

ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

УДК 681.32

Л.Д. Черемисинова

ПОСТРОЕНИЕ ТЕСТОВ ПОЛНОГО ПЕРЕБОРА ДЛЯ ОЦЕНКИ ЭНЕРГОПОТРЕБЛЕНИЯ ПОСЛЕДОВАТЕЛЬНОСТНЫХ СХЕМ

Рассматривается задача оценки мощности, потребляемой КМОП-схемой, путем ее моделирования на тестовой последовательности входных воздействий. Предлагается метод построения тестовых последовательностей полного перебора на основе автоматного описания последовательностных схем. Метод основан на обходе дуг реберного орграфа, соответствующего графу переходов конечного автомата, задающего последовательностную схему.

Введение

Благодаря последним достижениям в области микроэлектронной технологии стала возможной более компактная реализация устройств интегральными микросхемами при сохранении той же функциональности. Расширилась также сфера применения устройств с автономным питанием (от сравнительно простых, таких как калькуляторы, до достаточных сложных, таких как ноутбуки и сотовые телефоны), для которых важно увеличить время их работы без подзарядки. При проектировании этих устройств на первый план выходит проблема минимизации энергопотребления [1].

В настоящей статье рассматривается проблема оценки энергопотребления интегральных КМОП-микросхем, которые составляют подавляющую часть (до 95 %) всей логики заказных устройств [2]. Потребляемая мощность в КМОП-схемах состоит из трех компонентов [3]: динамического и статического энергопотребления, мощности тока короткого замыкания. При использовании надлежащих схемных решений последние два компонента можно считать незначительными по сравнению с динамическим энергопотреблением, которое является доминирующим источником энергопотребления в КМОП-схемах, где на него приходится около 75 % общего энергопотребления.

Основной метод сокращения динамического энергопотребления на уровне логического проектирования состоит в минимизации интенсивности переключений сигналов на полюсах КМОП-микросхем за счет построения удачной логической структуры [4]. Необходимыми составляющими энергосберегающего синтеза являются программы оценки прогнозируемого энергопотребления схем на разных этапах их проектирования, в том числе и на более высоких уровнях (логическом и алгоритмическом). Наличие средств прогнозной оценки дает возможность выбрать наиболее подходящие варианты оптимизации и избежать процедуры перепроектирования схемы.

В основе подхода к оценке энергопотребления лежит моделирование исследуемой схемы [5]. Для его выполнения задается последовательность тестовых воздействий (упорядоченных последовательностей наборов значений сигналов на ее входах), которая должна позволить оценить энергопотребление схемы в заданном режиме ее работы. Эта последовательность в идеале не должна включать входные воздействия, которые никогда не могут возникать во время оцениваемого режима работы схемы. В частности, при оценке среднего рассеивания мощности [6] тестовые воздействия должны соответствовать штатному (нормальному) режиму работы исследуемой схемы и покрывать по возможности представительную часть ее области функционирования.

В настоящей работе рассматривается задача построения тестовой последовательности для оценки энергопотребления последовательностной схемы для случая, когда в генерируемой тестовой последовательности переключения всех входных сигналов считаются равновероятными, что в общем случае не имеет места при функционировании схемы в нормальном режиме ее

использования. Между тем когда такие вероятностные оценки неизвестны (или подсчитать их сложно), наилучшим выходом при оценке энергопотребления является построение тестов полного перебора возможных входных воздействий [7]. Предлагается метод генерации тестовой последовательности полного перебора на основе автоматного описания последовательностной схемы. Это описание задает поведение схемы в нормальном режиме ее использования путем задания только тех состояний, которые возможны в процессе ее функционирования. Соответственно, если сгенерировать тестовую последовательность полного перебора на основе автоматного описания, можно надеяться на получение достоверной оценки потребляемой мощности для тестируемой последовательностной схемы (при допущении одинаковой переключающей активности входных сигналов).

Полный перебор входных воздействий, сгенерированный на основе автоматного описания схемы, позволяет ограничить анализ схемы лишь теми состояниями, которые достижимы при ее функционировании. При этом структура схемы, полученная по заданному автоматному описанию, не принимается во внимание; используется только ее описание на автоматном уровне – граф переходов автомата.

1. Особенности оценки энергопотребления для схем с памятью

Большинство известных в литературе методов вычисления оценок энергопотребления предложены для случая комбинационных схем. Процедуры получения аналогичных оценок для последовательностных схем (схем с памятью) существенно отличаются от процедур, предложенных для схем без памяти. Если в первой постановке задача оценки рассеивания мощности КМОП-схемой на основе динамического анализа сводится к поиску тестовой последовательности наборов входных сигналов, которые последовательно подаются на входы схемы, и подсчету в конечном счете числа переключений значений сигналов на выходах элементов схемы, то во второй постановке, в случае последовательностных схем, приходится учитывать также и тот факт, что картина переключений сигналов в схеме существенно зависит не только от входных воздействий, но и от состояний, в которых схема находится перед их подачей. Одна и та же тестовая последовательность наборов значений входных сигналов должна использоваться для моделирования последовательностной схемы столько раз, каково число возможных состояний элементов памяти, а оно равно 2^k , где k – число элементов памяти. При этом следует учитывать и тот факт, что начальные состояния элементов памяти должны соответствовать достижимым состояниям тестируемой последовательностной схемы (состояниям, которые возможны при ее функционировании).

Тестовая последовательность наборов для последовательностной схемы задается парой «начальное состояние элементов памяти последовательностной схемы – последовательность входных воздействий на схему», что представляется набором $(s, x_1, x_2, \dots, x_l)$, где s – вектор значений внутренних переменных (соответствующих выходам элементов памяти), x_i – вектор значений входных сигналов, подаваемых на входы схемы на i -м такте ее работы. Вектор s имеет длину, равную числу триггеров в блоке памяти схемы; векторы x_i – длину, равную числу n входных полюсов схемы. Входящие в тестовую последовательность (x_1, x_2, \dots, x_l) n -разрядные векторы выбираются из булева пространства E^n мощности 2^n , где n – число входных полюсов схемы. Начальное состояние элементов памяти назначается самим проектировщиком или выбирается случайным образом из числа достижимых состояний, которые могут быть найдены и с помощью алгоритмических средств [8].

Когда условия использования устройства (характер входной последовательности сигналов) неизвестны, наиболее эффективными являются тесты (тестовые воздействия) полного перебора [7]. Для случая комбинационных схем тестовая последовательность (x_1, x_2, \dots, x_l) полного перебора включает в себя всевозможные упорядоченные пары (x_i, x_j) входных наборов из булева пространства размерности n и любая упорядоченная пара (x_i, x_j) смежных элементов представлена ровно по одному разу.

В работе [9] предложен метод построения тестовой последовательности полного перебора минимальной длины для случая комбинационных схем. Для схемы с n входными полюсами длина такой тестовой последовательности $t = 2^n(2^n - 1) + 1$ [9]. Ясно, что число тестовых наборов

t в данной последовательности быстро возрастает с увеличением числа n . Так, для $n = 5, 6, 7, 8, 9, 10, \dots$ число тестовых наборов будет $t = 993, 4033, 16\ 257, 65\ 281, 261\ 633, 1\ 047\ 553, \dots$. Недостатком такого подхода является то, что некоторые из фрагментов тестовой последовательности не будут реализовываться при нормальном режиме функционирования схемы.

Для схем с памятью проблема построения тестовой последовательности полного перебора значительно усложняется, так как помимо переключения сигналов на входах необходимо учитывать также изменение состояний элементов памяти и их достижимость в процессе функционирования последовательностной схемы.

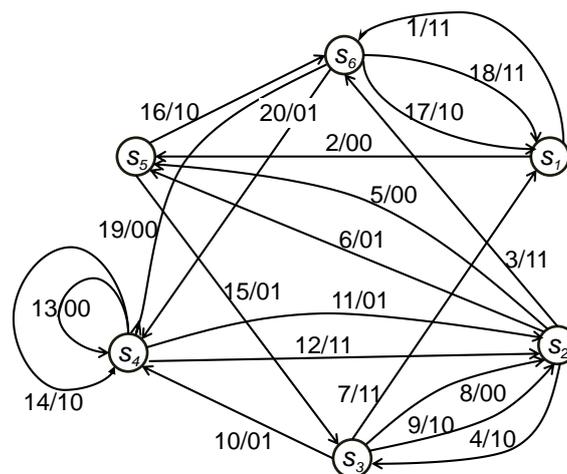
2. Основные определения и постановка задачи

Рассматриваемая задача заключается в построении тестовой последовательности полного перебора для последовательностной схемы, реализующей конечный автомат.

Конечный автомат Мили с абстрактным состоянием s , структурными входным и выходным состояниями (представляемыми векторами \mathbf{x} и \mathbf{y}) задается функциями переходов и выходов в виде $\psi(\mathbf{x}, s) = s^+$, $\varphi(\mathbf{x}, s) = \mathbf{y}$. В искомой тестовой последовательности $(s_{i0}, \mathbf{x}_{i1}, \mathbf{x}_{i2}, \dots, \mathbf{x}_{ii})$ состояние s_{i0} элементов памяти представляет собой начальное состояние, а пара $(s_{i0}, \mathbf{x}_{i1})$ выполняет инициализацию схемы и предопределяет следующее состояние s_{i1} последовательностной схемы и состояния выходных переменных \mathbf{y}_{i1} . При моделировании схемы на тестовой последовательности $(s_{i0}, \mathbf{x}_{i1}, \mathbf{x}_{i2}, \dots)$ реализуется следующая последовательность изменения внутренних и выходных состояний автомата $(s_{i0}, \mathbf{x}_{i1}, \mathbf{y}_{i1}, s_{i1}, \mathbf{x}_{i2}, \mathbf{y}_{i2}, s_{i2}, \dots)$:

$$\varphi(\mathbf{x}_{i1}, s_{i0}) = \mathbf{y}_{i1}, \psi(\mathbf{x}_{i1}, s_{i0}) = s_{i1}, \varphi(\mathbf{x}_{i2}, s_{i1}) = \mathbf{y}_{i2}, \psi(\mathbf{x}_{i2}, s_{i1}) = s_{i2}, \dots$$

Будем использовать далее графическое задание автомата, причем для целей построения тестовой последовательности достаточно рассматривать только граф переходов автомата (выходные состояния автомата для этой задачи несущественны). Граф переходов представляет собой ориентированный мультиграф, в котором могут быть петли и кратные дуги. Вершинам графа соответствуют состояния s_i автомата. Дуге, соответствующей переходу из состояния s_i в состояние s_j , приписывается входное состояние \mathbf{x}_{ij} – набор значений входных переменных $\mathbf{x} = (x_1, x_2, \dots, x_n)$, при котором происходит соответствующий переход. Предполагается, что граф переходов автомата является сильно связным, т. е. для любой пары состояний всегда найдется последовательность наборов значений входных сигналов, переводящих автомат из одного состояния в другое. На рисунке приведен пример такого графа переходов автомата, где каждая дуга (s_i, s_j) графа помечена парой «номер дуги – набор \mathbf{x}_{ij} ».



Граф переходов автомата

Энергопотребление схемы существенно зависит от числа переключений сигналов в ней. Соответственно в процессе моделирования последовательностной схемы с целью оценки ее

среднего энергопотребления анализируется реакция схемы на смену входных воздействий при каждом ее состоянии, поэтому следует рассмотреть все возможные упорядоченные тройки (x_{i1}, s_{i0}, x_{i2}) наборов состояний элементов памяти и значений входных переменных, допустимые при нормальном (штатном) функционировании схемы. В тестовой последовательности для схемы с памятью эти тройки будут следовать друг за другом, начиная с некоторого начального состояния s_0 : (x_{i1}, s_{i1}, x_{i2}) , (x_{i2}, s_{i2}, x_{i3}) , (x_{i3}, s_{i3}, x_{i4}) , ... При этом каждая пара наборов значений входных сигналов разделена набором состояний элементов памяти. Например, дуге 7 (см. рисунок), соответствующей переходу из состояния s_3 в состояние s_1 под воздействием входного условия $x_1 x_2$ (набор 1 1), должна непосредственно предшествовать одна из дуг 4 или 15 (соответствующих переходам, переводящим автомат в состояние s_3), т. е. в тестовой последовательности должны быть фрагменты, соответствующие парам дуг 4, 7 и 15, 7, которые реализуют переходы в s_3 и из s_3 : $(s_2, \bar{x}_2 x_1, s_3, x_1 x_2)$ и $(s_5, \bar{x}_1 x_2, s_3, x_1 x_2)$.

Тестовая последовательность полного перебора для вычисления среднего энергопотребления схемы с памятью представляет собой последовательность наборов значений входных переменных, которая удовлетворяет следующим условиям:

1. Начальным состоянием s_0 является некоторое исходное состояние (например, в которое устанавливается схема после ее сброса).
2. Соответствующая тестовая последовательность обеспечивает обход всех дуг графа переходов автомата (не обязательно по одному разу), начиная с заданной начальной вершины.
3. Тестовая последовательность включает в себя все возможные упомянутые выше тройки (x_{i1}, s_{i0}, x_{i2}) .

Существование такой тестовой последовательности обеспечивается предположением о сильной связности графа переходов автомата.

3. Графовые модели для поиска тестовой последовательности

Исходной графовой моделью для поиска тестовой последовательности является сильно связный ориентированный граф $G = (V, E)$, представляющий граф переходов автомата, где V и E – множества вершин (состояний) и дуг (переходов автомата). Граф $G = (V, E)$ является мультиграфом, так как может содержать кратные ребра и петли.

Задача состоит в построении такого кратчайшего ориентированного маршрута (чередующейся последовательности вершин и дуг) на орграфе $G = (V, E)$, который:

- проходит через каждую дугу графа по крайней мере один раз (в общем случае не один раз);
- обеспечивает выполнение вышеприведенного условия 3.

В данной постановке задача построения кратчайшего ориентированного маршрута напоминает задачу китайского почтальона [10], в которой ищется кратчайший путь, проходящий через все дуги заданного орграфа. В самом деле, решение задачи китайского почтальона гарантирует прохождение каждой дуги графа (т. е. выполнение условия 2), однако оно не обеспечивает прохождение каждой комбинации из пары дуг, следующих друг за другом (выполнение условия 3). Последнее требование значительно сложнее сформулировать и учесть в рамках постановки задачи китайского почтальона для графа $G = (V, E)$.

Идея предлагаемого в статье решения данной проблемы состоит в использовании другой графовой модели графа $G = (V, E)$ переходов автомата, которая позволяет легко манипулировать парами дуг графа G . Такой графовой моделью является реберный орграф [11] для графа $G = (V, E)$. Ориентированному графу G соответствует также ориентированный реберный граф $L(G)$. Вершины графа $L(G)$ соответствуют дугам графа, и $L(G)$ задает отношение смежности на множестве дуг графа G . Две вершины орграфа $L(G)$, отображающие дуги из v_i в v_j и из v_k в v_l в $G = (V, E)$ (вершины $v_i, v_j, v_k, v_l \in V$, дуги $v_i v_j, v_k v_l \in E$), связаны дугой из $v_i v_j$ в $v_k v_l$ в $L(G)$, если $v_j = v_k$. Получается, что каждая дуга в реберном орграфе $L(G)$ соответствует пути длиной 2 в исходном графе G , а значит, и упорядоченной паре переходов автомата, порождающей тройку (x_{i1}, s_{i0}, x_{i2}) наборов состояний элементов памяти и значений входных переменных, допустимых при нормальном функционировании схемы.

Таким образом, если в реберном орграфе $L(G)$ найти маршрут, проходящий через все его дуги по крайней мере по разу, то условие 3, предъявляемое к тестовой последовательности полного перебора, которая формируется по графу $G = (V, E)$ переходов автомата, будет выполнено.

Далее за начало маршрута будем принимать одну из дуг, исходящих из вершины графа, соответствующей начальному состоянию s_1 автомата, в котором производится сброс элементов памяти. Искомая тестовая последовательность $(s_1, x_1^t, x_2^t, x_3^t, \dots)$ будет состоять из входных наборов, приписанных проходимым в найденном маршруте вершинам реберного орграфа $L(G)$ (или соответствующим дугам орграфа $G = (V, E)$).

Задача состоит в поиске наиболее короткого из маршрутов. Ясно, что минимум длины маршрута достигается в том случае, если каждая дуга проходится ровно один раз. Такое решение может иметь место для частного случая орграфов – эйлеровых графов, в остальных случаях (коих, как известно из теории графов, подавляющее большинство) цель состоит в минимизации числа повторных проходов дуг графа.

4. Метод поиска тестовой последовательности

Приведенный на рисунке оргграф $G = (V, E)$ состоит из 20 дуг, значит, его реберный оргграф $L(G)$ имеет 20 вершин. Матрица R смежности орграфа $L(G)$ представляется как

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	d^+
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	4
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	2
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	4
4	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	4
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	2
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	2
7	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
8	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
8	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
10	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	4
11	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
12	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
13	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	2
14	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	2
15	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	4
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	4
17	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
18	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
19	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	4
20	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	4
d^-	3	3	4	4	4	4	2	2	2	2	5	5	3	3	3	3	3	3	3	3	3

Строкам и столбцам матрицы R соответствуют вершины орграфа $L(G)$ (или дуги орграфа G). Элемент матрицы $r_i^j \in R$ равен единице, если из i -й вершины исходит дуга, входящая в j -ю вершину (что означает наличие упорядоченной пары дуг i и j графа G). В последнем столбце и строке матрицы указаны полустепени исхода d^+ и захода d^- вершин орграфа $L(G)$. Из значений этих полустепеней следует, что граф R не является эйлеровым (и полуэйлеровым) и искомый маршрут будет содержать повторяющиеся дуги. В ходе поиска тестовой последовательности полного перебора минимальной длины необходимо минимизировать число таких повторов.

Решаемая задача схожа с классической задачей китайского почтальона и ее вариацией для орграфов – задачей нью-йоркского подметальщика [12]. Некоторые отличия заключаются только в том, что не обязательно получать именно цикл, и в том, что веса всех дуг равны. Известно, что обе эти задачи NP-трудны, поэтому нахождение точного решения (с минимумом

числа возможных повторных проходов дуг графа $L(G)$) не представляется возможным в общем случае. О сложности решаемой задачи говорит также и то, что каждая вершина степени k графа $G = (V, E)$ порождает $k(k-1)/2$ дуг реберного графа $L(G)$. Из этого факта следует, что «густым» графам G соответствуют достаточно сложные реберные графы $L(G)$.

Для задач практической размерности целесообразно использовать приближенные методы построения искомого маршрута в орграфе $L(G)$. Будем строить искомый маршрут, начиная с начальной вершины u_k (по договоренности начальное значение $k = 1$) графа $L(G)$, путем выбора на каждом шаге одной из исходящих из u_k дуг, а затем дуги из множества дуг, исходящих из последней выбранной вершины, и т. д. Перед началом поиска маршрута заводится копия C матрицы R для того, чтобы иметь возможность отследить уже пройденные вершины и пройти их повторно. Введем последовательность вершин D , включаемых в формируемый маршрут (сначала она содержит одну – начальную – вершину u_k).

Предлагаемый приближенный метод построения кратчайшего маршрута обхода орграфа $L(G)$ состоит из следующих шагов:

1. Вектору s_k присваивается значение строки $c_k \in C$ или, если $c_k = \mathbf{0}$ (0-вектор), значение строки $r_k \in R$.

2. Из множества вершин графа, помеченных единицами в s_k (и достижимых из вершины u_k), выбирается l -я вершина, строка c_l для которой имеет наибольший вес (число единиц). Если для всех порождаемых строкой s_k вершин c_l имеем $c_l = \mathbf{0}$, то в качестве l -й вершины выбирается та, которой соответствует строка $r_l \in R$ с наибольшим весом (что соответствует повторному проходу дуги).

3. Если на шаге 2 имеется несколько равнозначных вариантов выбора вершины u_l , то для каждой из них находится объединение (дизъюнкция) строк $c_i \in C$, отмеченных единицами в c_l (или r_l , если $c_l = \mathbf{0}$), и выбор l -й вершины производится по весу результата.

4. После выбора l -й вершины элемент $c_k^l \in C$ обнуляется, номер l вносится в последовательность D пройденных вершин, номеру k присваивается значение l . Если после этого в матрице C остаются единичные элементы, процедура построения маршрута продолжается с шага 1. В противном случае искомый маршрут найден.

Номера вершин графа $L(G)$, перечисленные в последовательности D , соответствуют номерам дуг графа $G = (V, E)$ (переходов автомата), которым приписаны наборы значений входных переменных, порождающие искомую тестовую последовательность.

Для приведенного выше примера реберного орграфа $L(G)$ на первом шаге рассматривается первая вершина u_1 , смежная вершинам u_{17} , u_{18} , u_{19} и u_{20} (что следует из строки $c_1 \in C$ матрицы смежности). Так как две вершины u_{19} и u_{20} равнозначны по числу и номерам исходящих из них дуг, выбирается первая из них.

Повторив процедуру построения маршрута до конца, получаем маршрут длиной 85, тогда как орграф $L(G)$ содержит 64 дуги. Следовательно, маршрут проходит повторно через 21 дугу. Начальный фрагмент полученной последовательности вершин:

$$D = \{u_1, u_{19}, u_{11}, u_3, u_{20}, u_{12}, u_4, u_8, u_3, u_{19}, u_{12}, u_5, u_{15}, u_9, u_6, u_{16}, u_{17}, u_1, u_{20}, u_{11}, u_4, u_{10}, u_{13}, \dots\}.$$

Соответствующий начальный фрагмент тестовой последовательности для оценки среднего энергопотребления схем начинается с состояния s_1 , кодируемого кодом 000 элементов памяти, и состоит из следующих наборов значений двух входных сигналов:

$$\{11, 00, 01, 11, 01, 11, 10, 00, 11, 00, 11, 00, 01, 10, 01, 10, 01, 11, 01, 01, 10, 01, 00, \dots\}.$$

Заключение

Оценка среднего значения потребляемой мощности для случая последовательностных схем может быть выполнена достаточно точно, если известно исходное автоматное описание этой схемы. Для этого случая сформулирована задача генерации тестовой последовательности полного перебора для оценки прогнозируемого энергопотребления схемы и показано, как можно построить и минимизировать по длине последовательность, покрывающую представительную часть возможной области функционирования схемы.

Список литературы

1. Kuroda, T. Low-power high-speed CMOS VLSI design / T. Kuroda // Proc. of IEEE Intern. Conf. on Computer Design. – Los Alamitos, California, 2002. – P. 310–315.
2. Zimmermann, R. Low-power logic styles: CMOS versus pass-transistor logic / R. Zimmermann, W. Fichtner // IEEE J. of Solid-State Circuits. – 1997. – Vol. 32(7), no. 7. – P. 1079–1090.
3. Chandrakasan, A.P. Low-power CMOS digital design / A.P. Chandrakasan, S. Sheng, R.W. Broderson // IEEE J. of Solid-State Circuits. – 1992. – Vol. 27(4), no. 4. – P. 473–484.
4. Estimation of average switching activity in combinational and sequential circuits / A. Ghosh [et al.] // 29th ACM/IEEE Design Automation Conference. – Anaheim, CA, 1992. – P. 253–259.
5. Benini, L. Logic Synthesis for Low Power / L. Benini, G. De Micheli // Logic Synthesis and Verification; eds. S. Hassoun, T. Sasao, R.K. Brayton. – Boston, Dordrecht, London : Kluwer Academic Publishers, 2002. – P. 197–223.
6. BPNN Based Power Estimation of Sequential Circuits / S.A.K. Arasu [et al.] // Intern. J. of Advanced Research in Computer Science and Software Engineering. – 2013. – Vol. 3, no. 11. – P. 256–260.
7. Bushnell, M.L. Essentials of electronic testing for digital, memory and mixed-signal VLSI circuits / M.L. Bushnell, V.D. Agrawal. – Kluwer Academic Publishers, 2002. – 128 p.
8. Бибило, П.Н. Оценка энергопотребления логических КМОП-схем по их переключательной активности / П.Н. Бибило, Н.А. Кириенко // Микроэлектроника. – 2011. – Т. 40, № 6. – С. 1–14.
9. Закревский, А.Д. Минимизация перебора ориентированных пар / А.Д. Закревский // Танаевские чтения : докл. Четвертой Междунар. науч. конф., Минск, 29–30 марта 2010 г. – Минск : ОИПИ НАН Беларуси, 2010. – С. 58–62.
10. Кристофидес, Н. Теория графов. Алгоритмический подход / Н. Кристофидес. – М. : Мир, 1978.
11. Харари, Ф. Теория графов / Ф. Харари ; пер. с англ. – М. : Едиториал УРСС, 2003. – 296 с.
12. Bodin, L. A Model for Municipal Street Sweeping Operations / L. Bodin // Discrete and System Models (Modules in Applied Mathematics). – 1983. – Vol. 3. – P. 76–111.

Поступила 10.10.2017

*Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: kir@newman.bas-net.by,
cld@newman.bas-net.by*

L.D. Cheremisinova

TESTS GENERATION FOR POWER CONSUMPTION ESTIMATION OF SEQUENTIAL CIRCUITS

The article reflects the problem of the average power estimation which is consumed by a CMOS circuit by means of its simulation on the test sequence of input actions. The method of forming test sequences of complete enumeration for sequential circuits is offered. The method is based on arcs traversal of the line digraph corresponding to the state transition graph of finite state machine representing the sequential circuit.