

ПРОЕКТИРОВАНИЕ САМОСИНХРОННЫХ СХЕМ: СТРУКТУРНЫЕ МЕТОДЫ В ИЕРАРХИЧЕСКОМ АНАЛИЗЕ*

Л. П. Плеханов¹

Аннотация: Самосинхронные схемы (СС-схеиы) имеют уникальные свойства независимости от задержек и отказобезопасности. Рассмотрена одна из главных проблем проектирования таких схем — анализ самосинхронности больших схем. В традиционном подходе схемы анализируются событийными методами, по переключениям элементов. Сложность вычислений в таком подходе экспоненциально растет от размера и/или других параметров схем, что не позволяет анализировать большинство практически значимых схем. Решение проблемы предлагается в функциональном подходе — без использования переключений — и иерархическом описании схем. В иерархическом анализе самосинхронности наряду с анализом логических функций предлагается использовать структурные методы — исследование взаимосвязей элементов и фрагментов. Такой способ позволяет резко уменьшить трудоемкость вычислений и в итоге решить одну из главных проблем проектирования СС-схем — анализ схем любого размера. Эффективность предложенных методов подтверждена с помощью экспериментальных программных средств.

Ключевые слова: самосинхронные схемы; асинхронные схемы; проектирование схем; анализ самосинхронности

DOI: ?????

1 Введение

Самосинхронные схемы относятся к классу асинхронных схем, то есть схем, не имеющих тактовых генераторов. Русскоязычный термин «самосинхронная схема» предложен в книге [1] как аналог термина *speed-independent* [2] — «схема, правильность функционирования которой не зависит от величин задержек элементов». С практической точки зрения использование такого термина неудобно, так как он определен для замкнутых схем с одним начальным состоянием. (Другие близкие термины подробнее обсуждены в [3].) Поэтому для практических целей проектирования СС-схем было предложено более широкое определение [4, 5].

Самосинхронная схема — это разомкнутая или замкнутая схема, при всех реальных начальных состояниях и переходах между ними имеющая два свойства: отсутствие соствязаний при любых конечных задержках элементов и отказобезопасность по отношению к константным залипаниям на 0 и 1 выходов элементов.

Так определенная схема удовлетворяет также и критериям [1, 2] (в случае разомкнутой схемы — при правильном замыкании). При реализации на чипе самосинхронность может зависеть от задержек в трассах. Поэтому на функционально-логическом уровне разработки свойства, указанные в опре-

делении, становятся необходимыми условиями и должны быть обеспечены схемотехническими решениями. Но и в случае возможного нарушения самосинхронности от задержек в трассах есть реальные способы (схемотехнические и топологические) обеспечения самосинхронности [5].

Уникальные свойства СС-схем, присущие им по определению, имеют и уникальные следствия. Они обеспечивают максимально широкий диапазон правильного функционирования, определяемый только физическими (не схемотехническими) возможностями переключений элементов, что недостижимо для других типов схем. Устойчивость работы СС-схем при малых напряжениях питания позволяет создавать схемы с малым энергопотреблением. А свойство отказобезопасности дает возможность получения высоконадежных схем, в том числе с самопроверкой и саморемонтом.

Разработка практических СС-схем началась в СССР группой В. И. Варшавского в 70-х гг. прошлого века [6] и с запозданием на 20 лет за рубежом [7]. Последняя ссылка указывает на единственную зарубежную методологию построения СС-схем *Null Convention Logic* (NCL). Эта методология была создана чисто интуитивно, по принципу «запрос—ответ». Методология имеет ряд существенных недостатков: ограниченный базис реализации

* Работа выполнена при частичной финансовой поддержке по Программе фундаментальных исследований Президиума РАН № 1 за 2014 г. (проект № 43П) и РФФИ (проекты 13-07-12062 офи.м и 13-07-12068 офи.м).

¹Институт проблем информатики Российской академии наук, LPlehanov@inbox.ru

(27 элементов), жесткая конвейерная структура построения схем, не допускающая альтернатив, и др. В результате схемы, получаемые по методологии NCL, имеют большие затраты в транзисторах, что и отмечено бывшими сотрудниками В. И. Варшавского [8].

Методология разработки СС-схем в ИПИ РАН опирается на математические методы Маллера и Варшавского и развивает методы группы В. И. Варшавского. Одним из результатов разработок ИПИ РАН являются 22 патента РФ и 2 США в области СС-схемотехники.

Методология ИПИ РАН позволяет создавать СС-схемы в тысячи элементов как на базовых матричных кристаллах (БМК), так и в заказном исполнении. Методология дает возможность сравнения множества вариантов и выбора оптимального. Подробный сравнительный анализ NCL-методологии, выполненный в ИПИ РАН [9], показывает, что схемы NCL уступают идентичным по функционированию схемам ИПИ РАН по всем показателям: затратам в транзисторах, быстродействию и энергопотреблению.

В настоящее время институтом в содружестве с другими учреждениями (НТЦ МИЭТ, НИИСИ РАН) разработаны, изготовлены и испытаны ряд изделий на БМК и в заказном формате [10–12]. Работы по развитию СС-схем продолжаются [9, 13–15].

Одна из главных проблем проектирования СС-схем заключается в необходимости применения специальных математических и других методов, требующих больших, а подчас огромных вычислительных ресурсов. Существующие классические методы ограничивают возможности анализа и синтеза СС-схем размером до нескольких десятков элементов. Поэтому актуальной является задача разработки и реализации новых методов, альтернативных существующим, с целью довести возможности проектирования до реальных потребностей, т.е. до схем любого размера. Одним из путей решения этой задачи могут служить методы анализа и синтеза, основанные на функциональном подходе [5, 14].

Данная статья продолжает развитие методов, приведенных в докладе на МЭС-2012 [16].

2 Проблемы анализа в функциональном подходе

Как показано в [5, 16], проблему полного анализа СС-схем можно решить только иерархическим способом. В этом способе на нижнем уровне

используются описания фрагментов в логических уравнениях, а на верхних уровнях — взаимосвязи фрагментов и результаты анализа более низких уровней.

В классическом подходе, основанном на событийных описаниях замкнутых схем (событие — изменение состояния схемы, в частности изменение одного сигнала), иерархического анализа пока не предложено.

В рамках функционального подхода (разомкнутое представление схем) анализ проводится иерархически, с разделением схем на фрагменты. Для каждого фрагмента должны проверяться оба требования определения СС-схем: отказобезопасность и отсутствие состязаний. При этом анализ фрагментов нижнего уровня и анализ на верхних уровнях делаются по-разному.

Основные проблемы анализа здесь возникают на нижнем уровне. На нем с неизбежностью необходимо использовать полные описания фрагментов в уравнениях. Уравнения должны учитывать, помимо функционирования, все возможные реальные состояния и переходы между ними.

Для учета всех состояний уравнения элементов записываются в зависимости от параметров анализа [15, 16]: независимых переменных информационных входов и переменных памяти. Для проверки отказобезопасности используется индизирование сигналов [1], для определения состязаний предложен способ проверочных функций для каждого элемента [5, 16].

Созданная по этому методу программа ФАЗАН [17] показала правильное выполнение анализа, а также выявила ряд трудностей. Выяснилось, что нахождение индизируемости сигналов выполняется достаточно быстро даже при большом числе параметров анализа и не представляет реальной проблемы. Основная трудность заключается в определении состязаний. Проверочные функции и часть уравнений в общем случае имеют увеличенное число аргументов, поскольку добавляются изоперемные, необходимые для выявления состязаний. В результате определение монотонности проверочных функций становится громоздким и ограничивает размер анализируемого фрагмента нижнего уровня.

Простым решением этой проблемы может быть построение нижнего уровня схем из фрагментов небольшого размера, что вполне реализуемо. Однако это не всегда удобно и ограничивает маневр разработчика.

Задачу радикального уменьшения трудоемкости анализа СС-схем в функциональном подходе можно решить привлечением структурных методов.

3 Структурные методы в иерархическом анализе

Идея структурных методов состоит в том, что при анализе состязаний можно отказаться от вычисления логических функций, а вместо этого выявить элементарные ячейки памяти и анализировать взаимосвязи этих ячеек и подключенных к ним сигналов.

Необходимым требованием отсутствия состязаний станет блокировка (запрет записи) запоминающих ячеек на время изменения их входов. Двухфазный характер работы СС-схем (чередование рабочей фазы и вспомогательной — спейсера) обеспечивает полную возможность для проверки этого требования.

Поскольку в СС-схемах допустимы любые конечные задержки элементов, то в пределах одной фазы нельзя допустить и блокировку, и изменение входов ячеек. Поэтому необходимое требование выполняется с разделением по фазам: в фазе, в которой входы запоминающих ячеек меняются, делается блокировка, в другой фазе, когда эти входы не изменяются, разрешается перезапись. Такой порядок и делает возможным анализ состязаний структурным методом.

3.1 Анализ на верхних уровнях

Описание схемы на любом верхнем уровне содержит только фрагменты, которые были успешно проверены ранее.

Иерархический анализ на верхних уровнях подробно изложен в [5, 16].

Предложенный метод был реализован и проверен с помощью экспериментальной программы ЛИМАН. Испытания программы на схемах средней сложности (8-битный микропроцессор «Микроядро») показали высокую эффективность метода. Анализ на любом из верхних уровней иерархии с полнотой по всем состояниям и переходам занимает сотые или десятые доли секунды.

Результаты анализа программой ЛИМАН в сравнении с событийным методом приведены в разд. 4.

3.2 Анализ на нижнем уровне

В отличие от верхних уровней, фрагмент на нижнем уровне описания содержит логические уравнения элементов и, возможно, другие фрагменты, ранее прошедшие анализ. Совокупность всех логических уравнений фрагмента будем называть его *логической частью*.

Предлагается описание логической части представить в виде взаимосвязанных фрагментов для иерархического анализа, полностью исключив при этом трудоемкий анализ состязаний нижнего уровня. Тем самым логическая часть может быть проанализирована как фрагмент верхнего уровня.

На первом шаге определяются необходимые для дальнейшего атрибуты интерфейса [5, 16] логической части по ее взаимосвязям: СС-типы ее входов и выходов (фазовые, нефазовые и др.), значения входных спейсеров.

На следующем шаге функциональным методом [3] проверяется индицирование всех внутренних сигналов логической части на ее фазовых выходах. Как упоминалось выше, это действие не представляет больших вычислительных трудностей. Попутно на этом шаге вычисляются и параметры интерфейса, необходимые для дальнейшего: значения спейсеров фазовых выходов, списки индицируемости.

Далее в логической части следует выделить комбинационную часть (КЧ) и отдельно все бистабильные ячейки (БСЯ). (Довольно редко используемые многостабильные ячейки для простоты не рассматриваются.) Выделение производится по взаимосвязям элементов, т. е. уже структурным способом.

Одну БСЯ будут составлять два элемента с перекрестными связями — с выхода одного элемента на один из входов другого. Элементы, не попавшие в БСЯ, будут относиться к КЧ.

И КЧ, и все БСЯ затем оформляются как фрагменты для иерархического анализа. Все требуемые при этом атрибуты интерфейсов новых фрагментов определяются либо непосредственно, либо на основании предыдущих шагов.

В соответствии с общим правилом анализа самосинхронности [5, 16] каждый новый фрагмент должен быть проверен на отказобезопасность и отсутствие состязаний внутренних сигналов.

Отказобезопасность фрагмента обеспечивается индицированием всех его внутренних сигналов на его внешних фазовых выходах. Ранее на одном из шагов была проверена индицируемость внутренних сигналов логической части. По свойству транзитивности индикации, если внутренний сигнал фрагмента индицируется на выходах логической части, то он с необходимостью индицируется и на выходах фрагмента, так как другого пути для его индикации нет.

Таким образом, после проверки индикации логической части все новые ее фрагменты также становятся проверенными на отказобезопасность.

Отсутствие состязаний внутренних сигналов в БСЯ априорно следует из их структуры: внутренних

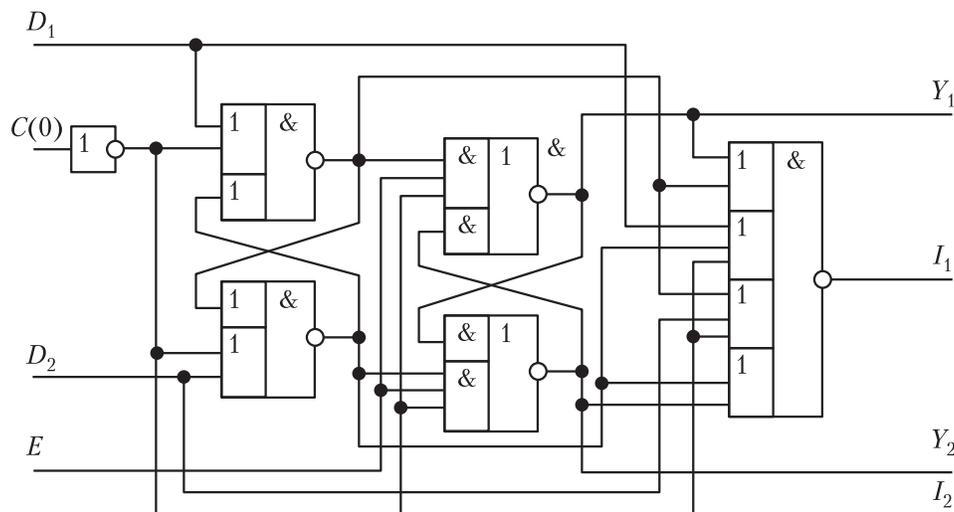


Рис. 1 Ячейка сдвигового регистра

сигналов всего два — это сигналы, связывающие последовательно два элемента в кольцо. Такие сигналы не могут вызвать состязаний. Состязания могут появиться из-за внешних сигналов БСЯ, но их проверка осуществляется уже на более высоком уровне иерархии.

Для КЧ отсутствие состязаний следует из серии доказанных в [1] теорем для разомкнутых схем, заканчивающихся следующим:

«Утверждение 4.4. Двухфазная комбинационная схема является аperiodической тогда и только тогда, когда она индицируема».

В цитируемой книге термин «аperiodическая» является синонимом термина «самосинхронная», а под индицируемой понимается схема, все внутренние элементы которой индицируются на внешних выходах. Поэтому КЧ не нуждается в отдельной проверке на отсутствие состязаний.

В результате описанной процедуры логическая часть будет представлена как соединение фрагментов, уже прошедших анализ.

Последним шагом предлагаемого метода будет иерархический анализ сначала логической части, затем целиком рассматриваемого фрагмента нижнего уровня.

Можно заметить, что вычислительная сложность этого метода линейно зависит от количества элементов и их взаимосвязей. Такая зависимость не порождает громоздких вычислений на ЭВМ рассматриваемой части задачи.

Рассмотрим пример схемы ячейки сдвигового регистра [18] (рис. 1). Здесь и далее знак « \wedge » означает отрицание, знак « \vee » — операцию ИЛИ, отсутствие знака — операцию И.

Схема задается уравнениями:

$$I_2 = \wedge C; \quad (1)$$

$$Q_1 = \wedge((D_1 \vee I_2)Q_2); \quad (2)$$

$$Q_2 = \wedge((D_2 \vee I_2)Q_1); \quad (3)$$

$$Y_1 = \wedge(Q_1 E I_2 \vee Y_2); \quad (4)$$

$$Y_2 = \wedge(Q_2 E I_2 \vee Y_1); \quad (5)$$

$$I_1 =$$

$$= \wedge((Y_1 \vee Q_1)(D_1 \vee Q_2 \vee I_2)(D_2 \vee Q_1 \vee I_2)(Y_2 \vee Q_2)). \quad (6)$$

Здесь C — управляющий сигнал; E — сигнал дополнительной блокировки.

В данном случае вся схема составляет логическую часть. Анализ индикации показывает, что все ее элементы индицируются: (2)–(5) — на индикаторном выходе I_1 , инвертор (1) — на выходе I_2 .

Схема разбивается на три фрагмента: БСЯ-1 — элементы (2) и (3), БСЯ-2 — элементы (4) и (5) и КЧ, в которую входят инвертор (1) и индикатор (6). Все фрагменты являются самосинхронными, так как прошли общую проверку на индицирование.

Проверка самосинхронности всей схемы будет состоять в анализе межсоединений фрагментов иерархическим методом [15, 16], для чего требуется вычислить параметры блокировки нефазовых сигналов Q_1 , Q_2 , Y_1 , Y_2 . Здесь входы и выходы БСЯ-1 и нефазовые входы индикатора блокируются в фазе спейсера сигналом C с задержкой. Входы и выходы БСЯ-2 блокируются в рабочей фазе сигналом C также с задержкой и сигналом E без задержки.

Опуская подробности, можно сделать вывод, что все соединения фрагментов корректны по отсутствию состязаний. Сигнал E , вследствие задержки блокировки БСЯ-2, следует подключать к сигналам

лу I_2 последующей ячейки регистра, если таковая предусмотрена.

4 Сравнение результатов иерархического и событийного анализа

Для анализа была выбрана СС-схема 8-битного микропроцессора «Микроядро», ранее подробно проанализированная событийными методами [19]. Верхний уровень схемы показан на рис. 2 (на рисунке не показан управляющий блок, задающий режимные константы и не подлежащий анализу).

Иерархическая структура схемы по уровням и именам блоков выглядит следующим образом (в скобках даны размеры блоков в вентилях). MicroCore (892)

- 1 — ROM (81)
 - 2 — GI2 (3), ROM_REG (28), PC_S (32), ROM_DC (18)
- 1 — DCF (11)
 - 2 — INV (1), GI2 (3), R0RE10N (6), INV (1)
- 1 — CLK_MC (123)
- 1 — MUL (448)
 - 2 — GI2 (3), INV (1), INV (1), GI2 (3), INV (1), GI2 (3), GI3 (4), SSMULT (337)
 - 3 — SSSM (12), MINUSA (33), SSSM (12), SSSM (12), SSDC (48), SUM20 (23), SSSM1 (6), SUM21 (19), SUM22 (22), MULIND (51), PARTP0 (17), PARTP1 (34), PARTP2 (48)
 - 2 — R1R10 (64)
 - 3 — R1RE11 (6), R1RE11 (6), INV (1), GI4 (5), INV2 (1), GI2 (3), GI3 (4), INV2 (1), INV (1), R1RE11 (6), R1RE11 (6), R1RE11 (6), R1RE11 (6), R1RE11 (6), R1RE11 (6)
 - 2 — MX22 (25)
- 1 — ROT (152)
 - 2 — R010 (4), R010 (4), R010 (4), R010 (4), GI2 (3), GI2 (3), GI4 (5), MX21R (17), SHFT43_N (110)
 - 3 — GI3M (3), GI2 (3), GI2 (3), INV (1), GI3 (4), INV3 (2), MX21N (25), PPTR (13), PPTR (13), PPTR (13), PPTR (13), MX310N (17)
- 1 — MX311 (29)

Суммарное время анализа всех блоков схемы программой ЛИМАН (без учета файловых операций) составило 0,84 с.

Иерархический анализ в функциональном подходе, как указывалось ранее, обеспечивает исчерпывающую полноту анализа, т. е. учитываются все реальные начальные состояния и все возможные

реальные сочетания значений входов блоков и переходов между ними.

Однократный анализ событийным методом этого же микропроцессора с одним начальным состоянием и одной из комбинаций входных значений [12] занимает 7 с.

Для обеспечения необходимой полноты требуется выполнить десятки тысяч таких сеансов анализа.

Приведенные данные показывают, какие практические трудности возникают при проектировании СС-схем классическими (событийными) методами.

5 Заключение

Структурные методы в иерархическом анализе самосинхронности схем состоят в том, что исследуются не уравнения элементов, а их взаимосвязи. Применение этих методов оказалось возможным в рамках функционального подхода, когда рассматриваются разомкнутые схемы, работающие по принципу чередования двух фаз.

Эффективность структурных методов следует из того, что их вычислительная сложность линейно зависит от количества элементов/фрагментов и их взаимосвязей.

На всех уровнях иерархического анализа выше нижнего проверки осуществляются исключительно структурным методом. Так, анализ 8-битного ядра микропроцессора «Микроядро» занимает десятки доли секунды.

На нижнем уровне одна из двух проверок — отсутствие состязаний — сводится к такому же структурному методу, что и анализ на верхних уровнях. Вторая проверка — индицирование — заключается в последовательном вычислении логических функций. На практике она обычно не вызывает затруднений, особенно с учетом того, что размер фрагмента нижнего уровня может быть выбран самим пользователем.

Таким образом, применение структурных методов в анализе СС-схем позволяет радикально уменьшить сложность вычислений и решить проблему анализа схем любого размера (не решенную в классическом событийном подходе).

Литература

1. Автоматное управление асинхронными процессами в ЭВМ и дискретных системах / Под ред. В. И. Варшавского. — М.: Наука, 1986. 398 с.
2. Muller D. E., Bartky W. C. A theory of asynchronous circuits // Symposium (International) on the Theory of

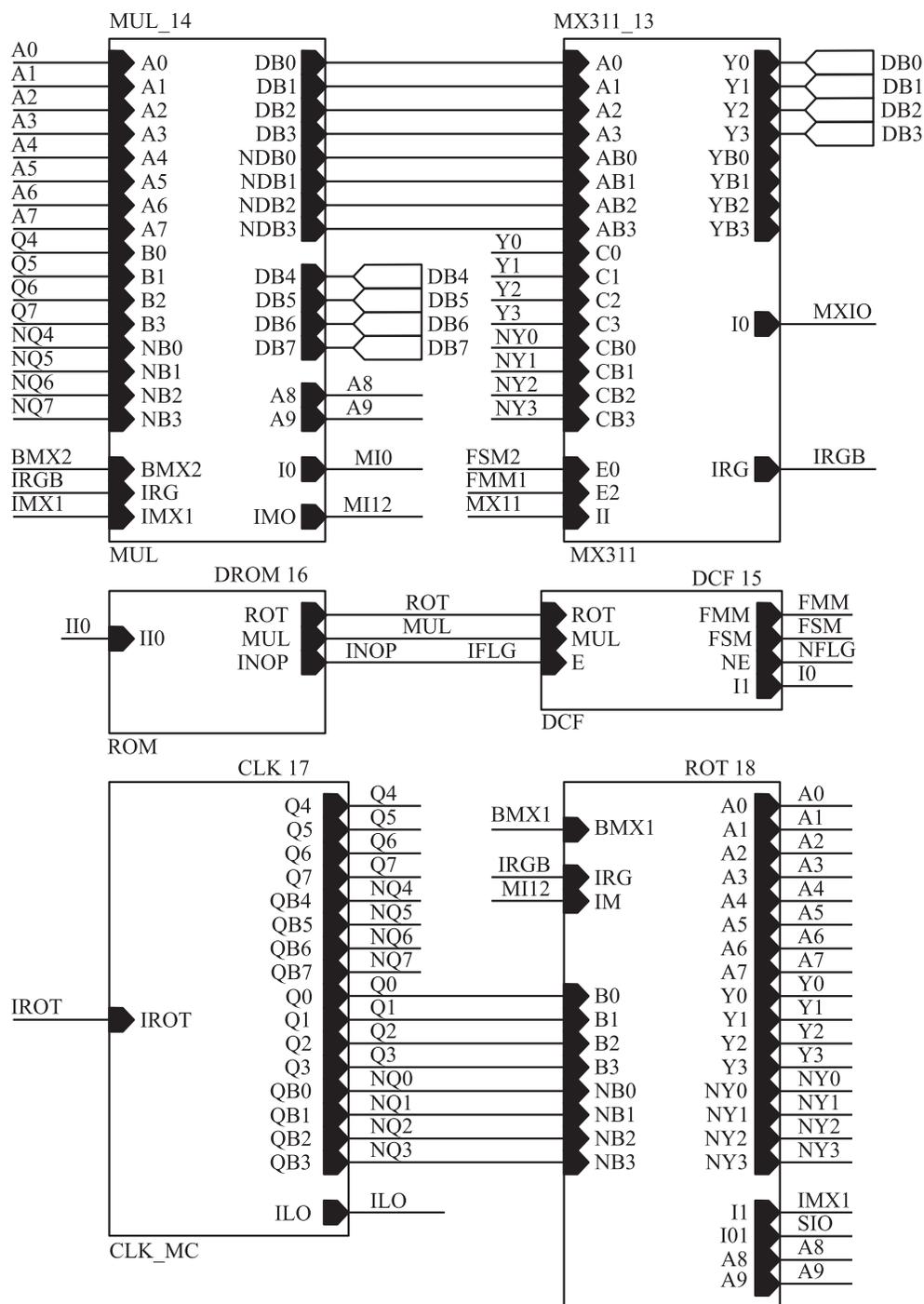


Рис. 2 Верхний уровень микропроцессора «Микроядро»

- Switching. — Harvard University Press, 1959. Part 1. P. 204–243.
3. Плеханов Л. П. Проектирование самосинхронных схем: функциональный подход // Проблемы разработки перспективных микро- и нанoeлектронных систем: IV Всеросс. научно-технич. конф. (МЭС-2010): Сб. науч. тр. — М.: ИППМ РАН, 2010. С. 424–429.
 4. Плеханов Л. П. О свойстве самосинхронности цифровых электронных схем // Системы и средства информатики, 2011. Вып. 21. № 1. С. 84–91.
 5. Плеханов Л. П. Основы самосинхронных электронных схем. — М.: Бином. Лаборатория знаний, 2013. 208 с.
 6. Аperiodические автоматы / Под ред. В. И. Варшавского. — М.: Наука, 1976. 423 с.
 7. Fant K.M., Brandt S.A. NULL convention logic // Theseus Research: Technical Papers. P. 1–26. <http://www.theseusresearch.com/NCLPaper01.htm>.
 8. Taubin A., Cortadella J., Lavagno L., Kondratyev A., Peeters A. Design automation of real-life asynchronous devices and systems // Foundations and Trends in Electronic Design Automation, 2007. Vol. 2. No. 1. P. 1–133.
 9. Соколов И. А., Степченко Ю. А., Бобков С. Г. и др. Базис реализации супер-ЭВМ экзафлопного класса // Системы и средства информатики, 2014. Т. 25. № 1. С. 5–34.
 10. Степченко Ю. А., Петрухин В. С., Дьяченко Ю. Г. Опыт разработки самосинхронного ядра на базовом матричном кристалле // Проблемы разработки перспективных микро- и нанoeлектронных систем: I Всеросс. научно-технич. конф. (МЭС-2005): Сб. научных трудов. — М.: ИППМ РАН, 2005. С. 235–242.
 11. Степченко Ю. А., Дьяченко Ю. Г., Бобков С. Г. Квазисамосинхронный вычислитель: методологические и алгоритмические аспекты // Проблемы разработки перспективных микро- и нанoeлектронных систем: III Всеросс. науч.-технич. конф. (МЭС-2008): Сб. науч. тр. — М.: ИППМ РАН, 2008. С. 441–446.
 12. Степченко Ю. А., Дьяченко Ю. Г., Рождественский Ю. В., Морозов Н. В., Степченко Д. Ю. Самосинхронный вычислитель для высоконадежных применений // Проблемы разработки перспективных микро- и нанoeлектронных систем: IV Всеросс. науч.-технич. конф. (МЭС-2010): Сб. науч. тр. — М.: ИППМ РАН, 2010. С. 418–423.
 13. Степченко Ю. А., Денисов А. Н., Дьяченко Ю. Г., Гринфельд Ф. И., Филимонов О. П., Морозов Н. В., Степченко Д. Ю. Библиотека элементов для проектирования самосинхронных полузаказных БМК микросхем серий 5503/5507 и 5508/5509. — М.: ИПИ РАН, 2013. 391 с.
 14. Бобков С. Г., Горбунов М. С., Дьяченко Ю. Г., Рождественский Ю. В., Степченко Ю. А., Сурков А. В. Использование самосинхронной логики для снижения потребляемой мощности и повышения надежности микропроцессоров // Проблемы разработки перспективных микро- и нанoeлектронных систем: VI Всеросс. науч.-технич. конф. (МЭС-2014): Сб. науч. тр. — М.: ИППМ РАН, 2014. С. 254–257.
 15. Степченко Ю. А., Рождественский Ю. В., Дьяченко Ю. Г., Морозов Н. В., Степченко Д. Ю., Сурков А. В. Самосинхронное устройство умножения-сложения гигафлопного класса: варианты реализации // Проблемы разработки перспективных микро- и нанoeлектронных систем: VI Всеросс. науч.-технич. конф. (МЭС-2014): Сб. науч. тр. — М.: ИППМ РАН, 2014. С. 258–263.
 16. Плеханов Л. П. Функциональный метод анализа самосинхронных схем любого размера // Проблемы разработки перспективных микро- и нанoeлектронных систем: V Всеросс. науч.-технич. конф. (МЭС-2012): Сб. науч. тр. — М.: ИППМ РАН, 2012. С. 107–112.
 17. Плеханов Л. П. Программа анализа самосинхронных схем функциональным методом (ФАЗАН). Свидетельство о государственной регистрации программы для ЭВМ № 2011611102 от 01.02.2011.
 18. Степченко Ю. А., Дьяченко Ю. Г., Рождественский Ю. В., Рождественскене А. В. Разряд самосинхронного регистра сдвига. Патент на изобретение 2319232 (РФ). Приоритет от 10.03.2008.
 19. Степченко Ю. А., Дьяченко Ю. Г., Рождественский Ю. В., Морозов Н. В., Степченко Д. Ю. Разработка вычислителя, не зависящего от задержек элементов // Системы и средства информатики, 2010. Вып. 20. № 1. С. 5–23.

Поступила в редакцию 10.07.14

DESIGN OF SELF-TIMED CIRCUITS: STRUCTURAL METHODS IN HIERARCHICAL ANALYSIS

L. Plekhanov

Institute of Informatics Problems, Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation

Abstract: Self-timed circuits have unique properties of the delay-independence and fail-safe. One of the major problems of circuits design, self-timed analysis of large circuits, is considered. In the traditional approach, circuits

are analyzed by event methods with elements switches. Computational complexity in this approach increase exponentially with the size and/or other circuit parameters, which does not allow analyzing the most practically important circuits. The solution is proposed in the functional approach, without using switches, and in the hierarchical description of circuits. In the hierarchical analysis along with the analysis of logical functions, the author proposes to use structural methods, i. e., to study the interaction of elements and fragments. This method allows reducing the complexity of calculations dramatically and thus solves one of the major problems of self-timed circuits design — analysis of circuits of any size. Efficiency of the suggested methods is confirmed using the experimental software.

Keywords: self-timed circuits; asynchronous circuits; circuit design; self-timed analysis

DOI: ?????

Acknowledgments

The work was performed with partial financial support of the Program of Basic Research of the Presidium of the Russian Academy of Sciences No. 1 for 2014 (project No. 43P) and Russian Foundation for Basic Research (projects 13-07-12062 and 13-07-12068).

References

1. Varshavskiy, V. I., ed. 1986. *Avtomatnoe upravlenie asinkhronnymi protsessami v EVM i diskretnykh sistemakh* [Automata control of asynchronous processes in computers and discrete systems]. Moscow: Nauka. 398 p.
2. Muller, D. E., and W. C. Bartky. 1959. A theory of asynchronous circuits. *Symposium (International) on the Theory of Switching Proceedings*. Harvard University Press. 1:204–243.
3. Plekhanov, L. P. 2010. Proektirovanie samosinkhronnykh skhem: Funktsional'nyy podkhod [The design of self-timed circuits: Functional approach]. *Tr. IV Vseross. nauch.-tekhnich. konf. "Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem" (MES-2010)* [4th Russian Scientific and Technical Conference "Problems of the Perspective Micro- and Nanoelectronic Systems Development" (MES-2010) Proceedings]. Moscow. 424–429.
4. Plekhanov, L. P. 2011. O svoystve samosinkhronnosti tsifrovyykh elektronnykh skhem [About the self-timed property of digital electronic circuits]. *Sistemy i Sredstva Informatiki — Systems and Means of Informatics* 21(1):84–91.
5. Plekhanov, L. P. 2013. *Osnovy samosinkhronnykh elektronnykh skhem* [Basics of self-timed electronic circuits]. Moscow: Binom. Laboratoriya znaniy [Binom. Laboratory of knowledge]. 208 p.
6. Varshavskiy, V. I., ed. 1976. *Aperiodicheskie avtomaty* [Aperiodic automata]. Moscow: Nauka. 423 p.
7. Fant, K. M., and S. A. Brandt. NULL convention logic. *Theseus Research: Technical Papers*. 1–26. Available at: <http://www.theusresearch.com/NCLPaper01.htm> (accessed July 17, 2014).
8. Taubin, A., J. Cortadella, L. Lavagno, A. Kondratyev, and A. Peeters. 2007. Design automation of real-life asynchronous devices and systems. *Foundations and Trends in Electronic Design Automation*. 2(1):1–133.
9. Sokolov, I. A., Yu. A. Stepchenkov, S. G. Bobkov, et al. 2014. Baza realizatsii super-EVM eksaflopnoy klassa [The basis for the implementation of the super computer of exaflops class]. *Sistemy i Sredstva Informatiki — Systems and Means of Informatics* 25(1):5–34.
10. Stepchenkov, Yu. A., V. S. Petrukhin, and Yu. G. D'yachenko. 2005. Opyt razrabotki samosinkhronnogo yadra na bazovom matrichnom kristalle [The experience of developing self-timed kernel on basic matrix crystal]. *Tr. I Vseross. nauchno-tekhnich. konf. "Problemy Razrabotki Perspektivnykh Mikro- i Nanoelektronnykh Sistem" (MES-2005)* [1st Russian Scientific and Technical Conference "Problems of the Perspective micro- and nanoelectronic systems development" (MES-2005) Proceedings]. Moscow. 235–242.
11. Stepchenkov, Yu. A., Yu. G. D'yachenko, and S. G. Bobkov. 2008. Kvazisamosinkhronnyy vychislitel': Metodologicheskie i algoritmicheskie aspekty [Quasi-self-timed calculator: Methodological and algorithmic aspects]. *Tr. III Vseross. nauch.-tekhnich. konf. "Problemy Razrabotki Perspektivnykh Mikro- i Nanoelektronnykh Sistem" (MES-2008)* [3rd Russian Scientific and Technical Conference "Problems of the Perspective Micro- and Nanoelectronic Systems Development" (MES-2008) Proceedings]. Moscow. 441–446.
12. Stepchenkov, Yu. A., Yu. G. D'yachenko, Yu. V. Rozhdestvenskiy, N. V. Morozov, and D. Yu. Stepchenkov. 2010. Samosinkhronnyy vychislitel' dlya vysokonadezhnykh primeneniy [Self-timed calculator for high reliable applications]. *Tr. IV Vseross. nauch.-tekhnich. konf. "Problemy Razrabotki Perspektivnykh Mikro- i Nanoelektronnykh Sistem" (MES-2010)* [4th Russian Scientific and Technical Conference "Problems of the Perspective Micro- and Nanoelectronic Systems Development" (MES-2010) Proceedings]. Moscow. 418–423.
13. Stepchenkov, Yu. A., A. N. Denisov, Yu. G. D'yachenko, F. I. Grinfel'd, O. P. Filimonov, N. V. Morozov, and D. Yu. Stepchenkov. 2013. *Biblioteka elementov dlya proektirovaniya samosinkhronnykh poluzakaznykh BMK mikroskhem seriy 5503/5507 i 5508/5509* [Library of elements for designing self-timed semicustom VLSI 5503/5507 and 5508/5509]. Moscow: IPI RAN. 391 p.

14. Bobkov, S. G., M. S. Gorbunov, Yu. G. D'yachenko, Yu. V. Rozhdestvenskiy, Yu. A. Stepchenkov, and A. V. Surkov. 2014. Ispol'zovanie samosinkhronnoy logiki dlya snizheniya potrebyaemoy moshchnosti i povysheniya nadezhnosti mikroprotessorov [The use of self-timed logic to reduce power consumption and increase reliability of microprocessors]. *Tr. VI Vseross. nauch.-tekhnich. konf. "Problemy Razrabotki Perspektivnykh Mikro- i Nanoelektronnykh Sistem" (MES-2014)* [6th Russian Scientific and Technical Conference "Problems of the Perspective Micro- and Nanoelectronic Systems Development" (MES-2014) Proceedings]. Moscow. 254–257.
15. Stepchenkov, Yu. A., Yu. V. Rozhdestvenskiy, Yu. G. D'yachenko, N. V. Morozov, D. Yu. Stepchenkov, and A. V. Surkov. 2014. Samosinkhronnoe ustroystvo umnozheniya-slozheniya gigaflopsnogo klassa: Varianty realizatsii [Self-timed device for multiplication-addition of gigaflops class]. *Tr. VI Vseross. nauch.-tekhnich. konf. "Problemy Razrabotki Perspektivnykh Mikro- i Nanoelektronnykh Sistem" (MES-2014)* [6th Russian Scientific and Technical Conference "Problems of the Perspective Micro- and Nanoelectronic Systems Development" (MES-2014) Proceedings]. Moscow. 258–263.
16. Plekhanov, L. P. 2012. Funktsional'nyy metod analiza samosinkhronnykh skhem lyubogo razmera [The functional method of analysis of self-timed circuits of any size]. *Tr. V Vseross. nauch.-tekhnich. konf. "Problemy Razrabotki Perspektivnykh Mikro- i Nanoelektronnykh Sistem" (MES-2012)* [5th Russian Scientific and Technical Conference "Problems of the Perspective Micro- and Nanoelectronic Systems Development" (MES-2012) Proceedings]. Moscow. 107–112.
17. Plekhanov, L. P. 2011. Programma analiza samosinkhronnykh skhem funktsional'nym metodom (FAZAN) [The program of the analysis of self-timed circuits with the functional method (FAZAN)]. Svidetel'stvo o gosudarstvennoy registratsii programmy dlya EVM [Certificate of the State Registration of the Computer Program] No. 2011611102.
18. Stepchenkov, Yu. A., Yu. G. D'yachenko, Yu. V. Rozhdestvenskiy, and A. V. Rozhdestvenskene. 2008. Razryad samosinkhronnogo registra sdviga [Binary digit of self-timed shift register]. Patent RF No. 2319232.
19. Stepchenkov, Yu. A., Yu. G. D'yachenko, Yu. V. Rozhdestvenskiy, N. V. Morozov, and D. Yu. Stepchenkov. 2010. Razrabotka vychislitelya, ne zavisyashchego ot zaderzhek elementov [Design of speed-independent calculator]. *Sistemy i Sredstva Informatiki — Systems and Means of Informatics* 20(1):5–23.

Received July 10, 2014

Contributor

Plekhanov Leonid P. (b. 1943) — Candidate of Science (PhD) in technology, senior scientist, Institute of Informatics Problems, Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation; LPlekhanov@inbox.ru