

Окончание. Начало в № 2 `2007

Валерий ЗОТОВ
walerry@km.ru

Способы подготовки файлов, предназначенных для инициализации содержимого элементов запоминающих устройств, формируемых с помощью генератора параметризованных модулей CORE Generator

При подготовке описаний элементов постоянной и оперативной памяти с помощью генератора параметризованных модулей CORE Generator для определения содержимого формируемого запоминающего устройства, которое автоматически записывается в этот элемент сразу после завершения процесса конфигурирования ПЛИС, можно изменять файлы с расширением COE. Файлы этого типа имеют текстовый формат (ASCII), поэтому для их создания и модификации можно использовать различные текстовые редакторы. В частности, файлы, предназначенные для инициализации содержимого элементов запоминающих устройств, можно создать с помощью редактора *Блокнот (Notepad)*, входящего в состав операционных систем Windows 2000 и Windows XP.

Синтаксис COE-файлов был подробно рассмотрен при описании параметризованного модуля Distributed Memory Generator (см. КиТ № 4, 2007 г.), поэтому в настоящем разделе кратко описывается только структура этих файлов. В начале файла, предназначенного для инициализации содержимого элементов запоминающих устройств, с помощью ключевого слова *memory_initialization_radix* определяется используемый формат представления данных (двоичный, десятичный или шестнадцатеричный). Далее в форме вектора *memory_initialization_vector* следует массив данных, которые должны быть записаны в соответствующие ячейки генерируемого элемента памяти. В качестве приме-

Разработка базовых компонентов цифровых устройств, реализуемых на базе ПЛИС FPGA фирмы Xilinx, с помощью генератора параметризованных модулей CORE Generator

ра приведен текст COE-файла, сформированного для инициализации содержимого запоминающего устройства с информационной емкостью 512 бит и организацией 32 слова × 16 разрядов:

```
MEMORY_INITIALIZATION_RADIX=2;
MEMORY_INITIALIZATION_VECTOR=
0010000000000000,
0000000000000001,
0000000000100101,
0000000000110000,
0010000001000100,
0000000000010111,
0000000000110010,
0000000000010010,
0000000000101010,
0000000001100101,
0000000001010101,
000000000100100,
001000000110100,
0000000000011001,
0000000001110010,
0000000001100111,
0000000000010100,
0000000000100010,
0000000000100011,
0000000000010000,
0000000000010001,
0000000000111000,
0000000001100001,
0000010001100011,
0000001001110001,
0000000001110010,
0000000001010111,
0000000010000011,
0000000000110011,
0000000000010101,
0100000000100000;
```

В представленном файле информационные данные, предназначенные для автоматической загрузки в ячейки сгенерированного элемента памяти, записаны в двоичном формате.

Подготовку файлов инициализации содержимого запоминающих устройств с помощью текстовых редакторов целесообразно осуществлять в тех случаях, когда информационная емкость формируемого элемента памяти невелика. В большинстве случаев для создания файлов, определяющих содержимое элементов памяти в формате COE, рекомендо-

дуется использовать специализированный редактор Memory Editor, входящий в состав генератора параметризованных модулей CORE Generator. Этот инструмент не только повышает наглядность процесса подготовки файлов инициализации для элементов ОЗУ и ПЗУ, сгенерированных на основе соответствующих параметризованных модулей, но и предоставляет возможность существенно ускорить выполнение этого процесса. Применение встроенного специализированного редактора Memory Editor позволяет значительно снизить количество возможных ошибок, вызванных неправильной адресацией указываемых информационных данных.

Пользовательский интерфейс, входящего в состав генератора параметризованных модулей CORE Generator редактора файлов Memory Editor, предназначенных для инициализации содержимого элементов запоминающих устройств

Основу пользовательского интерфейса интегрированного редактора файлов, предназначенных для автоматической инициализации содержимого элементов запоминающих устройств, формируемых с помощью генератора параметризованных модулей CORE Generator, образуют два взаимосвязанных основных окна, которые одновременно отображаются на экране монитора. Окно с заголовком *Memory Editor* предназначено для определения параметров элемента памяти, для которого создается файл описания загружаемых данных, и формата представления информации, используемого этом файле. В этом же окне сосредоточены инструменты управления процессом генерации файлов описания содержимого элементов запоминающих устройств в различных форматах (COE

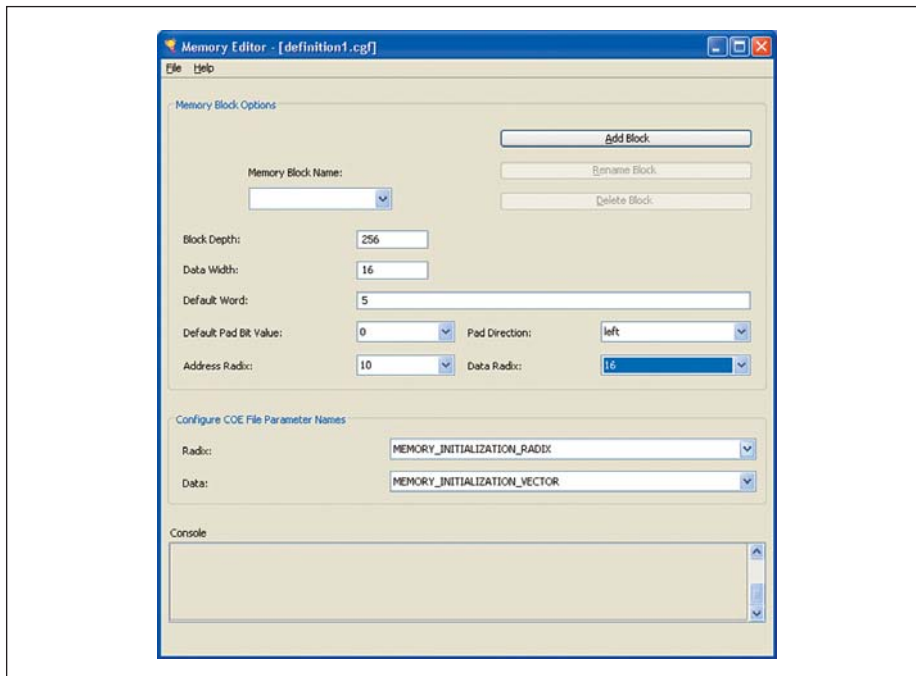


Рис. 186. Вид основного окна управления Memory Editor специализированного редактора файлов

и CSV). Вид основного окна управления с заголовком *Memory Editor* показан на рис. 186.

Структура основного окна управления *Memory Editor* включает в себя следующие элементы:

- заголовок окна;
- главное меню;
- встроенную панель определения параметров конфигурации формируемого описания содержимого запоминающих устройств (*Memory Block Options*);
- встроенную панель выбора ключевых параметров для создаваемых файлов инициализации генерируемых элементов памяти в формате COE (*Configure COE File Parameter Names*);
- встроенное окно консольных сообщений (*Console*).

В заголовке окна управления *Memory Editor* отображается название текущего файла спецификации в формате CGF (COE Generation Format), в который записывается информация о структуре данных, предназначенных для инициализации соответствующих элементов памяти, а также сведения обо всех параметрах каждого блока описания содержимого сформированных запоминающих устройств. В этот же файл заносятся данные, указанные разработчиком для каждого блока описания содержимого соответствующего элемента памяти. Формат и содержание файла конфигурации в формате CGF продемонстрированы на примере спецификации *memory_config.cgf*, текст которой выглядит следующим образом:

```
#version3.0
#memory_block_name=ram_1
#block_depth=32
#data_width=16
```

```
#default_word=01
#default_pad_bit_value=0
#pad_direction=left
#data_radix=16
#address_radix=10
#coe_radix=MEMORY_INITIALIZATION_RADIX
#coe_data=MEMORY_INITIALIZATION_VECTOR
#data=
@1
1a
1b
@4
11
@7
1c
@10
1f
@12
1e
@15
10
12
13
14
@22
15
17
@25
16
18
19
#end
#memory_block_name=ram_2
#block_depth=16
#data_width=8
#default_word=02
#default_pad_bit_value=0
#pad_direction=left
#data_radix=16
#address_radix=10
#coe_radix=MEMORY_INITIALIZATION_RADIX
#coe_data=MEMORY_INITIALIZATION_VECTOR
#data=
@1
2a
2b
@4
2c
2d
@7
2e
21
@11
2f
2e
@15
25
#end
```

В приведенном тексте файла спецификации определяются два блока инициализации содержимого запоминающих устройств *ram_1* и *ram_2* с организацией данных 32 слова \times 16 разрядов и 16 слов \times 8 разрядов. Каждый блок описания содержимого соответствующего элемента памяти начинается с ключевого слова *memory_block_name*, с помощью которого указывается название этого модуля, и завершается ключевым словом *end*. В начале каждого блока инициализации определяются его параметры с помощью соответствующих ключевых слов. Все ключевые слова в файле спецификации CGF следуют после символа #. Информационные данные, предназначенные для записи в соответствующие ячейки запоминающего устройства, которые были явно определены пользователем в процессе работы с программой *Memory Editor*, перечисляются вслед за ключевым словом *data* в форме последовательностей значений, начинающихся с указанных адресов. При этом начальные адреса последовательности ячеек памяти задаются с помощью префикса @.

Главное меню предоставляет доступ ко всем группам команд, которые используются для управления процессом создания и редактирования файлов инициализации содержимого элементов памяти, формируемых с помощью генератора параметризованных модулей CORE Generator. В основном меню программы *Memory Editor* представлено два пункта — **File** и **Help**. При выборе одного из этих пунктов на экран выводится одноименное всплывающее меню, в котором расположена соответствующая группа команд. Всплывающее меню **File** содержит команды создания, открытия и сохранения файлов спецификации в формате CGF, а также команды, управляющие процессом генерации файлов инициализации для сформированных элементов запоминающих устройств в формате COE и/или CSV. Кроме того, в эту группу включены команды импортирования данных из файлов описания содержимого запоминающих устройств, представленных в форматах CSV и MIF (Memory Initialization File).

Во всплывающем меню **Help** находятся команды, позволяющие получить информацию о текущей версии программы *Memory Editor* и открывающие доступ к документации генератора параметризованных модулей CORE Generator, включающей в себя описание интегрированного редактора файлов содержимого формируемых элементов памяти.

Встроенная панель управления параметрами конфигурации формируемого описания содержимого запоминающих устройств *Memory Block Options* предоставляет разработчику все необходимые инструменты, используемые в процессе создания блоков данных, которые применяются для инициализации соответствующих элементов памяти, и определения их организации. В этой встроенной

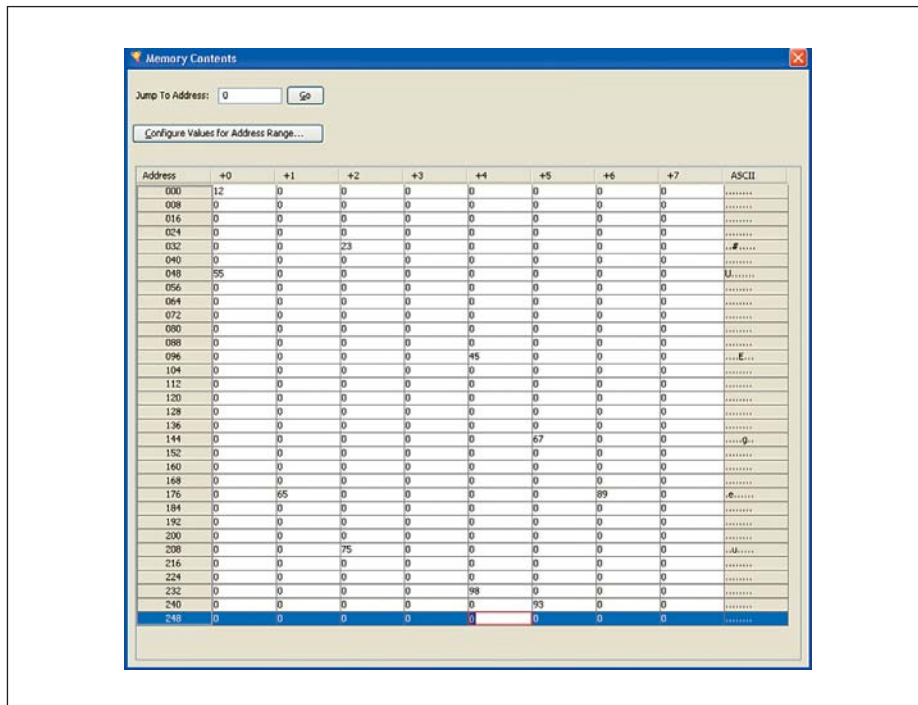


Рис. 187. Вид окна редактирования Memory Contents программы формирования файлов инициализации

панели находятся кнопки *Add Block*, *Rename Block* и *Delete Block*, предназначенные соответственно для создания описания содержимого запоминающих устройств, переименования и удаления сформированных блоков. Во встроенной панели *Memory Block Options* также расположены поле выбора *Memory Block Name* и поля редактирования *Block Depth* и *Data Width*, с помощью которых указывается название формируемого блока описания содержимого элемента памяти и его организация. Кроме того, рассматриваемая панель содержит поле редактирования *Default Word* и поля выбора *Default Pad Bit Value* и *Pad Direction*, позволяющие указать значения слов данных и его отдельных разрядов, записываемые автоматически (по умолчанию) в те разряды ячеек памяти, содержимое которых не определено пользователем явно в окне редактирования *Memory Contents*. Поля выбора *Address Radix* и *Data Radix*, также присутствующие во встроенной панели *Memory Block Options*, предназначены для указания форматов представления адресов ячеек памяти и информационных данных, которые будут использоваться при подготовке текущего блока описания содержимого запоминающего устройства.

Встроенная панель *Configure COE File Parameter Names* позволяет пользователю выбрать варианты идентификаторов ключевых параметров, которые будут использоваться в формируемых файлах инициализации генерируемых элементов памяти в формате COE. В состав этой панели входят два поля выбора — *Radix* и *Data*. С их помощью указываются идентификаторы ключевых параметров, определяющих фор-

мат представления информационных данных, и название массива значений в создаваемом файле COE.

Во встроенном окне консольных сообщений *Console* отображается информация о ходе выполнения каждой фазы процесса подготовки файлов, предназначенных для инициализации содержимого элементов памяти, которые создаются с помощью генератора параметризованных модулей CORE Generator.


Окно с заголовком *Memory Contents*, вид которого приведен на рис. 187, позволяет в наглядной форме быстро определить содержимое ячеек генерируемого запоминающего устройства.

Основную область данного окна занимает таблица, которая включает в себя десять колонок. В первом столбце этой таблицы с названием *Address* отображается последовательность базовых значений адресов ячеек запоминающего устройства. Эта последовательность начинается с нулевого адреса и включает в себя значения адресов ячеек, кратные восьми. Формат отображения адресов в колонке *Address* определяется значением параметра *Address Radix*, которое указывается в одноименном поле выбора, представленном в окне управления программы *Memory Editor* (рис. 186). Следующие восемь столбцов таблицы содержат поля редактирования значений содержимого соответствующих ячеек памяти. Адрес каждой ячейки представляет собой сумму значения базового адреса, отображаемого в соответствующей строке первой колонки таблицы, и значения смещения, указанного в первой строке выбранного столбца. В по-

следней колонке таблицы отображается содержимое ячеек, входящих в состав соответствующей строки, в виде последовательности символов в формате ASCII.

Для быстрого перехода к полю определения содержимого требуемой ячейки памяти в окне редактирования *Memory Contents* предусмотрены соответствующие инструменты. Адрес нужной ячейки запоминающего устройства указывается в поле редактирования *Jump to Address* (рис. 187). Переход к полю редактирования содержимого этой ячейки и его активизация осуществляется нажатием кнопки *Go*, которая расположена справа от поля редактирования значения адреса. В окне редактирования *Memory Contents* также расположена кнопка *Configure Values for Address Range*, которая позволяет определить значение содержимого последовательности ячеек запоминающего устройства, относящихся к указанному пользователем диапазону адресного пространства памяти.

Подготовка файлов, предназначенных для инициализации содержимого элементов запоминающих устройств, формируемых с помощью генератора параметризованных модулей CORE Generator, в среде встроенного редактора Memory Editor

Для активизации встроенного редактора файлов, используемых в процессе инициализации содержимого элементов запоминающих устройств, следует выполнить команду *Tools* главного меню основного окна средств CORE Generator, а затем в открывшемся всплывающем меню выбрать команду *Memory Editor*. С этой же целью можно воспользоваться кнопкой быстрого доступа , которая расположена на оперативной панели управления генератора параметризованных модулей.

Чтобы приступить к созданию новой спецификации содержимого элементов памяти, необходимо выполнить команду *New Memory Definition* из всплывающего меню *File* интегрированного редактора *Memory Editor*. Затем следует добавить в состав данной спе-

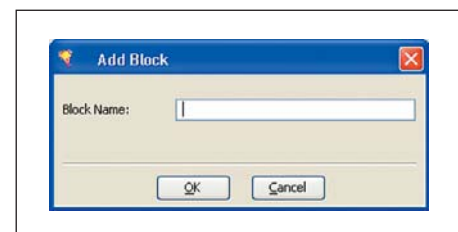


Рис. 188. Вид диалоговой панели Add Block, предназначенной для определения названия создаваемого блока инициализации содержимого запоминающего устройства

цификации новый блок описания содержимого соответствующего запоминающего устройства. Для этого нужно воспользоваться кнопкой *Add Block*, которая расположена во встроенной панели управления параметрами конфигурации формируемого описания содержимого запоминающего устройства *Memory Block Options* (рис. 186). При нажатии на эту кнопку на экран выводится диалоговая панель с заголовком *Add Block*, вид которой показан на рис. 188.

В этой диалоговой панели представлено поле редактирования *Block Name*, в котором необходимо указать название создаваемого блока данных, предназначенного для инициализации содержимого соответствующего запоминающего устройства. В составе идентификатора блока описания содержимого элемента памяти могут использоваться строчные буквы латинского алфавита (от 'a' до 'z'), цифры (от 0 до 9) и символ подчеркивания ('_'). Следует обратить внимание на то, что указываемое название должно обязательно начинаться со строчной буквы латинского алфавита. После ввода текста идентификатора блока инициализации в поле редактирования *Block Name* следует подтвердить его нажатием клавиши *OK*, расположенной в нижней части диалоговой панели *Add Block*. При этом название нового блока инициализации автоматически отобразится в поле выбора *Memory Block Name*, которое находится во встроенной панели *Memory Block Options* (рис. 186).

Далее нужно определить значения основных параметров создаваемого блока описания содержимого запоминающего устройства. В поле редактирования *Block Depth* следует указать количество слов данных в создаваемом блоке инициализации. Данный параметр может принимать любые значения в диапазоне от 1 до 1 048 576 слов. По умолчанию в поле редактирования *Block Depth* предлагается значение, равное 256. Длина (количество двоичных разрядов) слова данных в формируемом блоке задается в поле редактирования *Data Width*. По умолчанию для этого параметра используется значение, равное 16. В поле редактирования *Default Word* целесообразно указать значение слова данных, которое будет автоматически занесено во все ячейки таблицы, представленной в окне редактирования *Memory Contents* (рис. 187). По умолчанию данный параметр принимает нулевое значение.

Если значения слов данных, указываемые пользователем в таблице окна *Memory Contents*, содержат меньшее число двоичных разрядов по сравнению с длиной слов, заданной в поле редактирования *Data Width*, то эти значения будут автоматически дополнены слева или справа соответствующим количеством битов. Значение бита, которым дополняются недостающие разряды слов данных, определяется с помощью поля выбора *Default Pad Bit Value*. Выпадающий список возможных

значений этого параметра содержит два варианта: 0 и 1. По умолчанию неопределенные пользователем разряды слова данных заполняются нулевыми битами. Направление заполнения недостающих разрядов значений слов данных задается с помощью поля выбора *Pad Direction*. В выпадающем списке этого поля выбора представлено два варианта допустимых значений — *left* и *right*. По умолчанию в поле выбора *Pad Direction* предлагается вариант *left*, при использовании которого значения слов данных дополняются указанными битами слева.

Прежде чем перейти непосредственно к определению содержимого ячеек памяти, описываемой в создаваемом блоке, с помощью таблицы, расположенной в окне редактирования *Memory Contents*, рекомендуется выбрать формат представления (основание системы счисления) значений адресов и данных. Формат отображения значений адресов ячеек запоминающего устройства в этой таблице указывается в поле выбора *Address Radix*, представленном во встроенной панели *Memory Block Options* (рис. 186). Выпадающий список возможных значений этого параметра содержит четыре варианта — 2, 8, 10, 16, которые соответствуют двоичному, восьмеричному, десятичному и шестнадцатеричному форматам представления чисел. По умолчанию значения адресов ячеек в таблице, входящейся в окне редактирования *Memory Contents*, отображаются в десятичной системе счисления. Формат записи данных в ячейках этой таблицы задается с помощью поля выбора *Data Radix*, которое также расположено во встроенной панели *Memory Block Options*. Выпадающий список значений этого поля выбора включает в себя пять вариантов — 2, 8, 10 (signed), 10 (unsigned), 16, которые соответствуют указанным выше форматам представления чисел. В отличие от значений адресов, данные, записываемые в десятичной системе счисления, могут быть представлены как в знаковом формате (signed), так и формате без знака (unsigned). По умолчанию предлагается шестнадцатеричный формат записи значений данных в ячейках таблицы, находящейся в окне редактирования *Memory Contents*.

Завершает процесс определения параметров создаваемого блока данных, предназначенных для инициализации содержимого сгенерированного элемента памяти, процедура выбора вариантов идентификаторов ключевых параметров. Файлы данных в формате COE применяются не только для определения содержимого типовых оперативных и постоянных запоминающих устройств, но и для указания необходимой информации, используемой в ряде других элементов, формируемых с помощью генератора параметризованных модулей CORE Generator. В частности, примерами таких элементов являются сдвиговые регистры, реализуемые на основе распределенной памяти ПЛИС, которые ге-

нерируются на основе ядра RAM-based Shift Register. Идентификаторы ключевых параметров, используемых в файле данных в формате COE, зависят от типа элемента, для которого создается этот файл.

Для определения идентификатора параметра, устанавливающего формат представления информационных данных в формируемом файле COE, следует воспользоваться полем выбора *Radix*, которое находится во встроенной панели *Configure COE File Parameter Names* окна управления *Memory Editor* (рис. 186). В выпадающем списке данного поля выбора представлено два возможных варианта этого ключевого слова: *MEMORY_INITIALIZATION_RADIX* и *RADIX*. По умолчанию в создаваемый файл данных в формате COE записывается ключевое слово *MEMORY_INITIALIZATION_RADIX*. Требуемый вариант идентификатора параметра, который определяет название массива значений, используемых для инициализации содержимого запоминающего устройства или других элементов, создаваемых с помощью генератора параметризованных модулей CORE Generator, указывается в поле выбора *Data*. Выпадающий список этого поля выбора содержит два варианта ключевого слова: *MEMORY_INITIALIZATION_VECTOR* и *MEMDATA*. По умолчанию в поле выбора *Data* представлен вариант *MEMORY_INITIALIZATION_VECTOR*. В процессе подготовки файлов данных в формате COE, которые предназначены для автоматического определения содержимого элементов оперативных и постоянных запоминающих устройств, следует использовать идентификаторы ключевых параметров, предлагаемые по умолчанию в полях выбора *Radix* и *Data*.

Чтобы приступить к вводу значений данных, предназначенных для инициализации содержимого элементов памяти, необходимо перейти в окно редактирования *Memory Contents* (рис. 187). Затем нужно в таблице, представленной в этом окне, активизировать поле редактирования содержимого соответствующей ячейки памяти (поместив на него курсор и щелкнув левой кнопкой мыши) и с помощью клавиатуры ввести требуемое значение. Эту операцию следует повторить для всех ячеек, в которые должна быть записана информация, отличающаяся от значения, предлагаемого по умолчанию. Количество двоичных, восьмеричных, десятичных или шестнадцатеричных символов, указываемых в каждом поле редактирования, не должно превышать длины слов данных для создаваемого блока, заданной в поле редактирования *Data Width* (рис. 186). При обнаружении ошибки в указываемом слове данных это значение выделяется красным цветом. Для быстрой активизации поля редактирования значения, загружаемого в некоторую ячейку памяти, рекомендуется указать ее адрес в поле редактирования *Jump to Address* и нажать кнопку

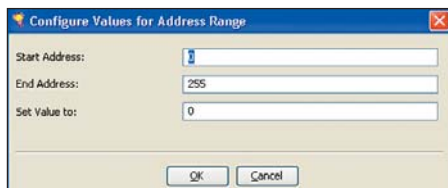


Рис. 189. Вид диалоговой панели Configure Values for Address Range

Go (рис. 187). Если в ячейки запоминающего устройства, образующие непрерывный сегмент в адресном пространстве памяти, необходимо записать одинаковые данные, то целесообразно воспользоваться кнопкой *Configure Values for Address Range*. При нажатии на эту кнопку на экране монитора отображается диалоговая панель с одноименным заголовком, вид которой показан на рис. 189.

В диалоговой панели *Configure Values for Address Range* нужно вначале поочередно опделить значения начального и конечного адресов требуемого диапазона, используя поля редактирования *Start Address* и *End Address* соответственно. Затем в поле редактирования *Set Value to* следует указать значение слова данных, которое будет загружаться в процессе инициализации в ячейки памяти заданного диапазона адресного пространства памяти. После нажатия клавиши *OK*, расположенной в нижней части диалоговой панели *Configure Values for Address Range*, это значение будет автоматически занесено в соответствующие ячейки таблицы.

Завершив процесс определения создаваемого блока инициализации, рекомендуется сохранить все указанные параметры этого блока и информационные данные в файле спецификации содержимого элементов памяти. Для этого следует выбрать команду *Save Memory Definition* из всплывающего меню *File* программы *Memory Editor*. При выполнении указанной команды на экран выводится диалоговая панель с заголовком *Save Memory Definition As...*, вид которой представлен на рис. 190.



Рис. 190. Вид диалоговой панели Save Memory Definition As...

В этой диалоговой панели в поле редактирования *Memory Definition* отображается полное название создаваемого файла спецификации, предлагаемое по умолчанию. В данном поле редактирования с помощью клавиатуры можно сразу указать название формируемого файла и его расположение на диске. Для выбора раздела, в который нужно записать создаваемый файл спецификации со-

держимого элементов памяти, целесообразно воспользоваться кнопкой *Browse*, которая находится справа от поля редактирования *Memory Definition* (рис. 190). При нажатии на эту кнопку на экране отображается стандартная диалоговая панель сохранения файла, которая позволяет быстро найти нужный каталог и задать идентификатор файла. После закрытия этой диалоговой панели, название выбранного каталога и идентификатор нового файла спецификации автоматически заносятся в поле редактирования *Memory Definition*. Процесс создания файла спецификации выполняется после нажатия клавиши *OK*, расположенной в диалоговой панели с заголовком *Save Memory Definition As...* При успешном завершении этого процесса на экране появляется информационная панель, вид которой приведен на рис. 191. Соответствующая информация также отображается в поле редактирования *Memory Definition*. Процесс создания файла спецификации выполняется после нажатия клавиши *OK*, расположенной в диалоговой панели с заголовком *Save Memory Definition As...* При успешном завершении этого процесса на экране появляется информационная панель, вид которой приведен на рис. 191. Соответствующая информация также отображается в поле редактирования *Memory Definition*. Процесс создания файла спецификации выполняется после нажатия клавиши *OK*, расположенной в диалоговой панели с заголовком *Save Memory Definition As...* При успешном завершении этого процесса на экране появляется информационная панель, вид которой приведен на рис. 191. Соответствующая информация также отображается в поле редактирования *Memory Definition*.

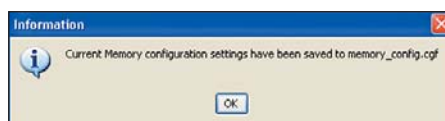


Рис. 191. Вид информационной панели, сообщающей об успешном создании (сохранении) файла спецификации содержимого элементов памяти

Если в состав создаваемой спецификации содержимого элементов памяти необходимо включить еще один или несколько блоков данных, предназначенных для инициализации соответствующих запоминающих устройств, то следует повторить всю последовательность операций, рассмотренную в настоящем разделе. При этом следует учитывать, что для каждого блока описания содержимого запоминающего устройства будет сформирован отдельный файл данных в формате COE и/или CSV. Название каждого из этих файлов состоит из имени файла спецификации и идентификатора соответствующего блока инициализации, разделяемых символом подчеркивания.

Чтобы приступить к процессу генерации файлов описания содержимого элементов памяти в требуемом формате, нужно воспользоваться командой *Generate*, которая входит в состав всплывающего меню *File* програм-

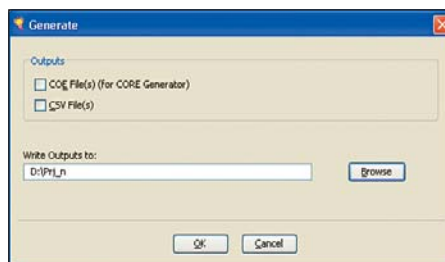


Рис. 192. Вид диалоговой панели Generate

мы *Memory Editor*. В результате выполнения этой команды на экране появляется диалоговая панель с заголовком *Generate*, вид которой изображен на рис. 192.

В открывшейся диалоговой панели следует выбрать типы формируемых файлов и место их расположения на диске. Типы генерируемых файлов указываются с помощью индикаторов состояния, расположенных во встроенной панели *Outputs* (рис. 192). Для создания файлов описания содержимого элементов памяти в формате COE необходимо перевести в состояние «Включено» индикатор *COE File(s) (for CORE Generator)*. Чтобы сформировать файл данных в формате CSV, следует установить в активное состояние индикатор *CSV File(s)*. Файлы CSV имеют текстовый формат, в котором каждое слово данных располагается на отдельной строке. Формат CSV поддерживается, в частности, средствами Microsoft® Excel™. При открытии файла данных в формате CSV, сгенерированного программой *Memory Editor*, в среде Microsoft Excel каждое слово данных заносится в отдельную ячейку первого столбца электронной таблицы. Полное название раздела (включающее путь доступа к нему), в который должны быть записаны создаваемые файлы, указывается в поле редактирования *Write Outputs to*. Для быстрого поиска требуемого каталога можно воспользоваться стандартной диалоговой панелью сохранения файла, которая открывается при нажатии кнопки *Browse*, расположенной справа от поля редактирования *Write Outputs to*.

Запуск процесса генерации файлов, используемых для описания содержимого элементов памяти в указанных форматах, осуществляется нажатием клавиши *OK*, которая находится в нижней части диалоговой панели *Generate*. При успешном завершении этого процесса на экране появляется информационная панель, в которой отображаются названия сформированных файлов. Вид этой информационной панели показан на рис. 193. Сведения о результатах выполнения процесса генерации указанных файлов данных появляются также во встроенном окне консольных сообщений *Console* окна управления *Memory Editor* (рис. 186).

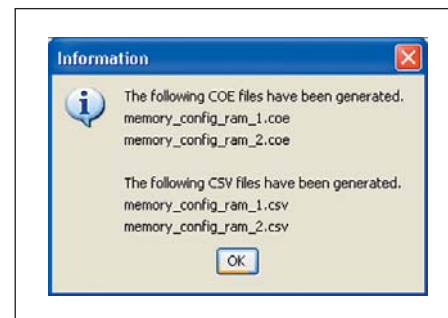


Рис. 193. Вид информационной панели, сообщающей об успешном создании файлов описания содержимого элементов памяти

Редактирование файлов, предназначенных для инициализации содержимого элементов запоминающих устройств, формируемых с помощью генератора параметризованных модулей CORE Generator, средствами Memory Editor

Чтобы приступить к редактированию ранее созданных файлов, предназначенных для инициализации содержимого элементов запоминающих устройств, следует открыть соответствующий файл спецификации в формате CGF. Для этого необходимо выполнить команду *Open Memory Definition* из всплывающего меню **File** интегрированного редактора *Memory Editor*, после чего с помощью стандартной диалоговой панели открытия файла нужно найти соответствующий раздел на диске и выбрать требуемый идентификатор спецификации содержимого элементов памяти в формате CGF. Затем нужно указать название редактируемого блока инициализации запоминающего устройства в выпадающем списке поля выбора *Memory Block Name*, которое находится во встроенной панели *Memory Block Options* (рис. 186). В результате выполненных действий в окне управления *Memory Editor* отображаются все параметры данного блока, заданные при его создании, а в окне редактирования *Memory Contents* — указанные ранее информационные значения, предназначенные для инициализации содержимого соответствующего запоминающего устройства. После внесения всех необходимых изменений следует сохранить новую редакцию файла спецификации в формате CGF и повторить процесс генерации требуемых файлов описания содержимого элементов памяти.

Если в состав текущей спецификации содержимого элементов запоминающих устройств необходимо включить новый блок данных, предназначенный для инициализации соответствующего элемента памяти, то следует нажать кнопку *Add Block*, находящуюся во встроенной панели управления *Memory Block Options*, и выполнить последовательность операций, рассмотренную в предыдущем разделе. Для переименования какого-либо блока инициализации, представленного в открытой спецификации, нужно указать его название в выпадающем списке поля выбора *Memory Block Name*, а затем вос-

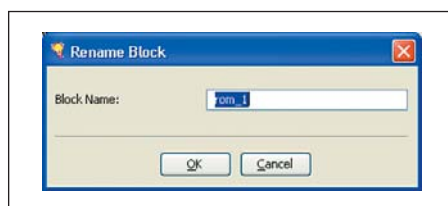


Рис. 194. Вид диалоговой панели Rename Block

пользоваться кнопкой *Rename Block* (рис. 186). При нажатии на эту кнопку на экран выводится диалоговая панель с заголовком *Rename Block*, вид которой показан на рис. 194.

В этой диалоговой панели, в поле редактирования *Block Name* отображается название выбранного блока инициализации. С помощью клавиатуры нужно ввести текст нового идентификатора и подтвердить его нажатием клавиши *OK*, расположенной в нижней части диалоговой панели с заголовком *Rename Block*. После этого новое название данного блока инициализации автоматически отображается в поле выбора *Memory Block Name*. Кроме того, во встроенном окне консольных сообщений *Console* окна управления *Memory Editor* появляется сообщение о переименовании указанного блока описания содержимого элемента запоминающего устройства.

Чтобы удалить неиспользуемый блок инициализации из текущей спецификации содержимого запоминающих устройств, следует найти и выделить его название в выпадающем списке поля выбора *Memory Block Name*, после чего нажать кнопку *Delete Block* (рис. 186). В результате выполненных действий на экране появляется диалоговая панель с одноименным заголовком, в которой содержится запрос подтверждения на удаление выбранного блока описания содержимого элемента памяти. Вид этой диалоговой панели представлен на рис. 195.

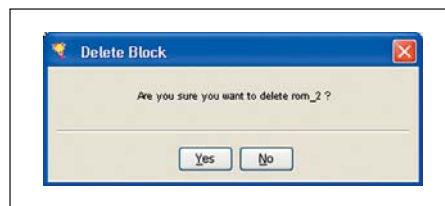


Рис. 195. Вид диалоговой панели Delete Block

При утвердительном ответе на этот запрос (нажатии клавиши *OK* в диалоговой панели *Delete Block*) указанный блок инициализации будет удален из состава спецификации, о чем информирует соответствующее сообщение во встроенном окне консольных сообщений *Console* окна управления *Memory Editor*.

Преобразование массивов данных, предназначенных для инициализации содержимого элементов запоминающих устройств, из форматов CSV и MIF в формат COE с помощью программы Memory Editor

В качестве исходных данных для формирования файлов описания содержимого запоминающих устройств в формате COE можно использовать файлы инициализации, представленные в формате MIF (Memory Initialization File), и текстовые файлы, соответствующие формату CSV. Для импорти-

рования данных, представленных в виде файлов инициализации MIF, нужно во всплывающем меню **File** программы *Memory Editor* выбрать пункт *Import*, а затем в открывшемся всплывающем меню следующего уровня активизировать команду *3rd Party MIF file*. После этого на экране появляется стандартная диалоговая панель открытия файла, в которой нужно указать название и расположение на диске требуемого файла инициализации с расширением MIF.

Если в качестве источника данных, предназначенных для инициализации содержимого элемента запоминающего устройства, используется текстовый файл формата CSV, то для загрузки его содержимого нужно после выбора пункта *Import* из всплывающего меню **File** выполнить команду *CSV file*, представленную в выпадающем меню следующего уровня. При активизации этой команды на экране монитора отображается стандартная диалоговая панель открытия файла, с помощью которой выбирается требуемый файл формата CSV. При успешном открытии указанного файла выводится диалоговая панель с заголовком *Memory Editor Settings*, вид которой приведен на рис. 196. Эта диалоговая панель позволяет определить значения всех необходимых параметров импортируемого блока данных.

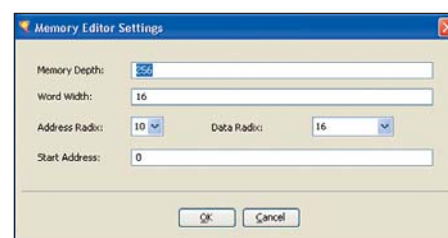


Рис. 196. Вид диалоговой панели Memory Editor Settings

Импортируемому блоку данных автоматически присваивается название, совпадающее с идентификатором выбранного файла CSV. Количество слов данных, представленных в указанном файле CSV, задается в поле редактирования *Memory Depth* диалоговой панели *Memory Editor Settings*. Максимальный объем импортируемого блока данных не должен превышать 1 048 576 слов. Длина слова данных в этом блоке инициализации указывается в поле редактирования *Word Width*. Поля выбора *Address Radix* и *Data Radix* позволяют определить форматы представления адресов и данных импортируемого блока в ячейках таблицы, расположенной в окне редактирования *Memory Contents*. Выпадающие списки возможных значений этих параметров содержат те же варианты, что и одноименные поля выбора, расположенные во встроенной панели *Memory Block Options* (рис. 186). При завершении процесса определения параметров импортируемого блока данных необходимо указать стартовый адрес,

начиная с которого будут располагаться значения в адресном пространстве соответствующего элемента памяти. Для этой цели следует воспользоваться полем редактирования *Start Address*. По умолчанию в качестве начального адреса для импортируемого блока данных предлагается нулевое значение.

После ввода требуемых значений параметров, представленных в диалоговой панели *Memory Editor Settings*, нужно подтвердить их нажатием клавиши *OK* (рис. 196). Сразу вслед за этим стартует процесс автоматического преобразования данных, результаты которого отображаются в таблице, расположенной в окне редактирования *Memory Contents*. Если данные в файле *CSV* не соответствуют длине слова, указанной в поле редактирования *Word Width*, в выбранном формате представления *Data Radix*, то соответствующие значения в таблице окна редактирования *Memory Contents* выделяются красным цветом. При этом во встроенном окне консольных сообщений *Console* окна управления *Memory Editor* появляется сообщение об ошибке преобразования значения слова данных.

Перед запуском процесса генерации файла, определяющего содержимое запоминающего устройства в формате *COE*, который соответствует импортированному блоку данных, нужно проверить и установить требуемые значения параметров *Default Word*, *Default Pad Bit Value* и *Pad Direction*. Для этой цели следует воспользоваться одноименными полями редактирования и выбора, которые расположе-

ны во встроенной панели *Memory Block Options* окна управления *Memory Editor* (рис. 186). Кроме того, при необходимости, с помощью полей выбора *Radix* и *Data*, находящихся во встроенной панели *Configure COE File Parameter Names*, следует указать идентификаторы ключевых параметров, которые будут использоваться в создаваемом файле. Прежде чем активизировать команду генерации файла инициализации в формате *COE*, рекомендуется сохранить все установленные значения параметров импортируемого блока данных в соответствующем файле спецификации. Дальнейшие операции выполняются так же, как и при создании нового модуля описания содержимого запоминающего устройства.

Заключительные замечания

На этом завершается изучение процесса разработки наиболее часто востребованных компонентов цифровых устройств на основе параметризованных модулей, поддерживаемых средствами *CORE Generator* без получения каких-либо лицензий. Следует обратить внимание на то, что в состав генератора параметризованных модулей входят также несколько групп ядер, используемых для формирования элементов специального назначения, рассмотрение которых выходит за рамки настоящей статьи.

Приведенные примеры *VHDL*-описаний компонентов, подготовленных с помощью

средств *CORE Generator*, наглядно демонстрируют возможность оперативного изменения значений параметров этих элементов в среде встроенного *HDL*-редактора САПР серии *Xilinx ISE*.

При использовании в новых проектах элементов, сформированных ранее с помощью генератора параметризованных модулей *CORE Generator*, следует учитывать регулярную смену версий систем проектирования *ISE Foundation* и *ISE WebPack*. Практически ежегодно фирма *Xilinx* выпускает новые версии этих конфигураций САПР серии *Xilinx ISE*, в состав которых включаются новые модификации средств *CORE Generator* и поддерживаемых параметризованных модулей. Кроме того, в рамках текущей версии обновление средств разработки проектов выполняется с помощью модулей *Service Pack*. В процессе установки этих модулей следует выполнить обновление текущих версий параметризованных ядер, используя для этого дополнительный архив, предоставляемый фирмой *Xilinx*.

В случае перехода от старой версии средств *CORE Generator* к новой редакции следует убедиться в том, что элементы, созданные на основе параметризованных модулей прежних версий, совместимы с новыми модификациями ядер. При отсутствии такой совместимости нужно повторно выполнить процесс генерации соответствующих элементов на базе новых версий параметризованных модулей. ■