

Ами ГОРОДЕЦКИЙ, к. т. н. (JTAG.TEST)
amigo@jtag-test.ru

Тестирование компонент памяти в технологии JTAG. Часть 1

Несмотря на то, что JTAG-тестирование разнообразных компонент памяти (ЗУ) давно и со значительным успехом применяется для проверки исправности их монтажа на ПП, взрывообразно расширяющийся и обновляющийся спектр компонент ЗУ, их типов и технологий доставляет множество хлопот тест-инженеру, поскольку любые такие новшества неизбежно обуславливают те или иные нюансы при разработке JTAG-теста, и как правило — в сторону усложнения. Актуальность этой темы не утрачивается с течением времени, поэтому данную и несколько последующих колонок я посвящу рассмотрению некоторых важных аспектов JTAG-тестирования компонент памяти.

Традиционные методы тестирования ЗУ заключаются в использовании встроенного программного обеспечения для прогона функциональных тестов, а задача разработки теста возлагается при этом на программистов. Программа тестирования начинается, естественно, с проверки исправности монтажа ИС на ПП, несовпадение записываемого в ИС ЗУ набора векторов со считанным из ЗУ набором указывает на наличие неисправности монтажа, а алгоритмический анализ результатов прогона теста может указать на характер неисправности и ее местоположение. Если тест исправности монтажа ИС проходит, программа тестирования приступает к выполнению функционального теста ЗУ. Такой подход вполне достаточен для схем, содержащих микропроцессоры или микроконтроллеры общего назначения, программное обеспечение которых может быть произвольным, в том числе включающим и само-тестирование ИС ЗУ. Если же ИС ЗУ связаны со специализированными процессорами, такими как DSP, GPU (Graphics Processor Unit),

NPU (Network Processor Unit), или управляются от FPGA, встроенное программное обеспечение которых вовсе не предназначено для выполнения операций тестирования ЗУ, приходится искать другие методы.

Внутрисхемное тестирование (ICT), устраняющее зависимость от встроенного программного обеспечения микропроцессорных схем при тестировании ИС ЗУ, широко применяется в подобных ситуациях в течение длительного времени, а зачастую актуально и сегодня. При этом, разумеется, предполагается наличие полного физического (контактного) доступа ко всем цепям ИС ЗУ посредством специального адаптера. Однако с ростом степени интеграции корпусов ИС (BGA и т.д.) контактный доступ к цепям ИС ЗУ на поверхности ПП оказался затруднен или вовсе исчез, что, собственно, и обусловило широкое применение технологии JTAG для их тестирования.

Традиционный подход к JTAG-тестированию ИС ЗУ, которые, как правило, не содержат структур JTAG, заключается в использовании окружающих их ИС JTAG (рисунок

[ПЭ.2007.8]. Шины адреса и данных, а также сигналы управления ЗУ, приводимые в действие контактами, снабженными ячейками JTAG, могут быть связаны с различными ИС цепочки ГС или же сконцентрированы в одной из них, например в микропроцессоре. На рисунке показана часть физических дефектов монтажа ИС ЗУ, которые могут быть обнаружены JTAG-тестами. Все существующие автоматизированные системы генерации JTAG-тестов (АСГТ) [ПЭ.2008.2, ПЭ.2008.3] содержат постоянно пополняемые библиотеки моделей для ИС ЗУ различных типов. Каждая такая модель описывает алгоритм доступа к ЗУ и протокол записи/чтения, хотя модели ЗУ одной из АСГТ неприменимы, как правило, для других, и взаимозаменяемость между ними нет. Разработка JTAG-тестов ЗУ на базе библиотечных моделей сводится к описанию ограничений, вносимых в модель общего вида данным окружением и условиями функционирования тестируемой ИС ЗУ в данной схеме ПП. Зачастую требуется также проведение отладки тайминга в алгоритме доступа к ЗУ, меняющегося от схемы к схеме. Если в функциональном режиме синхросигнал ЗУ подключен к внутрисхемному осциллятору, то в режиме JTAG-теста этот синхросигнал должен быть тем или иным способом переключен на один из предназначенных для этого JTAG-контактов, как правило — на свободный контакт ПЛИС. Причина заключается в том, что тест ЗУ в структурных JTAG-тестах проводится со строгой привязкой к синхросигналам ЗУ, генерируемым самой JTAG-структурой цепочки, управляющей прогоном такого теста.

Одной из особенностей JTAG-тестов межэлементных связей (interconnect) для ИС ЗУ является необходимость выполнения нескольких операций записи для обеспечения переходов между состояниями встроенного контроллера ИС памяти еще до того, как записываемые данные (тест-вектор) доберутся до ячеек памяти. Иными словами, обеспечение каждой

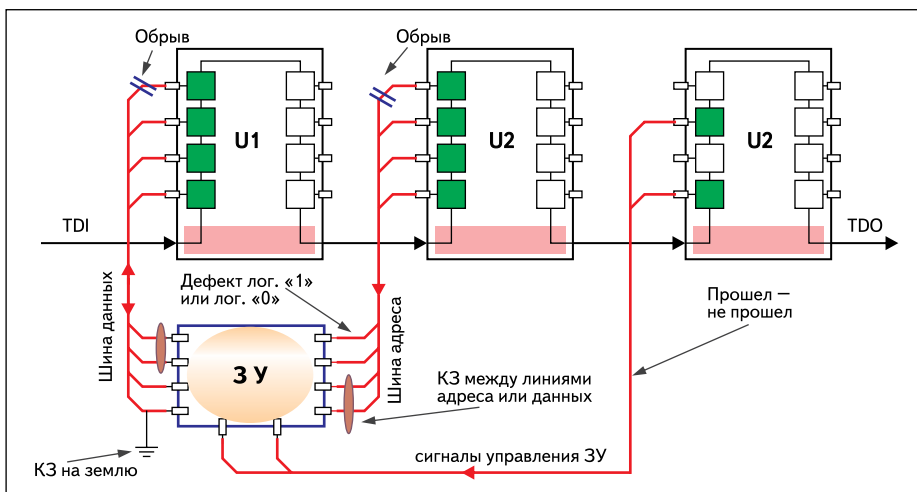


Рисунок. Типичная структура JTAG-тестирования ИС ЗУ

операции записи или чтения требует ввода нескольких векторов JTAG-сканирования. Другая особенность таких тестов — прямая зависимость результата теста (запись/чтение) от исправности отдельных цепей управления, таких как /WE, /CE и т.д., что существенно затрудняет формулирование алгоритма точной диагностики результатов теста.

Все многообразие существующих сегодня ИС ЗУ с точки зрения сложности их JTAG-тестирования можно свести к трем классам — ПЗУ, статические ЗУ и динамические ЗУ. Первые два из них, включающие в себя разнообразные ИС EEPROM, SRAM, NVRAM, Flash (NOR и NAND), не создают заметных проблем при их JTAG-тестировании и позволяют выполнять устойчивые тесты при сравнительно низкочастотных синхросигналах в несколько мегагерц, генерируемых JTAG-структурами.

Существенным отличием динамических ЗУ, представленных очень широким и непрерывно пополняющимся диапазоном ИС (SDRAM, RLDRAM, DDR SDRAM, DDR2 SDRAM, DDR3 SDRAM, FB-DIMM и т.д.), является необходимость периодической регенерации их содержимого. Именно этот тип ИС ЗУ является наиболее проблематичным с точки зрения JTAG-тестирования, поскольку надежность сохранения информации этими ЗУ напрямую зависит от соблюдения тайминга регенерации. Более того, многие из современных динамических ИС ЗУ требуют выполнения определенной последовательности команд инициализации перед началом «диалога» с ними для настройки их интерфейса в соответствии со схемой, внутри которой они размещаются. Важно отметить, что любые отличия в структуре интерфейса с ИС одного и того же типа в двух разных схемах могут обусловить отличия в деталях теста для этой ИС ЗУ.

В таблице приведен пример лишь одного цикла записи в ЗУ DDR2 SDRAM типа MT47H32M8*, требующий выполнения 20 векторов JTAG-сканирования. Строки рисунка представляют собой, в сущности, фрагмент модели ЗУ этого типа в любой АСГТ, описывающий алгоритм функционирования цикла записи при подаче параллельных векторов на контакты тестируемой ИС. Это своего рода эмуляция средствами JTAG функций любого контроллера ЗУ, включающая в себя также и эмуляцию тайминга, различного для разных типов ЗУ. Для обеспечения диагностики неисправностей на шине адреса обычно применяются адресация типа «бегущая 1 или 0» и подобным же образом организованные данные на шине данных.

Процесс отладки теста ЗУ DRAM с моделью общего вида любой АСГТ в конкретной схемной реализации этой ИС зачастую оказывается весьма трудоемким, так как падение теста на заведомо исправной ИС ЗУ может указывать на несоответствие применяемой моде-

Таблица. Один цикл записи ИС ЗУ DDR2 SDRAM MT47H32M8

	Команда	BA[1..0], A[12..0]	D[7..0]	CLK	CLKN	CKE	CAS	RAS	CS	WE	DM0	DQS0	Примечание
1	ACT	АБ, адрес строки	xxxxxxx	0	1	1	1	0	0	1	x	z	Активизация строки
2		АБ, адрес строки	xxxxxxx	1	0	1	1	0	0	1	x	z	
3	NOP	xxxxxxxxxxxxxx	xxxxxxx	0	1	1	1	1	0	1	x	z	Цикл NOP
4		xxxxxxxxxxxxxx	xxxxxxx	1	0	1	1	1	0	1	x	z	
5	WRITE	АБхх1номер_колонки	xxxxxxx	0	1	1	0	1	0	0	x	z	АБ=адрес банка, А10=1 (предзарядка)
6		АБхх1номер_колонки	xxxxxxx	1	0	1	0	1	0	0	x	z	
7	NOP	xxxxxxxxxxxxxx	xxxxxxx	0	1	1	1	1	0	1	x	z	Цикл NOP
8		xxxxxxxxxxxxxx	xxxxxxx	1	0	1	1	1	0	1	x	z	
9	WPRE	xxxxxxxxxxxxxx	Данные	0	1	1	1	1	0	1	0	0	Запись данных
10		xxxxxxxxxxxxxx	Данные	1	0	1	1	1	0	1	0	1	
11	MSK	xxxxxxxxxxxxxx	xxxxxxx	0	1	1	1	1	0	1	1	0	DM0=1
12		xxxxxxxxxxxxxx	xxxxxxx	1	0	1	1	1	0	1	1	1	
13	MSK	xxxxxxxxxxxxxx	xxxxxxx	0	1	1	1	1	0	1	1	0	DM0=1
14		xxxxxxxxxxxxxx	xxxxxxx	1	0	1	1	1	0	1	1	1	
15	MSK	xxxxxxxxxxxxxx	xxxxxxx	0	1	1	1	1	0	1	1	0	DM0=1
16		xxxxxxxxxxxxxx	xxxxxxx	1	0	1	1	1	0	1	1	1	
17	NOP	xxxxxxxxxxxxxx	xxxxxxx	0	1	1	1	1	0	1	x	z	NOP
18		xxxxxxxxxxxxxx	xxxxxxx	1	0	1	1	1	0	1	x	z	
19	NOP	xxxxxxxxxxxxxx	xxxxxxx	0	1	1	1	1	0	1	x	z	NOP
20		xxxxxxxxxxxxxx	xxxxxxx	1	0	1	1	1	0	1	x	z	

ли таймингу тестируемой схемы и тестового стенда в целом. Факторами такого несоответствия могут являться структура управления данной ЗУ DRAM по каналам JTAG, характеристики компьютера тестового стенда, параметры интерфейса применяемого контроллера JTAG, структура JTAG-цепочки и некоторые другие.

Критическим параметром при отладке устойчивого теста является частота подачи тестовых векторов. Следует учитывать, к примеру, что для ИС ЗУ типа MT47H32M8 и им подобных шаг регенерации содержимого ячеек во всех строках и во всех столбцах матрицы памяти должен составлять примерно 7,8 мкс. Если применяемая в тесте частота ТСК равна 40 МГц (весьма высокая частота, далеко не всегда достижимая), а результирующая длина активных JTAG-регистров, участвующих в тесте (регистров граничного сканирования PGC и регистров обхода PO), равна 1000 бит, то частота подачи тестовых векторов на ИС ЗУ будет примерно равна 40 кГц, или 25 мкс на тестовый вектор. Как показано в таблице, один цикл синхросигнала CLK эмулируется двумя тестовыми векторами, а весь цикл записи (в одну ячейку!) займет 20×25 = 500 мкс, что совершенно не соответствует спецификации ИС ЗУ. Тест, тем не менее, может работать, поскольку рассматриваемая ИС ЗУ снабжена механизмом автоматической предзарядки (precharge), или регенерации строки матрицы памяти, выполняемой перед записью в ячейку данной строки внутри самого цикла записи (векторы 5 и 6 в таблице), что устраняет необходимость в специальном этапе регенерации.

Если тестируемая ИС ЗУ имеет схемное управление не от микропроцессора или микроконтроллера общего назначения, а от FPGA, это может обусловить дополнительные трудности. Одна из них заключается в том, что для обеспечения рабочего режима схемы FPGA обычно предварительно конфигурируются, тем самым создавая необходимый интерфейс с периферийными по от-

ношению к ней ИС памяти. В таких случаях невозможно обеспечить правильную функциональную JTAG-эмуляцию доступа к тестируемым компонентам памяти без предварительного конфигурирования FPGA, обеспечивающего необходимые уровни сигналов, их терминацию, тайминг и т.д. Следует иметь при этом в виду, что для многих типов ИС FPGA файлы BSDL [ПЭ.2007.7] различны для состояний ИС до и после процесса конфигурирования. Другая трудность может заключаться в длине PGC [ПЭ.2007.6] ИС FPGA, достигающей для современных ИС такого типа значительных величин. Например, длина PGC для FPGA Stratix-IV типа EP4SE530H40 фирмы Altera составляет 2970 бит! JTAG-сканирование тест-векторов сквозь цепочку, состоящую хотя бы из одной такой ИС, может привести к критическому падению частоты синхронизации при эмуляции протокола доступа к ИС ЗУ, и тест просто не будет работать. Это тем более критично, если в дополнение к ИС FPGA в JTAG-цепочке находятся и другие микросхемы.

В качестве одного из решений для подобной ситуации может быть рекомендовано создание псевдо-PGC внутри ИС FPGA, содержащего ровно столько JTAG-ячеек, сколько необходимо для создания интерфейса с тестируемой ИС ЗУ. В роли такого псевдо-PGC может выступить любой из регистров общего назначения, обеспечиваемых современными FPGA. Необходимо только позаботиться о подготовке кода конфигурирования интерфейсного сопряжения контактов FPGA, участвующих в тестировании ИС ЗУ, и выполнить предтестовое конфигурирование FPGA уже с этим новым кодом. Исходный файл BSDL данной ИС FPGA уже не может быть, конечно, использован, и понадобится написать новый, содержащий псевдо-PGC и псевдо-TAP-контроллер. Схемные соединения линий JTAG-цепочки в схеме должны быть, разумеется, подключены к новым контактам псевдо-TAP-контроллера ИС FPGA. ■

* Treuren B.G., van. Problems Using Boundary-Scan for Memory Cluster Tests. International Test Conference, IEEE, 2008.