

БЫСТРЫЙ АЛГОРИТМ РАСЧЕТА МОЩНОСТИ В ЦИФРОВЫХ КМОП СХЕМАХ

В работе описан алгоритм расчета потребляемой мощности для статических КМОП-схем, основанный на ключевом моделировании. Он учитывает как мощность, идущую на перезарядку емкостей, так и мощность сквозных токов. Благодаря высокому быстродействию алгоритм применим в оптимизационных процедурах при моделировании больших схем.

Также предложена очень быстрая версия алгоритма ключевого моделирования с вычислением мощности, основанная на SP-BDD-представлении для стандартных цифровых КМОП-схем [1]. Поскольку для точного вычисления потребляемой мощности крайне важно правильное определение задержек КМОП-вентилей, представлено описание оригинального алгоритма вычисления таких задержек, являющегося частью алгоритма вычисления мощности и также основанного на SP-BDD-модели.

ВВЕДЕНИЕ

Проектирование схем с малой потребляемой мощностью, в том числе – статических КМОП-схем [2, 3], привлекает все больше внимания. И вместе с ним растет актуальность средств быстрой и точной оценки потребляемой мощности – одного из ключевых инструментов в проектировании и оптимизации подобных схем. Известны два основных подхода к решению проблемы оценки потребляемой мощности. В первом случае мощность оценивают, задавшись определенной частотой переключения вентилях схемы, исходя из статистической информации о сигналах на ее первичных входах. При этом не моделируют процессы переключений в схеме при прохождении определенных сигналов. Этот подход эффективен на этапе логического синтеза с оптимизацией задержки [4] и мощности [5, 6] КМОП-схем в силу его простоты и очень высокого быстродействия. Однако точность при этом невысока, даже с учетом корреляций сигналов.

Второй подход (к нему относится и предлагаемый нами алгоритм) использует моделирование переключений в схеме на тестовой последовательности векторов входных сигналов. Результат получается много точнее, но и время расчета существенно возрастает, что может стать проблемой при многократных вычислениях потребляемой мощности в цикле оптимизации больших схем.

Ясно, что детальное электрическое моделирование, подобное реализованному в программе SPICE, – процесс слишком медленный. Также медленны либо не достаточно точны упрощенные варианты детального моделирования, включая алгоритм ELOGIC [7, 8].

С.Гаврилов, А.Глебов,
А.Стемпковский

Недостаточно точно и логическое моделирование на уровне вентилей, поскольку не учитывает переключений внутренних узлов схемы и реальные задержки сигналов.

Для наших целей наиболее пригодно ключевое моделирование [8–10], как достаточно точный и в то же время быстрый для оптимизационных алгоритмов метод. Оно позволяет для единичного переключения входов вычислить потенциалы для стационарных состояний схемы до и после переключения, а также время и длительность переключения каждого транзистора. Однако все известные модификации данного метода ориентированы на проверку логики работы или быстродействия схемы и, как правило, не вычисляют точные значения потенциалов для всех ее узлов. Следовательно, для точной оценки мощности метод должен быть доработан.

ОЦЕНКА МОЩНОСТИ ПОТРЕБЛЕНИЯ КМОП-СХЕМЫ

Под *переключением схемы* будем понимать все процессы в схеме, к которым приводят одновременное переключение на одном или более ее входах. При расчете можно учитывать рассеивание мощности только на транзисторах либо на всех элементах фрагмента схемы, в том числе и мощность перезарядки входных емкостей. Рассмотрим оценку мощности потребления только внутри схемы, без учета перезарядки входных емкостей фрагмента.

Представляем авторов статьи

СТЕМПКОВСКИЙ Александр Леонидович, доктор технических наук, член-корреспондент РАН. Автор более 86 научных работ, в том числе – монографии и двух изобретений. Специалист в области автоматизации проектирования электронно-вычислительных систем. Директор Института проблем проектирования в микроэлектронике РАН (г. Москва), профессор кафедры МГИЭТ (ТУ).

ГАВРИЛОВ Сергей Витальевич, кандидат технических наук, заведующий сектором автоматизации топологического проектирования ИППМ РАН. Область научных интересов – методы оптимизации СБИС, методы быстрого электрического моделирования, символический анализ схем, анализ помехоустойчивости.

ГЛЕБОВ Алексей Львович, кандидат физико-математических наук, заведующий сектором автоматизации логического проектирования ИППМ РАН. Область научных интересов – методы логического синтеза, анализ помехоустойчивости и оптимизации СБИС. E-mail: alex@ici.ru

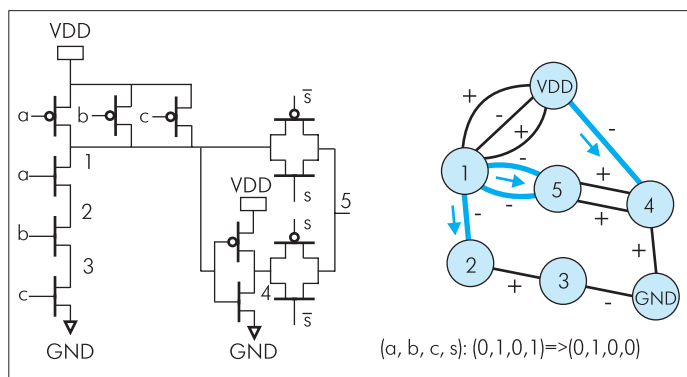


Рис.1. Пример DCCC и соответствующего графа схемы.
На графе "+" – проводящий транзистор, "-" – непроводящий.
Толстой линией показаны транзисторы закоротки

В отдельно взятой подсхеме, связанной по постоянному току (DCCC – dc-connected component), некоторые транзисторы переключаются в проводящее состояние (включаются), другие – в непроводящее (выключаются). В стационарном состоянии после переключения часть DCCC оказывается связанной по постоянному току (dc-связанные) с VDD-узлом через проводящие транзисторы (множество А узлов). Остальные DCCC изолированы по постоянному току от VDD-узла (множество В узлов). Можно построить граф схемы с вершинами, соответствующими узлам, и ребрами, соответствующими каналам транзисторов. После каждого переключения множества узлов А и В порождают подграфы А и В. Так, для схемы на рис.1 при состоянии вектора входов $(a, b, c, s) = (0, 1, 0, 0)$ подграф А содержит узлы $\{VDD, 1\}$, а подграф В – $\{GND, 2, 3, 4, 5\}$.

Подграфы А и В связаны через отдельные транзисторы. Назовем их транзисторами сквозных токов, или транзисторами закоротки (на рис.1 показаны толстыми линиями). Это понятие относится, естественно, к состоянию схемы после конкретного переключения.

Исходя из вышесказанного, определим **энергию перезарядки емкостей (CR-энергия – capacity recharging)** для единичного переключения схемы как **полную потерю энергии, вычисленную в предположении, что все транзисторы закоротки переключаются мгновенно в начале переключения схемы.** Тогда **энергия сквозных токов (SC-энергия – short circuit energy)** для единичного переключения схемы – **это полная энергия за вычетом CR-энергии.**

Мгновенное рассеивание мощности в транзисторе (или нескольких транзисторах), соединяющих своими каналами k -й и l -й узлы, определяется выражением

$$P_{kl} = (V_k - V_l) \cdot I_{kl}, \quad (1)$$

где V_k – потенциал k -го узла, I_{kl} – ток от k -го к l -му узлу.

Суммируя (1) по всем транзисторам схемы, получим:

$$P = \sum_k V_k I_k + V_{dd} I_{dd}, \quad (2)$$

где суммирование производится по всем узлам схемы, кроме узла питания (VDD), I_k – полный ток, вытекающий из k -го узла через проводящие транзисторы, I_{dd} – полный ток, втекающий в схему через VDD-узел, т.е. ток питания. Предполагается, что все входы схемы с потенциалом V_{dd} также присоединены к VDD-узлу. Считая, что в каждом узле k есть заземленная емкость C_k , можно записать

$$I_k = -C_k dV_k/dt. \quad (3)$$

Потери энергии при одном переключении получим, интегрируя (2) по времени переключения с учетом (3). Интегрирование первого члена в (2) дает

$$\int (\sum_k V_k I_k) dt = 0,5 \sum_k C_k (U_k^2 - V_k^2), \quad (4)$$

где U_k, V_k – старый и новый потенциал k -го узла. Суммирование – по всем узлам схемы, кроме VDD-узла.

Полный заряд, протекший в схему через VDD-узел, равен увеличению полного заряда на емкостях всех узлов, dc-связанных с VDD-узлом (все узлы подграфа А, кроме VDD-узла), плюс полный заряд, вытекший из подграфа А через транзисторы закоротки. Следовательно, интегрируя второй член в (2), получим

$$\int V_{dd} I_{dd} dt = V_{dd} \sum C_i (V_i - U_i) + V_{dd} Q, \quad (5)$$

где суммирование производится по узлам схемы, dc-связанным с VDD-узлом через проводящие транзисторы, V_{dd} – напряжение питания. Член $V_{dd} Q$ соответствует потере энергии сквозных токов. Q – полный заряд, перетекший через транзисторы закоротки в течение одного переключения схемы: $Q = \int (\sum_m J_m) dt$, где J_m – ток через m -й транзистор. Интегрирование производится по времени переключения, суммирование – по транзисторам закоротки.

Складывая (4) и (5), окончательно получим потери энергии в статической КМОП-схеме при единичном переключении:

$$E = 0,5 \sum_i C_i (U_i^2 - V_i^2) + V_{dd} \sum_j C_j (V_j - U_j) + V_{dd} Q. \quad (6)$$

Первая сумма вычисляется по всем узлам схемы, вторая – по узлам схемы, dc-связанным с узлом питания после переключения. Первые два члена показывают потери энергии в пренебрежении сквозными токами, которые характеризует последний член.

Для вычисления энергии сквозных токов для одного переключения необходимо идентифицировать транзисторы закоротки для этого переключения, а затем вычислить полный заряд, перетекший через эти транзисторы из подграфа А. Существенно, что энергия сквозных токов может быть отрицательной. Это означает, что энергия, накопленная в узловых емкостях перед переключением, частично высвобождается в процессе переключения. Возможно, что конструирование схем со значительной отрицательной энергией сквозных токов – один из способов снижения потерь мощности.

В примере переключения на рис.1 SC-энергия, конечно, положительна (положительные направления токов через транзисторы закоротки показаны стрелками). Но для другого переключения с тем же конечным состоянием, $(0,0,0,1) \rightarrow (0,1,0,0)$, SC-энергия может быть отрицательной, благодаря частичному высвобождению энергии из емкостей узлов 1, 4 и 5.

ПОСЛЕДОВАТЕЛЬНОСТЬ ПРОЦЕДУР МОДЕЛИРОВАНИЯ

Ключевое моделирование уже использовали для оценки мощности. Так, в работе [11] применяли коммерчески доступную программу ключевого моделирования. Главным недостатком этого подхода в том, что не вычислялись точные значения потенциалов всех узлов схемы, а использовались лишь множество $\{0, 1\}$ (т.е. только два возможных уровня, соответствующих логическим 0 и 1). Кроме того, не вычислялась мощность сквозных токов.

Использовать же точные значения узловых потенциалов при оценке мощности важно, потому что только часть узлов в статической КМОП-схеме переключается между потенциалами земли и питания. Потенциалы остальных узлов принимают некоторые промежуточные значения вследствие “плохих” логических уровней и разделения заряда.

Например, в трехходовом вентиле И-НЕ (рис.2) при напряжении питания 5 В и одинаковых n - и p -канальных транзисторах с пороговым напряжением 1 В узел 1 переключается между 0 и 4 В из-за “плохого” логического уровня, а узел 2 переключается между 0 и 2 В из-за разделения заряда между узлами 1 и 2. Следовательно,

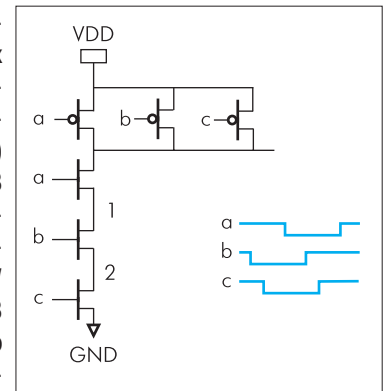


Рис.2. Вентиль 3И-НЕ и типичный входной сигнал.

оценка мощности без учета точных значений узловых потенциалов может приводить к значительной ошибке.

Для решения данной задачи был разработан новый алгоритм ключевого моделирования. От обычного алгоритма такого типа (например, от алгоритма SLS [12]) он отличается тем, что разработан специально для статических КМОП-схем, что упростило обработку графа схемы – как правило, каждый проводящий транзистор проходит единожды и в одном направлении. Кроме того, алгоритм точно вычисляет потенциал каждого узла схемы, а не оперирует множеством состояний $\{0, 1, X\}$.

Рассмотрим моделирование схемы небольшого размера, содержащей одну или несколько DCCC. Используется простая ключевая модель МОП-транзистора, где его затвор, исток и сток присоединены к линейной заземленной емкости, величина которой зависит от размеров и типа транзистора. Транзистор может быть либо в проводящем, либо в непроводящем состоянии, в зависимости от его типа и потенциала затвора. Потери мощности, вычисляемые в пренебрежении сквозными токами (CR-мощность), не зависят от конкретного поведения транзистора в проводящем состоянии (линейное или нелинейное сопротивление канала и т.д.).

Пусть V_{tn} и V_{tp} – пороговые напряжения n - и p -канальных транзисторов (оба положительные). В каждом стационарном состоянии статической КМОП-схемы граф схемы содержит три подграфа: А, С и D. Подграф А включает узлы, dc-связанные с VDD-узлом (значение потенциала V_{dd} или $V_{dd} - V_{tn}$). Подграф С – узлы, dc-связанные с GND-узлом (значение потенциала 0 или V_{tp}). Подграф D объединяет узлы, dc-изолированные как от VDD, так и от GND (любое промежуточное значение потенциала). Очевидно, объединение подграфов С и D – это подграф В, упоминавшийся выше. Узлы подграфа D, связанные между собой, но изолированные от остальных узлов этого подграфа, образуют CS-группы (Charge Sharing – разделение заряда).

В процессе моделирования все переключения схемы обрабатываются последовательно. В процессе обхода графа схемы при прохождении через проводящий транзистор вычисляется новый потенциал узла, исходя только из нового потенциала другого узла (другого терминала транзисторного канала) и типа транзистора. Повторный обход некоторых частей графа схемы до установления потенциалов необходим, только если встретилась петля обратной связи – внутри одной DCCC или охватывающая несколько DCCC (итерации в пределах одной или нескольких DCCC), а также если при прохождении графа схемы получен “плохой” логический уровень узла, для которого ранее (при попадании в него по другому пути графа, т.е. через другой транзистор) был вычислен новый потенциал “хорошего” логического уровня.

```

прочитать описание транзисторной схемы;
разбить схему на DCCC;
упорядочить DCCC;
for (каждой DCCC)
    вычислить задержки; /* задержки вычисляются для каждого входа DCCC, для обоих типов переключения */
рассчитать исходное состояние схемы;
очередь событий пуста;
for (каждой DCCC)
    DCCC.энергия = 0;
for (каждого тест-вектора, кроме исходного) {
    for (каждого переключения первичного входа)
        вставить новое событие в очередь событий;
    while (очередь событий не пуста) {
        текущее_событие = первое в очереди событий;
        текущее_время = текущее_событие.время;
        if (текущее_время > ВРЕМЯ_МОДЕЛИРОВАНИЯ)
            break;
        if (тествектор не последний && текущее_время > время следующего тествектора)
            break; /* пора взять следующий тествектор */
        for (каждой DCCC из fanout (текущее_событие.узел)) {
            рассчитать новое состояние DCCC; /* новые потенциалы всех узлов DCCC */
            рассчитать новое событие для выхода DCCC;
            вставить новое событие в очередь событий;
            рассчитать DCCC.CR_энергия;
            рассчитать DCCC.SC_энергия;
            DCCC.энергия += DCCC.CR_энергия + DCCC.SC_энергия;
        }
        удалить текущее_событие из очереди событий;
    }
    if (текущее_время > ВРЕМЯ_МОДЕЛИРОВАНИЯ)
        break;
}
полная_энергия = 0;
for (каждой DCCC)
    полная_энергия += DCCC.энергия;

```

Как только идентифицирована CS-группа, новый потенциал для ее узлов вычисляется из условия сохранения заряда, предполагая мгновенное переключение всех транзисторов к их окончательным состояниям:

$$V = (\sum_i C_i U_i) / (\sum_i C_i), \quad (7)$$

суммирование производится по узлам CS-группы.

При моделировании единичного переключения для одной DCCC сначала обходятся все нижние пути из подграфа С, затем все верхние пути из подграфа А, далее – остальная часть подграфа С, остальная часть подграфа А и подграф D (CS-группы).

Большая КМОП-схема при моделировании разбивается на отдельные DCCC (см. врезку), которые упорядочиваются в соответствии с направлением распространения сигнала (если это возможно) [13]. Моделирование всей схемы – событийное (под событием здесь понимают переключение узла), с использованием глобальной очереди событий. Если событие происходит на одном из входов DCCC, для этой DCCC производится ключевое моделирование с вычислением CR-энергии, как было описано выше. Если на выходе DCCC генерируется новое событие, то время этого события определяется по времени события на входе с использованием Элморовской задержки [14]. Аналогично вычисляется длительность фронта для нового события. Кроме того, вычисляется и SC-энергия переключения (модель будет описана ниже).

Если для некоторой DCCC два входных события оказываются очень близкими по времени, то на выходе DCCC может быть сгене-

рирован неполный паразитный импульс (импульс, образованный двумя событиями, настолько близкими по времени, что каждое из них не является в действительности полным переключением). В этом случае вычисляется потеря энергии для текущей DCCC, однако неполный паразитный импульс далее по схеме не распространяется.

Нетрудно увидеть, что после небольшой модификации данный алгоритм может также вычислять зависимость мгновенного потребления мощности от времени (как для всей схемы, так и для отдельных вентилях и DCCC).

КЛЮЧЕВОЕ МОДЕЛИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ SP-BDD

В работах [1, 15] было описано новое и эффективное представление для статических КМОП-схем – последовательно-параллельная диаграмма двоичных решений (SP-BDD, Series-Parallel Binary Decision Diagram). Там же приведены алгоритмы экстракции SP-BDD и основных процедур для работы с ними.

SP-BDD для верхней (pull-up) цепи КМОП-вентиля – это граф, каждая вершина которого соответствует р-МОП-транзистору, а ветви, выходящие из вершин, – возможным логическим состояниям транзистора (непроводящее/проводящее, 0/1) (рис.3). Соответственно, следующие вершины на этих ветвях называют 0-потомком и 1-потомком (на рис.3 – левая и правая исходящие ветви, соответственно). Существенно, что любая вершина SP-BDD не может быть одновременно 0- и 1-потомком других вершин: если вершина х – 1-потомок вершины у, то она может быть только 1-потомком дру-

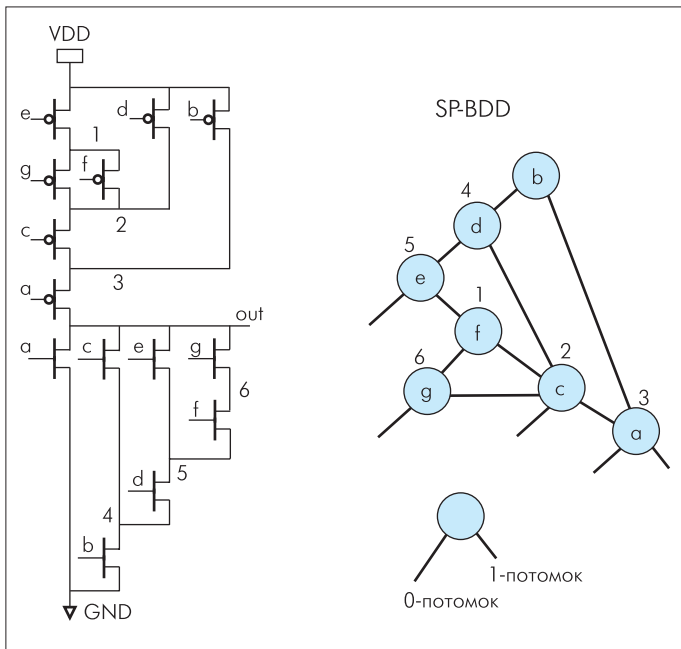


Рис.3. КМОП-вентиль и ассоциированная с ним SP-BDD. Цифрами показаны узлы КМОП-вентиль

гой вершины z . В работе [15] доказано, что для КМОП-вентиль существует единственная SP-BDD, причем она описывает как реализуемую им булевскую функцию, так и электрическую схему (рис.3).

Для КМОП-вентиль каждая вершина SP-BDD соответствует паре транзисторов (p -МОП в верхней и n -МОП в нижней цепи) с объеди-

ненными контактами затворов, образующими вход вентиля. Будем называть транзистор, подключенный к узлу истоком, верхне-присоединенным, а подключенный стоком – ниже-присоединенным (считаем, что исток p -канального транзистора всегда направлен в сторону узла питания VDD, n -канального – в сторону узла GND). Можно показать, что число внутренних узлов КМОП-вентиль всегда на единицу меньше числа пар его транзисторов. Поэтому каждой вершине SP-BDD, кроме корневой, можно поставить в соответствие определенный внутренний узел вентиля. Если вершина – 1-потомок другой вершины, она соответствует внутреннему узлу верхней цепи, если 0-потомок – узлу нижней цепи. Выходной узел вентиля не имеет соответствующей вершины в вентильной BDD. Как для верхней, так и для нижней цепи под соответствующим узлом понимаем узел, к которому транзистор вершины верхне-присоединен (узел подключен к истоку транзистора). Физически это означает, что в проводящем состоянии транзистор распространяет потенциал соответствующего узла на другой терминал своего канала.

Таким образом, можно перейти от “традиционного” представления схемы в виде графа с вершинами-узлами и ребрами-транзисторами к графу, в котором вершины – пары транзисторов, но, кроме того, вершины однозначно ассоциированы с узлами, – что очень удобно для моделирования схемы.

Задача ключевого моделирования заключается в разбиении всех узлов вентиля после каждого переключения на подграфы A, C и CS-группы. Далее новые потенциалы всех узлов легко вычисляются по их старым значениям. Опишем алгоритм такого разбиения вершин вентильной BDD, а также выходного узла (следовательно, и соответствующих узлов схемы).

Алгоритм разбиения узлов вентильной BDD

```

/* структура данных для вершины SP_BDD */
struct vertex {
    struct g_vertex          /* вершина вентильной SP-BDD */
        struct g_vertex     /* указатель на 0-потомка */
            *low,           /* указатель на 1-потомка */
            *high;         /* индекс вершины */
    int index;              /* указатель на входной узел вентиля */
    struct node *input;     /* групповой индекс для внутреннего узла верхней цепи */
    int pU_group,          /* групповой индекс для внутреннего узла нижней цепи */
        pD_group;         /* другие поля */
    ...
};
/* данные для SP-BDD в целом */
struct vertex *root;      /* указатель на корневую вершину */
int out_pU_group;        /* групповой индекс для выходного узла по отношению к верхней цепи */
int out_pD_group;       /* групповой индекс для выходного узла по отношению к нижней цепи */

/* алгоритм разбиения */
root->pU_group = 1;      /* отнести корневую вершину к группе 1 */
root->pD_group = 1;
for (каждой вершины SP-BDD, в линейном порядке) { /* распространение значения групповых индексов */
    присвоить значение pU_group 0-потомку /* служебная информация */;
    присвоить значение pD_group 1-потомку /* служебная информация */;
    if (верхний транзистор вершины проводит) /* соответственно, нижний не проводит */
    {
        присвоить 1-потомку текущее значение pU_group;
        присвоить 0-потомку новое значение pD_group;
    }
    else {
        присвоить 1-потомку новое значение pU_group;
        присвоить 0-потомку текущее значение pD_group;
    }
}
/* Замечание: если ссылка на 1-потомка (0-потомка) имеет значение NULL,
то значение групповых индексов присваивается выходному узлу */

```



Все узлы вентиля разбиваются на группы узлов, связанные между собой через проводящие (после очередного переключения входов вентиля) транзисторы. Группы индексируются, и каждому узлу – следовательно, и каждой вершине SP-BDD присваивается индекс pU_group и pD_group (соответственно для транзистора верхней и нижней цепи). Корневой вершине присваивается индекс $pU_group = 1$, что означает группу узлов, dc-связанных с узлом питания VDD ($pD_group = 1$ означает связь с узлом GND). Причем индексы pU_group для 0-потомков, т.е. вершин, соответствующих узлам нижней цепи (и pD_group для 1-потомков), используются не для разбиения на подграфы A, C и CS-группы, а как служебная информация о структуре цепи, показывающая, например, к какой pU -группе верхне-присоединен p -МОП транзистор вершины, ассоциированной с узлом нижней цепи. Далее считается, что:

- все внутренние узлы верхней цепи с $pU_group=1$ относятся к подграфу A;
- выходной узел относится к подграфу A, если его групповой индекс $out_pU_group=1$, в противном случае он относится к подграфу C;
- все внутренние узлы верхней цепи с $pU_group=out_pU_group$ относятся к тому же подграфу, к которому отнесен выходной узел;
- все остальные внутренние узлы верхней цепи, имеющие одинаковое значение параметра pU_group , относятся к одной и той же CS-группе;
- то же справедливо для внутренних узлов нижней цепи, если произвести следующие замены: $pU_group \rightarrow pD_group$, $out_pU_group \rightarrow out_pD_group$, подграфы $A \leftrightarrow C$ (выходной узел

рассматривается алгоритмом с точки зрения верхней и нижней цепи).

Алгоритм использует только один линейный обход вентиляльной BDD и работает намного быстрее описанного выше (предполагая, что предварительно произведена экстракция SP-BDD представления КМОП-схемы). Псевдокод данного алгоритма приведен во врезке (параметры pU_group , pD_group проинициализированы нулем).

АЛГОРИТМ РАСЧЕТА ЗАДЕРЖЕК С ИСПОЛЬЗОВАНИЕМ SP-BDD

Расчет задержек КМОП-вентилей – важная часть алгоритма ключевого моделирования, ориентированного на расчет мощности.

Рассмотрим верхнюю цепь статического КМОП-вентиля как транзисторную последовательно-параллельную цепь (ПП-цепь). По определению, ПП-цепь – это *единичный транзистор либо последовательное или параллельное соединение ПП-цепей*. Потенциал одного из ее терминалов – V_{dd} (исток), другой терминал (выход) до переключения имеет нулевой потенциал (ПП-цепь находится в непроводящем состоянии). Допустим, что некоторый транзистор x переключается в проводящее состояние, и это приводит к переключению в проводящее состояние ПП-цепи. Пусть транзистор x присоединен к узлам M и N (контактами истока и стока), причем M предшествует N. Для оценки задержки переключения узла выхода будем использовать задержку Элмора [14] для наихудшего случая, определяемую формулой:

$$d = (R_x + r_x)C_x + A_{N(x)}, \quad (8)$$

где $R_x = \max_{\{P\}} (\sum r_i)$ – максимальное предшествующее сопротив-

ление, r_i – сопротивление проводящего i -го транзистора, $\{P\}$ – множество несамопересекающихся путей от узла VDD к транзистору x ; r_x – сопротивление проводящего транзистора x ; $N(x)$ – узел, к которому транзистор x ниже-присоединен; C_x – максимальная емкость, перезаряжаемая при переключении транзистора x в проводящее состояние при заданном проводящем пути P (то же C_j); $A_{M(x)}$ – RC-сумма для узла $N(x)$,

$$A_{M(x)} = \max_{\{Q\}} [\sum (r_j C_j)], \quad (9)$$

$\{Q\}$ – множество путей от узла $N(x)$ к узлу выхода.

Покажем, что максимальная перезаряжаемая емкость C_x не зависит от проводящего пути P . C_x – это сумма емкостей всех тех узлов, которых можно достичь из $N(x)$, не пересекая P . Этими узлами являются все узлы ПП-цепи кроме узлов пути P и узлов всех путей, соединенных параллельно с подпутями пути P . Следовательно, это все узлы ПП-цепи кроме узлов всех возможных вариантов пути P . Следовательно, C_x не зависит от конкретного пути P .

Таким образом, C_x зависит только от x , и для каждого транзистора x можно вычислить максимальные предшествующее сопротивление R_x и перезаряжаемую емкость C_x независимо. Аналогично, при вычислении $A_{M(x)}$ можно вычислить C_j без учета части пути Q , соединяющей $N(x)$ и j -ый транзистор. Фактически C_x – это сумма емкостей всех узлов ПП-цепи (включая выходной узел) минус сумма емкостей узлов ПП-цепи, к которой x выше-присоединен. Действительно, когда транзистор открывается, перезаряжаются “ниже-расположенные” емкости. Поэтому если транзистор x выше-присоединен к истоку ПП-цепи, C_x равна емкости узлов всей ПП-цепи. Если x ниже-присоединен к выходу ПП-цепи, C_x равна емкости выходного узла. Естественно, для транзисторов, выше-присоединенных к какому-либо узлу, максимальная перезаряжаемая емкость C_i одинакова.

Алгоритм вычисления задержки использует представление ПП-цепи в виде SP-BDD. Он одновременно вычисляет задержку переключения для каждого транзистора. Причем применительно к КМОП-вентилю одновременно производится обработка как верхней, так и нижней цепи (т.е. вычисляются задержки как для переключения вентиля $0 \rightarrow 1$, так и для переключения $1 \rightarrow 0$). Алгоритм выполняется за четыре прохода SP-BDD (для простоты рассмотрим только верхнюю цепь):

- инициализация и подготовка обратного прохода SP-BDD (от узла выхода к VDD);
- вычисление узловых емкостей и максимальных предшествующих сопротивлений;
- вычисление максимальных переключаемых емкостей C_x ;
- вычисление RC-сумм и задержек (обратный проход SP-BDD).

Емкость узла равна сумме емкостей транзисторов, непосредственно присоединенных к этому узлу. Емкости транзисторов для данного алгоритма считаются заданными.

Максимальные предшествующие сопротивления вычисляются по алгоритму Беллмана-Форда для направленного ациклического графа: $R(N) = \max_{\{x\}} (R(M_x) + r_x)$, где $R(N)$ – максимальное предшествующее сопротивление для узла N ; $\{x\}$ – множество транзисторов, ниже-присоединенных к узлу N ; r_x – сопротивление проводящего транзистора x ; M_x – узел, к которому транзистор x выше-присоединен.

Максимальные переключаемые емкости вычисляются при прямом проходе SP-BDD. Для описания алгоритма ее вычисления введем понятие фрагмента ПП-цепи. *Фрагмент ПП-цепи – это ПП-цепь, представляющая собой либо единичный транзистор, либо несколько параллельно соединенных ПП-цепей, к которым ничего*

более не присоединено параллельно. Максимальный фрагмент ПП-цепи – это фрагмент ПП-цепи, включающий максимально возможное число элементов (скажем, если транзистор x ниже-присоединен к узлу N и параллельно ему подключена цепочка из последовательных транзисторов y и z , то фрагментом ПП-цепи будет и транзистор x , и вся цепь из трех транзисторов, а максимальным фрагментом – только цепь из трех транзисторов). Если транзистор x выше-присоединен к узлу N , то $C_x = C_y - \sum_{\{M\}} C(M) - C(N)$, где $\{M\}$ – множество внутренних узлов максимального фрагмента ПП-цепи, ниже-присоединенного к узлу N ; C_y – максимальная переключаемая емкость для любого транзистора y , выше-присоединенного к узлу истока этого максимального фрагмента; $C(M)$ – емкость узла N .

Вычисление RC-сумм $A_{M(x)}$ происходит при обратном проходе графа. Если узел M не является узлом выхода ПП-цепи, то для него RC-сумма $A_M = \max_{\{y\}} (A_{M(y)} + r_y C_y)$, где $\{y\}$ – множество транзисторов, выше-присоединенных к M ; $M(y)$ – узел, к которому y ниже-присоединен; r_y – сопротивление проводящего транзистора y ; C_y – максимальная переключаемая емкость для y . После того, как RC-сумма вычислена, по формуле (8) в том же проходе определяется задержка каждого транзистора.

Еще раз подчеркнем, что для одновременного вычисления всех задержек требуется четыре прохода SP-BDD. Данный алгоритм расчета задержек при той же точности обладает значительно большим быстродействием, чем аналогичный алгоритм, непосредственно использующий схему соединений транзисторов.

РАСЧЕТ ЭНЕРГИИ СКВОЗНЫХ ТОКОВ

При вычислении SC-энергии обрабатывается отдельно каждое переключение входа вентиля. Нетрудно показать, что в соответствии с определением SC-энергия равна нулю, когда выход вентиля не переключается. Если переключение входа вентиля приводит к переключению выхода, то величина Q в формуле для SC-энергии – это полный заряд, протекший через транзистор, переключающийся в непроводящее состояние.

Если пренебречь емкостями всех внутренних узлов вентиля и учесть только емкость выходного узла, то необходимо просто определить средние (по времени переключения) сопротивления верхней и нижней цепей. Они вычисляются также посредством SP-BDD-модели за один проход вентильной BDD.

После этого SC-энергия одного вентиля вычисляется по формуле $E_{SC} = (d - \tau + \tau e^{-d/\tau}) \cdot K \tau V_{dd}^2 / \tau_1 R_1$, где d – длительность фронта переключения входа (полагается равной задержке переключения предыдущего вентиля). Если выход переключается из 0 в 1, то $R_1 = R_n$ и $\tau_1 = \tau_p$, в противном случае $R_1 = R_p$ и $\tau_1 = \tau_n$, где R_n – R_p – средние сопротивления верхней и нижней цепей, C – емкость выходного узла, R_p и R_n – средние сопротивления верхней и нижней цепей, K – эмпирический параметр алгоритма.

Данную формулу можно получить, рассматривая SC-энергию КМОП-инвертора.

ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ

Описанный алгоритм ключевого моделирования с расчетом потребляемой мощности был реализован и протестирован в сравнении с известной программой PowerMill, которая также производит достаточно быстрое моделирование КМОП-схемы и расчет потребляемой мощности, однако использует более сложную модель транзистора. Для всех тестов использовалась сгенерированная случайным образом последовательность входных векторов длиной 2000.

Результаты тестирования алгоритма в сравнении с программой моделирования PowerMill

Схема	Число транзисторов	Средний ток, мА		Время счета, с	
		Наш алгоритм	PowerMill	Наш алгоритм	PowerMill
c432	804	0,129	0,126	6,3	116
c1355	2308	0,400	0,406	28,6	452
c1908	3482	0,599	0,601	46,6	614
cla	1008	0,185	0,175	6,8	123
cla1	956	0,226	0,227	8,1	136
cla2	956	0,184	0,180	8,0	135
cnt_0	352	0,320	0,030	2,9	58
cnt_1	372	0,340	0,033	3,4	65
cnt_ones	350	0,280	0,028	3,8	56
cnt_zeros	336	0,280	0,029	3,4	53
newckt	696	0,500	0,055	4,0	42
testsckt	1138	0,119	0,121	6,1	65

Результаты (см. табл.) свидетельствуют об очень хороших точности и быстродействии предлагаемого алгоритма. Разница в среднем токе питания между алгоритмом и программой PowerMill не превышает несколько процентов, однако данный алгоритм много быстрее – в среднем в 15–20 раз. Это делает его пригодным как для быстрого расчета мощности, потребляемой большими КМОП- схемами, так и для использования внутри оптимизационного цикла.

ЛИТЕРАТУРА

1. **A.L.Glebov, D.Blaauw, L.G.Jones.** Transistor Reordering for Low Power CMOS Gates Using an SP-BDD Representation. – Intern. Symp. on Low Power Design, 1995, p.161.

2. **A.P.Chandrakasan, S.Sheng, R.W.Brodersen.** Low-Power CMOS Digital Design. – IEEE Journal of Solid-State Circuits, 1992, v.27, № 4, p.473.

3. **Глебов А.Л., Степковский А.Л.** Оптимизация низкоомных цифровых КМОП-схем. – Автоматизация проектирования, 1997, №3, с.11.

4. **В.Kick.** Timing Correction in Logic Synthesis. – Proc. of 1st Int. Conf. "VLSI and Computers", 1987, p.299.

5. **C.Y.Tsui, M.Pedram, A.M.Despain.** Technology Decomposition and Mapping Targeting Low Power Dissipation. – In: Proc. of 30th DAC, 1993, p.68.

6. **V.Tiwari, P.Ashar, S.Matic.** Technology Mapping for Low Power. – In: Proc. of 30th DAC, 1993, p.74.

7. **Y.H.Kim, S.H.Hwang, A.R.Newton.** Electrical-Logic Simulation and Its Applications. – IEEE Trans. on CAD, 1989, v.8, №1, p.8.

8. **R.A.Saleh, A.R.Newton.** Mixed-Mode Simulation, 1990.

9. **J.P.Hayes.** A Unified Switching Theory with Applications to VLSI Design. – Proc. IEEE, 1982, v.70, №10, p.1140.

10. **R.E.Bryant.** A Switch-Level Model and Simulator for MOS Digital Systems. – IEEE Trans. on Computers, 1984, v.33, №2, p.160.

11. **C.M.Huizer.** Power Dissipation Analysis of CMOS VLSI Circuits by Means of Switch-Level Simulation. – Proc. of 16th European Solid-State Circuit Conf., 1990, p.61.

12. **Z.Barzikai, D.Beece, L.M.Huisman, V.Iyengar, G.Zilberman.** SLS – a Fast Switch-Level Simulator. – IEEE Trans. on CAD, 1988, v.7, №8, p.838.

13. **V.B.Rao, T.N.Trick.** Network Partitioning and Ordering for CMOS VLSI Circuits. – IEEE Trans. on CAD, 1987, v.6, №1, p.128.

14. **J.P.Caisso, E.Cerny, N.C.Rumin.** A Recursive Technique for Computing Delays in Series-Parallel MOS Transistor Circuits. – IEEE Trans. on CAD, 1991, v.10, №5, p.589.

15. **Глебов А.Л.** SP-BDD модель цифровых КМОП-схем и ее приложения в оптимизации и моделировании. – Информационные технологии, 1997, №10.