

Продолжение. Начало в № 5'2005.

Система автоматизированного проектирования программируемых аналоговых интегральных схем AnadigmDesigner[®]2

Часть 2. Особенности разработки проектов в среде программирования ПАИС Anadigm

Александр ПОЛИЩУК
a.polishuk@prosoft.ru
Анна ПОЛИЩУК
anna@vek.ru

Мы продолжаем серию статей, посвященных новому продукту на российском рынке электронных компонентов — программируемым аналоговым интегральным схемам (ПАИС). В этом цикле мы продолжаем рассмотрение особенностей работы в среде программирования ПАИС Anadigm и расскажем о самом интересном моменте — разработке динамически переконфигурируемых устройств.

4. Конфигурирование ПАИС с помощью управляющего процессора (конфигурация хоста)

В предыдущих главах рассматривалось применение AnadigmDesigner[®]2 для создания статических конфигураций для одного или нескольких устройств. Этот процесс включал в себя разработку аналоговой схемы, сохранение соответствующего ей файла конфигурационных данных и программирования последовательного EPROM при помощи этого файла конфигурационных данных. После программирования EPROM при включении напряжения питания схемы ПАИС конфигурируется автоматически. Хотя статически конфигурируемое программируемое аналоговое устройство обеспечивает системных разработчиков новыми мощными ресурса-

ми, настоящие преимущества ПАИС в полной мере проявляются при использовании динамического переконфигурирования процессором в соответствии с изменяющимися системными требованиями.

В этом разделе рассматриваются следующие основные операции хоста: алгоритмическое динамическое конфигурирование, управляемое динамическое конфигурирование и статическое конфигурирование. Предполагается что пользователь уже хорошо знаком с процессом создания статической конфигурации в AnadigmDesigner[®]2, встроенным программным обеспечением, использующим язык Си со свойствами ПАИС, включая детали конфигурационного интерфейса.

4.1. Операции хоста

Все ПАИС компании Anadigm включают в себя настраиваемый конфигурационный

интерфейс, который легко подключается к любому подходящему микропроцессору. Устройства AN120E04 и AN121E04 более всего подходят для систем, в которых ПАИС обычно конфигурируются при включении или при перезагрузке. Конфигурационные интерфейсы устройств AN220E04, AN221E04 и AN221E02 содержат специальные функции, которые позволяют загружать реконфигурационные данные «на лету», без необходимости перезагружать устройство. На рис. 1 показано типовое подключение ПАИС к шине данных МП.

4.1.1. Основные различия между статическим и динамическим конфигурированием

AnadigmDesigner[®]2 создает несколько различных типов данных и Си-кода файлов источников для поддержки статической и динамической конфигурации.

Статическое конфигурирование

Используя основные возможности AnadigmDesigner[®]2, можно легко создавать схемы и соответствующие им наборы конфигурационных данных как для устройств AN12xE04, так и для AN22xE0x. С помощью управляющего процессора данные статической конфигурации могут быть просто переданы в ПАИС. Конечно же, в управляющем процессоре может быть сохранено несколько законченных наборов конфигурационных данных, и подключенные к нему ПАИС могут быть переконфигурированы по мере необходимости. Устройства AN120E04 и AN121E04 сначала должны быть перезагружены, и лишь затем их можно переконфигурировать. Для устройств AN220E04, AN221E04 и AN221E02 возможна как полная, так и частичная загрузка новых конфигурационных данных «на лету»,

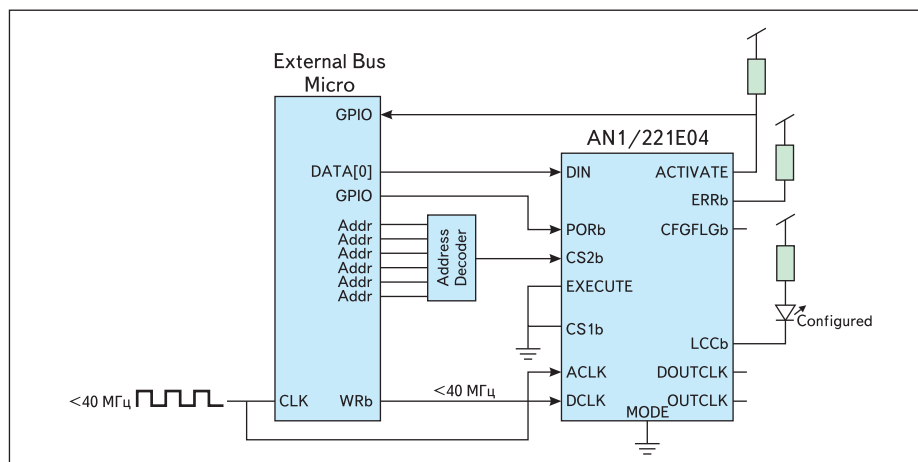


Рис. 1

без их перезагрузки. Передача управляющему процессору наборов статических конфигурационных данных хороша для приложений, в которых поведение аналоговой схемы заранее известно.

В приложениях, в которых поведение аналогового устройства должно изменяться «на лету», генератор Си-кода AnadigmDesigner®2 предоставляет управляющему процессору удобные методы создания и получения доступа к необходимому данным для конфигурирования или переконфигурирования ПАИС.

Динамическое конфигурирование

Генератор Си-кода предоставляет возможность динамического переконфигурирования устройств AN220E04, AN221E04 и AN221E02. Это означает, что измененные данные аналоговых схем или даже совсем новые аналоговые схемы могут быть загружены в ПАИС «на лету», без необходимости их перезагрузки. Новая конфигурация активируется за один такт синхронизации. Генератор Си-кода создает минимальный набор конфигурационных данных, которые необходимо загрузить для достижения требуемого изменения, что обеспечивает наиболее быстрое и компактное переконфигурирование. Их использование позволяет загрузить основной набор функций и изменять их «на лету» по мере необходимости (алгоритмическое динамическое конфигурирование) или же загрузить предварительно откомпилированные топологии схем (управляемое динамическое конфигурирование).

4.2. Алгоритмическое динамическое конфигурирование (только для AN220E04, AN221E04 и AN221E02)

При алгоритмическом динамическом конфигурировании (АДК) формируется Си-код, который используется управляющим процессором для создания или загрузки данных, необходимых для переконфигурирования ПАИС в ответ на изменившиеся требования обработки аналогового сигнала. Функции АДК поддерживают исходную первичную конфигурацию ПАИС и в дальнейшем позволяют настраивать некоторые программируемые параметры аналоговой схемы; топология схемы остается неизменной. Одним из примеров может служить настраиваемый фильтр, в котором число и типы ячеек постоянны, а частота среза, добротность и усиление можно изменять «на лету».

Для каждого КАМ существует своя функция, при помощи которой можно управлять ее программируемыми параметрами. Используя режим АДК программного обеспечения AnadigmDesigner®2, можно создать файлы Си-кода, содержащие такие функции. Этот код содержит информацию о компонентах схемы низшего уровня. Функции кода используют эту информацию при создании динамических конфигурационных данных для ПАИС в момент их вызова. Данные, полученные этими функциями для переконфигурирования устройства, добавляются в буфер

данных. Функции не переконфигурируют ПАИС напрямую. Они подготавливают данные, необходимые для изменения конфигурации устройства. Когда приходит время для пересылки данных в ПАИС, вызываются функции API, такие как GetReconfigData, для извлечения данных реконфигурации из буфера. Передача данных в ПАИС — это уже задача управляющего процессора.

Ниже будут рассмотрены типичные разработки, иллюстрирующие применение генератора Си-кода для создания АДК.

4.2.1. Разработка схемы

На рис. 2 показана схема простого аудиофильтра. Низкие частоты обрабатываются в левом канале, высокие частоты — в правом. Данная схема не является оптимальной с точки зрения компактности и эффективности, но в ней используется ряд стандартных конфигурируемых аналоговых модулей, что позволяет упростить процесс понимания алгоритма создания схем с АДК.

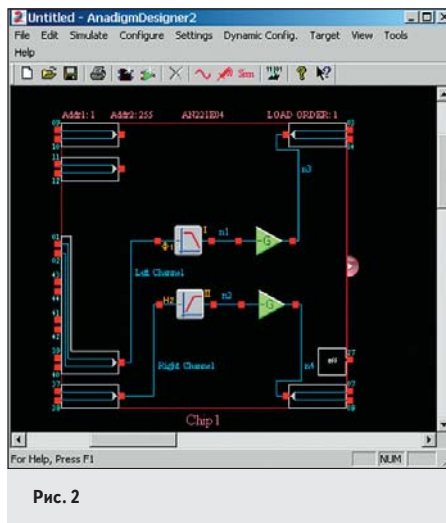


Рис. 2

4.2.2. Выбор генерируемых функций

После того, как завершена разработка схемы, необходимо определить, какие из ее частей могут нуждаться в динамическом переконфигурировании. При наличии этой информации каждая из функций, соответствующих КАМ на схеме, должна быть преобразована в Си-код.

4.2.3. Выбор функций КАМ в Си-коде

В рассматриваемом примере предположим, что необходимо динамически переконфигурировать ФВЧ в правом канале и усилитель в левом канале. По умолчанию полагаем, что функции Си-кода для всех КАМ уже определены. Для того чтобы создать компактный код, необходимо отключить формирование функций для модулей, которые не будут динамически переконфигурироваться.

Для того чтобы вызвать функцию Си-кода для какого-либо из КАМ, необходимо выполнить одно из двух действий:

1. Щелкнуть правой кнопкой мыши по КАМ и в появившемся меню выбрать подпункт «C code functions».

2. Щелкнуть правой кнопкой мыши по КАМ и выбрать подпункт «CAM Settings», а затем нажать кнопку «C code...».

Используя один из вышеприведенных методов, откройте окно «C Code Functions Window» для усилителя в правом канале. В заголовке окна появится название выбранного элемента КАМ: **GainInv_Right**. Поскольку этот модуль не будет динамически переконфигурироваться, необходимо снять «галочки» со всех предлагаемых функций (рис. 3).

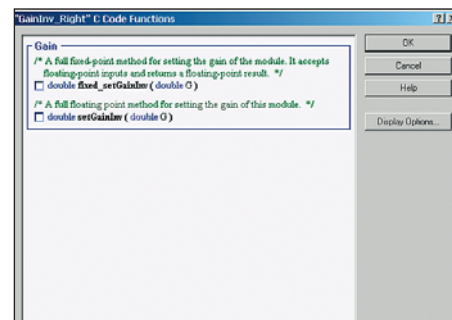


Рис. 3

Затем необходимо отключить все функции для ФНЧ левого канала. Можно воспользоваться окном «CAM C Code Functions Window» (рис. 4). Для того чтобы открыть его, выберите пункт меню *Dynamic Config* → *Algorithmic Method...* и щелкните левой кнопкой мыши по кнопке «CAM Functions...». В этом окне отображены все функции Си-кода каждого КАМ, организованные несколькими различными способами.

Если окно открыто в первый раз, в нем отображен список всех доступных типов КАМ на панели выбора (в правой части окна). В окне показано, что все функции для модуля **ANx21 StandardGainInv** отключены. Это означает, что для всех экземпляров КАМ данного типа, участвующих в разработке, функции Си-кода не будут созданы.

Обратите внимание, что мультиплексированная входная ячейка (**ADdata\ANx21_INMUX**) имеет функции в Си-коде, и они не отключены. Перед нами не стоит задача ее программирования, поэтому снимите «галочку» напротив соответствующего указателя, чтобы отключить и эти функции.

Следующий шаг — отключение функций для ФНЧ. Пользоваться окном «CAM C Code Functions Window» (рис. 4) при этом нельзя. Если будем снята «галочка» с **ANx21 StandardFilterBiquad**, то тем самым будут отключены функции Си-кода для всех КАМ данного типа, и для ФВЧ в том числе. Для того чтобы отключить создание функций Си-кода для конкретного экземпляра данного типа КАМ, щелкните на элемент с выпадающим списком «Group by», и выберите из предлагаемого перечня «Instance Name» (рис. 5). На панели выбора справа отобразится список всех экземпляров КАМ, участвующих в разработке. Обратите внимание, что для **GainInv_Right**

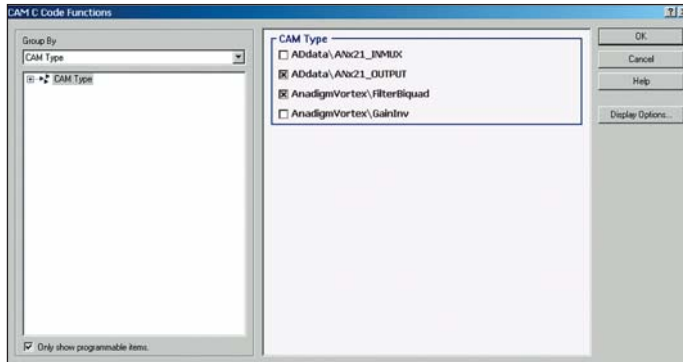


Рис. 4

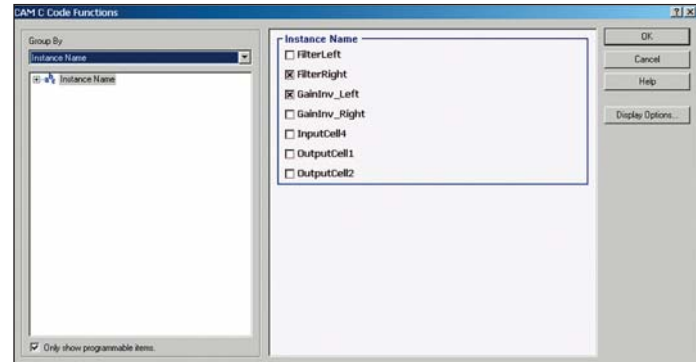


Рис. 5

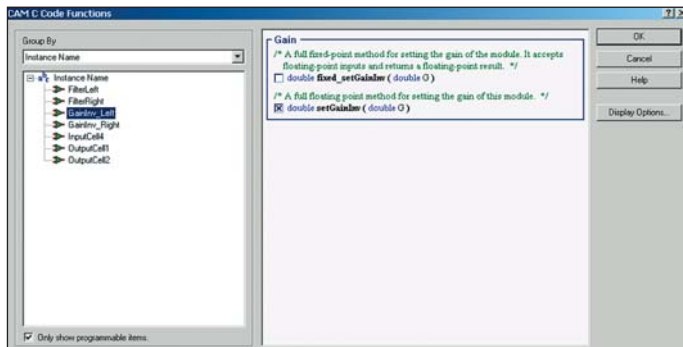


Рис. 6

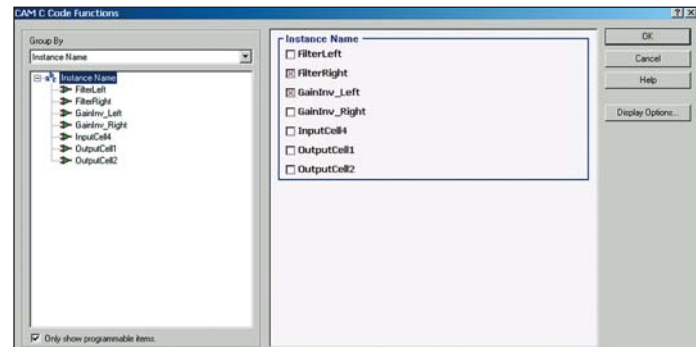


Рис. 7

и **InputCell4** все функции Си-кода уже отключены. Чтобы отключить все функции для ФНЧ левого канала, снимите «галочку» напротив строки **Filter_Left**.

Теперь будут созданы все функции Си-кода для **Filter_Right** и для **Gain_Left**. Возможно, что в процессе работы ряд функций не понадобится. Чтобы их исключить из Си-кода при создании проекта, необходимо проделать следующую процедуру.

Открыть в левой части окна список всех имеющихся экземпляров КАМ. Для этого необходимо нажать «+» напротив «Instance Name». Сначала выберите **Filter_Right**. В правой части окна на панели выбора будут отображены все возможные для данного экземпляра КАМ функции Си-кода. В рассматриваемом нами примере нет необходимости изменять значения конденсатора при помощи Си-кода, так что отключите функцию **SetBQHighPassCaps**. Затем выберите в левой части окна **Gain_Left**. Отключите функцию **fixed_setGainHold**, сняв «галочку» напротив нее на панели выбора в правой части окна (рис. 6).

Теперь у обоих модулей **Filter_Right** и **Gain_Left** имеется по одной включенной и по одной выключенной функции Си-кода. Выберите корневую ветку в левой части окна «Instance Name» путем выбора соответствующего пункта в управляющем элементе со списком «Group By». Обратите внимание, что «галочка» напротив этих двух экземпляров КАМ серого цвета. Это означает, что некоторые, но не все функции Си-кода для данных экземпляров отключены (рис. 7).

4.2.4. Создание Си-кода

В этом разделе будет рассмотрен процесс создания файлов Си-кода. Предполагается, что все описанные выше шаги уже завершены.

Установка параметров генерации:

Окно «C Code Generation Options Window» может быть открыто одним из нижеприведенных способов:

1. Выбором подпункта *C Code* → *Generation Options...* основного меню программного обеспечения AnadigmDesigner®2.
2. Выбором подпункта *C Code* → *Generate* основного меню программного обеспечения AnadigmDesigner®2, а затем нажатием кнопки *Generate Options* (рис. 8).

Закладка General

Рекомендуется не изменять настроек по умолчанию на данной закладке. Во избежание дальнейшей путаницы с именами

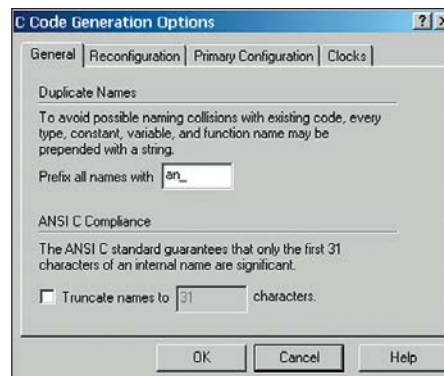


Рис. 8

в любом ранее созданном коде, все создаваемые функции и переменные имеют строковый префикс. По умолчанию, это строка «an_». Таким образом, функция **setGainHold** будет создана с именем **an_setGainHold**. AnadigmDesigner®2 поддерживает соответствие стандарту Си ANSI, обрезая все имена до 31-го символа. Ядро генератора укоротит слишком длинные имена самостоятельно и не допустит конфликта двух одинаковых имен.

Закладка Reconfiguration

При помощи закладки *Reconfiguration* можно управлять памятью, в которой будут сохранены данные для переконфигурирования, динамически создаваемые посредством Си-кода. Как только управляющей программой будет вызвана функция Си-кода, соответствующая какому-либо из КАМ, конфигурационные данные будут добавлены в буфер реконфигурационных данных. В этой закладке как раз и можно управлять размером этого буфера. Поставьте «галочку» напротив **Enable automatic growth** во избежание переполнения буфера.

Закладка Primary Configuration

В этой закладке можно выбрать устройства, которые будут сконфигурированы в самом начале посредством Си-кода. Не снимайте «галочек» напротив наименований устройств.

Закладка Clocks

В этой закладке можно выбрать устройства, которым потребуется динамическое переконфигурирование делителей частоты. Оставьте пока элементы неотмеченными.

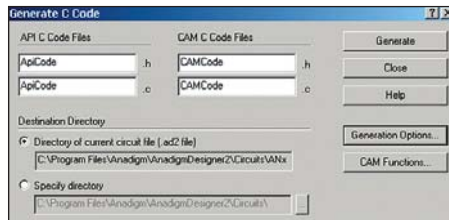


Рис. 9

Создание файлов Си-кода

Для того чтобы создать файлы Си-кода, откройте окно «Generate C Code Window». Для этого воспользуйтесь подпунктом *C Code* → *Generate* основного меню Anadigm Designer®2. В этом окне вы можете задать имена файлов, которые будут созданы, а так же указать директорию, в которой они будут размещены. Оставьте имена файлов, какими они были предложены по умолчанию, и сохраните их в той же директории, что и файл схемы (рис. 9).

Теперь все готово. Для создания файлов нажмите кнопку «Generate». Если в указанной директории уже существуют файлы с указанными именами, то всплывет окошко с вопросом о подтверждении перезаписи этих файлов. После того, как файлы будут созданы, Anadigm Designer®2 уведомит вас об успешном окончании процесса генерации файлов.

Функции созданы, и теперь можно приступить к программированию с использованием API на Си.

В рамках данной статьи мы не будем подробно рассматривать все основные функции Си-кода и примеры их использования. С этой информацией можно ознакомиться в руководстве [1].

4.3. Управляемое динамическое конфигурирование (только для AN220E04, AN221E04 и AN221E02)

Управляемое динамическое конфигурирование (УДК) — это обеспечение доступа и загрузка предварительно откомпилированных конфигурационных данных управляющим процессором во все подключенные к нему ПАИС. В отличие от АДК, в процессе управляемого динамического конфигурирования для управляющего процессора не генерируется никакого Си-кода, создающего конфигурационные данные ПАИС. Вместо этого в процессе УДК создаются наборы предварительно откомпилированных данных и всего две функции Си-кода, вызываемые для получения доступа к этим данным и для компиляции в управляющей программе.

Если используется алгоритмическое динамическое конфигурирование, то могут быть настроены только параметры КАМ. При использовании же УДК может быть изменена топология схемы. Обновление схемы может служить для настройки некоторого параметра КАМ или же для полного изменения ее содержания и внутренних связей.

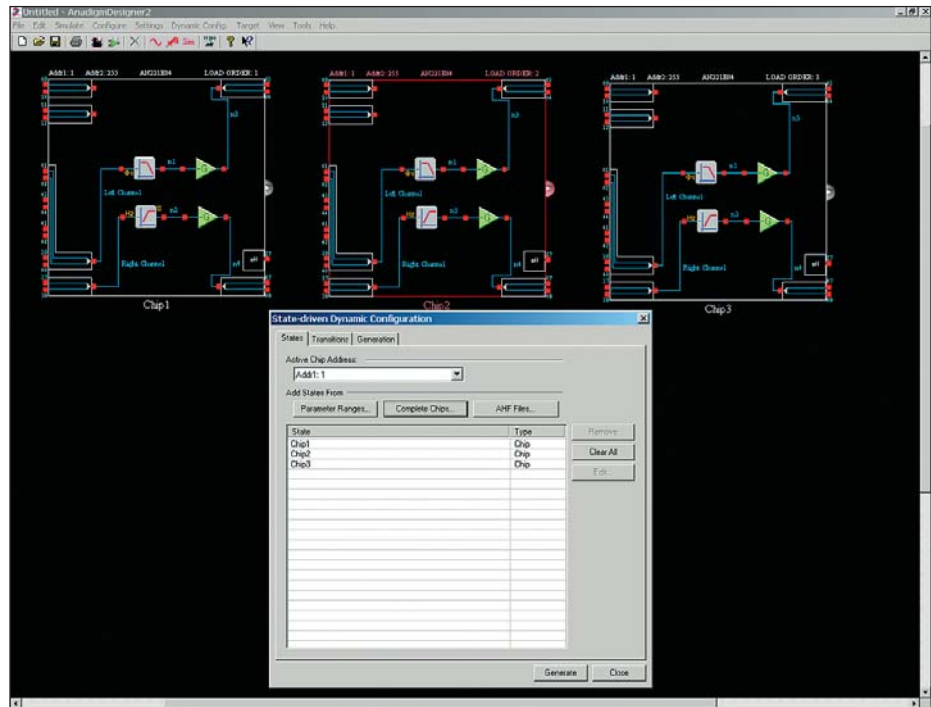


Рис. 10

По выбору подпункта меню *Dynamic Config.* → *State-driven method...* открывается окно выбора. В это окно добавляются разработки, конфигурационные данные которых необходимо откомпилировать. При помощи элемента управления «Complete Chips...» разработки могут быть импортированы из текущей сессии посредством элемента управления «Transition». Разработки также можно импортировать при помощи «AHF Files...» в виде существующего конфигурационного файла с расширением .AHF (рис. 10).

Имя функции при алгоритмическом динамическом конфигурировании присваивается в соответствии с наименованием ПАИС (например, Chip2). В УДК группировка разработок осуществляется по идентификационному номеру устройства ID (например, ID1). Должна иметься как минимум одна версия полного набора данных для первичной конфигурации для каждой ПАИС (в каждом из DeviceID). Для всех остальных устройств с одинаковыми ID будут созданы различные данные. Полученные в результате откомпилированные файлы данных, включающие в себя несколько вариантов структур ПАИС, получаются достаточно компактными.

При нажатии левой кнопки мыши кнопки Generate открывается диалоговое окно для ввода значений разнообразных параметров, связанных с созданием Си-кода и файла Raw Data для УДК.

4.3.1. Закладка Generation

Аналогично рассмотренному выше диалоговому окну создания Си-кода «Generate C Code Window» (рис. 9), на закладке «C Code» также необходимо указать имена файлов и место их расположения (рис. 11).

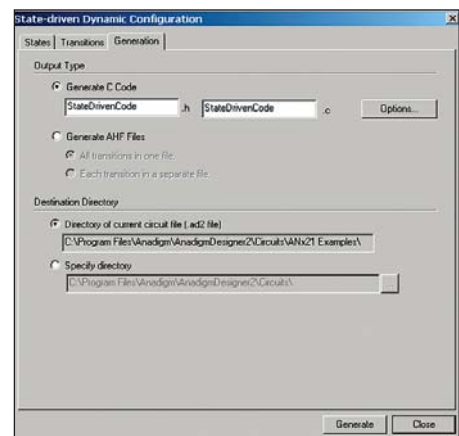


Рис. 11

В файле с расширением .c будут содержаться массивы констант для первичной конфигурации и данные для переконфигурирования (данные различий), необходимые для всех выбранных устройств, а также функции для осуществления доступа к этим данным.

В файле с расширением .h содержатся прототипы функций в виде совокупности их описания и определения типов. Этот файл для облегчения понимания снабжен подробными комментариями.

Управляющий процессор вызывает функции для того, чтобы получить доступ к первичной конфигурации, а в последующем к данным переконфигурирования (данным различий).

Для совместимости с версиями программного обеспечения управляющего процессора на каком-либо другом языке (например,

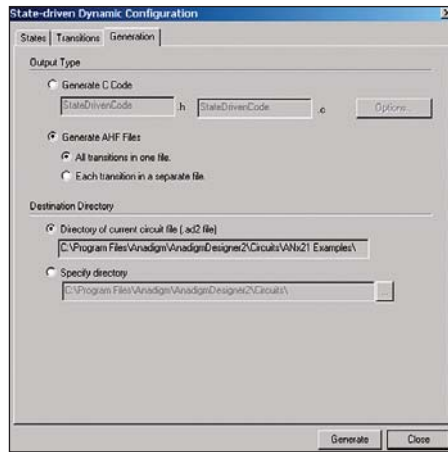


Рис. 12

на ассемблере), AnadigmDesigner@2 создает файлы данных, содержащие те же наборы данных для первичного конфигурирования и для последующего переконфигурирования (данные различий) в шестнадцатеричном формате ASCII (AHF). Данные могут быть выгружены в один файл (на ПАИС с соответствующим ID) или же в отдельные файлы (рис. 12).

В случае выгрузки данных в один файл создается .txt файл, содержащий AHF-данные, с именем StateDrivenDevice nnn .txt, где nnn — это номер ID (ID1). В файле содержатся только поля комментариев для разграничения наборов данных и данные в формате AHF.

В случае же, когда выбрана опция «Each transition in a separate file», создается поддиректория с именем StateDrivenDevice nnn . Внутри этой директории создаются отдельные .txt файлы данных в формате AHF, не содержащие каких-либо комментариев и включающие в себя либо данные для первичного конфигурирования, либо данные для переконфигурирования (данные различий). Имена файлам даются в соответствии с их содержанием. Например, файл (Primary) chip1.txt содержит данные первичной конфигурации для устройства с именем chip1.

4.4. Статическое конфигурирование (для всех типов устройств)

Статическое конфигурирование — это процесс передачи наборов данных первичной конфигурации управляющим процессором в ПАИС после их перезагрузки. Статическое конфигурирование с помощью внешнего управляющего процессора целесообразно лишь для случаев, когда ПАИС нуждаются в конфи-

гурировании только при первом включении питания или после перезагрузки системы.

Среда AnadigmDesigner@2 поддерживает различные схемы статического конфигурирования путем создания трех различных форм данных. Две из них (Си-код и Raw Data) были рассмотрены выше для УДК. В случае первой перезагрузки ПАИС обычно используется только часть созданных этим способом данных — часть первичной конфигурации.

Третья форма, создаваемая AnadigmDesigner@2 — это файл данных конфигурации. Все эти различные форматы файлов данных хорошо подходят для статического конфигурирования управляющим процессором в случае, когда используется программатор PROM. На самом деле, формат Raw Data, используемый при УДК, идентичен формату .AHF файла конфигурационных данных, рассмотренному в разделе «Шестнадцатеричный ASCII формат файлов» в предыдущей статье цикла. ■

Продолжение следует.

Литература

1. AnadigmDesigner@2. User Manual. Anadigm, Inc., 2004