

JTAG-тестирование

(часть 2)

Иосиф Каршенбойм (Санкт-Петербург)

Данная статья представляет собой обзор материалов, посвящённых тестированию плат и устройств по интерфейсу JTAG. В первой части статьи приводилось описание технологии граничного сканирования и было показано её место в «жизни» разрабатываемого изделия. В этой части статьи начато рассмотрение команд тестирования, выдаваемых по интерфейсу JTAG, и структуры BSDL-файла.

ДИАГРАММА TAP-КОНТРОЛЛЕРА ИЗ ПАКЕТА ПРОГРАММ XILINX ISE

В качестве программы для изучения этого пункта работы интерфейса JTAG можно использовать симулятор, входящий в пакет Xilinx ISE.

Ниже приводится подробный перечень действий, которые необходимо выполнить, чтобы воспользоваться симулятором:

1. Выбираем Configure Device (iMPACT), нажимаем правую кнопку мыши, выбираем пункт Open Without Updating;
2. Запускаем Configure Device (iMPACT);

3. Получаем сообщение Launching Application for process Configure Device (iMPACT);
4. Если не выбран режим Boundary-Scan, то выбираем Mode>Configuration mode и переключаемся так, чтобы «попасть» в Boundary-Scan;
5. Запускается iMPACT, закладка Boundary-Scan;
6. Появляется окно Configure Device;
7. Выбираем радиокнопку Boundary-Scan Mode, нажимаем Next;
8. Выбираем радиокнопку Enter a Boundary-Scan Chain и нажимаем Finish;
9. Add Device – выбираем Cancel;
10. Выбираем кнопку 9 слева – Debug Chain;

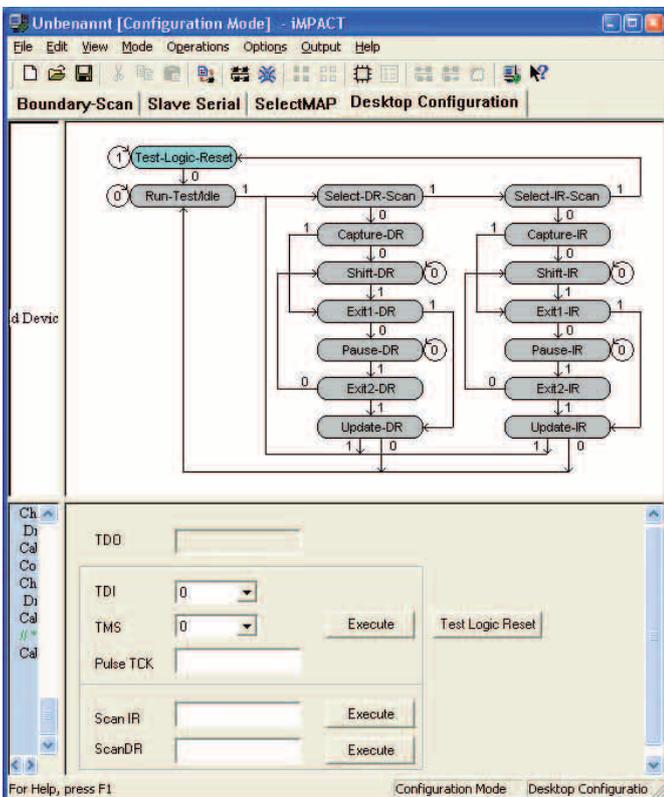


Рис. 11. Структура управляющего автомата JTAG-сканирования на примере программы iMPACT

11. Программа сообщает, что не нашла кабель;
12. Заходим в Cable Comm. Выбираем Setup или Cancel – здесь можно выбрать любой режим;
13. После этого справа откроется диаграмма TAP-контроллера.

На рис. 11 приведён фрагмент программы iMPACT, входящей в состав пакета проектирования Xilinx ISE. Эту программу можно использовать как для целей обучения, так и для проверки изделий, содержащих микросхемы с JTAG-портами, поскольку программа позволяет вводить и считывать биты данных через аппаратный кабель сопряжения.

ДИАГРАММА ПЕРЕХОДОВ, ПРОДОЛЖЕНИЕ

Если пользователь после загрузки регистра команд хочет работать с данными, то он может из состояния автомата Update_IR сразу перейти в состояние Select_DR. Этот переход на рис. 12 выделен зелёным цветом.

Теперь необходимо сказать несколько слов о том, какие команды выполняет TAP. Эти команды условно можно разбить на несколько групп, в зависимости от того, что выполняет данная команда и кто встроил в TAP функциональный узел, с которым команда работает. Первая группа команд, пока-

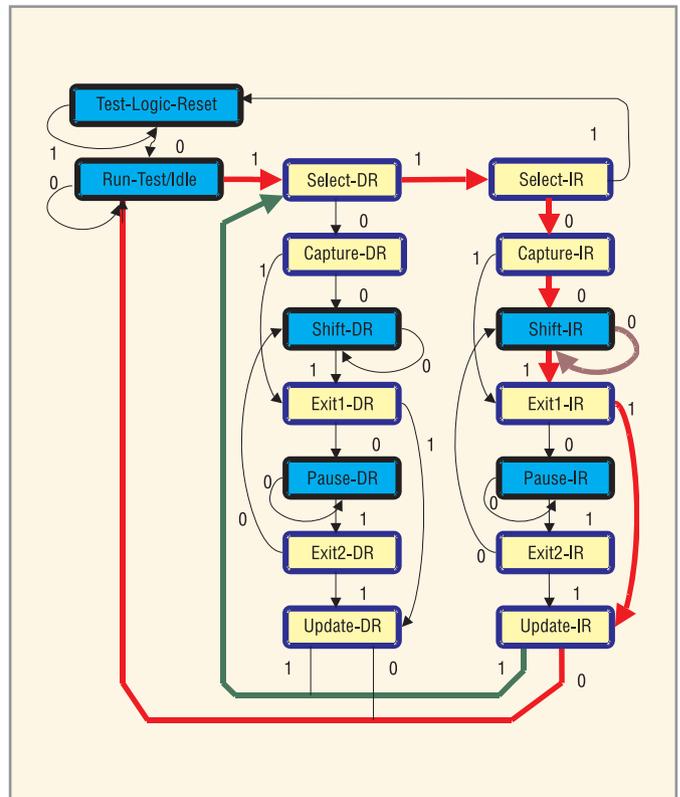


Рис. 12. Диаграмма переходов, необходимых для загрузки в контроллер новой команды

Таблица 2. Описание форматов инструкций

Номер	Название	Описание
1	INSTRUCTION_LENGTH	Определяет длину регистра инструкций и, соответственно, число бит, передаваемых (сдвигаемых) в TAP
2	INSTRUCTION_OPCODE	Строки, в которых приводятся коды команды. Правый бит каждой из этих строк первым сдвигается в TDO
3	INSTRUCTION_CAPTURE	Строка, показывающая те данные, которые будут приняты при сдвиге команды в режиме Capture-IR Стандартное значение первых двух принятых битов – 01
4	INSTRUCTION_PRIVATE	Строки, показывающие дополнительные команды, определённые производителем для каждой конкретной микросхемы. Использовать данные команды можно только так, как это рекомендует и разрешает разработчик микросхемы. Для остальных пользователей, не получивших разрешение и рекомендации по использованию данных команд, использование таких команд не рекомендовано

занная в табл. 1 (см. часть 1), – это обязательные команды, выполняемые интерфейсом. Адресация этих команд, порядок их выполнения и функциональность узлов, связанных с ними, жёстко оговорена стандартом.

Из обязательных команд, команды EXTEST, INTEST, SAMPLE/PRELOAD и BYPASS уже были кратко упомянуты выше. А сейчас необходимо более подробно проанализировать команду BYPASS. После получения этой команды TAP «превращается» в один триггер, включенный между входом TDI и выходом TDO. При этом данные на выходе TDO относительно входа TDI появляются с задержкой в один такт частоты TCK. Для того чтобы точно знать, какие именно команды выполняет та или иная микросхема, необходимо посмотреть BSDL-файл, который её описывает. Но осторожно, дорогой читатель! Учтите, что тот файл, который вы взяли сегодня с сайта фирмы-производителя, завтра может быть заменен на другой. Это не важно, что файл помечен как проверенный. Важно только то, что фирма не сообщает о замене файлов. Именно так поступает, например, фирма Xilinx, которая не сообщает о замене BSDL-файлов. Поэтому всегда необходимо проверять, не произошли ли изменения в документации. Ещё одно хорошее место для поисков BSDL-файлов – это тот пакет программ, с которым вы работаете, т.к. обычно среди прочих папок находится и скромная папка с названием BSDL.

Команды и структура BSDL-файла

BSDL – это VHDL-подобный файл, описывающий регистр сканирования и его параметры. В файле приведены коды команд и их длина. Просмотреть файл можно обычным текстовым редактором. Одним из разделов BSDL-файла является раздел, характеризующий команды для микросхемы. Описание этого раздела приведено в табл. 2.

Далее приведена часть BSDL-файла, описывающего ресурсы микросхемы BlackFin BF533. Данный файл можно получить на сайте фирмы – производителя микросхемы.

И здесь необходимо сделать одно замечание для читателей. Как мы знаем из описания стандарта BSDL, в BSDL-файле текстовая строка должна заканчиваться разделителем в виде знака точки с запятой – «;». А в том случае, если необходимо разорвать эту текстовую строку и продолжить её на другой строчке текстового листа, применяется знак «&». Однако формат журнала не позволяет печатать текст в колонке длиной больше, чем 32 символа. Поэтому чтобы не «искажать» текст исходного файла, здесь будут выполнены переносы строк, что по форме не соответствует формату BSDL-файла.

```
attribute INSTRUCTION_LENGTH of
ADSP_BF533: entity is 5;

-- Unspecified opcodes
--
assigned to Bypass.
attribute INSTRUCTION_OPCODE
of ADSP_BF533: entity is
    "BYPASS (11111),» &
    "EXTEST (00000),» &
    "SAMPLE (10000),» &
    "IDCODE (00010),» &
    "MEMBIST (01010),»
&
    "EMULATION
(00100,10100,
01000,11110,01100),» &
    "CUSTOMER_KEY (10110),»
&
    "TESTKEY
(00110)";
```

Таблица 3. Обязательные команды интерфейса

Команда	Функция
Extest	Тестирование производится для сигналов, находящихся снаружи микросхемы
Sample	Приём данных
Preload	Предустановка данных
Bypass	Передача данных со входа на выход через регистр обхода. Задержка при передаче данных – 1 такт синхрочастоты

```
attribute
INSTRUCTION_CAPTURE
of ADSP_BF533:
entity is «00001»;

attribute INSTRUCTION_PRIVATE
of ADSP_BF533: entity is
    "EMULATION,» &
    "MEMBIST,» &
    "CUSTOMER_KEY,» &
    "TESTKEY» ;

attribute IDCODE_REGISTER
of ADSP_BF533: entity is

    "0010» &
Version
    "0010011110100101» &
    "00001100101» &
    "1";
```

Тут можно увидеть разделы, описывающие коды команд и их символические имена. Как мы видим, в описании микросхемы BlackFin BF533 имеется набор стандартных команд и набор команд, «добавленных» производителем. Это несколько команд, входящих в разделы INSTRUCTION_OPCODE и INSTRUCTION_PRIVATE. Эти команды предназначены для тестирования и отладки программ в BlackFin BF533.

Начнём с рассмотрения обязательных команд интерфейса (см. табл. 3). Об этих командах мы уже говорили, поэтому здесь никаких дополнительных комментариев не приводится.

Ко второй группе команд, показанной в табл. 4, относятся дополнительные команды, выполняемые интерфейсом. Адресация этих команд, порядок их выполнения и функциональность узлов, связанных с ними, также в большинстве случаев жёстко оговорена

стандартом. Исключения в этой группе составляют команды Usercode и Runbist. К сожалению, в рамках журнальной статьи нет возможности подробно описывать все дополнительные команды интерфейса. Да и выполнение таких команд, как Runbist, целиком зависит от конкретной микросхемы. Из этой группы мы более подробно рассмотрим только одну команду – Idcode. После получения этой команды, TAP немедленно готов «в данных» выдавать на выходе TDO значение идентификационного кода микросхемы.

Данные, получаемые при выполнении команды «чтение идентификационного кода», могут быть в соответствии с IEEE STD 1149.1 представлены в виде следующих полей:

- Manufacturers ID – код производителя;
- Part Size Code – код, описывающий микросхему в данной серии;
- Family Code – код серии микросхем;
- Revision Code – код ревизии изделия.

В табл. 5 приведены примеры кодировки полей идентификационного регистра для нескольких микросхем. Более полную информацию о кодировке производителей микросхем можно получить из документа [17].

Поскольку регистр Boundary Scan сдвиговый, то очень важно, в какую сторону производится сдвиг. Для JTAG-технологии всё, что находится в регистрах, сдвигается ВПРАВО. Поэтому иногда в документации наблюдается лёгкая путаница с младшими и старшими битами. И если в одном месте данные для сдвига изображены вот так:

```
attribute INSTRUCTION_OPCODE
of XC2V250_FG456 : entity is
```

```
"EXTTEST (000000) ,» &
"SAMPLE (000001) ,» & ... ,
```

то это значит, что, например, команда SAMPLE выдаётся на TDI как 1, потом 0, и затем, 0,0,0,0. В документации иногда можно встретить совершенно разные кодировки команд, путаницу в обозначениях старших и младших битов в поле команды и т.п.

Несколько более подробно следует описать необходимость приёма «лог. 1» в последнем поле. Как было сказано выше, данные сдвигаются вправо, младшим битом вперед. Поэтому программа, обслуживающая работу с портом, должна отслеживать появление «лог. 1» как первого бита данных в строке ID. После чего необходимо принять оставшиеся 31 бит. А как определить, когда появится нужная нам единица? Сканирование данных будет выполнено после сканирования команды. А код, выдаваемый, например, микросхемой BlackFin BF533, – «00001».

Таким образом, последние биты, принятые после сканирования команды – нули, а первый бит сканирования данных – единица. По этому поводу необходимо также сделать дополнительное замечание. Когда программа получает из микросхемы данные, соответствующие идентификационному коду, необходимо учитывать, что эти данные были «защиты» в кристалле. А кристалл «не знает», в каком корпусе он разварен. Таким образом, кристаллы с одинаковыми объёмами FPGA, но находящиеся в разных корпусах, будут выдавать одинаковые идентификационные коды. И, следовательно, программа может определить тип микросхемы, серию, но пользователь не имеет воз-

можности точно определить тип корпуса и выбрать BSDL-файл, соответствующий данному корпусу.

В последнюю группу команд выделим команды, добавленные производителем для повышения функциональности выпускаемых микросхем. Адресация этих команд, порядок выполнения и функциональность связанных с ними узлов не оговорена стандартом. К этим командам относятся команды, позволяющие работать с внутрисхемными эмуляторами, встроенными в микросхемы, с технологическими проверочными узлами, которые производители используют для контроля параметров микросхем. К этой же группе мы отнесём и те встроенные узлы, которые производители FPGA предоставляют в распоряжение пользователей.

СОСТОЯНИЕ МИКРОСХЕМЫ ПРИ ПЕРЕДАЧЕ КОМАНДЫ

Во время передачи команды в микросхему мы можем прочитать из неё данные. Эти данные описаны в BSDL-файле, в разделе INSTRUCTION_CAPTURE. Посмотрим, как описываются ресурсы микросхемы BlackFin BF533.

Из этой микросхемы мы прочитаем строку «00001». Единственное, что мы можем сказать по этим данным, так это то, что операция чтения прошла правильно. Совсем по-другому обстоит дело у FPGA фирмы Xilinx, например, у микросхемы Virtex2.

```
attribute INSTRUCTION_CAPTURE of
XC2V1000_BG575 : entity is
-- Bit 5 is 1 when DONE is
released (part of startup
sequence)
-- Bit 4 is 1 if house-cleaning
is complete
-- Bit 3 is ISC_Enabled
-- Bit 2 is ISC_Done
«XXXX01»;
```

Так, для микросхемы Virtex2-XC2V1000, в соответствии с её BSDL-файлом, при передаче команды в микросхему из неё читаются следующие данные:

- бит 5 = 1, когда сигнал DONE «отпущен» и не удерживается в низком уровне;
- бит 4 = 1, когда произошла очистка внутренней памяти;
- бит 3 = 1, когда конфигурация решена;

Таблица 4. Дополнительные команды интерфейса

Команда	Функция
Intest	Тестирование производится внутри микросхемы
Idcode	Команда чтения идентификационного кода. Действует всегда, с момента включения микросхемы
Usercode	Команда чтения кода, загруженного пользователем
Runbist	Выполнить встроенный тест самопроверки
Clamp	Данные на выходах зафиксированы
HighZ	Выходы находятся в третьем состоянии

Таблица 5. Примеры кодировки полей идентификационного регистра микросхем

Поле	Названия полей идентификационного регистра				
	Revision Code	Family Code	Part Size Code	Manufacturers ID	LSB
	31 - 28	27 - 21	20 - 12	11 - 1	0
Virtex XC2V250	XXXX	0001000	000011000	00001001001	1
Virtex XC2V1000	XXXX	0001000	000101000	00001001001	1
ADSP BF533	0010	0010011	110100101	00001100101	1
Altera EPF10K20R240	0000	0001000	000100000	00001101110	1

- бит 2 = 1, когда конфигурация выполнена.
- Именно по битам этой строки можно точно узнать, завершилась ли операция загрузки FPGA.

Продолжение следует

ЛИТЕРАТУРА

1. <http://www.analog.com>.
2. www.eltech.spb.ru.
3. Boundary Scan Coach. GOEPEL Electronic. <http://www.goepel.com>.
4. <http://www.pld.ttu.ec/applets>.
5. <http://www.universalscan.com>.
6. Платунов А.Е., Постников Н.П., Чистяков А.Г. Механизмы граничного сканирования в неоднородных микропроцессорных системах. Chip News. http://lmt.cs.ifmo.ru/article_chip_news.html.
7. Рустин В., Городецкий А. Разделяй и властвуй – принцип граничного сканирования. Chip News. http://chip-news.gaw.ru/html.cgi/arhiv/01_06/stat-3.htm.
8. Каршенбойм И. Виртуальные кнопки и светодиоды, или Неизвестное обо всем известном JTAG сканировании. Компоненты и технологии. 2005. № 6.
9. <http://www.national.com/appinfo/scan/index.html>.
10. IEEE 1149.1 (JTAG) Boundary-Scan Testing for Stratix II Devices. Altera. Chapter 9. www.altera.com.
11. Kuznetsov D. JTAG Boundary-Scan Test – introduction. http://www.orc.ru/~dkuzn/j_intro.htm.
12. <http://jtagtools.sourceforge.net/download.html>.
13. IEEE Standard Test Access Port and Boundary-Scan Architecture. IEEE Std 1149.1-2001.
14. Boundary-Scan Test and In-System Programming Software. Corelis. http://www.corelis.com/products/Test_Software.htm.
15. Serial Vector Format Specification. ASSET InterTech. Texas Instruments. www.asset-intertech.com/support/svf.pdf.
16. www.amontec.com. JTAG Interface: Common Pinouts amt_ann003 (v1.1). Application Note
17. EIA/JEP106, JEDEC Publication 106, Standard Manufacturer's Identification Code.
18. Каршенбойм И. Микропроцессор своими руками/4. Как отладить встроенный в FPGA микроконтроллер? Компоненты и технологии. 2006. № 11. 

Новости мира News of the World Новости мира

Рынок ТВ: из японцев верит в «плазму» только Panasonic

Позиция Matsushita Electric Industrial в отношении плазменных телевизионных панелей известна. Компания, владеющая знаменитой торговой маркой Panasonic, является крупнейшим поставщиком «плазмы» и занимает около трети этого рынка. Более того, она планирует наращивание производства. В то же время другие крупные японские бренды, такие как Pioneer и Hitachi, занимающие уверенные позиции на рынке «плазмы», постепенно теряют к нему интерес, – утверждает тайваньский индустриальный рупор DigiTimes.

Сообщается, что компания решила повременить с инвестициями в строительство нового завода по выпуску плазменных телевизоров. По данным агентства Reuters, Hitachi также намерена на некоторое время приостановить разработку планов по усилению производственных мощностей для «плазмы». Принятие окончательного решения по этому вопросу руководством Hitachi отложено до конца года. Всё это происходит на фоне общего роста продаж плазменных телевизоров.

Hitachi делает ставку сразу на две конкурирующие дисплейные технологии: плазменную и ЖК. Благодаря совместному предприятию Fujitsu Hitachi Plasma Display (FHP), японская компания имеет возможность держаться на рынке телевизионной «плазмы». За позиции Hitachi на рынке ЖК-телевизоров следует благодарить компанию IPS Alpha Technology. Pioneer на сегодняшний день является приверженцем только плазменных плоскостельных телевизоров.

Компания Matsushita, сделавшая в январе заявление о строительстве завода для «плазмы», в недалёком будущем грозит стать монополистом среди японских производителей в этой области. У потребителя может возникнуть ошибочное мнение, что ЖК-дисплеи в большинстве случаев значительно лучше и предпочтительнее.

digitimes.com

Рынок полупроводников: ожидаемый спад в первой половине 2007 г.

Аналитик Брюс Дизен, специалист компании Terra Securities ASA, заявил, что в 2007 г. рост рынка полупроводников не превысит отметки в 7%. Серьезную роль в его исследовании сыграли аналитические данные декабря 2006 г., представленные организацией Semiconductor Industry Association. Последний месяц 2006 г. нельзя назвать прогрессивным и успешным. Среди аналитиков бытует мнение, что первая половина 2007 г. не будет отличаться высокими показателями в этой сфере. Возможно, свой отпечаток оставит сезонный фактор и стабильный рост начнётся во второй половине этого года.

Если оценить данные продаж полупроводников за три прошедших месяца, то можно увидеть медленную, но стабильную тенденцию снижения продаж. Этот факт подтверждают финансовые данные – падение объёмов продаж с \$22,55 млрд. в ноябре до \$21,75 млрд. в декабре. Дизен предполагал, что в декабре рынок полупроводников ознаменуется показателем в \$21,9 млрд., и это немного выше фактических данных. Среднегодовой рост продаж полупроводников составил 5,9%. Но если срав-

нить поквартальные показатели, очевиден спад. В этот период продажи мобильных телефонов, игровых консолей Playstation 3, цифровых фото- и видеокамер не оправдали оптимистичных прогнозов производителей. Продажи процессоров для ПК по сравнению с ноябрем оказались практически на том же уровне, а поставки чипов памяти DRAM даже немного выросли, хотя роста отгрузок DRAM ждали.

eetimes.com

К 2012 г. рынок OLED-дисплеев достигнет отметки в 10,9 млрд. долл.

Согласно опубликованному на днях отчёту специалистов компании NanoMarkets, сегмент OLED-дисплеев может достичь оборотов в \$10,9 млрд. к 2012 г., а спустя два года – в \$15,5 млрд. Почти половина этой суммы будет приходиться на мобильные коммуникационные устройства, поскольку одним из основных преимуществ OLED является низкое энергопотребление. Эту технологию сегодня рассматривают как одну из наиболее перспективных для использования в дисплеях, поскольку кроме низкого энергопотребления она обеспечивает высокий уровень яркости и быструю реакцию при переключении пикселей. OLED-дисплеи могут изготавливаться на гибком субстрате, что открывает совершенно новые горизонты для разработчиков мобильных устройств, причём не только телефонов, но и ноутбуков. Специалисты отмечают, что технология OLED должна найти широкое применение и в приборах освещения. Этот сегмент они оценивают в \$1 млрд. к 2014.

<http://itc.ua>