

Энергосберегающий синтез конечных автоматов на основе совмещенной структурной модели

В.В. Соловьев

Высший государственный колледж связи (Минск, Беларусь), valsol@mail.ru

Аннотация — Рассматривается метод энергосберегающего синтеза конечных автоматов на основе совмещенной структурной модели ADE, который позволяет использовать триггеры входных и выходных буферов в качестве элементов памяти конечных автоматов. Метод применим при реализации конечных автоматов на программируемых логических интегральных схемах (ПЛИС) и системах на программируемом кристалле. Результаты экспериментальных исследований показали, что предложенный подход, в среднем, превосходит метод NOVA в 2,41 раза (в отдельных случаях – в 7,3 раза), а метод JEDI в 1,96 раза (в отдельных случаях – в 5,43 раза). В заключении указывается на особенности практического использования рассмотренного метода, а также на перспективные направления дальнейших исследований в данной области.

Ключевые слова — энергосбережение; потребляемая мощность; конечные автоматы; логический синтез; программируемые логические интегральные схемы; ПЛИС; системы на программируемом кристалле; кодирование внутренних состояний; структурные модели.

I. ВВЕДЕНИЕ

В последнее время в связи с широким использованием переносных и бортовых встроенных систем (мобильные телефоны, плееры, навигаторы, космические системы управления) особую актуальность приобретает задача снижения потребляемой мощности цифровых систем. Имеется несколько подходов к решению данной задачи: технологические (использование энергосберегающих технологий); логические [1]-[3] (применение специальных методов синтеза); системные [4]-[6] (управление частотой синхронизации и напряжением питания для отдельных частей системы) и др. Одним из путей решения указанной проблемы является уменьшение потребляемой мощности конечных автоматов. Как известно, конечные автоматы широко используются в различных цифровых системах в качестве контроллеров, стандартных функциональных узлов, а также оригинальных последовательностных схем.

При проектировании цифровых устройств и систем на СБИС общепризнанной практикой является буферизация всех входных и выходных сигналов. Основной идеей предлагаемого подхода к энергосберегающему синтезу конечных автоматов является использование

триггеров входных и выходных буферов СБИС в качестве элементов памяти конечных автоматов, а наборы значений входных и выходных сигналов – в качестве части кода внутренних состояний. Следует отметить, что всякое изменение входных и выходных сигналов вызывает потребление мощности соответствующими буферами цифровой системы. Обычно эта мощность не учитывается при логическом синтезе и считается «неуменьшаемой» потребляемой мощностью.

При реализации цифровых систем на программируемых логических интегральных схемах (ПЛИС) буферизация сигналов может выполняться как в элементах ввода-вывода, так и с помощью внутренней логики ПЛИС: логическими элементами FPGA (Field Programmable Gate Arrays) или макроячейками CPLD (Complex Programmable Logic Arrays). Элементы ввода-вывода ПЛИС обычно используются для буферизации внешних сигналов, а внутренняя логика – для буферизации сигналов, передаваемых между подсистемами и блоками цифровой системы. Предлагаемый подход может также применяться и в случае использования систем на программируемом кристалле (System On Programmable Chip – SOPC), поскольку SOPC включают программируемую область, архитектура которой подобна FPGA или CPLD.

Отметим одно важное свойство ПЛИС: возможность устанавливать для отдельных групп запоминающих элементов буферов индивидуальные сигналы синхронизации. Если в качестве сигнала синхронизации элементов памяти буферов выбрать сигнал синхронизации конечного автомата, то триггеры буферов можно использовать в качестве элементов памяти конечного автомата. Кроме того, логические элементы (макроячейки) многих ПЛИС могут программироваться с триггером на входе, с триггером на выходе, с триггером в цепи обратной связи и др. Указанные свойства ПЛИС позволили предложить новые структурные модели для снижения стоимости реализации и потребляемой мощности, а также повышения быстродействия конечных автоматов [7].

II. СТРУКТУРНАЯ МОДЕЛЬ ADE

В [7] приведена классификация структурных моделей конечных автоматов, в результате которой все конечные автоматы можно разделить на 6 классов: А, В,

С, D, E и F. Автоматы классов А и В – это традиционные структурные модели конечных автоматов типа Мили и Мура, соответственно. В автоматах классов С и D в качестве элементов памяти используются триггеры выходных буферов ПЛИС, а в автоматах E и F – входных буферов ПЛИС.

На практике редко удается непосредственно реализовать автоматы классов C-F. Дело в том, что не всегда входные и (или) выходные наборы однозначно определяют коды внутренних состояний конечного автомата. Поэтому для построения автоматов классов C-F часто вводятся дополнительные элементы памяти, аналогичные элементам памяти автоматов классов А и В. В последнем случае образуются совмещенные модели конечных автоматов. Однако совмещать модели конечных автоматов можно только при совпадении времени формирования их выходных сигналов. Анализ [7] временных диаграмм функционирования конечных автоматов различных классов показал, что допустимы следующие четыре совмещенные модели конечных автоматов: ADE, AD, AE и BF.

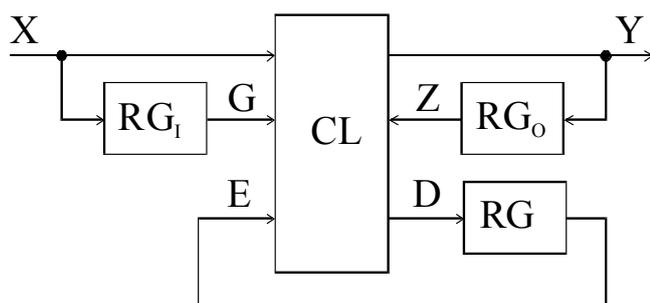


Рис. 1. Структура совмещенной модели ADE

Как наиболее общий случай рассмотрим совмещенную структурную модель ADE на рис.1, где CL – комбинационная часть конечного автомата; RG_I , RG_O и RG – регистры для хранения части кода внутренних состояний, определяемой значениями входных (автомат класса E), выходных (автомат класса D) и внутренних (автомат класса A) переменных, соответственно; $X = \{x_1, \dots, x_L\}$ – множество входных переменных; $Y = \{y_1, \dots, y_N\}$ – множество выходных переменных; $D = \{d_1, \dots, d_R\}$ – множество функций возбуждения элементов памяти регистра RG ; $E = \{e_1, \dots, e_R\}$ – множество внутренних переменных обратной связи; $G = \{g_1, \dots, g_L\}$ – множество внутренних переменных, соответствующих входным переменным; $Z = \{z_1, \dots, z_N\}$ – множество внутренних переменных, соответствующих выходным переменным. Код внутреннего состояния автомата определяется значениями переменных множеств E, G и Z.

Для реализации регистра RG_I могут использоваться триггеры входных буферов ПЛИС или внутренней логики, регистра RG_O – триггеры выходных буферов ПЛИС или внутренней логики, регистра RG – триггеры внутренней логики. Отметим, что в методах логического синтеза конечных автоматов мощность, рассеиваемая на буферных регистрах RG_I и RG_O , не рассматривается, поскольку изменения входных и выходных

сигналов являются обязательным условием функционирования конечного автомата.

III. МЕТОД ЭНЕРГОСБЕРЕГАЮЩЕГО СИНТЕЗА СОВМЕЩЕННОЙ МОДЕЛИ ADE

Суть данного метода заключается в специальном кодировании внутренних состояний конечного автомата, которое позволяет использовать значения входных и выходных переменных в качестве части кода внутренних состояний. Для этого строится матрица кодов W . Строки матрицы W соответствуют внутренним состояниям конечного автомата множества A , а столбцы – элементам множеств G и Z . На пересечении строки a_i и столбца g_j матрицы W ставится единица, если на всех переходах в состояние a_i входная переменная x_j входит в условие перехода в прямом значении, ноль – в инверсном значении, и неопределенное значение, обозначаемое дефисом (“-”), если переменная x_j не входит в условия переходов в состояние a_i или входит в условия переходов в различных значениях для разных переходов. На пересечении строки a_i и столбца z_j матрицы W ставится единица, если на всех переходах в состояние a_i выходная переменная y_j принимает единичное значение, ноль – нулевое значение, и неопределенное значение, если переменная y_j принимает различные значения на разных переходах.

Если все строки матрицы W взаимно ортогональны, то в качестве кодов внутренних состояний принимаются значения соответствующих строк матрицы W (две строки матрицы W считаются ортогональными, если в одинаковых разрядах они имеют различные значащие значения: 0 или 1). В противном случае решается задача ортогонализации строк матрицы W путем дополнительного кодирования переменными множества E. Последняя задача сводится к построению графа H ортогональности строк матрицы W и разбиению его на минимальное число полных подграфов H_1, \dots, H_T . С учетом вышеизложенного общий процесс энергосберегающего синтеза совмещенной модели конечного автомата ADE можно представить в виде следующего алгоритма.

Алгоритм 1 (общий алгоритм синтеза).

1. Строится матрица W кодов внутренних состояний конечного автомата, определяемая значениями переменных множеств G и Z .
2. Строится граф H ортогональности строк матрицы W .
3. Граф H разбивается на минимальное число T полных подграфов H_1, \dots, H_T .
4. Выполняется кодирование подграфов H_1, \dots, H_T двоичными кодами длиной R , $R \geq \text{intlog}_2 T$, соответствующими значениям переменных множества E с учетом минимизации потребляемой мощности.
5. В матрицу W добавляются столбцы, соответствующие переменным множества E . Значения этих столбцов определяются двоичными кодами соответствующих подграфов.

6. В качестве кодов внутренних состояний конечного автомата принимается значение строк матрицы W .
7. Конец.

Вершины графа H , который строится в пункте 2 алгоритма 1, соответствуют строкам матрицы W (т.е. внутренним состояниям конечного автомата). Две вершины i и j графа H соединяются ребром, если строки i и j матрицы W ортогональны между собой. Для разбиения графа H на минимальное число T полных подграфов H_1, \dots, H_T (пункт 3 алгоритма 1) может быть использован следующий алгоритм.

Алгоритм 2 (разбиения графа на подграфы)

1. Из графа H удаляются вершины, связанные со всеми другими вершинами графа (соответствующие им строки матрицы W ортогональны всем другим строкам). Полагается $t := 0$.
2. Полагается $t := t + 1$. В графе H отыскивается максимальный полный подграф H_t .
3. Из графа H удаляются вершины подграфа H_t . Если множество вершин графа H пусто, то выполняется переход к пункту 4, иначе - к пункту 2.
4. Конец.

Кодирование подграфов H_1, \dots, H_T , выполняемое в пункте 4 алгоритма 1, осуществляется с помощью следующего алгоритма.

Алгоритм 3 (кодирования подграфов)

1. Согласно [8] строится граф G_P вероятностей переходов между внутренними состояниями конечного автомата, в котором вероятность w_{ij} перехода между состояниями a_i и a_j определяется из выражения:

$$w_{ij} = P(a_i \rightarrow a_j) + P(a_j \rightarrow a_i),$$

где $P(a_i \rightarrow a_j)$ – вероятность перехода между состояниями a_i и a_j , определяемая согласно [8].

2. На основании графа G_P и разбиения графа ортогональности H строится граф G_H вероятностей переходов между состояниями подграфов H_1, \dots, H_T . Вершины графа G_H соответствуют подграфам H_1, \dots, H_T . Вес ребра w_{st} графа G_H , соединяющего подграфы H_s и H_t , $t, s \in \overline{1, T}$, $t \neq s$, определяется из выражения:

$$w_{st} = \Sigma \{ w_{ij} / a_i \in H_s, a_j \in H_t, i, j \in \overline{1, M}, i \neq j \}.$$

3. Выполняется кодирование вершин графа G_H с целью минимизации потребляемой мощности одним из известных методов [9].
4. Конец.

IV. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНЫХ ИССЛЕДОВАНИЙ

Отметим, что на величину потребляемой мощности конечного автомата в значительной степени влияет метод кодирования, применяемый в пункте 3 алгоритма 3. В приводимых ниже экспериментальных исследованиях с этой целью использовались последовательный метод, итерационный метод, а также метод с изменяющимся числом R разрядов кода [9]. Из полученных результатов выбирался конечный автомат с наименьшей потребляемой мощностью.

Предложенный метод использования совмещенной модели ADE для энергосберегающего синтеза конечных автоматов сравнивался со следующими методами: NOVA [10], JEDI [11], столбцовым [1], последовательным и итерационным [9]. В качестве исходных данных были приняты эталонные примеры (benchmarks) конечных автоматов, разработанные в центре MCNC [12].

Таблица 1

Результаты экспериментальных исследований методов кодирования внутренних состояний конечных автоматов для снижения потребляемой мощности

FSM	P_N	P_J	P_K	P_S	P_I	P_C
bbara	83,91	59,44	56,26	52,77	52,77	52,39
bbtas	144,29	112,50	83,15	83,15	83,15	72,76
beecount	160,50	108,05	108,92	89,42	89,42	75,37
cse	93,36	65,34	55,85	44,97	44,96	44,70
dk16	487,86	416,60	377,53	309,09	290,41	228,31
dk27	299,11	325,89	223,21	223,21	223,21	151,77
dk512	380,58	450,89	319,75	238,84	215,40	191,69
donfile	324,22	351,56	265,63	222,66	207,03	140,63
ex1	438,21	307,75	157,70	138,55	133,29	133,95
keyb	191,40	121,67	121,25	104,35	104,35	103,92
pma	273,73	203,53	105,55	104,76	104,28	37,51
s27	216,65	170,43	168,33	168,33	166,23	86,92
s8	53,21	42,41	33,90	33,90	33,90	35,09
shiftreg	281,25	281,25	257,81	210,94	187,50	187,50
train11	107,34	77,45	86,96	63,52	63,52	51,63
mid	250,23	212,84	172,18	152,24	133,29	106,28

По результатам кодирования внутренних состояний для всех примеров согласно [8] была вычислена потребляемая мощность при вероятности появления единицы (нуля) на каждом входе конечного автомата 0,5 и следующих значениях параметров: $V_{DD} = 5V$, $f = 5MHz$ и $C = 3pF$. Результаты экспериментальных исследований приведены в табл.1 и табл.2, где FSM – имя эталонного примера; P_N , P_J , P_K , P_S , P_I и P_C – потребляемая мощность в милливаттах (mW) конечного автомата в случае применения метода NOVA, JEDI, столбцового, последовательного, итерационного и совмещенной модели ADE соответственно; P_N/P_C , P_J/P_C , P_K/P_C , P_S/P_C и P_I/P_C – отношения соответствующих параметров; mid – среднеарифметическое значение параметра.

Таблица 2

Сравнение предлагаемого метода с известными методами

FSM	P_N/P_C	P_V/P_C	P_K/P_C	P_S/P_C	P_I/P_C
bbara	1,60	1,13	1,07	1,01	1,01
bbtas	1,98	1,55	1,14	1,14	1,14
beecount	2,13	1,43	1,45	1,19	1,19
cse	2,09	1,46	1,25	1,01	1,01
dk16	2,14	1,82	1,65	1,35	1,27
dk27	1,97	2,15	1,47	1,47	1,47
dk512	1,99	2,35	1,67	1,25	1,12
donfile	2,31	2,50	1,89	1,58	1,47
ex1	3,27	2,30	1,18	1,03	1,00
keyb	1,84	1,17	1,15	1,00	1,00
pma	7,30	5,43	2,81	2,79	2,78
s27	2,49	1,96	1,94	1,94	1,91
s8	1,52	1,21	0,97	0,97	0,97
shiftreg	1,50	1,50	1,37	1,13	1,00
train11	2,08	1,50	1,68	1,23	1,23
mid	2,41	1,96	1,51	1,34	1,30

Анализ табл.2 показывает, что предлагаемый метод, в среднем, превосходит метод NOVA в 2,41 раза (в отдельных случаях – в 7,3 раза), метод JEDI в 1,96 раза (в отдельных случаях – в 5,43 раза), столбцовый метод в 1,51 раза (в отдельных случаях – в 2,81 раза), последовательный метод в 1,34 раза (в отдельных случаях – в 2,79 раза) и итерационный метод в 1,3 раза (в отдельных случаях – в 2,78 раза).

Такое большое преимущество предлагаемого метода над лучшими из известных энергосберегающих методов кодирования внутренних состояний конечных автоматов объясняется отступлением от традиционных структурных моделей конечных автоматов и использованием новой структурной модели ADE. Данная структурная модель позволяет уменьшить число внутренних переменных конечного автомата и, как следствие, число внутренних элементов памяти и число функций возбуждения этих элементов памяти. Кроме того, используемая структурная модель позволяет снизить стоимость реализации конечного автомата за счет упрощения его комбинационной части. Упрощение комбинационной части конечного автомата в свою очередь приводит к уменьшению потребляемой мощности, которая в применяемой методике определения потребляемой мощности не учитывается. Следует также отметить, что упрощение комбинационной части конечного автомата уменьшает величину задержки сигналов, т.е. увеличивает быстродействие конечного автомата.

V. ЗАКЛЮЧЕНИЕ

В данной работе рассмотрено использование со-вмещенной структурной модели ADE для энергосберегающего синтеза конечных автоматов на ПЛИС. Чтобы

реализовать модель ADE ПЛИС должна обладать следующими архитектурными свойствами: входные буферы должны иметь два типа связей с внутренней логикой – комбинационную и регистровую; логические элементы или выходные буферы должны допускать конфигурацию с триггером в цепи обратной связи. В случае, если целевая ПЛИС не обладает одним из указанных свойств, для синтеза конечных автоматов можно выбрать другие структурные модели конечных автоматов: AD, AE, BF или C.

Работа выполнена при частичной финансовой поддержке Белостокского технического университета (грант № S/WI/4/2008).

ЛИТЕРАТУРА

- [1] Benini L., De Micheli G. State assignment for low power dissipation // IEEE Journal on Solid-State Circuits. 1995. Vol. 30. No. 3. P. 259–268.
- [2] Wu X., Pedram M., Wang L. Multi-code state assignment for low power design // IEE Proc.: Circuits, Devices and Systems. 2000. Vol. 147. No. 5. P. 271–275.
- [3] Partitioned state encoding for low power in FPSAs / Mengibar L., Entrena L., Garcia Lorenz M., San Millan E. // Electronic Letters. 2005. Vol. 41. No. 17. P. 948–949.
- [4] Pradhan S.N., Kumar M.T., Chattopadhyay S. Integrated power-gating and state assignment for low power FSM synthesis // Proc. of the IEEE Computer Society Annual Symposium on VLSI: Trends in VLSI Technology and Design. 2008. P. 269–274.
- [5] Saleem A., Khan S.A. Low power state machine design on FPGA // Proc. of the 3rd International Conference on Advanced Computer Theory and Engineering. 2010. Vol. 4. P. 4442–4445.
- [6] Nath P.S., Tilak K.M., Chattopadhyay S. Low power finite state machine synthesis using power-gating // Integration, the VLSI Journal. 2011. Vol. 44. No. 3. P. 175–184.
- [7] Соловьев В.В. Проектирование цифровых систем на основе программируемых логических интегральных схем. М.: Горячая линия – Телеком, 2001. 636 с.
- [8] Power estimation methods for sequential logic circuits / Tsui C.-Y., Monteiro J., Devadas S., Despain A.M., Lin B. // IEEE Trans. on VLSI Systems. 1995. Vol. 3. No. 3. P. 404 – 416.
- [9] Grzes T., Salauyou V., Bulatova I. Algorithms of coding the internal states of finite-state machine focused on the reduced power consumption // Radioelectronics and Communications Systems. 2010. Vol. 53. No. 5. P. 265–273.
- [10] Villa T., Vincentelli A.S. NOVA: state assignment of finite state machines for optimal two-level logic implementation // IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems. 1990. Vol. 9. No. 9. P. 905–924.
- [11] SIS: A system for sequential circuit synthesis / Sentovich E.M., Singh K.J., Lavagno L., et al. // Memorandum No. UCB/ERL M92/41, Electronics Research Laboratory, Berkley: Department of Electrical Engineering and Computer Science, University of California. 1992.
- [12] Yang S. Logic synthesis and optimization benchmarks user guide. Version 3.0. Technical Report. North Carolina. Microelectronics Center of North Carolina. 1991. 43 p.