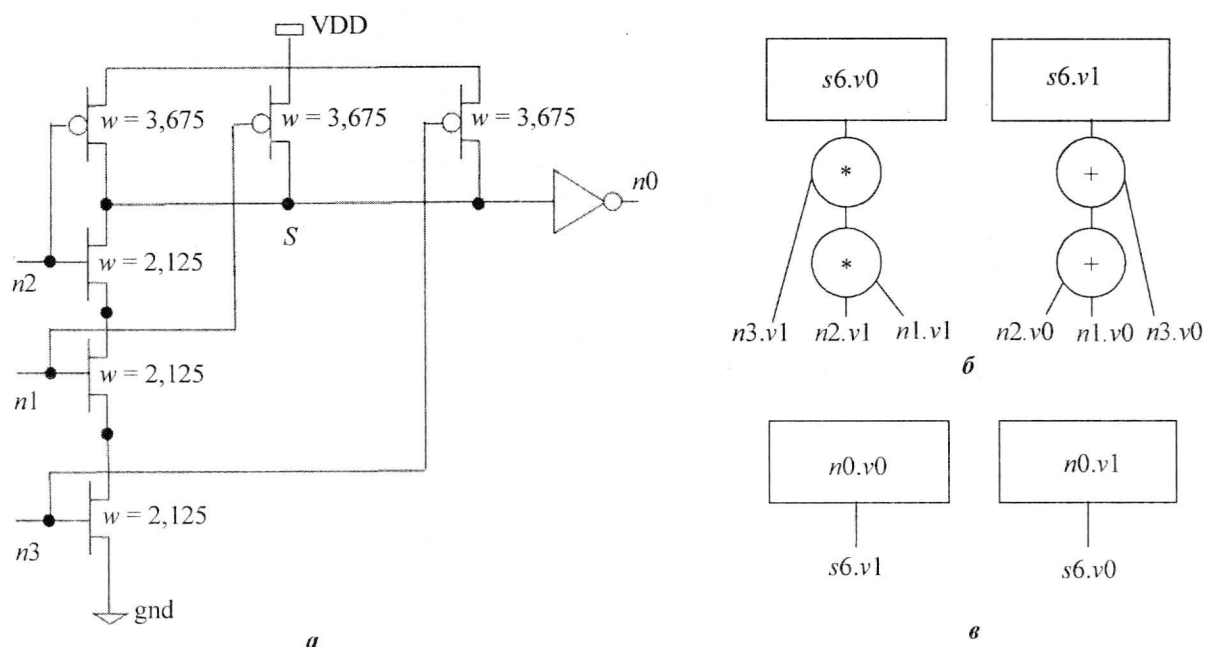


Рис.1. КМОП-элемент AND3: а – принципиальная схема; б – SP-DAG-граф первого каскада; в – SP-DAG-граф второго каскада

Предложенная модификация SP-DAG-графа предполагает наличие в нем следующих типов вершин: листовые (транзисторы и резисторы); корневые (выходы DCCC); внутренние (* – последовательные соединения; + – параллельные соединения).

В стандартных КМОП-элементах цепи установки выхода в состояние логической единицы (pull-up network) и нуля (pull-down network) не пересекаются, т.е. для таких схем можно построить два отдельных дерева (рис.1,б). Однако в случае более сложной схемы pull-up- и pull-down-деревья пересекаются (рис.2,б).

Для формирования SP-DAG-графа заказной схемы предлагается использовать алгоритм гауссовского исключения. Основным результатом формирования SP-DAG-графа является экстракция логики из транзисторного описания схемы [2, 3]. Однако в дополнение к подходу, описанному в статьях [2, 3], в процессе гауссовского исключения предлагается сохранять информацию также и о полной RC-структуре схемы в целях ее использования для анализа задержки. Таким образом, предложенный подход со-

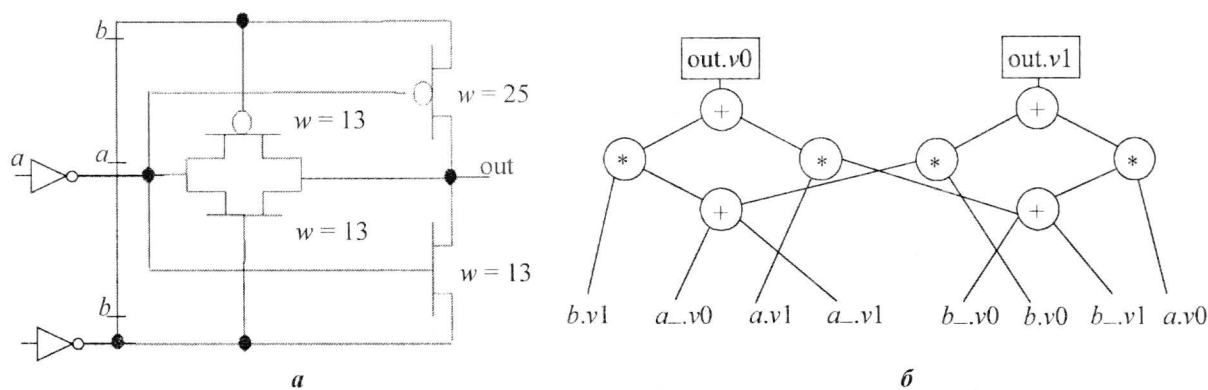


Рис.2. КМОП-элемент XNOR2: а – принципиальная схема; б – SP-DAG-граф

вмещает метод экстракции логики и RLC-редукцию [4, 5]. Однако особенность этого подхода заключается в том, что он предполагает обработку не цепей с линейными R/C компонентами, как в работах [4, 5], а переключательной схемы, состоящей из МОП-транзисторов.

Алгоритм формирования SP-DAG-графа состоит из следующих основных шагов:

Шаг 1. Выбрать очередной узел <N> для гауссовского исключения.

Шаг 2. Провести процедуру исключения выбранного узла <N>: добавить ветви, последовательно соединяющие все пары двухполосников, соединенных с узлом <N>.

Шаг 3. Пометить узел <N> и все соединенные к нему двухполосники как исключенные.

Шаг 4. Повторить шаги 1–3 для всех неисключенных узлов, кроме выхода DCCC.

Пример исключения одного узла (b) из графа двухполосников приведен на рис.3.

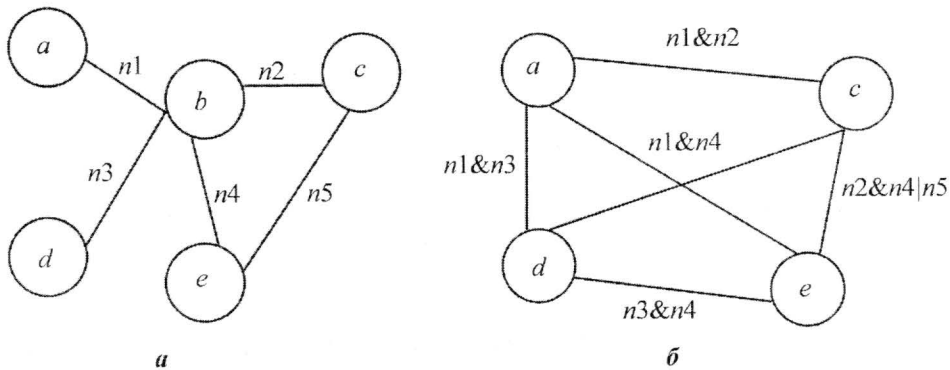


Рис.3. Пример исключения узла из схемы: а – граф до исключения узла b; б – граф после исключения узла b

Общая формула для определения числа новых ветвей имеет следующий вид:

$$\phi = l(l-1)/2 - l - p,$$

где l – число узлов, соседних тому, который надлежит исключить; p – число ранее имеющихся из вновь создаваемых двухполосников (ветвей), соединенных с этими узлами. Критерием выбора очередного узла для гауссовского исключения является количество новых ветвей, образующихся в графе после удаления выбранного узла.

Для уменьшения размерности задачи следует исключать тот узел, после удаления которого создается меньшее количество новых ветвей.

Экстракция логики выполняется отдельно для каждого выхода DCCC, т.е. процедура исключения повторяется для всех узлов DCCC, кроме ее выхода. В результате этой процедуры DCCC будет состоять всего из пары двухполосников, один из которых соединяет выход DCCC с источником питания (VDD), а второй – с «землей» (рис.4). Таким образом, алгоритм гауссовского исключения работает с двухполосниками. Исходный набор двухполосников – совокупность транзисторов и резисторов из SPICE-описания схемы. На каждом шаге исключения из схемы

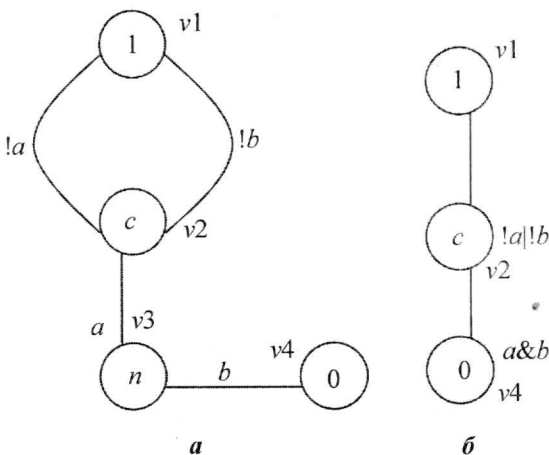


Рис.4. Пример преобразования цепей установки выхода в «1» и «0» для элемента NAND2: а – исходная схема; б – конечная схема

удаляется очередной узел и вместо двухполосников, соединенных с ним, появляются новые, которые являются последовательными или параллельными комбинациями предыдущих. Иными словами, шаг гауссовского преобразования заменяет текущую структуру двухполосников новой – сокращенной. Такое преобразование напоминает алгоритм *RC*-редукции, использованный в программе TICER [6]. Следовательно, *SP-DAG*-граф можно использовать для хранения информации о ряде *RC*-редукций. Важной отличительной особенностью предложенного подхода является создание так называемого «списка исключений» в процессе экстракции *SP-DAG*-графа. На каждом шаге редукции в данный список включаются все двухполосники, соединенные с исключенным узлом. Впоследствии этот список используется для анализа логических состояний внутренних узлов схемы, а также для перерасчета емкостей.

Формирование эквивалентной π -модели. Основной целью преобразования *RC*-цепей является максимально возможное упрощение сложной структуры схемы, то есть ее преобразование в простую структуру для расчета ее элморовской задержки. Существуют алгоритмы расчета эквивалентной задержки как для *RC*-цепей, так и для произвольных последовательно-параллельных структур. Однако на практике схема не всегда представляет собой чисто последовательно-параллельную структуру, либо *RC*-цепь. В данной работе модифицирован алгоритм *RLC*-преобразования, который позволяет свести произвольную схему к последовательно-параллельной структуре. Также предложен метод преобразования последовательно-параллельной структуры в эквивалентную π -модель, состоящую из трех элементов: C_0-R-C_1 , сохранив при этом элморовскую задержку неизменной.

Существует множество алгоритмов преобразования *RC*-структур. Как правило, основной идеей данных алгоритмов является сокращение числа элементов структуры в целях ускорения процесса электрического моделирования. Сокращение числа паразитных элементов цепей межсоединений, получаемых после экстракции, представляет собой актуальную задачу с точки зрения ускорения работы программы так называемого «быстрого моделирования». За последние годы в этом направлении был проведен ряд исследований, основные результаты которых отражены в работах [6–8].

В общем случае *RLC*-редукция представляет собой аппроксимацию заданной схемы при помощи схемы более низкого порядка с использованием согласования моментов и проекционных способов на основе *AWE*-преобразования [7]. Самой известной модификацией *AWE*-преобразования является подход, использованный в программе PRIMA [8]. Однако для решения этой задачи в данном конкретном случае применять матричное преобразование, описанное в работе [6], не имеет смысла, так как интерес представляет только локальное преобразование описания схемы.

Единичный шаг локальной редукции может быть реализован двумя возможными способами: 1) удаление элемента и соединение двух узлов, с которым он был соединен; 2) удаление узла и преобразование всех элементов, соединенных с данным узлом.

Второй способ используется в так называемом методе гауссовского исключения и его модификациях, таких как TICER [6]. Обычно гауссовское исключение подразумевает выборочное удаление узлов схемы и преобразование соединенных с удаленным узлом элементов в новые элементы на основе аппроксимации передаточной функции методом Тейлора. Метод, использованный в TICER, предусматривался для обработки массивов, содержащих только *R*- и *C*-элементы. В более поздних публикациях описываются попытки расширения метода гауссовского исключения для работы со схемами, также содержащими индуктивные элементы (*L*) и взаимные индуктивности (*M*). В данной работе предлагается ограничиться общим случаем применения гауссовского исключения, которое будет использоваться для преобразования *RC*-структуры в эквивалентную π -модель с точки зрения элморовской задержки. Также предложен алгоритм

для перерасчета схемы и хранения данных о предыдущих состояниях, который применяется для транзисторной схемы, где каждый транзистор может работать в проводящем либо в закрытом режиме.

RLC -цепь может быть представлена в виде множества соединенных двухполюсников, каждый из которых относится к одному из следующих типов:

- резистор с номиналом $R_{ij} = (g_{ij})^{-1}$, где g_{ij} – проводимость, Ом^{-1} ;

- конденсатор с емкостью C_{ij} , Ф ;

- индуктивность $L_{ij} = (b_{ij})^{-1}$, где b_{ij} – значение индуктивной реактивной проводимости, Гн^{-1} .

Допустим, что в RLC -цепи имеется n узлов. Напряжения $V(s)$ в s -области удовлетворяют следующему условию:

$$Y(s)V(s) = 0,$$

где $Y(s)$ – матрица обобщенных проводимостей (адмиттанс-матрица) с размерностью $n \times n$. Значения обобщенных проводимостей для первичных двухполюсников определяются следующим образом:

$y_{ij} = \frac{1}{R_{ij}} = g_{ij}$ – проводимость между узлами n_i и n_j , соединенными резистором;

$y_{ij} = sc_{ij}$ – адмиттанс между узлами n_i и n_j , соединенными конденсатором;

$y_{ij} = \frac{1}{sL_{ij}} = \frac{b_{ij}}{s}$ – адмиттанс между узлами n_i и n_j , соединенными индуктивностью;

При параллельном соединении резистора, конденсатора и индуктивности суммарный адмиттанс между узлами равен сумме обобщенных проводимостей отдельных элементов, т.е. полная проводимость между узлами n_i и n_j определяется как

$$y_{ij} = \frac{b_{ij}}{s} + g_{ij} + sc_{ij} = \frac{1}{s}(b_{ij} + sg_{ij} + s^2c_{ij}).$$

В случае последовательного соединения суммарный адмиттанс определяется по формуле суммы обратных адмиттансов:

$$y_{ij} = \frac{y_{ik}y_{kj}}{y_{ik} + y_{kj}}. \quad (1)$$

Формула (1) справедлива только для случаев, когда к узлу k подключены два двухполюсника. Однако в общем случае гауссовского исключения рассматривается узел с произвольным числом соединенных к нему элементов. Допустим, требуется исключить из схемы узел n_k . Следовательно, из схемы должны быть удалены все двухполюсники (n_i, n_k) . Каждая пара двухполюсников (n_i, n_k) , (n_k, n_j) заменяется новым двухполюсником (n_i, n_j) в соответствии с формулой

$$y_{ij} = \frac{y_{ik}y_{kj}}{Y_k}, \quad (2)$$

где Y_k – суммарная полная обобщенная проводимость узла n_k до исключения элемента:

$$Y_k = \sum_i y_{ki} = \frac{B_k}{s} + G_k + sC_k.$$

Если ветвь (n_i, n_j) присутствует в RLC -графе до исключения узла n_k , то по формуле (2) можно рассчитать дополнительную проводимость ветви (n_i, n_j) , в противном случае в граф добавляется новая ветвь.

Допустим, рассматриваются два соседних двухполюсника с проводимостями

$$y_{ki} = \frac{1}{s}(b_i + sg_i + s^2c_i) \text{ и } y_{kj} = \frac{1}{s}(b_j + sg_j + s^2c_j).$$

В этом случае общая формула для новой ветви получается на основе выражения (2) и будет выглядеть следующим образом:

$$y_{ki} = \frac{1}{s^2 Y_k} (b_i b_j + s(b_i g_j + b_j g_i) + s^2(g_i g_j + b_i c_j + b_j c_i) + s^3(g_i c_j + g_j c_i) + s^4 c_i c_j). \quad (3)$$

Как правило, большинство элементов этой формулы равно нулю. Для реализации быстрой аппроксимации этой формулы вводится понятие «быстрого» узла с малой временной постоянной. RC -постоянная времени определяется формулой $\tau_{RCk} = C_k/G_k$, а LC -постоянная – формулой $\tau_{LCk} = \sqrt{C_k/B_k}$. Также можно ввести понятие RL -постоянной времени: $\tau_{RLk} = G_k/B_k$. Узел считается «быстрым», если для него удовлетворяется условие: $\max(\tau_{RCk}, \tau_{LCk}, \tau_{RLk}) < \tau_{limit}$. При аппроксимации быстрых узлов в формуле (3) не учитываются компоненты старшего s -порядка. В частности, для случая RC -цепей без индуктивностей в программе PICER согласно элморовскому подходу используются только два первых члена ряда Тейлора:

$$y_{ij} = \frac{1}{G_k} (g_i g_j + s(c_i g_j + c_j g_i) + \dots). \quad (4)$$

Первый коэффициент в скобках формулы (4) представляет собой дополнительную проводимость (сопротивление), а коэффициент после s – дополнительную емкость.

«Быстрый» узел с индуктивностью может быть представлен с помощью следующей аппроксимации:

$$y_{ij} = \frac{1}{sB_k} (b_i b_j + s(b_i g_j + b_j g_i) + s^2(g_i g_j + b_i c_j + b_j c_i) + \dots). \quad (5)$$

Первый коэффициент в скобках формулы (5) представляет собой дополнительную индуктивность, второй – дополнительную проводимость, а третий – дополнительную емкость. Для «быстрого» узла имеет место соотношение $B_k > sG_k > s^2C_k$. Стоит отметить, что более точный вариант формулы (5) может быть получен при использовании

полного ряда Тейлора для случая $\frac{1}{Y_k} = \frac{1}{B_k + sG_k + s^2C_k}$.

Более простым является случай, когда к узлу подключены только последовательные емкости и отсутствуют резистивные и индуктивные элементы. В этом случае ($B_k = 0, G_k = 0$):

$$y_{ij} = \frac{1}{C_k} (c_i c_j). \quad (6)$$

Так как в нашем случае целью является редукция схемы без изменения элморвской задержки, можно пренебречь условием $\max(\tau_{RCK}, \tau_{LCK}, \tau_{RLK}) < \tau_{limit}$. Таким образом, преобразование осуществляется одним из следующих способов:

- если $B_k = 0, G_k \neq 0$, используется формула (4);
- если $B_k \neq 0$, используется формула (5);
- если $B_k = 0, G_k = 0$, используется формула (6).

Анализ задержек. Как уже отмечалось, SP-DAG-граф описывает весь процесс преобразования исходной структуры схемы в π -модель C_0-R-C_1 , которая эквивалентна исходной схеме с точки зрения элморвской задержки. Задержка π -модели определяется следующим выражением:

$$d = C_1/g, \quad (7)$$

где C_1 – выходная эквивалентная заземленная емкость; $g = 1/R$ – эквивалентная внутренняя проводимость. Следовательно, для расчета задержки необходимо определить значения емкости и проводимости. Для этой цели используется информация, хранящаяся в SP-DAG-графе и в списке исключений.

Анализ задержки можно разделить на следующие основные этапы:

- расчет логических состояний всех внутренних и выходных узлов для заданного входного набора до и после переключения;
- расчет эквивалентных проводимостей pull-up- и pull-down-цепей;
- расчет эквивалентных нагрузочных емкостей для переключений из «1» в «0» и обратно;
- оценка задержек переключения из «1» в «0» (fall delay) и обратно (rise delay).

Несложно заметить, что определяемая емкость будет зависеть от предыдущих и от текущих значений узлов схемы, так как если узел не переключается, соответствующая емкость не влияет на задержку. Также очевидно, что проводимость будет зависеть только от текущего значения узла.

Расчет логических состояний всех узлов схемы проводится с помощью метода анализа переключательной логики схемы [2]. При оценке проводимостей в настоящей работе делаются предположения: проводимость линейного сопротивления – постоянная величина, обратная значению резистора из исходного описания схемы; проводимость транзистора равна нулю, если он закрыт, и отлична от нуля, если он открыт. Режим транзистора определяется логическим значением на его затворе. Проводимость рассчитывается исходя из предположения, что она прямо пропорциональна ширине его канала.

Оценка емкостей проводится только для тех узлов, которые переключаются при заданном переключении на входе. При этом для перехода к π -модели емкость каждого из внутренних узлов преобразуется в две боковые емкости.

Реализация метода ветвей и границ. Основной задачей, решаемой в рамках данной работы, является определение заранее неизвестных значений на входах, при которых заданное на одном из входов переключение будет иметь максимальную задержку распространения до выхода. Естественно, можно найти эти значения методом простого перебора. Однако такой подход эффективен лишь для схем с небольшим количеством входов, поскольку сложность алгоритма полного перебора имеет экспоненциальную зависимость ($O(N \cdot 2^{N-1})$) от числа входов (N). Для преодоления этой проблемы предлагается более эффективный подход на основе метода ветвей и границ. Суть данного подхода заключается в попытке определить неизвестные значения входов на основе имеющихся логических ограничений. Для того чтобы не занижать реальную оценку задержки, необходимо каждое неопределенное состояние узла схемы рассматривать как

потенциальное переключение. Это заставляет внести изменения в процедуры анализа логики и оценки емкостей и проводимостей. Так, общая проводимость любого из узлов схемы, за невозможностью ее точной оценки, заменяется двумя крайними значениями – минимальным и максимальным. Первое используется для нижней оценки проводимости ($\min(g)$), второе – для верхней оценки емкости ($\max(C)$). Возможная максимальная задержка определяется как $\max(d) = \max(C)/\min(g)$.

Метод ветвей и границ заключается в следующем. Задаются входные наборы до и после переключения, при этом известны значения лишь части входных сигналов до и после переключения, а остальные входы неизвестны. Ниже приведены основные шаги алгоритма.

Шаг 1. Установить наихудшую задержку равной нулю.

Шаг 2. Выбрать из списка входов вход с неопределенным состоянием. Установить начальное и конечное состояния входа последовательно равными нулю и единице. Для каждого из двух вариантов значений выполнить шаг 3.

Шаг 3. Если найденный входной набор удовлетворяет условиям переключения и приводит к большей задержке, чем при найденном ранее наборе, то задержка, найденная для данного набора, становится наихудшей.

Шаг 4. Если сохранились неопределенные входы, перейти к следующему входу с неопределенным состоянием, для которого повторить шаги 2 и 3.

Состояние любого входа может быть автоматически определено, если задано достаточное количество логических ограничений. Таким образом, наличие логических ограничений на входные состояния схемы позволяет ускорить процесс нахождения входных наборов.

Программная реализация. Общая блок-схема предложенного подхода представлена на рис.5. Все использованные алгоритмы реализованы в виде С-кода (стандарт ANSI). Разработана экспериментальная программа «S2S», которая была проинтегрирована в среде

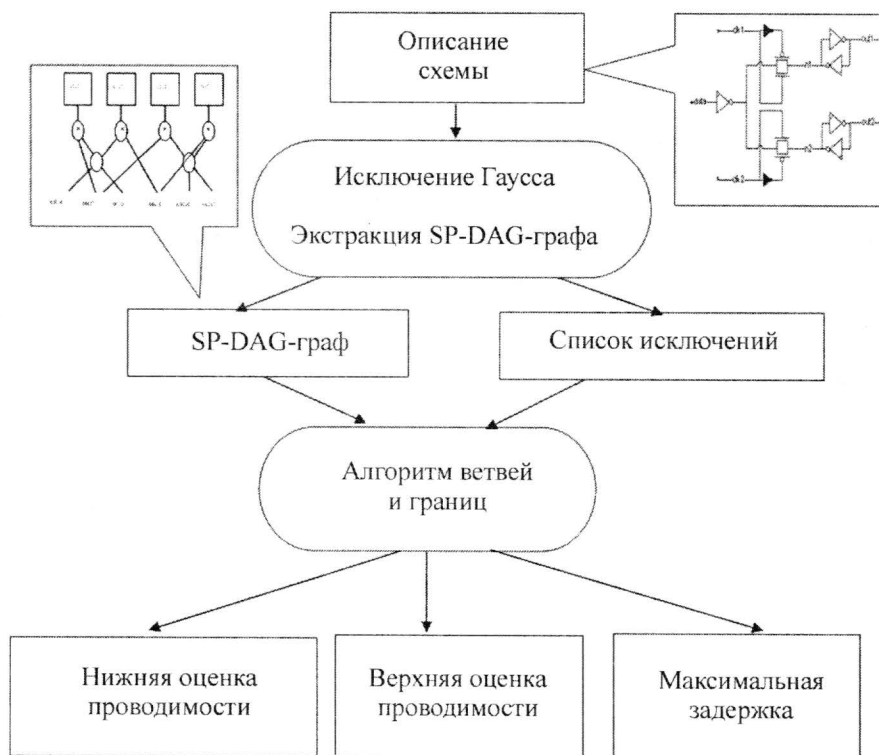


Рис. 5. Общая блок-схема предложенного подхода

коммерческой САПР компании Intel и опробована на реальных схемах, разрабатываемых с использованием новейших технологий с проектной нормой 65 и 45 нм. Проведено сравнение результатов предлагаемого подхода с результатами точного электрического моделирования более чем на 30 тыс. тестовых примеров. Полученные результаты позволили сделать вывод о высокой степени эффективности предлагаемого подхода.

Авторы выражают глубокую признательность компании Intel за поддержку исследований по данному направлению работ.

Литература

1. **Bryant R.E.** Graph-based algorithms for boolean function manipulation // IEEE Trans. Computers. – 1986. – P. 677–691.
2. **Bryant R.E.** Algorithmic aspects of symbolic switch network analysis // IEEE Trans. on CAD. – 1987. – P. 618–633.
3. **Bryant R.E.** Boolean analysis of MOS circuits // IEEE Trans. on CAD. – 1987. – P. 634–649.
4. **Qin Z., Cheng C.-K.** Realizable parasitic reduction using generalized Y- Δ transformation // Proc. of DAC, 2003. – P. 220–225.
5. **Amin C.S., Chowdhury M.H., Ismail Y.I.** Realizable RLCK circuit crunching // Proc. of DAC. – 2003. – P. 226–231.
6. **Sheehan B.N.** TICER: Realizable reduction of extracted RC circuits // Digest of Technical Papers, IEEE/ACM Proc. of ICCAD. – 1999. – P. 200–203.
7. **Pillage L.T., Rohrer R.A.** Asymptotic waveform evaluation for timing Analysis // IEEE Trans. on CAD. – 1990. – Vol. 9, N 4. – P. 352–366.
8. **Odabasioglu A., Celik M., and Pillegi L.T.** PRIMA: Passive reduced-order interconnect macromodeling Algorithm // IEEE Trans. on CAD. – 1998. – P. 645–654.

Статья поступила
2 июня 2008 г.

Стемпковский Александр Леонидович – академик РАН, лауреат Государственной премии РФ, директор Института проблем проектирования в микроэлектронике РАН. *Область научных интересов:* автоматизация проектирования микроэлектронных систем.

Гаврилов Сергей Витальевич – доктор технических наук, заведующий сектором Института проблем проектирования в микроэлектронике РАН. *Область научных интересов:* методы оптимизации СБИС, методы быстрого электрического моделирования, символический анализ схем, анализ помехоустойчивости.

Каграманян Эмиль Рудольфович – аспирант кафедры проектирования и конструирования интегральных микросхем МИЭТ. *Область научных интересов:* методы проектирования аналоговых и цифровых СБИС, анализ аналого-цифровых блоков интеллектуальной собственности.