

РАЗРАБОТКА УНИФИЦИРОВАННОЙ СИСТЕМЫ НА КРИСТАЛЛЕ ДЛЯ БОРТОВОЙ И НАЗЕМНОЙ АППАРАТУРЫ

А. И. Егоров, С. Н. Коянкин, А. Г. Кузякин, И. А. Малахов

ФГУП «РФЯЦ-ВНИИЭФ», г. Саров Нижегородской обл.

Сокращения, используемые в докладе:

БМК – базовый матричный кристалл;

ВИП – вторичный источник питания;

ГОЧ – генератор опорной частоты;

ДЦ – дизайн-центр;

ИУ – исполнительное устройство;

КМКПД – контроллер мультиплексного канала передачи данных;

КПОС – контроллер приема и первичной обработки сигналов;

ОЗУ – оперативное запоминающее устройство,

ПВВФ – повышенные внешние воздействующие факторы;

ПЗУ – постоянное запоминающее устройство,

ПЛИС – программируемая логическая интегральная схема;

ПО – программное обеспечение;

ПП – премопередатчик;

САПР – система автоматизированного проектирования;

СБИС – сверхбольшая интегральная схема;

СНК – система на кристалле;

ЦВМ – цифровая вычислительная машина;

ЦП – центральный процессор;

ASIC – Application Specific Integrated Circuits;

GPIO – general-purpose input/output (интерфейс ввода/вывода общего назначения);

JTAG – Joint Test Action Group (стандарт IEEE 1149);

RISC – Restricted (reduced) instruction set computer (компьютер с сокращенным набором команд);

SoC – System on a Chip;

VHDL – Very high speed integrated circuit Hardware Description Language.

Введение

Постоянные требования по уменьшению габаритов и увеличению вычислительной производительности управляющих систем привели к появлению нового класса устройств, которые получили название «системы на кристалле». Под этим определением скрываются перепрограммируемые, полу-заказные

и заказные микросхемы, содержащие большинство необходимых для вычислительного устройства функциональных блоков.

Основная цель данной работы – это разработка унифицированной платформы для создания цифровых вычислительных машин на основе сложно-функциональных блоков. Возможности конфигурирования платформы позволяют проектировать вычислительные модули и модули обработки данных, отвечающие требованиям наземной и бортовой аппаратуры. Результаты проектирования могут быть реализованы в виде интегральных схем класса СНК. Унифицированная платформа предоставляет ядра центрального процессора и периферии на выбор, средства проектирования программного обеспечения, операционные системы, тесты, отладочные платы и некоторые готовые решения по реализации ЦВМ.

В докладе рассмотрен подход к разработке системы на кристалле, состоящей из центрального процессора с системой команд RISC-V и периферийных блоков, таких как контроллер памяти, контроллер МКПД по ГОСТ Р 52070-2003, GPIO контроллер, UART контроллер, SPI контроллер, USB контроллер. Представлены результаты логического синтеза и моделирования функциональных тестов некоторых ЦП.

Кроме того, в докладе рассмотрены вопросы проектирования цифровых вычислительных устройств на основе системы на кристалле с точки зрения особенностей аппаратного и программного обеспечения.

Элементная база

Основой для проектирования всех современных вычислительных устройств являются СБИС. С точки зрения внутренней структуры СБИС можно разделить на ПЛИС, БМК и полностью заказные схемы (ASIC). Основное отличие между ними заключается в том, что в ПЛИС и в БМК производителями жестко закладывается внутренняя архитектура базовых блоков, за пользователем остается выбор, как эти блоки соединить, а в полностью заказных схемах пользователь имеет возможность полностью определять архитектуру микросхемы. Преимущества и недостатки рассмотренных типов СБИС приведены в табл. 1.

Таблица 1
Сравнение характеристик ПЛИС, БМК
и заказных схем

Характеристики	ПЛИС	БМК	Заказные СБИС
Энергонезависимость	+/-*)	+	+
Низкое энергопотребление	**)	++	+++
Быстродействие	+	++	+++
Стойкость к ПВВФ	-	+	+
Перепрограммируемость	+	-	-
Быстрый цикл разработки устройства	+	-	--
Низкие затраты на производство	+	-	--
Отсутствие затрат на изготовление экспериментальных образцов	+	+/-*)	+/-*)
Стоимость изделий при крупносерийном производстве	-	+	+

Примечание:
* – в зависимости от производителя;
**) – в сравнении с эквивалентным устройством на дискретных элементах.

Разработкой СБИС занимаются специализированные отделы проектирования, или так называемые дизайн-центры. В зависимости от требований технического задания выбирается тот или иной тип СБИС, и хотя они сильно отличаются, маршрут проектирования для них схож. Актуальный на сегодняшний день маршрут проектирования СБИС включает в себя три основных этапа: ввод проекта на языке высокого уровня VHDL или Verilog, синтез проекта в выбранном базисе и, наконец, размещение на кристалле. В случае с ПЛИС все три этапа выполняются непосредственно разработчиками устройства, если же применяется БМК или заказная СБИС, то размещением на кристалле занимаются специалисты завода-изготовителя БМК. Этапы проектирования СБИС приведены на рис. 1. Цифрами условно показаны уровни участия ДЦ в процессе проектирования в зависимости от возможностей ДЦ и завода-изготовителя.

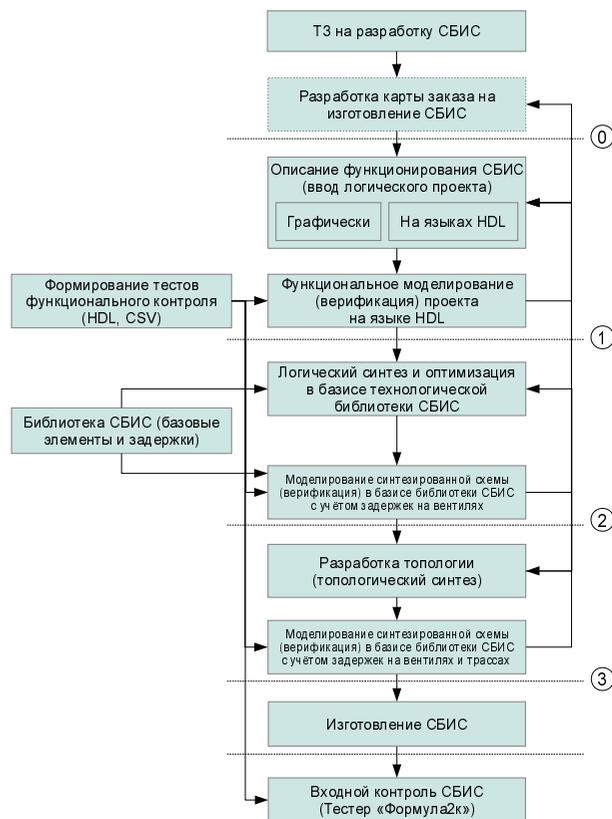


Рис. 1. Этапы проектирования СБИС

Ядро центрального процессора

В основе любой ЦВМ находится центральный процессор. Опыт отечественных [1], [2] и зарубежных разработчиков показывает, что встраивание в СНК высокопроизводительного центрального процессора приводит к возрастанию универсальности СНК. Кроме того, это позволяет быть полностью независимыми от производителей конкретных процессоров и СБИС. Имея ядро центрального процессора внутри СНК можно менять технологическую платформу СБИС, сохраняя при этом все наработки по программному обеспечению. Сейчас насчитываются сотни программных ядер центральных процессоров. Для выбора самого оптимального решения нами были выдвинуты следующие основные критерии:

- лицензионная чистота – ядро должно быть открыто и свободно, а значит лицензия должна предоставлять свободу изучения, свободу модификации, свободу распространения копий; большинство высокопроизводительных ядер, например с системами команд ARM, этому критерию не удовлетворяют;

– ядро должно оперировать с 32-х или 64-х разрядными числами – восьмиразрядные ядра хотя и занимают мало места на кристалле, однако не обеспечивают заданный уровень производительности;

– платформонезависимость – в ядре не должно быть блоков, привязанных к определенной технологической платформе, этому критерию не соответствуют, например, такие ядра, как MicroBlaze (фирма Xilinx) и Nios II Fast (фирма Altera).

Указанным критериям удовлетворяют ядра, представленные в табл. 2.

Таблица 2

Открытые и свободные ядра центральных процессоров

Система команд	Ядро
OpenRISC 1000	OpenRISC 1200, Altor32, Carpe, OR1knd, mor1kx, ORbuild
MIPS 1	Plasma
SPARC v8	Leon3
ARMv2	Amber
RISC-V [3]	PicoRV32, Rocket, Hardfloat, Uncore, Sodor, RIVER, pulpino

Другими требуемые критерии:

– масштабируемость – ядро должно иметь возможность конфигурирования на уровне подключаемых вычислительных блоков;

– системные шины – ядро должно иметь общепринятые системные шины для подключения периферийных блоков (например, AMBA или Wishbone);

– наличие заказных СБИС на основе ядра – данный опыт косвенно свидетельствует о качестве реализации;

– наличие средств проектирования программного обеспечения;

– высокая производительность;

– занимаемое место на кристалле – СНК с ядром должна занимать мало места, чтобы быть размещенной в доступных нам СБИС.

В табл. 3 приведена сводная информация по площади рассматриваемых ядер после синтеза.

Моделирование всех ядер проводилось в пакетном автоматическом режиме с использованием сценария. Результаты моделирования программы, скомпилированной под каждый ЦП с оптимизацией и без, сведены на рис. 2. По оси абсцисс указан номер вычислительного цикла, а по оси ординат – время выполнения тестовой задачи в мс, которое затрачивает ЦП на выполнение данного цикла.

Проанализировав доступные ядра на соответствие указанным критериям, мы остановились на архитектуре RISC-V как самой современной, производительной и широко применяемой. Для бортовых применений было выбрано 64-разрядное ядро RIVER с аппаратной поддержкой вещественной арифметики, а для пультовой аппаратуры – 32-разрядное целочисленное ядро PicoRV32.

Таблица 3

Сравнение результатов синтеза

Критерии	SPARC Leon	ARM	MIPS	OpenRISC Mor1k	RISC-V Pulpino
Площадь комбинационной логики	30555592	739022	2952515	13452962	2110479
Площадь последовательностной логики	44286783	561859	1039500	15768648	958381
Общая площадь	74842375 ¹⁾	1300881	3992015	29221610 ²⁾	3068861
Общая площадь в вентилях	1979957	34414	105608	773058	81186

Примечания:
 1) ЦП с кэш-памятью;
 2) ЦП с FPU;

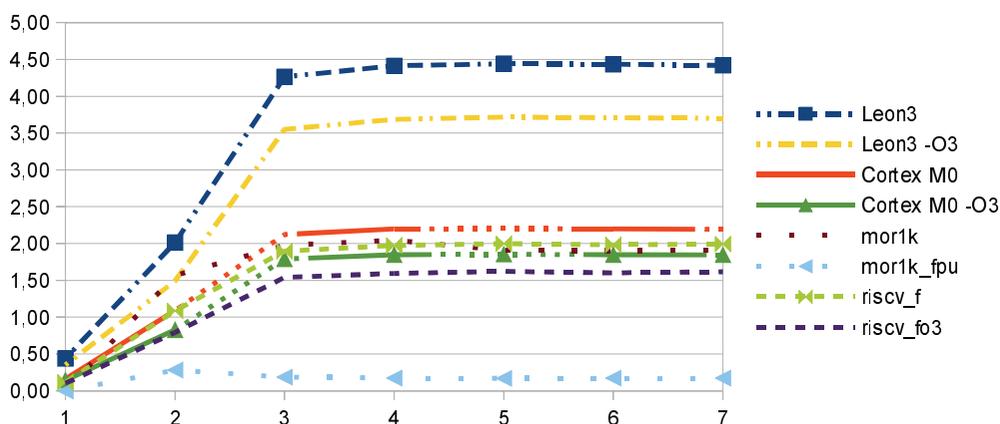


Рис. 2. Сводная таблица результатов моделирования ЦП

Система на кристалле

Для того чтобы минимизировать габариты ЦВМ и увеличить ее производительность все чаще применяются системы на кристалле. СНК (SoC), или однокристальная система – электронная схема, выполняющая функции целого устройства (например, компьютера) и размещенная на одной интегральной схеме.

Принципиальные особенности подхода СНК заключаются в:

- интеграции на одну микросхему технологической платформы (как правило, сверхвысокой степени интеграции) функционально законченного набора модулей управления и обработки данных;

- организации потока данных внутри системы непосредственно между контроллерами, а не через микропроцессорную шину;

- ориентации встроеного микропроцессора преимущественно на задачи управления (см. рис. 3) [4].

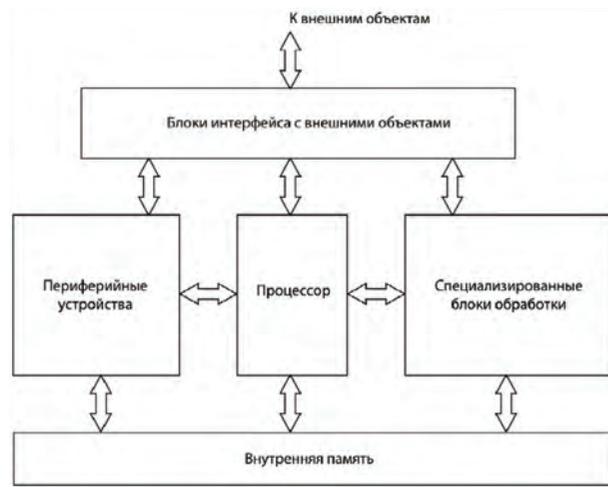


Рис. 3. Обобщенная схема СНК

Рассмотрим основные функции, выполняемые СНК в составе бортового специализированного вычислителя, структурная схема которого приведена на рис. 4.:

- ЦП – реализация определенного алгоритма управления;
- контроллер МКПД – обеспечение взаимодействия с системой управления;
- контроллер ПОС – прием и первичная обработка сигналов датчиков;
- контроллер ИУ – управление работой исполнительных устройств;
- ОЗУ – сохранение переменных данных ЦП.

СНК для наземной аппаратуры выполняет аналогичные функции при большем количестве интерфейсов.

При разработке СНК мы применяем FuseSoC – новый генератор описаний СНК, который не только предоставляется ядра с системами команд RISC-V, OpenRISC 1000 и многими другими, но также управляет ядрами периферии и позволяет легко настраивать и генерировать СНК. По своей сути он является менеджером пакетов и набором инструментов для

сборки HDL-кода (например, VHDL). Основной целью FuseSoC является увеличение объема повторно-го использования IP-ядер и быть вспомогательным средством для создания, построения и моделирования СНК.

Менеджер пакетов можно рассматривать как ПО apt, portage, yum, dnf, pacman для случая IP-ядер ПЛИС/ASIC. Обычно в простом ini-файле описывается какие необходимы файлы ядра, зависимости от других IP-ядер, и где FuseSoC должен получить код этих ядер.

Коллекция ядер вместе с описанием верхнего уровня называется системой, и системы могут быть смоделированы или переработаны в САПР от производителей ПЛИС для построения загружаемых прошивок ПЛИС.

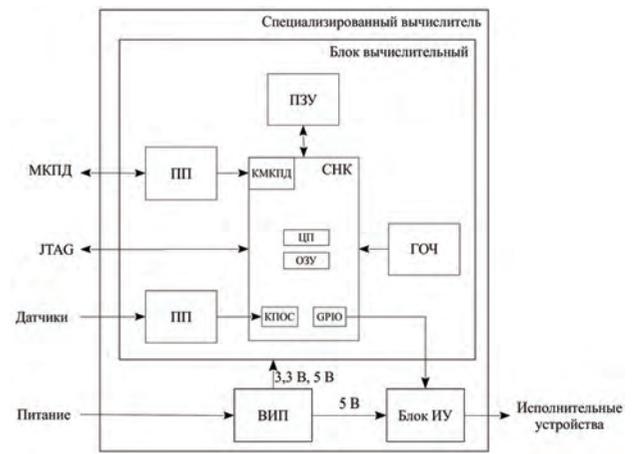


Рис. 4. Структурная схема специализированного вычислителя

FuseSoC облегчает следующие рутинные операции (рис. 5):

- повторное использование кода;
- создание конфигураций для компиляции и исполнения;
- многократный запуск регрессивного тестирования;
- портирование систем на новые аппаратные платформы;
- обмен кодом ядер со сторонними проектами;
- настройка непрерывной компоновки СНК.

Основные свойства FuseSoC:

- легкость в применении – для работы с FuseSoC большинству существующих проектов не требуется модернизации; любые специализированные исправления проектов для работы с FuseSoC могут быть применены непосредственно со время создания проекта или моделирования;

- модульность – генератор может быть использован во всем стандартном маршруте проектирования СНК, он может создавать проектные файлы для САПР или же встраиваться в специфичный маршрут;

- стандартизированная структура – могут применяться как стандартные форматы IP-ядер, такие как IP-XACT, так и специализированные;



Рис. 5 Этапы работы с FuseSoC

- расширяемость – поддерживается симуляция в GHDL, Icarus Verilog, Isim, ModelSim, Verilator и Xsim; генератор также поддерживает создание прошивок ПЛИС в САПР Lattice Semiconductor Project Icestorm, Xilinx ISE и Vivado, Altera Quartus. Для организации поддержки нового САПР достаточно написать всего лишь около 100 строчек кода;
- нетребовательность к ресурсам ПК;
- свобода использования, модернизации и расширения – генератор не накладывает ограничений на свое применение и дает возможность управления как корпоративными, так и публичными ядрами;
- проверка множеством проектов – FuseSoC успешно применяется для создания и симуляции таких проектов, как Nyuzi, Pulpino, VScale, различные OpenRISC СНК, picorv32, osvwm и многие другие;
- ресурсоемкость – стандартная библиотека ядер на данный момент включает более 100 ядер, в том числе ЦП, периферийные контроллеры, внутренние соединения, законченные СНК и библиотечные утилиты. В дополнение к стандартной библиотеке можно добавить сторонние библиотеки и ядра.

Унифицированная платформа для разработки ЦВМ и маршрут проектирования устройств на основе СНК

Для ускорения разработки ЦВМ нами была подготовлена унифицированная платформа, включающая в себя следующие блоки:

- набор ядер ЦП (32-х и 64-битные) и периферии (МКПД по ГОСТ Р 52070-2003, GPIO, UART, SPI, USB);
- средства проектирования программного обеспечения – для создания своего программного обеспечения имеется набор инструментальных средств от проекта GNU, которые позволяют разрабатывать ПО на языках С и С++. При создании ПО под процессоры RISC-V используется кросс-компиляторы. Набор инструментов исполняется главным компьютером и генерирует бинарный код программ или библиотек, которые могут быть запущены потом на процессоре;
- операционные системы – FreeRTOS, ZephyrOS, Linux, RTEMS;

- тесты, проводящие формальную верификацию СНК и измеряющие производительность;
- отладочные комплекты на основе плат «Марсоход3» (производитель ООО «Инпро Плюс», город Таганрог);
- готовые решения для бортовой и наземной аппаратуры.

Данная платформа охватывает все основные этапы проектирования:

- разработка аппаратного обеспечения;
- разработка ПО;
- функциональное моделирование аппаратуры;
- отладка, симуляция и тестирование ПО;
- совместное моделирование аппаратного и программного обеспечения;
- прототипирование вычислительного устройства.

Создание и расширение такой платформы в дальнейшем позволит многократно сократить сроки разработки ЦВМ.

Разработка приемопередатчика интерфейса USB на отечественной элементной базе

В процессе создания унифицированной платформы для разработки ЦВМ перед нами одной из основных целей было импортозамещение в части применяемой элементной базы. При наличии отечественных микросхем ПЛИС, БМК, приемопередатчиков МКПД и RS-232, дискретные ПП USB отсутствуют, хотя данный интерфейс широко применяется в контрольно-технологической аппаратуре. Поэтому нами было принято решение реализовать такой ПП на отечественной элементной базе.

ПП на основе схемы из спецификации интерфейса USB «Universal Serial Bus. Specification. Revision 2.0. 2000» представлен на рис. 6. Для обеспечения работы встроенного в СБИС контроллера USB требуется компаратор (например, 1467СА3Т), два триггера Шмитта (например, 5584ТЛ2АТ) и два буфера с третьим состоянием (например, 5584АП6АТ). В зависимости от технологии изготовления СБИС данные элементы можно интегрировать внутри СНК. После блока, реализующего описанный физический уровень USB, внутри СНК следует IP-ядро, реализующее протокольную часть.

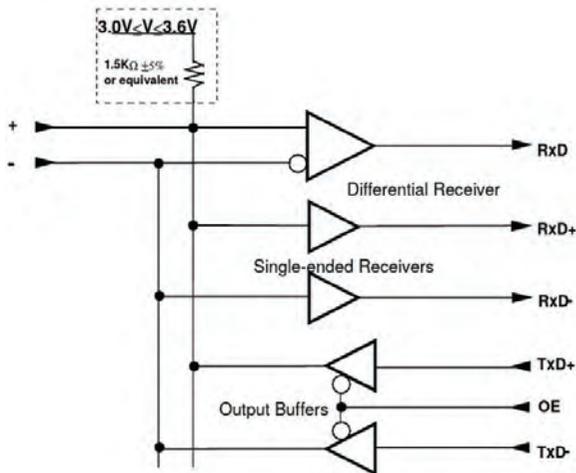


Рис. 6. Приемопередатчик USB

Заключение

В докладе был рассмотрен накопленный опыт проектирования унифицированной СНК, показаны особенности элементной базы и проектирования СНК.

Проведен анализ доступных ядер центральных процессоров. Рассмотрена разработанная платформа для проектирования ЦВМ, а также показан способ реализации ПП USB.

Представленные в докладе подходы к проектированию позволяют многократно увеличить скорость проектирования с одновременным улучшением характеристик ЦВМ.

Литература

1. Стешенко В., Руткевич А., Бумагин А., Гулин Ю., Воронков Д., Гречишев Д., Евстигнеева Е., Синельникова М. Опыт разработки СБИС типа СНК на основе встроенных микропроцессорных ядер. Компоненты и технологии, 2008, № 10.
2. Немудров В., Мартин Г. Системы-на-кристалле. Проектирование и развитие. М.: Техносфера, 2004.
3. <http://riscv.org/>
4. Шагурин И. Системы на кристалле. Особенности реализации и перспективы применения. Электронные компоненты, 2009, № 1.