

SVF

S ЭРИАЛ ЭКТОР F ОРМАТ

S ХАРАКТЕРИСТИКА

JTAG | В ИСТОРИЯ S МОГУ

Т ОН ДЕ-ФАКТО СТАНДАРТ

ОСМОТР Е

Пользователям этого документа предоставляется право копировать и использовать информацию, содержащуюся в документе, бесплатно. Пользователям запрещается вносить изменения в документ или распространять документ под другим именем.

© 1997-2013 гг., АКТИВ ЯНТЕР ТЕХ , INC .

©1994, Т РАЗДРАЖАЮЩИЕ ИНСТРУМЕНТЫ

Содержание

Введение	5
Описание файла SVF	6
Пример SVF-файла	6
Краткое описание команд	7
Синтаксис	8
Состояния подключения IEEE 1149.1	8
Указание действительных чисел	9
Команды ENDDR, ENDIR	9
Синтаксис	9
Цель	9
Параметры	9
Примеры	10
Общая информация	10
Частотная команда	10
Синтаксис	10
Цель	10
Параметры	10
Примеры	10
Общая информация	11
Команды HDR, HIR (регистр данных заголовка, регистр инструкций заголовка)	11
Синтаксис	11
Цель	11
Параметры	11
Примеры	12
Общая информация	12

Команда PIO (параллельный ввод/вывод)	13
Синтаксис	13
Цель	13
Параметры	13
Пример	14
Команда PIOMAP (Карта параллельного ввода/вывода)	14
Синтаксис	14
Цель	14
Параметры	14
Примеры	15
Команда RUNTEST	15
Синтаксис	15
Цель	15
Параметры	16
Примеры	17
Команды SDR, SIR (регистр данных сканирования, регистр инструкций сканирования).....	18
Синтаксис	18
Цель	18
Параметры	18
Примеры	19
Общая информация	19
ГОСУДАРСТВЕННАЯ команда	19
Синтаксис	19
Цель	19
Параметры	19
Примеры	20
Общая информация	20
Команды TDR, TIR (реестр данных прицепа, реестр инструкций прицепа)	21
Синтаксис	21

Цель 21

Параметры 22

Примеры 23

Общая информация 23

Команда TRST (тестовый сброс) 23

 Синтаксис 23

 Цель 23

 Параметры 24

 Примеры 24

Таблица цифр

Рисунок 1: Диаграмма состояний IEEE 1149.1 5

Таблица таблиц

Таблица 1: Названия состояний SVF для состояний TAP IEEE 1149.1 9

Таблица 2: Символы, определяющие направление и состояние PIN-кода тестового вектора 13

Таблица 3: Состояния и описания TRST 21

Таблица примеров

Пример 1: Пример файла SVF 6

Пример 2: Состояния и описания TRST 24

Введение

Файлы последовательного векторного формата (SVF) используются для описания высокоуровневых операций с шиной IEEE 1149.1. В общем, операции шины IEEE 1149.1 состоят из операций сканирования и перемещений между различными стабильными состояниями на диаграмме состояний IEEE 1149.1 (см. Рисунок ниже). SVF не явно описывает состояние шины IEEE 1149.1 на каждом тестовом такте.

SVF разработан для поощрения повторного использования серийных векторов на протяжении всего жизненного цикла продукта, с момента его создания на этапе проектирования до развертывания на этапе эксплуатации в полевых условиях и на всех этапах между ними. Переносимость в течение всего жизненного цикла накладывает ограничения на конструкцию и возможности SVF.

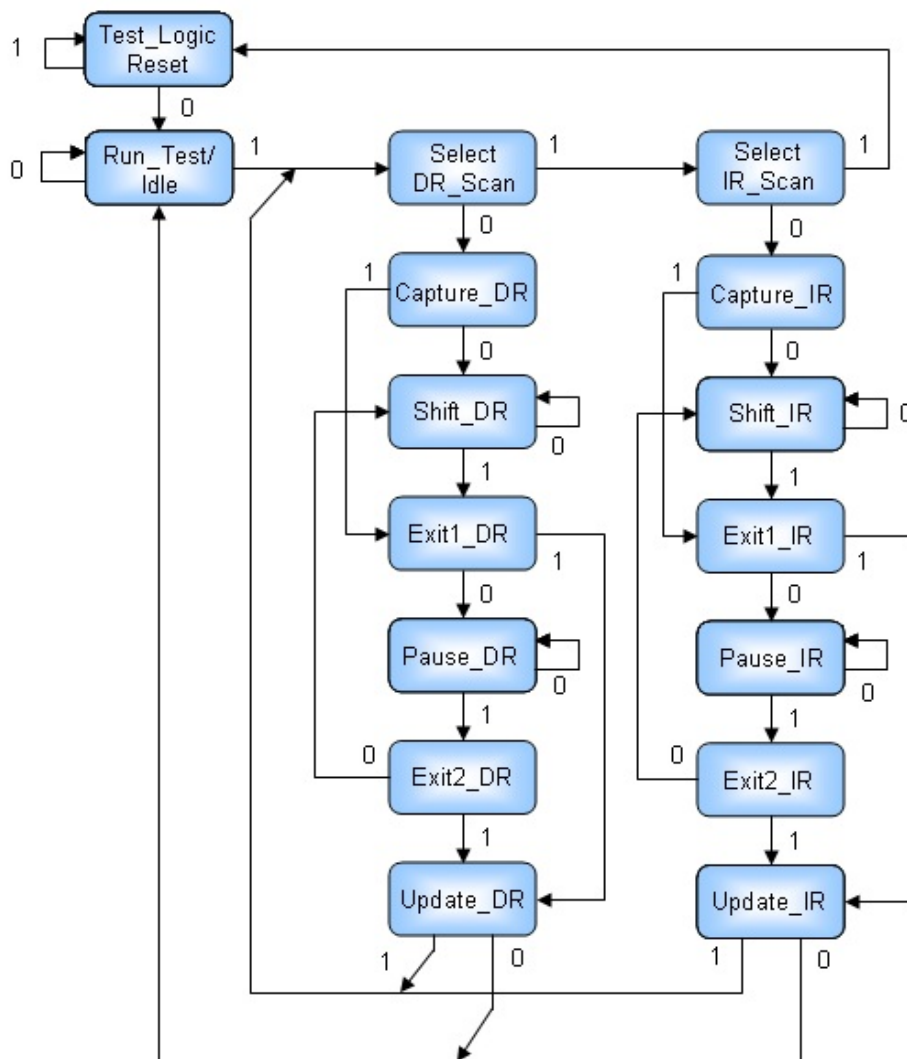


Рисунок 1: Диаграмма состояний IEEE 1149.1

Описание SVF-файла

SVF-файл представляет собой ASCII-файл, состоящий из набора SVF-инструкций. Внутри каждого файла:

- Максимальное количество символов, допустимое в строке, равно 256. Один оператор SVF может занимать более одной строки.
- Каждый оператор состоит из команды и связанных с ней параметров.
- Каждый оператор SVF заканчивается точкой с запятой.
- SVF не чувствителен к регистру.
- Комментарии могут быть вставлены в SVF-файл после восклицательного знака '!' или пары косых черт '//'. Либо '//', либо '!' закомментируют оставшуюся часть строки.
- Данные сканирования в инструкции представлены в шестнадцатеричном формате и всегда заключены в круглые скобки.
 - В данных сканирования не может быть указана строка данных, размер которой превышает указанный бит длина.
 - Нули в шестнадцатеричной строке в старшем значащем бите (MSB) не учитываются при определении того, не слишком ли велика строка.
 - Порядок битов для данных сканирования соответствует соглашению о том, что младший значащий бит (крайний правый бит) является первым битом, отсканированным в аппаратном обеспечении для TDI и SMASK данные сканирования и являются первым битом, отсканированным для данных TDO и MASKI. Этот бит Порядок следования соответствует стандарту IEEE 1149.1.

Пример SVF-файла

Ниже приведен пример SVF-файла. Комментарии начинаются с восклицательного знака (!).

Ссылки в примере приведут вас к подробному описанию этой SVF-команды.

Пример 1: Пример файла SVF

```
!Начать программу тестирования
ВЫКЛЮЧИТЕ;
!Отключить тестовую строку сброса
ЗАВЕРШЕНИЕ РАБОТЫ в режиме ОЖИДАНИЯ;
!Завершить ИК-сканирование в режиме ОЖИДАНИЯ
ЗАВЕРШИТЬ РЕЖИМ ОЖИДАНИЯ DR;
!Завершить сканирование DR в режиме ОЖИДАНИЯ
HIR 8 TDI (00);
!8-разрядный ИК-заголовок
МАСКА HDR 16 TDI (FFFF) TDO (FFFF) (FFFF);
```

```
!16-разрядный заголовок DR
TIR 16 TDI (0000);
!16-разрядный ИК-трейлер
TDR 8 TDI (12);
!16-битный трейлер DR
SIR 8 TDI (41);
!8-битное ИК-сканирование
SDR 32 TDI (ABCD1234) TDO (11112222);
!32-разрядное DR-сканирование
СОСТОЯНИЕ DRPAUSE;
!Перейти в стабильное состояние DRPAUSE
ВРЕМЯ ВЫПОЛНЕНИЯ ТЕСТА 100 TCK С КОНЕЧНЫМ СОСТОЯНИЕМ IRPAUSE;
!RUNBIST за 100 TCKs
!Завершить программу тестирования
```

Краткое описание команд

- **КОНЕЦ**
Задаёт конечное состояние по умолчанию для операций DR-сканирования.
- **ENDIR**
Задаёт конечное состояние по умолчанию для операций ИК-сканирования.
- **ЧАСТОТА**
Задаёт максимальную тестовую тактовую частоту для работы шины IEEE 1149.1.
- **HDR (регистр данных заголовка)**
Задаёт шаблон заголовка, который добавляется к началу последующих операций DR-сканирования.
- **HIR (регистр инструкций заголовка)**
Задаёт шаблон заголовка, который добавляется к началу последующего ИК-сканирования операции.
- **ПНО**
(Параллельный ввод/вывод)
Задаёт шаблон параллельного тестирования.
- **PIOMAP (карта параллельного ввода/вывода)**
Сопоставляет позиции столбцов PIO с логическим выводом.
- **ТЕКУЩИЙ ТЕСТ**
Переводит шину IEEE 1149.1 в рабочее состояние на указанное количество тактов или на указанный период времени.
- **SDR (регистр данных сканирования)**

Выполняет сканирование регистра данных IEEE 1149.1.

- СЭР (регистр команд сканирования)
Выполняет сканирование регистра команд IEEE 1149.1.
- СОСТОЯНИЕ
Переводит шину IEEE 1149.1 в заданное стабильное состояние.
- TDR (Реестр данных прицепа)
Задаёт шаблон завершения, который добавляется в конце последующих операций DR-сканирования.
- TIR (Реестр инструкций для прицепа)
Задаёт шаблон трейлера, который добавляется в конце последующих операций ИК-сканирования.
- TRST (тестовый сброс)
Управляет необязательной линией сброса теста.

Синтаксис

При обсуждении SVF операция сканирования определяется как выполнение команды SIR или SDR и любых связанных с ней команд заголовка или завершения.

Некоторые необязательные параметры команды, такие как MASK, SMASK и TDI, являются "липкими" (они запоминаются из предыдущей команды до тех пор, пока не будут изменены или признаны недействительными), чтобы минимизировать SVF-файл размер. Параметры MASK, SMASK и TDI "запоминаются" отдельно для команд SIR, SDR, NIR, HDR, TIR и TDR.

IEEE 1149.1 Состояния TAP

Некоторые команды SVF ссылаются на состояния касания IEEE 1149.1 (см. Диаграмму состояний касания). В следующей таблице перечислены все имена состояний SVF, используемые для каждого имени состояния TAP IEEE 1149.1.

Таблица 1: Названия состояний SVF для состояний TAP IEEE 1149.1

IEEE 1149.1 Коснитесь названия состояния	SVF Коснитесь названия состояния
Тест-Логика-Сброс	Сброс
Запуск-тест/холостой ход	ХОЛОСТОЙ ХОД
Выбрать-DR-сканирование	DRSELECT
Захват-DR	DRCAPTURE
Сдвиг-DR	DRSHIFT
Exit1-DR	DREXIT1
Пауза-DR	DRPAUSE
Exit2-DR	DREXIT2
Обновление-DR	ДАТА ПУБЛИКАЦИИ
Выбор-ИК-сканирование	IRSELECT
Захват-IR	IRCAPTURE
Сдвиг-IR	IRSHIFT
Exit1-IR	IREXIT1
Пауза-IR	IRPAUSE
Exit2-IR	IREXIT2
Обновление-IR	ОБНОВЛЕНО

Указание действительных чисел

Действительные числа в SVF задаются с использованием синтаксиса:

цифры [. цифры] [Е [+|-] цифры]

Цифры состоят из одной или нескольких десятичных цифр 0-9. Этот синтаксис соответствует десятичным литералам VHDL, исключая подчеркивания. Таким образом, 1, 1E0, 1E+0, 1E-0, 1.0, 1.0E0, 1.0E+0, и 1.0E-0 - все эквивалентные действительные числа; 1., 1.E0, .5 и .5E0 недопустимы. Диапазон и точность действительных чисел определяются реализацией.

Команды ENDDR, ENDIR

Синтаксис

ENDDR stable_state;
КОНЕЧНЫЙ ПАРАМЕТР stable_state;

Назначение

Устанавливает конечное состояние IEEE 1149.1 для операций сканирования.

Параметры

stable_state
В стабильном IEEE 1149.1 указано, что шина IEEE 1149.1 будет принудительно подключена к завершению операции сканирования. Допустимыми стабильными состояниями являются IRPAUSE, DRPAUSE, RESET и IDLE.

Примеры

БЕСКОНЕЧНЫЙ ХОЛОСТОЙ ХОД;

КОНЕЧНАЯ ПАУЗА;

Общая информация

Команды ENDDR и ENDIR определяют стабильное состояние IEEE 1149.1, которое соответствует стандарту IEEE 1149.1 шина будет принудительно отключена по завершении DR- или ИК-сканирования соответственно. После указания Команды ENDDR/ENDIR остаются в силе до тех пор, пока не будут переопределены другой командой ENDDR или ENDIR. При запуске ENDDR и ENDIR оба переводятся в режим ОЖИДАНИЯ.

Частотная команда

Синтаксис

ЧАСТОТА [циклы Гц];

Параметры являются обязательными, если только они не заключены в квадратные скобки '[]'. Параметры, заключенные в квадратные скобки, являются необязательными.

Цель

Устанавливает максимальную тактовую частоту тестирования IEEE 1149.1 (TCK) для последующих сканирований (SDR и SIR), изменений состояния (STATE) и тестовых операций (RUNTEST). Новая частота остается в силе до тех пор, пока не будет выполнен следующий оператор FREQUENCY или не будет достигнут конец файла. Максимальная безопасная частота может быть характеристикой UUT, крепления или любого другого устройства RUNBIST или Конструкции INTEST. Пропуск циклов снимает любые ограничения на максимальную частоту, позволяя тесту вернуться к более высокой тактовой частоте после временного замедления.

Параметры

Циклы

[Необязательно] Максимальная частота TCK в Гц, выраженная как действительное число, большее нуля (0).

Примеры

SIR 8 TDI(F3) TDO(01) МАСКА(03); ! Установить BIST, полная скорость
 ЧАСТОТА 90Е3 Гц; ! Уменьшить до 90 кГц
 ВЫПОЛНИТЬ ТЕСТ 100000 ТСК; ! Выполнить BIST
 ЧАСТОТА 1Е5 Гц; ! Увеличить до 100 кГц
 ВЫПОЛНИТЬ ТЕСТ 300000 ТСК за 1 секунду! Ошибка! 300000 ТСК при 100 кГц - это
 МАКСИМУМ 2 СЕКУНДЫ; ! 3 СЕКУНДЫ, но МАКСИМУМ 2 СЕКУНДЫ
 ЧАСТОТА; ! Вернуться к полной скорости

Общая информация

Значение *cycles* задается как действительное число. Диапазон, количество и точность поддерживаемых частоты определяются реализацией. Реализация, которая не может передать свой TCK на или ниже указанной частоты, должна сообщить об ошибке. Если заданы циклы, новая частота вступает в силу перед следующей командой, поведение которой она изменяет; то есть перед следующим ЗАПУСК, SDR, SIR или команда СОСТОЯНИЯ.

Начальная частота определяется реализацией и, вероятно, задается пользователем извне по отношению к SVF-файлу на основе характеристик привязки к UUT. Если это так, то при пропуске циклов возвращается к этой заданной извне частоте. Используется ли значение *cycles*, когда оно превышает заданную извне частоту, определяется реализацией. Каждый SVF-файл заканчивается подразумеваемой командой FREQUENCY;, чтобы один SVF-файл не влиял на другой SVF-файл.

Реализация должна сообщить об ошибке в команде RUNTEST, если указаны циклы, и команда RUNTEST указывает как значение TCK *run_count*, так и значение *max_time*, которое не может быть выполнено задана максимальная частота.

HDR, HIR (регистр данных заголовка, регистр инструкций заголовка)

Команды

Синтаксис

Длина HDR [TDI (*tdi*)] [TDO (*tdo*)] [МАСКА (*mask*)] [SMASK (*smask*)];

ЕГО длина [TDI (*tdi*)] [TDO (*tdo*)] [МАСКА (*mask*)] [SMASK (*smask*)];

Параметры обязательны, если только они не заключены в квадратные скобки '[']. Параметры, заключенные в квадратные скобки, являются необязательными.

Цель

Задаёт шаблон заголовка по умолчанию, который сдвигается перед каждой операцией сканирования. Эта команда позволяет легко адаптировать набор инструкций сканирования для конкретного компонента IEEE 1149.1 к среде, в которой компонент размещается на пути сканирования, содержащем другие устройства стандарта IEEE 1149.1. Шаблон заголовка определяет, как дополнять инструкции сканирования набором начальных битов, которые соответствуют устройствам, расположенным на пути сканирования за пределами интересующего компонента.

Параметры

длина

32-разрядное десятичное целое число без знака, определяющее количество битов, подлежащих сканированию. При установке значения длины равным 0 заголовков удаляется.

[TDI (*tdi*)]

[Необязательно] TDI (*tdi*) - это значение, которое будет отсканировано в целевой объект, выраженное в виде шестнадцатеричного значения.

Если параметр отсутствует, значение TDI, которое будет отсканировано в целевой объект, будет равно

Значение TDI указано в последней предыдущей инструкции HDR/HIR. Значения HDR и HIR TDI "запоминаются" независимо. Параметр TDI должен быть явно указан для первой команды или при изменении длины.

[TDO (tdo)]

[Необязательно] TDO (*tdo*) - это значение для сравнения с фактическими значениями, отсканированными из целевой объект, выраженный в виде шестнадцатеричной строки. Если этот параметр отсутствует, сравнение не будет выполняться не будет.

[MASKA (mask)]

[Необязательно] Маска, которая будет использоваться при сравнении значений TDO с фактическими значениями отсканированная из целевого объекта, выраженная в виде шестнадцатеричной строки. '1' в определенной позиции бита указывает на заботу об этой позиции. '0' указывает на отсутствие заботы. Если этот параметр отсутствует, используемая маска будет равна последнему ранее указанному значению MASKИ, указанному для инструкции HDR/HIR. Значения масок HDR и HIR "запоминаются" независимо.

Если команда заголовка изменяет длину последнего заголовка того же типа, а Параметр MASK отсутствует, то все заботы сводятся к используемому шаблону маски. Если параметр TDO не указан присутствует, MASKA использоваться не будет.

[SMASK (smask)]

[Необязательно] Указывает, что данные TDI не имеют значения, выражаются в виде шестнадцатеричной строки. '1' в определенной битовой позиции указывает на то, что данные TDI в этой битовой позиции являются тщательными. '0' означает, что мне все равно. Если этот параметр отсутствует, маскировка для всех битов не предполагается. Если этот параметр отсутствует, используемая маска будет равна последней ранее указанной SMASK значение, указанное для инструкции HDR/HIR. Значения HDR и HIR SMASK "запоминаются" независимо. Если команда заголовка изменяет I последнего заголовка того же типа, а параметр SMASK отсутствует, то все заботы сводятся к используемому шаблону smask. SMASK будет использоваться, даже если параметр TDI отсутствует.

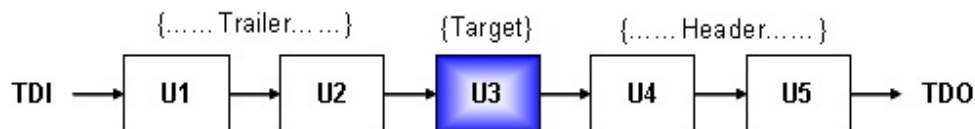
Примеры

```
HDR 32 TDI(00000010) TDO(81818181) MASKA (FFFFFFF) SMASK(0);
HIR 16 TDI (ABCD);
•
•
•
HDR 0; ! Удаляет предыдущий заголовок DR-сканирования.
```

Общая информация

HDR (регистр данных заголовка) задает шаблон заголовка по умолчанию, который будет предшествовать началу всех последующих команд SDR. HIR (регистр команд заголовка) задает шаблон заголовка по умолчанию, который будет предшествовать началу всех последующих команд SIR. Команды заголовка имеют набор параллельных команд прицепа (TIR, TDR), которые описаны позже. Заголовок можно удалить, установив значение length равным 0.

Например, предположим, что SVF-файл разработан для ASIC. Затем ASIC размещается на плате в виде u3, как показано ниже:



Набор инструкций SVF, первоначально разработанных для ASIC, может быть повторно использован с минимальными изменениями, если соответствующие инструкции header и trailer определены для соответствия устройства перед и за u3. В этом примере шаблон заголовка будет определен для устройств u4 и u5, а для u2 и u1 будет определен шаблон прицепа.

Необязательные параметры могут быть указаны в любом порядке. Каждый необязательный параметр может быть указан только один раз. Шестнадцатеричные строки, указанные для TDI, TDO, MASK или SMASK, не могут иметь значение больше максимального значения, подразумеваемого параметром length . Для шестнадцатеричной строки предполагаются начальные нули если не указано явно.

Команда PIO (параллельный ввод/вывод)

Синтаксис

PIO (векторная строка);

Назначение

Задаёт параллельный вектор тестирования. Оператор PIOMAP должен быть определен ранее, и количество векторных символов в операторе PIO должно равняться количеству логических имен в ранее указанном операторе PIOMAP.

Параметры

векторная строка

Ориентированный по столбцам набор из одного или нескольких векторных символов. Каждый символ задает направление и состояние определенного вывода для одного тестового вектора. Определены следующие символы :

Таблица 2: Символы, определяющие направление и состояние вывода тестового вектора

Векторный символ	Значение
H	Логический диск 1
L	Логический 0 на диске
Z	Привод с высоким сопротивлением
U	Определить логическую единицу
D	Определить логический 0
X	Обнаружить Неизвестное

Для двунаправленных каналов буква "Z" означает отсутствие привода / обнаружения.

Пример

```
PIO (HLUDXZHHLL);
```

Команда PIOMAP (Карта параллельного ввода/вывода)

Синтаксис

```
PIOMAP (направление 1 logical_name1...[направление 1 logical_namen]);
```

```
PIOMAP (столбец 1 logical_name1...[столбец n logical_namen]);
```

Параметры обязательны, если только они не заключены в квадратные скобки '[']. Параметры, заключенные в квадратные скобки, являются необязательными.

Цель

Определяет направление ввода-вывода и логическое имя для каждого столбца в операторе PIO. Инструкция PIOMAP Требуется, если SVF-файл использует инструкцию PIO. Если используется PIO, инструкция PIOMAP должна быть помещена в SVF-файл перед любой инструкцией SIR, SDR, STATE, RUNTEST или PIO. Для каждого файла допускается только один оператор PIOMAP.

Первая форма PIOMAP определяет направление ввода-вывода и логическое имя для каждого столбца в порядке появления векторных символов в операторе PIO. Первое logical_name соответствует первый символ vector_string в операторах PIO; второе logical_name соответствует второй символ vector_string; и так далее.

Вторая форма PIOMAP определяет сопоставление между определенным столбцом в инструкции PIO и логическим именем, связанным с этим столбцом. Направление ввода-вывода не определено, поэтому в любом столбце может использоваться любой вектор. Эта вторая форма не рекомендуется для текущего использования поскольку направление ввода-вывода не может быть подтверждено.

Параметры

direction1-n

Задаёт направление ввода-вывода логического имени. Направление может быть IN, OUT или INOUT. Столбец с направлением IN является входом для пользовательского интерфейса, и этот столбец в инструкции PIO может использовать символы "drive". Столбец OUT представляет собой выходные данные из UUT и может использовать символы "detect". Столбец INOUT является двунаправленным сигналом, и может использовать любой векторный символ.

logical_name1-n

Символьная строка, идентифицирующая логическое имя pin-кода, связанного со столбцом. Одна и та же строка не может использоваться более одного раза в инструкции PIOMAP.

столбец 1-n

Задаёт номер столбца инструкции PIO. Первым символом в операторе PIO является столбец 1, вторым символом - столбец 2 и т.д. Номер столбца -

десятичная дробь. Определенный номер столбца нельзя использовать более одного раза. Эта форма PИОМАР не рекомендуется для текущего использования.

Примеры

!Карта PИОМАР должна быть помещена перед инструкцией PИО

PИОМАР (В STROBE

В ALE

ВЫХОД ОТКЛЮЧИТЬ

ВЫХОД ВКЛЮЧИТЬ

ВНЕ ПОЛЯ ЗРЕНИЯ

В КОМПЛЕКТЕ);

PИО (HLUDXZ);

!Вектор равен:

! СТРОБОСКОП <- Н

! АЛЕ <- L

! ОТКЛЮЧИТЬ <- U

! ВКЛЮЧИТЬ <- D

! ОЧИСТИТЬ <- X

! УСТАНОВИТЬ <- Z

Команда RUNTEST

Синтаксис

```
RUNTEST [run_state] run_count run_clk [минимальное время В СЕКУНДЕ [МАКСИМАЛЬНОЕ max_time В СЕКУНДЕ]]
[КОНЕЧНОЕ СОСТОЯНИЕ end_state];
```

```
ВЫПОЛНЕНИЕ ТЕСТА [run_state] min_time SEC [МАКСИМАЛЬНОЕ время выполнения max_time SEC] [КОНЕЧНОЕ СОСТОЯНИЕ end_state];
```

Параметры обязательны, если только они не заключены в квадратные скобки '[']. Параметры, заключенные в квадратные скобки, являются необязательными.

Цель

Переводит целевую шину IEEE 1149.1 в указанное рабочее состояние на указанное количество тактов (либо тестовые такты, либо системные такты), на указанный промежуток времени или на то и другое вместе, затем перемещает целевую шину переведите шину в указанное конечное состояние. Это может быть использовано для управления операциями RUNBIST в целевом объекте.

Первая форма команды RUNTEST выполняет тест в указанном состоянии выполнения в течение указанного количества тактов. При необходимости можно

. Вторая форма команды RUNTEST выполняет тест в течение указанного

минимального и максимального количества секунд. Поскольку RUNTEST - это команда, которая генерирует векторы, ТСК синхронизируется, даже если run_count не указан.

Необходимо указать либо `run_count`, либо `min_time`. Если указаны как `run_count`, так и `min_time`, перед завершением выполнения команды `RUNTEST` должны быть выполнены оба условия. Если значение `max_time` превышено, `RUNTEST` останавливается до достижения значения `run_count`.

И `min_time`, и `max_time` задаются как действительные числа.

Разрешение и диапазон временной задержки определяются реализацией.

Транслятор должен выдать предупреждение, если он не может гарантировать максимальное время, когда текущее состояние перед выполнением команды `RUNTEST` совпадает с `run_state` и `max_time` указан. Переводчик также должен выдать предупреждение о переносимости, что максимальное время может быть нарушено в некоторых реализациях при данных условиях. Реализация, использующая свободный запущенный TCK мог бы находиться в состоянии выполнения и работать с тактовой частотой дольше, чем `max_time`, поскольку у него было бы ввел значение `run_state` в конце предыдущей команды, а не в начале `RUNTEST` команда.

Параметры

`run_state`

[Необязательно] В стабильном IEEE 1149.1 указано, что шина IEEE 1149.1 будет принудительно подключена к во время выполнения команды `RUNTEST` (см. Диаграмму состояния TAP). Допустимыми состояниями запуска являются **IRPAUSE**, **DRPAUSE**, **СБРОС** и **ХОЛОСТОЙ ХОД**. Если тестовая шина уже находится в рабочем состоянии, никаких переходов в состояние не происходит. Как только указано `run_state`, последующие команды `RUNTEST` по умолчанию будут иметь то же состояние выполнения, если оно не указано. Начальное значение по умолчанию - **IDLE**. Для получения информации о пути по умолчанию, используемом при переходе из одного состояния IEEE 1149.1 в другое, обратитесь к таблице, включенной в описание СОСТОЯНИЯ.

`run_count`

[Необязательно] Количество тактов, в течение которых шина IEEE 1149.1 будет оставаться в рабочем состоянии, выражается в виде 32-разрядного десятичного числа без знака, большего 0.

`run_clk`

[Необязательно] Указывает используемые часы, либо TCK (тестовые часы), либо SCK (системные часы). Системные часы относятся к часам в пользовательском интерфейсе, которые являются асинхронными по отношению к TCK. Системные часы зависят от реализации.

[`min_time SEC`]

[Необязательно] Минимальное время в секундах, которое должна выполнить команда `RUNTEST` перед завершением.

[МАКСИМАЛЬНОЕ время в СЕКУНДЕ `max_time`]

[Необязательно] Максимальное время в секундах, в течение которого команда `RUNTEST` может выполняться до того, как она должна завершиться. Максимальное время должно быть больше минимального время. Если указаны как `run_count`, так и `max_time`, и не вся синхронизация завершена по достижении `max_time` команда завершается, даже если не все запрошенные синхронизация выполнена. Можно ли гарантировать максимальное время или нет, зависит от определяется реализация.

[КОНЕЧНОЕ СОСТОЯНИЕ end_state]

[Необязательно] В стабильном IEEE 1149.1 указано, что шина IEEE 1149.1 будет принудительно отключена после выполнения указанного количества тактов, ожидания указанного промежутка времени или того и другого.

Допустимыми конечными состояниями являются **IRPAUSE**, **DRPAUSE**, **RESET** и **IDLE**. Если тестовая шина уже находится в конечном состоянии, переходов между состояниями не происходит. Если значение end_state не указано, используется конечное состояние по умолчанию. Когда указано значение end_state, оно становится значением по умолчанию. Когда указывается run_state, новое run_state становится конечным состоянием по умолчанию. Если параметр run_state не указан, значение end_state по умолчанию остается в силе. Начальное значение по умолчанию для end_state - **IDLE**.

Примеры

! Запустите в режиме Run-Test / Idle в течение 1000 TCKs, затем перейдите в режим Pause-DR.

ВЫПОЛНЕНИЕ ТЕСТА с КОНЕЧНЫМ СОСТОЯНИЕМ DRPAUSE 1000 TCK;

! Вернитесь в режим Run-Test / Idle на 20 секунд, затем перейдите в режим Pause-DR.

ВРЕМЯ ВЫПОЛНЕНИЯ 20 SCK;

! Запускайте в режиме Run-Test / Idle в течение 1000000 повторных подключений или не менее одной секунды,

! затем перейдите в режим Pause-DR.

ВЫПОЛНЕНИЕ ТЕСТА 1000000 TCK за 1 СЕКУНДУ;

! Запускайте в режиме Run-Test / Idle не менее одной миллисекунды и не более

! 50 миллисекунд, затем оставайтесь в режиме Run-Test /Idle.

ВРЕМЯ ВЫПОЛНЕНИЯ ТЕСТА 10,0Е-3 СЕКУНДЫ, МАКСИМАЛЬНОЕ ВРЕМЯ ОЖИДАНИЯ 50,0Е-3 СЕКУНДЫ В КОНЕЧНОМ РЕЖИМЕ;

! Запустите в режиме Pause-DR не менее 50 мс, затем перейдите в режим Run-Test /Idle.

ПАУЗА ЗАПУСКА DRPAUSE 50Е-3 СЕКУНДЫ ОЖИДАНИЯ В КОНЕЧНОМ СОСТОЯНИИ;

! Запустите в режиме Pause-DR не менее одной секунды, затем перейдите в режим Run-Test/Idle.

ВЫПОЛНЕНИЕ ТЕСТА 1 СЕК.;

! Запустите в режиме Run-Test / Idle не менее 10 мс, затем оставайтесь в

! Run-Test / Idle.

ВРЕМЯ ВЫПОЛНЕНИЯ ТЕСТА В РЕЖИМЕ ОЖИДАНИЯ 1Е-2 СЕКУНДЫ;

Команды SDR, SIR (регистр данных сканирования, регистр инструкций сканирования)

Синтаксис

Длина SDR [TDI (*tdi*)] [TDO (*tdo*)] [МАСКА (*mask*)] [SMASK (*smask*)];

БОЛЬШАЯ длина [TDI (*tdi*)] [TDO (*tdo*)] [МАСКА (*mask*)] [SMASK (*smask*)];

Параметры обязательны, если только они не заключены в квадратные скобки '[']. Параметры, заключенные в квадратные скобки, являются необязательными.

Цель

Задает шаблон сканирования, который будет применен к целевому регистру (регистрам) сканирования.

Параметры

длина

32-разрядное десятичное целое число без знака, большее 0, определяющее количество битов, которые необходимо сканировать.

[TDI (*тди*)]

[Необязательно] Значение, которое будет отсканировано в целевой объект, выраженное в виде шестнадцатеричного значения. Если этот параметр отсутствует, значение TDI, которое будет сканироваться в целевой системе, будет равно значению TDI , указанному в последней предыдущей инструкции SDR/SIR. Значения SDR и SIR TDI "запоминаются" независимо. Параметр TDI должен быть явно указан для первой команды или при изменении длины.

[TDO (*тдо*)]

[Необязательно] Значения, подлежащие сравнению с фактическими значениями, отсканированными из целевого объекта, выражаются в виде шестнадцатеричной строки. Если этот параметр отсутствует, сравнение не будет проводиться . Если параметр TDO отсутствует, МАСКА использоваться не будет.

[МАСКА (*mask*)]

[Необязательно] Маска, которая будет использоваться при сравнении значений TDO с фактическими значениями отсканированная из целевого объекта, выраженная в виде шестнадцатеричной строки. '1' в определенной позиции бита указывает на заботу об этой позиции. '0' указывает на отсутствие заботы. Если этот параметр отсутствует , используемая маска будет равна последнему ранее указанному значению МАСКИ, указанному для инструкции SDR/SIR. Значения масок SDR и SIR "запоминаются" независимо. Если команда сканирования изменяет продолжительность последнего сканирования того же типа и МАСКУ параметр отсутствует, все заботы связаны с используемым шаблоном маски. Если параметр TDO отсутствует, МАСКА использоваться не будет.

[СМАСК (*smask*)]

[Необязательно] Указывает, что данные TDI не имеют значения, выражаются в виде шестнадцатеричной строки. '1' в определенной битовой позиции указывает на то, что данные TDI в этой битовой позиции являются тщательными. '0' означает, что мне все равно. Если этот параметр отсутствует, используемая маска будет равна последнему ранее указанному значению SMASK, указанному для инструкции SDR/SIR. SDR и СЭР СМАСК

значения "запоминаются" независимо. Если команда сканирования изменяет продолжительность последнего сканирования того же типа, а параметр SMASK отсутствует, все, что нужно, - это используемый шаблон smask . SMASK будет использоваться, даже если параметр TDI отсутствует.

Примеры

```
SDR 24 TDI(000010) TDO(818181) MASKA(FFFFFF) SMASK(0);
```

```
SIR 16 TDI (ABCD);
```

Общая информация

Регистр данных сканирования (SDR) определяет шаблон сканирования, который будет отсканирован в целевой регистр данных. Регистр команд сканирования (SIR) определяет шаблон сканирования, который должен быть отсканирован до целевого Регистра команд.

Перед сканированием значений, указанных в команде SDR или SIR, последний определенный заголовок команда (HDR или HIR соответственно) будет предшествовать началу SDR или SIR шаблон данных и последняя определенная команда прицепа (TDR или TIR соответственно) будут добавлены к в конце шаблона данных SDR или SIR.

Необязательные параметры могут быть указаны в любом порядке. Каждый необязательный параметр можно задать только один раз. Шестнадцатеричные строки, указанные для TDI, TDO, MASK или SMASK, не могут иметь значение больше максимального значения, подразумеваемого параметром length . Для шестнадцатеричной строки предполагаются начальные нули если не указано явно.

КОМАНДА СОСТОЯНИЯ

Синтаксис

```
СОСТОЯНИЕ [pathstate1 [pathstate2 ...[pathstaten]]] stable_state ;
```

Параметры обязательны, если не заключены в квадратные скобки '[' ']' . Параметры, заключенные в квадратные скобки необязательны.

Назначение

Переводит целевую шину IEEE 1149.1 в стабильное состояние IEEE 1149.1.

Параметры

состояние пути_{1-n}

Необязательный список одного или нескольких состояний IEEE 1149.1, описывающий явный путь через диаграмма состояний порта тестового доступа (TAP), которая используется для достижения окончательного стабильного состояния. Допустимыми состояниями являются **RESET**, **IDLE**, **DRSELECT**, **DRCAPTURE**, **DRSHIFT**,

DRPAUSE, DREXIT1, DREXIT2, DRUPDATE, IRSELECT, IRCAPTURE, IRSHIFT, IRPAUSE, IREXIT1, IREXIT2 и IRUPDATE. Состояния должны быть перечислены в порядке, соответствующем диаграмме состояний ПОДКЛЮЧЕНИЯ. Если параметры *pathstate* отсутствуют в списке, предполагается путь по умолчанию на основе текущего состояния и конечного стабильного состояния, как указано в таблице 3.

stable_state

Стабильный IEEE 1149.1 утверждает, что шина IEEE 1149.1 будет вынуждена. Допустимые стабильные состояния - это **IRPAUSE, DRPAUSE, RESET** и **IDLE**.

Примеры

!Принудительно перевести шину в режим DRPAUSE из текущего состояния, когда она находится в

состоянии DRPAUSE;

!Продуктите четкий путь, по которому автобус будет двигаться, начиная с

! ПЕРЕХОД от DRPAUSE к IRPAUSE

СОСТОЯНИЕ DREXIT2 DRUPDATE DRSELECT IRSELECT IRCAPTURE IREXIT1 IRPAUSE;

Общая информация

Команда STATE используется для перевода шины IEEE 1149.1 из одного стабильного состояния в другое для Инициализации, завершения теста UUT или тестирования контроллера TAP. Значения TDI и TDO не определены при использовании команды STATE.

Если параметры *pathstate* не указаны, команда STATE всегда следует по одному и тому же пути через диаграмму состояний IEEE 1149.1 при переходе из одного стабильного состояния в другое. В следующей таблице указан путь, пройденный для каждого стабильного состояния. Предполагается, что для каждого пути состояния что будет выполнен хотя бы один тестовый такт.

Таблица 3: Состояния и описания TRST

Текущее состояние	Новое состояние	Путь состояния
Сброс	Сброс	Сброс
Сброс	БЕЗДЕЙСТВУЮЩИЙ	СБРОС-ХОЛОСТОЙ ХОД
Сброс	ПАУЗА	СБРОС-РЕЖИМ ОЖИДАНИЯ-DRSELECT-DRCAPTURE-DREXIT1-DRPAUSE
Сброс	ПАУЗА	СБРОС-РЕЖИМ ОЖИДАНИЯ-IRSELECT-IRCAPTURE-IREXIT1-IRPAUSE
БЕЗДЕЙСТВУЮЩИЙ	Сброс	ХОЛОСТОЙ ХОД-DRSELECT-IRSELECT-СБРОС
ХОЛОСТОЙ ХОД	ПРОСТОЯ	ПРОСТОЯ
ПРОСТОЯ	ПАУЗА	РЕЖИМ ОЖИДАНИЯ-DRSELECT-DRCAPTURE-DREXIT1-DRPAUSE
ПРОСТОЯ	ПАУЗА	ХОЛОСТОЙ ХОД-DRSELECT-IRSELECT-IRCAPTURE-IREXIT1-IRPAUSE
ПАУЗА	Сброс	DRPAUSE-DREXIT2-DRUPDATE-DRSELECT-IRSELECT-СБРОС
ПАУЗА	БЕЗДЕЙСТВИЕ	DRPAUSE-DREXIT2-DRUPDATE-БЕЗДЕЙСТВИЕ
ПАУЗА	ПАУЗА	DRPAUSE-DREXIT2-DRUPDATE-DRSELECT-DRCAPTURE-DREXIT1-DRPAUSE
ДРПАУЗА	ИРПАУЗА	DRPAUSE-DREXIT2-DRUPDATE-DRSELECT-IRSELECT-IRCAPTURE-IREXIT1-IRPAUSE
ПАУЗА	Сброс	IRPAUSE-IREXIT2-IRUPDATE-DRSELECT-IRSELECT-СБРОС
ПАУЗА	БЕЗДЕЙСТВИЕ	IRPAUSE-IREXIT2-IRUPDATE-БЕЗДЕЙСТВИЕ
ПАУЗА	ПАУЗА	IRPAUSE-IREXIT2-IRUPDATE-DRSELECT-DRCAPTURE-DREXIT1-DRPAUSE
ПАУЗА	ПАУЗА	IRPAUSE-IREXIT2-IRUPDATE-DRSELECT-IRSELECT-IRCAPTURE-IREXIT1-IRPAUSE

Команды TDR, TIR (Реестр данных прицепа, Реестр инструкций прицепа)

Синтаксис

Длина TDR [TDI (*tdi*)] [TDO (*tdo*)] [МАСКА (*mask*)] [SMASK (*smask*)];

Длина TIR [TDI (*tdi*)] [TDO (*tdo*)] [МАСКА (*mask*)] [SMASK (*smask*)];

Параметры обязательны, если только они не заключены в квадратные скобки '[]'. Параметры, заключенные в квадратные скобки, являются необязательными.

Цель

Задаёт шаблон трейлера по умолчанию, который сдвигается после всех последующих операций сканирования. Эта команда предназначена для того, чтобы позволить набору инструкций сканирования для конкретного компонента IEEE 1149.1 легко адаптироваться к среде, в которой компонент размещен на пути сканирования, содержащем

другие устройства стандарта IEEE 1149.1. Шаблон завершения определяет, как дополнять инструкции сканирования набором завершающих битов, которые соответствуют устройствам, расположенным на пути сканирования после интересующего

Параметры

длина

32-разрядное десятичное целое число без знака, определяющее количество битов, подлежащих сканированию. При установке значения длины 0 трейлер удаляется.

[TDI (тди)]

[Необязательно] Значение, которое будет отсканировано в целевой объект, выраженное в виде шестнадцатеричного значения. Если этот параметр отсутствует, значение TDI, которое будет отсканировано в целевом объекте, будет равно значению TDI значение, указанное в последней предыдущей выписке TDR/TIR. Значения TDR и TIR TDI "запоминаются" независимо. Параметр TDI должен быть явно указан для первой команды или при изменении длины.

[TDO (тдо)]

[Необязательно] Значения, подлежащие сравнению с фактическими значениями, отсканированными из целевого объекта, выражаются в виде шестнадцатеричной строки. Если этот параметр отсутствует, сравнение не будет проводиться

[MASK (mask)]

[Необязательно] Маска, которая будет использоваться при сравнении значений TDO с фактическими значениями отсканированными из целевого объекта, выраженными в виде шестнадцатеричной строки. '1' в определенной позиции бита указывает на заботу об этой позиции. '0' указывает на отсутствие заботы. Если этот параметр отсутствует, используемая маска будет равна последнему ранее указанному значению MASKI, указанному для в заявлении TDR/TIR. Значения маски TDR и TIR "запоминаются" независимо. Если команда trailer изменяет длину последнего трейлера того же типа, а параметр MASK отсутствует, то все заботы сводятся к используемому шаблону маски. Если параметр TDO отсутствует, MASK использоваться не будет.

[SMASK (smask)]

[Необязательно] Указывает, какие данные TDI являются безразличными, выраженными в виде шестнадцатеричной строки. '1' в определенной битовой позиции указывает на то, что данные TDI в этой битовой позиции являются тщательными. '0' означает, что мне все равно. Если этот параметр отсутствует, используемая маска будет равна последнему ранее указанному значению SMASK, указанному в заявлении TDR/TIR. TDR и SMASK МДП значения "запоминаются" независимо. Если команда trailer изменяет длину последнего трейлера того же типа, а параметр SMASK отсутствует, используемый шаблон smask не имеет значения. SMASK будет использоваться, даже если параметр TDI отсутствует.

Примеры

```
TDR 32 TDI(00000010) TDO(81818181) MASKA (FFFFFFF) SMASK(0);
```

```
TIR 16 TDI(ABCD);
```

```

•
•
•

```

```
TDR 0; ! Удаляет предыдущий трейлер DR-сканирования.
```

Общая информация

TDR (регистр данных прицепа) определяет шаблон прицепа, который будет добавлен в конце всех последующих команд SDR. TIR (Реестр инструкций прицепа) задает шаблон прицепа по умолчанию, который будет добавлен в конце всех последующих команд SIR. Прицеп можно снять, установив значение длины 0.

Например, предположим, что SVF-файл разработан для ASIC. Затем ASIC размещается на плате в виде u3, как показано ниже:



Набор инструкций SVF, первоначально разработанных для ASIC, может быть повторно использован с минимальными изменениями, если соответствующие инструкции header и trailer определены для соответствия устройства перед и за u3. В этом примере шаблон заголовка будет определен для устройств u4 и u5, а для u1 и u2 будет определен шаблон прицепа.

Необязательные параметры могут быть указаны в любом порядке. Каждый необязательный параметр может быть указан только один раз. Шестнадцатеричные строки, указанные для TDI, TDO, MASK или SMASK, не могут иметь значение больше максимального значения, подразумеваемого параметром length . Для шестнадцатеричной строки предполагаются начальные нули если не указано явно.

Команда TRST (тестовый сброс)

Синтаксис

```
TRST trst_mode;
```

Назначение

Команда TRST описывает работу с дополнительным сигналом тестового сброса.

Параметры

trst_mode

Указывает, является ли линия TRST активной, неактивной, имеет высокое сопротивление или вообще существует.
Допустимыми состояниями trst_mode являются:

Пример 2: Текущие состояния и описания

Состояние	Описание
ВКЛЮЧЕНО	Активен (логический 0)
ВЫКЛ	Неактивен (логика 1)
Z	Высокий импеданс
ОТСУТСТВУЕТ	Отсутствует

Если указан TRST с параметром ABSENT , он должен быть помещен в начало SVF
файл подается перед любыми операторами SIR, SDR, STATE, RUNTEST или PIO, и никакие дальнейшие инструкции TRST не допускаются
. Оператор TRST не зависит от состояния TAP IEEE 1149.1.
Это позволяет тестировать цель, используя либо линию TRST, либо состояние IEEE 1149.1
TAP.

Примеры

ПРОДОЛЖАЮ;
ПРОДОЛЖАЙТЕ;