

Запуск сессии сбора данных




Вы можете выполнить сбор данных с помощью плагина Nios II таким же способом, как будто вы собираете данные с помощью логического анализатора SignalTap II. Сначала запрограммируйте FPGA сгенерированным в программе Quartus II **.sof** файлом. Затем запустите анализ SignalTap II либо вручную через менеджер элементов SignalTap II, либо автоматически после программирования FPGA, когда выбраны триггеры включения. Если система зафиксировала состояния триггера, логический анализатор SignalTap II отображает захваченные данные в окне результатов SignalTap II.

Вы можете использовать плагин Nios II для двух различных видов сессий захвата данных: первая - совместно с Nios II SBT на Eclipse, вторая - в автономном режиме.

Выполнение сбора данных совместно с Nios II SBT на Eclipse

Для использования плагина Nios II совместно с Nios II SBT на Eclipse, вы должны вручную загружать образ программы Nios II и контролировать работу процессора через отладчик. Вы можете использовать этот тип сессий сбора данных, когда вы разрабатываете и отлаживаете программное приложение Nios II.

Для запуска сессии сбора данных SignalTap II с процессором Nios II, контролируемым Nios II SBT на Eclipse, выполните следующие пункты:

1. В окне SignalTap II запрограммируйте чип FPGA с помощью сгенерированного в программе Quartus II **.sof** файла, выполнив следующие пункты:
 - a. В меню **Hardware** выберите кабель программирования, подключенный к плате разработчика Nios II.
 - b. В поле **SOF Manager** кликните **browse**.
 - c. В диалоге **Select Programming File** выберите **.sof** файл, сгенерированный для вашего проекта.
 - d. Кликните **Open**. Кнопка **Program Device** сейчас активна.
 - e. Кликните на кнопку **Program Device** для загрузки **.sof** файла в FPGA .
2. В окне SignalTap II на панели менеджера элементов кликните кнопку **Run Analysis** для запуска сессии сбора данных логического анализатора .
3. В Nios II SBT на Eclipse правым кликом выберите имя программного проекта, который вы хотите запустить на процессоре Nios II, и кликните **Debug As** затем **Nios II Hardware**. Этим действием вы запускаете отладчик, загружаете **.elf** файл в системную память и удерживаете процессор на точке входа в **main()**.
4. На вкладке **Debug** кликните на кнопку **Resume** для запуска исполнения процессора Nios II .

Логический анализатор SignalTap II продолжит исполнение, пока не будут достигнуты состояния триггеров, определённых в плагине Nios II. Пока запущен логический анализатор SignalTap II, вы можете использовать средства отладки Nios II SBT на Eclipse (например, вы можете устанавливать точки останова и останавливать процессор).

Когда вы запускаете отладчик, плагин Nios II срабатывает, когда процессор переходит на адрес триггера. Если расположение точки останова при запуске находится после расположения адреса триггера, заданного в плагине Nios II, то это может вызвать неверное срабатывание отладчика. Для изменения расположения точки останова запуска отладчика, следуйте следующим пунктам:

1. В меню **Run** кликните **Debug Configurations**. Появится окно **Debug Configurations**.
2. В окне **Debug Configurations** кликните на вкладку **Debugger**.
3. Выберите расположение точек останова, с которых вы хотите запускать отладчик и кликните **Apply**.

Альтернативно, вместо использования опции **Debug As**, вы можете использовать опцию **Run As**. Использование опции **Run As** принуждает Nios II SBT на Eclipse загрузить и запустить образ программы из системной памяти без запуска средства отладчика Nios II SBT на Eclipse.

Выполнение сбора данных без внешней загрузки программы

Если процессорная система Nios II сконфигурирована в режиме самозагрузки, без необходимости внешней загрузки программы, и вы выбрали средство триггеров включения, логический анализатор SignalTap II запускается автоматически после программирования FPGA.

В этом случае логический анализатор SignalTap II может уже иметь собранные данные. Чтобы определить, что данные получены, или что логический анализатор до сих пор работает, кликните **Run Analysis** в менеджере элементов SignalTap II.

За дополнительной информацией о создании самозагружающейся процессорной системе Nios II обратитесь к настольной книге программиста Nios II.

Анализ результатов

Плагин Nios II позволяет вам просматривать захваченные данные процессора Nios II. В этой главе описаны некоторые средства, позволяющие работать с собранными данными плагином Nios II.

Просмотр данных

Собранные в SignalTap II данные появляются на вкладке **Data** окна SignalTap II. Каждый отсчёт, собранный плагином Nios II, отображает следующую информацию:

- **Address** - адрес инструкции, отображаемый в шестнадцатеричном формате. Если в настройках конфигурации плагина был указан **.elf** файл, то адрес инструкции может быть трансформирован в имя символа и офсет.
- **Assembly Language Mnemonic** - Nios II ассемблер - эквивалент двоичного операционного кода инструкции.

Если в настройках конфигурации плагина был указан **.elf** файл, то программа перерабатывает файл сразу после завершения сбора данных и конвертирует адреса инструкций в виде имени символа и офсета. Такая переработка помогает предотвратить случайное использование устаревшего образа программы.

Используйте вкладки SignalTap II для контроля, вы сможете просматривать прокруткой во время исполнения программы в процессоре Nios II. Если заданный такт захвата соответствует такту процессора Nios II, то каждый нарастающий фронт будет связан с циклом инструкции.

Вам могут попасться одна или несколько "пустых" блоков инструкций - отведённых данных от процессора Nios II. Это означает, что нет исполняемых инструкций процессором Nios II во время этого тактового цикла. Такое поведение нормальное, оно может быть в следующих случаях:

- **Cache Miss** - Запрашиваемая инструкция по адресу генерирует пропуск в кэше инструкций, требуется дополнительный тактовый цикл для заполнения кэш линии и возврата инструкции.
- **Memory Contention or Speed** - Обращение по адресу инструкции в память, к которой требуется несколько тактовых циклов для доступа, или к памяти, уже контролируемой другой периферией или процессором.

Также вы можете наблюдать собранные плагином Nios II данные в формате файла списка SignalTap II. В этом табличном формате отсчёты отображаются в хронологических строках, декодированных по количеству отсчётов и ассоциированной с ними ассемблерной мнемоники. Формат файла списка очень похож на выход команды `nios2-elf-objdump`, что упрощает процесс анализа. Для создания файла списка SignalTap II в меню File выберите **Create/Update** и кликните **Create SignalTap II List File**.

Некоторые инструкции процессора Nios II занимают несколько тактовых циклов. За дополнительной информацией о количестве тактовых циклов, необходимых для конкретной инструкции, обратитесь к главе "[Подробности реализации ядра Nios II](#)" в настольной книге процессора Nios II

Сопоставление отведённых данных с .elf файлом процессора

Вы можете сравнить собранные плагином Nios II отведённые инструкции с образом программы, исполняемой в процессоре Nios II, изучив содержимое **objdump** файла. Вы можете использовать команду `nios2-elf-objdump` для создания **objdump** файла. Эта команда копирует **.elf** файл процессора в удобочитаемый формат **objdump**, который содержит фрагменты C/C++ кода, символьные имена функций, инструкции на ассемблере и адрес.

Исполняемый компонент командной строки `nios2-elf-objdump` является частью Nios II EDS. Вы можете сконфигурировать этот инструмент с помощью набора опций командной строки. Список опций конвертирования для команды `nios2-elf-objdump` доступен после ввода следующей команды в командной среде Nios II:

```
nios2-elf-objdump --help ➡
```

Несмотря на то, что **objdump** файл содержит большое количество информации, декодированное из **.elf** файла, инструкции процессора Nios II в этом файле выводятся построчно в следующем виде:

<Address>: <opcode> <Assembly Mnemonic>

Например, правильная инструкция:

200a8c0: e0800417 ldw r2,16(fp)

Сохранение и конвертирование собранных данных

Вы можете сохранять любые собранные в плагине Nios II данные с помощью средства журнала SignalTap II. Набор данных плагины Nios II сохраняется вместе с информацией **.elf** файла процессора. Чтобы разрешить сохранение данных в журнал, включите опцию **Data Log** в окне SignalTap II.

Плагин Nios II также поддерживает средство конвертирования данных SignalTap II. Для экспорта собранных данных, в меню File кликните и задайте **File Name** (имя файла), **Export Format** (экспортируемый формат) и **Clock Period**(тактовый период).