

Часть II

АРХИТЕКТУРЫ И СИСТЕМНЫЕ РЕШЕНИЯ
ВЫЧИСЛИТЕЛЬНЫХ КОМПЛЕКСОВ И СЕТЕЙ НОВЫХ
ПОКОЛЕНИЙ

УДК 621.3.049.77:004.312

**АНАЛИЗ САМОСИНХРОННОСТИ
ЭЛЕКТРОННЫХ СХЕМ
ФУНКЦИОНАЛЬНЫМ МЕТОДОМ¹**

Л. П. Плеханов

В статье описывается впервые разработанный функциональный метод анализа самосинхронности электронных схем на основе индицируемости сигналов. В отличие от событийных функциональный метод позволяет получить детальную информацию об индицируемости внутренних и внешних сигналов схемы за один сеанс, сразу по всем сочетаниям параметров схемы. Метод дает возможность исследовать более широкий, чем ранее, класс неисправностей, а также открывает путь для анализа схем практически неограниченного размера. Он характеризуется полиномиальной сложностью вычислений от числа сигналов схемы. На основе функционального метода создана программа анализа самосинхронности САМАН.

1. Введение

Анализ самосинхронности представляет собой важнейший и весьма трудоёмкий этап проектирования электронных схем, не зависящих от задержек.

В книге В. И. Варшавского с соавторами [1] вводятся понятия двух типов родственных схем: *самосинхронных* (в оригинале — самосинхронизирующихся, или апериодических, но эти термины не прижились) и *полумодулярных*. Общим для этих схем яв-

¹ Работа выполнена при финансовой поддержке программы ОНИТ РАН «Оптимизация вычислительных архитектур под конкретные классы задач» (проект 1.4).

ляется то, что их правильное функционирование не зависит от величин задержек составляющих элементов.

Основное свойство полумодулярных схем — отсутствие запрещённых переходов элементов из возбуждённого состояния в исходное (известно, что на практике такие переходы могут порождать гонки и сбои). Как видно, это свойство лежит в области событий (переходов из одних состояний в другие), и методы проверки полумодулярности реализуются также в событийной области.

Как показал Д. Маллер [2], полумодулярные схемы должны быть замкнуты, а их основное свойство зависит не только от элементов, но и от начальных состояний.

Самосинхронные схемы по контексту — разомкнутые (в противном случае они были бы тождественны полумодулярным). В качестве таковых в книге [1] рассматриваются исключительно двухфазные схемы — схемы с чередованием рабочей и промежуточной фазы (спейсера). Хотя теоретически существование других видов самосинхронных схем возможно, они не имеют практического значения по ряду соображений. Электронные схемы создаются как инструмент преобразования информации. Для них необходимы как собственно преобразование (информационный канал), так и организация вычислений (организационный канал). В синхронных схемах организацию вычислений обеспечивают тактовый генератор и цепи синхронизации, в самосинхронных — сигналы и цепи индикации. Как показали предыдущие исследования тех же авторов [3], именно двухфазная дисциплина необходима для создания таких безгоночных, а значит и самосинхронных, схем. Искусственно сконструированные самосинхронные схемы без двухфазной дисциплины не имеют информационного канала, т. е. практически бессмысленны.

Следуя традиции книг [1; 3], будем считать, что самосинхронные схемы имеют двухфазную дисциплину.

Основное свойство самосинхронных схем — индицируемость сигналов на выходах схемы, которая не связана непосредственно с событиями, а основывается на свойствах логических уравнений, описывающих элементы.

Понятие полумодулярных схем возникло в теории автоматов; в них игнорируются категории информационного и организационного каналов. При использовании моделей полумодулярных схем на практике возникают трудности как в получении этих

моделей (согласованное замыкание), так и в интерпретации результатов их исследований.

Практические схемы данного класса (независимые от задержек элементов) изначально разрабатываются как самосинхронные, а модель полумодулярных схем привлекается только для проверки их независимости от задержек. Методы и программы такой проверки существуют в настоящее время только для полумодулярных схем. Поэтому разработка метода непосредственного анализа самосинхронных схем без использования полумодулярных моделей представляется актуальной.

Более подробно сравнение подходов к проектированию полумодулярных и самосинхронных схем (событийный и функциональный подходы) приведены в статьях [4; 5].

В данной статье описывается метод и программа анализа частичной и полной самосинхронности по постановке задачи, приведённой в [5].

2. Основы функционального метода

Для анализа предъявляется схема в типовом фазовом представлении [5], т. е. она является двухфазной и на входе и выходе схемы должно быть хотя бы по одному фазовому сигналу. Реальные схемы могут содержать и некоторые дополнительные внешние сигналы, которые легко учитываются и несущественны для метода: установочные, режимные и некоторые другие.

Всем внешним сигналам схемы (портам) должен быть присвоен тип, определяющий роль порта при анализе. Возможные типы портов и условия индикации для них приведены в [5].

2.1. Константы анализа. Информационные сигналы на входах и выходах схемы при самосинхронном кодировании являются связанными. Поэтому каждая пара входных портов типов парафазных со спейсером и бистабильных порождает одну независимую величину, называемую далее *константой анализа*. Аналогично входные мультифазные и мультистабильные порты порождают столько независимых двоичных констант, сколько необходимо для кодирования всех возможных наборов значений этих портов.

Другим видом констант анализа являются произвольные значения переменных памяти, появляющиеся в запоминающих схемах, например триггерах. Эти константы выявляются на предварительном этапе анализа.

Указанных констант достаточно для однозначного вычисления всех значений сигналов схемы, и их совокупность будет исчерпывающе определять условия индицируемости всех сигналов схемы.

Обозначим совокупность констант анализа через \mathbf{D} .

2.2. Проверка индикации. В процессе анализа проверяется индицируемость каждого из сигналов схемы: портов и выходов всех элементов.

Рассмотрим некоторый сигнал схемы A .

Индикация сигнала A предполагает, что если сигнал меняется, то будут меняться и некоторые выходы схемы.

В двухфазной схеме любые изменения сигналов происходят только при смене фазы и не более одного раза при каждой такой смене. Поэтому для анализа индикации достаточно рассматривать два установившихся значения каждого сигнала: в рабочей фазе и спейсере.

Пусть в рабочей фазе сигнал A имеет значение $f_W(\mathbf{D})$, а в спейсере — $f_S(\mathbf{D})$. Эти значения будем называть нормальными в соответствующей фазе.

Сигнал A меняется при изменении фазы только при тех константах анализа, для которых выполняется условие:

$$f_W(\mathbf{D}) \oplus f_S(\mathbf{D}) = 1. \quad (1)$$

Поскольку переходов два, то следует говорить о двух видах индикации каждого сигнала: индикация в рабочей фазе — при переходе из спейсера в рабочую фазу — и индикация в спейсере — при переходе из рабочей фазы в спейсер.

Для проверки индикации сигнала будем изменять его нормальное значение и сравнивать нормальные и изменённые значения выходных фазовых сигналов. Для удобства целесообразно свести выходные фазовые сигналы в один оценочный (виртуальный) индикаторный сигнал I известными способами локальной индикации и применения Г-триггеров. В общем случае сигнал I будет зависеть от констант анализа $I = I(\mathbf{D})$.

Предположим, что необходимо проверить индикацию сигнала A в рабочей фазе и нормальное значение оценочного сигнала в этой фазе: $I_W(\mathbf{D}) = 1$ (общий индикаторный сигнал с необходимостью должен быть одинаков при всех константах анализа). Придадим сигналу A в рабочей фазе изменённое значение F , га-

рантированно отличающееся от его нормального значения в этой фазе. С учётом условия (1) это будет:

$$F = \hat{f}_W(\mathbf{D})f_S(\mathbf{D}). \quad (2)$$

Несимметричность этого выражения относительно фаз подтверждает и несимметричность понятия индицируемости: индикация сигнала в рабочей фазе будет отличаться от его индикации в спейсере.

Далее по функциям элементов вычисляются изменённые значения сигналов схемы, фазовых выходов и изменённый оценочный сигнал $I_F(\mathbf{D})$.

Полученное значение оценочного сигнала полностью решает вопрос об индикации сигнала A в рабочей фазе. Возможны следующие варианты:

1) $I_F(\mathbf{D}) = 0$. При любых константах анализа это значение противоположно нормальному. Сигнал A индицируется всегда;

2) $I_F(\mathbf{D}) = 1$. При любых константах анализа сигнал A не индицируется, так как оценочное значение совпадает с нормальным и схема «не замечает» изменения;

3) оценочный сигнал является функцией констант анализа. Тогда уравнение

$$I_F(\mathbf{D}) = 1 \quad (3)$$

задаёт условие на константы анализа, при которых нарушается индикация.

Аналогично проводится проверка индицируемости и в спейсере.

2.3. Вычислительная сложность процедуры. Известно, что вычисление логических функций имеет экспоненциальную сложность относительно числа их аргументов. Поэтому по отношению к константам анализа сложность приведённой процедуры составит

$$C_K = O(2^K), \quad (4)$$

где K — эффективное число констант. За счёт того, что переменные памяти, как правило, разные в рабочей фазе и спейсере, величину K можно выразить следующим образом:

$$K = K_{\text{in}} + \max(M_W, M_S), \quad (5)$$

где K_{in} — количество констант, порождённых входными портами, M_W и M_S — количество переменных памяти соответственно в рабочей фазе и спейсере.

Пусть схема содержит N элементов. Проверка индицируемости каждого выхода элемента требует вычисления изменённых значений максимально всех N функций. В результате для вычислительной сложности относительно числа элементов можно дать оценку:

$$C_N = O(N^2). \quad (6)$$

2.4. Связь с фиксацией неисправностей. Фиксация (обнаружение) неисправностей необходима для построения надёжных вычислительных устройств.

Рассматривая выражение (2), можно заметить, что оно в точности описывает одиночную константную неисправность сигнала (выхода элемента или входа схемы) типа залипание (ОКНЗ) при переходе из спейсера в рабочую фазу. Аналогичное выражение легко записывается и для перехода из рабочей фазы в спейсер. Вычисление оценочного сигнала представляет собой способ обнаружения неисправности.

Ранее неоднократно высказывалась мысль (например, в [1]), не требующая особых доказательств, что самосинхронные схемы фиксируют свои ОКНЗ. Обратное утверждение нигде не упоминалось. Изложенная выше процедура анализа конструктивно доказывает и это обратное утверждение, в результате чего можно сделать важный обобщающий вывод.

Анализ самосинхронности схемы на основе индикации и определение фиксации ОКНЗ сигналов схемы эквивалентны.

Процедура анализа инвариантна к выражению в правой части (2), и её можно использовать для исследования фиксации и других логических неисправностей, отличных от ОКНЗ. Такая возможность значительно расширяет покрытие неисправностей при создании надёжных устройств.

Следует отметить, что в событийных методах анализа отсутствует какое-либо понятие, связанное с неисправностями, и поэтому исследование фиксации неисправностей для них недоступно.

2.5. Иерархический анализ. Анализ самосинхронности больших, практически значимых схем целиком (за один сеанс) представляет собой актуальную проблему, не решённую к настоящему моменту. Анализ на основе индикации предоставляет возможность её решения.

Если все внутренние сигналы схемы индицируются, то, используя свойство транзитивности индикации, можно проводить анализ самосинхронности каждого устройства однократно. Анализ схемы более высокого уровня иерархии уже не потребует проверки внутренних, ранее проверенных сигналов, а ограничится учётом индикации только портов проверенных устройств.

Такой иерархический анализ невозможен с помощью событийных методов, которые требуют участия всех охваченных анализом элементов.

Подходящее разбиение схем на блоки (устройства) с целью избежать чрезмерной сложности (4) позволит в будущем выполнять анализ схем практически неограниченного размера.

3. Алгоритм анализа

Алгоритм анализа делится на две части: подготовка и тестирование.

На этапе подготовки вычисляются все данные, необходимые для проверки индикации сигналов. Подготовка состоит из следующих шагов:

1. Проверка монотонности всех функций элементов по каждой переменной. Это необходимое для индикации требование;
2. Вычисление входных констант анализа;
3. Определение компактных множеств элементов, охваченных обратными связями. Для каждого такого множества выполняются следующие действия:
 - а) определение входов и выходов множества, составление системы логических уравнений;
 - б) решение уравнений, определение устойчивости (отсутствие самогенерации) множества и выявление переменных памяти. При обнаружении неустойчивости анализ прекращается;
 - в) добавление переменных памяти к константам анализа;
 - г) приведение уравнений множества к комбинационной форме;
4. Вычисление значений всех сигналов в спейсере и рабочей фазе в зависимости от констант анализа;
5. Вычисление значений оценочного сигнала в спейсере и рабочей фазе и их проверка. Оба этих значения должны быть равны 0 или 1 и быть взаимно-обратными, иначе констатируется неверное построение схемы.

После выполнения подготовки делается собственно проверка индицируемости сигналов — тестирование.

На этом этапе тестируются все внутренние и некоторые внешние сигналы схемы в соответствии с условиями индикации, приведёнными в [5]. Тестирование проводится отдельно в рабочей фазе и спейсере.

При проверке каждого сигнала выполняются следующие действия (на примере рабочей фазы):

1. Проверка условия (1). Если условие (1) противоречиво (левая часть тождественно равна 0), то это означает, что данный сигнал никогда не меняется и потому не должен тестироваться (возможно, это ошибка схемы);

2. Присваивание сигналу значения (2), расчёт новых значений сигналов схемы и оценочного сигнала;

3. Формирование по п. 2.2 заключения об индицируемости сигнала и о частичной или полной самосинхронности схемы.

4. Программа анализа схем на самосинхронность САМАН

На основе описанного функционального метода впервые разработана несобытийная программа анализа самосинхронности САМАН. Программа функционирует в составе САПР самосинхронных схем РОНИС.

Исходными данными программы служат задание на анализ, VHDL-описание схемы (иерархическое любой глубины) и библиотека базовых элементов. В описании схемы могут использоваться как базовые элементы, так и непосредственно логические функции в формате языка VHDL. Библиотека базовых элементов содержит информацию о логических функциях всех элементов.

VHDL-описание схемы должно включать необходимые атрибуты типов портов, значений спейсеров и некоторые другие.

Программа проводит полный анализ схемы и выдаёт заключение об её полной или частичной самосинхронности. В случае нарушения индикации внутреннего сигнала приводятся условия, при которых это нарушение возникает.

Программа на практических примерах показала свою достоверность и эффективность.

5. Заключение

1. Впервые разработан несобытийный функциональный метод анализа самосинхронности на основе индицируемости сигналов.

2. Метод позволяет получить детальную информацию об индицируемости внутренних и внешних сигналов схемы за один сеанс, сразу по всем сочетаниям констант анализа схемы.

3. Метод характеризуется экспоненциальной сложностью вычислений от числа констант анализа и полиномиальной сложностью порядка 2 от числа сигналов схемы.

4. Метод позволяет прямо определять фиксацию не только одиночных константных неисправностей типа залипание (ОКНЗ), но и любых других логических неисправностей. (Событийные методы косвенно определяют фиксацию только ОКНЗ.)

5. Функциональный метод открывает путь для анализа схем практически неограниченного размера.

6. На основе функционального метода создана программа анализа самосинхронности САМАН, проверенная на реальном проекте.

Список литературы

1. Автоматное управление асинхронными процессами в ЭВМ и дискретных системах / Под ред. В.И. Варшавского. М.: Наука, 1986.
2. *Muller D.E., Bartky W.C.* A theory of asynchronous circuits // Proc. Int. Symp. on the Theory of Switching, Part 1. Harvard University Press, 1959. P. 204–243.
3. Аперiodические автоматы / Под ред. В.И. Варшавского. М.: Наука, 1976.
4. *Плеханов Л.П.* Проблемы функционального подхода в проектировании самосинхронных схем // Системы и средства информатики. Вып. 15. М.: Наука, 2005. С. 329–337.
5. *Плеханов Л.П.* Самосинхронность и задачи анализа строго самосинхронных электронных схем // Системы и средства информатики. Вып. 17. М.: Наука, 2007. С. 492–502.