

Процессоры семейства SEAforth

Александр КАЛАЧЕВ
forther@yandex.ru

В данной работе приводится описание семейства многоядерных матричных масштабируемых процессоров SEAforth24 и SEAforth40. Рассмотрена структура как процессоров, так и отдельных ядер процессора, а также механизмы их взаимодействия. Описаны основные возможности ввода/вывода, дано краткое описание системы команд и вычислительных возможностей отдельных ядер и процессора в целом.

Введение

Создатель RISC-процессоров и RAID-массивов Дейв Патерсен выделил следующие аспекты, которые в основном определяют развитие технологий в современной вычислительной технике на данный момент [1]:

- Цена энергии намного выше цены транзистора.
- Системное быстродействие сдерживает память, любые операции обработки выполняются быстрее.
- Естественный параллелизм, команды могут быть длинные и короткие, но выполняются на разных ядрах.

Это мнение в целом отражает основные тенденции развития средств процессорной техники: стремление к увеличению производительности путем совершенствования технологий изготовления, совершенствования внутренней архитектуры, увеличения числа ядер на кристалле. Это касается и встраиваемых систем: промышленных контроллеров, средств индивидуальной связи, сенсорных систем и др. Для встраиваемых систем, помимо требований к производительности, существенно также требование к экономии энергии (так как чаще всего устройство имеет автономное питание). В данном классе систем в целях снижения потребления прибегают, кроме перечисленных мер, к различным схемам управления тактовой частотой, частичному или полному останову процессора и периферии, снижению токов утечки. Также из тенденций можно выделить совершенствование уже выпускаемых серий и появление новых архитектурных решений. Наиболее яркие примеры — внедрение фирмой Atmel технологии picoPower при производстве контроллеров серии AVR и появление новой экономичной высокопроизводительной архитектуры AVR32 этой же фирмы.

Можно также выделить растущую потребность в высокопроизводительных встраиваемых системах (High Performance Embedded Computing, HPEC), возникающую при управ-

лении сложными техническими объектами, обработке потоковой информации, в мультимедийных приложениях, работе в условиях реального времени, в коммуникационных средствах.

Среди экономичных контроллеров, применяемых в аппаратуре, можно выделить семейства picoPower AVR [2], AVR32 [3] от Atmel, семейство низкопотребляющих контроллеров MSP430 [4] от Texas Instruments, процессоры на базе ядра ARM [5, 6] и ряд других. При этом уровень потребления энергии находится в пределах нескольких микроампер/МГц (в среднем 200–500 мкА на частотах 1–20 МГц — для picoPower и MSP).

Один из примеров развития направления многоядерных процессоров: компания Intel в рамках проекта Larrabee разрабатывает процессоры усовершенствованной архитектуры x86 с количеством ядер от 16 до 24 с тактовой частотой 2 ГГц [7]. Также следует отметить многоядерные процессоры семейств Cell, Niagara (UltraSPARC), масштабируемые решения от MIPS — MIPS1004. Как правило, процессоры этих классов предназначены для работы, как в настольных компьютерах, так и в серверных станциях. Процессоры Tile64 [8] от Tilera, содержащие матрицу процессоров 8×8, рассчитаны на использование в высокопроизводительном сетевом оборудовании, а также серверах для потоковой обработки видеоданных. Каждое ядро процессора, выполненное на основе архитектуры RISC, работает на частоте от 600 МГц до 1 ГГц. Внутрикристалльная сеть iMesh позволяет каждому отдельному ядру взаимодействовать с находящимися поблизости ядрами, скорость передачи данных при этом может достигать 27 Тбит/с. Каждое из ядер процессора Tile64 имеет собственную кэш-память первого и второго уровней. Энергопотребление составляет 170–300 мВт в расчете на ядро.

В данной статье будут рассмотрены процессоры SEAforth фирмы Intellasy, являющиеся результатом работы Чарльза Мура, создателя языка Forth.

Процессоры SEAforth [9, 10] позиционируются как многоядерные процессоры для встраиваемых систем. Они построены по технологии встраиваемого масштабируемого массива — Scalable Embedded Array. Процессоры этой серии были анонсированы в конце 2006 года. В апреле 2008 года стали доступны процессоры SEAforth24 и отладочные комплекты на их основе. В октябре того же года была отработана и налажена технология производства 40-ядерных процессоров SEAforth40. На данный момент в семейство входят два типа процессоров — SEAforth24 и SEAforth40 (40C18) с числом ядер 24 и 40 соответственно.

Процессоры SEAforth представляют собой практически многопроцессорную систему с распределенной памятью, размещенную на кристалле (NORMA MMP-SoC). Процессор состоит из множества независимых вычислителей, соединенных по топологии двумерной прямоугольной решетки, имеет ряд последовательных портов, параллельные порты, выводы общего назначения, АЦП и ЦАП. Каждый из вычислителей является стеквым процессором фон Неймана с сокращенным набором команд и имеет собственную оперативную память, порты ввода/вывода и постоянную память с рядом предустановленных фирменных функций. Взаимодействие вычислителей (далее — ядер) осуществляется посредством механизма параллельных взаимодействующих процессов (парадигма ОККАМ). В некотором приближении процессор можно рассматривать как сеть из миниатюрных транспьютероподобных узлов. Глобальная синхронизация ядер отсутствует. Интересной особенностью является то, что ядра асинхронные.

Краткий обзор процессоров SEAforth

Данные процессоры отвечают практически всем современным тенденциям, к которым относятся параллелизм, низкое энерго-

потребление, высокая скорость работы ядра. Есть и моменты, которые ставят их особняком: кроме стековой архитектуры ядра, это отсутствие модуля вещественных вычислений и кэша, если сравнивать с процессорами. Такие процессоры имеют малый объем памяти, небольшое количество периферийных устройств, если проводить сравнение с микроконтроллерами.

Производительность ядра — примерно 700 MIPS (время выполнения инструкции — около 1,4 нс) при уровне потребления мощности ~12 мВт (условно ~17 мкВт на мегагерц). Соответственно, максимальная мощность, рассеиваемая процессором при полной загрузке всех узлов, — не более 250–400 мВт, что сравнимо со многими сигнальными процессорами, но при гораздо большем уровне параллелизма и суммарной вычислительной мощности.

Основные характеристики процессоров семейства представлены в таблице 1.

Имея в своем распоряжении несколько десятков ядер, программист может выделять из них некоторые группы. Для типовых задач возможно создание так называемых параметризованных макроблоков из нескольких ядер. В результате получается гибкая мультипроцессорная система, предназначенная для решения широкого круга прикладных задач.

Поскольку объем памяти для размещения программ мал, типичная практика в программировании процессора — это замена исполнимого кода во время выполнения программы [11]. Как правило, каждое ядро исполняет относительно небольшую функцию или блок приложения, что характерно для большого количества задач управления или обработки потоков данных. В качестве примера можно привести реализацию QPSK приемника и передатчика на SEAForth24 [12].

Основные применения процессов SEAForth — производственные контролирующие и сенсорные системы, контроллеры сенсорных сетей [13], мультимедийные системы. Причины, обуславливающие это:

- возможность параллельной обработки данных в реальном времени;
- возможность распределенного управления промышленной сетью.

Немаловажен и ценовой фактор.

К процессору в системе сбора данных могут быть подключены датчики практически с любым интерфейсом (при условии согласования уровней напряжения) и различными протоколами обмена. Это достигается за счет программного управления линиями ввода/вывода, таким образом, разные ядра процессора имеют возможность взаимодействовать с датчиками по различным протоколам. Упрощается также замена датчиков: при необходимости вносятся коррективы только в программное обеспечение.

Кроме приема данных с линий ввода/вывода, процессор может быть подключен непосредственно к сенсору, что уменьшает ко-

Таблица 1. Характеристики процессоров семейства SEAForth

Характеристика	SEAForth24	SEAForth40
Количество ядер, шт.	24	40
Производительность ядра/общая, MIPS	700, суммарная — 18 TIPS	700, суммарная — 26 TIPS
Линии ввода/вывода (GPIO)	5	5
АЦП, количество/разрядность, биты	2/18 (характеристика нелинейная)	3/18 (характеристика нелинейная)
ЦАП, количество/разрядность, биты	2/6 (токовый выход, номинальная нагрузка 75 Ом)	3/9 (токовый выход, номинальная нагрузка 75 Ом)
Последовательные порты	SPI, 3 асинхронных последовательных порта, 3 синхронных порта	SPI, 3 синхронных порта, 2 асинхронных последовательных порта, 2 — SERDES
Параллельные порты / разрядность	2/(14 бит выходной порт, 18 бит — двунаправленный порт)	2/18 бит, двунаправленные
Напряжение питания, В	1,8	1,8
Корпус	BGA81 (9×9)	QFN88

личество используемых микросхем и упрощает схемотехнику плат.

Помимо непосредственно сбора данных, параллельно процессор может вести их обработку в реальном времени, включая такие вычислительно емкие задачи, как цифровая фильтрация, быстрые преобразования (Фурье, Уолша, Хартли и др.), анализ аудио-, видео- или радиосигналов, PID-регулирование, ФАПЧ.

Линии ввода/вывода могут быть необходимы для программной реализации различного рода интерфейсов, помимо ограниченной поддержки основных последовательных интерфейсов в ПЗУ некоторых ядер. Последовательные порты могут быть использованы для соединения нескольких процессоров.

Программное управление линиями ввода/вывода позволяет поддерживать несколько форматов цифровых сигналов и видов модуляций аналоговых сигналов, в отличие от специализированных СБИС, поддерживающих 1–2 стандарта.

Интерфейсы и протоколы, реализуемые программно, могут варьироваться от простых протоколов, аналогичных RS-232 и SPI, до достаточно сложных алгоритмов кодирования типа QPSK или QAM. При этом максимальные скорости обмена данными лежат в диапазоне 10–20 Мбит/с [12].

Программное обеспечение включает в себя компилятор-симулятор языка VentureForth [14], имеющий версии под операционные системы Windows и Linux. Симулятор позволяет производить отладку программ, отслеживать состояние регистров и памяти любого из ядер процессора. При отладке программ одновременно отображается состояние всех ядер процессора (основные регистры ядра), выделяются ядра, находящиеся в активном состоянии. Есть возможность одновременного отображения подробной информации об 1, 2 или 4 ядрах процессора, а также просмотра содержимого памяти. Симулятор поддерживает работу с отладочными комплектами: ForthDrive для SEAForth24 [15] и SEK40C18 для SEAForth40 [16].

Компилятор/симулятор VentureForth представляет собой программу, написанную на языке Forth, и ее исходные тексты (для систем SwiftForth и Gforth) доступны со средой разработки. Таким образом, пользователь мо-

жет, в случае необходимости, вносить изменения в код симулятора, тем самым моделируя различные внешние устройства, протоколы и ситуации.

В процессорах SEAForth не используется механизм прерываний. В случае необходимости ядро переходит в пассивный режим, ожидая записи в коммуникационный порт или чтения. При этом время перехода из пассивного в активный режим и наоборот крайне мало — <10 нс. Благодаря наличию нескольких периферийных ядер, процессор может одновременно обслуживать несколько источников событий с минимальной задержкой, что идеально для систем, работающих в режиме жесткого реального времени. Поскольку ядра асинхронные, исполнение программы происходит с максимальной возможной скоростью. Процессор не нуждается в режимах пониженной частоты, характерных для обычных микроконтроллеров, для уменьшения энергопотребления.

В процессорах применен ряд технологий, позволяющих добиться значительной экономии энергии. Например, ядро, первым подошедшее к точке передачи данных, автоматически переходит в спящий (пассивный) режим до тех пор, пока взаимодействующее с ним ядро не подойдет к этой же точке. При этом время выхода из пассивного режима составляет примерно 10 нс. В спящем режиме ядро рассеивает порядка 1 мкВт, плюс обеспечивается прозрачность для прикладного программиста синхронизация процессов, выполняющихся на различных ядрах. Как правило, во время работы приложения одновременно активны 8–16 ядер. Вторая особенность — во внутреннем представлении данных. На физическом уровне в ядре в соседних битах одинаковые логические уровни представлены противоположными уровнями напряжений. Например, если в регистре находится число 100010b, то физические уровни напряжений на его выходах будут следующие: НННВВВ. И третий момент: отсутствие глобальной синхронизации ядер и асинхронное исполнение самого ядра (отсутствуют тактовые генераторы). Низкое энергопотребление и высокая энергетическая эффективность — одни из ключевых свойств процессоров SEAForth.

Проведем сравнение процессоров SEAForth с наиболее распространенными целочислен-

Таблица 2. Сравнительные характеристики процессоров различных архитектур

Характеристика	AVR (picoPower)	AVR32	MSP430	ARM*	SEAForth ядро C18 (суммарно по 24/40 ядрам)	TILE64 (суммарно по 64 ядрам)
Разрядность	8	32	16	32	18	32
MIPS	20	72–210	8	50–150	700 (18 000/26 000)	700–866 (443 000)
Потребляемая мощность (максимальная), мВт	13,3	7,6	4,9	5,8–48,75	12,6 (302/504)	343 (22 000)
Потребление в пассивном режиме (энергосберегающем), мкВт	0,06/0,9	9,9	0,22/1,76/70	20–1000	1 (24/40)	
Время перехода в активное состояние, мкс			6	0,16–60	<0,01	
Затраты энергии на выполнение операций (средние значения), нДж						
Логические	0,67	0,127	1,8	0,2–3,5	0,018 (0,42/0,7)	0,52 (32)
Арифметические	0,67	0,127	2	0,2–3,5	0,036 (0,84/1,4)	0,52 (32)
Умножение	1,33	0,127	1,25	0,2–3,5	1,15** (27,5/46)	0,52 (32)
Операции с памятью	1,33	0,19	2,4	0,2–3,5	0,064 (1,54/2,56)	0,52 (32)

* — средние по семействам; ** — программная реализация умножения.

ными процессорами различных архитектур [2–6, 8–10, 13, 14] (табл. 2).

Судя по таблице 2, SEAForth выигрывает по показателям энергоэффективности в активном режиме и производительности, как по отдельным ядрам, так и по суммарной. Большим преимуществом является очень малое время реакции на событие и высокая скорость выдачи данных на внешние выходы — до 90 МГц. Наиболее близки по этим показателям процессорные ядра AVR32 и ARM Cortex-M3 [5, 6, 17, 18]. Процессор проигрывает в операциях типа умножение, умножение с накоплением, деление, поскольку они в нем реализованы программными средствами. Также может несколько снизить производительность ограниченный набор команд и необходимость динамической замены кода во время работы приложений. К недостаткам можно отнести малый размер памяти, небольшой набор периферийных

устройств, необходимость подключения внешней памяти для хранения пользовательских программ.

Рассмотрим особенности реализации каждого из процессоров. Помимо количества ядер, они отличаются набором линий ввода/вывода и периферийных устройств, реализацией АЛУ (затрагивает операции +, +*), составом фирменных функций, прошитых в ПЗУ ядер на стадии производства.

Процессор SEAForth24 (ar24c18)

Этот процессор имеет в своем составе 24 ядра, объединенных в решетку 4×6. Особенности АЛУ: операция сложения производится между регистрами T и S, запоминание возникающего при переполнении переноса не происходит. Операция +* производится между вершиной и вторым элементом стека (между регистрами T и S).

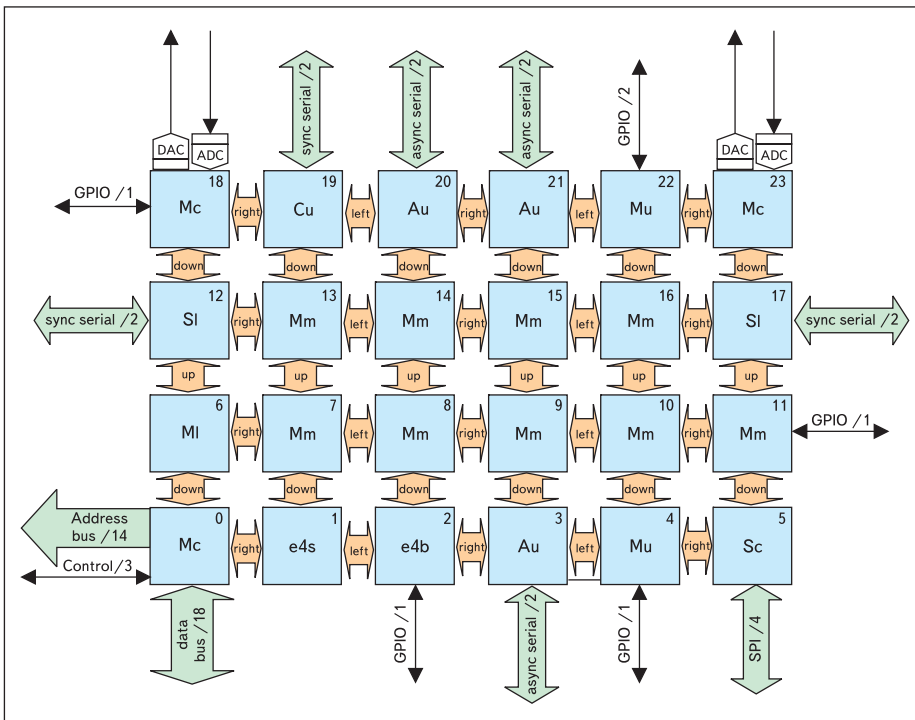


Рис. 1. Структурная схема процессора SEAForth24

Основные функции, размещенные в ПЗУ процессора SEAForth24, — это ожидание записи/чтения в коммуникационный порт, умножение, умножение с накоплением, поддержка SPI, синхронного и асинхронного ввода/вывода (включая слова приема/выдачи данных, слова исполнения кода с последовательных портов, начальной загрузки) [14]. Фирменные функции, размещенные в ПЗУ ядра, позволяют экономить пространство ОЗУ при создании прикладных программ за счет вызова функций, что отчасти освобождает пользователя от создания подпрограмм, реализующих типовые действия. Условно можно выделить следующие наборы слов:

- Mult — умножение знаковых/беззнаковых чисел. Имеет несколько точек входа, реализующих операцию умножения с накоплением — MAC.
- Pause (cornerpause, centerpause, uppause, leftpause) — слова ожидания записи в один из коммуникационных портов (проверка статуса порта).
- Cold, warm — ожидание записи в порт и исполнение инструкций.
- SPI — слова начальной загрузки с последовательной Flash-памяти, набор примитивов чтения/записи данных в порт, управление выводами.
- Serial — слова начальной загрузки с асинхронного порта, чтения потока данных и одиночных данных с порта.
- Sget — слова начальной загрузки с синхронного порта, чтения потока данных и одиночных данных с порта.

Структурная схема процессора и распределение фирменных функций (табл. 3) по ПЗУ ядер представлены на рис. 1.

Таблица 3. Слова и наборы слов ПЗУ процессоров SEAForth

Обозначение	Слова и наборы слов ПЗУ SEAForth24/SEAForth40
Mc	mult, warm, cornerpause, !pump, @pump
	relay, cornerwarm, poly, mlt, fmlt, taps, interp, triangle, -u/mod
Mm	mult, warm, centerpause, !pump, @pump
	relay, centerwarm, poly, mlt, fmlt, taps, interp, triangle, -u/mod
Mu	cold, mult, warm, uppause, !pump @pump
	relay, upwarm, poly, mlt, fmlt, taps, interp, triangle, -u/mod
Ml	mult, warm, leftpause, !pump, @pump
	relay, leftwarm, poly, mlt, fmlt, taps, interp, triangle, -u/mod
Au	cold, warm, serial, uppause
	relay, upwarm, poly, mlt, fmlt, -dac, interp, triangle, -u/mod
Ac	—
	relay, cornerwarm, poly, mlt, fmlt, -dac, interp, triangle, -u/mod
Ul	—
Fu	relay, leftwarm, poly, mlt, fmlt, taps, interp, triangle, bget
	—
Uu	—
	relay, upwarm, serial, lshift, rshift
Sl	cold, warm, sget, leftpause, !pump, @pump, smult
	relay, leftwarm, sget, mlt, taps, triangle
Cu	cold, warm, sget, uppause, !pump, @pump, smult
	relay, upwarm, upserdes, mlt, fmlt, taps, interp, triangle, -u/mod
e4s	—
	—
e4b	—
	—
e4c	—
	—
e4d	—
	—
e4e	—
	—
e4f	—
	—
e4g	—
	—
e4h	—
	—
e4i	—
	—
e4j	—
	—
e4k	—
	—
e4l	—
	—
e4m	—
	—
e4n	—
	—
e4o	—
	—
e4p	—
	—
e4q	—
	—
e4r	—
	—
e4s	—
	—
e4t	—
	—
e4u	—
	—
e4v	—
	—
e4w	—
	—
e4x	—
	—
e4y	—
	—
e4z	—
	—

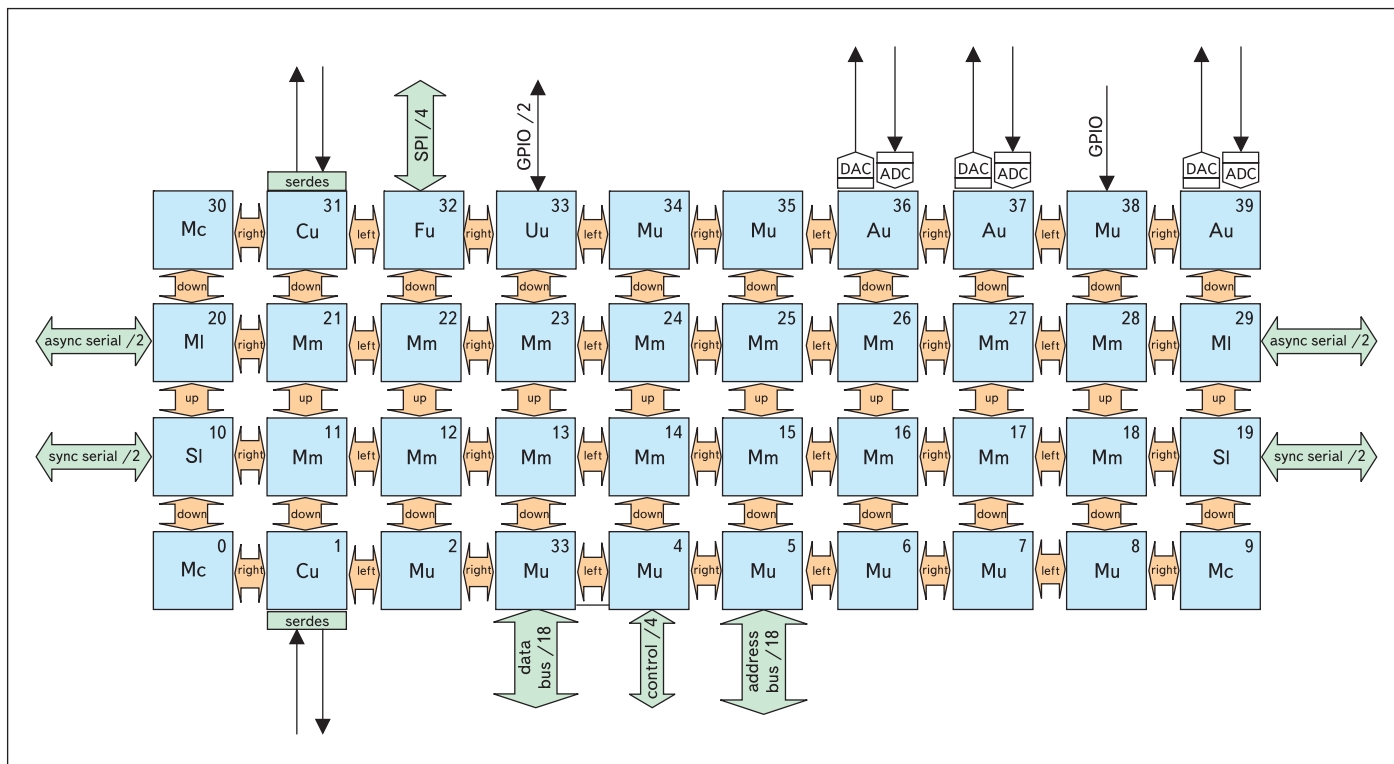


Рис. 2. Структурная схема процессора SEAForth40

Процессор SEAForth40 (S40C18)

Он содержит 40 ядер, объединенных в решетку 4×10. По сравнению с предыдущей моделью в его ядра C18 внесены изменения, существенно упрощающие организацию вычислительного процесса. Был добавлен так называемый режим расширенной арифметики, при котором перенос сохраняется в 10-м бите регистра P (данный бит в адресации памяти не участвует), таким образом, появляется возможность производить операции над многоразрядными числами. Вторая особенность АЛУ — операция +* производится между вторым элементом стека и регистром A. Старшее слово результата помещается на вершину стека, младшее остается в регистре A, второй элемент стека (регистр S) остается без изменений. Два ядра процессора имеют высокоскоростные интерфейсы SERDES, отсутствовавшие в SEAForth24.

Процессор SEAForth40 имеет достаточно большой объем математических библиотек и библиотек ввода/вывода, размещенных в ПЗУ ядер и позволяющих пользователю строить на их основе свои алгоритмы обработки и передачи данных.

Математическая библиотека содержит наборы функции умножения: mlt, fmult, для операций с целыми числами и с числами в формате с фиксированной точкой, деления с остатком um/mod (имеет несколько точек входа), линейной интерполяции — interp, работы с векторами — rotation, вычисление полиномов — poly, вычисления треуголярной

функции — triangle, taps — слова для организации расчета цифровых фильтров.

Библиотека ввода/вывода представлена наборами слов для последовательных интерфейсов. SPI — слова начальной загрузки с Flash-памяти, чтения последовательной памяти. Serial — чтение данных с последовательного порта. Sget — работа с синхронным последовательным портом. Vget — чтение данных с асинхронного последовательного порта. Upserdes — поддержка работы с высокоскоростными последовательными портами SERDES (загрузка и исполнение кода).

Структурная схема и распределение функций (табл. 3) по ПЗУ ядер представлена на рис. 2.

Аналоговые функции: набор слов dac — вывод импульсов напряжения с определенными параметрами.

Практически во всех ядрах присутствуют наборы слов для реализации системных функций. Switch — складывает значение на вершине стека возвратов с вершиной стека данных и результат помещает на стек возвратов. Warm (corner warm, up warm, left warm, corner cage) — переводит ядро в режим ожидания записи в один из коммуникационных портов. Pause (corner pause, left pause, up pause, center pause) — проверяет статус коммуникационных портов, если данные есть, они трактуются как команда (ядро начинает исполнять инструкции, входящие на порт). По завершении исполнения продолжается выполнение основной программы. Relay — продвижение пакета исполнимого кода через последовательность ядер.

Структура ядра C18

Архитектура ядра. Каждое ядро, входящее в состав процессоров SEAForth, содержит 18-разрядный микропроцессор C18, 64 18-разрядных слова ОЗУ, 64 18-разрядных слова ПЗУ, содержащего заранее прошитые функции (так называемый Intellasy-bios), и 4 порта ввода/вывода. Некоторые узлы, находящиеся по краям решетки, имеют дополнительно последовательный или параллельный порт, АЦП или ЦАП. Функции, прошитые в ПЗУ отдельных ядер, также несколько отличаются.

Помимо ОЗУ и ПЗУ он содержит два стека — данных и возвратов, с выделенными регистрами вершин стеков, и логикой управления, регистр слова-инструкции, программный счетчик, АЛУ, декодер команд и логику выборки команд, а также два индексных регистра. Полный список регистров приведен в таблице 4.

Таблица 4. Регистры C18

PC	9-битный программный счетчик (10-битный для процессора 40с18)
T, S	Первый и второй элемент стека данных
R	Верхний регистр стека возвратов. Один из 9 регистров стека возвратов, доступен через push/pop, call/return
A	18-битный регистр общего назначения, адресный, автоинкрементный
B	9-битный адресный регистр
IW	18-битное слово инструкции
RIGH, DOWN, LEFT, UP	Коммуникационные регистры (являются общими с соседними ядрами)
IOCS	Регистр режима выводов и статуса ввода/вывода
DATA	Внешняя шина данных
ADDRESS	Адресный регистр внешней памяти

Стеки в С18 — массивы регистров. Стек данных нужен для выполнения арифметических операций. Два наиболее часто используемых регистра — Т и S — могут быть доступны непосредственно. Оба они подключены к АЛУ, результат операций помещается в регистр Т. Ниже находится циклический массив из 8 регистров. Один из этих регистров в каждый момент времени выбирается как регистр, следующий за S. АЛУ, использующее регистры Т и S, вычисляет все возможные арифметические операции параллельно, используя комбинаторную логику. Текущей командой выбирается необходимый результат из АЛУ и помещается в Т.

Структура микропроцессора С18 представлена на рис. 3.

С18 имеет девять 18-битных регистров стека возвратов. Верхняя позиция стека возвратов находится в регистре R. Под R также находится циклический массив из 8 регистров. Инструкции вызова составляют значение программного счетчика на стеке возвратов. Команды возврата снимают только 9 младших разрядов. Стек возвратов также может быть использован для временно-

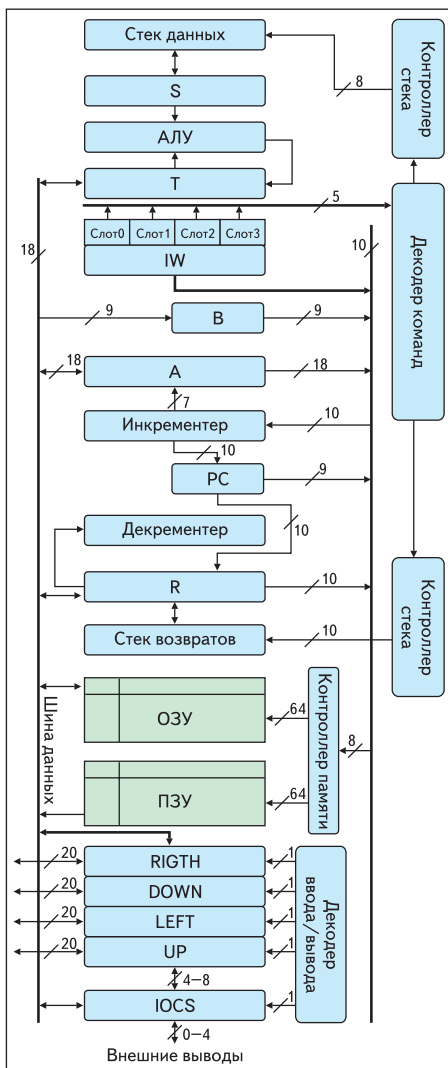


Рис. 3. Структура процессорного ядра С18

го хранения данных и в качестве счетчика циклов.

Аппаратный контроль за переполнением и исчерпанием стеков не предусмотрен. Поскольку регистры в стеке связаны в кольцо, они не могут переполниться или исчерпаться, они просто «прокручиваются». Так как глубина стека ограничена, добавление элемента на стек означает затирание самого нижнего элемента. Когда идет извлечение из стека, нижние 8 элементов будут повторяться. После двух чтений Т и S будут содержаться копии двух элементов массива стековых регистров.

Выходы программного счетчика РС управляют адресной шиной (подаются на шину адреса), программный счетчик по выполнении инструкции инкрементируется. Данные, адресуемые РС, загружаются в регистр инструкции IW.

Инструкции имеют длину всего лишь 5 бит, что позволяет упаковывать три или четыре инструкции в одно 18-битное слово. 18-битное слово содержит до 4 опкодов, выбираемых слот-селектором, и передается на декодер и логику контроля, управляющую функциями ядра. Восемь из 5-битовых инструкций могут быть помещены в 3-битный слот как последний код операции в слове. Таким образом, максимально в оперативной памяти С18 можно разместить 256 команд, что с учетом высокой рентабельности форт-кода достаточно для реализации многих алгоритмов и прикладных программ.

Типичная последовательность выполнения кода начинается со значения, загруженного в программный счетчик и выставленного на шину адреса. Это значение используется двумя способами: выбирает адрес в памяти (в некоторых случаях — порты ввода/вывода) и управляет «инкрементером», который записывает увеличенное значение обратно в счетчик. Выбранное значение подается на шину данных, значение шины фиксируется в регистре IW. Значение в первом слоте подается на блок логики контроля и декодирования, который управляет работой элементов ядра. Также блок имеет логику «предсказания»: если обнаруживается, что ни один из опкодов не использует шину адреса, разрешается выдача значения РС на шину адреса для предварительной выборки следующей команды.

Структура адресного пространства. Карта распределения адресного пространства микропроцессора С18 выглядит следующим образом (рис. 4). Оперативная, постоянная память, специальные регистры и порты ввода/вывода находятся в одном адресном пространстве, и для работы с ними применяются одни и те же команды.

При выполнении программы логика берет значение с адресной шины и вычисляет новый адрес параллельно со временем доступа шины. Результат становится доступным для

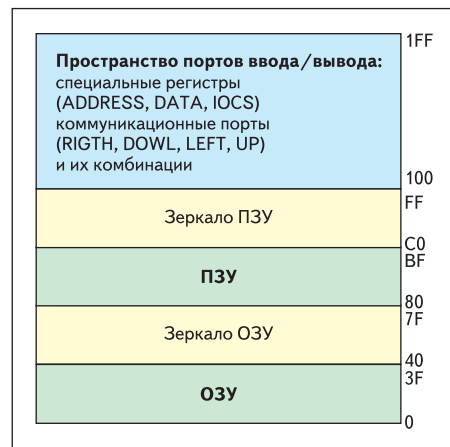


Рис. 4. Карта памяти С18

РС или для регистра А. Использование инкрементированного адреса определяется текущей инструкцией. При нормальном потоке выполнения результат записывается в РС. В пределах границы 128 слов инкрементированный адрес может прокрутиться к началу страницы, следовательно, RAM и ROM появляются в двух различных диапазонах.

Когда бит 8 адреса равен 1, идет адресация в пространстве ввода/вывода и увеличение адреса запрещено, то есть когда регистр указывает на порт, указатель не сдвигается. Это значит, что выборка инструкций, литералов, запись литералов выполняются непосредственно с порта. Вызов с порта возвращает на порт.

Ввод/вывод в SEAForth программно доступен по трем путям:

- цифровые выходы, доступные через специальные регистры или регистры IOCS;
- межпроцессорные коммуникации, выполняемые при помощи направленных портов;
- аналоговый ввод/вывод, доступный через специальные регистры или регистры IOCS.

Каждое ядро С18 разделяет до четырех портов ввода/вывода со своими соседями. В общем случае межпроцессорные коммуникации являются блокирующими и самосинхронизирующимися, это значит, что процессор уходит в спящее состояние, пока операция не будет завершена.

Каждый межпроцессорный коммуникационный порт соединен напрямую со своими соседями: соседние узлы разделяют один порт. Общий порт имеет для соседних процессоров один и тот же адрес. Нет регистров или FIFO-буфера: одни линии порта напрямую соединены с соседними линиями записи. Значение, записанное в порт, может иметь различные интерпретации: узел, проводящий чтение, может исполнить считанный код. Когда процессор проводит операцию чтения, он блокирует запись соседнего процессора, когда процессор пишет, он блокирует операцию записи соседа. Подобный метод синхронизирует соседние процессоры. Блокировка — ключевой элемент в уменьшении потребляемой процессором мощности и средство син-

Таблица 5. Список команд процессорного ядра C18

Имя	Аннотация	Описание
call	R: -- a	Вызов подпрограммы
;	R: a --	Возврат из подпрограммы
jump		Безусловный переход
::	R: a1 -- a2; p: a2 -- a1	Передача управления подпрограмме (примитив для реализации многозадачности)
if	D: x -- x	Переход, если T = 0
-if	D: x -- x	Переход, если число в T отрицательное (старший бит = 1)
next	R: n -- n-1; R: 0 --	Если n <> 0, переход по адресу, указанному в поле перехода
unext	R: n -- n-1; R: 0 --	Аналогично next, но в пределах одного слова инструкции
@a	D: -- x	T = (A). Помещает на стек значение, по адресу в регистре A
!a	D: x --	(A) = T. Заносит в память по адресу, указанному в A, значение с вершины стека
@a+	D: --x; A = A+1	Помещает на стек значение по адресу в регистре A, содержимое регистра инкрементируется
!a+	D: x --; A = A+1	Заносит в память по адресу, указанному в A, значение с вершины стека, содержимое регистра инкрементируется
@b	D: -- x	Помещает на стек значение по адресу в регистре B
!b	D: x --	Заносит в память по адресу, указанному в B, значение с вершины стека, содержимое регистра инкрементируется
!p+	D: x --	Значение с вершины стека записывается по адресу, указанному в p
@p+	D: -- x	Помещает на стек число с адреса, указанного в p
.		Пустая операция
push	D: x -- ; R: -- x	Переносит значение со стека данных на стек возвратов
pop	R: x -- ; D: -- x	Переносит значение со стека возвратов на стек данных
dup	D: x -- x x	Дублирует вершину стека
drop	D: x --	Удаляет значение с вершины стека
over	D: x1 x2 -- x1 x2 x1	Дублирует второй элемент стека на вершину
al	D: x -- ; A = x	Переносит значение с вершины стека в регистр a
a@	a=x; D: -- x	Копирует значение регистра a на вершину стека
bl	D: x -- ; b=x	Переносит значение с вершины стека в регистр b
not	D: x -- not(x)	Инвертирует вершину стека
and	D: x1 x2 -- (x1 and x2)	На вершину стека помещается логическое И регистров T и S
xor	D: x1 x2 -- (x1 xor x2)	На вершину стека помещается побитовое исключающее ИЛИ регистров T и S
2/	D: x -- x/2	Деление вершины стека на 2
2*	D: x -- x*2	Умножение вершины стека на 2
+	D: x1 x2 -- x1+x2	Суммирует вершину и второй элемент стека
+*		Шаг умножения. Выполнение и размещение операндов для SEAForth24 и SEAForth40

хронизации ядер (процессов). В заблокированном состоянии ядро практически не потребляет мощности. Только один из процессоров блокируется во время транзакции: тот, что входит в нее первым. Блокировки можно избежать, считывая бит статуса в регистре IOCS перед операцией чтения или записи. Возможны операции множественного чтения/записи в порт. При наличии готового к транзакции соседнего узла операции чтения/записи занимают порядка 4,2 нс.

Регистры IOCS предназначены как для ввода/вывода, так и для мониторинга межпроцессорного обмена. Они являются комбинацией элементов хранения бит статуса, показывающих, произошли ли операции чтения или записи (с той или другой стороны). Также они содержат значение, выводимое или считанное с линии ввода/вывода. Режим и выходное состояние линии устанавливается записью

в IOCS. Типовое время доступа к регистрам IOCS — 2,8 нс.

Система команд C18 состоит из 32 базовых инструкций, составляющих язык VentureForth. VentureForth имеет все достоинства языка Forth: это экономичность, простота и расширяемость. IntellaSys расширяет возможности VentureForth добавлением поддержки Forthlet — объектов, которые могут распространяться между ядрами. Список команд с их кратким описанием представлен в таблице 5.

Пользовательский код располагается в RAM каждого ядра. Для этого существует несколько путей. Типичным является загрузка из Flash-памяти через SPI-порт. Одна Flash-память может загружать несколько соединенных процессоров SEAForth.

Для загрузки литералов (констант/чисел), вызовы, переходы, обращения к памяти и портам необходимы операнды. Команда перехода или вызова может иметь 3-, 8- или 9-битный аргумент. Инструкции-литералы используют 5-битный опкод и 18-битное слово — литерал, который будет помещен на стек.

Периферийные устройства. Некоторые ядра, находящиеся по краям решетки, имеют дополнительные устройства в виде параллельных или последовательных портов, аналого-цифрового преобразователя (АЦП), цифро-аналогового преобразователя (ЦАП) и, соответственно, определенное количество внешних выводов.

Параллельные порты доступны для чтения/записи, либо как коммуникационные порты ядра (как правило, UP-порт), при этом операции с ними являются блокирующими, или при помощи специальных регистров DATA, ADDRESS, в этом случае блокировки не происходят. Параллельные порты могут быть использованы как для подключения внешней памяти, так и как порты ввода/вывода общего назначения. Подключение внешней памяти будет иметь свою специфику для каждого из процессоров семейства: в SEAForth24 параллельные порты принадлежат одному ядру, в SEAForth40 — двум, и третий управляет контролирующими линиями. Время доступа к параллельным портам — менее 5 нс.

По умолчанию ядра, имеющие два вывода, считаются ядрами с последовательными портами, поскольку есть возможность организовать полнодуплексный режим работы. Ядра с одним выводом относят к портам общего назначения. Ядра с синхронными и асинхронными последовательными портами отличаются только поддерживающими кодами программ в ПЗУ, никаких реальных отличий в устройстве ядер нет.

Асинхронные порты функционируют как универсальные асинхронные приемопередатчики (UART) и служат для подключения внешних устройств или других процессоров SEAForth. Код в ПЗУ позволяет загружать устройство через асинхронный порт и дает возможность ядру выходить из режима покоя при поступлении стартового бита. Отли-

чия ядер с синхронными интерфейсами в том, что они используют отдельную линию как сигнал синхронизации.

SPI-порты реализуются четырьмя линиями ввода/вывода и также имеют программную поддержку в ПЗУ. Код в ПЗУ предусматривает возможность загрузки исполнимого кода из последовательной Flash-памяти: высокий уровень на линии spi-in начинает процесс загрузки. В процессе загрузки SPI работает на скорости 150 кбит/с, позволяя использовать относительно дешевые устройства памяти. По окончании загрузки интерфейс может функционировать со скоростью порядка 10 Мбит/с.

Ядра с одиночным выводом могут использовать его как входной либо как выходной одноканальный порт. Данный порт доступен и конфигурируется через регистр IOCS. Время перехода вывода из состояния с высоким импедансом в активное составляет порядка 110–115 нс.

АЦП, имеющиеся на некоторых ядрах, представляют собой генераторы, управляемые напряжением (ГУН), связанные с 18-разрядными счетчиками. Аналого-цифровое преобразование осуществляется посредством двух последовательных процедур чтения DATA регистра и вычисления скорости работы генератора. Время полного счета для входного напряжения 400 мВ — 40 мкс, для 1500 мВ — 75 мкс. Выделенными битами в регистре IOCS ядра счетчик может быть запущен или остановлен, на ГУН подано напряжение с внешнего вывода, или напряжения ноля, или питания для калибровки. Характеристика «напряжение – код» ($V_{in} - ADCcount$) АЦП является нелинейной. Зависимость $V_{in}(ADCcount)$ можно с некоторой точностью рассматривать как кубическую вида:

$$V_{in}(ADCcount) = a_0(ADCcount - a_1)^3 + a_2.$$

При работе АЦП потребляет не более 4,5 мВт. Время, затрачиваемое на чтение данных из АЦП, примерно 5–5,1 нс.

ЦАП реализован как набор двоично-взвешенных источников тока и рассчитан на номинальную нагрузку 75 Ом при токе 17 мА. Программно доступен через регистр IOCS.

SERDES — специализированные параллельно-последовательные регистры, предназначенные для осуществления коммуникаций между процессорами. Могут передавать 18-разрядные слова (которые в свою очередь могут быть как данными, так и инструкциями), могут служить генераторами функций. Имеют две двунаправленные линии: данных и тактовую. Направление устанавливается в регистре IOCS ядра, данные записываются по адресу DATA, для контроля прихода/отправки слова используется адрес UP. Тактирование осуществляется от специального осциллятора. Передача идет на скорости порядка 400 Мбит/с, на передачу одного слова требуется 19 тактов.

Жесткой аппаратной специализации периферийных ядер, за исключением ядер с АЦП, ЦАП и SERDES, нет. При помощи программного кода, загружаемого в оперативную память, можно эмулировать практически любой последовательный или параллельный интерфейс.

Заключение

В целом, процессоры SEAForth — достаточно мощное решение для встраиваемых систем реального времени. Это достигается посредством высокой скорости исполнения команд и высокой степени параллелизма, недоступной многим другим процессорам. Неоспоримым плюсом является низкое энергопотребление, особенно в пересчете на количество операций в секунду. Система команд доведена до определенного минимума, по функциональности и удобству ее можно сравнить с системой RISC-контроллеров, таких как PIC16xx, z86, z89. Набор периферийных устройств, по сравнению с современными контроллерами, несомненно мал, сводится в основном к устройствам передачи данных и преобразователям сигнала. Упор сделан на программную эмуляцию необходимой периферии.

Области применения — встраиваемые приложения реального времени: медиаустройства, бортовая электроника, измерительная аппаратура, управление технологическими про-

цессами. Также интересны возможности данных процессоров для создания карманных компьютеров и коммуникаторов, реализации нейронных сетей, систем с перестраиваемой архитектурой. ■

Литература

1. Черняк Л. Архитектура фон Неймана, реконфигурируемые компьютерные системы и антимашина // Открытые системы. 2008. № 6.
2. Технология picoPower для 8-разрядных RISC-микроконтроллеров AVR. http://www.gaw.ru/html.cgi/txt/doc/micros/avr/pico_power/start.htm
3. 32-разрядные микроконтроллеры / ЦПОС семейства AVR32. http://www.gaw.ru/html.cgi/txt/ic/Atmel/micros/avr_32/start.htm
4. Семейство микроконтроллеров MSP430 Texas Instruments. http://www.gaw.ru/html.cgi/txt/ic/Texas_Instruments/micros/msp430/start.htm
5. 32-разрядные высокопроизводительные RISC-процессоры семейства ARM. <http://www.gaw.ru/html.cgi/txt/doc/micros/arm/arh/index.htm>
6. Пономарев В. Новые микроконтроллеры фирмы STMicroelectronics на базе ядра ARM Cortex-M3 // Электроника: Наука, Технология, Бизнес. 2007. № 6.
7. Дудкин А. Новые многоядерные процессоры Intel в 2008–2009 годах // IT-дайджест «Русские документы». 2008. Март.
8. TILE64Processor Family. <http://www.tilera.com/products/tile64.php>

9. IntellaSys — SEAForth-24. http://www.intellasy.net/index.php?option=com_content&task=view&id=35&Itemid=63
10. IntellaSys — SEAForth 40C18. http://www.intellasy.net/index.php?option=com_content&task=view&id=60&Itemid=75
11. Pelc S. Extreme Forth. <http://www.ddj.com/hpc-high-performance-computing/210603608;jsessionid=HWJRZLEQ3HPZ4QSNDLRSKH0CJUNN2JVN?pgno=2>
12. Snively L. O. RF Processing Using SEAForth. http://www.intellasy.net/index.php?option=com_content&task=view&id=24&Itemid=43
13. Snively L. O. SEAForth In Industrial Control and Sensing Applications. http://www.intellasy.net/index.php?option=com_content&task=view&id=24&Itemid=43
14. New VentureForth Programmers Guide. http://www.intellasy.net/index.php?option=com_content&task=view&id=57&Itemid=68
15. S24 FORTHdrive Data Sheet 1.0. http://www.intellasy.net/index.php?option=com_content&task=view&id=57&Itemid=68
16. SEK 40C18 DataSheet 1.1. http://www.intellasy.net/index.php?option=com_content&task=view&id=57&Itemid=68
17. Сидоренко Б. AVR32 — микроконтроллеры для применений XXI столетия / Chip News Украина. 2008. № 8.
18. Сравнительный анализ микроконтроллеров с ядром ARM. <http://www.gaw.ru/html.cgi/txt/pub/micros/arm.htm>