

# ОГЛАВЛЕНИЕ

Предисловие автора .....	9
<b>Часть 1. Справочная информация по семейству LPC2000 .....</b>	<b>13</b>
<b>Глава 1. Общее описание микроконтроллеров LPC2000 .....</b>	<b>14</b>
1.1. Характерные особенности микроконтроллеров семейства LPC2000 .....	14
1.2. Приложения .....	15
1.3. Особенности архитектуры .....	18
1.4. Процессорное ядро ARM7TDMI-S .....	18
1.4.1. Режимы работы и регистровые модели процессора в состоянии ARM .....	21
1.4.2. Способы адресации и система команд ARM .....	24
1.4.3. Регистровые модели процессора в состоянии THUMB .....	34
1.4.4. Система команд THUMB .....	35
1.4.5. Обработка исключительных ситуаций .....	39
1.4.6. Время реакции на прерывания .....	43
1.5. Flash-память на кристалле .....	44
1.6. Статическое ОЗУ на кристалле .....	44
1.7. Специальные регистры микроконтроллера .....	44
1.8. Блок-схемы микроконтроллеров LPC2000 .....	45
<b>Глава 2. Организация системной памяти .....</b>	<b>54</b>
2.1. Массив системной памяти .....	54
2.2. Карта памяти и операционные режимы .....	59
2.3. Переотображение памяти .....	61
2.4. Аварийное прекращение работы выборки данных .....	62
<b>Глава 3. Контроллер внешней памяти (EMC) .....</b>	<b>64</b>
3.1. Функциональные характеристики .....	64
3.2. Описание .....	64
3.3. Регистры EMC .....	65
3.4. Интерфейс внешней памяти .....	67
3.5. Типичные шинные последовательности сигналов .....	69
3.6. Выбор внешней памяти .....	70
<b>Глава 4. Блок управления системой .....</b>	<b>71</b>
4.1. Основные функции .....	71
4.2. Линии интерфейса управления .....	71
4.3. Регистры интерфейса управления .....	72
4.4. Стабилизированный кварцевый генератор .....	73
4.4.1. Алгоритм выбора тактовой частоты .....	74
4.5. Входы внешних прерываний .....	74
4.5.1. Описание регистров .....	75
4.5.2. Множества выводов внешних прерываний .....	79
4.6. Управление памятью .....	80
4.6.1. Рекомендации по использованию возможности управления памятью .....	81
4.7. Встроенная система ФАПЧ .....	82
4.7.1. Описание регистров ФАПЧ .....	83
4.7.2. Прерывание от ФАПЧ .....	85
4.7.3. Режимы ФАПЧ .....	86
4.7.4. Система ФАПЧ и режим Power Down .....	86
4.7.5. Вычисление частоты ФАПЧ .....	87
4.7.6. Процедура настройки параметров системы ФАПЧ .....	87
4.8. Управление потребляемой мощностью .....	88
4.8.1. Описание регистров .....	89
4.8.2. Рекомендации по использованию управления потребляемой мощностью .....	93
4.9. Системный сброс .....	93
4.10. Делитель VPB .....	94
4.11. Таймер «пробуждения» .....	96

<b>Глава 5. Модуль акселератора памяти (МММ)</b> . . . . .	98
5.1. Описание . . . . .	98
5.2. Состав модуля акселератора памяти . . . . .	99
5.3. Банки Flash-памяти . . . . .	100
5.4. Защелки команд и защелки данных . . . . .	100
5.5. Проблемы программирования Flash-памяти . . . . .	100
5.6. Режимы работы модуля акселератора памяти . . . . .	101
5.7. Конфигурация модуля акселератора памяти . . . . .	102
5.8. Описание регистров модуля акселератора памяти . . . . .	102
5.9. Рекомендации по использованию модуля акселератора памяти . . . . .	104
<b>Глава 6. Контроллер векторов прерываний (VIC)</b> . . . . .	105
6.1. Описание VIC . . . . .	105
6.2. Регистры VIC . . . . .	106
6.3. Источники прерываний . . . . .	113
6.4. Ложные прерывания . . . . .	114
6.5. Детальные исследования ложных прерываний . . . . .	115
6.6. Рекомендации по использованию VIC . . . . .	117
<b>Глава 7. Цоколевка и назначение выводов</b> . . . . .	120
<b>Глава 8. Блок коммутации выводов</b> . . . . .	178
8.1. Назначение . . . . .	178
8.2. Описание регистров . . . . .	178
8.3. Значения регистров выбора функций выводов . . . . .	194
8.4. Управление загрузкой в микроконтроллерах с контроллером внешней памяти . . . . .	194
<b>Глава 9. Линии ввода/вывода общего назначения (GPIO)</b> . . . . .	195
9.1. Описание регистров GPIO . . . . .	196
9.2. Рекомендации по использованию GPIO . . . . .	199
<b>Глава 10. Универсальный асинхронный приемопередатчик UART0</b> . . . . .	200
10.1. Описание регистров UART0 . . . . .	200
10.2. Архитектура UART0 . . . . .	208
<b>Глава 11. Универсальный асинхронный приемопередатчик UART1</b> . . . . .	210
11.1. Описание регистров UART1 . . . . .	211
11.2. Архитектура UART1 . . . . .	223
<b>Глава 12. Интерфейс I<sup>2</sup>C</b> . . . . .	225
12.1. Типовые особенности LPC2000 . . . . .	225
12.2. Общее описание . . . . .	225
12.3. Операционные режимы I <sup>2</sup> C . . . . .	226
12.3.1. Режим ведущего передатчика . . . . .	226
12.3.2. Режим ведущего приемника . . . . .	229
12.3.3. Режим ведомого приемника . . . . .	232
12.3.4. Режим ведомого передатчика . . . . .	236
12.4. Описание выводов . . . . .	239
12.5. Описание регистров . . . . .	239
<b>Глава 13. Интерфейс SPI</b> . . . . .	247
13.1. Архитектура блока SPI . . . . .	247
13.2. Описание интерфейса SPI . . . . .	248
13.2.1. Краткий обзор SPI . . . . .	248
13.2.2. Передача данных через SPI . . . . .	248
13.3. Детализация периферии SPI . . . . .	250
13.3.1. Общая информация . . . . .	250
13.3.2. Операции ведущего . . . . .	250
13.3.3. Операции ведомого . . . . .	251
13.3.4. Условия возникновения исключительных ситуаций SPI . . . . .	252
13.4. Описание выводов SPI . . . . .	253
13.5. Описание регистров SPI . . . . .	254
<b>Глава 14. Контроллеры CAN и приемные фильтры</b> . . . . .	258
14.1. Отличительные особенности контроллеров CAN семейства LPC2000 . . . . .	258
14.2. Описание выводов CAN . . . . .	259
14.3. Карта памяти блока CAN . . . . .	259

14.4. Регистры контроллера CAN	259
14.5. Операции контроллера CAN	274
14.5.1. Обработка ошибок	274
14.5.2. «Спящий» режим	275
14.5.3. Прерывания	275
14.5.4. Приоритет передачи	276
14.6. Центральные регистры CAN	276
14.7. Глобальный приемный фильтр	278
14.7.1. Регистры приемного фильтра	281
14.7.2. Примеры таблиц приемного фильтра и значений идентификаторов	284
14.8. Режим «fullCAN»	285
<b>Глава 15. Таймеры TIMER0 и TIMER1</b>	287
15.1. Общее описание	287
15.2. Архитектура таймера	288
15.3. Описание выводов	289
15.4. Описание регистров таймеров	289
15.5. Пример операций таймера	296
<b>Глава 16. Широтно-импульсный модулятор (PWM)</b>	298
16.1. Основные функциональные характеристики	298
16.2. Структура и общее описание модуля PWM	299
16.2.1. Правила изменения выходного сигнала в канале PWM с управлением одиночным перепадом	302
16.2.2. Правила изменения выходного сигнала в канале PWM с управлением двойным перепадом	302
16.3. Описание выводов PWM	303
16.4. Описание регистров PWM	303
<b>Глава 17. Аналого-цифровой преобразователь (ADC)</b>	312
17.1. Описание выводов ADC	312
17.2. Описание регистров ADC	314
17.3. Операции модуля ADC	317
17.3.1. Преобразование, вызываемое аппаратно	317
17.3.2. Генерация тактового сигнала	318
17.3.3. Прерывания	318
17.3.4. Влияние на точность АЦП цифровых цепей	318
<b>Глава 18. Модуль часов-календаря реального времени (RTC)</b>	319
18.1. Архитектура модуля RTC	319
18.2. Описание регистров RTC	320
18.2.1. Прерывания от модуля RTC	321
18.2.2. Смешанная группа регистров	322
18.2.3. Прерывание от инкрементирования счетчиков	323
18.2.4. Сигнальная маска	324
18.2.5. Регистры консолидированного времени	325
18.2.6. Группа счетчиков времени	326
18.2.7. Определение високосного года	327
18.2.8. Группа сигнальных регистров	327
18.3. Рекомендации по использованию модуля RTC	328
18.4. Делитель опорной частоты (предделитель) RTC	328
18.4.1. Примеры использования предделителя	330
18.4.2. Операции предделителя RTC	331
<b>Глава 19. сторожевой таймер (WDT)</b>	332
19.1. Описание WDT	332
19.2. Описание регистров WDT	334
19.3. Рекомендации по использованию сброса от WDT и внешнего запуска	336
<b>Глава 20. Организация и программирование Flash-памяти</b>	337
20.1. Система Flash-памяти LPC2000	337
20.2. Загрузчик Flash-памяти	337
20.3. Карта памяти после сброса	338
20.4. Критерий действительного пользовательского программного кода	339

20.5. Коммуникационные протоколы	340
20.6. Блок-схема процесса загрузки	342
20.7. Номера секторов Flash-памяти	343
20.8. Защита кода от чтения	347
20.9. Команды ISP	348
20.10. Команды IAP	355
20.11. Интерфейс JTAG программирования Flash-памяти	361
<b>Глава 21. Модуль ЦАП (DAC)</b>	<b>362</b>
21.1. Описание выводов ЦАП	362
21.2. Описание регистров ЦАП	362
21.3. Операции ЦАП	363
<b>Глава 22. Встроенные логические схемы отладки (ICE)</b>	<b>364</b>
22.1. Описание ICE	365
22.2. Описание выводов ICE	366
22.2.1. Состояние мультиплексированных выводов JTAG при сбросе	366
22.3. Описание регистров ICE	366
<b>Глава 23. Встроенная макроячейка трассировки (ETM)</b>	<b>368</b>
23.1. Описание ETM	368
23.2. Конфигурация ETM	369
23.3. Описание выводов ETM	370
23.3.1. Мультиплексированные выходы ETM в состоянии сброса	370
23.4. Описание регистров ETM	370
<b>Глава 24. Отладчик RealMonitor</b>	<b>372</b>
24.1. Описание RealMonitor	372
24.2. Компоненты RealMonitor	373
24.3. Порядок работы Real Monitor	374
24.3.1. Разрешение работы RealMonitor	375
24.4. Обработка исключительных ситуаций	376
24.4.1. Обработка исключительных ситуаций RealMonitor	376
24.4.2. Инициализация RMTarget	377
24.5. Пример кода с использованием RealMonitor	377
24.6. Опции компоновки RealMonitor	379
Список рекомендуемой литературы к Части 1	381
<b>Часть 2. Разработка и отладка приложений на базе LPC2000</b>	<b>382</b>
<b>Глава 25. Интегрированная среда IDE IAR Embedded Workbench™</b>	<b>383</b>
25.1. Структура IDE	384
25.1.1. Структура директорий IDE	384
25.1.2. Микроконтроллеры, поддерживаемые IDE	386
25.1.3. Типы файлов IDE	386
25.2. Создание проектов приложений в IDE	388
25.2.1. Параметры настройки нового проекта	388
25.2.2. Создание окна рабочей области	389
25.2.3. Создание нового проекта	390
25.2.4. Добавление файлов к проекту	392
25.2.5. Задание опций проекта	393
25.3. Компилирование и компоновка приложения	395
25.3.1. Компилирование исходных файлов	395
25.3.2. Анализ файла листинга	396
25.3.3. Компоновка приложения	398
25.3.4. Анализ файла карты компоновщика	400
25.4. Отладка приложения в режиме симуляции	400
25.4.1. Старт отладчика	401
25.4.2. Организация окон	402
25.4.3. Просмотр команд исходного текста	402
25.4.4. Отслеживание переменных	404
25.4.5. Установка и мониторинг точек останова	405

25.4.6. Отладка в режиме дизассемблирования . . . . .	406
25.4.7. Мониторинг регистров . . . . .	407
25.4.8. Мониторинг памяти . . . . .	408
25.4.9. Просмотр окна терминала ввода/вывода . . . . .	409
25.4.10. Завершение выполнения приложения . . . . .	410
<b>Глава 26. Совместное использование в приложении модулей на Си и на ассемблере . . . . .</b>	<b>411</b>
26.1. Соглашение о вызовах . . . . .	411
26.2. Добавление ассемблерного модуля к проекту . . . . .	412
26.2.1. Модификации заданных установок проекта . . . . .	412
26.2.2. Исследование ассемблерного файла листинга . . . . .	414
26.2.3. Завершение компоновки проекта . . . . .	414
<b>Глава 27. Симуляция прерываний . . . . .</b>	<b>415</b>
27.1. Добавление программы обработки прерывания . . . . .	415
27.1.1. Текст программы обработки прерывания . . . . .	415
27.2. Задание параметров проекта . . . . .	416
27.2.1. Установка среды симуляции . . . . .	416
27.2.2. Определение макрофайла установок C-SPY . . . . .	417
27.2.3. Определение опций C-SPY . . . . .	417
27.2.4. Формирование проекта . . . . .	418
27.2.5. Старт симулятора . . . . .	419
27.2.6. Определение симулируемого прерывания . . . . .	419
27.2.7. Задание точки останова немедленного чтения . . . . .	420
27.3. Выполнение приложения . . . . .	421
27.4. Использование макроопределений для прерываний и точек останова . . . . .	421
<b>Глава 28. Работа с модулями библиотек . . . . .</b>	<b>423</b>
28.1. Использование библиотек . . . . .	423
28.2. Создание нового проекта . . . . .	424
28.3. Создание библиотечного проекта . . . . .	424
28.4. Использование библиотеки в прикладном проекте . . . . .	425
<b>Глава 29. Универсальный компоновщик IAR XLINK Linker™ . . . . .</b>	<b>426</b>
29.1. Объектный формат . . . . .	427
29.2. Функции IAR XLINK . . . . .	427
29.3. Выходной формат . . . . .	427
29.4. Входные файлы и модули . . . . .	427
29.4.1. Библиотеки . . . . .	428
29.4.2. Создание библиотек . . . . .	428
29.5. Сегменты . . . . .	429
29.5.1. Управление сегментами . . . . .	429
29.5.2. Переадресация . . . . .	430
29.5.3. Типы распределения сегментов . . . . .	430
29.5.4. Типы сегментов памяти . . . . .	431
29.5.5. Ошибки перекрытия . . . . .	432
29.5.6. Ошибки диапазона . . . . .	432
29.5.7. Примеры размещения сегментов . . . . .	432
29.6. Настройки XLINK в IDE IAR Embedded Workbench . . . . .	433
29.6.1. Опции XLINK . . . . .	433
29.7. Примеры командных файлов компоновщика . . . . .	435
<b>Глава 30. Отладка приложений в целевой системе . . . . .</b>	<b>441</b>
30.1. Аппаратные средства отладчика C-SPY . . . . .	442
30.2. Отладка с помощью аппаратных средств C-SPY . . . . .	447
30.2.1. Старт отладчика . . . . .	447
30.2.2. Задание контрольных точек в целевой системе . . . . .	449
<b>Глава 31. Запись пользовательских программ во Flash-память микроконтроллеров . . . . .</b>	<b>451</b>
31.1. Загрузка Flash-памяти с помощью LPC2000 Flash Utility . . . . .	451
31.2. Загрузка Flash-памяти микроконтроллеров LPC2000 с помощью C-SPY . . . . .	455
31.2.1. Задание загрузчика Flash-памяти . . . . .	456
31.2.2. Механизм загрузки во Flash-память . . . . .	458
31.2.3. Требования к компоновке . . . . .	458

31.2.4. Порядок действий при загрузке во Flash-память . . . . .	460
<b>Глава 32. Общее описание настроек проектов в IAR EWARM . . . . .</b>	<b>461</b>
<b>Глава 33. Пользовательские проекты на базе микроконтроллеров LPC2000 . . . . .</b>	<b>464</b>
33.1. Платформа для создания приложений . . . . .	464
33.2. Интерфейс ввода/вывода в целевой пользовательской системе. . . . .	472
33.2.1. Интерфейс кнопок управления . . . . .	472
33.2.2. Интерфейс ЖКИ . . . . .	477
33.3. Первый пользовательский проект . . . . .	481
33.4. Интерфейс обслуживания RTC . . . . .	482
33.5. Интерфейс обслуживания АЦП . . . . .	491
33.6. Интерфейс обслуживания ШИМ (PWM) . . . . .	494
33.7. Интерфейс обслуживания UART . . . . .	499
33.8. Интерфейс обмена с внешней памятью через I <sup>2</sup> C . . . . .	506
33.9. Интерфейс обмена с внешним устройством через SPI . . . . .	516
33.10. Обслуживание внутрисхемного программирования IAP . . . . .	526
Список рекомендуемой литературы к Части 2 . . . . .	529
Краткий словарь терминов . . . . .	531
<i>Приложение 1.</i> Предельно допустимые параметры микроконтроллеров LPC2000 . . . . .	534
<i>Приложение 2.</i> Статические характеристики микроконтроллеров LPC2000 . . . . .	536
<i>Приложение 3.</i> Статические характеристики АЦП микроконтроллеров LPC2000 . . . . .	552
<i>Приложение 4.</i> Динамические характеристики микроконтроллеров LPC2000 . . . . .	554
<i>Приложение 5.</i> Принципиальная схема одного из возможных исполнений интерфейса Wiggler . . . . .	556
<i>Приложение 6.</i> Таблица фонтов (шрифтов) для русифицированного ЖКИ со встроенным контроллером управления, совместимым с HD44780 . . . . .	557
Материалы, размещенные на компакт-диске . . . . .	558

## Предисловие автора

В настоящее время особой популярностью у разработчиков электронных устройств пользуются так называемые встраиваемые микроконтроллеры. В мире выпускается большое количество семейств микроконтроллеров, в основном на базе приборов с 8-битной<sup>1)</sup> шиной данных и процессорными RISC- и CISC-ядрами. Их производительность и объем памяти вполне достаточны для решения множества бытовых и промышленных задач, а архитектура ядер многих из этих семейств хорошо изучена пользователями. Для 8-битных микроконтроллеров написано огромное количество программ, которые, наряду с многочисленными русифицированными описаниями самих этих устройств, свободно доступны в сети Интернет.

16- и 32-битные микроконтроллеры, которые обеспечивают более высокую производительность, пока распространены не столь широко. Их применение обусловлено повышенной сложностью решаемых задач, жесткими требованиями к производительности встраиваемых контроллеров управления, необходимостью иметь в электронных устройствах развитые пользовательские интерфейсы, предназначенные для отображения информации, управления, индикации и т. д. Типичные для 16- и 32-битных микроконтроллеров приложения (сотовые телефоны, дисководы, модемы и т. п.) предъявляют к встраиваемым управляющим контроллерам непрерывно возрастающие требования. Особенно важно обеспечить их высокую эффективность, сохранив при этом низкую стоимость, отличавшую 8-битные микроконтроллеры.

Значительных успехов в области создания 16/32-битных микропроцессорных (микроконтроллерных) ядер добилась британская фирма Advanced RISC Machines (ARM), специализирующаяся на разработке микропроцессоров и периферии к ним, и продающая лицензии на их производство другим фирмам-производителям. 32-битные микроконтроллеры, использующие процессорное ядро с архитектурой ARM, приобрели широкую популярность у разработчиков. Благодаря высокой производительности и выгодному соотношению «цена/качество», микроконтроллеры с ядром ARM представляются многим потребителям элементной базы весьма перспективными. Фактически, микроконтроллеры с ядром ARM сегодня выделились в отдельный класс встраиваемых контроллеров — ARM-контроллеров.

---

<sup>1)</sup> В отечественной литературе принято употреблять термины «разряд», «разрядный». Но так как современной тенденции в цифровой технике более соответствует употребление понятий «бит», «битный» (бит — двоичный разряд), то в данном издании будет использоваться зарубежная терминология. (*Прим. ред.*)

Фирма ARM, являясь разработчиком процессорных ядер, не имеет собственного полупроводникового производства и не поставляет готовых кристаллов. Ядра поставляются заказчикам — производителям микросхем — в форме поведенческого описания на языке VHDL или Verilog (такие ядра называются синтезируемыми), в виде электрической схемы процессорного ядра или в виде топологической макроячейки (описания топологии ядра для реализации в составе интегральной микросхемы). Используя выбранную форму представления ядра, производители элементной базы создают различные модели микроконтроллеров, дополняя процессорное ядро набором периферийных устройств. При производстве микроконтроллеров используется современная нанотехнология, обеспечивающая минимальные размеры элементов транзисторной структуры микропроцессора 0.25, 0.18 или 0.13 мкм.

«Кремниевыми» партнерами фирмы ARM, т. е. компаниями, которые используют ее разработки при создании своих приборов, являются такие производители элементной базы, как Alcatel, Atmel, Asahi Kasei Microsystems, Cirrus Logic, Digital, GEC Plessey, Hyundai, Lucent, Lucky GoldStar, NEC, OKI, Philips, Rockwell, Rohm, Samsung, Sharp, Sony, Symbios, Texas Instruments, VLSI, Yamaha. Некоторые из них применяют разработанные фирмой ARM процессоры для решения специальных задач, но чаще всего они используются в устройствах, предназначенных для массового применения (мобильные телефоны, системы управления автомобильными двигателями, лазерные принтеры и т. д.). Пользователь — разработчик аппаратуры — имеет дело с процессорными ядрами ARM, функционирующими в составе готовой микросхемы микроконтроллеров перечисленных производителей.

Процессоры ARM поддерживаются многими программными продуктами как самой компании, так и других производителей. Среди этих продуктов, которые образовали солидную инфраструктуру программного обеспечения и средств разработки, — отладчики, компиляторы C/C++, внутрисхемные эмуляторы, таблицы разработки, операционные системы реального времени, драйверы низкого уровня, а также программные приложения высокого уровня. Фирмы Accelerated Technology, Enea OSE Systems, ISI, JavaSoft, JMI, Microtec, Microsoft, Perihelion, Psion, Wind River и другие компании обеспечивают совместимость своих ОС и средств разработки с процессорами ARM.

Фирмой ARM разработан целый ряд 32-битных RISC-процессоров с различными возможностями и различной производительностью, а ядро ARM7, разработанное еще в 1994 году, используется до настоящего времени. Сама фирма определяет процессор ARM7 как универсальное ядро 32-битного RISC-микропроцессора с малым энергопотреблением, предназначенное для использования в различных заказных и специальных ИС. Малые размеры RISC-ядра позволяют успешно интегрировать его в большие заказные схемы, которые могут содержать RAM, ROM (Flash), DSP, дополнительную логику и другие элементы.

К областям применения ядра ARM7 фирма-производитель относит:

- **телекоммуникации** — контроллеры GSM-терминалов;
- **обмен данными** — средства преобразования протоколов и модемы;
- **портативные вычисления** — Palmtop-компьютеры;
- **портативные измерительные устройства** — карманные устройства сбора данных;
- **автомобильную технику** — устройства управления двигателями;
- **информационные системы** — Smart-карты;
- **средства отображения** — JPEG-контроллеры.



Пришедшее на смену ARM6 THUMB семейство ядер ARM7 THUMB разработано на основе 32-битной целочисленной RISC-архитектуры и обеспечивает максимальную производительность до 130 MIPS. Удельная производительность ядра ARM7 THUMB, изготовленного по 0.18-мкм технологии, составляет 0.9...1.0 MIPS/МГц тактовой частоты, а энергопотребление составляет не более 0.39...0.40 мВт/МГц. Ядро ARM7TDMI-S, о котором пойдет речь в книге, является синтезируемым.

Все ядра семейства ARM7 THUMB имеют традиционную архитектуру фон Неймана с общей памятью команд и данных. В последующих семействах ARM9 и ARM10 реализуется гарвардская архитектура с отдельной внутренней кэш-памятью команд и данных.

Один из «кремниевых» партнеров ARM, корпорация NXP (до 2006 г. она называлась Philips Semiconductors), выпускает широкую номенклатуру 16/32-битных микроконтроллеров с ядром ARM7TDMI-S — семейство LPC2000. Эти устройства имеют весьма привлекательное для потребителей соотношение «цена/качество» и обеспечивают производительность в десятки MIPS.

Назначением данной книги представляется оказание помощи разработчикам в использовании микроконтроллеров семейства LPC2000 при проектировании электронной аппаратуры. В своей справочной части книга основана на переводе оригинальной документации производителя. В целом же, она представляет собой не только справочное пособие для изучения микроконтроллеров семейства LPC2000, но и практическое руководство по их прикладному использованию. Автор не ставил перед собой задачу создать всеобъемлющий справочник по всем микроконтроллерам семейства LPC2000, в связи с чем, описание ряда периферийных узлов LPC2000 дано в книге на примере нескольких, наиболее типичных, представителей этого семейства.

Книга предназначена для специалистов в области разработки электронной аппаратуры, студентов соответствующих специальностей и радиолюбителей. Предполагается, что читатель знаком с основами цифровой и аналоговой схемотехники электронных устройств, а также обладает базовыми навыками программирования на языке С.

В первой части книги содержится подробная справочная информация по большинству микроконтроллеров семейства LPC2000, собранная путем перевода и анализа источников, предлагаемых фирмами ARM и NXP. Сюда входят: перечень типовых технических характеристик контроллеров LPC2000, описание их архитектуры и программной модели (дано описание ядра ARM7TDMI-S и приведена его система команд), описание аппаратного построения и программирования большинства периферийных модулей семейства LPC2000, рекомендации производителя по программированию и применению отдельных узлов устройств LPC2000.

Во второй части книги содержится подробная справочная информация по программному пакету IDE IAR Embedded Workbench™ (IAR EWARM), который представляет собой популярное среди профессиональных разработчиков электронной техники средство разработки/отладки/программирования приложений на основе микроконтроллеров семейства LPC2000. Эта информация дополнена сведениями из личного опыта автора, приобретенного при работе в IAR EWARM с микроконтроллерами LPC2000. Описания программных средств разработки-отладки и программирования снабжены примерами, поясняющими их применение на практике. Помимо указанных сведений, во вторую часть книги вошли описания алгоритмов и исходных текстов ряда управляющих программ

на языке С, демонстрирующих пользователю в «железе» работу ядра и основных периферийных узлов одного из представителей семейства LPC2000 — LPC2129. Эти программы написаны и отлажены до рабочего состояния лично автором. Каждая из них является полностью законченным продуктом и, помимо демонстрационных компонентов, содержит отдельные программные блоки — драйверы узлов LPC2000 и нескольких внешних устройств. Приведенное в книге встраиваемое программное обеспечение снабжено комментариями на русском языке, что облегчает возможность его применения в пользовательских приложениях, разработанных на базе микроконтроллеров семейства LPC2000.

Отличительной особенностью данной книги является то, что она не только содержит сведения справочного характера, но и охватывает все этапы проектирования приложений на основе микроконтроллеров LPC2000. Это позволяет в короткие сроки овладеть навыками работы с LPC2000 даже начинающим разработчикам. После прочтения книги им уже не надо будет думать о том, с чего начать работу с LPC2000, как написать первую программу, где взять примеры программных кодов для LPC2000, как адаптировать эти примеры под свою задачу, — все эти и многие другие сведения имеются в книге.

Материал, изложенный в книге, многократно проверен. Однако, поскольку вероятность технических ошибок все-таки существует, автор не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим, автор не несет ответственности за возможные ошибки, связанные с использованием данной книги.

# СПРАВОЧНАЯ ИНФОРМАЦИЯ ПО СЕМЕЙСТВУ LPC2000

Микроконтроллеры семейства LPC2000 представляют собой 32-битные устройства с процессорным ядром ARM7TDMI-S. Согласно рекомендациям производителя, они предназначены для использования в автомобилестроении, медицине, сетевых контроллерах, а также в промышленной и бытовой электронике, в том числе с батарейным питанием. Выпускаемые в настоящее время приборы работают на тактовой частоте до 60 МГц, обеспечивая производительность до 54 MIPS, имеют встроенную Flash-память объемом 128/256/512 Кбайт со 128-битной шиной, статическую оперативную память (SRAM) объемом 16/32/64 Кбайт, модуль ШИМ, многоканальный 10-битный модуль АЦП, 10-битный модуль ЦАП, интерфейсы I<sup>2</sup>C, SPI, CAN, два UART (один UART с полным модемным интерфейсом), USB.

## **1.1. Характерные особенности микроконтроллеров семейства LPC2000**

Микроконтроллеры семейства LPC2000 имеют следующие типовые характеристики:

- высокопроизводительное 32/16-битное ядро ARM7TDMI-S;
- встроенный генератор 1...30 МГц со схемой ФАПЧ;
- 8/16/32/64 Кбайт памяти типа SRAM (статическое ОЗУ);
- 64/128/256/512 Кбайт памяти типа Flash с 128-битной шиной и обращением без цикла ожидания (ресурс не менее 10000 циклов стирания/записи в рабочем температурном диапазоне);
- стандартный отладочный интерфейс JTAG;
- ISP (In-System Programming) внутрисистемное программирование Flash-памяти;
- IAP (In-Application Programming) режим побайтового программирования в составе целевой программы (пользовательского приложения). Программирование 512-байтовой (256-байтовой) строки занимает 1 мс, полное стирание всей памяти занимает 400 мс;
- работа с командами длиной 32 битов (стандартный режим ARM) и 16 битов (режим THUMB);
- внешняя 8/16/32-битная шина (только для устройств LPC2000 в 144-выводном корпусе);
- многоканальный модуль ШИМ;
- интерфейсы UART, быстрый I<sup>2</sup>C (400 Кбит/с), SPI;
- интерфейсы CAN с приемными фильтрами;
- два 32-битных таймера с каналами сравнения и захвата;
- таймер реального времени и сторожевой таймер WDT;
- интерфейс USB;
- многоканальный 10-битный АЦП, обеспечивающий время преобразования в одном канале 2.44 мкс;
- 10-битный модуль ЦАП;
- встроенный отладочный интерфейс ICE, позволяющий задавать точки останова;
- раздельное питание для ядра и встроенной периферии (1.65...1.95 В/3.0...3.6 В), отключение неиспользуемой периферии, два режима экономии потребляемой мощности;
- индивидуальное управление питанием встроенной периферии;

- линии ввода/вывода общего назначения GPIO допускают подачу на них напряжения ВЫСОКОГО уровня 5 В (номинальная величина напряжения ВЫСОКОГО уровня на этих линиях 3.0...3.6 В).

## 1.2. Приложения

- индустриальный контроль (управление);
- медицинские системы;
- управление доступом;
- межсетевые интерфейсы;
- встроенный модем;
- универсальные приложения.

В сводной таблице **Табл. 1.1** приведены основные параметры большинства микроконтроллеров семейства LPC2000. Как можно видеть из таблицы, контроллеры семейства LPC2000 представлены обширным модельным рядом и выпускаются в индустриальном и коммерческом исполнениях. Применение в приборах LPC2000 ядра ARM7TDMI-S с встроенным трассировщиком (Real-Time Trace) и монитором реального времени (Real-Time Monitor) обеспечивает преимущества, обусловленные широкой поддержкой данного ядра промышленностью и наличием большого выбора программных и аппаратных средств разработки (ARM, Ashling, Hitex, Keil Software, IAR, Nohau, Phytes и др.). В ближайших планах фирмы NXP значится дальнейшее расширение номенклатуры ARM-микроконтроллеров семейства LPC2000, в частности, за счет введения интерфейсов 802.11, Ethernet и наращивания числа встроенных модулей CAN. Помимо уже выпускающихся микроконтроллеров семейства LPC2000, фирма-производитель недавно объявила о создании новой серии внутри семейства LPC2000 — ARM-7 Mini. На момент написания этой книги было известно о трех представителях новой серии — LPC2101, LPC2102, и LPC2103. Они будут производиться по технологии 0.16 мкм. Наличие 128-битной шины доступа к Flash-памяти позволяет обеспечить работу на частоте до 70 МГц и достичь производительности 63 MIPS, что делает эти микроконтроллеры абсолютными лидерами на рынке Flash ARM7TDMI-S-устройств.

Предельно допустимые параметры контроллеров LPC2000, описанных в данной книге, приведены в Приложении 1, статические характеристики — в Приложении 2, характеристики АЦП — в Приложении 3, динамические характеристики — в Приложении 4. Символ «ТВД» (To Be Discussed) в таблицах приложений указывает на то, что значение параметра еще не определено производителем.

Таблица 1.1. Сводная таблица микроконтроллеров семейства LPC2000

Устройство	ОЗУ [байт]	Flash-память		SAР/СОМ		Питание модуля RTC		Интерфейсы				ШИМ	АЦП	ЦАП	Внешняя шина [бит]			F <sub>СРU</sub> [МГц]	Напряжение питания [В]		Температурный диапазон [°С]	Корпус
		защита	назначение [бит]	каналы	каналы	внутреннее	внешнее	UART	CAN	I <sup>2</sup> C	SPI				SSP	USB	бит/канал		бит/канал	Адрес		
LPC2101FBD48	2К	8К	•	2×32 2×16	7	•	•	2	2	2	•	—	10/8	—	—	—	32	1...70	1.8	3.3	—40...+85	LQFP48
LPC2102FBD48	4К	16К	•	2×32 2×16	7	•	•	2	2	2	•	—	10/8	—	—	—	32	1...70	1.8	3.3	—40...+85	LQFP48
LPC2103FBD48	8К	32К	•	2×32 2×16	7	•	•	2	2	2	•	—	10/8	—	—	—	32	1...70	1.8	3.3	—40...+85	LQFP48
LPC2104BBD48	16К	128К	—	2×32	7	•	•	2	1	1	—	6	—	—	—	—	32	1...60	1.8	3.3	0...+70	LQFP48
LPC2105BBD48	32К	128К	—	2×32	7	•	•	2	1	1	—	6	—	—	—	—	32	1...60	1.8	3.3	0...+70	LQFP48
LPC2106BBD48	64К	128К	—	2×32	7	•	•	2	1	1	—	6	—	—	—	—	32	1...60	1.8	3.3	0...+70	LQFP48
LPC2106FBD48	64К	128К	—	2×32	7	•	•	2	1	1	—	6	—	—	—	—	32	1...60	1.8	3.3	—40...+85	LQFP48
LPC2106FHN48	64К	128К	—	2×32	7	•	•	2	1	1	—	6	—	—	—	—	32	1...60	1.8	3.3	—40...+85	HVQFN48
LPC2114FBD64	16К	128К	•	2×32	8	•	•	2	1	2	—	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+85	LQFP64
LPC2114FHN64	16К	128К	•	2×32	8	•	•	2	1	2	—	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+85	HVQFN64
LPC2119FBD64	16К	128К	•	2×32	8	•	•	2	2	1	2	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+85	LQFP64
LPC2119FHN64	16К	128К	•	2×32	8	•	•	2	2	1	2	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+85	HVQFN64
LPC2124FBD64	16К	256К	•	2×32	8	•	•	2	1	2	—	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+85	LQFP64
LPC2124FHN64	16К	256К	•	2×32	8	•	•	2	1	2	—	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+85	HVQFN64
LPC2129FBD64	16К	256К	•	2×32	8	•	•	2	2	1	2	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+85	LQFP64
LPC2129FHN64	16К	256К	•	2×32	8	•	•	2	2	1	2	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+85	HVQFN64

Таблица 1.1. Сводная таблица микроконтроллеров семейства LPC2000 (продолжение)

Устройство	ОЗУ [байт]	Flash-память		Назначение [бит]	SAР/ СОМ	Питание модуля RTC		Интерфейсы						ШИМ каналы	АЦП бит/канал	ЦАП бит/канал	Внешняя шина			F <sub>СРU</sub> [МГц]	Напряжение питания, [В]		Температурный диапазон [°С]	Корпус		
		защита	назначение [байт]			внутреннее ФАПЧ	внешнее 32 кГц	UART	CAN	I <sup>2</sup> C	SPI	SSP	USB				каналы	бит/канал	Адрес		Данные	CS			I/O	CPU
LPC2129FHN64	16К	256К	•	2×32	8	•	•	2	2	2	2	2	2	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+85	HVQFN64		
LPC2131FBD64	8К	64К	•	2×32	8	•	—	•	2	—	1	2	—	6	10/8	10/1	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2132FBD64	16К	64К	•	2×32	8	•	—	•	2	—	1	2	—	6	10/8	10/1	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2134FBD64	16К	128К	•	2×32	8	•	—	•	2	—	1	2	—	6	10/8	10/1	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2136FBD64	32К	256К	•	2×32	8	•	—	•	2	—	1	2	—	6	10/8×2	10/1	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2138FBD64	32К	512К	•	2×32	8	•	—	•	2	—	2	1	•	6	10/8×2	10/1	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2141FBD64	8К	32К	•	2×32	8	•	—	•	2	—	1	2	—	6	10/8	—	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2142FBD64	16К	64К	•	2×32	8	•	—	•	2	—	1	2	—	6	10/8	10/1	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2144FBD64	16К	128К	•	2×32	8	•	—	•	2	—	1	2	—	6	10/8	10/1	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2146FBD64	32К	256К	•	2×32	8	•	—	•	2	—	1	2	—	6	10/8×2	10/1	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2148FBD64	32К	512К	•	2×32	8	•	—	•	2	—	2	1	•	6	10/8×2	10/1	—	—	47	1...60	3.3	3.3	—40...+85	LQFP64		
LPC2194JBD64	16К	256К	•	2×32	8	•	—	•	2	4	1	2	—	6	10/4	—	—	—	46	1...60	1.8	3.3	—40...+105	LQFP64		
LPC2210FBD144	16К	—	—	2×32	8	•	•	•	2	—	1	2	—	6	10/8	—	—	24	32	4	76	1...60	1.8	3.3	—40...+85	LQFP144
LPC2212FBD144	16К	128К	•	2×32	8	•	•	•	2	—	1	2	—	6	10/8	—	—	24	32	4	112	1...60	1.8	3.3	—40...+85	LQFP144
LPC2214FBD144	16К	256К	•	2×32	8	•	•	•	2	—	1	2	—	6	10/8	—	—	24	32	4	112	1...60	1.8	3.3	—40...+85	LQFP144
LPC2220FBD144	64К	—	—	2×32	8	•	•	•	2	—	1	3	—	6	10/8	—	—	24	32	4	112	1...60	1.8	3.3	—40...+85	LQFP144
LPC2290FBD144	16К	—	—	2×32	8	•	•	•	2	2	1	2	—	6	10/8	—	—	24	32	4	76	1...60	1.8	3.3	—40...+85	LQFP144
LPC2292FBD144	16К	256К	•	2×32	8	•	•	•	2	2	1	2	—	6	10/8	—	—	24	32	4	112	1...60	1.8	3.3	—40...+125	LQFP144
LPC2294JBD144	16К	256К	•	2×32	8	•	•	•	2	4	1	2	—	6	10/8	—	—	24	32	4	112	1...60	1.8	3.3	—40...+125	LQFP144

### 1.3. Особенности архитектуры

Все микроконтроллеры семейства LPC2000 имеют процессорное ядро ARM7TDMI-S с поддержкой эмуляции, шину ARM7 контроллера встроенной памяти, шину AMBA AHB (Advanced Microcontroller Bus Architecture — Advanced High-performance Bus) контроллера прерываний и шину VPB (VLSI Peripheral Bus) встроенных периферийных устройств.

Под периферийные устройства, подключенные к шине АНВ, в общем адресном пространстве программной модели ARM размером 4 Гбайт LPC2000 отведено адресное пространство размером 2 Мбайт. Каждому периферийному устройству АНВ выделено адресное пространство размером 16 Кбайт, в пределах адресного пространства периферии АНВ.

Под периферийные устройства, подключенные к шине VPB, также отведено адресное пространство размером 2 Мбайт, которое начинается с адреса, находящегося на отметке 3.5 Гбайт. Каждому периферийному устройству VPB также выделено адресное пространство размером 16 Кбайт, в пределах адресного пространства периферии VPB.

Сопряжение периферийных устройств АНВ с периферийными устройствами VPB осуществляется посредством интерфейса шины VPB к шине АНВ. Для управления связью периферийных устройств с выводами микроконтроллера предназначен блок коммутации выводов (PIN CONNECT BLOCK), который должен быть корректно сконфигурирован пользовательским программным обеспечением и должен соответствовать определенным прикладным требованиям для использования периферийных функций и выводов. Подробнее об этом будет рассказано ниже.

### 1.4. Процессорное ядро ARM7TDMI-S

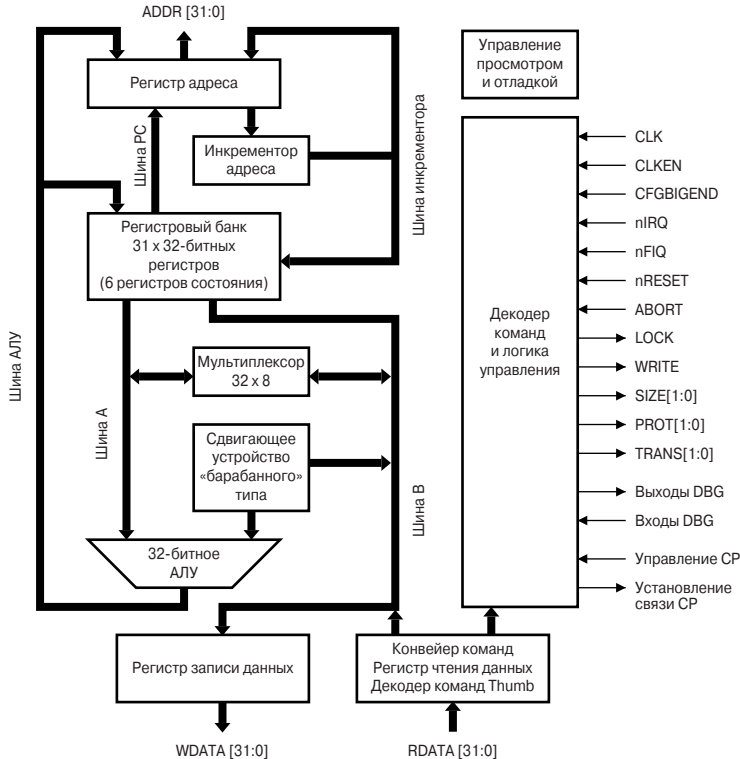
Буквенные обозначения «TDMI-S» в названии ядра ARM7TDMI-S, согласно спецификациям фирмы ARM, имеют следующую расшифровку:

- Т — указывает на наличие в составе ядра, помимо базовой архитектуры ARM, дополнительной архитектуры THUMB. Это позволяет пользователю выбрать для своего приложения одно из двух состояний (систем команд): ARM или THUMB;
- D — ядро имеет в своем составе дополнительные отладочные модули;
- M — ядро имеет в своем составе дополнительный аппаратный множитель, позволяющий выполнять команды умножения так называемой длинной формы (с 64-битным результатом);
- I — ядро имеет в своем составе встроенную логику отладки (ICE);
- S — ядро является полностью синтезируемым, т. е. его функционирование может быть описано на языках описания цифровых устройств VHDL или Verilog.

ARM7TDMI-S — универсальный 32-битный микропроцессор, который потребляет сравнительно небольшую мощность и при этом обеспечивает высокую эффективность. Архитектура ARM основана на использовании сокращенного набора команд процессора (RISC — Reduced Instruction Set Computer) и реализует намного более простой механизм декодирования команд, нежели у микроконтроллеров с архитектурой, основанной на использовании полного набора команд процессора (CISC — Complex Instruction Set Computer). Эта простота обеспечивает высокую производительность при выполнении каждой команды, а также малое время реакции на прерывания в реальном времени. В состав ядра включены отладочные ин-



терфейсы JTAG и ETM, а также встроенная логическая схема отладки ICE. Упрощенная блок-схема процессорного ядра ARM7TDMI-S приведена на **Рис. 1.1**. Более подробную информацию об архитектуре и особенностях построения ядра ARM7TDMI-S можно найти в технической документации, которая находится на прилагаемом к данной книге компакт-диске, а также в источниках [10, 11].



**Рис. 1.1.** Упрощенная блок-схема процессорного ядра ARM7TDMI-S

Для того, чтобы обеспечить непрерывность при обработке программ и обращениях к памяти, в ARM7TDMI-S, как и во многих других ядрах, реализован конвейер команд. Трехступенчатый конвейер ARM7TDMI-S работает следующим образом: пока одна инструкция выполняется, из двух следующих за ней в программе инструкций первая декодируется, а вторая считывается в памяти. Наличие конвейера позволяет выполнять последовательно поступающие команды (в том числе, команды умножения), затрачивая на обработку каждой из них один такт.

Ядро ARM7TDMI-S, помимо стандартного базового состояния, называемого ARM, также использует уникальное архитектурное решение, известное как состояние THUMB. Это решение в большей степени, нежели состояние ARM, подходит для крупномасштабных приложений, реализация которых предполагает ограничение объема используемой памяти или размера программного кода. Идея состояния THUMB заключается в использовании уменьшенной, по сравнению с системой команд ARM, системы команд. По существу, процессор ARM7TDMI-S имеет две системы команд:

- стандартная 32-битная система команд ARM;
- 16-битная система команд THUMB.

При использовании состояния THUMB с 16-битной длиной команды, в ряде случаев удается достичь двойной плотности относительно стандартного кода ARM; при этом сохраняется большинство функций кода ARM. Дело в том, что код THUMB оперирует тем же самым набором 32-битных регистров, что и код ARM. По некоторым оценкам, код THUMB «ужимается» до 65% размера функционально эквивалентного ему кода ARM, а также обеспечивает до 160% производительности по сравнению с 16-битной процессорной системой, архитектура которой отлична от ARM.

Процессор ARM7TDMI-S использует следующие типы данных:

- слово (32 бита);
- полуслово (16 битов);
- байт (8 битов).

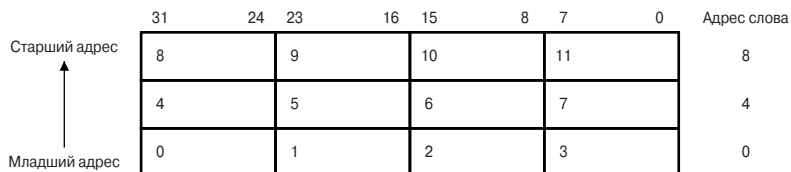
Процессор ARM7TDMI-S рассматривает память как линейное расположение байтов, пронумерованных в возрастающем порядке от нуля:

- байты 0...3 содержат первое хранимое в памяти слово;
- байты 4...7 содержат второе хранимое в памяти слово;
- байты 8...11 содержат третье хранимое в памяти слово и т. д.

Процессор ARM7TDMI-S использует следующие форматы хранения данных (программ) в памяти:

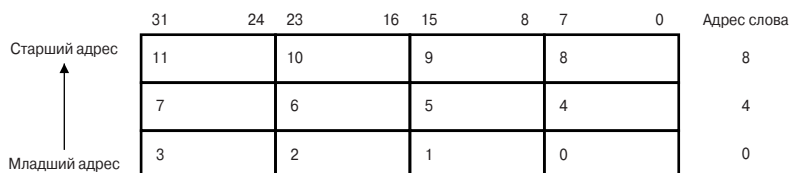
- big-endian;
- little-endian.

В соответствии с форматом big-endian, процессор сохраняет старший байт слова данных (программ) в байте памяти с наименьшим номером, а младший байт слова — в байте памяти с наибольшим номером, как изображено на **Рис. 1.2**. Например, в байте 0 памяти хранятся биты слова 31...24.



**Рис. 1.2.** Формат хранения данных (программ) в памяти big-endian

В соответствии с форматом little-endian, процессор сохраняет младший байт слова данных (программ) в байте памяти с наименьшим номером, а старший байт слова — в байте памяти с наибольшим номером, как изображено на **Рис. 1.3**. Например, в байте 0 памяти хранятся биты слова 7...0.



**Рис. 1.3.** Формат хранения данных (программ) в памяти little-endian

В исходном тексте пользовательской встроенной программы с помощью специальных директив можно указать, какой из форматов хранения данных будет использоваться.