

РЕАЛИЗАЦИЯ ФУНКЦИОНАЛЬНОГО МЕТОДА АНАЛИЗА САМОСИНХРОННОСТИ ЭЛЕКТРОННЫХ СХЕМ

Л.П. Плеханов

В статье описываются вопросы расширенной реализации впервые разработанного функционального метода анализа самосинхронности электронных схем. Использование свойства индицируемости сигналов в данном методе позволяет определить независимость работы схемы от задержек ее элементов (самосинхронность), а также получить необходимую информацию об индицируемости внешних и внутренних сигналов схемы. В расширенной реализации, помимо собственно анализа самосинхронности, производится вычисление параметров, необходимых и достаточных для применения схемы без ее раскрытия в самосинхронных схемах более высокого уровня. Расширенные задачи реализованы в программе анализа ФАЗАН, которая, как и предыдущая реализация, характеризуется полиномиальной сложностью вычислений от числа элементов.

1. Введение

Строго самосинхронные схемы (далее – просто *самосинхронные*) – это схемы, правильное поведение которых, в смысле безошибочного логического преобразования информации, не зависит от задержек ее элементов. В зарубежной литературе их обычно называют *self-timed* или *delay independent* (общепринятой терминологии нет). Они относятся к классу асинхронных схем, но имеют ряд уникальных свойств по сравнению с другими видами внутри класса [1].

Одной из основных проблем проектирования самосинхронных схем, не решенных в настоящее время, остается анализ «больших» схем, состоящих из нескольких десятков элементов и больше. Существующие событийные методы пока не справляются с подобными задачами.

Как показано в [2], возможность блочного, иерархического анализа (а значит, анализа любых по сложности схем) открывает функциональный подход. Поэтому работы в данном направлении представляются актуальными.

Известны реализации событийных методов анализа: ТРАНАЛ, ТРАСПЕК (созданные группой В.И. Варшавского) [3], БТРАН,

АСИАН (созданные в ИПИ РАН) [4]. Программы ТРАНАЛ, БТРАН, АСИАН реализуют метод Маллера в полных состояниях, и потому размер анализируемых схем весьма ограничен. Программа ТРАСПЕК основана на предложенном В.И. Варшавским методе диаграмм изменений, что в принципе позволяет увеличить объем схем. Однако эта реализация воспринимает только весьма узкий класс схем – так называемые дистрибутивные схемы [5]. Кроме того, доступные версии программ ТРАНАЛ и ТРАСПЕК содержат ошибки, морально устарели и не запускаются в современных операционных системах. Недостатком приведенных реализаций является также и собственный нестандартный язык описания схем.

Реализации событийных методов дают очень мало информации об анализируемой схеме: только «да» – в случае полумодулярности – и «нет» плюс указание на запрещенный переход в случае ее нарушения. Такое свойство требует дополнительных исследований при проектировании самосинхронных схем.

Одной из особенностей событийных методов является необходимость согласованного замыкания схемы (соединения выходов и входов схемы по определенным правилам) перед началом анализа [5]. Правила замыкания хотя и не сложны, но формально нигде не описаны. Все постановки задач анализа событийными методами даются уже для замкнутых схем.

Смысл замыкания состоит в самосинхронной имитации внешней для схемы среды. Как правило, при анализе каждой схемы используется простейшее для нее замыкание. Однако в ряде случаев такое замыкание не позволяет выявить нарушения самосинхронности (пример дан в конце статьи). Таким образом, при применении событийных методов возникает еще одна задача, внешняя по отношению к самим методам, но совершенно необходимая для практики, – задача выбора замыкания, достаточного для проверки самосинхронности схемы. Данная задача не только не исследована, но и должным образом не сформулирована.

Функциональный подход не опирается на событийные модели типа сигнальных графов, диаграмм переходов и изменений, сетей Петри и других, а базируется на описании схем в виде логических функций. Замыкания схемы в этом подходе не требуется. В зарубежной литературе работы, посвященные подобному подходу, не встречаются.

Концептуальная основа функционального метода анализа самосинхронных схем приведена в статьях [2, 6]. В методе используется прямой способ вычисления индикации. Этот способ

имеет ряд методических преимуществ. Во-первых, автоматически учитываются требования теории [5], относящиеся к переходам – допустимость, правильность и другие, во-вторых, он дает возможность «попутно» получить информацию о внешних сигналах, необходимую для иерархического анализа.

В предыдущей реализации функционального метода – программе САМАН [7] – выполнялся только анализ самосинхронности схемы. Однако необходимость развития анализа в сторону иерархического принципа (без раскрытия схем) требует дополнительной информации о внешних сигналах. Как оказалось, эту информацию целесообразно вычислять одновременно с анализом самосинхронности рассматриваемой схемы.

В результате существенной модернизации предыдущей версии появилась новая программа анализа ФАЗАН. При ее разработке ставились две главные цели: собственно анализ самосинхронности и подготовка информации о внешних сигналах для иерархического анализа.

Постановка задачи анализа осталась прежней [2].

2. Расширенные задачи программы

Как и ранее, анализируются разомкнутые схемы в типовом фазовом представлении [2].

Разработка практических самосинхронных схем показала, что для будущего иерархического анализа необходимо ввести дополнительные характеристики внешних сигналов, прежде не использовавшихся при анализе.

Вспомогательные индикаторы – это сигналы, индицирующие часть схемы (как правило, входные цепи) и направляемые обратно в схему-источник информации с целью ускорения взаимодействия.

Эквивалентные индикаторы – это выходные фазовые сигналы, имеющие в точности одинаковые индицирующие свойства. Например, такими сигналами являются прямой и инверсный (если есть) выходы Г-триггера.

Произвольные значения спейсеров. Ряд схем, например, Г-триггеры с унарными входами, допускают любое значение спейсера – 0 или 1, что должно учитываться при анализе на верхнем уровне.

Установочные режимы. Эти режимы применяются для начальной установки схем с памятью. Установки бывают обычные

(синхронные) – без каких-либо проверок, и самосинхронные – с проверкой установки ячеек памяти в нужные значения. В последнем случае требуется анализ самосинхронности этих режимов.

Задачи программы по собственно анализу самосинхронности подробно изложены в [2]. Новые задачи по обеспечению иерархического анализа заключаются в следующем.

1) Вычисление спейсерных значений выходных фазовых сигналов. Это очевидные параметры для состыковки анализируемой схемы с другими схемами.

2) Расчет транзитных фаз входных и выходных бистабильных сигналов. Эти параметры также необходимы для правильного сопряжения схем, поскольку несогласованность транзитных фаз при последовательном соединении с необходимостью приводит к нарушению самосинхронности.

3) Проверка произвольной дисциплины изменений фазовых входов. Результат работы схемы не должен зависеть от порядка изменений этих входов. Такой расчет является необходимым элементом проверки самосинхронности как анализируемой, так и внешней схемы.

4) Расчет индикации внешних сигналов. Сведения об индикации внешних сигналов требуются для анализа индицируемости внешней схемы без раскрытия данной.

5) Анализ самосинхронности установочных режимов.

3. Программа ФАЗАН

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

Алгоритм анализа программы ФАЗАН включает в себя как подмножество этапы алгоритма предыдущей версии [7]. Обобщенно алгоритм состоит из следующих шагов:

- подготовка необходимых данных для расчетов,
- вычисление значений выходов элементов в спейсере и рабочей фазе,
- проверка правильности значений выходов схемы по категориям: фазовые выходы, бистабильные выходы,

- проверка произвольной дисциплины входных фазовых сигналов схемы,
- расчет транзитных фаз входных и выходных бистабильных сигналов,
- расчет и оценка индикации внутренних и внешних сигналов,
- анализ самосинхронности установочных режимов,
- вывод пользователю результатов анализа,
- запись результатов анализа в файл описания на VHDL в виде соответствующих атрибутов для использования на верхних уровнях.

Как и предыдущая программа САМАН, программа ФАЗАН интегрирована в САПР самосинхронных схем РОНИС.

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

Программа на практических примерах показала свою правильность и эффективность.

4. Пример анализа

На рис. 1 показана схема, первоначально задуманная как самосинхронный триггер с усиленными выходами.

При анализе этой схемы событийной программой БТРАН с обычным (простейшим) замыканием было получено заключение о ее полумодулярности (аналога самосинхронности для замкнутых схем). Лишь более сложное замыкание, в котором при каждом изменении фазы меняются входы R и S, позволило найти нарушение полумодулярности. Однако это уже искусственный прием, поскольку сложное замыкание выводит схему из множества рабочих значений.

Программа ФАЗАН сразу определила нарушение самосинхронности при изменении сигналов Q и QB.

Данный пример показывает различие между понятиями полумодулярности и самосинхронности (на основе индицируемости). В большинстве случаев они эквивалентны как критерии независимости от задержек. Однако, если происходит только одно изменение сигнала (в примере Q или QB), то такое изменение никогда не нарушает полумодулярность, а индицируемость может нарушаться. Для практики имеет смысл именно индицируемость.

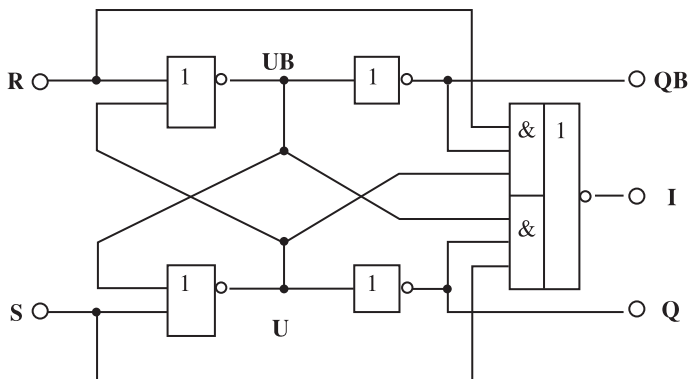


Рис. 1. Схема триггера с усиленными выходами (с нарушением самосинхронности). R и S – парафазные входы, Q и QB – бистабильные выходы, I – индикатор

Таким образом, понятие полумодулярности не покрывает всех потребностей практики, так как существует целый ряд схем с однократными переходами, которые в исходном виде невозможно проверить с помощью этого понятия на независимость от задержек. На это обстоятельство впервые обратил внимание Ю.А. Степченков.

Использование понятия индицируемости в программе ФАЗАН позволяет анализировать все подобные случаи, в частности, встроенные самосинхронные предустановки схем с памятью.

5. Заключение

1. Разработана расширенная реализация несобытийного функционального метода анализа самосинхронности на основе индицируемости сигналов – программа ФАЗАН.

2. Введены новые интерфейсные характеристики схем, ранее не использовавшиеся при анализе и необходимые для иерархического анализа без раскрытия схем.

3. В реализации решаются задачи как собственно анализа схемы на самосинхронность, так и вычисления данных для иерархического анализа на верхнем уровне.

4. Для схем с однократными переходами (например, самосинхронных предустановок в схемах с памятью) событийные программы анализа не выявляют нарушений независимости от задержек без специальных искусственных манипуляций. Программа ФАЗАН определяет подобные нарушения стандартным образом.

Список литературы

1. *Филин А.В., Степченко Ю.А.* Компьютеры без синхронизации // Системы и средства информатики. М.: Наука, 1999. Вып. 9. С. 247–261.
2. *Плеханов Л.П.* Самосинхронность и задачи анализа строго самосинхронных электронных схем // Там же М.: Наука, 2007. Вып. 17. С. 492–502.
3. *Kishinevsky M., Kondratyev A., Taubin A., Varshavsky V.* Analysis and identification of speed-independent circuits on an event model // Formal Methods in System Design. 1994. Vol. 4, N 1. P. 33–75.
4. *Рождественский Ю.В., Морозов Н.В., Степченко Ю.А., Рождественские А.В.* Универсальная подсистема анализа самосинхронных схем // Системы и средства информатики. М.: Наука, 2006. Вып. 16. С. 463–475.
5. Автоматное управление асинхронными процессами в ЭВМ и дискретных системах / Под ред. В.И. Варшавского. М.: Наука, 1986. 398 с.
6. *Плеханов Л.П.* Проблемы функционального подхода в проектировании самосинхронных схем // Системы и средства информатики. М.: Наука, 2005. Вып. 15. С. 329–337.
7. *Плеханов Л.П.* Анализ самосинхронности электронных схем функциональным методом // Там же. М.: Наука, 2008. Вып. 18. С. 225–233.