

# 1. Общее представление о Nios II Flash Programmer

## Введение

Многие аппаратные проекты с процессором Nios<sup>®</sup> II имеют на плате флеш память для хранения конфигурации FPGA или программы Nios II. Флеш программатор Nios II (Nios II Flash Programmer) является частью пакета разработки Nios II EDS. Он используется для программирования данных в чип флеш памяти, подключенной к Altera<sup>®</sup> FPGA.

Nios II Flash Programmer передаёт содержимое файла с помощью загрузочного кабеля Altera, такого как USB Blaster<sup>®</sup>, в систему Nios II, запущенную в FPGA, и инструктирует систему Nios II на запись данных во флеш память.

Nios II Flash Programmer может запрограммировать три типа содержимого во флеш память:

- Исполняемые файлы программы Nios II – множество систем используют флеш память для хранения неизменяемого кода программы или встроенного программного обеспечения. Система Nios II может загружаться из флеш памяти.
- Данные конфигурации FPGA – при подаче питания контроллер конфигурации FPGA на плате может прочитать данные о конфигурации FPGA из флеш памяти. В зависимости от проекта конфигурации контроллера, данные могут быть из нескольких файлов конфигурации, хранящихся во флеш памяти.
- Произвольные файлы данных - Nios II Flash Programmer может запрограммировать бинарный файл по произвольному офсету во флеш памяти для любых целей. Например, программа Nios II может использовать эти данные в качестве таблицы коэффициентов или в виде постоянной таблицы соответствия.

Вы можете использовать Nios II Flash Programmer для программирования следующих типов памяти:

- Общий интерфейс с памятью (CFI) – совместимой флеш памяти – CFI – это промышленный стандарт, предлагающий общий, независимый от производителя интерфейс с чипами флеш памяти.
- Чип Altera EPCS – для хранения данных конфигурации FPGA и исполняемой программы Nios II.

За дополнительной информацией о спецификации CFI, обратитесь к публикации JEDEC Common Flash Interface standard JESD68.01 и JEDEC publications JEP137x, доступной на сайте организации JEDEC Solid State Technology Association ([www.jedec.org](http://www.jedec.org)). За дополнительной информацией о чипах EPCS, обратитесь к [технической документации на чипы \(EPCS1, EPCS4, EPCS16, EPCS64 и EPCS128\)](#) в томе 2 Altera Configuration Handbook и к главе [Ядро контроллера чипа EPCS](#) в томе 5: Встроенная периферия Настольной книги Quartus II.

В этом документе термин флеш памяти относится к обоим типам чипов CFI и EPCS, если это не указано дополнительно.

### Необходимые условия

В этом руководстве пользователя подразумевается, что вы уже знакомы с процессом аппаратной и программной разработки Nios II. Вам необходимо ознакомиться со следующими документами:

- [Учебное руководство по аппаратной разработке Nios II](#),
- Глава [Начало работы с графическим интерфейсом пользователя](#) в Настольной книге программиста Nios II.

Если вы используете Nios II Flash Programmer для программирования во флеш память данных конфигурации FPGA, вам необходимо знать метод конфигурации, используемый на этой плате.

Обратитесь к [AN346](#): Using the Nios II Configuration Controller Reference Designs или к руководству на вашу оценочную плату Altera.

### Графическая оболочка Nios II Flash Programmer и утилиты командной строки

Вы можете запускать Nios II Flash Programmer из GUI или из командной строки. GUI отображает эквивалент выполненных вами действий в командной строке.

За дополнительной информацией о flash programmer GUI, обратитесь к главе 2, "Использование Flash Programmer GUI", а за информацией об утилитах командной строки – к главе 3, "Использование Flash Programmer из командной строки".

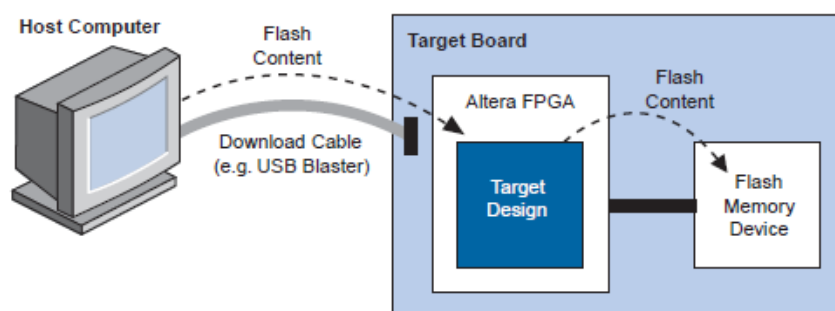
Следующие инструменты позволяют вам запускать Nios II Flash Programmer:

- Nios II Software Build Tools for Eclipse™ - предлагает простой доступ к Nios II Flash Programmer GUI. GUI имеет простой в использовании интерфейс, позволяющий вам контролировать средство флеш программатора, используя настройки, которые вы сохраняли для многократного использования. Такой метод доступа подходит для большинства задач программирования.
- Командная среда Nios II – предлагает команды, которые контролируют средства флеш программатора. Вам необходимо рассчитывать некоторые параметры вручную. Также из командной строки вы сможете запустить Nios II Flash Programmer GUI.

### Как работает Flash Programmer

Nios II Flash Programmer делится на две части: хост и целевое устройство, как это показано на рис. 1-1. Часть хост запускается на вашем компьютере. Она запускает файлы флеш программирования и команды программирования через загрузочный кабель в целевое устройство. Часть целевого устройства – это аппаратный проект, запущенный в FPGA. Он задаёт данные для программирования – содержимое флеш и требуемую информацию о выбранном чипе флеш памяти – посылаемые хостом, и следует инструкциям для записи данных во флеш память.

**Figure 1-1. How the Nios II Flash Programmer Works**



## Целевой проект для Nios II Flash Programmer

Для использования Nios II Flash Programmer вы должны иметь соответствующий целевой проект флеш программатора, загруженный в вашу плату. Этот проект должен состоять из системы SOPC Builder с некоторыми компонентами SOPC Builder, отображёнными в табл. 1-1.

Чипы Hardcopy® II также поддерживают программирование CFI Flash посредством Nios II Flash Programmer, если проект Hardcopy II удовлетворяет требованиям, описанным в этом документе.

Минимальный набор компонентов обеспечивает целевому проекту связь с хостом и запись во флеш память. Минимальный набор компонентов зависит от типа флеш памяти, предназначенной для программирования. В табл. 1-1 перечислены минимальные наборы компонентов для программирования каждого типа флеш памяти.

**Табл. 1-1.** Минимальный набор компонентов для целевого проекта Flash Programmer

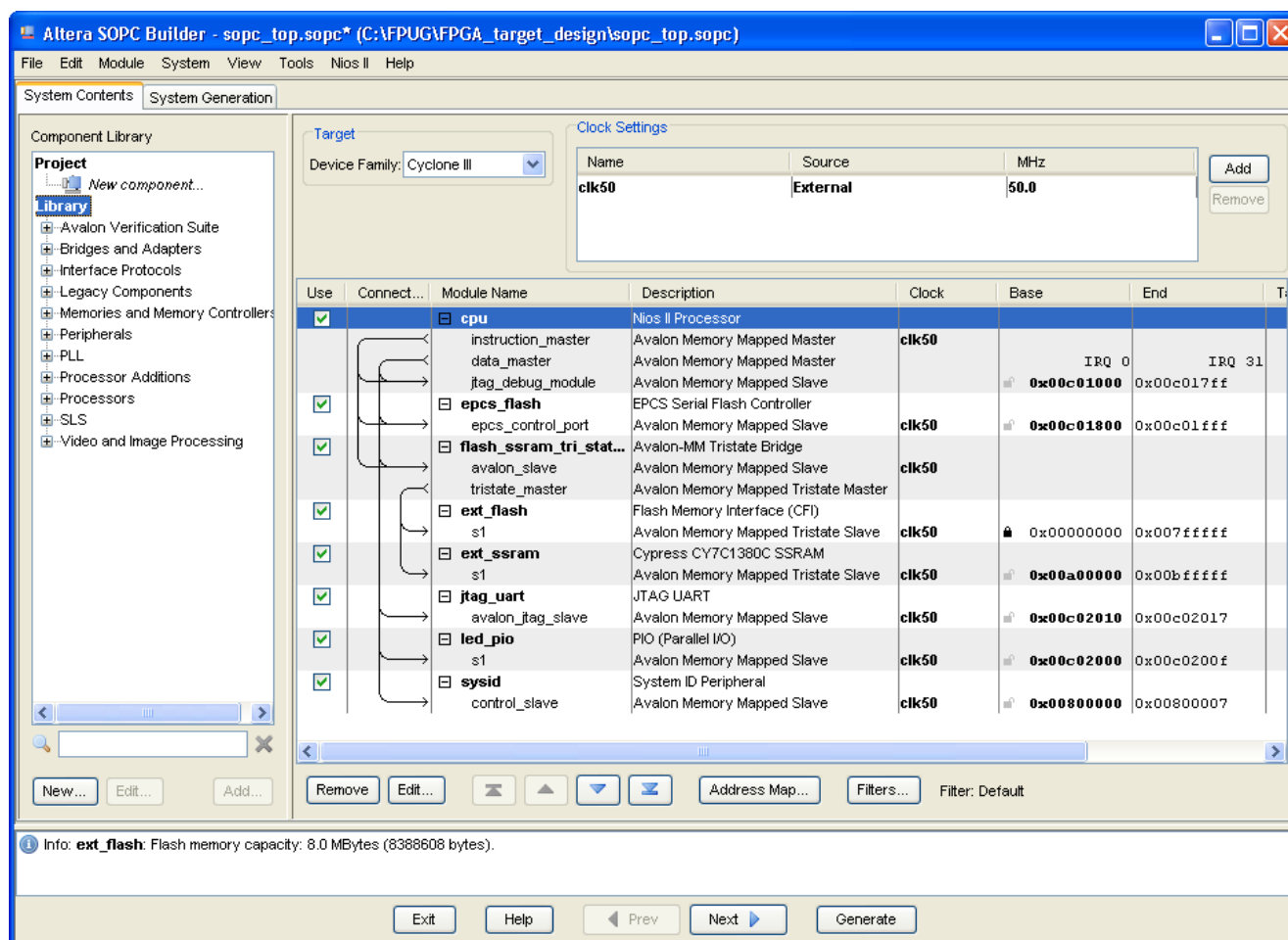
Компонент	Флеш память для программирования		
	только CFI	только EPCS	CFI и EPCS
Процессор Nios II с модулем JTAG уровня 1 и выше	требуется	требуется	требуется
Периферия System ID	рекомендовано (1)	рекомендовано (1)	рекомендовано (1)
Флеш память (Common Flash Interface)	требуется (2)		требуется (2)
Avalon-MM Tristate Bridge	требуется (3)		требуется (3)
EPCS Serial Flash Controller		требуется (3)	требуется (3)

### Примечание к табл. 1-1:

- (1) Если компонент периферии System ID присутствует, это позволяет Nios II Flash Programmer проверять целевой проект перед программированием флеш памяти. Система Nios II может иметь интерфейс более чем с одним чипом памяти CFI.
- (2) При этом она должна иметь один компонент флеш памяти (Common Flash Interface) для каждого чипа флеш памяти на плате.
- (3) Один компонент Avalon-MM Tristate Bridge подключается к нескольким контроллерам флеш памяти.

На рис. 1-2 показан пример системы SOPC Builder, состоящей из минимального набора компонентов для системы с одним чипом CFI флеш памяти и одним чипом конфигурации EPCS. Система также содержит и другие компоненты, относящиеся к функционированию системы, но не флеш программатора.

**Рис. 1-2.** Пример целевого проекта с минимальным набором компонентов



Большинство примеров аппаратных проектов, включающих Nios II EDS, являются готовыми целевыми проектами, которые можно использовать для программирования флеш памяти на оценочных платах Altera. Если вы разрабатываете собственную плату, вы можете использовать один из этих примеров проектов в качестве отправной точки для вашего целевого проекта флеш программатора.

## **2. Использование Flash Programmer GUI**

### **Введение**

Nios II Flash Programmer GUI - это простой в использовании графический интерфейс, автоматизирующий процесс программирования флеш памяти и позволяющий вам легко контролировать параметры программирования. Nios II Flash Programmer GUI передаёт вашу программу в любой комбинации программы, аппаратной части и бинарных данных во флеш память за одну операцию. Nios II Flash Programmer GUI также генерирует флеш файлы для дальнейшего использования и хранит их без программирования их во флеш память. При генерировании флеш файлов или при программировании флеш памяти из flash programmer GUI генерируются скрипты, которые можно использовать в дальнейшем из командной строки.

Вы можете запустить Nios II Flash Programmer GUI из Nios II SBT for Eclipse или из командной строки. Также вы можете использовать флеш программатор из командной строки. В главе 3, "Использование Flash Programmer из командной строки" описаны утилиты командной строки флеш программатора.

Altera рекомендует вам использовать Nios II Flash Programmer GUI для генерирования автоматических скриптов, и для использования этих скриптов для автоматизации Nios II процесса программирования флеш.

### **Запуск Flash Programmer GUI**

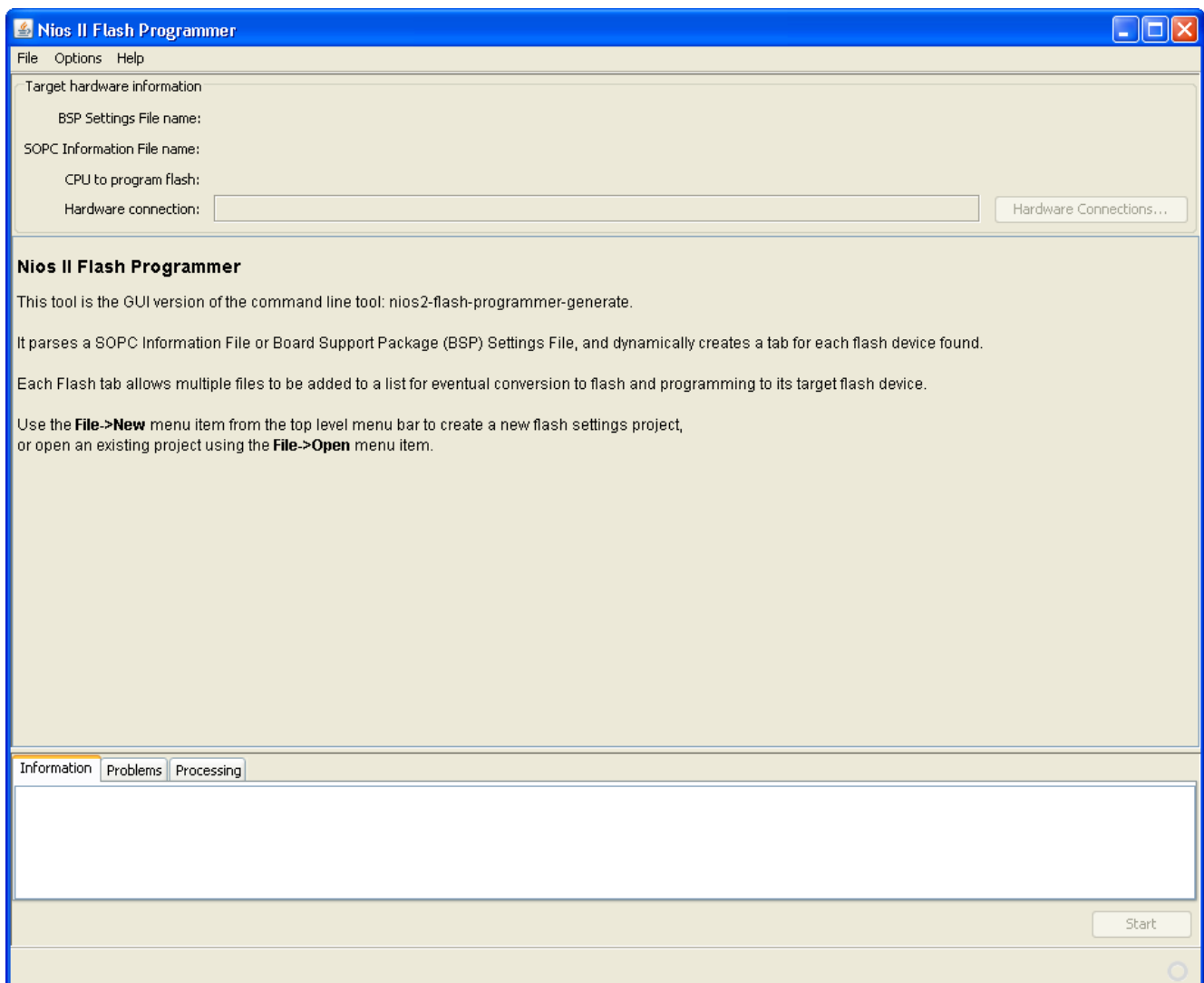
Для запуска Nios II Flash Programmer GUI из командной строки (Nios II Command Shell) введите следующую команду:

```
nios2-flash-programmer-gui ➤
```

Для запуска Nios II Flash Programmer GUI из Nios II SBT for Eclipse в меню Nios II выберите Flash Programmer.

Появляется диалоговое окно Flash Programmer, как показано на рис. 2-1.

Рис. 2-1. Диалоговое окно Flash Programmer



### Задание настроек Flash Programmer

Прежде чем записать данные во флеш память, вы должны определить настройки флеш программатора.

Для создания нового набора настроек флеш программатора, выполните следующие пункты:

1. В меню **File** выберите **New**. Появится диалоговое окно **New Flash Programmer Settings File**.

2. Выберите **Get flash programmer system details from BSP Settings File** (получение информации для флеш программатора о системе из файла настроек BSP) или **Get flash programmer system details from SOPC Information File** (получение информации для флеш программатора о системе из файла информации SOPC).

3. Найдите ваш файл настроек BSP (.bsp) или файл информации SOPC (.sopcinfo).

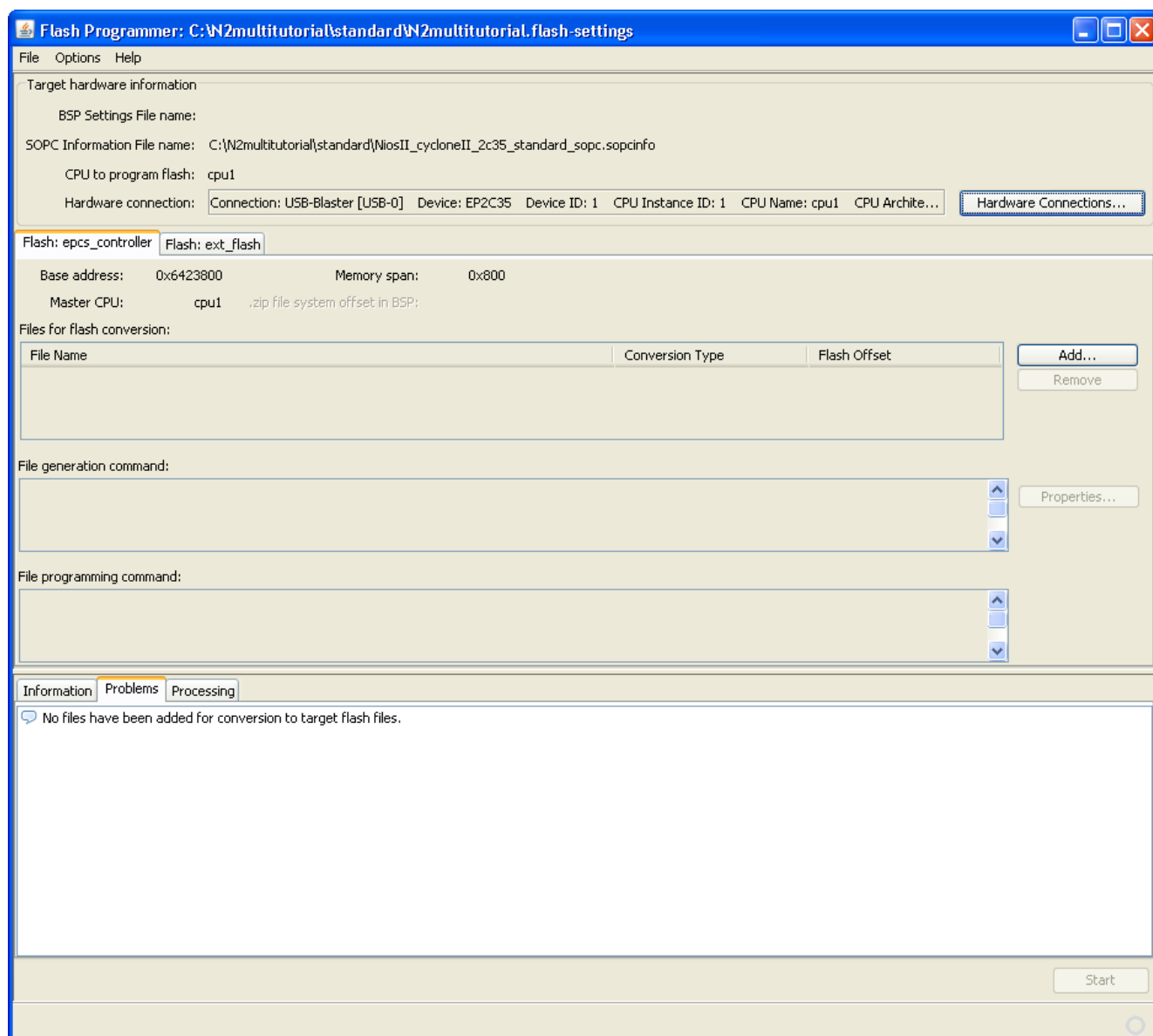
4. Для мультипроцессорных систем выберите процессор. При выборе .bsp файла процессор задаётся автоматически.

5. Кликните **OK**. Закроется диалоговое окно **Flash Programmer Settings File**, а Nios II Flash Programmer GUI будет отображать информацию о вашем процессоре, и информацию из файла .bsp или .sopcinfo.

В зависимости от вашего выбора в пункте 2, настройки вашего флеш программатора могут не иметь информации о файле настроек BSP. Новый набор настроек флеш программатора, основанный на файле настроек BSP, содержит информацию об имени файла .sopcinfo, тогда как новый набор настроек флеш программатора, основанный на файле .sopcinfo не может иметь информации о файле настроек BSP, поскольку они не связаны. В этом случае вам необходимо напрямую идентифицировать .bsp файл. Файл .bsp содержит информацию о секции вашего процессора, тогда как файл .sopcinfo имеет информацию обо всех доступных процессорах, но ни один из них не выбран.

На рис. 2-2 показан флеш программатор с новым набором настроек, основанных на файле .sopcinfo, для проекта с несколькими процессорами и двумя компонентами флеш памяти.

**Рис. 2-2.** Диалоговое окно Flash Programmer с настройками



### Работа с файлами настроек Flash Programmer

Nios II Flash Programmer GUI - это полный набор инструментов с множеством опций. Altera рекомендует вам сохранять ваши настройки флеш программатора в



файле настроек Nios II Flash Programmer Settings File (.flash-settings), чтобы использовать их в дальнейшем.

Файл .flash-settings содержит настройки, описанные в следующих параграфах, такие как, пути к директориям файлов скриптов и флеш, необходимо ли выполнять проверку system ID и времени создания системы, и нужно ли генерировать флеш файлы или программировать флеш память.

Для сохранения текущих настроек флеш программатора, в меню **File** выберите **Save** или **Save As** для обновления или создания .flash-settings файла. После того, как вы сохранили файл, вы можете продолжить редактирование его в Nios II Flash Programmer GUI.

Для открытия существующего файла настроек флеш программатора, в меню **File** выберите **Open** и найдите .flash-settings файл.

### Установка аппаратного подключения

В этом параграфе описывается, как выбрать корректный загрузочный кабель, чип и процессор для программирования флеш памяти. Если ваша система имеет только один загрузочный кабель и один процессор, то процесс упрощается. В этом параграфе описываются все пункты для систем с несколькими загрузочными кабелями, процессорами и чипами.

Прежде чем вы сможете запрограммировать флеш память на вашей плате, вы должны сконфигурировать ваш FPGA целевым проектом флеш программатора, содержащим минимальный набор компонентов, определённый в табл. 1-1 на стр. 1-3.

За инструкциями о программировании FPGA обратитесь к главе "[Quartus II Programmer](#)" в томе 3 Настольной книги Quartus II.

После загрузки целевого проекта в ваш FPGA, вы можете установить аппаратное подключение для программирования флеш памяти.

Для задания аппаратного подключения выполните следующие пункты:

1. Выберите **Hardware Connections**. Откроется диалоговое окно Hardware Connections (аппаратные подключения).
2. В диалоговом окне Hardware Connections выберите **Refresh Connections** (обновить подключения).
3. Если вы применяете .flash-settings файл, и проект Quartus® II был перекомпилирован после создания .flash-settings файла или столбец **Name** в списке **Processors** пуст, выполните следующие пункты:
  - a. Рядом с полем JTAG Debugging Information File name (имя файла информации об отладчике JTAG) разверните и найдите файл JTAG Debugging Information (.jdi) вашего проекта.
  - b. Кликните **Resolve Names** (разрешить имена). Флеш программатор использует .jdi файл для точного поиска доступной информации о подключении.
4. Если ваш проект имеет несколько загрузочных кабелей, выберите нужный.
5. Если ваш проект мультипроцессорный, выберите процессор Nios II, который соответствует CPU to program flash (процессору для программирования флеш) под Target hardware information в диалоговом окне Nios II Flash Programmer.
6. Выберите **Close**.

### Проверка System ID и времени создания системы

Если целевой проект флеш программатора содержит компонент System ID, Nios II Flash Programmer может выполнять проверку ID системы и времени создания системы перед программированием флеш памяти. Когда флеш программатор



выполняет проверку ID системы и(или) времени создания системы, а имеющаяся система не сконфигурирована в FPGA, флеш программатор не будет программировать флеш память.

Altera рекомендует вам использовать в целевом FPGA проекте компонент System ID, а также, чтобы вы разрешили проверку ID системы и времени создания системы.

По умолчанию, обе проверки ID системы и времени создания системы разрешены. Для запрета проверки ID системы или времени создания системы выполните следующие пункты:

1. Выберите **Hardware Connections**. Откроется диалоговое окно Hardware Connections.
2. Для запрета проверки ID системы включите **Ignore mismatched system ID** (игнорирование несовпадающего ID системы).
3. Для запрета проверки времени создания системы включите **Ignore mismatched system timestamp** (игнорирование несовпадающего времени создания системы).
4. Выберите **Close**.

После задания аппаратных подключений вы можете запросить ID и время создания системы, выполнив следующие пункты:

1. Выберите **Hardware Connections**. Откроется диалоговое окно Hardware Connections.
2. Выберите **System ID Properties**.
3. Проверьте соответствие значений Expected system ID и Actual system ID, и что значения ожидаемого ID системы соответствует актуальному ID системы.
4. В диалоговом окне System ID Properties выберите **Close**.
5. В диалоговом окне Hardware Connections выберите **Close**.

За дополнительной информацией об ID системы и времени создания системы обратитесь к табл. 3-2 на стр. 3-2.

В зависимости от значения компонента System ID, вы не сможете запрограммировать флеш память, если аппаратный проект, сконфигурированный в FPGA, не имеет корректного целевого проекта флеш программатора, содержащего минимальный набор компонентов, перечисленных в табл. 1-1 на стр. 1-3.

### Генерирование флеш файлов и программирование флеш памяти

Nios II Flash Programmer может генерировать флеш файлы и программировать флеш память с помощью флеш файла. Флеш программатор может генерировать флеш файлы из следующих различных типов файлов:

- SRAM Object File (.sof) - содержит данные конфигурации FPGA,
- формат исполняемого и компоновочного файла (.elf) - содержит вашу исполняемую программу,
- файл Altera Zip файловой системы только для чтения (.zip) - содержит zip файловую систему только для чтения, ассоциированную с вашей программой Nios II,
- произвольный бинарный файл.

Nios II EDS предлагает программный компонент файловой системы только для чтения Altera Zip, который является простым в использовании инструментом для хранения и доступа к данным во флеш памяти. В зависимости от вашего приложения, вы можете находить использование Zip системы более удобным по сравнению с хранением обычных бинарных данных во флеш памяти. За информацией о файловой

системе только для чтения Altera Zip обратитесь к главе "[Файловая система только для чтения Altera Zip](#)" в Настольной книге программиста Nios II.

Для генерации флеш файлов или для записи во флеш память, используя настройки вашего флеш программатора, выполните следующие пункты:

1. В меню Options включите действия, которые вы собираетесь выполнить.

Доступны следующие действия:

- Generate Files - генерирование флеш файлов,
- Program Files - программирование флеш памяти сгенерированными флеш файлами,
- Erase Flash Before Programming - стирает выбранную флеш память перед записью в неё каждого флеш файла.
- Run From Reset After Programming - запускает процессор по его вектору сброса после завершения программирования флеш памяти.

Последние два пункта зависят от того, что вы выбрали Program Files.

2. Для определения директории, в которой вы хотите, чтобы флеш программатор сохранял сгенерированные флеш файлы и скрипты, в меню Options выберите Staging Directories и задайте пути к директории скриптов и флеш файлов.

3. На вкладке для целевого устройства флеш памяти выберите **Add**.

Nios II Flash Programmer добавляет файл в список Files for flash conversion (файлов для конвертации во флеш) и вычисляет флеш офсет из файла. В зависимости от текущих настроек, окна File generation command (команда генерации файла) и File programming command (команда программирования файла) отображают командную строку с командами, которые генерируют .flash файл и программируют его в вашу флеш память соответственно, предлагая удобный способ изучить утилиты командной строки.

4. Если вы хотите добавить любые дополнительные аргументы к команде генерации файла, под File generation command выберите Properties.

5. Если необходимо, отредактируйте настройки Conversion Type (тип конвертации) для файлов, перечисленных в таблице Files for flash conversion (файлы для флеш конвертации), кликнув на соответствующую ячейку таблицы. Разрешённые для конвертации типы файлов: ELF, SOF и BINARY.

6. Если потребуется, отредактируйте настройку Flash Offset (офсет флеш) для файлов, перечисленных в таблице Files for flash conversion дважды кликнув на соответствующие ячейки таблицы.

Не требуется задавать офсет флеш для файлов конвертации типа ELF.

7. Выберите **Start**. Nios II Flash Programmer выполнит действия, заданные командами: сохранит сгенерированные флеш файлы в заданной флеш директории и выполнит команды их по запуску в flash\_programmer.sh bash shell скрипта в заданной директории скриптов.

Вы можете использовать этот скрипт для повторения этих же самых действий в будущем и для изучения утилит командной строки.