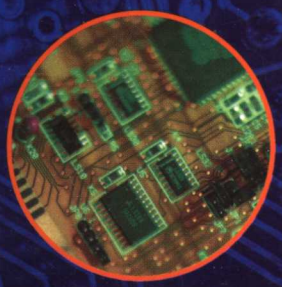
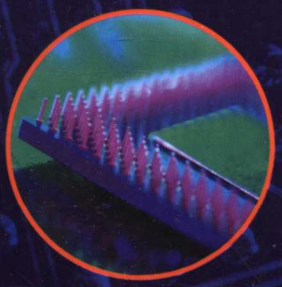
嵌入式系统开发技术丛书

ARM 系统开发

从实践到提高

丁 峰 徐 靖 鲁 立 编著





- 全面介绍了基于μc/os-II操作系统进行嵌入式开发的基本原理和方法
- 详细阐述了μc/os-II在ARM上的移植，基于μc/os-II应用程序的开发及ARM硬件接口驱动设计的过程与方法
- 内容翔实，密切联系实际，便于读者在最短的时间内提高独立开发的能力
- 理论与实践紧密结合，是从事ARM系统开发的初学者、高等院校相关专业学生的理想参考书

中国电力出版社
www.infopower.com.cn

嵌入式系统开发技术丛书

ARM 系统开发

从实践到提高

丁 峰 徐 靖 鲁 立 编著

中国电力出版社
www.infopower.com.cn

内 容 简 介

本书以 ARM 嵌入式微处理器、uC/OS-II 实时操作系统为主，从 ARM 体系结构、开发环境等基础知识开始，重点介绍了 ARM 硬件接口驱动的设计、uC/OS-II 在 ARM 上的移植及基于 uC/OS-II 的应用程序的开发，并详细分析了 uC/OS-II 内核源代码，包括任务机制、通信机制等，以帮助读者掌握嵌入式系统开发的基本流程和软硬件设计方法。

本书理论与实践紧密结合，是从事嵌入式技术相关工作的工程技术人员、高等院校相关专业的学生及相关培训班学员的理想选择。

图书在版编目（CIP）数据

ARM系统开发从实践到提高 / 丁峰，徐靖，鲁立编著. —北京：中国电力出版社，2007.5
（嵌入式系统开发技术丛书）
ISBN 978-7-5083-5371-5

Ⅰ.A… Ⅱ.①丁…②徐…③鲁… Ⅲ.微处理器，ARM－系统设计 Ⅳ.TP332

中国版本图书馆CIP数据核字（2007）第040595号

责任编辑：夏华香
责任校对：崔燕菊
责任印制：李文志

书　　名：ARM 系统开发从实践到提高
编　　著：丁峰　徐靖　鲁立
出版发行：中国电力出版社
地址：北京市三里河路 6 号　邮政编码：100044
电话：（010）68362602　传真：（010）68316497
印　　刷：航远印刷有限公司
开本尺寸：185 × 260　印　张：18　字　数：421 千字
书　　号：ISBN 978-7-5083-5371-5
版　　次：2007 年 7 月北京第 1 版
印　　次：2007 年 7 月第 1 次印刷
印　　数：0001—4000
定　　价：29.80 元

丛书序

这是一个令人激动的时代，嵌入式系统的应用深入到了整个社会。环顾四周，总会发现嵌入式系统正在深刻地改变着我们的生活。十年前，一个数字寻呼机就能彰显用户的身份，而如今，手机已经成为我们生活的必备品，不仅能打电话和发短信，还能拍照片、玩游戏、看电影；十年前，我们得冲洗胶卷后才能看到拍摄的相片，发现拍摄的失误时往往已经为时已晚，十年后的今天，数码相机已经深入到每个家庭，它不仅可以让我们马上看见拍摄的照片，而且不使用胶卷，数字信息也更易于传递。这一切的变化，都得益于嵌入式系统发展所起的举足轻重的作用。

▶ 关于嵌入式系统

嵌入式系统是一种专用的计算机系统，它根据应用要求，把相应的计算机直接嵌入到应用系统中。嵌入式系统涉及到了当前信息技术最新成果的方方面面，融合了计算机软/硬件技术、通信技术、半导体微电子技术等，设计与制造相当不易。嵌入式系统的重要性与日俱增，而这方面的人才却十分紧缺。

▶ 丛书的内容

本丛书包括 ARM 系统开发、ASIC 芯片设计、嵌入式系统无线互联、FPGA 应用开发、SoC 系统开发和 DSP 应用开发 6 个专题，共分 6 本书。每本书以一个或几个案例为基础展开，并提供了大量的源代码，便于读者学习和使用。

▶ 丛书特点

在编写本丛书的过程中，力求以简明扼要的语言，重点突出地描述清楚基本概念和开发流程。在本丛书的内容中，融入了作者以往的研发经验和科研工作中的实例。通过案例分析与设计，逐步完成某个完整嵌入式系统的设计。案例剖析不仅能够提高读者的阅读兴趣，克服对复杂问题的恐惧心理，而且能够使设计思想与设计过程更容易理解，帮助读者尽快上手训练。

⊙ 作者优势

我们组织了在嵌入式领域具有长期开发经验的研究人员与工程师编写了本套丛书，他们都有丰富的电子产品研发编程经验，在专业期刊上发表过很多学术论文，在实际开发过程中积累了丰富的项目实践经验，相信他们提供的应用方法和技巧能有效地帮助读者提高实际操作能力。

⊙ 读者对象

本丛书可作为有关科学研究与产品开发人员的工作学习参考书，也可作为高等院校相关专业本科生与研究生的教学参考书。

⊙ 其他声明

尽管作者做了很大努力，但限于水平和时间，错误和不妥之处在所难免，敬请读者批评指正，我们的联系方式是 liu_chi@cepp.com.cn。同时希望各行业从事嵌入式系统及相关技术工作的专家、学者、工程技术人员借此机会积极参与图书的选题开发和编写工作，将您在工作实践中获得的丰富经验总结出来，共同推进我国嵌入式系统技术的发展！

丛书编委会

2007 年 5 月

前　言

计算机的发展已经进入后 PC 时代，后 PC 时代是嵌入式系统和网络发展的时代。在嵌入式系统开发方面，最核心的技术就是处理器芯片和嵌入式操作系统。

如今，嵌入式系统开发设计人员越来越多地选择特定的处理器内核和配套的工具、库及现成的组件来缩短开发周期。十多年来，ARM 体系结构已经成为世界上最受欢迎的 32 位 RISC 架构嵌入式微处理器，已广泛应用于各种移动电话、通信设备、消费电子和工业控制产品中。

小型实时嵌入式操作系统 uC/OS-II 具有低成本、易控制、小规模、高性能的特性，而且很适合初学者学习，有相当好的发展前景。

本书的 3 位作者都是长期从事于嵌入式系统开发，而且经验丰富的科研工作者，曾经也是徘徊在入门阶段的菜鸟，了解初学者的需求，希望能把自己的经验奉献给读者。而且市场上很少有既有对嵌入式开发实例介绍的，又有对 uC/OS-II 内核代码详细分析的书，本书较全面地介绍了基于 uC/OS-II 操作系统进行嵌入式开发的所有相关内容，可以让初学者快速入门，也可以让入门者再上一个台阶，并为那些准备基于 linux 开发的读者做好铺垫。

笔者认为，学习嵌入式系统开发首先得有自己的开发板或实验板，动手很重要。在学习过程中还要注意培养自己的兴趣，并且要有阶段性的成就感才能让你不断前进，所以得勤奋学习和训练。只有勤奋才能让你有所收获，有一定的成就感，接着让你产生浓厚的兴趣，形成这种良性循环，不断进步。

本书大部分是一些实验原理的介绍和代码分析，读者可以结合自己的板子，在弄懂实验原理的基础上调试程序。

本书主要讲解了 ARM 硬件接口设计、基于 uC/OS-II 的应用程序设计，并详细分析了 uC/OS-II 的内核源代码。本书由丁峰、鲁立、徐靖共同编写，借鉴了很多北京博创兴业科技有限公司 2410-s 开发板的资料，在此表示衷心地感谢！

限于作者知识水平，书中不足之处恳请各位读者和专家指正。

作者

2007 年 3 月

目　录

第 1 章 ▸▸

1 ARM 处理器基础

1.1 ARM 概述

ARM（Advanced RISC Machines，ARM），既可以认为是一个公司的名字，也可以认为是对一类微处理器的通称，还可以认为是一种技术的名字。1991 年 ARM 公司成立于英国剑桥大学，主要出售芯片设计技术的授权。目前，采用 ARM 技术知识产权（IP）核的微处理器，即通常所说的 ARM 微处理器，已遍及工业控制、消费类电子产品、通信系统、网络系统、无线系统等各类产品市场，基于 ARM 技术的微处理器应用约占据了 32 位 RISC 微处理器 75%以上的市场份额，ARM 技术正在逐步渗入到人们生活的各个方面。ARM 公司是专门从事基于 RISC 技术芯片设计开发的公司，作为知识产权供应商，本身不直接从事芯片生产，靠转让设计许可由合作公司生产各具特色的芯片，世界各大半导体生产商从 ARM 公司购买其设计的 ARM 微处理器核，根据各自不同的应用领域，加入适当的外围电路，从而形成自己的 ARM 微处理器芯片进入市场。目前，全世界有几十家大的半导体公司都使用 ARM 公司的授权，因此既使得 ARM 技术获得更多的第三方工具、制造、软件的支持，又使整个系统成本降低，使产品更容易进入市场被消费者所接受，更具有竞争力。

1.1.1 ARM 应用领域

ARM 微处理器及技术的应用已经深入到工业控制领域、无线通信领域、网络应用、消费类电子产品、成像和安全产品等。作为 32 位的 RISC 架构，基于 ARM 核的微控制器芯片不但占据了高端微控制器市场的大部分份额，同时也逐渐向低端微控制器应用领域扩展，ARM 微控制器的低功耗、高性价比，向传统的 8 位/16 位微控制器提出了挑战；目前已有超过 85%的无线通信设备采用了 ARM 技术，ARM 以其高性能和低成本，在该领域的地位日益巩固；随着宽带技术的推广，采用 ARM 技术的 ADSL 芯片正逐步获得竞争优势。此外，ARM 在语音及视频处理上进行了优化，并获得广泛支持，也对 DSP 的应用领域提出了挑战，ARM 技术在目前流行的数字音频播放器、数字机顶盒和游戏机中得到广泛应用；现在流行的数码相机和打印机中绝大部分采用 ARM 技术，手机中的 32 位 SIM 智能卡也采用了 ARM 技术。ARM 微处理器及技术还应用到许多不同的领域，并会在将来取得更加广泛的应用。

1.1.2 RISC 体系结构

传统的 CISC（Complex Instruction Set Computer，复杂指令集计算机）结构有其固有的缺点，随着计算机技术的发展而不断引入新的复杂的指令集，为支持这些新增的指令，计算机的体系结构会越来越复杂，然而，在 CISC 指令集的各种指令中，其使用频率却相差悬殊，大约有 20%的指令会被反复使用，占整个程序代码的 80%。而余下的 80%的指令却不经常使用，在程序设计中只占 20%，显然，这种结构是不太合理的。1979 年美国加州大学伯克利分校提出了 RISC（Reduced Instruction Set Computer，精简指令集计算机）的概念，RISC 并非只是简单地去减少指令，而是把着眼点放在了如何使计算机的结构更加简单合理地提高运算速度上。RISC 结构优先选取使用频率最高的简单指令，避免复杂指令；将指令长度固定，指令格式和寻地方式种类减少；以控制逻辑为主，不用或少用微码控制等措施来达到上述目的。

ARM 微处理器采用 RISC 结构，一般具有如下特点：体积小、低功耗、低成本、高性能；支持 Thumb（16 位）/ARM（32 位）双指令集，能很好地兼容 8 位/16 位器件；大量使用寄存器，指令执行速度更快；大多数数据操作都在寄存器中完成；寻址方式灵活简单，执行效率高；指令长度固定。

1.1.3 ARM 分类系列

ARM 微处理器系列目前包括 ARM7、ARM9、ARM9E、ARM10E、SecurCore、Intel 的 Xscale、Intel 的 StrongARM，以及其他厂商基于 ARM 体系结构的处理器，除了具有 ARM 体系结构的共同特点以外，每一个系列的 ARM 微处理器都有各自的特点和应用领域。其中，ARM7、ARM9、ARM9E 和 ARM10 为 4 个通用处理器系列，每一个系列提供一套相对独特的性能来满足不同应用领域的需求。SecurCore 系列专门为安全要求较高的应用而设计。

ARM7 系列微处理器为低功耗的 32 位 RISC 处理器，最适合用于对价位和功耗要求较高的消费类应用。ARM7 微处理器系列具有如下特点：具有嵌入式 ICE－RT 逻辑，调试开发方便；极低的功耗，适合对功耗要求较高的应用，如便携式产品；能够提供 0.9MIPS/MHz 的三级流水线结构；代码密度高并兼容 16 位的 Thumb 指令集；对操作系统的支持广泛，包括 Windows CE、Linux、Palm OS 等；指令系统与 ARM9 系列、ARM9E 系列和 ARM10E 系列兼容，便于用户的产品升级换代；主频最高可达 130MIPS，高速的运算处理能力能胜任绝大多数的复杂应用。ARM7 系列微处理器的主要应用领域为工业控制、Internet 设备、网络和调制解调器设备、移动电话等多种多媒体和嵌入式应用。ARM7 系列微处理器包括如下几种类型的核：ARM7TDMI、ARM7TDMI-S、ARM720T、ARM7EJ。其中，ARM7TMDI 是目前使用最广泛的 32 位嵌入式 RISC 处理器，属低端 ARM 处理器核。TDMI 的基本含义为 T，支持 16 位压缩指令集 Thumb；D 支持片上 Debug；M 内嵌硬件乘法器（Multiplier）；I 嵌入式 ICE，支持片上断点和调试点。

ARM9 系列微处理器在高性能和低功耗特性方面提供最佳的性能。具有以下特点：5 级整数流水线，指令执行效率更高；提供 1.1MIPS/MHz 的哈佛结构；支持 32 位 ARM 指令集和 16 位 Thumb 指令集；支持 32 位的高速 AMBA 总线接口；全性能的 MMU，支持 Windows CE、Linux、Palm OS 等多种主流嵌入式操作系统；MPU 支持实时操作系统；支持数据 Cache

和指令 Cache，具有更高的指令和数据处理能力。ARM9 系列微处理器主要应用于无线设备、仪器仪表、安全系统、机顶盒、高端打印机、数字照相机和数字摄像机等。ARM9 系列微处理器包含 ARM920T、ARM922T 和 ARM940T 三种类型，以适用于不同的应用场合。

ARM10E 系列微处理器具有高性能、低功耗的特点，由于采用了新的体系结构，与同等的 ARM9 器件相比较，在同样的时钟频率下，性能提高了近 50%，同时，ARM10E 系列微处理器采用了两种先进的节能方式，使其功耗极低。ARM10E 系列微处理器的主要特点如下：支持 DSP 指令集，适合于需要高速数字信号处理的场合；6 级整数流水线，指令执行效率更高；支持 32 位 ARM 指令集和 16 位 Thumb 指令集；支持 32 位的高速 AMBA 总线接口；支持 VFP10 浮点处理协处理器；全性能的 MMU，支持 Windows CE、Linux、Palm OS 等多种主流嵌入式操作系统；支持数据 Cache 和指令 Cache，具有更高的指令和数据处理能力；主频最高可达 400MIPS；内嵌并行读/写操作部件。ARM10E 系列微处理器主要应用于下一代无线设备、数字消费品、成像设备、工业控制、通信和信息系统等领域。ARM10E 系列微处理器包含 ARM1020E、ARM1022E 和 ARM1026EJ-S 三种类型，以适用于不同的应用场合。

SecurCore 系列微处理器专为安全需要而设计，提供了完善的 32 位 RISC 技术的安全解决方案，因此，SecurCore 系列微处理器除了具有 ARM 体系结构的低功耗、高性能的特点外，还具有其独特的优势，即提供了对安全解决方案的支持。SecurCore 系列微处理器除了具有 ARM 体系结构各种主要特点外，还在系统安全方面具有如下特点：带有灵活的保护单元，以确保操作系统和应用数据的安全；用软内核技术，防止外部对其进行扫描探测；集成用户自己的安全特性和其他协处理器。SecurCore 系列微处理器主要应用于一些对安全性要求较高的应用产品及应用系统，如电子商务、电子政务、电子银行业务、网络和认证系统等领域。SecurCore 系列微处理器包含 SecurCore SC100、SecurCore SC110、SecurCore SC200 和 SecurCore SC210 四种类型，以适用于不同的应用场合。

Intel StrongARM SA-1100 处理器是采用 ARM 体系结构高度集成的 32 位 RISC 微处理器。它融合了 Intel 公司的设计和处理技术以及 ARM 体系结构的电源效率，采用在软件上兼容 ARMv4 体系结构，同时采用具有 Intel 技术优点的体系结构。Intel StrongARM 处理器是便携式通信产品和消费类电子产品的理想选择，已成功应用于多家公司的掌上电脑系列产品。

Xscale 处理器是基于 ARMv5TE 体系结构的解决方案，是一款全性能、高性价比、低功耗的处理器。它支持 16 位的 Thumb 指令和 DSP 指令集，已使用在数字移动电话、个人数字助理和网络产品等场合。Xscale 处理器是 Intel 目前主要推广的一款 ARM 微处理器。

1.2　ARM 处理器的寄存器

ARM 处理器共有 37 个寄存器，其中包括 31 个通用寄存器和 6 个状态寄存器，这些寄存器都是 32 位寄存器。

ARM 处理器共有 7 种不同的处理器模式，每一种模式中都有一组相应的寄存器组。在任何时刻，可见的寄存器包括 15 个通用寄存器（R0～R14），一个或两个状态寄存器及程序

计数器（PC）。在所有的寄存器中，有些是各模式共用一个物理寄存器，有一些寄存器各模式拥有自己独立的物理寄存器。

1.2.1 通用寄存器

通用寄存器分为三类：未备份寄存器、备份寄存器、程序计数器 PC。

（1）未备份寄存器。未备份寄存器包括 R0～R7。对于每一个未备份寄存器来说，所有处理器模式下都是使用同一个物理寄存器。未备份寄存器没有被系统用于特别的用途，任何可采用通用寄存器的场合都可以使用未备份寄存器。

（2）备份寄存器。对于 R8～R12 备份寄存器来说，每个寄存器对应两个不同的物理寄存器。系统未将备份寄存器用于任何的特殊用途，但是当中断处理非常简单，仅仅使用 R8～R14 寄存器时，FIQ 处理程序可以不必执行保存和恢复中断现场的指令，从而可以使中断处理非常迅速。对于 R13、R14 备份寄存器来说，每个寄存器对应 6 个不同的物理寄存器，其中的一个是系统模式和用户模式共用的；另外的 5 个对应于其他的 5 种处理器模式。采用下面的记号来区分各个物理寄存器：R13_<MODE>，其中 MODE 可以是下面几种模式之一：usr、svc、abt、und、irq、fiq。

（3）程序计数器 PC。可以作为一般的通用寄存器使用，但有一些指令在使用 R15 时有一些限制。由于 ARM 采用了流水线处理器机制，当正确读取了 PC 值时，该值为当前指令地址值加上 8 个字节。也就是说，对于 ARM 指令集来说，PC 指向当前指令的下两条指令的地址。由于 ARM 指令是字对齐的，PC 值的第 0 位和第一位总为 0。需要注意的是，当使用 str/stm 保存 R15 时，保存的可能是当前指令地址值加 8 个字节，也可能保存的是当前指令地址值加 12 个字节。到底是哪种方式取决于芯片的具体设计。对于用户来说，尽量避免使用 STR/STM 指令来保存 R15 的值。当成功地向 R15 写入一个数值时，程序将跳转到该地址执行。由于 ARM 指令是字对齐的，写入 R15 的值应满足 bits[1:0]为 0b00，具体要求 arm 版本有所不同：对于 arm3 以及更低的版本，写入 R15 的地址值 bits[1:0]被忽略，即写入 r15 的地址值将与 0xFFFF FFFC 做与操作；对于 ARM4 以及更高的版本，程序必须保证写入 R15 的地址值 bits[1:0]为 0b00，否则将产生不可预知的后果；对于 Thumb 指令集来说，指令是半字对齐的，处理器将忽略 bit[0]。

1.2.2 程序状态寄存器

CPSR（当前程序状态寄存器）在任何处理器模式下被访问。它包含了条件标志位、中断禁止位、当前处理器模式标志以及其他的一些控制和状态位。每一种处理器模式下都有一个专用的物理状态寄存器，称为 SPSR（备份程序状态寄存器）。当特定的异常中断发生时，这个寄存器用于存放当前程序状态寄存器的内容。在异常中断退出时，可以用 SPSR 来恢复 CPSR。由于用户模式和系统模式不是异常中断模式，所以没有 SPSR。当用户在用户模式或系统模式访问 SPSR 时，将产生不可预知的后果。

SPSR 和 CPSR 格式相同，CPSR 格式如下所示。

31	30	29	28	27	26	7	6	5	4	3	2	1	0
N	Z	C	V	Q	DNM(RAZ)	I	F	T	M4	M3	M2	M1	M0

（1）条件标志位：

N——本位设置成当前指令运算结果的 bit[31]的值。当两个表示的有符号整数运算时，n=1 表示运算结果为负数，n=0 表示结果为正数或零。

Z——Z=1 表示运算的结果为 0；z=0 表示运算的结果不为 0。对于 CMP 指令，Z=1 表示进行比较的两个数大小相等。

C——下面分 4 种情况讨论 C 的设置方法：

在加法指令中（包括比较指令 CMP），当结果产生了进位时，C=1，表示无符号运算发生上溢出；其他情况 C=0。

在减法指令中（包括减法指令 CMP），当运算中发生错位时，C=0，表示无符号运算数发生下溢出；其他情况下 C=1。

对于包含移位操作的非加减运算指令，C 中包含最后一次溢出的位的数值。

对于其他非加减运算指令，C 位的值通常不受影响。

V——对于加减运算指令，当操作数和运算结果为二进制的补码表示的带符号数时，V=1 表示符号为溢出；通常其他指令不影响 V 位。

Q 标识位

在 ARM V5 的 E 系列处理器中，CPSR 的 bit[27]称为 q 标识位，主要用于指示增强的 DSP 指令是否发生了溢出。同样地，spsr 的 bit[27]位也称为 q 标识位，用于在异常中断发生时保存和恢复 CPSR 中的 Q 标识位。在 ARM V5 以前的版本及 ARM V5 的非 E 系列的处理器中，Q 标识位没有被定义。

（2）CPSR 中的控制位。CPSR 的低八位 I、F、T、M[4:0]统称为控制位。当异常中断发生时这些位发生变化。在特权级的处理器模式下，软件可以修改这些控制位。

中断禁止位：当 I=1 时禁止 IRQ 中断，当 F=1 时禁止 FIQ 中断。

T 控制位：T 控制位用于控制指令执行的状态，即说明本指令是 ARM 指令还是 Thumb 指令。对于 ARM V4 以更高版本的 T 系列 ARM 处理器，T 控制位含义如下：

T=0 表示执行 ARM 指令；T=1 表示执行 Thumb 指令。

对于 ARM V5 以及更高版本的非 T 系列处理器，T 控制位的含义如下。

T=0 表示执行 ARM 指令；

T=1 表示强制下一条执行的指令产生未定指令中断。

M 控制位：M 控制位控制处理器模式，具体含义如下：

M[4:0] 处理器模式可访问的寄存器

10000　　user pc,r14~r0,CPSR

10001　　FIQ PC,R14_FIQ-R8_FIQ,R7~R0,CPSR,SPSR_FIQ

10010　　IRQ PC,R14_IRQ-R13_IRQ,R12~R0,CPSR,SPSR_IRQ

10011　　SUPERVISOR PC,R14_SVC-R13_SVC,R12~R0,CPSR,SPSR_SVC

10111　　ABORT PC,R14_ABT-R13_ABT,R12~R0,CPSR,SPSR_ABT

11011　　UNDEFINEED PC,R14_UND-R8_UND,R12~R0,CPSR,SPSR_UND

11111　　SYSTEM PC,R14-R0,CPSR(ARM V4 以及更高版本)

CPSR 中的其他位：这些位用于将来扩展，应用软件不要操作这些位。

1.3 程序控制方式

在 ARM 体系中通常有以下 3 种方式控制程序的执行流程：

在正常执行过程中，每执行一条 ARM 指令，程序计数器（PC）的值加 4 个字节；每执行一条 Thumb 指令，程序计数器寄存器（PC）加 2 个字节。整个过程是按顺序执行的。

跳转指令，程序可以跳转到特定的地址标号处执行，或者跳转到特定的子程序处执行。其中，B 指令用于执行跳转操作；BL 指令在执行跳转操作的同时，保存子程序的返回地址；BX 指令在执行跳转操作的同时，根据目标地址可以将程序切换到 Thumb 状态；BLX 指令执行 3 个操作，跳转到目标地址处执行，保存子程序的返回地址，根据目标地址可以将程序切换到 Thumb 状态。

当异常中断发生时，系统执行完当前指令后，将跳转到相应的异常中断处理程序处执行。当异常中断处理程序执行完成后，程序返回到发生中断指令的下条指令处执行。在进入异常中断处理程序时，要保存被中断程序的执行现场，从异常中断处理程序退出时，要恢复被中断程序的执行现场。

1.4 异常中断的种类

1．异常中断的种类

（1）复位（RESET）

当处理器复位引脚有效时，系统产生复位异常中断，程序跳转到复位异常中断处理程序处执行。复位异常中断通常用在下面几种情况下：系统加电时、系统复位时、跳转到复位中断向量处执行成为软复位。

（2）未定义的指令

当 ARM 处理器或者是系统中的协处理器认为当前指令未定义时，产生未定义的指令异常中断，可以通过改异常中断机制仿真浮点向量运算。

（3）软件中断

这是一个由用户定义的中断指令，可用于用户模式下的程序调用特权操作指令。在实时操作系统中可以通过该机制实现系统功能调用。

（4）指令与取终止（PrefechAbort）

如果处理器预取的指令的地址不存在，或者该地址不允许当前指令访问，当被预取的指令执行时，处理器产生指令预取终止异常中断。

（5）数据访问终止（DATAABORT）

如果数据访问指令的目标地址不存在，或者该地址不允许当前指令访问，处理器产生数据访问终止异常中断。

（6）外部中断请求（IRQ）

当处理器的外部中断请求引脚有效，而且 CPSR 的寄存器的 I 控制位被清除时，处理器产生外部中断请求异常中断。系统中外设通过该异常中断请求处理服务。