УДК 004.312.4

Расщепление внутренних состояний конечных автоматов для минимизации потребляемой мощности

В.В. Соловьев, Т.Н. Грэсь

Белостокский технологический университет (Польша), valsol@mail.ru

Аннотация **— Рассмотрен метод минимизации энергопотребления конечных автоматов путем расщепления внутренних состояний. Предложены два эвристических алгоритма уменьшения энергопотребления конечных автоматов за счет расщепления внутренних состояний с большой и малой вычислительной сложностью, которые оказались близкими по эффективности. Результаты экспериментальных исследований показали, что предложенный подход позволяет снизить энергопотребление конечных автоматов в 73% случаев, при этом уменьшение потребляемой мощности составляет, в среднем, 7,25%, а для отдельных примеров – 81%.**

Ключевые слова **— конечный автомат, потребляемая мощность, энергопотребление, расщепление внутренних состояний.**

I. Введение

Широкое распространение переносных и бортовых цифровых систем предъявляет очень жесткие требования к энергопотреблению (потребляемой мощности) разрабатываемых проектов. В общем случае цифровую систему можно представить как совокупность взаимодействующих между собой комбинационных схем и конечных автоматов, поэтому одним из путей решения указанной проблемы является минимизация потребляемой мощности конечных автоматов.

Известно много различных подходов для снижения потребляемой мощности конечных автоматов: путем кодирования внутренних состояний [1]-[5], декомпозиции конечного автомата [6], стробирования синхросигнала [7], выбора типов триггеров элементов памяти [8], изменения числа разрядов кода внутренних состояний [9], использования специальных структурных моделей конечных автоматов [10] и др.

В настоящей работе для уменьшения энергопотребления конечных автоматов предлагается использовать операцию расщепления внутренних состояний.

II. Определение энергопотребления конечного автомата

Пусть $A = \{a_1,\dots,a_M\}$ – множество внутренних состояний конечного автомата, $X = \{x_1,\dots,x_L\}$ – множество входных переменных, $Y = \{y_1,\dots,y_N\}$ – множество выходных переменных, $D = \{d_1,\dots,d_R\}$ – множество функций возбуждения элементов памяти, где R – число элементов памяти (число разрядов кодов внутренних состояний), $R \in [\mathrm{intlog}_2 M, M]$.

Для определения энергопотребления конечного автомата воспользуемся методикой работы [11], которая позволяет определить динамическую мощность конечного автомата на основании результатов кодирования внутренних состояний и вероятности появления единицы (нуля) на каждом входе конечного автомата.

Значение потребляемой мощности конечного автомата определяется из выражения:

$$P = \sum_{r=1}^{R} P_r = \frac{1}{2} \times V_{DD}^2 \times f \times C \times \sum_{r=1}^{R} N_r \,, \qquad (1)$$

где P_r – мощность, потребляемая триггером r; V_{DD} – величина напряжения питания; f – частота функционирования конечного автомата; C – ёмкость выхода каждого триггера; N_r – переключательная активность триггера r, $r = \overline{1, R}$.

Пусть k_i – некоторый бинарный код состояния $a_i \in A$. Обозначим через k_i^r значение бита r в коде k_i состояния a_i, $r = \overline{1, R}$. Тогда активность N_r переключения триггера r памяти конечного автомата можно выразить следующим образом:

$$N_r = \sum_{m=1}^{M} \sum_{s=1}^{M} P(a_m \to a_s) \times (k_m^r \oplus k_s^r) \,, \qquad (2)$$

где $P(a_m \to a_s)$ – вероятность перехода конечного автомата из состояния a_m в состояние a_s (a_m, $a_s \in A$); $\oplus$ - логическая операция Исключающее ИЛИ.

Вероятность $P(a_m \to a_s)$ перехода конечного автомата из состояния a_m в состояние a_s (a_m, $a_s \in A$) определяется выражением:

$$P(a_m \to a_s) = P(a_m) \times P(X(a_m, a_s)) \,, \qquad (3)$$

где $P(a_m)$ – вероятность нахождения конечного автомата в состоянии a_m; $P(X(a_m,a_s))$ – вероятность появления на входе конечного автомата входного вектора $X(a_m,a_s)$, который инициирует переход из состояния a_m в состояние a_s.

Вероятность $P(X(a_m,a_s))$ появления на входе конечного автомата входного вектора $X(a_m,a_s)$ определяется из выражения:

$$P(X(a_m, a_s)) = \prod_{b=1}^{L} P(x_b = d), \qquad (4)$$

где $d \in \{0,1,\text{'-'}\}$; $P(x_b=d)$ – вероятность того, что входная переменная x_b во входном векторе $X(a_m,a_s)$ примет значение d. В данной работе принято, что вероятность появления 0 или 1 на каждом входе конечного автомата одинакова, поэтому $P(x_b=0) = P(x_b=1) = 0{,}5$, а $P(x_b=\text{'-'}) = 1$.

Вероятность $P(a_i)$ нахождения конечного автомата в каждом состоянии a_i, $i=\overline{1,M}$ можно определить в результате решения следующей системы уравнений:

$$P(a_i) = \sum_{m=1}^{M} P(a_m) \times P(X(a_m, a_s)), i = \overline{1,M}. \qquad (5)$$

В случае, когда переход между состояниями a_m и a_i отсутствует, принимается $P(X(a_m,a_i)) = 0$. В случае же, когда из состояния a_m в состояние a_i ведут несколько переходов, значение $P(X(a_m,a_i))$ определяется как сумма вероятностей появления каждого входного вектора, который инициирует переход из состояния a_m в состояние a_i.

Система уравнений (5) представляет собой линейную систему M уравнений от M неизвестных $P(a_1),\ldots,P(a_M)$, для решения которой может быть применен любой из известных методов, например, Гаусса. Поскольку конечный автомат всегда находится в одном из своих внутренних состояний, справедливым является следующее равенство:

$$\sum_{m=1}^{M} P(a_m) = 1. \qquad (6)$$

Для решения системы уравнений (5) одно из уравнений в (5) заменяется равенством (6).

С учетом вышеизложенного алгоритм определения оценки энергопотребления (потребляемой мощности) конечного автомата выглядит следующим образом.

Алгоритм 1 (определения оценки энергопотребления).

1. Согласно (4) для каждого входного вектора $X(a_m,a_s)$ $(a_m,\ a_s \in A)$ определяется вероятность $P(X(a_m,a_s))$ его появления на входе конечного автомата.
2. Решается система уравнений (5) для определения вероятностей $P(a_i)$ нахождения конечного автомата в каждом состоянии a_i, $a_i \in A$.
3. Согласно (3) определяются вероятности переходов $P(a_m \to a_s)$ конечного автомата, a_m, $a_s \in A$.
4. На основании результатов кодирования внутренних состояний согласно (2) определяется активность каждого триггера N_r, $r=\overline{1,R}$.
5. Согласно (1) вычисляется потребляемая мощность P конечного автомата для следующих значений параметров: VDD = 5V, f = 10MHz и C = 5pF (типичные значения для большинства микросхем CMOS-технологии).
6. Конец.

III. Суть предлагаемого подхода

В основу предлагаемого метода положена следующая гипотеза: расщепление внутренних состояний конечного автомата может приводить к снижению энергопотребления конечного автомата. На первый взгляд данное предположение может показаться абсурдным, поскольку в результате расщепления внутренних состояний увеличивается число внутренних состояний конечного автомата, что может привести даже к увеличению числа R разрядов, необходимых для кодирования внутренних состояний. Однако расщепление внутренних состояний позволяет уменьшить связанность состояний в графе автомата, что упрощает решение задачи кодирования внутренних состояний с целью минимизации энергопотребления.

Анализ выражения (2) показывает, что активность N_r некоторого триггера r, $r=\overline{1,R}$, может быть изменена как за счет уменьшения вероятностей переходов между состояниями $P(a_m \to a_s)$, a_m, $a_s \in A$, так и за счет уменьшения числа разрядов кода, по которым различаются коды состояний a_m и a_s.

Вначале проанализируем, как расщепление внутренних состояний влияет на вероятность $P(a_m \to a_s)$ перехода из состояния a_m в состояние a_s, a_m, $a_s \in A$. Значение вероятности $P(a_m \to a_s)$ определяется из выражения (3). Вероятность $P(X(a_m,a_s))$ появления на входе конечного автомата входного вектора $X(a_m,a_s)$ при расщеплении внутренних состояний не изменяется. Однако при расщеплении состояний будет увеличиваться общее число M состояний конечного автомата, а из выполнения равенства (6) следует, что при увеличении M вероятности $P(a_m)$ для отдельных состояний будут уменьшаться. С другой стороны, при увеличении числа состояний M будет увеличиваться число слагаемых в выражении (2), поэтому на уменьшение энергопотребления только за счет увеличения числа состояний M надеяться не приходиться.

Теперь проанализируем, как расщепление внутренних состояний влияет на число разрядов, по которым различаются коды состояний a_m и a_s, m, $s = \overline{1,M}$. Пусть кодирование внутренних состояний конечного автомата осуществляется бинарными кодами с числом разрядов $R = \mathrm{intlog}_2 M$. Общее число кодов, доступных для кодирования внутренних состояний, равно 2^R. На практике часто справедливым является неравенство $M < 2^R$, т.е. число возможных кодов больше, чем число внутренних состояний. При расщеплении некоторого состояния a_i, $a_i \in A$ уменьшается множество состояний $B(a_i)$, переходы из которых ведут в состояние a_i. В результате упрощается подбор кодов с целью минимизации потребляемой мощности для состояний, связанных с состоянием a_i, $a_i \in A$.

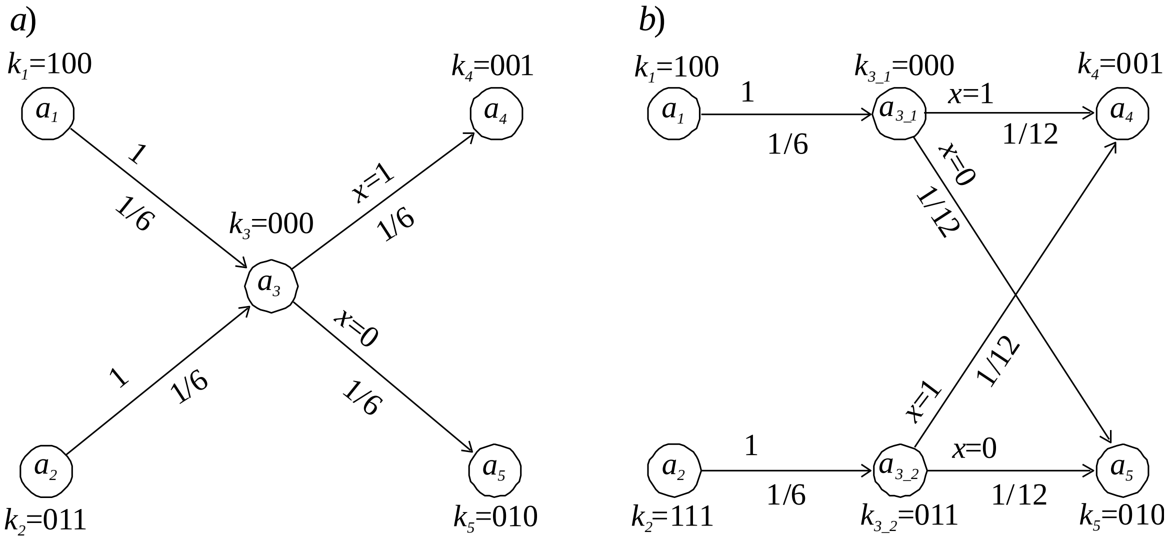

Рис. 1. Фрагмент графа автомата: *a* – до расщепления; *b* – после расщепления состояния a_3

Другими словами, в результате расщепления некоторого состояния a_i ($a_i \in A$) для переходов в состояние a_i и из состояния a_i с большими вероятностями вида $P(a_m \rightarrow a_i)$ и $P(a_i \rightarrow a_s)$ (a_m, $a_s \in A$) становится возможным подобрать коды, которые различаются по меньшему числу разрядов.

В случае, когда $M = 2^R$, расщепление внутренних состояний требует увеличения на единицу числа R разрядов кода, что приводит к увеличению в два раза числа доступных кодов для кодирования состояний. В результате значительно увеличиваются возможности алгоритмов кодирования внутренних состояний с целью уменьшения потребляемой мощности [5].

Таким образом, расщепление внутренних состояний может приводить к снижению потребляемой мощности как за счет уменьшения вероятности нахождения конечного автомата $P(a_m)$ для отдельных состояний, так и за счет уменьшения числа различных значений разрядов кода для состояний с большой вероятностью переходов $P(a_m \rightarrow a_s)$.

Пример. Рассмотрим фрагмент графа конечного автомата, показанный на рис. 1a.

Для данного фрагмента переходы из состояний a_1 и a_2 в состояние a_3 являются безусловными, т.е. $P(X(a_1,a_3)) = P(X(a_2,a_3)) = 1$. Переход из состояния a_3 в состояние a_4 осуществляется при значении входной переменной $x = 1$, а в состояние a_5 – при $x = 0$, т.е. $P(X(a_3,a_4)) = P(X(a_3,a_5)) = 0{,}5$. Пусть вероятности нахождения конечного автомата в состояниях a_1 и a_2 равны, т.е. $P(a_1) = P(a_2)$. Воспользовавшись указанными вероятностями переходов и равенством (6) можно решить систему уравнений (5). В результате получим следующие вероятности нахождения конечного автомата в состояниях $a_1,\ldots, a_5$: $P(a_1) = P(a_2) = P(a_4) = P(a_5) = 1/6$ и $P(a_3) = 1/3$. Согласно (3) определяются вероятности переходов между состояниями (на рис. 1a значения вероятностей переходов записаны под соответствующей дугой): $P(a_1 \rightarrow a_3) = P(a_2 \rightarrow a_3) = P(a_3 \rightarrow a_4) = P(a_3 \rightarrow a_5) = 1/6$. Пусть выполнено следующее кодирование внутренних состояний: k_1=100, k_2=011, k_3=000, k_4=001, k_5=010 (на рис. 1a коды внутренних состояний записаны рядом с состояниями). Отметим, что для данного фрагмента графа автомата при значениях M=5 и R=3 невозможно выполнить такое кодирование внутренних состояний, чтобы значения кодов различались не более чем в одном разряде. Приняв значение постоянного множителя $\frac{1}{2} \times V_{DD}^2 \times f \times C$ равным 1, на основании (2) и (1) значение потребляемой мощности P для данного фрагмента конечного автомата равно 5/6.

Пусть выполнено расщепление состояния a_3 на два состояния a_{3_1} и a_{3_2}, как показано на рис. 1b. Принимая соглашение $P(a_1) = P(a_2)$, решение системы уравнений (5) позволяет определить вероятности нахождения конечного автомата в каждом состоянии: $P(a_1) = P(a_2) = P(a_{3\_1}) = P(a_{3\_2}) = P(a_4) = P(a_5) = 1/6$. Согласно (3) определяются вероятности переходов между состояниями: $P(a_1 \rightarrow a_{3\_1}) = P(a_2 \rightarrow a_{3\_2}) = 1/6$; $P(a_{3\_1} \rightarrow a_4) = P(a_{3\_1} \rightarrow a_5) = P(a_{3\_2} \rightarrow a_4) = P(a_{3\_2} \rightarrow a_5) = 1/12$. В отличие от фрагмента на рис. 1a, преобразованный фрагмент графа конечного автомата на рис. 1b позволяет закодировать состояния таким образом, чтобы коды соседних состояний различались не более чем в одном разряде: k_1=100, k_2=111, k_{3_1}=000, k_{3_2}=011, k_4=001, k_5=010. Поэтому приняв значение постоянного множителя $\frac{1}{2} \times V_{DD}^2 \times f \times C$ равным 1, на основании (2) и (1) значение потребляемой мощности P для данного фрагмента конечного автомата равно 4/6. Таким образом, расщепление состояния a_3 на два состояния a_{3_1} и a_{3_2} привело к уменьшению потребляемой мощности для данного фрагмента конечного автомата на 20%.

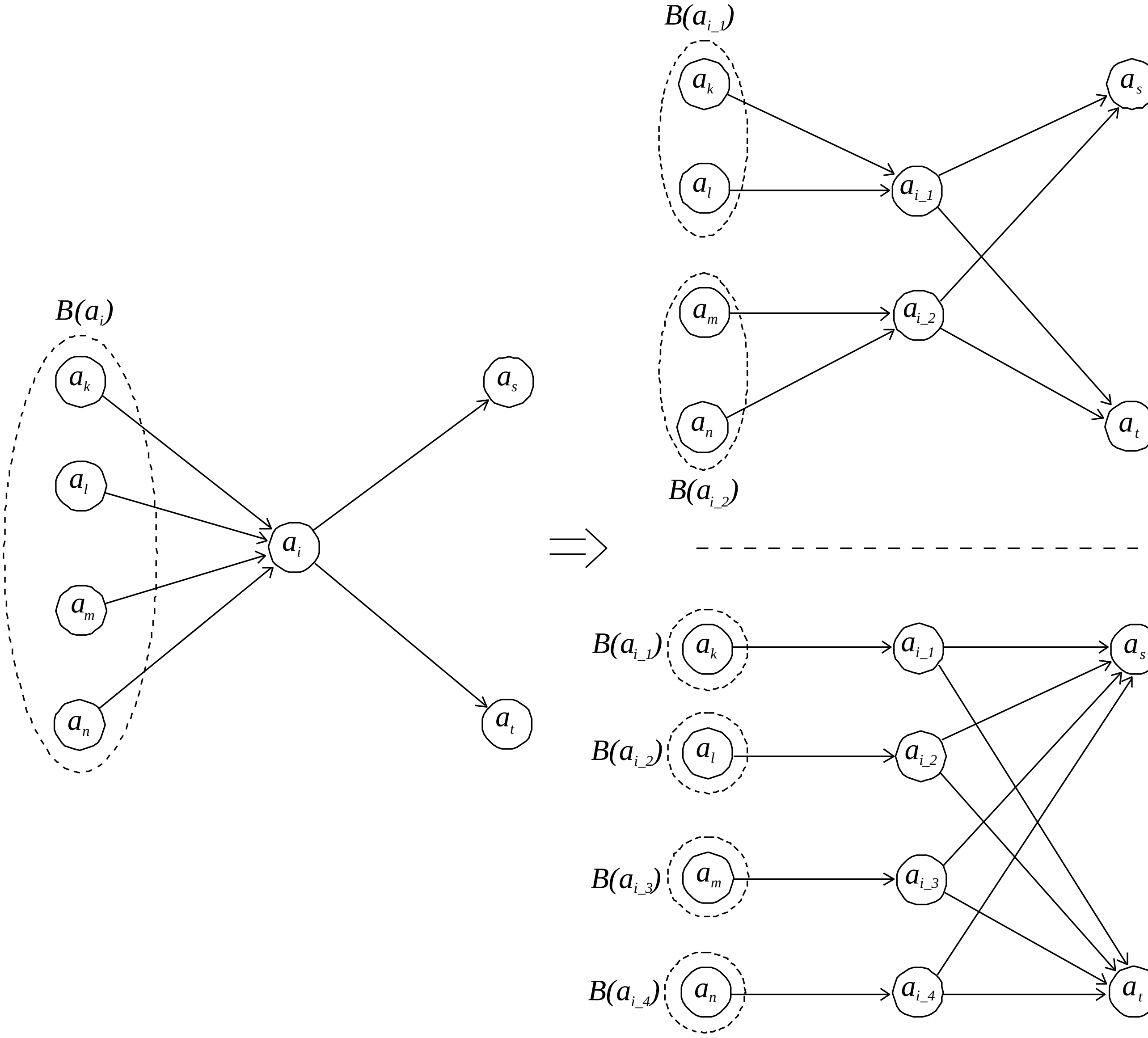

Рис. 2. Возможные варианты расщепления состояния a_i при $|B(a_i)| = 4$

IV. МЕТОД РАСЩЕПЛЕНИЯ ВНУТРЕННИХ СОСТОЯНИЙ КОНЕЧНЫХ АВТОМАТОВ С ЦЕЛЬЮ МИНИМИЗАЦИИ ПОТРЕБЛЯЕМОЙ МОЩНОСТИ

Прежде всего, следует отметить, что расщеплять можно только такие состояния, для которых $|B(a_i)| > 1$, где |A| - мощность множества A (в рассмотренном примере $|B(a_3)|=2$).

Пусть для некоторого состояния a_i ($a_i \in A$) $|B(a_i)| = H > 1$. Состояние a_i может быть расщеплено на 2,…,H состояний. В первом случае множество $B(a_i)$ разбивается на две группы состояний, а в последнем случае множество $B(a_i)$ разбивается на H групп состояний, причем каждая группа состояний состоит из одного состояния, из которого ведет один или несколько переходов во вновь образованные состояния $a_{i\_1},\ldots,a_{i\_H}$ (рис. 2). Кроме того, каждая группа состояний $B(a_{i\_h})$, $h = \overline{2,H}$ может включать любые состояния множества $B(a_i)$. Другими словами, способы расщепления состояния a_i определяются способами разбиения множества $B(a_i)$ на непересекающиеся подмножества $B(a_{i\_1}),\ldots,B(a_{i\_h})$, $h = \overline{2,H}$.

Предлагаемый ниже алгоритм позволяет рассматривать все возможные способы расщепления состояний и выбирать для расщепления такое сочетание групп состояний $B(a_{i\_1}),\ldots,B(a_{i\_h})$, $h = \overline{2,H}$, которое приводит к наибольшему уменьшению потребляемой мощности конечного автомата.

Алгоритм 2 (расщепления внутренних состояний для уменьшения энергопотребления конечных автоматов).

1. Выполняется последовательный алгоритм [5] кодирования внутренних состояний конечного автомата с целью уменьшения потребляемой мощности. С помощью алгоритма 1 определяется потребляемая мощность P конечного автомата. Полагается P* := P, P’ := P.
2. Последовательно рассматриваются внутренние состояния a_i, $a_i \in A$, для которых $|B(a_i)| = H > 1$.
3. Рассматриваются все возможные способы разбиения множества $B(a_i)$ на непересекающиеся подмножества $B(a_{i\_1}),\ldots,B(a_{i\_h})$ для всех $h = \overline{2,H}$.
4. Для некоторого разбиения $B(a_{i\_1}),\ldots,B(a_{i\_h})$, полученного в пункте 3, выполняется пробное расщепление состояния a_i на состояния $a_{i\_1},\ldots,a_{i\_h}$ таким образом, чтобы переходы из состояний каждого подмножества $B(a_{i\_j})$ вели в состояние a_{i_j}, $j = \overline{1,h}$.
5. Для вновь образованного конечного автомата выполняется последовательный алгоритм [5] кодирования внутренних состояний конечного автомата с целью уменьшения потребляемой мощности. С помощью алгоритма 1 определяется потребляемая мощность P. Выполняется возврат к конечному автомату до расщепления состояния a_i в п.4.
6. Если P < P’, то полагается P’ := P, запоминается разбиение $B(a_{i\_1}),\ldots,B(a_{i\_h})$. Иначе перейти к п.3 для рассмотрения следующего разбиения $B(a_{i\_1}),\ldots,B(a_{i\_h})$.
7. Пункты 2-6 выполняются для всех состояний a_i, $a_i \in A$.
8. Если P’ < P*, то выполняется расщепление состояния a_i на состояния $a_{i\_1},\ldots,a_{i\_h}$ в соответствии с разбиением $B(a_{i\_1}),\ldots,B(a_{i\_h})$, запомненным в п.6, далее перейти к п.1. Иначе перейти к п.9.
9. Конец.

Отметим, что для улучшения результатов кодирования внутренних состояний с целью уменьшения потребляемой мощности, после выполнения алгоритма 2 может быть выполнен итерационный алгоритм кодирования внутренних состояний [4].

Несмотря на то, что рассмотренный алгоритм 2 является эвристическим, его применение связано с большим перебором при выполнении пункта 3. Поэтому для практического использования может быть рекомендован упрощенный вариант алгоритма 2. В упрощенном варианте алгоритма 2 каждое подмножество $B(a_i)$ разбивается только на два непересекающихся подмножества $B(a_{i\_1})$ и $B(a_{i\_2})$, причем подмножество $B(a_{i\_1})$ включает только одно состояние. В качестве такого состояния в подмножество $B(a_{i\_1})$ выбирается состояние, для которого вероятность перехода в состояние a_i максимальна. С учетом вышеизложенного в упрощенном варианте алгоритма 2 пункт 3 выглядит следующим образом:

3’. Множество $B(a_i)$ разбивается на два непересекающихся подмножества $B(a_{i\_1})$ и $B(a_{i\_2})$. Для этого в множестве $B(a_i)$ находится состояние a_j такое, что $P(a_j \rightarrow a_i)$ = max; полагается $B(a_{i\_1}) := \{a_j\}$, $B(a_{i\_2}) := B(a_i) \setminus \{a_j\}$.

Назовем упрощенный таким образом вариант алгоритма 2 *алгоритмом 3*.

V. Результаты экспериментальных исследований

Для оценки эффективности предлагаемого метода использованы эталонные примеры конечных автоматов, разработанные в центре MCNC [12]. С помощью алгоритмов 2 и 3 исследовано 33 эталонных примера конечных автоматов. Для 24 примеров или в 73% случаев применение алгоритмов 2 и 3 позволило снизить энергопотребления автоматов. Результаты экспериментальных исследований для 24 примеров приведены в табл. 1, где FSM – название эталонного примера; M и Q – число внутренних состояний и число переходов исходного конечного автомата; P – потребляемая мощность исходного конечного автомата (в микроваттах); P_3 – потребляемая мощность конечного автомата после применения упрощенного алгоритма 3; $\Delta P_3\%$ – изменение мощности (в процентах) конечного автомата после применения алгоритма 3; P_2 и $\Delta P_2\%$ – аналогичные параметры для алгоритма 2; mid – среднеарифметическое значение параметра.

Таблица 1

Результаты экспериментальных исследований

FSM	**M**	**Q**	**P**	**P_3**	**$\Delta P_3\%$**	**P_2**	**$\Delta P_2\%$**
bbara	10	60	52,39	51,72	1,28	51,71	1,30
bbsse	16	56	146,47	144,48	1,36	144,48	1,36
cse	16	91	44,97	44,67	0,67	44,67	0,67
dk14	7	56	223,65	204,25	8,67	203,52	9,00
dk15	4	32	159,46	154,81	2,91	154,81	2,91
dk16	27	108	303,95	291,87	3,97	291,20	4,19
dk17	8	32	194,98	187,80	3,68	187,80	3,68
dk27	7	14	223,21	218,75	2,00	218,75	2,00
donfile	24	96	222,66	208,98	6,14	208,98	6,14
ex1	20	138	138,68	117,35	15,38	117,35	15,38
ex4	14	21	91,97	90,01	2,12	90,01	2,12
ex6	8	34	199,22	188,56	5,35	188,13	5,57
keyb	19	170	104,35	104,32	0,03	104,32	0,03
mark1	15	22	178,29	172,27	3,37	172,27	3,37
opus	10	22	133,28	132,75	0,47	132,27	0,47
planet	48	115	216,87	207,32	4,41	207,32	4,41
pma	24	73	104,41	93,06	10,87	93,06	10,87
s1	20	107	200,38	197,76	1,31	196,65	1,86
s27	6	34	168,33	165,18	1,87	165,18	1,87
s386	13	64	145,50	145,29	0,14	145,29	0,14
sand	32	184	116,29	106,20	8,68	106,20	8,68
shifreg	8	16	210,94	199,22	5,56	199,22	5,56
sse	16	56	146,47	145,50	0,66	144,48	1,36
tma	20	44	65,25	12,38	**81,02**	12,38	**81,02**
mid					**7,16**		**7,25**

Анализ табл. 1 показывает, что использование алгоритма 3 позволяет, в среднем, уменьшить энергопотребление конечных автоматов на 7,16%, а для отдельных примеров – на 81,02% (пример tma). В то же время применение алгоритма 2, по сравнению с алгоритмом 3, позволило незначительно снизить потребляемую мощность (от 0,22% до 0,70%) для 6 примеров конечных автоматов. Среднее снижение потребляемой мощности при этом составляет 7,25%. Поэтому для практического использования можно рекомендовать упрощенный алгоритм 3, который позволяет получить результаты близкие к алгоритму 2, однако требует значительно меньше количество вычислений, по сравнению с алгоритмом 2.

Таким образом, даже упрощенный вариант предложенного подхода (алгоритм 3) расщепления внутренних состояний с целью снижения потребляемой мощности конечных автоматов показал свою достаточно высокую эффективность в 73% случаев. При этом среднее снижение энергопотребления составляет 7,16%, а для отдельных примеров – 81,02%.

VI. Заключение

Рассмотрен подход снижения потребляемой мощности конечных автоматов путем использования операции расщепления внутренних состояний. Показано, что снижение энергопотребления осуществляется за счет увеличения возможностей для кодирования внутренних состояний с целью снижения потребляемой мощности. Предложены два эвристических алгоритма уменьшения энергопотребления конечных автоматов путем расщепления внутренних состояний: большой (алгоритм 2) и малой (алгоритм 3) вычислительной сложности. На основании анализа результатов экспериментальных исследований показано, что алгоритм 3 незначительно уступает алгоритму 2 по качеству получаемого решения и может быть рекомендован для практического использования.

Дальнейшее развитие данного направления уменьшения энергопотребления конечных автоматов может идти по пути использования других операций эквивалентных преобразований конечных автоматов (например, операции склеивания внутренних состояний), а также их совместного использования.

Поддержка

Работа выполнена при частичной финансовой поддержке Белостокского технологического университета (Польша), грант № S/WI/1/2013.

Литература

[1] Benini L., De Micheli G. State assignment for low power dissipation // IEEE Journal on Solid-State Circuits. 1995. V. 30. № 3. P. 259-268.

[2] Tsui C.-Y., Pedram M., Despain A.M. Low power state assignment targeting two and multilevel logic implementations // IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems. 1998. V. 17. № 12. P. 1281–1291.

[3] Shiue W.-T. Novel state minimization and state assignment in finite state machine design for low-power portable device // Integration, the VLSI Journal. 2005. V. 38. № 3. P. 549-570.

[4] Соловьев В.В., Грэсь Т.Н. Итерационный алгоритм кодирования внутренних состояний конечных автоматов с целью минимизации потребляемой мощности // Микроэлектроника. 2013. Т. 42. № 3. С. 233-240.

[5] Соловьев В.В., Грэсь Т.Н. Последовательный алгоритм кодирования внутренних состояний конечных автоматов для минимизации потребляемой мощности // Изв. Российской академии наук. Теория и системы управления. 2014. № 1. С. 87-95.

[6] Chow. S-H., Ho Y-C., Hwang T., Liu C.L. Low power realization of finite state machines – a decomposition approach // ACM Transactions on Design Automation of Electronic Systems. 1996. V. 1. № 3. P. 315-340.

[7] Benini L., De Micheli G. Automatic synthesis of low power gated clock finite state machines // IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems. 1996. V. 15. № 6. P. 630-643.

[8] Chattopadhyay S. Low power state assignment and flip-flop selection for finite state machine synthesis – a genetic algorithmic approach // IEE Proceedings – Computers and Digital Techniques. 2001. V. 148. № 45. P. 147-151.

[9] Соловьев В.В. Изменение числа разрядов кода внутренних состояний при минимизации потребляемой мощности конечных автоматов // Радиотехника и электроника. 2012. Т. 57. № 6. С. 705-712.

[10] Соловьев В.В. Энергосберегающий синтез конечных автоматов на основе совмещенной структурной модели // V Всероссийская научно-техническая конференция «Проблемы разработки перспективных микро- и наноэлектронных систем – 2012». Сб. трудов / под ред. академика РАН А.Л.Стемпковского. М.: ИППМ РАН, 2012. С. 79-82.

[11] Tsui C.-Y., Monteiro J., Devadas S., Despain A.M., Lin B. Power estimation methods for sequential logic circuits // IEEE Transactions on VLSI Systems. 1995. V. 3. № 3. P. 404-416.

[12] Yang S. Logic synthesis and optimization benchmarks user guide. Version 3.0. Technical Report. North Carolina. Microelectronics Center of North Carolina. 1991. 46 p.