



## **4. EPCS Serial Flash Controller Core**

### **4. Ядро последовательного флеш контроллера EPCS**

#### **Общее представление о ядре**

Ядро последовательного флеш контроллера EPCS с интерфейсом Avalon® предоставляет доступ системам Nios® II к чипам последовательной конфигурации Altera® EPCS. Altera предоставляет драйверы, которые интегрированы в системную библиотеку слоя аппаратной абстракции (HAL) Nios II, позволяя вам читать и записывать в чип EPCS, используя схожий интерфейс прикладного программирования HAL API для флеш устройств.

Используя ядро последовательного флеш контроллера EPCS, система Nios II может:

- Хранить код программы в чипе EPCS. Ядро последовательного флеш контроллера EPCS предоставляет средство загрузчика, позволяющее системе Nios II хранить основной код программы в чипе EPCS.
- Хранить в энергонезависимом устройстве программные данные, такие как серийный номер, сетевой номер (NIC) и прочие постоянные данные.
- Управлять данными конфигурации чипа. Например, встроенная система с разрешённым подключением к сети может принимать новые данные конфигурации FPGA через сеть, и использовать ядро для программирования новых данных в чип последовательной конфигурации EPCS.

Ядро последовательного флеш контроллера EPCS предназначено для SOPC Builder и легко интегрируется в любую систему, генерируемую SOPC Builder. Утилита флеш программатора в Nios II IDE позволяет вам управлять содержимым и программировать данные в чипе EPCS.

За подробной информацией о семействе чипов последовательной конфигурации EPCS, обратитесь к технической документации на чипы последовательной конфигурации (EPCS1, EPCS4, EPCS16, EPCS64 и EPCS128). За подробной информацией об использовании Nios II HAL API для чтения и записи флеш памяти, обратитесь к настольной книге программиста Nios II. За подробной информацией об управлении содержимым и программировании данных в чипе EPCS, обратитесь к руководству пользователя флеш программатором Nios II.

Для использующих процессор Nios II, ядро последовательного флеш контроллера EPCS заменяет устройство активного последовательного интерфейса с памятью (ASMI). Новые проекты должны использовать ядро последовательного флеш контроллера EPCS вместо ядра ASMI.

Эта глава состоит из следующих секций:

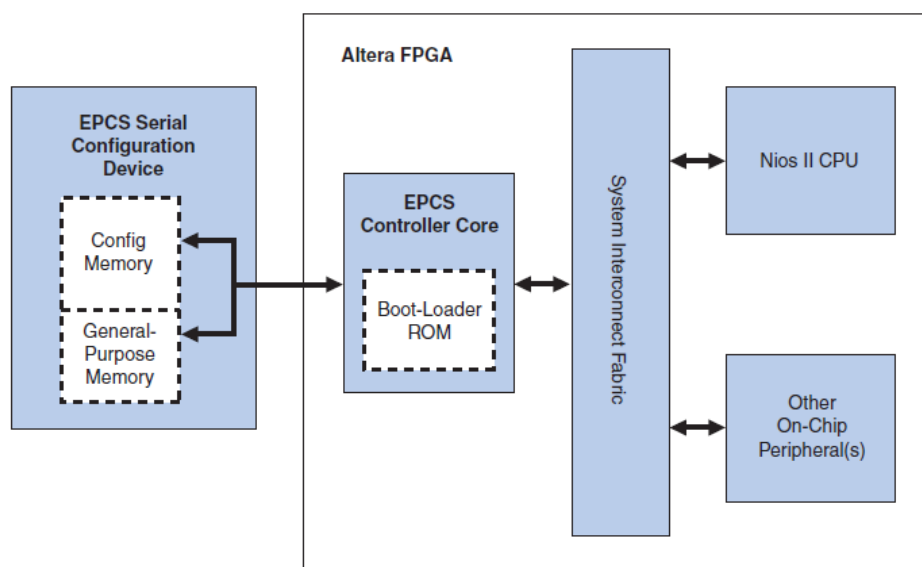
- "Функциональное описание" на странице 4–2
- "Поддержка чипов и инструментов" на странице 4–4
- "Инсталляция ядра в SOPC Builder" на странице 4–4
- "Программная модель" на странице 4–4

## Функциональное описание

На рис. 4-1 показана блок диаграмма ядра последовательного флеш контроллера EPCS при обычной конфигурации системы. Как показано на рис. 4-1, память в чипе EPCS состоит из двух отдельных регионов:

- Память конфигурации FPGA – в этом регионе находятся данные о конфигурации FPGA.
- Память общего назначения – если данные о конфигурации FPGA не занимают весь чип EPCS, то остаток памяти может быть использован под данные общего назначения или под код запуска системы.

**Figure 4-1.** Nios II System Integrating an EPCS Serial Flash Controller Core



Достоинством общей модели устройства HAL для флеш устройств, является возможность доступа к чипу EPCS, используя HAL API, точно также как и доступ к любому другому флеш устройству. Чип EPCS имеет аппаратный интерфейс специального назначения, поэтому Nios II программы должны читать и записывать в EPCS память, используя предоставляемые HAL флеш драйверы.

Ядро последовательного флеш контроллера EPCS имеет память на чипе для хранения программы загрузчика. Если оно используется вместе с чипами Cyclone® and Cyclone II, то ядру требуется 512 байт под ROM загрузчика. Для чипов Cyclone III, Cyclone IV, Stratix® II и других новых семейств чипов серии Stratix, ядру требуется 1 Кбайт под ROM загрузчика. Процессор Nios II может быть сконфигурирован на загрузку из ядра последовательного флеш контроллера EPCS. Чтобы сделать это, необходимо установить адрес сброса на начальный адрес ядра последовательного флеш контроллера EPCS. В этом случае, после сброса процессора, сначала выполняется код из ROM загрузчика, который копирует данные из региона общего назначения в RAM. Кроме того, программа контролирует трансферты в RAM. Nios II IDE предоставляет средства для компиляции программы с опцией хранения её в чипе EPCS и создаёт файл программирования для программирования в чип EPCS.

За подробной информацией обратитесь к руководству пользователя флеш программатором Nios II.

Если вы программируете чип EPCS, используя программатор Quartus® II, всё содержимое чипа стирается. Для программирования чипа EPCS комбинацией данных конфигурации FPGA и данных программы Nios II, используйте утилиту флеш программатора Nios II IDE.

Чип конфигурации Altera EPCS подключен к FPGA посредством специальных выводов FPGA, а не через I/O выходы общего назначения. Во всех семействах чипов Altera, за исключением Cyclone III и Cyclone IV, ядро последовательного флеш контроллера EPCS не может создавать какие-либо I/O порты в модуле верхнего уровня системы SOPC Builder. Если чипы EPCS и FPGA соединены на печатной плате, для конфигурации используется чип EPCS (другими словами, режим активной последовательной конфигурации), создание дополнительного соединения между чипом EPCS и ядром последовательного флеш контроллера EPCS не требуется. Когда вы компилируете систему SOPC Builder в программе Quartus II, сигналы ядра последовательного флеш контроллера EPCS автоматически разводятся на выводы чипа для подключения к EPCS.

Однако вы можете воспользоваться опцией "не использовать специальные выводы FPGA" (режим активной последовательной конфигурации), выключив соответствующие параметры в интерфейсе MegaWizard. Когда эта опция выключена или когда выбран чип Cyclone III или Cyclone IV, вы можете воспользоваться гибкостью подключения внешних выводов, которые экспортируются в проект верхнего уровня, к любому чипу EPCS. Выполнение следующих пунктов в программе Quartus II создаст необходимые назначения для выводов:

- На странице **Dual-purpose pins (Assignments > Devices > Device and Pin Options)**, проследите за тем, чтобы следующим выводам были заданы соответствующие значения:
  - Data[0] = Use as regular I/O
  - Data[1] = Use as regular I/O
  - DCLK = Use as regular I/O
  - FLASH\_nCE/nCS0 = Use as regular I/O
- Используя планировщик выводов (**Assignments > Pins**), проследите за тем, чтобы следующим выводам были заданы соответствующие функции конфигурации чипа:
  - data0\_to\_the\_epcs\_controller = DATA0
  - sdo\_from\_the\_epcs\_controller = DATA1,ASDO
  - dclk\_from\_epcs\_controller = DCLK
  - sce\_from\_the\_epcs\_controller = FLASH\_nCE

За дополнительной информацией о конфигурационных выводах на чипе Altera, обратитесь к странице "Файлы выводов для чипов Altera"

#### Интерфейс Avalon-MM слейв и регистры

Ядро последовательного флеш контроллера EPCS имеет единственный интерфейс Avalon-MM слейв, который предоставляет доступ к загрузочному коду и регистрам контроля ядра. Как показано в табл. 4-1, первый сегмент предназначен для загрузочного кода, а следующие семь слов являются регистрами контроля и данных. Процессор Nios II может читать слова инструкций, начиная с базового адреса ядра, как однородное пространство памяти, которое позволяет процессору сбросить адресное пространство ядра.

Ядро последовательного флеш контроллера EPCS содержит сигнал прерывания, который может использоваться для сброса процессора по окончании трансфера.

**Table 4-1.** EPCS Serial Flash Controller Core Register Map

Offset—Cyclone and Cyclone II (32-bit Word Address)	Offset—Other Device Families (32-bit Word Address)	Register Name	R/W	Bit Description
				31:0
0x00 .. 0x7F	0x00 .. 0xFF	Boot ROM Memory	R	Boot Loader Code
0x080	0x100	Read Data	R	(1)
0x081	0x101	Write Data	W	
0x082	0x102	Status	R/W	
0x083	0x103	Control	R/W	
0x084	0x104	Reserved	—	
0x085	0x105	Slave Enable	R/W	
0x086	0x106	End of Packet	R/W	

Примечание к табл. 4-1:

(1) Altera не разглашает правила использования регистров контроля и данных. Для доступа к чипу EPCS вы должны использовать драйверы HAL, предлагаемые Altera.

## Поддержка чипов и инструментов

Ядро последовательного флеш контроллера EPCS поддерживается всеми семействами чипов Altera, за исключением серии Hardcopy®. Ядро должно быть подключено к процессору Nios II. Ядро предлагает драйверы для системы Nios II, основанной на HAL, и делает предкомпиляцию кода загрузчика под процессор Nios II.

## Инсталляция ядра в SOPC Builder

Вы можете добавить ядро последовательного флеш контроллера EPCS на вкладке **System Contents** в SOPC Builder. В семействах чипов Cyclone III и Cyclone IV, вы можете использовать интерфейс MegaWizard™ для конфигурирования ядра, чтобы использовать I/O выводы вместо специальных выводов, выключив оба параметра: **Automatically select dedicated active serial interface, if supported** и **Use dedicated active serial interface**.

Только одно ядро последовательного флеш контроллера EPCS может быть использовано в проекте FPGA.

## Программная модель

В этой секции описывается программная модель ядра последовательного флеш контроллера EPCS. Altera предлагает драйверы системной библиотеки HAL, которые позволяют вам стирать и записывать EPCS память, используя функции HAL API. Altera не разглашает правила использования регистров контроля и данных. Поэтому вы должны использовать драйверы HAL, предлагаемые Altera, для доступа к чипу EPCS.

### Поддержка системной библиотеки

Предлагаемый Altera драйвер реализует HAL драйвер флеш устройства, который интегрирован в системную библиотеку HAL для системы Nios II. Программы вызывают родственные функции HAL API для программирования памяти EPCS. Вам нужно детально разбираться в основных драйверах, чтобы их использовать.

Драйвер для чипа EPCS исключается, когда разрешается опция уменьшения драйверов устройств в BSP или системной библиотеке. Чтобы форсировать включение драйверов EPCS в BSP с разрешённой опцией уменьшения драйверов устройств (reduced device drivers), вы можете задать символ предпроцессора - ALT\_USE\_EPCS\_FLASH – перед включением заголовочного файла, т.е.:

```
#define ALT_USE_EPCS_FLASH  
  
#include <altera_avalon_epcs_flash_controller.h>
```

HAL API для программирования флеш памяти, включая примеры Си кода, описано подробнее в настольной книге программиста Nios II. За подробной информацией об управлении содержимым и программировании данных в чипе EPCS, обратитесь к руководству пользователя флеш программатором Nios II.

#### Программные файлы

Ядро последовательного флеш контроллера EPCS предлагает следующие программные файлы. Эти файлы предоставляют доступ на аппаратном уровне к устройству и управляют его интеграцией в системную библиотеку the Nios II HAL. Разработчики приложений не должны модифицировать эти файлы.

- **altera\_avalon\_epcs\_flash\_controller.h, altera\_avalon\_epcs\_flash\_controller.c** – заголовочный и исходный файлы, которые определяют драйверы, необходимые для интеграции в системную библиотеку HAL.
- **epcs\_commands.h, epcs\_commands.c** - заголовочный и исходный файлы, которые напрямую контролируют аппаратную часть устройства EPCS при чтении и записи в него. Эти файлы также зависят от драйверов ядра Altera SPI.

Оглавление

4. Ядро последовательного флеш контроллера EPCS .....	4-1
Общее представление о ядре .....	4-1
Функциональное описание.....	4-2
Интерфейс Avalon-MM слейв и регистры .....	4-3
Поддержка чипов и инструментов.....	4-4
Инсталляция ядра в SOPC Builder .....	4-4
Программная модель .....	4-4
Поддержка системной библиотеки .....	4-4
Программные файлы .....	4-5