НЕСЛОЖНОЕ ПРОЕКТИРОВАНИЕ СЛОЖНЫХ ВСТРАИВАЕМЫХ СИСТЕМ

МАРИО ХАЛАФ (MARIO KHALAF), научный руководитель, Altera Corporation
АДЖАЙ ДЖАГТИАНИ (AJAY JAGTIANI), менеджер по маркетингу, Altera Corporation

В статье рассматривается однокристальная архитектура, совмещающая преимущества аппаратных процессоров и ПЛИС. Приведены практические примеры реконфигурации областей ПЛИС с помощью аппаратно встроенного процессора. Статья представляет собой перевод [1].

Контроль над конфигурацией ПЛИС, как правило, осуществляется встроенным процессором, который связан с запоминающим устройством, где хранятся данные конфигурации.

Механизмы конфигурации зависят от конкретной ПЛИС. Они требуют определенных соединений на кристалле и соблюдения соответствующих правил. Чтобы оценить размер конфигурационного файла для хранения потоков данных ПЛИС, в кристалл устанавливается флэш-память. Предлагаемые в статье архитектура устройства и метод программирования облегчают решение этой задачи и расширяют возможности процессора.

ОПИСАНИЕ АРХИТЕКТУРЫ

На рисунке 1 показана архитектура ПЛИС со встроенным аппаратным процессором, у которого имеются порты Ethernet и DDR, тогда как с ПЛИС процессор соединяется через высокоскоростную шину. Кроме того, этот процессор связан с блоком управления конфигурацией ПЛИС. Данный блок позволяет реализовать все схемы конфигурации ПЛИС, управляя последовательностью ее запуска. Кристалл разделен на две области таким образом, что все аппаратные периферийные устройства, подключенные к ПЛИС, а также процессор могут функционировать независимо от ПЛИС. Например, процессор включен, пока ПЛИС конфигурируется, реконфигурируется или управляет приложением в пользовательском режиме.

Кроме того, ПЛИС оснащена двумя основными функциями: возможностью конфигурирования (и повторного конфигурирования) при включенном процессоре; возможностью частичного изменения конфигурации при работающей ПЛИС.

Как правило, эта система построена как встраиваемое двухкристальное решение. Преимущества однокристальной системы над двухкристальной хорошо известны, особенно если два кристалла могут работать независимо. Процессор, как правило, управляет встраиваемой операционной системой (например, VxWorks). Каждое из внешних аппаратных устройств имеет собственный драйвер, работающий под управлением встроенной операционной системы. Мы предлагаем два дополнительных драйвера устройств, которые обеспечат управление: портом, соединяющим процессор с блоком управления ПЛИС, и функцией полной конфигурации ПЛИС; портом, соединяющим процессор с блоком управления ПЛИС, и функцией частичной реконфигурации ПЛИС.

Рис. 1. **Однокристальное устройство с процессором и ПЛИС**

Их назначение – отображать конфигурацию ПЛИС в интерфейсах прикладного программирования (API) на процессоре. Поток данных конфигурации может храниться вне кристалла, если у процессора при необходимости имеется доступ к любому из внешних аппаратных устройств. Из рисунка 1 видно, что процессор получает доступ к потоку данных с любого устройства, подключенного к интернету через порт Ethernet.

ПРИМЕНЕНИЕ ПЛИС

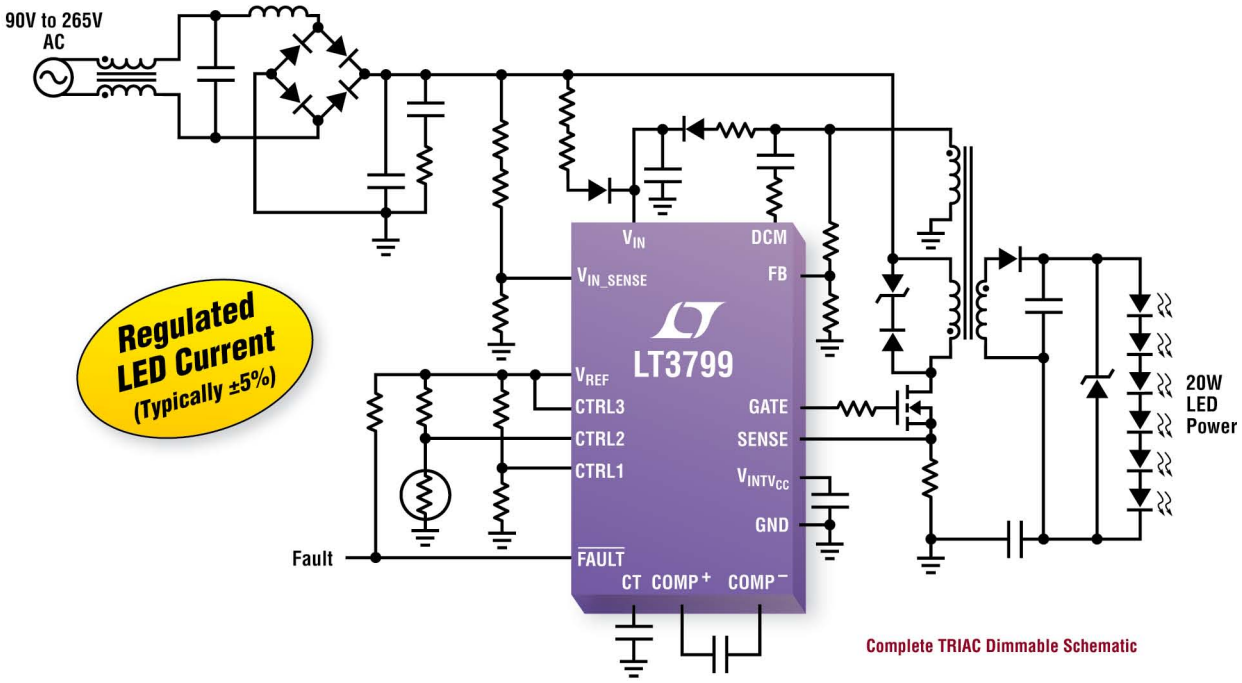
Конфигурацию ПЛИС можно полностью или частично изменить, пока работает процессор (см. рис. 2). При полной реконфигурации образ ПЛИС меняется целиком, и задаются новые области для последующих частичных реконфигураций.

ПЛИС используются во встраиваемых системах для многих специфических нужд. К наиболее распространенным применениям относятся аппаратное ускорение для реализации ресурсоемких алгоритмов (см. рис. 3) и расширения процессорной периферии (см. рис. 4).

Для максимального повышения коэффициента использования ПЛИС в обоих приведенных случаях используется частичная реконфигурация. Например, в случае, изображенном на рисунке 4, протокол может переключаться с eSATA на Fibre Channel при работе канала SDI. Это достигается путем частичной реконфигурации программного обеспечения на уровне протокола. В случае использования сопроцессора алгоритм переключается на него для перераспределения нагрузки. В том случае, если речь идет о многопоточной или многопроцессорной системе, для ускорения обработки операций можно реконфигурировать ПЛИС таким образом, чтобы каждый из потоков использовал по одной секции ПЛИС.

Конфигурация ПЛИС (частичная или полная) отображается во встроенной операционной системе в виде драйверов устройств. Дополнительный API-интерфейс записывается во встроенной ОС для управления динамическим переключением упомянутой функции, благо-

Изолированная схема стабилизации тока светодиодов с активным корректором мощности.

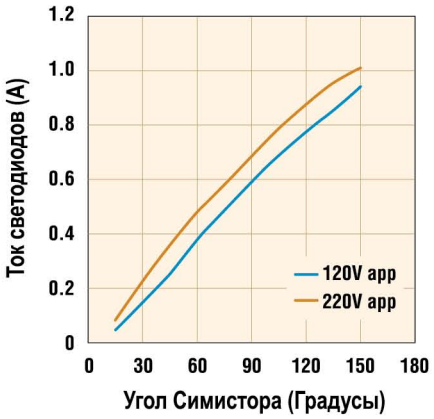


Complete TRIAC Dimmable Schematic

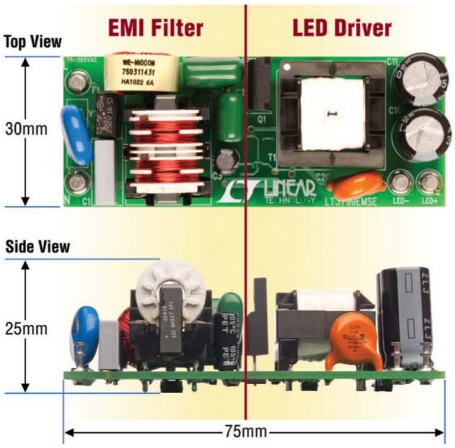
Драйвер Светодиодов с управлением яркости через симистор не требует оптопары

Наш контроллер LT3799 для изолированной схемы управления светодиодами с активным управлением коэффициента мощности (PFC) специально разработан для питания светодиодов в широком диапазоне напряжений от 24В до 480В+. Решение на этом контроллере идеально подходит для применений использующих светодиоды общей мощностью от 4Ватт до 100Ватт и совместимо со стандартным сетевым регулятором яркости выполненного на схеме с симистором (TRIAC). LT3799 использует уникальную архитектуру с управлением по току, обеспечивая хорошую стабилизацию тока во вторичной цепи, без использования оптопар и при этом, представляя точность стабилизации тока на светодиодах ±5%. Такая схема дает очень малое количество шумов и обеспечивает КПД более 90%. Защита от короткого замыкания и размыкания в цепи светодиодов гарантирует высокую надежность, а простое и компактное решение делает возможным применение контроллера в широком спектре применений.

Ток светодиодов по отношению к углу Симистора (TRIAC)



Демо плата с LT3799 (25 Ватт)



Получить информацию и купить можно

www.linear.com/product/LT3799

Free Power Management for LEDs Brochure

www.linear.com/indsolutions

LT, LTC, LTM, Linear Technology and the Linear logo are registered trademarks of Linear Technology Corporation. All other trademarks are the property of their respective owners.

РЕКЛАМА

Linear Technology AB (Finland) +358-46-712 21 71
Linear Technology AB (Sweden) +46-8-623 16 00
nordicsales@linear.com

LINEAR TECHNOLOGY

Официальные Дистрибьюторы

Гамма	+7-813-783 15 09
Петербургская Электронная Компания	+7-812-448 87 77
Arrow Electronics Russia	+7-495-626 55 97

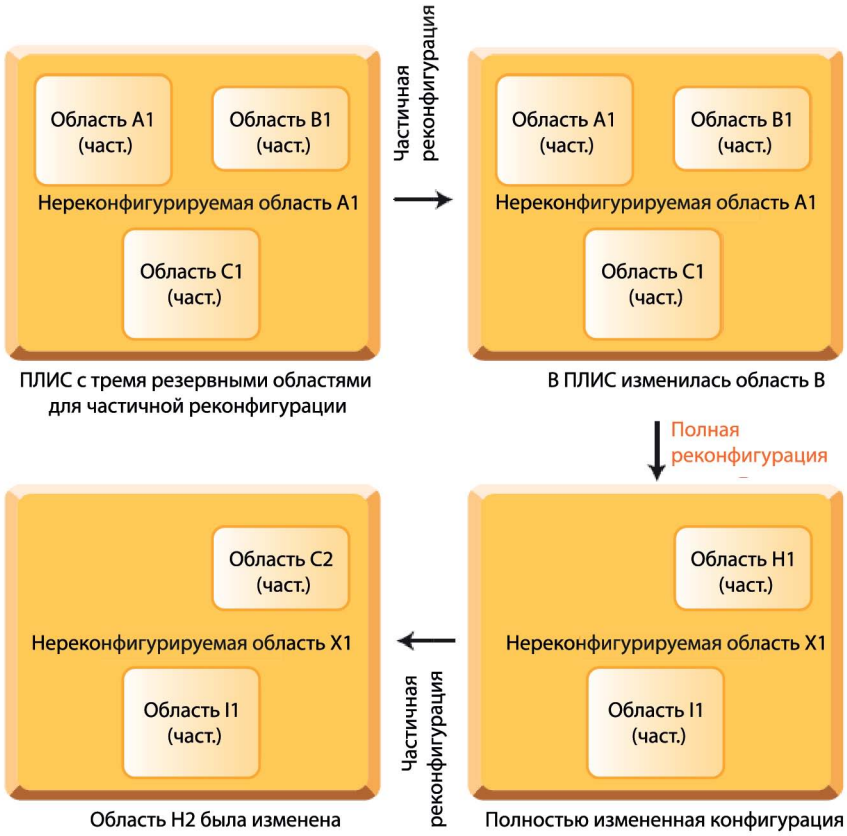


Рис. 2. **Возможности реконфигурации ПЛИС**

Рис. 3. **ПЛИС как сопроцессор для выполнения ресурсоемких алгоритмов**

Рис. 4. **ПЛИС, используемая для расширения системы портов ввода-вывода**

даря чему аппаратная реализация становится изолированной. Преимущества этой общей платформы и программной реализации следующие.

– Разработчик прикладной системы не занимается аппаратной реализацией. Программный код не учитывает динамическую реконфигурацию ПЛИС при расширении периферии. Таким образом, оборудование можно изменить, не меняя кода пользовательского приложения при минимальных изменениях в драйверах устройств.
– Конфигурационные файлы для ПЛИС могут храниться вне кристалла, что обеспечивает соблюдение мер безопасности. При необходимости процессор использует специальные методы дешифровки файла конфигурации.

Одной из важных функций этой платформы является предварительный выбор типов периферийных устройств для их размещения на ПЛИС или алгоритмов ускорения. Это значит, что в системе не выполняется динамическая компиляция кода Verilog/VHDL. Предполагается, что все образы ПЛИС (полные и частичные) компилируются заранее. Хотя задержка при загрузке и выгрузке конфигурационных файлов в ПЛИС серьезно отражается на ее работе, предполагается, что эта операция выполняется нечасто во время работы системы.

1. Экономия места на ПЛИС путем частичной реконфигурации известных протоколов или алгоритмов ускорения в установленных областях ПЛИС.
2. Отображение конфигурационной функции и преобразованных областей в драйверах устройств и API-интерфейсах встроенной ОС. ПЛИС рассматривается как оборудование с динамическими возможностями, которое использует специализированную ОС для выбора аппаратных функций из большой библиотеки. Эти функции схожим образом загружаются в программные DLL. Одним из недостатков ПЛИС является сравнительно большое время компиляции с помощью программного обеспечения САПР. Каждая из этих реализаций в ПЛИС должна быть предварительно скомпилирована с использованием программного обеспечения ПЛИС. Кроме того, необходима система для управления процессом сложной разработки аппаратного обеспечения для описанной архитектуры.

ТРЕБОВАНИЯ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ

Для реализации всех преимуществ предложенной архитектуры программное обеспечение САПР должно иметь

два необходимых компонента: средства описания программно-аппаратной системы и возможность пошаговой повторной компиляции областей ПЛИС для частичной реконфигурации.

Требуется инструмент, который позволил бы выделять области для частичной реконфигурации и управлять временем компиляции. Кроме того, необходимо средство описания всей системы (СнК) на устройстве, которое соединяет периферию с процессором. С помощью этих двух инструментов задача разработки упрощается, а система наделяется требуемыми функциями.

Частичная реконфигурация устройства должна выполняться согласно принятой методологии, в соответствии с которой сначала определяются блоки с частичной реконфигурацией и нереконфигурируемые блоки. При этом разработчику необходимо придерживаться следующих рекомендаций:

- создать указанные блоки по правилам синтеза и соблюдения иерархии;
- определить блоки частичной реконфигурации и назначить им участок топологии с помощью метода пошаговой компиляции и ограничений на отведенную область;
- убедиться в том, что определения портов и иерархических границ всех блоков с частичной реконфигурацией и их переменные не изменяются. В противном случае придется изменить конфигурацию всего устройства;
- убедиться в том, что сигналы сброса между блоками с частичной реконфигурацией и нереконфигурируемыми блоками совместно не используются;

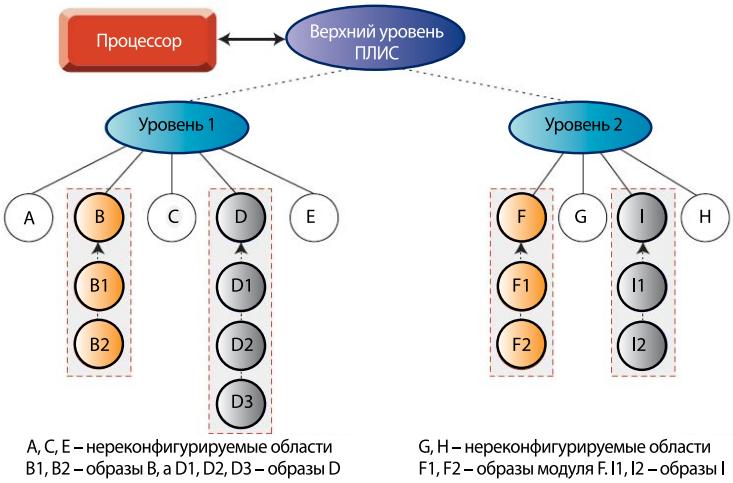


Рис. 5. **Высокоуровневое описание системы**

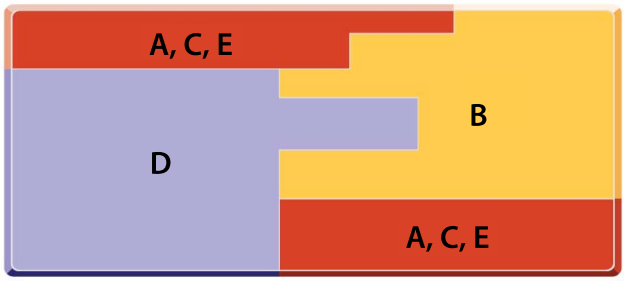


Рис. 6. **План построения Уровня 1**

- добавить сигналы подтверждения связи в блоки с частичной реконфигурацией, если в нереконфигурируемых блоках имеется логическая схема, срабатывание которой зависит от состояния блоков частичной реконфигурации.

Предлагаемый поток проектирования ПЛИС основан на принципе разделения: после резервирования неко-

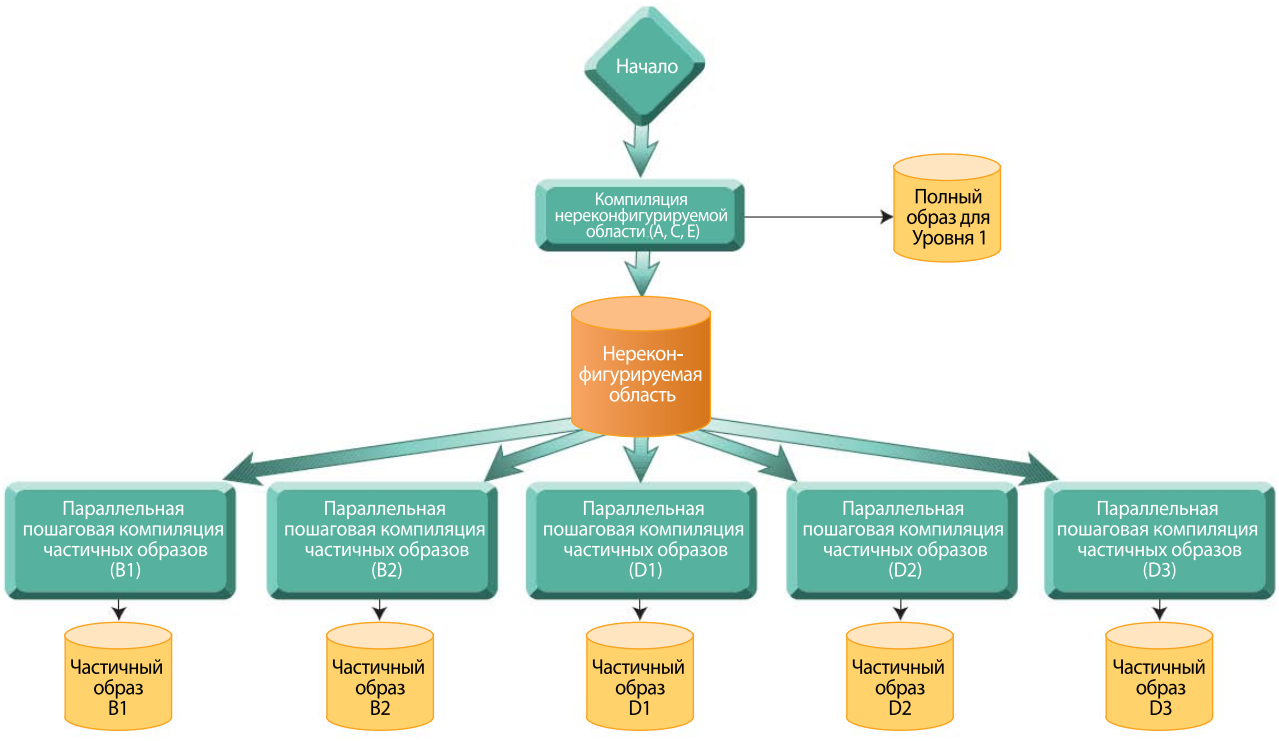


Рис. 7. **Совместное проектирование системы и ПЛИС**

торой области она и устройство могут подвергнуться реконфигурации. При этом оставшаяся часть системы продолжит работу. На рисунке 5 представлено высокоуровневое описание системы для предлагаемой архитектуры.

ПРОЕКТИРОВАНИЕ ЧАСТИ ПЛИС

На рисунке 5 представлены два образа ПЛИС верхнего уровня. Образ «Уровень 1» имеет нереконфигурируемую область, состоящую из иерархических модулей A, C, E и двух модулей B и D. Модуль B имеет две реализации – B1 и B2. Аналогично, у D имеются три реализации. Уровень 2 имеет нереконфигурируемую область, состоящую из иерархий G и H. У F и I – по две реализации.

Каждый образ ПЛИС связывается с процессором по шинной архитектуре. Аналогично, у всех реализаций имеется общий интерфейс со нереконфигурируемой областью. Каждая реализация назначается конкретному образу, загруженному в ПЛИС. В рассматриваемом примере с высокоуровневым описанием системы процесс компиляции генерирует два полных образа ПЛИС. Один полный образ состоит из пяти частичных образов (B1, B2, D1, D2, D3), другой — из четырех.

Для эффективной компиляции всех образов ПЛИС необходимо, чтобы программное обеспечение САПР позволяло реализовать две концепции построения ПЛИС: разбиение на разделы и создание закрытой области. Все изменения в проекте ограничиваются для последующей компиляции рамками этих разделов. Заблокированная область ПЛИС представляет собой физическую часть ПЛИС, зарезервированную для конкретной реализации, которая назначается при разработке общей топологии кристалла. Программное обеспечение САПР компилирует нереконфигурируемую область, после чего пошагово компилируется каждая реализация модуля для генерации всех образов. На рисунке 6 представлен поуровневый план построения Уровня 1 для системы на рисунке 5. В данном примере B и D являются закрытыми областями ПЛИС.

На рисунке 7 схематично показан процесс компиляции части Уровня 1 для описанной выше схемы. Необходимо, чтобы все образы были скомпилированы параллельно и пошагово на нескольких машинах. Этот способ проектирования позволяет заблокировать нереконфигурируемые области и целиком сосредоточиться на частичной области и соответствующих образах. В результате не только осуществляется возможность использования преимуществ пошагового проектирования, но и уменьшается время компиляции. При этом производительность системы

Рис. 8. **Пример системы с несколькими реализациями ПЛИС**

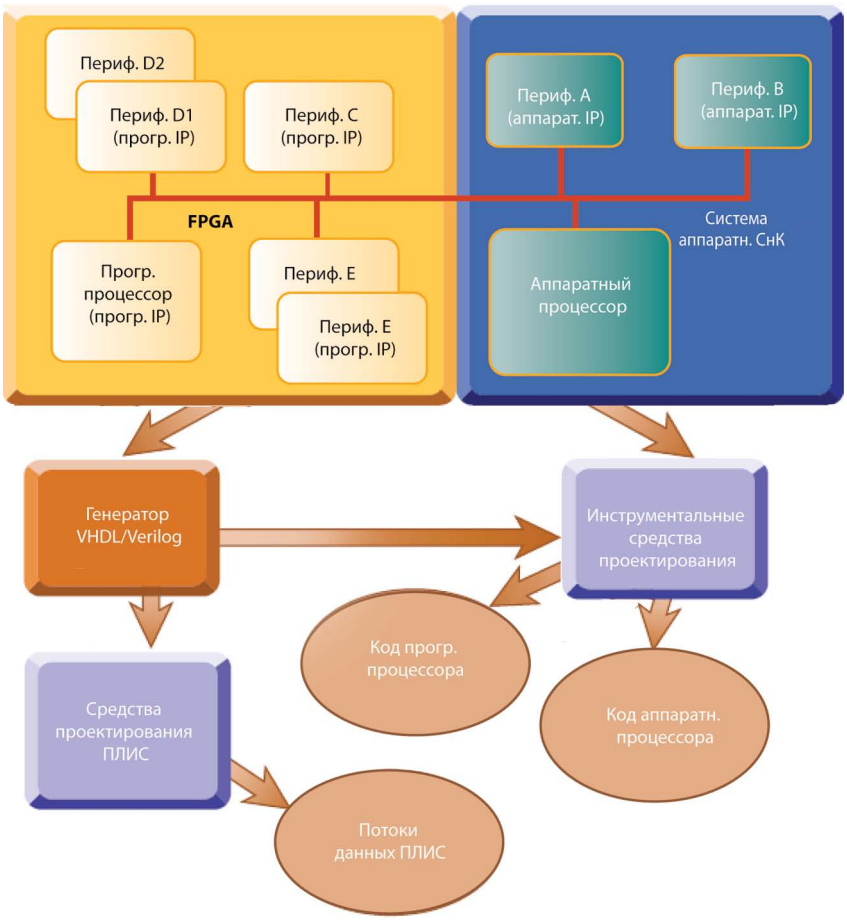


Рис. 9. **Этапы проектирования системы с помощью трех типов инструментов** (системных, ПЛИС и программных)

сохраняется даже в случае частичной реконфигурации, т.к. программируется лишь небольшой участок кристалла.

Инструмент проектирования предложенной архитектуры и процессора соединяет периферийные устройства с процессорами, назначает шины адреса и данных, а также создает систему с ее описанием на языке HDL, которое затем транслируется в устройство с помощью инструмента трассировки. При этом допускается проектирование заказных компонентов или образов для каждого из частичных блоков. На рисунке 8 представлен пример системы с двумя реализациями верхнего уровня ПЛИС.

При проектировании всей системы рекомендуется использовать такой инструмент, который позволяет одновременно моделировать аппаратную и программную (ПЛИС) части. Части ПЛИС и СнК должны совместно использовать одну систему распределения памяти, прерывания и адресное пространство. Инструмент проектирования системы автоматически интегрирует арбитражную шину и синхронизирует ее, обеспечивая единую платформу разработки ПО.

На рисунке 9 представлены этапы проектирования с помощью набора инструментальных средств. Вслед за моделированием системы выполняется разработка аппаратного и программного процессора с помощью набора программных инструментов. Часть ПЛИС компилируется с использованием генератора системы, который создает HDL-код для шины и периферий-

ных IP-блоков. Затем САПР компилирует HDL в несколько потоков данных – по одному для каждой частичной реконфигурации системы.

По завершению генерации HDL-кода он компилируется для трассировки ПЛИС. Поскольку в ходе проектирования системы HDL-код генерируется пошагово, всегда имеется возможность создать образы Уровней 1 и 2, а также добавить в систему «изюминку». На Уровнях 1 и 2 создаются частичные образы (см. рис. 9 – периф. D1, D2, периф. E1, E2). Средство трассировки определяет, изменилось ли положение Уровней 1 и 2, компилирует изменившуюся часть и генерирует соответствующий файл программирования. Аналогично обнаруживаются изменения для D1, D2, E1 и E2.

Таким образом, генерация файла программирования происходит следующим образом.

А. 1. Система подключается с помощью служебной программы. 2. Для этой служебной программы генерируется код HDL. 3. С помощью средства проектирования ПЛИС создается код верхнего уровня HDL для всего проекта. 4. Назначаются разделы для частичных блоков и их размещение в ПЛИС. 5. Компилируется вся схема – базовая конфигурация ПЛИС. 6. Генерируется файл программирования для базовой конфигурации. 7. Программируется устройство.

Б. Осуществляется генерация частичных образов: 1. В служебной программе повторно генерируется новый образ для Уровней 1 или 2 (и для D1, D2, E1, E2). 2. Генерируется код HDL для новых компонентов. 3. Полученный код компилируется для трассировки ПЛИС. 4. Остальная часть системы использует последнюю компиляцию схемы. 5. Для образов генерируются частичные файлы программирования. 6. Устройство программируется с частичным файлом реконфигурации.

ПРЕИМУЩЕСТВА

Таким образом, мы рассмотрели однокристальную архитектуру, совмещающую преимущества аппаратных процессоров и ПЛИС.

- Частичная реконфигурация позволяет сэкономить площадь, занимаемую периферийными устройствами (или ядрами), которые не функционируют одновременно.
- Потоки данных ПЛИС можно загружать удаленно, что позволяет сэкономить место, отводимое на флэш-память, и обеспечивает безопасное использование информации, т.к. она не хранится на кристалле.
- Ядро с малым потреблением загружается на ПЛИС без перезагрузки встроенной ОС для реализации того или иного режима обработки.
- В ПЛИС можно добавить ядро для отладки, чтобы контролировать трафик данных, пока работают все другие периферийные устройства.
- В ПЛИС также добавляются различные алгоритмы ускорения, используемые динамически как сопроцессор.
- Система имеет возможность обновляться, т.к. функционирование всех ПЛИС смоделировано в виде драйверов программных устройств, включая конфигурацию.
- Один инструментальный набор для управления всей системой.

ВЫВОДЫ

Мы рассмотрели аппаратную архитектуру с набором инструментальных средств, которая совмещает преимущества аппаратного процессора и ПЛИС во встраиваемой системе. Многие современные встраиваемые системы используют несколько кристаллов по одному сценарию. Добавление в ПЛИС возможности реконфигурации (полной и частичной) непосредственно из процессора без перезагрузки ОС и приостановки работы процессора повышает гибкость системы. Кроме того, применение одного набора инструментальных средств позволяет упростить проектирование сложных встраиваемых систем.

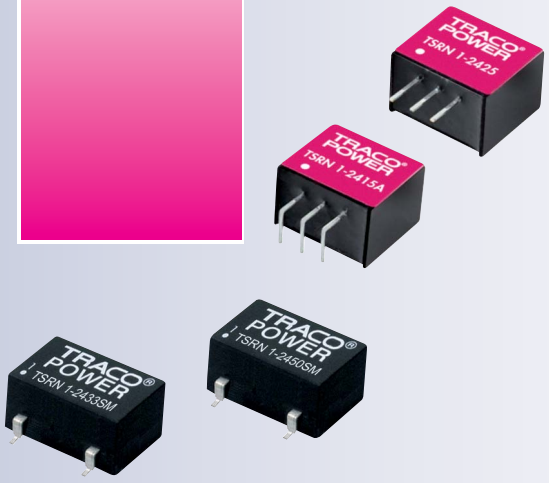
ЛИТЕРАТУРА

1. Mario Khalaf and Ajay Jagtiani. Making Hardware More Like Software//www.eetimes.com.

TRACO POWER

Неизолированные понижающие импульсные стабилизаторы напряжения

Серия TSRN-1 – высокоэффективная альтернатива линейным стабилизаторам



- Выходной ток до 1 А
- Диапазон входных напряжений 4,5…36 В DC
- КПД до 95%
- Компактный SMD или SIP корпус
- Диапазон рабочих температур -40…+85°C
- Не требуют радиатора
- Положительная или отрицательная полярность выходного напряжения
- Защита от короткого замыкания и перегрева
- Высокая стабильность выходного напряжения

www.tracopower.com

Аргуссофт (495) 660-28-55 www.argussoft.ru
ПетроИнТрейд (812) 324-63-50 www.petrointrade.ru
СЭА Электроникс (495) 228-32-82 www.searu.com

РЕКЛАМА

GEOSTAR NAVIGATION

ГЛОНАСС•GPS•COMPASS•GALILEO•SBAS

Лауреат премии «Живая электроника России» 2011

Мультисистемные навигационные модули ГеоС

ГеоС-3М
- 32 спутниковых канала слежения
- Размер: 14x14x2,5 мм
- Время первого определения: 28/25/2 с
- Чувствительность: -144/-161 дБмВт
- Потребление: < 100 мВт
- Поддержка A-GNSS и D-GNSS
- Режимы энергосбережения
- Изменяемое напряжение ввода/вывода для упрощенной интеграции в системы 3,3 В
- Готов для ЭРЫ ГЛОНАСС

ГеоС-3
- 32 спутниковых канала слежения
- Размер: 22,1x15,9x2,5 мм
- Время первого определения: 28/25/2 с
- Чувствительность: -144/-161 дБмВт
- Потребление: < 100 мВт
- Поддержка A-GNSS и D-GNSS
- Режимы энергосбережения
- Изменяемое напряжение ввода/вывода для упрощенной интеграции в системы 3,3 В
- Готов для ЭРЫ ГЛОНАСС

ГеоС-1М
- 24 спутниковых канала слежения
- Размер: 35x33x3,5 мм
- Время первого определения: 36/32/4 с
- Чувствительность: -141/-151дБмВт
- Потребление: 350 мВт
- Лидер продаж российского рынка 2010-2011 гг.
- Внесен в Государственный реестр средств измерений

ГеоС-1
- 24 спутниковых канала слежения
- Размер: 47x35x9 мм
- Время первого определения: 36/32/4 с
- Чувствительность: -141/-151дБмВт
- Потребление: 400 мВт
- OEM модуль для объемного монтажа (MMCX, розетка)
- Внесен в Государственный реестр средств измерений

АТОМА
(812) 441-23-31
www.atoma.spb.ru

AURORA Mobile Technologies
(812) 667-33-85
www.auroramobile.ru

Компэл
(495) 995-09-01
www.compel.ru

РЕКЛАМА