

Методы тестирования СБИС с использованием скан-технологии

1 Концепция скан-технологии

Основная концепция скан-технологии состоит в объединении элементов памяти СБИС (триггеры или защелки) в т.н. «сканирующие цепочки» («цепи сканирования», scan chains), сдвиговые операции через которые позволяют контролировать и управлять состояниями объекта контроля (ОК, device under test, DUT).

Запоминающие элементы цифровой схемы сначала заменяются на элементы сканирования, которые затем объединяются в один или несколько сдвиговых регистров, называемых цепями сканирования, с целью обеспечения внешнего доступа к ним.

Для этого в проект СБИС вводится дополнительная логика (scan insertion):

- простые ячейки памяти (триггеры или защелки) заменяются на т.н. «ячейки сканирования» (scan cells).
- ячейки объединяются, формируя одну или несколько цепочек.

На рисунках 1 и 2 представлен простой 4-разрядный счетчик до и после ввода соответствующей дополнительной логики (scan insertion) соответственно.

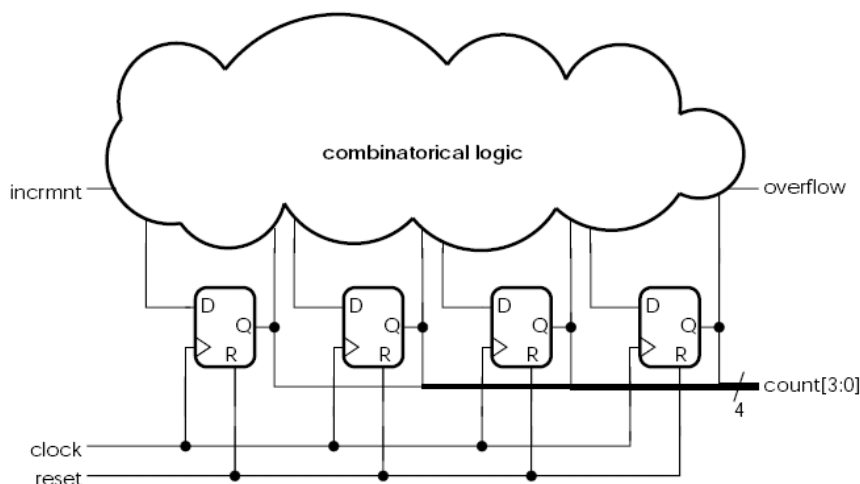


Рисунок 1. 4-х разрядный счетчик до «scan insertion»

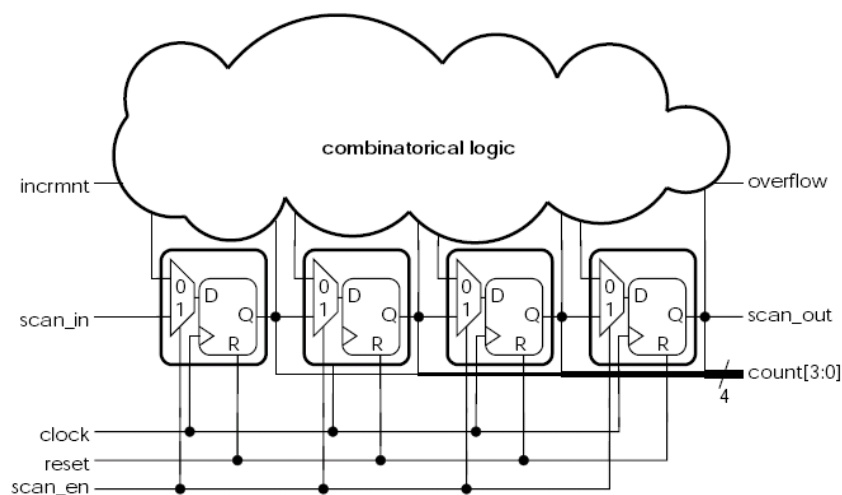


Рисунок 2. 4-х разрядный счетчик после «scan insertion»

Скан-цепочка обслуживается дополнительным входом «scan_in» и дополнительным выходом «scan_out». Вход «scan_en» управляет режимом работы ячеек сканирования следующим образом:

- scan_en= 0..... режим функционирования (functional/mission), нормальная работа;
- scan_en= 1..... режим сканирования, обеспечивается сдвиг (shifting) данных через цепочку.

В общем случае процесс «скан-тестирования» выполняется следующим образом:

- в скан-цепочку «вдвигаются» (загрузка, loading) данные для непосредственной установки необходимого состояния DUT;
- выполняется один или несколько тактов нормальной работы (functional/mission) для:
 - запуска (launch) определенных событий в DUT
 - захвата (capture) ответа DUT;
- опционально выходы DUT проверяются на правильные значения;
- результирующее состояние (ответ) «выдвигается» (выгрузка, unloading) наружу через скан-цепочку и сравнивается с ожидаемым состоянием.

ПРИМЕЧАНИЕ: выдвигание последовательности (scan pattern) результата из скан-цепочки на выход и загрузка со входа следующей последовательности воздействия могут осуществляться *одновременно*.

Как правило, цепи сканирования состоят из сотен или тысяч ячеек, а используемая частота сдвига – ниже, чем частоты, используемые для функционального контроля (ФК, functional test) ОК.

Время тестирования ОК зависит, главным образом, от количества скан-последовательностей, размера самой большой цепочки сканирования и частоты сдвига, и в меньшей степени от частоты, используемой во время запуска/захвата (launch/capture).

В общем случае «скан-тестирование» является более эффективным по сравнению с «классическим» ФК за счет возможности непосредственного доступа к «внутренним» ключевым триггерам проекта.

2 Скан-архитектуры

Основными скан-архитектурами для тестирования комбинационной логики являются:

– архитектура типа «Mixed-D Flipflop», представленная на рисунке 3, в которой запоминающие элементы модифицируются в мультиплексные D-триггеры: вход данных (поступающих с выхода тестируемой комбинационной схемы) и вход сканирования (служащий для записи определенной тестовой последовательности в сдвиговый регистр) являются входами мультиплексора, сигнал с выхода которого поступает на вход D-триггера;

– архитектура типа «clocked-scan», в которой выбор входа элемента сканирования осуществляется с помощью синхросигнала, в отличие от предыдущего случая, где выбор осуществлялся специальным сигналом;

– архитектура типа «Level Sensitive Scan Design» (LSSD), представленная на рисунке 4, в которой запоминающие элементы преобразуются в сдвиговые регистры, управляемые уровнем;

– расширенная архитектура, в которой каждый элемент сканирования состоит из D-защелки и мультиплексного D-триггера.

–

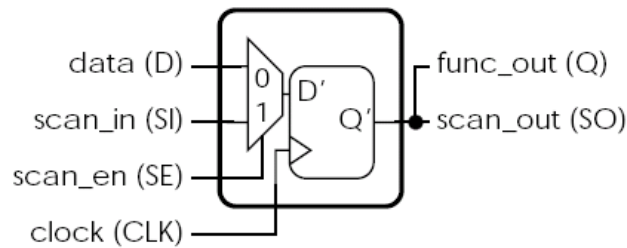


Рисунок 3. Типовая ячейка Mixed-D Flipflop

Архитектура Mixed-D используется наиболее широко, т.к. ячейки такого типа лишь незначительно увеличивают используемую площадь кристалла.

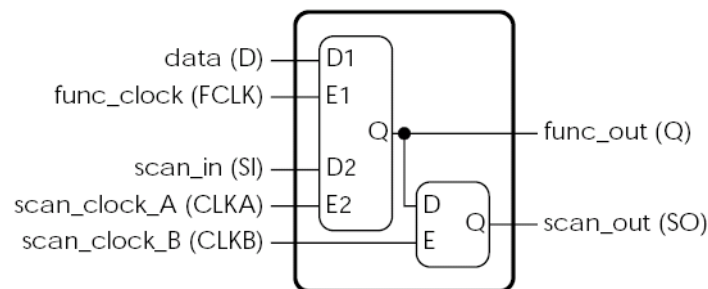


Рисунок 4. Типовая ячейка LSSD

Архитектура LSSD обладает рядом плюсов и минусов и обычно используется в высокоскоростных проектах, т.к., например, не вносит дополнительных задержек в пути распространения данных.

3 Полное и частичное сканирование

В современных разработках СБИС стремятся практически каждый триггер/защелку заменять на соответствующую ячейку сканирования, с целью увеличения тестового покрытия (fault coverage) и уменьшения размера тестовых векторов. Такой подход называется «дизайн с полным сканированием» (Full Scan Design).

На практике применяется также и «частичное сканирование» (partial scan), при котором не все триггеры/защелки заменяются ячейками сканирования и включаются в цепочки (обычно из соображений обеспечения меньшей площади или большего быстродействия кристалла). Надо понимать, что это приводит к уменьшению тестового покрытия и увеличению длины тестовых последовательности.

Учитывая, что стоимость одного транзистора и стоимость занимаемой им на кристалле площади стремительно снижается с каждым годом, представляется, что гораздо выгоднее окажется выделить на кристалле «дешевое» место под тестовую логику, сэкономив при этом «дорогие» время и память Тестера АТЕ.

4 Автоматическая генерация тестовых последовательностей

Автоматическая генерация тестовых последовательностей (АГТП, automatic test pattern generation, ATPG) – это процесс создания последовательностей для тестирования схемы, который точно описан сетевой топологией логического уровня (схематически).

Система автоматической генерации тестов используется для формирования тестовых воздействий и предсказания реакции на них исправной схемы. Набор тестовых воздействий формируется таким образом, чтобы выявить максимальное число неисправностей, вызванных дефектами технологического процесса, дефектами материалов и монтажа.

Условно говоря, АГТП-алгоритмы «вводят» неисправность в схему, и затем используют разнообразные механизмы, чтобы активировать неисправность и распространить её по схеме и пронаблюдать на выводе.

Эти алгоритмы обычно работают с программным генератором неисправностей, который создает минимальный сжатый список неисправностей. Таким образом, разработчику нет необходимости беспокоиться о генерировании неисправностей.

5 Модели неисправностей

В программном обеспечении ATPG принято рассматривать следующие модели неисправностей (fault model) СБИС.

5.1 Модель константной неисправности

Модель константной неисправности (неисправность типа «залипание», «stuck-at faults», SSF) является стандартной (классической) моделью для генерации тестов. Данная

модель предполагает, что дефект в схеме выражается узлом, значение сигнала на выходе которого зафиксировано в нуле либо единице.

5.2 Модель переходной неисправности

Модель переходной неисправности (gate delay fault model, transition, at-speed) используется для генерации тестовых шаблонов для обнаружения неисправности, связанной с медленным переходом конкретного узла схемы из состояния логического нуля в состояние логической единицы и наоборот.

Задаётся время, необходимое для перехода, поэтому, если переход не происходит или происходит вне выделенного времени, предполагается дефект временных характеристик ОК.

Соответствующий скан-тест (At-Speed Test) предназначен для проверки, что ОК может «работать достаточно быстро» или для градуации его быстродействия.

5.3 Модель неисправности, связанной с задержками

Модель неисправности, связанной с задержками (path delay model), характеризует критические пути распространения сигналов в схеме.

Для больших проектов учесть все существующие пути невозможно. Поэтому АТПГ должна рассматривать только критические пути, выявленные, например, в результате выполнения статического временного анализа (CVA, static timing analysis, STA).

5.4 Модель неисправности, связанной с замыканием

Модель неисправности, связанной с замыканием (bridging fault mode, deterministic bridging) необходима для тестов на наличие коротких замыканий между цепями внутри ОК.

На основе набора геометрических правил создается список пар цепей в проекте СБИС, которые потенциально могут иметь замыкания (или просто взаимовлияния).

При этом считается, что цепь «агрессор» через замыкание (bridge) может определять состояние цепи «жертва».

5.5 Модель неисправности, связанная с отклонением величины тока потребления IDDQ в установившемся режиме

IDDQ-тестирование основано на измерении тока потребления (I_{dd}) в состоянии покоя (когда цепи не переключаются и входные сигналы ОК пребывают в статических состояниях). Тестовые скан-последовательности переводят ОК в различные predetermined состояния. При измерении тока I_{dd} в каждом из этих статических положений можно выявить неисправности, приводящие к существенному отклонению силы тока от ожидаемой.

Преимуществом IDDQ-тестирования является возможность выявления множества ошибок с использованием очень небольшого количества скан-последовательностей.

Недостатком является увеличение времени тестирования за счет выполнения относительно медленных операций по измерению тока.

6 Статическое и динамическое скан-тестирование

В промышленности схемы, основанные на скан-технологии, часто используются для тестирования на малых частотах с целью обнаружения константных неисправностей (п. 5.1). Такие скан-тесты обычно называют «DC scan test» (статическими), т.к. они не зависят от временных характеристик (длительности фронта, спада и задержки распространения).

Однако основная ценность скан-технологии (особенно при тестировании микросхем, выполненных по нанометровому технологическому процессу) заключается в возможности тестирования высокоскоростных цифровых схем на их рабочей частоте. Скан-архитектуры, предназначенные для тестирования данных схем, позволяют обнаруживать неисправности, связанные с задержками (определить, является ли критический путь слишком медленным, п. 5.3), и переходными неисправностями (п. 5.2). Такие скан-тесты обычно называют «AC scan test» (динамическими).

7 Формирование синхросигнала сканирования

При выполнении скан-тестирования синхросигнал (тактовые импульсы, Scan Clock) сдвига данных (загрузки и выгрузки скан-цепочек) обычно формируется Тестером АТЕ. Синхросигнал же функционального режима ОК при запуске/захвате может формироваться как Тестером, так и самим ОК.

В частности, для обеспечения при скан-тестировании такой же скорости работы ОК в циклах запуска/захвата (launch/capture), что и в нормальном (рабочем) режиме, могут потребоваться высокие частоты тактовых импульсов. Если возложить формирование таких высоких (до нескольких ГГц) частот на Тестер АТЕ, то для качественной передачи синхросигнала на вход ОК потребуются использование сверхвысокоскоростных каналов и тщательное обеспечение целостности сигналов (signal integrity) на оснастке. Более того, возникают дополнительные конструктивные требования к ОК в части передачи через его выводы (pad) ВЧ-синхросигнала на кристалл. Такой подход может оказаться очень дорогостоящим или даже невозможным.

Часто ОК, работающий на высоких скоростях в нормальном режиме, содержит выделенный тактовый генератор, что может быть использовано для формирования высоких частот, необходимых при тестировании переходных неисправностей (At-Speed Test, п. 5.2). В проект СБИС должен быть добавлен соответствующий модуль для управления короткими последовательностями тактовых импульсов. Обычно этот модуль перенастраивается для каждой скан-последовательности (сколько импульсов и на какой частоте формировать). Таким образом Тестер должен обеспечить лишь формирование опорной частоты (для генератора синхросигналов на ОК).

Учитывая, что тактовый генератор на ОК может быть неисправным, обычно DFT-проект СБИС поддерживает также режим обхода (bypass), чтобы тактовые импульсы низкой частоты, используемые при запуске/захвате (launch/capture) могли формироваться классическим способом с использованием Тестера АТЕ.

8 Сжатие тестовых данных

При тестировании с применением детерминированного набора тестов, сгенерированного АТРГ, возникает задача уменьшения требований к памяти автоматического испытательного оборудования (АТЕ) и еще более важная задача – уменьшения общего времени тестирования.

Для решения указанных задач целесообразно использовать т.н. «сжатие тестовых данных» (Scan Compression), позволяющее сократить объем входных воздействий и выходных реакций в 10, 100 или более раз, что приводит, в частности, к передаче меньшего количества данных между Тестером АТЕ и тестируемым ОК по каналам с ограниченной пропускной способностью.

Возможность сжатия данных основано на том факте, что скан-тесты состоят из двух основных типов скан-последовательностей:

- последовательности, покрывающие *легко обнаруживаемые* неисправности, – почти каждый бит, «выдвигаемый» из скан-цепочки, позволяет обнаружить неисправность (группу неисправностей).

- последовательности, покрывающие *трудно обнаруживаемые* неисправности, – последовательность «заточена» на обнаружение небольшого количества неисправностей. Большинство битов, «вдвигаемых» в скан-цепочку и «выдвигаемых» из скан-цепочки не вносят вклад в обнаружение неисправности и, следовательно, могут быть «безболезненно» изменены.

Первый тип обычно представляет собой набор из сотен или нескольких сотен скан-последовательностей, в то время как существуют тысячи последовательностей второго типа. После установки неиспользуемых битов в «правильное» состояние скан-последовательности можно легко сжать с использованием соответствующих алгоритмов. При этом они зачастую превращаются в скан-последовательности, «покрывающие *легко обнаруживаемые* неисправности», так что необходимость в использовании скан-последовательностей первого типа может даже отпасть.

Сжатие тестовых данных поддерживается путем добавления в проект СБИС специального устройства «перед» цепями сканирования, распаковывающего тестовые воздействия, и устройства «после» цепей сканирования, сжимающего данные, представляющие собой реакцию ОК.

9 Различие между «Boundary Scan Test» и «Scan Test»

Скан-тестирование (Scan Test) используются для тестирования внутренней логики ОК, в то время как «периферийное сканирование» (Boundary Scan Test, JTAG) изначально фокусировалось на управлении выводами ОК для проверки связей между микросхемами на печатной плате.

Архитектура «периферийного сканирования» описана в стандарте IEEE 1149.1 и включает в себя 4-х или 5-ти выводной тестовый порт (Test Access Port, TAP), контроллер тестирования и некоторые дополнительные функции.

Для скан-тестирования такого стандарта *не существует*. Поэтому очень часто TAP-контроллер стандарта IEEE 1149.1 дополнительно используется для включения и настройки режимов скан-тестирования. Для этого контроллер «расширяется» дополнительными нестандартными инструкциями, при этом изменение TAP-порта не требуется.