

Проектирование конфигурируемого 32-разрядного RISC-V микропроцессора

С.А. Корнев, В.В. Андреев

МГТУ им. Н.Э. Баумана, Калужский филиал, г. Калуга, s.kornev.tov@gmail.com

Аннотация — Описаны характеристики разработанного 32-разрядного микропроцессора с архитектурой набора команд RISC-V. Рассмотрены его основные модули и доступные конфигурации, а также возможные сферы применения данного микропроцессора. Описана конструкция и характеристики тестовой микросхемы на основе разработанного микропроцессора, а также измерено её энергопотребление и падение напряжения на линиях питания при напряжении питания 3,3 В и 5 В. Проведён анализ производительности различных конфигураций микропроцессора. Представлен сравнительный анализ производительности разработанного микропроцессора и его аналогов. Проведён анализ зависимости площади, занимаемой микропроцессором на кристалле от его конфигурации. Сделаны выводы об оптимальности применения определённых конфигураций микропроцессора для решения различных типов задач.

Ключевые слова — микропроцессор, 32-разрядный, RISC-V, RISC, энергопотребление, производительность, бенчмарк, процессор, CPU.

I. ВВЕДЕНИЕ

Современные системы на кристалле (СнК) и ПЛИС используются для решения самых разных задач, от создания высокопроизводительных систем обработки данных, до разработки систем с низким энергопотреблением, используемых в портативных устройствах, устройствах интернета вещей и др. Системы на кристалле включают в себя различные аппаратные блоки, такие как микропроцессоры, встроенные блоки ОЗУ и ПЗУ, а также периферийные блоки для взаимодействия с внешним миром. Этому прежде всего способствует развитие технологических процессов в микроэлектронике, которое позволяет использовать элементы с меньшими размерами и энергопотреблением, что позволяет разместить на чипе, большее число функциональных блоков [1].

Использование языков описания аппаратуры и современных средств автоматизированного проектирования электроники (EDA), позволяет создавать реконфигурируемые модули для таких схем, которые легко могут быть адаптированы под различные задачи, технологические процессы и системы управления [2].

Возможность реконфигурирования полезна и для микропроцессоров. В зависимости от задачи, в которой используется СнК или ПЛИС, микропроцессор в её

составе может быть оснащён набором модулей, повышающих его производительность, или же наоборот он может иметь конфигурацию, понижающую его энергопотребление и размеры на кристалле [3].

Целью данной работы является анализ характеристик разработанного конфигурируемого 32-разрядного микропроцессора с архитектурой набора команд RISC-V, который нацелен применение в СнК и ПЛИС. Кроме того, в статье проведены моделирование его характеристик и сравнение с аналогами (серийно выпускаемыми микроконтроллерами и IP-блоками микропроцессоров).

II. НАЗНАЧЕНИЕ И ОПИСАНИЕ РАЗРАБОТАННОГО МИКРОПРОЦЕССОРА

Для разработанного микропроцессора была выбрана готовая архитектура набора команд RISC-V. Использование уже существующего набора команд в качестве базы для создания микропроцессора обладает множеством преимуществ. Прежде всего это позволяет:

- получить доступ к уже готовым средствам разработки и отладки;
- иметь возможность быстрого переноса программ между процессорами, использующими тот же набор команд;
- обеспечить более быстрое изучение разработчиками данного микропроцессора, в особенности теми, кто уже имел опыт написания программ с использованием архитектуры RISC-V.

Кроме того, преимуществами архитектуры RISC-V являются её открытость, бесплатность и модульность.

Модульность RISC-V позволяет выбрать лишь необходимые для процессора расширения набора команд, которые могут быть полезны при применении процессора в встраиваемых системах [4, 5].

Разработанный микропроцессор может быть использован в качестве IP-блока для управления функциональными элементами в составе ПЛИС, специализированных заказных микросхем и систем на кристалле, а также для создания микроконтроллеров.

Основные параметры, характеризующие разработанный микропроцессор:

- базовый набор целочисленных 32-битных команд RV32I или RV32E (идентичен RV32I, но имеет меньший регистровый файл);
- 5-ступенчатый вычислительный конвейер;
- возможность добавления блока предсказания переходов и байпаса (bypass) для ускорения обработки инструкций;
- поддержка добавления кэш-памяти;
- регистровый файл с 32 или 16 32-битными целочисленными регистрами для базовых наборов RV32I и RV32E соответственно;
- Поддержка невыровненного по 32-битной границе доступа к памяти данных;
- Два уровня привилегий – user и machine;
- поддержка 3 прерываний машинного уровня необходимых по стандарту RISC-V, а также 16 дополнительных прерываний машинного уровня;
- прямой (direct mode) и векторный (vector mode) режим обработки прерываний, с возможностью задания начального адреса таблицы векторов прерываний;
- наличие одного немаскируемого прерывания (NMI);
- поддержка режима ожидания прерываний (WFI).

Поддержка двух базовых наборов команд, с различным числом доступных регистров, позволяет выбрать между минимизацией размеров процессора и повышением производительности, за счёт использования большего числа регистров, для хранения информации [6].

Одним из основных преимуществ разработанного микропроцессора является его модульность. В зависимости от выполняемой задачи он может быть сконфигурирован с использованием различных расширений позволяющих получить либо повышенное быстродействие, либо уменьшить размеры, занимаемые процессором на кристалле. Это позволяет гибко адаптировать процессор под конкретную задачу путём выбора лишь необходимых для её решения модулей.

Структурная схема микропроцессора в конфигурации, включающей в себя все доступные модули, представлена на рис. 1.

Расширения набора поддерживаемых процессором команд описаны ниже:

- обработка сжатых 16-битных команд (расширение C) – позволяет экономить память

необходимую для хранения управляющей программы в среднем на 25-30 % [7];

- поддержка аппаратного целочисленного умножения и деления (расширение M), позволяет получить 64 битный результат целочисленного умножения 32-битных знаковых и беззнаковых чисел, а также неполное частное и остаток от деления;
- регистры статуса и управления (CSR) машинного уровня, используемые для управления прерываниями, таймерами, получения информации о доступных модулях и контроля различных параметров процессора, доступны в модуле Zicsr;
- расширение Zifencei используемое для взаимодействия с кэш-памятью инструкций.

Доступные 19 прерываний машинного уровня и одно немаскируемое прерывание, могут использоваться для управления различными датчиками и функциональными блоками системы, в которой используется разработанный микропроцессор. Большое число прерываний может быть добавлено с помощью внешнего контроллера прерываний.

Блок аппаратного умножения и деления позволяет получить полный 64-битный результат умножения двух знаковых чисел или умножения знакового на беззнаковое число за 64 такта. Полный 64-битный результат умножения двух беззнаковых чисел доступен за 32 такта. Блок позволяет получить неполное частное и остаток от деления целых чисел за 33 такта. Также блок сохраняет уже полученные результаты и позволяет их использовать, не затрачивая времени на повторное вычисление, при соблюдении условий, указанных в спецификации архитектуры RISC-V.

Для обеспечения доступа к памяти данных и инструкций, процессор использует 32-битные шины, а также 32-битное адресное пространство. Данные в памяти должны иметь формат порядок байтов little-endian. Процессор позволяет получить доступ к 8, 16 и 32-битным данным с помощью 4-битной маски записи и чтения данных.

Другая важная особенность процессора, это возможность невыровненного по 32-битной границе доступа к памяти данных. Доступ к данным по невыровненному адресу либо разбивается на две транзакции по выровненным адресам, либо используется только 4-битная маска записи/чтения для выбора нужных байт данных.

Минимальное время доступа к памяти данных составляет 2 такта, а доступ к памяти инструкций может осуществляться за 1 такт. Время доступа во многом зависит от используемой памяти и конфигурации системы, в которой установлен процессор.

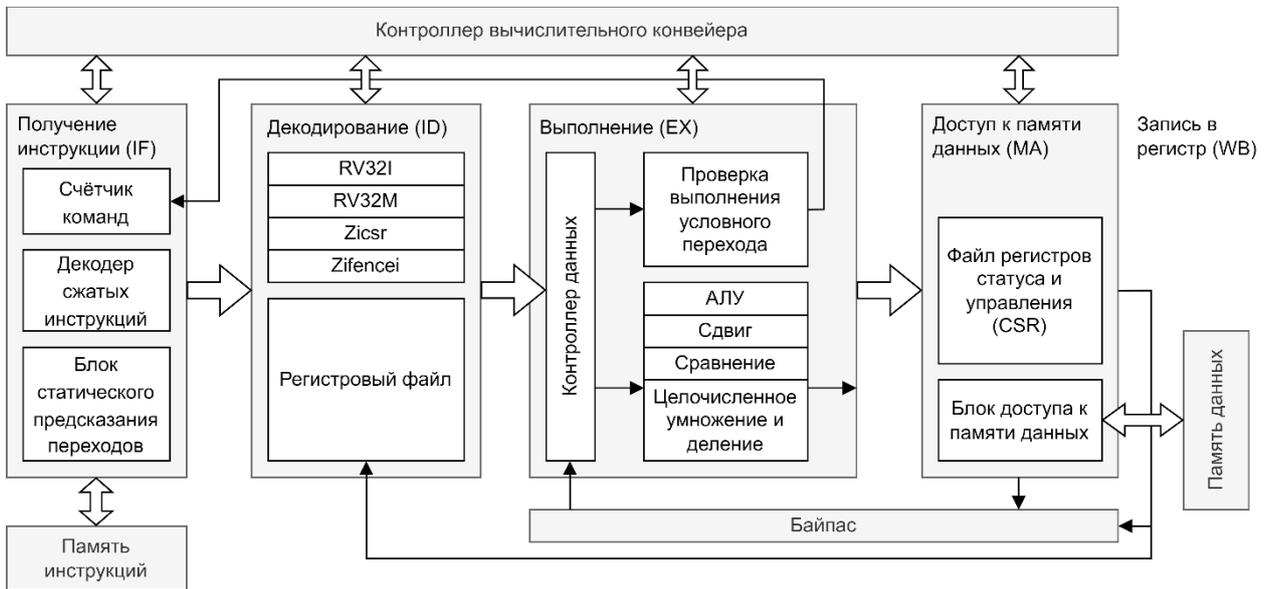


Рис. 1. Структурная схема вычислительного конвейера разработанного микропроцессора в конфигурации со всеми доступными модулями и расширениями архитектуры

III. МИКРОСХЕМА МИКРОПРОЦЕССОРА

На основе разработанного микропроцессора была создана микросхема, предназначенная для его дальнейшего тестирования и анализа его параметров [6]. Для микросхемы была выбрана КМОП КНИ технология с тремя металлами и нормами 0,6 мкм. Совмещённая топология микросхемы представлена на рис. 2.

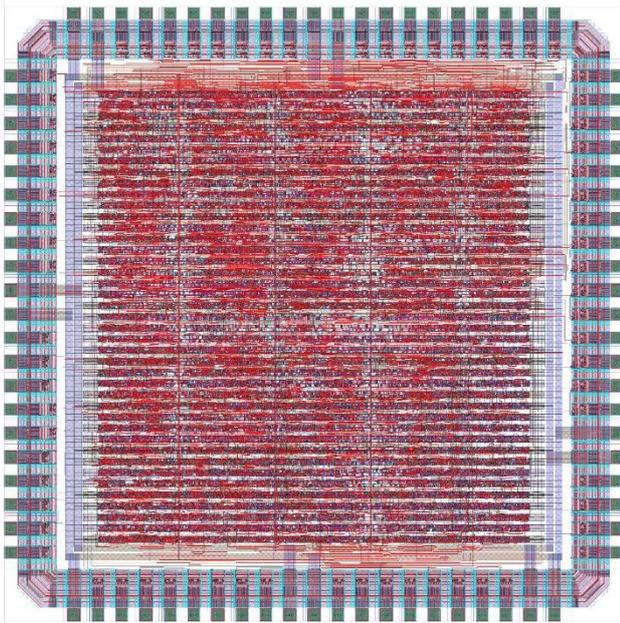


Рис. 2. Совмещённая топология микросхемы разработанного 32-битного RISC-V микропроцессора

Микросхема содержит микропроцессор и контроллер доступа к памяти. Микропроцессор представлен в конфигурации с поддержкой набора команд RV32IMC, а также он имеет блок статического

предсказания переходов и байпас. Контроллер доступа к памяти используется для распределения доступа к общей шине доступа к памяти при получении данных и инструкций.

Сам кристалл микросхемы имеет размеры 4898x4898 мкм, при этом для размещения цифрового ядра микросхемы доступно пространство 4070x4070 мкм. Микросхема имеет 84 вывода, 32 из которых занимает двунаправленная шина доступа к памяти. Также представлены выводы для 20-битной адресной шины и 7 прерываний машинного уровня. Подача питания осуществляется с помощью 6 пар выводов питания и земли, равномерно распределённых по сторонам кристалла. Остальные выводы используются для взаимодействия с внешней памятью, подачи тактового сигнала и сигнала сброса.

Цифровое ядро схемы содержит более 8 000 цифровых элементов и заполнено ими примерно на 70%, оставшуюся площадь занимают конденсаторы по питанию и блоки заполнители.

Микросхема способна работать в диапазоне температур от -40 °C до +125 °C на частоте до 30 МГц. Важной особенностью данной микросхемы является возможность работы при напряжении питания 3,3 и 5 В. Допустимый разброс напряжений питания составляет ±10%.

Процесс разработки и верификации микросхемы осуществлялся в САПР Cadence Genus, Cadence Innovus и Cadence Virtuoso.

Проведённое моделирование подтвердило её работоспособность во всём диапазоне рабочих температур и частот тактового сигнала при различных отклонениях технологического процесса.

Также в процессе моделирования было измерено энергопотребление микросхемы в зависимости от

частоты, его результаты приведены в табл. 1. Энергопотребление во многом зависит от выполняемых процессором задач, отклонений технологического процесса и температуры [8, 9]. В данном случае были взяты данные об энергопотреблении при выполнении процессором бенчмарка CoreMark, для температуры +125 °С и отклонений технологического процесса, дающих максимальное энергопотребление при данных условиях. Такие параметры дают максимально возможное энергопотребление.

Таблица 1

Среднее энергопотребление микросхемы микропроцессора при запуске теста CoreMark, мВт

Тактовая частота, МГц	Напряжение питания, В	
	3,3	5,0
10	63,86	177,95
20	94,51	245,90
30	128,73	327,15

Как видно из табл. 1, энергопотребление при напряжении питания 5 В примерно в 2,9 раза больше, чем при напряжении питания 3,3 В при одинаковой тактовой частоте. В основном энергопотребление определяется динамической мощностью, потребляемой во время переключения элементов схемы.

Кроме расчёта энергопотребления, также была проведена оценка падения напряжения на линиях питания (IR-drop). Микросхема имеет кольца питания и земли шириной 50 мкм, расположенные вокруг цифрового ядра схемы, а также дополнительные вертикальные линии питания и земли шириной 8 мкм, через каждые 750 мкм. Такая планировка системы питания позволяет снизить падения напряжения на линиях питания [10]. С помощью теоретических расчётов учитывающих максимально допустимую величину тока для каждого из металлов, была проведена оценка падения напряжения на линиях питания. В наихудшем случае оно может составить 2,35% и 1,55 % для напряжений питания 3,3 В и 5 В соответственно.

Важно заметить, что во время реальной работы микросхемы микропроцессора максимально допустимые величины токов на всех участках схемы практически невозможны и соответственно падение напряжения на линиях питания будет гораздо ниже. Тем не менее данная оценка может служить в виде верхней границы возможного падения напряжения на линиях питания.

IV. ПРОГРАММНЫЕ ТЕСТЫ

Проведение программных тестов является ещё одной важной частью процесса разработки микропроцессора.

В первую очередь были проведены тесты нацеленные на проверку совместимости

разработанного микропроцессора с архитектурой RISC-V. Для проведения тестирования совместимости микропроцессоров с стандартом архитектуры RISC-V существуют различные решения [11]. В данном случае использовался набор тестов из репозитория RISC-V Architecture Test SIG [12], который позволяет проверить правильное выполнение команд базового набора и используемых расширений. Тестирование на совместимость с RISC-V велось с помощью ПО Icarus Verilog, путём модулирования системы, включающей процессор и необходимые периферийные блоки, на которой были запущены данные тестовые программы. Компиляция тестовых программ для разработанного микропроцессора выполнялась с помощью компилятора GCC. Были проведены тесты на совместимость с набором команд RV32I, расширениями C, M, Zicsr, Zifencei, а также привилегированными режимами Machine и User.

Результаты показали, что микропроцессор успешно прошёл все представленные тесты из пакета RISC-V Architecture Test SIG для имеющегося у него набора команд, тем самым подтвердив свою совместимость с стандартом архитектуры RISC-V.

Следующим этапом стала оценка производительности. Для этого обычно используется специализированные тестовые программы (бенчмарки), для которых измеряется время выполнения, потребляемая память и другие параметры [13]. Одним из наиболее распространённых бенчмарков для процессоров является Dhrystone, который не содержит вычислений с плавающей запятой и позволяет оценить производительность процессоров, которые имеют аппаратную поддержку лишь целочисленных вычислений. Тем не менее, он имеет ряд ограничений и недостатков, к примеру сильную зависимость от оптимизаций компилятора. Поэтому для тестирования процессора было решено выбрать бенчмарк CoreMark, который был разработан специально для тестирования микропроцессоров, нацеленных на применение в встраиваемых системах. В бенчмарке проводится множество итераций различных вычислений над матрицами, расчёта контрольных сум данных и целочисленного умножения и деления. CoreMark позволяет более объективно, по сравнению с Dhrystone, оценить производительность микропроцессора, так как в нём используются вычисления, которые часто проводятся в предполагаемых сферах использования разработанного микропроцессора. Также он менее подвержен влиянию оптимизаций компилятора на результаты тестирования.

Кроме влияния оптимизаций компилятора, на результаты бенчмарков могут повлиять и параметры системы, в которой находится микропроцессор. Время доступа к памяти, наличие или отсутствие кэш-памяти, возможность одновременного доступа к памяти данных и инструкций и др., могут сильно повлиять на результаты оценки производительности [14].

Чтобы уменьшить влияние параметров тестовой системы на результаты тестирования, для всех конфигураций использовалась одна и та же тестовая система на ПЛИС, включающая в себя:

- тестируемую конфигурацию микропроцессора (тактовая частота работы системы 40 МГц) с блоком статического предсказания переходов и байпасом;
- внешнюю SPI Flash память для хранения управляющей программы (время доступа к 32 битам данных – 64 такта) в одной из серий измерений, и внутреннюю память ПЛИС, в другой серии измерений производительности конфигураций микропроцессора;
- кэш-память инструкций размером 1 Кбайт;
- ОЗУ размером 16 Кбайт;
- 64-битный таймер для замера времени выполнения программы бенчмарка;
- модуль универсального асинхронного приёмопередатчика (UART) для передачи информации о результатах оценки производительности.

Использование для проведения измерений производительности медленной внешней SPI FLASH памяти и быстрой внутренней памяти ПЛИС в качестве ПЗУ для хранения управляющей программы, позволит оценить влияние характеристик системы на производительность процессора.

Для измерения производительности разработанной тестовой микросхемы микропроцессора, использовалось моделирование Verilog описания микросхемы, а также блоков ПЗУ (32 Кбайт), ОЗУ (16 Кбайт), 64-битного таймера и модуля UART. Тактовая частота системы 30 МГц.

Результаты бенчмарка CoreMark для различных конфигураций микропроцессора представлены в табл. 2. Все тесты производились при одинаковых настройках компилятора GCC. Параметр CoreMark отражает число выполненных итераций бенчмарка в секунду. Он не слишком удобен для сравнения процессоров, так как зависит от тактовой частоты процессора. Поэтому для сравнения чаще используется параметр CoreMark / МГц отражающий число итераций бенчмарка в секунду, делённое на тактовую частоту в мегагерцах.

Как видно из табл. 2, параметры системы оказывают огромное влияние на производительность процессора. Так, конфигурации микропроцессора с набором инструкций RV32IMC на ПЛИС (с использованием внутренней памяти) и микросхеме различаются по параметру производительности CoreMark / МГц, примерно в 2,2 раза. Из полученных результатов очевидно влияние используемой ПЗУ на параметры процессора, использование внешней SPI FLASH памяти приводит к значительному снижению

производительности, по сравнению с аналогичными конфигурациями процессора, использующими внутреннюю память ПЛИС для хранения программы.

Низкая производительность тестовой микросхемы микропроцессора объясняется тем, что она имеет общую шину данных и инструкций, а также минимальное время доступа к памяти равное двум тактам.

Таблица 2

Сравнение производительности различных конфигураций микропроцессора с помощью бенчмарка CoreMark

Конфигурация	ПЗУ	CoreMark / МГц	CoreMark
RV32IM (ПЛИС)	Внутр.	1,14	45,23
RV32IMC (ПЛИС)	Внутр.	1,08	43,59
RV32I (ПЛИС)	Внутр.	0,86	34,76
RV32IC (ПЛИС)	Внутр.	0,84	33,52
RV32E (ПЛИС)	Внутр.	0,76	30,48
RV32IMC (ПЛИС)	Внешн. FLASH	0,75	30,06
RV32EC (ПЛИС)	Внутр.	0,74	29,61
RV32IMC (Микросхема)	Внешн.	0,52	15,60
RV32IM (ПЛИС)	Внешн. FLASH	0,41	16,33
RV32IC (ПЛИС)	Внешн. FLASH	0,30	11,93
RV32EC (ПЛИС)	Внешн. FLASH	0,30	11,89
RV32I (ПЛИС)	Внешн. FLASH	0,23	9,26
RV32E (ПЛИС)	Внешн. FLASH	0,21	8,50

V. СРАВНЕНИЕ ХАРАКТЕРИСТИК И РЕКОМЕНДАЦИИ ПО ВЫБОРУ КОНФИГУРАЦИИ

На основе полученных с помощью бенчмарка CoreMark данных о производительности различных конфигураций микропроцессора (табл. 2), было произведено их сравнение с другими микропроцессорами и микроконтроллерами со схожими характеристиками. Результаты сравнения по параметру CoreMark / МГц представлены в табл. 3.

Для сравнения была использована конфигурация микропроцессора для ПЛИС с набором команд RV32IM, использующая внутреннюю память ПЛИС для хранения программы, а также оснащённая статическим предсказателем переходов и байпасом.

Как видно из табл. 3, она близка по производительности к микроконтроллерам NXP LPC2939 и MSP430F5529. Используемые для сравнения данные взяты из официального сайта бенчмарка CoreMark [15] и репозитория микропроцессора Ibex RISC-V Core [16].

Таблица 3

Сравнение производительности микропроцессоров

Микропроцессор / Микроконтроллер	Архитектура	CoreMark / МГц
Ibex RISC-V Core (RV32IMC)	RISC-V	2,47
STM32F103RB	ARM	1,5
PIC24HJ128GP202	PIC24	1,29
NXP LPC2939	ARM	1,18
Разработанный микропроцессор (RV32IM)	RISC-V	1,14
MSP430F5529	MSP430	1,11
ATMEGA4809	AVR	1,05
ATXMEGA128A1U	AVR	0,92
Ibex RISC-V Core (RV32EC)	RISC-V	0,90
NXP i.MX25	ARM	0,86
NXP LH7A404	ARM	0,78
PIC24FJ64GA004	PIC24	0,75
MSP430F5438	MSP430	0,62
ATmega2560	AVR	0,53
AT89C51RE2	80C52 / 8051	0,11

Для того, чтобы оценить различие в занимаемой площади между различными конфигурациями разработанного микропроцессора был проведён их логический и физический синтез с помощью САПР OpenLane, которая позволяет реализовать полный маршрут проектирования цифровых микросхем от RTL описания проекта, до получения GDSII файлов. В ходе синтеза использовалось несколько PDK для сбора статистики о занимаемой площади, в частности PDK Google SkyWater с нормами 130 нм.

На рис. 3 представлено сравнение размеров конфигураций процессора с поддержкой различных расширений наборов команд. Размеры представлены в процентах относительно конфигурации RV32IMC, занимающей наибольшую площадь на кристалле, и усреднены по данным синтеза для использованных PDK. Все приведённые конфигурации включают в себя блок статического предсказания переходов и байпас.

Схожие данные по процентному соотношению размеров справедливы и для размещения микропроцессора на ПЛИС, к примеру, конфигурация с набором команд RV32IMC, статическим предсказателем переходов и байпасом занимает порядка 4 000 блоков LUT и 1250 FF на ПЛИС Xilinx Artix-7 100T.

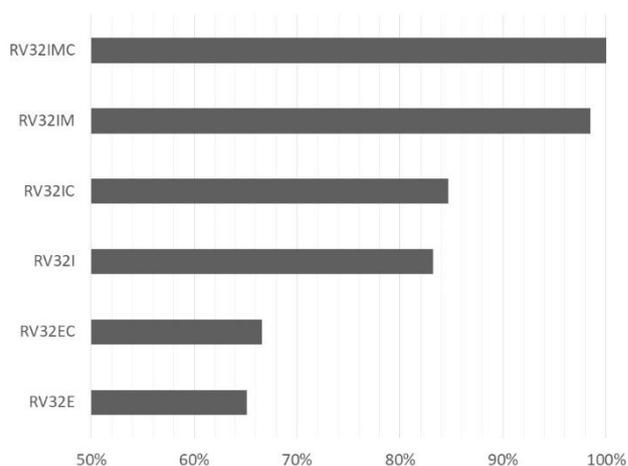


Рис. 3. Сравнение размеров различных конфигураций разработанного микропроцессора в % относительно конфигурации RV32IMC

На основе изложенной выше информации о характеристиках различных конфигураций разработанного микропроцессора можно сформировать перечень наиболее оптимальных конфигураций под различные задачи.

Конфигурация RV32IMC, обладающая одним из самых больших уровней производительности среди представленных конфигураций, подойдёт для решения задач связанных с большим числом вычислений, в которых от микропроцессора в первую очередь требуется высокая производительность, а требования к занимаемой площади на кристалле отходят на второй план. Использование блока аппаратного целочисленного умножения и деления, а также статического предсказателя переходов и байпаса, позволяет в среднем на 30 % повысить скорость выполнения инструкций по сравнению с другими конфигурациями. При этом наличие декодера сжатых 16-битных инструкций в среднем уменьшает затрачиваемую на хранение управляющей программы память на 25-30 % [7]. Также для высокопроизводительных систем может подойти конфигурация RV32IM, обладающая чуть большей производительностью, чем конфигурация RV32IMC, но при этом не имеющая декодера сжатых инструкций.

Конфигурация RV32EC может использоваться для приложений, где не требуется высокая производительность, но предъявляются высокие требования к минимизации размеров управляющих блоков. Основной вклад в уменьшение площади вносит использование регистрового файла с 16 32-битными регистрами вместо 32. Использование декодера сжатых 16-битных команд немного увеличивает занимаемую площадь по сравнению с конфигурацией RV32E, но позволяет достичь экономии общего пространства, занимаемого системой, за счёт уменьшения размера необходимого для хранения управляющей программы ПЗУ.

Компромиссом между достижением высокой производительности и минимизацией площади,

является конфигурация с поддержкой набора команд RV32IC. Она не имеет блока аппаратного целочисленного умножения и деления, что несколько снижает её производительность при большом числе арифметических вычислений, но тем не менее, она содержит полный регистровый файл с 32 32-битными регистрами, в отличие от конфигурации RV32EC, что позволяет держать большее число переменных в нём и совершать меньше запросов к памяти данных.

VI. ЗАКЛЮЧЕНИЕ

Разработанный 32-разрядный конфигурируемый RISC-V микропроцессор может использоваться для решения широкого спектра задач в встраиваемых системах. Данный микропроцессор может применяться в составе систем на кристалле, а также ПЛИС.

Правильность выполнения команд архитектуры RISC-V данным микропроцессором, была подтверждена результатами набора тестов RISC-V Architecture Test SIG.

Разработана тестовая микросхема микропроцессора с набором команд RV32IMC, блоком статического предсказания переходов и байпасом. Для её создания использовалась КМОП КНИ технология с нормами 0,6 мкм. Выполнено моделирование энергопотребления и проведён анализ падения напряжения на линиях питания разработанной микросхемы микропроцессора.

Проведено моделирование производительности различных конфигураций микропроцессора, а также получены данные о соотношении размеров данных конфигураций при их размещении на кристалле.

Произведён сравнительный анализ производительности разработанного микропроцессора и различных серийно выпускаемых микроконтроллеров, а также аналогичных IP-блоков микропроцессоров.

На основе полученной статистики о производительности и размерах различных конфигураций микропроцессора был разработан перечень рекомендаций по выбору оптимальной конфигурации микропроцессора в зависимости от выполняемой задачи и сферы его применения.

ЛИТЕРАТУРА

[1] Canal B., Bonatto A. Physical Implementation of a 32-bits RISC microprocessor using XFAB 600nm technology // 32° SIMPOSIO SUL DE MICROELETRONICA. 2017. P. 1–4.
 [2] Andreev D.V., Bondarenko G.G., Andreev V.V., Stolyarov A.A. Use of High-Field Electron Injection into Dielectrics to Enhance Functional Capabilities of Radiation MOS

Sensors // Sensors. 2020. V. 20. № 8. P. 2382(1-11). doi: 10.3390/s20082382
 [3] Barriga A. RISC-V processors design: a methodology for cores development // 2020 XXXV Conference on Design of Circuits and Integrated Systems. 2020. P. 1–6. doi: 10.1109/DCIS51330.2020.9268639
 [4] Patterson D., Hennessy J. Computer Organization and Design RISC-V Edition: The Hardware Software Interface. Second Edition. Morgan Kaufmann, 2020.
 [5] Raveendran A., Patil V., Selvakumar D., Desalphine V. A RISC-V instruction set processor-micro-architecture design and analysis // 2016 International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA). 2016. P. 1–7. doi: 10.1109/VLSI-SATA.2016.7593047
 [6] Корнев С.А., Андреев В.В., Столяров А.А. Разработка специализированного конфигурируемого 32-битного микропроцессора с архитектурой RISC-V / Сб. трудов «XLVI Академические чтения по космонавтике, посвященные памяти академика С.П. Королёва и других выдающихся отечественных ученых — пионеров освоения космического пространства. Королёвские чтения 2022». Т. 4. М.: Издательство МГТУ им. Н. Э. Баумана. 2022. С. 580–583.
 [7] Waterman A., Lee Y., Patterson D., Asanović K. The RISC-V Instruction Set Manual, Volume I: User-Level ISA, Document Version 20191213. RISC-V Foundation. 2019.
 [8] Weste N., Harris D. CMOS VLSI Design. A Circuits and Systems Perspective 4th. ed. Addison-Wesley, 2015.
 [9] Shetty A. ASIC Design Flow And Methodology – An Overview" SSRG International Journal of Electrical and Electronics Engineering // SSRG International Journal of Electrical and Electronics Engineering. 2019. V. 6. № 7. P. 1–5. doi: 10.14445/23488379/IJEEE-V6I7P101
 [10] Reddi V.J., et al. Voltage Noise in Production Processors // IEEE Micro. 2011. V. 31. № 1. P. 20–28. doi: 10.1109/MM.2010.104
 [11] Камкин А.С., Проценко А.С., Смолов С.А., Татарников А.Д. Генератор тестовых программ для архитектуры RISC-V на основе инструмента MicroTESK // Проблемы разработки перспективных микро- и нанoeлектронных систем. 2018. Выпуск 2. С. 2–8. doi:10.31114/2078-7707-2018-2-2-8
 [12] Gala N., Karasek M., et al. The RISC-V Architectural Compatibility Test Framework Version 2. RISC-V International, 2021.
 [13] Bora S., Paily R. A High-Performance Core Micro-Architecture Based on RISC-V ISA for Low Power Applications // IEEE Transactions on Circuits and Systems II: Express Briefs. 2021. V. 68. № 6. P. 2132–2136. doi: 10.1109/TCSII.2020.3043204
 [14] Tanenbaum A.S. Structured Computer Organization. 6th Edition. Pearson, 2012.
 [15] URL: <https://www.eembc.org/coremark/scores.php> (дата обращения: 25.08.2022)
 [16] URL: <https://github.com/lowRISC/ibex> (дата обращения: 25.08.2022).

Designing a Configurable 32-Bit RISC-V Microprocessor

S.A. Kornev, V.V. Andreev

Bauman Moscow State Technical University, the Kaluga branch, Kaluga, s.kornev.tov@gmail.com

Abstract — Implementation of a configurable 32-bit microprocessor based on RISC-V instruction set architecture for embedded applications and integration into systems on a chip is presented. Processor core can also be used in field-programmable gate arrays (FPGA). Advantages of using RISC-V architecture include common availability and free use, including for commercial purposes, as well as architecture modularity. The processor can be configured to use 16 (RV32E instruction set) or 32 registers (RV32I instruction set) with a 32-bit width. The following architecture extensions are available as well: Standard Extension for Integer Multiplication and Division in 32 clock cycles (extension M), Standard Extension for 16-bit Compressed Instructions to save memory (extension C), two privileged modes (Machine and User), as well as a set of control and status registers (CSR) required for them. The processor has 5-stage instruction pipeline. Branch predictors, bypass and instruction cache can be added as options. Capability to flexibly configure the processor core depending on the tasks and limitations imposed on the developed chip is demonstrated. Performance and area occupied on the chip indicators of various processor core configurations are defined. One of the microprocessor configurations with RV32IMC instruction set, branch predictors, bypass, as well as hardware integer multiplication and division was implemented in CMOS SOI 0.6 μm technology using the Cadence EDA Tools. Various parameters of this CMOS chip containing the processor core under development were simulated. In particular, power consumption and the voltage drop (IR drop) on the power lines were analyzed. The software tests performed on the processor demonstrated its full compatibility with the RISC-V instruction set architecture and allowed to evaluate its performance against other processor cores and microcontrollers. The results of the CoreMark benchmark for various microprocessor configurations are also presented. Comparison of the performance of the developed processor core with other processors and microcontrollers is shown.

Keywords — microprocessor, 32-bit, RISC-V, RISC, power consumption, performance, benchmark, processor, CPU.

REFERENCES

- [1] Canal B., Bonatto A. Physical Implementation of a 32-bits RISC microprocessor using XFAB 600nm technology // 32^o SIMPOSIO SUL DE MICROELETRONICA. 2017. P. 1–4.
- [2] Andreev D.V., Bondarenko G.G., Andreev V.V., Stolyarov A.A. Use of High-Field Electron Injection into Dielectrics to Enhance Functional Capabilities of Radiation MOS Sensors // Sensors. 2020. V. 20. № 8. P. 2382(1-11). doi: 10.3390/s20082382
- [3] Barriga A. RISC-V processors design: a methodology for cores development // 2020 XXXV Conference on Design of Circuits and Integrated Systems. 2020. P. 1–6. doi: 10.1109/DCIS51330.2020.9268639
- [4] Patterson D., Hennessy J. Computer Organization and Design RISC-V Edition: The Hardware Software Interface. Second Edition. Morgan Kaufmann, 2020.
- [5] Raveendran A., Patil V., Selvakumar D., Desalpine V. A RISC-V instruction set processor-micro-architecture design and analysis // 2016 International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA). 2016. P. 1–7. doi: 10.1109/VLSI-SATA.2016.7593047
- [6] Kornev S.A., Andreev V.V., Stolyarov A.A. Razrabotka specializirovannogo konfiguriruemogo 32-bitnogo mikroprocessora s arhitekturoj RISC-V (Designing a configurable 32-bit microprocessor based on RISC-V architecture for specialized applications) / Sb. trudov « XLVI Akademicheskie chteniya po kosmonavtike, posvyashchennye pamyati akademika S.P. Korolyova i drugih vydayushchihsvya otechestvennyh uchenyh — pionerov osvoeniya kosmicheskogo prostranstva. Korolyovskie chteniya 2022». V. 4. M.: Izdatel'stvo MGTU im. N. E. Baumana. 2022. S. 580–583.
- [7] Waterman A., Lee Y., Patterson D., Asanović K. The RISC-V Instruction Set Manual, Volume I: User-Level ISA, Document Version 20191213. RISC-V Foundation. 2019.
- [8] Weste N., Harris D. CMOS VLSI Design. A Circuits and Systems Perspective 4th. ed. Addison-Wesley, 2015.
- [9] Shetty A. ASIC Design Flow And Methodology – An Overview" SSRG International Journal of Electrical and Electronics Engineering // SSRG International Journal of Electrical and Electronics Engineering. 2019. V. 6. № 7. P. 1–5. doi: 10.14445/23488379/IJEEE-V6I7P101
- [10] Reddi V.J., et al. Voltage Noise in Production Processors // IEEE Micro. 2011. V. 31. № 1. P. 20–28. doi: 10.1109/MM.2010.104
- [11] Kamkin A.S., Protsenko A.S., Smolov S.A., Tatarnikov A.D. MicroTESK-Based Test Program Generator for the RISC-V Architecture // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2018. Issue 2. P. 2-8. doi:10.31114/2078-7707-2018-2-2-8
- [12] Gala N., Karasek M., et al. The RISC-V Architectural Compatibility Test Framework Version 2. RISC-V International, 2021.
- [13] Bora S., Paily R. A High-Performance Core Micro-Architecture Based on RISC-V ISA for Low Power Applications // IEEE Transactions on Circuits and Systems II: Express Briefs. 2021. V. 68. № 6. P. 2132–2136. doi: 10.1109/TCSII.2020.3043204
- [14] Tanenbaum A.S. Structured Computer Organization. 6th Edition. Pearson, 2012.
- [15] URL: <https://www.eembc.org/coremark/scores.php> (access date: 25.08.2022)
- [16] URL: <https://github.com/lowRISC/ibex> (access date: 25.08.2022)