

## **Средства временного анализа**

Временной анализатор TimeQuest поддерживает некоторые средства, которые расширяют и обеспечивают сквозной анализ вашего проекта. В этой главе описаны некоторые средства, доступные во временном анализаторе TimeQuest.

### **Многоугольный анализ**

Многоугольный анализ позволяет верифицировать проект с различными оперативными состояниями (напряжение, обработка и температура), когда выполняется статический временной анализ проекта.

Используйте команду `set_operating_conditions` для изменения оперативных состояний или градации скорости чипа, используемого в статическом временном анализе.

В примере 7-56 показана команда `set_operating_conditions` и опции.

### Example 7-56. set\_operating\_conditions Command

```
set_operating_conditions
[-model <fast|slow>]
[-speed <speed grade>]
[-temperature <value in °C>]
[-voltage <value in mV>]
[<operating condition Tcl object>]
```

В таблице 7-50 перечислены опции команды set\_operating\_conditions

**Таблица 7-50. Опции команды set\_operating\_conditions**

| Опция                            | Описание  |
|----------------------------------|---|
| -model <fast slow>               | Определяет временную модель   |
| -speed <speed grade>             | Определяет градации скорости чипа   |
| -temperature <value in °C>       | Определяет рабочую температуру  |
| -voltage <value in mV>           | Определяет рабочее напряжение   |
| <operating condition Tcl object> | Определяет оперативные состояния Tcl объекта, которые определяют оперативные состояния. |

Если используются оперативные состояния Tcl объекта, то опции: модель, скорость, температура и напряжение, - не требуются. Если не используются оперативные состояния Tcl объекта, то определяется модель, а опции -speed, -temperature и -voltage – не обязательны, используйте соответствующие умолчания для применяемого чипа.

В таблице 7-51 показаны некоторые доступные оперативные состояния, которые могут быть установлены для разных семейств чипов.

**Таблица 7-51. Оперативные состояния семейств чипов**

| Семейство чипов | Доступные состояния |          |                 |                  | Оперативные состояния Tcl объекта |
|-----------------|---------------------|----------|-----------------|------------------|-----------------------------------|
|                 | Градация скорости   | Модель   | Напряжение (мВ) | Температура (°C) |                                   |
| Stratix III     | 4                   | Медленно | 1100            | 85               | 4_slow_1100mv_85c                 |
|                 |                     | Медленно | 1100            | 0                | 4_slow_1100mv_0c                  |
|                 |                     | Быстро   | 1100            | 0                | MIN_fast_1100mv_0c                |
| Cyclone III     | 7                   | Медленно | 1200            | 85               | 7_slow_1200mv_85c                 |
|                 |                     | Медленно | 1200            | 0                | 7_slow_1200mv_0c                  |
|                 |                     | Быстро   | 1200            | 0                | MIN_fast_1200mv_0c                |
| Stratix II      | 4                   | Медленно | -               | -                | 4_slow                            |
|                 |                     | Быстро   |                 |                  | MIN_fast                          |
| Cyclone II      | 6                   | Медленно | -               | -                | 6_slow                            |
|                 |                     | Быстро   |                 |                  | MIN_fast                          |

Используйте команду get\_available\_operating\_conditions-all для получения списка возможных оперативных состояний для выбранного чипа.

В таблице 7-52 показаны оперативные состояния для каждой модели семейств чипов, которые поддерживают только два оперативных состояния, т.е. медленно и быстро.

**Таблица 7-52. Оперативные состояния для медленных и быстрых моделей**

| Модель   | Градация скорости                    | Напряжение                  | Температура     |
|----------|--------------------------------------|-----------------------------|-----------------|
| Медленно | Наименьшая скорость в плотности чипа | Vcc минимальная подача (1)  | Максимум TJ (1) |
| Быстро   | Наибольшая скорость в плотности чипа | Vcc максимальная подача (1) | Минимум TJ (1)  |

Замечание к таблице 7-52: (1) Обратитесь к главе "DC и характеристики переключений" в настольной книге применяемого чипа за информацией о Vcc и TJ.

Статический временной анализ может быть выполнен для всех возможных оперативных состояний. Это гарантирует, что не будет происходить нарушений в работе чипа при различных оперативных состояниях.

В примере 7-57 показано, как установить оперативные состояния для проекта Stratix III – медленная модель, 1100 мВ и 85 °C.

#### **Example 7–57. Setting Operating Conditions with Individual Values**

```
set_operating_conditions -model slow -temperature 85 -voltage 1100
```

Иначе, вы сможете установить оперативные состояния для примера 7-57 с помощью Tcl объекта, как показано на рисунке 7-58.

#### **Example 7–58. Setting Operating Conditions with a Tcl Object**

```
set_operating_conditions 4_slow_1100mv_85c
```

## **Расширенные временные назначения I/O и модели трассировки платы**

Временной анализатор Quartus II TimeQuest может использоваться для расширенных временных назначений I/O и модели трассировки платы для моделей задержек I/O буферов в вашем проекте.

Чтобы включить или выключить средство расширения I/O, во вкладке **Настройки**, под опцией временного анализатора TimeQuest, выберите включить или выключить.

Если вы включили или выключили расширенные временные назначения I/O или назначения модели трассировки, и не рекомпилировали перед временным анализом, вам необходимо использовать команду -force\_dat, чтобы создать временной список соединений. Наберите эту команду в Tcl консоли временного анализатора Quartus II TimeQuest:

```
create_timing_netlist -force_dat ↵
```

Если вы включили или выключили расширенные временные назначения I/O или назначения модели трассировки, и рекомпилировали перед временным анализом, вам не нужно использовать команду -force\_dat, чтобы создать временной список соединений. Вам необходимо использовать команду create\_timing\_netlist для создания списка соединений, или использовать панель **Создать Временной список соединений** в панели задач.

За дополнительной информацией о средствах расширенного временного I/O, обратитесь к главе "Управление I/O" в томе 2 Настольной книги Quartus II.

## **Назначение "дикой карты" и коллекции**

Для упрощения применения ограничений для нескольких узлов проекта, временной анализатор Quartus II TimeQuest использует символы "дикой карты" "\*" и "?". Используйте эти символы, чтобы уменьшить численность индивидуальных ограничений, которые должны быть определены в проекте.

Символ "\*" создаёт строку. Например, вы создаете назначения для узла, названного reg\*, временной анализатор Quartus II TimeQuest ищет и применяет назначения ко всем узлам проекта, которые имеют префикс reg с отсутствием символа после, с одиночным символом или несколькими символами, например, reg1, reg[2], regbank и reg12bank.

Символ "?" создаёт только один символ. Например, вы создаете назначения для узла, названного reg?, временной анализатор Quartus II TimeQuest ищет и применяет назначения ко всем узлам проекта, которые имеют префикс reg с любым одиночным символом после, например, reg1, rega и reg4.

Коллекция из двух команд `get_cells` и `get_pins`, которые имеют три опции, позволит вам улучшить поиск с символами "дикой карты". Для улучшения поиска, выберите свойство по умолчанию, опцию `-hierarchical` или `-compatibility`.

Символ черта используется для отделения одного иерархического уровня от другого во временном анализаторе Quartus II TimeQuest. Например, `<полное имя ячейки>|<суффикс вывода>` показывает иерархическое имя вывода, отделяется имя вывода символом `|` от иерархии.

Когда вы используете коллекцию команд `get_cells` и `get_pins` без опций, поиск по умолчанию будет выполнен на том же иерархическом уровне, что и имя вывода; это значит, поиск будет выполняться уровень за уровнем. Полное иерархическое имя может содержать несколько иерархических уровней, для которых используется символ `|`, чтобы разделить уровни, а каждый символ "дикой карты" захватывает только один иерархический уровень. Например, символ `"*"` описывает первый иерархический уровень, а `"*|*"` описывает первый и второй иерархические уровни.

Когда вы используете коллекцию команд `get_cells` и `get_pins` с опцией `-hierarchical`, будет выполнен возврат на родственный иерархический путь вида `<краткое имя ячейки>|<имя вывода>`. Поиск будет выполнен по имени узла; например, последняя версия иерархии имён, но не иерархических путей. В отличие от состояния по умолчанию, эта опция не ограничивает поиск на каждом иерархическом уровне, написанном после символа черты.

Символ черта не может быть использован в поиске с опцией `get_cells -hierarchical`. Символ черта используется только в поиске коллекции `get_pins`.

Когда вы используете коллекцию команд `get_cells` и `get_pins` с опцией `-compatibility`, будет выполнен поиск, подобный тому, что есть в Классическом временном анализаторе Quartus II. С помощью этой опции отыскиваются все иерархические пути, а символ черта не является специальным символом.

Если в проекте есть следующие ячейки:

```
foo
foo|bar
```

и следующие имена выводов:

```
foo|dataa
foo|datab
foo|bar|datac
foo|bar|datad
```

В таблице 7-53 приведены результаты поиска с использованием строк.

**Таблица 7-53. Примеры строк поиска и результаты (часть 1 из 2)**

| Строка поиска                                   | Строка результата                               |
|---|---|
| <code>get_pins * dataa</code>                   | <code>foo dataa</code>                          |
| <code>get_pins * datac</code>                   | пусто   |
| <code>get_pins * * datac</code>                 | <code>foo bar datac</code>                      |
| <code>get_pins foo* *</code>                    | <code>foo dataa</code> , <code>foo datab</code> |
| <code>get_pins -hierarchical * * datac</code>   | пусто (1)                                       |
| <code>get_pins -hierarchical foo *</code>       | <code>foo dataa</code> , <code>foo datab</code> |
| <code>get_pins -hierarchical * datac</code>     | <code>foo bar datac</code>                      |
| <code>get_pins -hierarchical foo * datac</code> | пусто (1)                                       |

**Таблица 7-53. Примеры строк поиска и результаты (часть 2 из 2)**

| Строка поиска                     | Строка результата |
|-----------------------------------|-------------------|
| get_pins -compatibility * datac   | foo bar datac     |
| get_pins -compatibility * * datac | foo bar datac     |

Замечания к таблице 7-53: (1) Результат добавления "\*"|" в строку поиска <пусто>.

## **Сброс проекта**

Используйте команду `reset_design`, чтобы удалить все временные ограничения и исключения из проекта во время анализа. Эта команда удаляет все такты, сгенерированные такты, выведенные такты, входные задержки, выходные задержки, тактовую задержку, неопределённость тактов, тактовые группы, ложные пути, мультицикловые пути, минимум задержки и максимум задержки.

Этой командой возможно установить начальное состояние временного анализа без необходимости удаления и воссоздания нового временного списка соединений.

## **Межпробник**

Средство межпробника позволяет вам локализовать пути и элементы из временного анализатора TimeQuest в других доступных средствах разработки программы Quartus II (и наоборот).

В TimeQuest GUI вы можете правым кликом на любом пути в панели **Вид** выбрать **Локализовать Путь** или **Локализовать**.

Исходным является элемент в столбце **От Узла**, а элементом назначения – в столбце **К Узлу**. Опция **Локализовать Путь** позволяет вам локализовать путь поступления данных, по умолчанию для текущей выбранной строки. Чтобы локализовать требуемое время поступления данных, выберите строку на панели требуемого пути данных.

Команда **Локализовать Требуемый Путь** доступна, если требуется показать этот путь; если пользователь хочет вывести путь, который вероятно охватывает только один узел, то эта команда не доступна.

Опция **Локализовать** позволяет вам локализовать выделенный элемент.

Команды **Локализовать путь** и **Локализовать** являются межпробниками для Планировщика Чипа, Просмотрщика Технологической Карты или Редактора Характеристики Ресурсов. В дополнении, опция **Локализовать Путь** является межпробником для Настроек Критических Путьей.

Из диалогового окна Настроек Критических Путьей в Планировщике Чипа, вы можете сделать межпробу для Временного Анализатора TimeQuest отчёта о критических путях проекта.

## **locate**

Используйте команду `locate` в панели Консоли для межпробы Редактора Чипа, Настроек Критических Путьей, Просмотрщика Технологической Карты или Редактора Характеристики Ресурсов.

В примере 7-59 показана команда locate и опции

**Example 7-59. locate Command**

```
locate
[-chip]
[-color <black|blue|brown|green|grey|light_grey|orange|purple|red|white>]
[-cps]
[-label <label>]
[-rpe]
[-tmv]
<items>
```

В таблице 7-54 описаны опции команды locate

**Таблица 7-54. Опции команды locate**

| Опция  | Описание  |
|--|---|
| -chip  | Локализация объекта в Планировщике Чипа   |
| -color<br><black blue brown green grey light_grey orange purple red white> | Цветовая идентификация локализованных объектов  |
| -cps   | Локализация объекта в диалоговом окне Настроек Критических путей Планировщика Чипа  |
| -label <label>   | Определяет метку, используемую для идентификации локализованных объектов  |
| -rpe   | Локализация в Редакторе Характеристик Ресурсов  |
| -tmv   | Локализация в Просмотрщике Технологической Карты  |
| <items>  | Элементы локализации. Коллекция или объекты (пути, точки, узлы, шины, защелки, регистры и т.д.), которые будут локализованы, при прохождении по ссылке на соответствующую коллекцию или объект. |

В примере 7-60 показано, как делается межпробник для десяти путей из временного анализатора TimeQuest в Планировщик Чипа, с последующей локализацией всех портов данных в Просмотрщике Технологической Карты.

**Example 7-60. Cross-probing from TimeQuest**

```
# Locate all of the nodes in the longest ten paths
# into the Chip Editor
locate [get_path -npaths 10] -chip

# locate all ports that begin with data to the Tech Map Viewer

locate [get_ports data*] -tmv
```