周立功单片机公司策划

ARM微控制器基础与实战

周立功　等编著

北京航空航天大学出版社
http://www.buaapress.com.cn

ARM 微控制器基础与实战

周立功　等编著

北京航空航天大学出版社
http://www.buaapress.com.cn

内 容 简 介

以 ARM 控制器 LPC2104 及基于 LPC2104 的开发学习板 EasyARM2104 入手，逐步引导读者掌握使用开发 ARM 的基本知识。本书分 3 个部分：第 1 部分为基础篇，包括第 1 章和第 2 章，主要从 ARM 芯片开发者的角度介绍 ARM7TDMI 的体系结构和指令系统。第 2 部分为实验篇，包括第 3 章、第 4 章和第 5 章，详细介绍 ARM 控制器 LPC2104 的内部结构和开发学习板 EasyARM2104 的硬件结构等知识，给出 LPC2104 的各个功能部件的编程方法，包括汇编代码和 C 语言代码。第 3 部分为提高篇，包括第 6 章～第 13 章，介绍如何把嵌入式实时操作系统 μC/OS－II移植到 LPC2104，在不同情况下如何编译这些代码。然后介绍 LPC2104 的几个重要功能部件在μC/OS－II的驱动程序（中间件），以及其它功能部件在 μC/OS－II 上的使用方法。

本书可作为高等院校嵌入式系统课程的参考用书，以及 ARM 应用技术开发人员的参考手册。

图书在版编目(CIP)数据

ARM 微控制器基础与实战/周立功等编著. —北京：
北京航空航天大学出版社，2003.11
ISBN 7－81077－383－6

Ⅰ.A… Ⅱ.周… Ⅲ.微处理器，ARM—系统设计
Ⅳ. TP332

中国版本图书馆 CIP 数据核字(2003)第 095233 号

ARM 微控制器基础与实战

周立功 等编著

责任编辑 王慕冰

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100083) 发行部电话：010－82317024 传真：010－82328026

http://www.buaapress.com.cn E-mail:bhpress@263.net

河北省涿州市新华印装厂印装 各地书店经销

*

开本：787×1092 1/16 印张：33.75 字数：864 千字

2003 年 11 月第 1 版 2003 年 11 月第 1 次印刷 印数：5 000 册

ISBN 7－81077－383－6 定价：49.00 元

ARM 微控制器快速入门的“敲门砖”

学习与开发的困境

回头来看，我国单片机的普及教育已经搞了这么多年，但为什么还是有很多大学生毕业之后不能立即投入到实际的应用开发工作之中呢？通过严峻的人才需求趋势可以看到：很多电子类专业的大学生毕业之后的出路只有改行，而很多企业却在感叹人才难求。年复一年如此惊人地相似，不得不引起我们对传统的教材和教学方法进行深刻的反思。单片机与嵌入式系统应用技术是一门实战性很强的学科，离开了实践就如无源之水。其实，几乎所有成才的开发工程师都是一边学习、一边实践干出来的，很少有真正的专家，包括我们业界知名的很多专家在内，是通过课堂教出来的。到今天为止，8279、8255 还买得到吗？已经过去这么多年了，学生还在学习那些已经很早就淘汰了的器件。一个大学本科生读了四年大学连一个可靠的复位电路都设计不好，要从可靠性设计出发来设计产品就更无从谈起。由此可以看出，我们的教材与教学方法违背了这门学科成才的规律，我们的大学一定要彻底地改变观念，要尽快地从“教育型”的阴影中走出来，迅速转向“研究型”培养人才的正确轨道上来。

其实，我一直认为企业不是培养人才的地方，这应该是大学的责任。但我们却不得不花很多的精力用于人才的培养，而我们得到的直接好处就是从 2000 年到现在公司几乎没有出现过优秀人才跳槽的现象，就是一般的人才跳槽也微乎其微。究其原因，在于我们一直在尝试如何培养一流人才。通过这些年的努力，我们掌握了快速、恰当地学习的方法和培养优秀而卓越人才的“秘诀”。关键是我们帮助那些年轻人找到了自己的位置和方向，而且我们也实现了自己的愿望，我和他们一样有共同的成就感。基于此，我愿意将我的经验和感想奉献出来与大家共同分享。

谈到学习就不得不谈到人们的思想与观念，这是人才成长与成功的源泉，超前的思想意识、正确的观念、良好的心态加上正确的学习方法就可以达到“不用扬鞭自奋蹄”的崭新境界。所谓教书育人这些根本的东西却常常容易被人所忽视，这看起来似乎是政治老师的责任，其实不然。我们应该将思想和方法教育放在首位，充分调动学习者的积极性，从而化被动为主动。效果最好的教学方法就是在实验室里上课，老师一边讲解理论，学生一边在计算机上操作和验证老师讲解的内容以加深理解。每个学生人手一套开发实验板，一个学期下来至少可以做到考试这门课程不用复习，毕业之后对开发流程和集成开发环境可以达到烂熟于心的程度。

回头来看我国的单片机推广历程，可以设想，如果当初 Intel 公司首先推出的不是 80C31 而是一颗 20 引脚的 Flash 单片机，那么中国的单片机发展之路肯定要顺利得多。回到现实中来看，我国的 ARM 推广和应用技术发展之路与单片机的启蒙是何等惊人地相似。多年来，单片机化电子产品的开发在很大程度上完全取决于开发工程师个人的经验、知识水平和智慧。大多数开发工程师在产品开发过程中的随意性、离散性都比较大，普遍缺乏系统的总结。从根本上来说，谈不上如何自觉地将离散的经验上升到知识的程度，更谈不上有比较清晰的项目管理和软硬件平台。无论是企业的决策者还是开发工程师，计较元器件的成本成为头等大事，还

未做出产品来第一件事情就想到“价格战”，缺乏做精品的意识，缺乏如何保持企业可持续发展的战略眼光和思想，其实这些都是能够看得见的成本。事实上，危害更大的、看不见的“无形成本”却很少有人关注。比如，投入的人力资源是否合理？要投入多少开发费用？员工跳槽以后如何对产品进行升级和继续开发？产品如何快速上市？如何“先知先觉”地设计出高可靠性的产品等等因素，这些都是制约企业可持续发展的根本所在。

回头来看，成千上万的企业家和开发工程师无不感到困惑，又欲罢不能。当我们今天面临由8位进化到32位微控制器市场的时候，我们到底如何克服过去学习和开发中所遇到的困难而少走弯路？

选择一个功能恰当的ARM微控制器

俗话说得好：万丈高楼从地起。也就是说，无论做什么事，打好基础是根本，因此要想彻底掌握32位ARM单片机和嵌入式操作系统应用技术一定要从基础做起。那些内嵌的功能模块如TCP/IP、CAN-bus、A/D转换器、802.11、调制解调器、USB…有如美丽的外衣一样仅仅是虚有其表而已，其实真正的根本还是ARM内核和一个基本的嵌入式操作系统内核，只有把握了“根”，所有其它的问题都将迎刃而解。

通过网站可以看到，关于ARM的开发套件不下20种，到底选择哪一种最合适？我认为，作为一个初学者挑选之前应该目标明确，如何以最少的代价得到最大的收获。因为学好之后将来做哪方面的产品开发到现在还没有完全定性。还有用多长的时间可以快速学会，这也是很重要的，因为起步不顺利会严重打击学习者的积极性。

事实上，有很多从事单片机应用开发的工程师可能一辈子都不会用到串行通信技术，甚至做了多年的开发工作之后还是没有深入地掌握串行接口应用技术，但却丝毫不影响他成为一个实实在在的单片机应用工程师。即使有一天我们在工作中要用到串行通信技术，我们通过长期的学习和开发所积累的“经验与知识”也足以帮助我们快速掌握串行通信技术。所以学习基于ARM内核的单片机应用技术同样如此，我们不应该过早地、过多地将精力用在TCP/IP等复杂功能模块和μClinux等复杂的嵌入式操作系统内核的研究和学习上，这些诱人的“热点”很容易使人迷失方向，进而忽视对基础知识的深入学习，到头来说得头头是道，可真的干起来却离实际的需求差得太远。我们一定要知道，知识是永远也学不完的——学海无涯、人生苦短！我们学习的目的是为了满足应用的需求。那么到底如何在有限的人生历程之中花最少的代价“创造”惊人的价值呢？常言道：伤其十指不如断其一指。即是说：我们做任何事情首先一定要“集中优势兵力”击中要害，然后再根据实际需要“各个击破”，我们常常所说的“不战而胜”、“取法其上、得乎其中”其实讲的就是方法。只要方法得当，掌握ARM和嵌入式操作系统应用技术其实十分简单。因此，以我们的学习体会和开发经验为基础，非常慎重地向读者推荐PHILIPS公司的LPC2106/2105/2104微控制器作为读者入门首选的ARM微控制器。

选择一个简单易懂的嵌入式操作系统

从过去到现在，学习操作系统的人越来越多，而真正深入掌握和搞通操作系统的人却越来越少。很多人从本科到博士都一直在学习操作系统，讲起来也头头是道，考试也不乏满分，可真正要他写一个简单的操作系统却又很困难。为什么？因为他在学习过程中根本就没有得到过很好的训练，大多数人也从来没有自己动手去写过一个哪怕错误百出而真正能够跑起来的

操作系统，尽管大家天天在那里呼吁：我们中国人要有自己知识产权的操作系统，到头来还不是一句空话。学习 ARM 微控制器也同样如此。如果不引起我们整个产业界的重视，如果没有良性循环的引导，我们最终的结局还是落后，那是早已写好的程序——“死循环”。

当然，有人会说我没有必要去研究嵌入式操作系统的源码，同样也能够做出产品，这一点也不假。而事实上，只有真正地掌握嵌入式操作系统才能达到随心所欲的境界。无论嵌入式操作系统多么复杂，无论代码多大，它们的机理都是完全一样的，因此选择哪一个嵌入式操作系统作为快速入门的“敲门砖”就很有讲究了。我认为首先要简单、有源码且稳定；其次，学了之后要能够实用。从目前来看，可以选择的嵌入式操作系统有 μClinux 和 μC/OS－II。对于这两个嵌入式操作系统，不同的人有不同的看法，可以说在各种网站的讨论社区形成了两大阵容，各持己见，对于初学者来说莫衷一是、无所适从。

其实，我并不反对选择使用 μClinux，相反我还提倡使用 μClinux 用于产品，这是未来的发展方向，但现实的情况如何呢？并非每个人和每个单位都有这个能力。首先，由于 μClinux 代码太大，对于大多数人来说，要完全或者基本上读懂不是一件容易的事情。其二，网上确实也有很多公开的中间件源码，但并不保证拿来就可以稳定、可靠地使用起来，还是需要我们进行二次开发，对于大多数人来说，这是一个不小的困难。其三，尽管讨论社区也很活跃，但却很难找到真正的技术支持，这是一个不可回避的事实。因此如果没有实力和优秀的开发团队，切不可贸然选择 μClinux 作为企业的开发平台。是不是开发产品就不能选择 μClinux？也不绝对，在一般的条件下，一般的企业最好选择成熟的 μClinux 开发平台和软硬件模块。商场如战场，一切从头开发不可取，对于企业和个人来说回避风险从来就是第一原则。如果产品开发失败，严重的结果就是企业倒闭；对于个人来说，就失去了一次本来可以成功表现自己的机会。

对于初学者，我慎重地推荐源码公开的 μC/OS－II 嵌入式操作系统。μC/OS－II 已经有很多产品成功使用的案例且得到了美国军方的认证，北京航空航天大学出版社也出版了配套的专著。国内熟悉这个嵌入式操作系统的开发人员特别多，十分容易通过网站上的讨论社区得到帮助，更关键的是，我们很容易通过阅读源码达到彻底掌握嵌入式操作系统的目的。最好的办法是通过阅读 μC/OS－II 之后，自己写一个简单的能够跑起来的 OS，这是我们公司培训相关人才的一个成功的经验。我们在北京航空航天大学出版社出版的《嵌入式实时操作系统 Small RTOS51 原理及应用》专著就是通过学习积累的成果，我们开发 EasyARM2104 开发套件和编写本书的工程师都是这样走过来的。事实证明：通过深入地学习和解剖 μC/OS－II 嵌入式操作系统是迅速培养嵌入式系统应用专家的成功之路。

必须要有深入浅出的配套教材

如果仅仅是购买了一个开发套件而没有相应的配套技术资料，学习的效果可想而知是很差的，因为只有配套的资料才能体现出设计者的原创思想，更为重要的是在学习过程中遇到了困难，配套的技术资料可以帮助加深理解、解决问题，我们可以看看设计者是怎样解释这些问题的。本书是 EasyARM2104 开发套件设计者通过一边做一边积累再加工的原创技术资料，非常实用。其精髓就是使用了简单易懂的语言和例题来解释复杂的技术难题，这也是本书最主要的特色。

良好的技术支持也是必不可少的

当然，我们购买产品不仅得到了硬件，而且也得到了软件，但一定要明白我们之所以选择某个产品，因为更重要的是我们购买了“服务”。如果得不到良好的技术支持，还不如不买，因为遇到了技术难题，读者却找不到专家解答，这是非常痛苦的事情，得到的效果往往是事倍功半。此时此刻，我们浪费的不仅仅是时间，更重要的是失去了稍纵即逝且可以展翅飞翔的机遇。

周立功

2003 年 9 月 3 日

前　言

相信比较敏锐的使用单片机开发项目的电子工程师会感觉到：32 位微控制器的广泛应用已经到了“山雨欲来风满楼”的时候了。这主要由两方面的因素引起，一个因素是需求推动，另一个因素是技术进步拉动。

有人说“需求决定一切”，这话虽然有些武断，但需求的确是很多事情的原动力。目前，随着人们生活水平的提高，人民对生活质量的追求也逐步提高。因此，人们对智能产品的需求增加了（如 PDA、手机和智能家居等），且对智能产品要求提高了（如更具人性化、操作更简单、功能更强大、容错性更好、更安全以及更具个性化等）。另外，随着网络的发展，越来越多的产品需要具有联网功能。这一切需要智能产品具有一个更强劲的“芯”，这是 8 位机很难做到的。同时，由于要生产这样的产品，对生产线的要求也同样提高了（全自动、更精确、效率更高、更安全、容错性更好以及可提供个性化产品等），这也需要一个更强劲的“芯”。同时，对于产品研发的企业来说，有时产品上市的时间很重要。有时上市时间落后于竞争对手就意味着市场的丢失，不过同时质量也要有起码的保证。这样就要求开发者尽量减少重复劳动。建立开发平台是一个很好的产品开发战略思想且具有现实意义，但 8 位系统的可怜资源对建立开发平台十分不利，用 32 位系统就好多了。

在个人电脑行业有著名的“摩尔定律”，它已经主宰个人电脑行业很多年了。在嵌入系统领域虽然“摩尔定律”没有那么明显，但技术的进步已经使 32 位系统不再高高在上，32 位微控制器的价格已经不比 8 位机高多少，有些系统使用 32 位机其整体成本甚至比用 8 位机还要低。这样，使用 32 位系统就没有技术和成本的障碍了。

目前，在 32 位市场上，ARM 扮演着 8 位市场上 8051 的角色，又由于感觉到“山雨欲来风满楼”之势，很多人觉得：应该学一学 ARM 了。但如何学 ARM 是一个很大的问题。目前，很多 ARM 开发板/学习板给人的印象是：学 ARM 必移植 LINUX（或者 uCLinux），必搞 TCP/IP。其实这是不对的。LINUX（μCLinux）、TCP/IP 均是计算机范畴的东西，与 ARM 没有必然的联系，它们本身就是一个十分复杂的体系。一个人要精通任何一种都很困难，又何必与 ARM 搅在一起，人为增加学习 ARM 的难度？而且，如果学习这些不成功，势必会打击学习者的积极性。本书就是以一个简单的 ARM 开发学习板 EasyARM2104 为基础，引导大家学习 ARM 的精髓，让想学 ARM 的人都读得懂、学得会，增强其信心，为以后可学习更复杂的 ARM 打下坚实的基础。

本书的各个章节安排如下：

第 1 章介绍了 ARM7TDMI 和 ARM7TDMI(-S)的体系结构中对程序员有用的部分。体系结构中很多东西仅仅对芯片设计者或编译器开发者或仿真器开发者有用，它们没有包含到第 1 章中。

第 2 章介绍 ARM7TDMI(-S)的指令集且介绍了如何编写汇编程序。ARM7TDMI 和 ARM7TDMI(-S)是基于 ARM 体系结构版本 v4T 的，这一章仅介绍 ARM 体系结构版本 v4T 支持的指令，ARM 体系结构版本 v5 及以上版本扩展的指令没有介绍。汇编程序设计是以

ARM 公司自己开发的基层开发环境 ADS1.2 中附带的汇编器为基础讲解，这一章还给出 C 与汇编混合编程的方法，并给出指令和汇编伪指令速查表。

第 3 章介绍 EasyARM2104 开发学习板的主控芯片 LPC2104 的硬件结构和功能部件，在介绍功能部件原理的同时通过简单的程序片断加深读者对相应功能件的理解。

第 4 章介绍了 EasyARM2104 开发学习板的特点、硬件原理和基本使用方法，相当于 EasyARM2104 开发学习板的简单说明书。阅读完这一章，读者应当知道如何设计基于芯片 LPC2106/2105/2104 的用户板。

第 5 章详细讲述了 LPC2106/2105/2104 的各个功能部件的编程方法。这一章首先介绍了系统最底层的一些操作代码，并给出了一个使用 C 语言但不使用 RTOS 时通用的启动代码。使用这个启动代码之后，这几个芯片最底层的一些操作读者可以不再关心，读者可以从 C 语言的 main 函数开始编写代码。然后本章分析各个功能部件的编程方法，并分别给出汇编的例子和 C 语言的例子。

第 6 章详细讲述了如何把嵌入式实时操作系统 μC/OS－II 移植到芯片 LPC2106/2105/2104 上。与一般公开的移植不同，本移植的任务不必在特权模式下运行(在用户和/或系统模式下运行)，任务可以任意使用 ARM 指令和/或 Thumb 指令。而且，它还支持 μC/OS－II 内核与用户任务分别编译，分别调入芯片执行的应用。

第 7 章介绍了各种情况下移植代码的使用方法，包括如何建立 ADS 工程。用户可以根据自己的需求参考相关部分编写自己的代码。

第 8 章介绍了移植代码相对原来的 μC/OS－II 增加的函数和配置参数。

第 9 章～第 12 章介绍了 LPC2106/2105/2104 几个重要功能部件的中间件，并分析了中间件的原理。它们包括数据队列、串口驱动驱动、I^2C 总线驱动和 SPI 总线驱动。它们应当可以直接在用户板中使用。

第 13 章则介绍了剩余功能部件在使用 μC/OS－II 的情况下如何编程。由于这些部件使用非常灵活，所以很难写出通用中间件。但读者可以通过这一章学会使用它们的方法。

参与本书写作、策划及 EasyARM2104 开发套件设计的工作主要人员有：陈明计、戚军、黄绍斌、钟亦峰、岳宪臣、朱旻和李仕兵等，全书由周立功负责策划、审定和统稿。

在此感谢美国 PHILIPS 半导体公司的潘志强先生、PHILIPS 亚太区经理华果先生、PHILIPS 香港区的梅润平、李建业、刘俊杰、郭志锐和陈华程等先生以及 PHILIPS 大陆区的刘忠、杨俊等先生多年来的大力支持和帮助。如果没有他们长期以来对我们的支持，我们肯定不能取得今天这样的成绩。

感谢北京航空航天大学出版社的大力支持，如果没有他们的帮助和努力，这本书不会这么快出版。

感谢“周立功单片机团队”所有成员几年来亲密无间的合作和默默无私的奉献，我之所以有今天，完全是与他们共同努力的结果，我仅仅是“周立功公司”的代表而已。

由于本书的所有作者都是初次使用 ARM，理解难免出现偏差。因此，本书在各个方面难免有疏忽、不恰当甚至完全错误的地方，恳请各位同行指正。

周立功

2003 年 9 月 3 日

目　录

第 1 章　从程序员角度看 ARM7TDMI(-S)

第 5 章 LPC2106/2105/2104 基础实验

第 6 章 移植 μC/OS - II