ARM体系结构与编程

（第2版）

杜春雷　主编

ARM嵌入式处理器是一种高性能、低功耗的RISC芯片。世界上几乎所有的主要半导体厂商都生产基于ARM体系结构的通用芯片，或在其专用芯片中嵌入ARM的相关技术。

目前，ARM芯片广泛应用于无线产品、PDA、GPS、网络、消费电子产品、STB及智能卡中。

清华大学出版社

ARM 体系结构与编程
(第 2 版)

杜春雷　主　编

清华大学出版社
北 京

内 容 简 介

ARM 处理器是一种 16/32 位的高性能、低成本、低功耗的嵌入式 RISC 微处理器，由 ARM 公司设计，然后授权给各半导体厂商生产，它目前已经成为应用最为广泛的嵌入式处理器。

本书共为分 14 章，对 ARM 处理器的体系结构、指令系统和开发工具进行了比较全面的介绍。其中包括 ARM 体系、ARM 程序设计模型、ARM 汇编语言程序设计、ARM C/C++语言程序设计、ARM 连接器的使用、ARM 集成开发环境 CodeWarrior IDE 的介绍及高性能的调试工具 ADW 的使用。并在此基础上介绍一些典型的基于 ARM 体系的嵌入式应用系统设计的基本技术。通过阅读本书，可以使读者掌握开发基于 ARM 的应用系统的各方面的知识。

本书既可作为学习 ARM 技术的培训材料，也可作为嵌入式系统开发人员的参考手册。

图书在版编目(CIP)数据

ARM 体系结构与编程/杜春雷主编. --2 版. --北京：清华大学出版社，2015
ISBN 978-7-302-40342-5

Ⅰ. ①A…　Ⅱ. ①杜…　Ⅲ. ①微处理器—计算机体系结构—高等学校—教材　②微处理器—程序设计—高等学校—教材　Ⅳ. ①TP332

中国版本图书馆 CIP 数据核字(2015)第 112749 号

责任编辑：章忆文　杨作梅
封面设计：杨玉兰
责任校对：宋延清
责任印制：何　芊

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 刷 者：清华大学印刷厂
装 订 者：三河市少明印务有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：33　　**字　　数**：797 千字
版　　次：2003 年 2 月第 1 版　2015 年 8 月第 2 版　　**印　　次**：2015 年 8 月第1 次印刷
印　　数：51501～54500
定　　价：49.80 元

产品编号：064160-01

前　言

嵌入式系统是指以应用为中心，以计算机技术为基础，软件硬件可裁剪，适应应用系统对功能、可靠性、成本、体积和功耗严格要求的专用计算机系统。

嵌入式系统并不是最近出现的新技术，只是随着微电子技术和计算机技术的发展，微控制芯片功能越来越强大，嵌入微控制芯片的设备和系统越来越多，从而使得这种技术越来越引人注目而已。嵌入式系统与通用的计算机系统既有相似之处，也有明显的区别。通常，嵌入式系统中的系统程序(包括操作系统)与应用程序是浑然一体的，这些程序被编译连接成一个可以执行的二进制映像文件(Image)，这个二进制映像文件被固化在系统中，在系统复位后自动执行。嵌入式系统的开发系统与实际运行的系统并不相同，需要交叉编译系统和适当的调试系统。

ARM 嵌入式处理器是一种高性能、低功耗的 RISC 芯片。它由英国 ARM 公司设计，世界上几乎所有的主要半导体厂商都生产基于 ARM 体系结构的通用芯片，或在其专用芯片中嵌入 ARM 的相关技术。如 TI、Motorola、Intel、NS、Philips、Altera、Agilent、Atmel、Hynix、Sharp、Triscend、NEC、Cirrus Logic、Samsung 和 LinkUp 等公司都有相应的产品。目前 ARM 芯片广泛应用于无线产品、PDA、GPS、网络、消费电子产品、STB 及智能卡中，基于 ARM 内核的处理器年产量突破 90 亿个，已经成为业界的龙头。本书比较全面地介绍基于 ARM 技术的嵌入式应用系统的开发技术。

1. 本书的主要读者

本书对 ARM 处理器的体系结构、指令系统、开发工具做了比较全面的介绍。并在此基础上讨论一些典型的基于 ARM 体系嵌入式应用系统设计时的基本技术。通过阅读本书，可以使读者能够掌握开发基于 ARM 的应用系统的各方面的知识。它既可作为学习 ARM 技术的培训材料，也可作为嵌入式系统开发人员的参考手册。

2. 本书的主要内容

本书以可执行的二进制映像文件(Image)为中心，介绍基于 ARM 微处理器的嵌入式系统的开发过程所涉及的知识，主要包括以下几部分。

- Image 文件的“原材料”，包括*.c、*.h、*.obj、*.asm 及*.lib 文件。这些文件包括操作系统，通常以*.lib 形式提供，也有一些操作系统附属的源代码，可以为*.c、*.h、*.asm；BSP(其实也是操作系统的一部分，因为它对于不同的计算机主板是不同的，这里将其单独列出)，它通常为*.c、*.h、*.asm；语言库(如 C 语言运行库)，通常为*.lib；用户自己的应用程序，通常为*.c、*.h、*.asm。

  本书将对应地介绍：ARM 的体系结构；ARM 的指令系统；ARM 汇编语言，对应于*.asm 文件；ARM C 语言的独到部分(与标准 C 相同的部分这里不做介绍)，

对应于*.c；ARM 的编程指南；ARM 的编译器使用。

本书还将介绍 ARM 公司提供的集成开发环境 CodeWarrior IDE 的使用方法。

- Image 文件各部分的组织方法以及在内存中的安排。

  本书将对应地介绍 ELF 格式的映像文件的组成、ARM 连接器的使用、程序在 ROM 中的存放技术。

- Image 文件中各部分的功能。

  本书将对应地介绍一个嵌入式系统各部分的功能，着重介绍系统启动部分的设计。这部分是嵌入式系统涉及的难点，将通过一些实例来介绍。

- Image 的调试。

  本书主要介绍 ARM 公司的调试工具 ADW 的使用方法。同时将介绍嵌入式系统的基本调试方法。

3. 本书的结构安排

全书包括 14 章。各章主要内容说明如下。

第 1 章简要介绍 ARM 公司的情况以及基于 ARM 技术的嵌入式系统的应用情况，比较详细地介绍当前 ARM 体系结构的主要版本，简要介绍目前 ARM 处理器的种类及其主要特点。通过这一章的介绍，读者可以对 ARM 技术有一个总体的了解。

第 2 章介绍 ARM 编程模型的基本知识。主要包括 ARM 处理器模式、ARM 体系中的寄存器及其使用方式、ARM 体系中异常中断处理的基本概念以及 ARM 体系中存储访问的基本知识。通过这一章的介绍，读者将了解 ARM 编程模型的基本知识，为详细了解 ARM 程序设计的各项技术打好基础。

第 3 章详细介绍 ARM 体系的指令系统以及寻址方式。将介绍 ARM 指令集和 Thumb 指令集各自的应用领域。虽然没有详细介绍 Thumb 指令集，但并不是因为 Thumb 指令集不重要，而是因为从功能上来讲，它是 ARM 指令集的子集，在了解 ARM 指令集的基础上很容易理解 Thumb 指令。介绍各指令的编码格式、语法格式、执行的操作以及应用方法。最后将介绍一些常用的 ARM 指令代码段，帮助用户进一步理解各指令的用法，积累一些 ARM 代码设计的基本方法。

第 4 章介绍 ARM 汇编语言程序设计的基本方法以及 ARM 汇编器 armasm 的使用方法。其中包括 ARM 汇编语言中的伪操作(Directives)、宏指令(Pseudo-instruction)、汇编语言格式、armasm 的使用方法以及一些汇编语言程序示例。通过这些介绍，读者可以掌握 ARM 汇编语言设计的方法。

第 5 章介绍 ARM 体系的存储系统。在一个嵌入式系统中，存储系统是非常重要的一部分。这里将介绍 ARM 体系中用于存储管理的协处理器 CP15、存储管理单元 MMU、写缓冲以及 Cache、快速上下文切换技术，还将介绍有关存储系统的程序设计。并以 LinkUp 公司 ARM 处理器芯片 L7210 中的存储系统为例，介绍 ARM 存储系统的设计技术。其中没有介绍存储保护单元 MPU，这是因为 MPU 更简单，而 MMU 的应用更为广泛。该章对于虚拟存储技术、缓冲技术以及 Cache 技术都将做比较详细的介绍，使那些从事基于低端单

片机应用的开发人员更容易理解 ARM 体系中存储系统的设计技术。

第 6 章介绍 ARM/Thumb 过程调用的标准。为了能使单独编译的 C 语言程序和汇编程序之间能够相互调用，必须为子程序间的调用制定一定的规则。ATPCS 规定了 ARM 程序和 Thumb 程序中子程序调用的基本规则。这些基本规则包括子程序调用过程中寄存器的使用规则、数据栈的使用规则和参数的传递规则等。同时，该章还将介绍支持数据栈检查的 ATPCS 以及与代码/数据位置无关的 ATPCS。

第 7 章介绍 ARM 程序和 Thumb 程序混合使用的方法。如果程序遵守支持 ARM 程序和 Thumb 程序混合使用的 ATPCS，则程序中的 ARM 子程序和 Thumb 子程序可以相互调用。对于 C/C++源程序而言，只要在编译时指定-apcs /interwork 选项，编译器生成的代码就遵守支持 ARM 程序和 Thumb 程序混合使用的 ATPCS。而对于汇编源程序而言，用户必须保证编写的代码遵守支持 ARM 程序和 Thumb 程序混合使用的 ATPCS。该章将介绍相关的选项和编程技术。

第 8 章介绍 ARM 汇编程序以及 C/C++程序之间相互调用的技术。其中将介绍 C 编译器中内嵌的汇编器的使用方法。

第 9 章详细介绍 ARM 体系中的异常中断技术。其中包括异常中断处理的处理过程，各种异常中断处理的进入和返回机制，在应用程序中使用异常中断处理的方法以及各种异常中断的详细使用技术。

第 10 章主要介绍 ARM 体系中 C/C++语言程序设计的基本知识。其中包括 ARM C/C++语言的一些特性、ARM C/C++编译器的使用方法，以及 ARM C/C++运行时库的使用方法。通过这些介绍，可以使读者掌握开发嵌入式 C/C++应用程序的基本知识和方法，进一步了解嵌入式应用系统的特点。

第 11 章介绍如何由目标文件以及库文件得到可执行的映像文件。其中包括 ELF 格式的可执行映像文件的组成、ARM 连接器的使用方法，以及连接过程所执行的各种操作。最后通过一些实例介绍在映像文件中各部分内容的地址映射关系。

第 12 章介绍嵌入式应用程序设计的基本知识，然后通过几个示例具体说明嵌入式应用程序的设计方法。对于每个示例，不仅详细介绍程序设计的要点，而且介绍如何使用 ARM 开发工具编译、连接这些程序，生成映像文件。该章是对前面几章知识的综合应用。

第 13 章介绍 CodeWarrior IDE 集成开发环境的使用方法。其中着重介绍在 CodeWarrior IDE 中工程项目的使用方法，以及生成目标的设置方法。这些知识是使用 CodeWarrior IDE 进行应用程序开发时最为重要的部分。

第 14 章介绍 ARM 体系的调试系统和 ARM 公司的高性能调试工具 ADW 的使用方法。ADW 的功能非常多，本书并不是一本专门介绍 ADW 的书。因而只是介绍其中的一些基本功能和嵌入式系统的基本调试方法。

4. 阅读本书时的注意事项

在嵌入式应用系统的开发技术中，涉及很多名词术语，本书主要使用在国内单片机技术领域中通用的一些名词术语，但仍有一些 ARM 体系中特有的名词术语较难翻译。本书

中有很多词是按照其技术含义来表达的，而不是按单词直接翻译。同时，对于一些名词术语，本书在括号内给出了其英文名称，便于读者理解。

对于 ARM 指令系统，本书给出了详细的介绍，是希望该部分能作为编写 ARM 汇编程序的开发人员的参考资料，提高开发人员的工作效率。

本书在编写过程中，得到了 ARM(上海)的大力支持，在此表示衷心的感谢。

目　录

第 1 章　ARM 概述及其基本编程模型

ARM 公司既不生产芯片也不销售芯片，它只出售芯片技术授权。采用 ARM 技术 IP 核的微处理器遍及汽车、消费电子、成像、工业控制、海量存储、网络、安保和无线等各类产品市场。目前，基于 ARM 技术的处理器已经占据了 32 位 RISC 芯片 75%的市场份额。可以说，ARM 技术几乎无处不在。

1990 年 11 月，ARM 公司在英国剑桥的一个谷仓里成立，最初只有 12 人。经过 20 多年的发展，ARM 公司已经拥有 1700 多名员工，其中 60%以上都从事研发工作。ARM 公司在全世界多个国家和地区设有分公司。

ARM 拥有广泛的全球技术合作伙伴，这其中包括领先的半导体系统厂商、实时操作系统(RTOS)开发商、电子设计自动化和工具供应商、应用软件公司、芯片制造商和设计中心。

ARM 合作伙伴包括了许多世界顶级的半导体公司。目前世界前 5 家大半导体公司全都使用了 ARM 的技术授权，而前 10 家大半导体公司中有 9 家，前 25 家大半导体公司中有 23 家都采用了 ARM 的技术授权。全世界有 70 多家公司生产 ARM 芯片。

ARM 技术具有很高的性能和功效，因而容易被厂商接受。同时，合作伙伴的增多，可获得更多的第三方工具、制造和软件支持，这又会使整个系统成本降低，让产品进入市场的时间加快，从而具有更大的竞争优势。

1.1　ARM 技术的应用领域及其特点

(1) ARM 技术的 IP 核在下列领域已经取得或正在取得很大的成功。

- 无线设备

  超过 85%的无线设备(手机等)都采用了 ARM 技术，在向 3G 升级的过程中，ARM 也地位稳固。在 PDA 一类的无线设备中，ARM 针对视频流进行了优化，并获得广泛的支持。

- 蓝牙技术

  ARM 已经为蓝牙的推广做好了准备，有 20 多家公司的元器件产品采用了 ARM 技术，如爱立信、英特尔、科胜讯、朗讯、阿尔卡特、菲利浦和德州仪器等。

- 联网

  随着宽带接入市场的成长，采用 ARM 技术的 ADSL 芯片组获得了竞争优势。

- 消费电子

  这是增长迅速的市场。ARM 技术在数字音频播放器、数字机顶盒和游戏机等产品中应用广泛。

- 汽车

  汽车上使用的 ARM 一直是厂家设计实验的热点，包括驾驶、安全和车载娱乐等各种功能在内的设备可采用若干个 ARM 微处理器统一实现。

- 海量存储设备
  采用 ARM 技术的存储产品包括硬盘系列、微型闪存卡和可读写光盘等，已经投入生产，并且将会有更加先进的产品。
- 成像
  包含 ARM 技术的相机和打印机。
- 安全产品
  在 GSM 和 3G 手机中的 32 位 SIM 智能卡。

(2) ARM 芯片具有 RISC 体系的一般特点。例如：

- 具有大量的寄存器。
- 绝大多数操作都在寄存器中进行，通过 Load/Store 的体系结构在内存和寄存器之间传递数据。
- 寻址方式简单。
- 采用固定长度的指令格式。

(3) 除此之外，ARM 体系采用了一些特别的技术，在保证高性能的同时，尽量减小芯片体积，减低芯片的功耗。这些技术包括：

- 在同一条数据处理指令中包含算术逻辑处理单元处理和移位处理。
- 使用地址自动增加(减少)来优化程序中的循环处理。
- Load/Store 指令可以批量传输数据，从而提高了数据传输的效率。
- 所有指令都可以根据前面指令执行的结果，来决定是否执行，以提高指令执行的效率。

1.2 ARM 体系结构的版本及命名方法

迄今为止，ARM 体系结构已经定义了多个版本，从低版本到高版本，ARM 体系的指令集功能不断扩大。同时，各版本中还有一些变种，这些变种定义了该版本指令集中不同的功能。ARM 处理器系列中的各种处理器，所采用的实现技术各不相同，性能差别很大，应用场合也有所不同，但是只要它们支持相同的 ARM 体系版本，基于它们的应用软件将是兼容的。

本节主要以前 6 个版本为例，介绍 ARM 体系结构中不同版本指令集的特点，以及各版本包含的一些变种的特点。

1.2.1 ARM 体系结构的版本

ARM 体系结构的前 6 个版本(V1～V6)的特点如下。

1. 版本 1(V1)

该版本在 ARM1 中实现，但没有在商业产品中使用。它包括下列指令：

- 处理乘法指令之外的基本数据处理指令。
- 基于字节、字和多字的读取和写入指令(Load/Store)。
- 包括子程序调用指令 BL 在内的跳转指令。

- 供操作系统使用的软件中断指令 SWI。

该版本中，地址空间是 26 位，目前已经不再使用。

2. 版本 2(V2)

与版本 1 相比，版本 2 增加了下列指令：

- 乘法指令和乘加法指令。
- 支持协处理器的指令。
- 对于 FIQ 模式，提供了额外的两个备份寄存器。
- SWP 指令及 SWPB 指令。

该版本中，地址空间是 26 位，目前已经不再使用。

3. 版本 3(V3)

版本 3 较以前的版本发生了比较大的变化。主要改进部分如下：

- 处理器的地址空间扩展到了 32 位，但除了版本 3G(版本 3 的一个变种)外的其他版本是向前兼容的，支持 26 位的地址空间。
- 当前程序状态信息从原来的 R15 寄存器移到一个新的寄存器中，新寄存器名为 CPSR(Current Program Status Register，当前程序状态寄存器)。
- 增加了 SPSR(Saved Program Status Register，备份的程序状态寄存器)，用于在程序异常中断程序时，保存被中断程序的程序状态。
- 增加了两种处理器模式，使操作系统代码可以方便地使用数据访问中止异常、指令预取中止异常和未定义指令异常。
- 增加了指令 MRS 和指令 MSR，用于访问 CPSR 寄存器和 SPSR 寄存器。
- 修改了原来的从异常中返回的指令。

4. 版本 4(V4)

与版本 3 相比，版本 4 增加了下列指令：

- 半字的读取和写入指令。
- 读取(Load)带符号的字节和半字数据的指令。
- 增加了 T 变种，可以使处理器状态切换到 Thumb 状态，在该状态下指令集是 16 位的 Thumb 指令集。
- 增加了处理器的特权模式。在该模式下，使用的是用户模式下的寄存器。

另外，在版本 4 中明确定义了哪些指令会引起未定义指令异常。版本 4 不再强制要求与以前的 26 位地址空间兼容。

5. 版本 5(V5)

与版本 4 相比，版本 5 增加或者修改了下列指令：

- 提高了 T 变种中 ARM/Thumb 混合使用的效率。
- 对于 T 变种的指令和非 T 变种的指令使用相同的代码生成技术。

同时，版本 5 还具有以下的特点。

- 增加了前导零计数(Count Leading Zeros)指令，该指令可以使整数除法和中断优先

级排队操作更为有效。

- 增加了软件断点指令。
- 为协处理器设计提供了更多的可选择的指令。
- 更加严格地定义了乘法指令对条件标志位的影响。

6. 版本 6(V6)

ARM 体系版本 6 的主要特点是增加了 SIMD 功能扩展。它适合使用电池供电的高性能的便携式设备。这些设备一方面需要处理器提供高性能，另一方面又需要功耗很低。SIMD 功能扩展为包括音频/视频处理在内的应用系统提供了优化功能，可以使音频/视频处理性能提高 4 倍。

1.2.2 ARM 体系的变种

这里将某些特定功能称为 ARM 体系的某种变种(variant)，例如支持 Thumb 指令集，称为 T 变种。目前 ARM 定义了一些变种。

1. Thumb 指令集(T 变种)

Thumb 指令集是将 ARM 指令集的一个子集重新编码而形成的一个指令集。ARM 指令长度为 32 位，Thumb 指令长度为 16 位。这样，使用 Thumb 指令集可以得到密度更高的代码，这对于需要严格控制产品成本的设计是非常有意义的。

(1) 与 ARM 指令集相比，Thumb 指令集具有以下局限：

- 完成相同的操作，Thumb 指令通常需要更多的指令。因此，在对系统运行时间要求苛刻的应用场合，ARM 指令集更为适合。
- Thumb 指令集没有包含进行异常处理时需要的一些指令，因此在异常中断的低级处理中，还是需要使用 ARM 指令。这种限制决定了 Thumb 指令需要与 ARM 指令配合使用。对于支持 Thumb 指令的 ARM 体系版本，使用字符 T 来表示。

(2) 相关 Thumb 指令集版本的示例如下：

- Thumb 指令集版本 1。用于 ARM 体系版本 4 的 T 变种。
- Thumb 指令集版本 2。用于 ARM 体系版本 5 的 T 变种。

(3) 与版本 1 相比，Thumb 指令集的版本 2 具有以下特点：

- 通过增加指令和对已有指令的修改，提高 ARM 指令和 Thumb 指令混合使用时的效率。
- 增加了软件断点指令。
- 更加严格地定义了 Thumb 乘法指令对条件标志位的影响。

这些特点与 ARM 体系版本 4 到版本 5 进行的扩展密切相关。实际上，通常并不使用 Thumb 版本号，而是使用相应的 ARM 版本号。

2. 长乘法指令(M 变种)

M 变种增加了两条用于进行长乘法操作的 ARM 指令。其中一条指令用于实现 32 位整数乘以 32 位整数，生成 64 位整数的长乘法操作；另一条指令用于实现 32 位整数乘以 32 位整数，然后再加上 32 位整数，生成 64 位整数的长乘加操作。在需要这种长乘法的应用

场合 M 变种很适合。

然而，在有些应用场合中，乘法操作的性能并不重要，但对于尺寸要求很苛刻，在系统实现时就不适合增加 M 变种的功能。

M 变种首先在 ARM 体系版本 3 中引入。如果没有上述的设计方面的限制，在 ARM 体系版本 4 及其以后的版本中，M 变种是系统中的标准部分。对于支持长乘法 ARM 指令的 ARM 体系版本，使用字符 M 来表示。

3. 增强型 DSP 指令(E 变种)

E 变种包含了一些附加的指令，这些指令用于增强处理器对一些典型的 DSP 算法的处理性能。主要包括：

- 几条新的实现 16 位数据乘法和乘加操作的指令。
- 实现饱和的带符号数的加减法操作的指令。所谓饱和的带符号数的加减法操作，是指在加减法操作溢出时，结果并不进行卷绕(Wrapping Around)，而是使用最大的整数或最小的负数来表示。
- 进行双字数据操作的指令，包括双字读取指令 LDRD、双字写入指令 STRD 和协处理器的寄存器传输指令 MCRR/MRRC。
- Cache 预取指令 PLD。

E 变种首先在 ARM 体系版本 5T 中使用，用字符 E 表示。在 ARM 体系版本 5 以前的版本中，以及在非 M 变种和非 T 变种的版本中，E 变种是无效的。

在早期的一些 E 变种中，未包含双字读取指令 LDRD、双字写入指令 STRD、协处理器的寄存器传输指令 MCRR/MRRC 以及 Cache 预取指令 PLD。这种 E 变种记作 ExP，其中 x 表示缺少，P 代表上述的几种指令。

4. Java 加速器 Jazelle(J 变种)

ARM 的 Jazelle 技术将 Java 的优势和先进的 32 位 RISC 芯片完美地结合在一起。Jazelle 技术提供了 Java 加速功能，可以得到比普通 Java 虚拟机高得多的性能。与普通的 Java 虚拟机相比，Jazelle 使 Java 代码运行速度提高了 8 倍，而功耗降低了 80%。

Jazelle 技术使得程序员可以在一个单独的处理器上同时运行 Java 应用程序、已经建立好的操作系统、中间件以及其他的应用程序。与使用协处理器和双处理器相比，使用单独的处理器可以在提供高性能的同时，保证低功耗和低成本。

J 变种首先在 ARM 体系版本 4TEJ 中使用，用字符 J 表示 J 变种。

5. ARM 媒体功能扩展(SIMD 变种)

ARM 媒体功能扩展为嵌入式应用系统提供了高性能的音频/视频处理技术。

新一代的 Internet 应用系统、移动电话和 PDA 等设备需要提供高性能的流式媒体，包括音频和视频等；而且这些设备需要提供更加人性化的界面，包括语音识别和手写输入识别等。这样，就要求处理器能够提供很强的数字信号处理能力，同时还必须保持低功耗，以延长电池的使用时间。ARM 的 SIMD 媒体功能扩展为这些应用系统提供了解决方案。它为包括音频/视频处理在内的应用系统提供了优化功能，可以使音频/视频处理性能提高 4 倍。