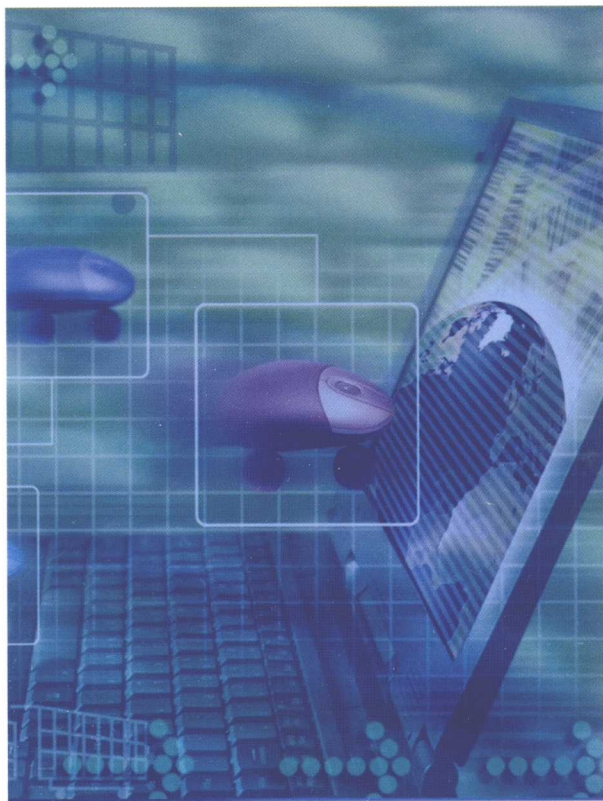


高等学校计算机应用规划教材

嵌入式系统开发基础

——基于ARM9微处理器C语言程序设计(第二版)

- ◆ ARM体系结构和发展现状
- ◆ ADS1.2开发环境的创建和使用
- ◆ 使用AXD进行代码调试
- ◆ S3C2410硬件资源的详细介绍
- ◆ 全部硬件驱动程序源代码



侯殿有 编著



清华大学出版社

013028331

TP332-43
42-2

高等学校计算机应用规划教材

嵌入式系统开发基础

——基于 ARM9 微处理器 C 语言
程序设计(第二版)

侯殿有 编著



清华大学出版社



北航

C1635088

TP332-43

42-2

13058331

内 容 简 介

本书对 32 位精简指令系统嵌入式微处理器 S3C2410 的硬件系统和 C 语言驱动程序进行了详细的讲解, 书中的源代码和实例程序对学习或从事嵌入式系统设计的读者都有很高的参考价值; 另外在人机界面设计、系统初始化程序编写、仿真器设置和复杂工程项目构建等方面给出了简化做法, 使初学者能够轻松、快速地掌握嵌入式系统设计方法。

本书以实用技术为主, 内容通俗易懂, 实例丰富, 特别适合初学者和从事嵌入式系统设计工作的读者使用。

本书配套的电子课件、实验讲义、习题答案和部分工具软件可以到 <http://www.tupwk.com.cn/downpage> 网站下载。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

嵌入式系统开发基础——基于 ARM9 微处理器 C 语言程序设计/侯殿有 编著. —2 版. —北京: 清华大学出版社, 2013.3

(高等学校计算机应用规划教材)

ISBN 978-7-302-31665-7

I. ①嵌… II. ①侯… III. ①微型计算机—系统开发—高等学校—教材 ②C 语言—程序设计—高等学校—教材 IV. ①TP360.21 ②TP312

中国版本图书馆 CIP 数据核字(2013)第 042925 号

责任编辑: 胡辰浩 袁建华

封面设计: 牛静敏

版式设计: 康 博

责任校对: 蔡 娟

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

课 件 下 载: <http://www.tup.com.cn>, 010-62796045

印 刷 者: 北京四季青印刷厂

装 订 者: 三河市兴旺装订有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 17 字 数: 392 千字

版 次: 2013 年 3 月第 2 版 印 次: 2013 年 3 月第 1 次印刷

印 数: 1~4000

定 价: 30.00 元

第二版前言

嵌入式控制系统的教学现状

嵌入式控制系统的教学一般分为两个层次,首先完成以 MCS-51 为代表的 8 位单片机的教学,这在各个高校都获得了重视,大多数学校安排理论课 64 学时,实验课 32 学时,课时比较充足。在这个层次上,无论是讲授 C 语言程序设计或汇编语言程序设计,可供选择的教材都比较多。

第二个层次,也就是以 32 位 ARM 为代表的嵌入式控制系统的教学,许多学校都没有开设,这主要有以下 3 个原因:

一是缺乏师资。毕竟以 ARM 为代表的嵌入式控制系统设计是 20 世纪 90 年代才发展起来的新技术,它不仅包括高性能、功能丰富的硬件平台,而且软件开发的难度和嵌入式操作系统的应用,都对教师提出了更高的要求。

二是在课时安排上也有一定困难。这么复杂的软硬件系统,包括嵌入式操作系统,即使用 96(包括实验)学时,也不一定能讲深讲透。况且,整个教学计划中也没有这么多的时间。

第三个原因是没有合适的教材。特别是深入浅出、条理分明、适应本科生水平、课时比较合理的教材非常少。

为了克服上述困难,也为了满足教学需要,作者根据多年科研和教学经验编写了本书。

作者的想法是:在 32 位 ARM 为代表的嵌入式控制系统的教学中,不讲带嵌入式操作系统的部分,而选择一种有代表性的 32 位单片机(类似 8 位机中的 MCS-51),这里选择韩国三星 S3C2410 ARM9 单片机,在 ADS1.2 For Windows 集成开发环境中,用 C 语言完成嵌入式控制系统的开发工作。理论课内容安排 48 学时,实验课时间和内容由教师根据各校的时间和条件自行决定。

在 48 学时(16 周,每周三学时)内,集中将 S3C2410 的最基本硬件结构,软件资源学深学透,学会用 C 语言编写应用程序。在用 C 语言编写驱动程序时,尽量借助系统资源,参考例子程序、减少设计者的工作量。通过较短时间的学习,学生可以很快掌握嵌入式控制系统设计的方法,完成嵌入式控制系统的设计工作。

本书篇幅虽然不长,但程序源代码较多,对于从事嵌入式系统开发和的学习来说是非常宝贵的资源,但是如果在课堂上讲解和分析这些代码,学时显然不够,建议教师主要讲解 S3C2410 的硬件资源和编程方法,具体程序代码留给学生课后慢慢消化理解。

教学实验平台介绍

有条件的学校,在完成理论课教学的同时,应安排一定的实验课,教学效果会更好。

作者接触的 ARM 9(SAMSUNG 2410)教学实验系统有:深圳英蓓特信息技术有限公司(<http://www.embedinfo.com>)的 Embest EDUKIT-II/III、北京博创科技集团(<http://www.up-tech.com>)的 UP-NETARM2410 教学实验系统、北京精仪达盛科技公司(<http://www.techshine.com>)

的 EL-ARM-830 教学实验系统, 都有基于 ARM 9 系统资源的 C 语言实验程序例子, 使用方便, 可供选择。随书下载的实验讲义有两册: 一是基于深圳英蓓特信息技术有限公司(<http://www.embedinfo.com>)的 Embest EDUKIT- II/III, 实验时应配合 Embest EDUKIT- II/III 教学实验系统平台, 并安装 Embest IDE; 二是基于北京精仪达盛科技公司(<http://www.techshine.com>)的 EL-ARM-830 教学实验系统, 实验时应配合 EL-ARM-830 教学实验系统平台。两套实验系统程序的执行都要去掉目录中的中文目录并尽量缩短目录深度。

本书主要内容和教学方法, 学习本书所需基础知识

第 1 章简单讲述嵌入式控制系统的定义、研究现状、研究方法。

第 2 章较详细地讲述基于 ARM 芯片的集成开发环境 ADS 1.2 的创建和使用。

第 3 章讲述 ARM 9 芯片 S3C2410 的片上资源和编程参考项目 2410test.mcp。

第 4 章讲述 S3C2410 的中断系统及编程。

第 5 章讲述 S3C2410 的 I/O 口和 I/O 口操作。

第 6 章讲述 S3C2410 的串口 URAT 及其编程。

第 7 章讲述 S3C2410 的 A/D 和 D/A 转换控制。

第 8 章讲述 ADC 和触摸屏控制。

第 9 章讲述 S3C2410 的实时时钟(RTC)和编程。

第 10 章讲述直接存储器存取(DMA)的工作原理及 S3C2410 的 DMA 控制器。

第 11 章讲述脉宽调制(PWM)的工作原理及 S3C2410 的 PWM 控制器。

第 12 章讲述看门狗(Watchdog)电路的工作原理及 S3C2410 的 Watchdog 控制。

第 13 章讲述双向二线制同步串行总线 I²C 及 S3C2410 的 I²C 控制电路。

第 14 章讲述数字音频信号(I²S)和 S3C2410 的 I²S 控制。

第 15 章对串行外设接口(SPI)进行了介绍。

第 16 章讲述 S3C2410 的人机界面设计。

以上各章内容除第 1~5 章外, 其他各章内容基本独立, 教师如果觉得在 48 学时内完成教学比较困难, 除第 2、3、4、5 章和第 16 章做为重点建议必讲之外, 其他各章可根据情况有选择地删节。

随书提供软件包一个, 其中有本书的电子课件、S3C2410 使用手册、实验讲义、各章习题答案、ADS1.2、参考项目 2410test.mcp、通用字模提取程序和部分例子程序, 可以在清华大学出版社网站(<http://www.tupwk.com.cn/downpage>)上免费下载。

第二版课件由孙颖馨老师在第一版基础上重新进行了制作, 作者对她的工作表示感谢。

本书的特点是通过深入浅出的讲述, 将基于 ARM 9 的嵌入式控制系统设计方法教给学生, 使学生感到嵌入式控制系统设计简单易学, 能够在最短的时间内入门。

学习本书时, 学生至少要有 C 语言基础, 如果有 MCS-51 单片机基础, 学习本书就会更加轻松。

第二版与第一版区别

为了满足教学急需, 第一版出书时间较紧, 书中难免有错误或不足之处, 在第二版中作者对书中内容进行了仔细斟酌研究, 更正了已发现的错误。并根据多年教学经验和指导学生参加全国和省级“嵌入式”和“电子设计”大赛体会, 删除了一些不适用的章节, 增加了一部分新内容。

书中实验程序的注释是本书的重要内容，仔细阅读这些注释对于理解书中内容和练习编程非常重要。

为了使读者正确理解原程序，凡是原参考文献给出的注释，书中仍然保留英文，凡是作者给出的注释，用中文给出。

根据读者意见，为了使版面工整，方便阅读，注释采用分散方式对齐。

虽然做了很大努力，并请孙俊喜、才华两位教授对书中内容进行审核校对，但百密一疏，难免有考虑不周或错误之处，真诚欢迎读者多提宝贵意见和建议。我们的信箱是 huchenhao@263.net，电话是 010-62796045。

本书通用字模提取程序密码：194512125019。

侯殿有

2012年12月

目 录

第 1 章 嵌入式控制系统简介	1
1.1 单片机和嵌入式控制系统的定义和分类	1
1.1.1 单片机和嵌入式控制系统的定义	1
1.1.2 嵌入式控制系统的设计方法	2
1.1.3 嵌入式控制系统各种设计方法的特点	2
1.2 ARM 处理器简介	4
1.2.1 ARM 体系结构的发展	4
1.2.2 ARM 体系结构的存储器格式	8
1.3 习题	9
第 2 章 ADS1.2 开发环境创建与简介	10
2.1 ADS1.2 开发环境创建	10
2.1.1 ADS1.2 概述	10
2.1.2 ADS1.2 的安装	11
2.2 ADS 集成开发环境的使用	14
2.2.1 建立一个新工程	14
2.2.2 开发环境设置	14
2.2.3 其他开发环境介绍	19
2.3 用 AXD 进行代码仿真、调试	19
2.3.1 AXD 简介	19
2.3.2 JTAG 概述	22
2.3.3 Nor 和 Nand Flash 的区别和使用	23
2.3.4 烧写 Flash	24
2.3.5 程序的运行	24
2.4 ARM C 语言程序的基本规则和系统初始化程序	25

2.4.1 ARM 使用 C 语言编程基本规则	26
2.4.2 初始化程序和开发环境设置	27
2.5 习题	28
第 3 章 ARM9 微处理器 S3C2410 资源	29
3.1 S3C2410 处理器介绍	29
3.1.1 AMBA、AHB、APB 总线特点	30
3.1.2 S3C2410 处理器体系结构	30
3.1.3 S3C2410 处理器管理系统	31
3.1.4 S3C2410 处理器存储器映射	31
3.1.5 S3C2410 处理器时钟和电源管理	31
3.2 S3C2410 处理器片上资源的定义和使用	33
3.3 参考软件资源 2410test.mcp	34
3.4 几个常用的输入/输出函数	39
3.5 DEF.H 头文件	43
3.6 习题	43
第 4 章 S3C2410 的中断系统	45
4.1 S3C2410 的中断源	45
4.2 S3C2410 的中断处理	46
4.3 中断控制	47
4.3.1 中断模式(INTMOD)寄存器	47
4.3.2 中断挂起寄存器和中断源挂起寄存器	48
4.3.3 中断屏蔽寄存器(INTMSK)	50
4.3.4 中断优先级寄存器(PRIORITY)	50

4.4	子中断源的中断控制	53	6.1.3	UART 通信操作	76
4.5	中断向量设置	54	6.2	UART 的控制寄存器	76
4.6	其他常用寄存器	54	6.2.1	UART 线路控制寄存器	
4.7	中断程序编写中需注意的问题	56		ULCON _n (n=0~2)	76
4.8	中断实验和中断程序编写	58	6.2.2	UART 控制寄存器	
4.9	习题	61		UCON _n (n=0~2)	76
第 5 章	S3C2410 的 I/O 口和 I/O 操作	62	6.2.3	UART FIFO 控制寄存器	
5.1	S3C2410 I/O 口描述	62		UFCON _n (n=0~2)	77
5.2	I/O 端口控制寄存器	63	6.2.4	UART 调制解调器控制寄存器 UMCON _n (n=0 或 1)	77
5.2.1	端口 A 控制寄存器和功能配置	63	6.2.5	发送寄存器 UTXH _n (n=0~2)和接收寄存器 URXH _n (n=0~2)	78
5.2.2	端口 B 控制寄存器和功能配置	64	6.2.6	UART TX/RX 状态寄存器 UTRSTAT _n (n=0~2)	78
5.2.3	端口 C 控制寄存器和功能配置	65	6.3	UART 通信程序例子	78
5.2.4	端口 D 控制寄存器和功能配置	66	6.3.1	RS232 接口电路	78
5.2.5	端口 E 控制寄存器和功能配置	68	6.3.2	UART 实验程序	79
5.2.6	端口 F 控制寄存器和功能配置	69	6.4	习题	85
5.2.7	端口 G 控制寄存器和功能配置	70	第 7 章	S3C2410 的 A/D、D/A 转换控制	86
5.2.8	端口 H 控制寄存器和功能配置	71	7.1	S3C2410 的 A/D、D/A 转换控制	86
5.3	I/O 口控制 C 语言编程实例	72	7.1.1	A/D 转换控制寄存器 (ADCCON)	86
5.3.1	硬件电路	72	7.1.2	A/D 转换控制程序的编制步骤	87
5.3.2	参考程序	73	7.2	参考程序	87
5.4	习题	74	7.3	习题	90
第 6 章	S3C2410 的串口 UART 及编程	75	第 8 章	触摸屏控制	91
6.1	S3C2410 的串口 UART 概述	75	8.1	触摸屏结构和工作原理	91
6.1.1	S3C2410 串行通信 (UART)单元	75	8.1.1	触摸屏工作原理	91
6.1.2	波特率的产生	75	8.1.2	S3C2410 的触摸屏控制	93
			8.2	触摸屏控制程序	96
			8.3	习题	98

第 9 章 S3C2410 的实时时钟(RTC)·····99	11.3.5 减法计数器观察寄存器 TCNTOn·····127
9.1 实时时钟在嵌入式系统中 的作用·····99	11.4 PWM 参考程序·····128
9.1.1 S3C2410 的实时时钟单元·····99	11.5 习题·····134
9.1.2 S3C2410 的实时 时钟寄存器·····100	第 12 章 S3C2410 的看门狗 电路控制·····136
9.2 参考程序及说明·····102	12.1 看门狗电路的功能 及工作原理·····136
9.3 习题·····108	12.1.1 S3C2410 的看门狗 控制·····136
第 10 章 直接存储器存取(DMA)控制···109	12.1.2 看门狗定时器寄存器·····137
10.1 DMA 基础知识·····109	12.2 参考程序及说明·····138
10.2 S3C2410 的 DMA 控制器·····111	12.3 习题·····139
10.3 DMA 方式实现存储器 到存储器的数据传送·····113	第 13 章 S3C2410 的 I²C 总线控制···140
10.3.1 头文件定义和函数 声明·····113	13.1 I ² C 接口和 EEPROM·····140
10.3.2 DMA 方式实现存储器 到存储器的数据传送·····114	13.2 EEPROM 读/写操作·····143
10.4 习题·····119	13.2.1 AT24C04 结构 与应用简述·····143
第 11 章 S3C2410 的 PWM 控制·····120	13.2.2 设备地址(DADDR)·····144
11.1 PWM 定时器概述·····120	13.2.3 AT24CXX 的数据 操作格式·····144
11.1.1 什么是脉宽调制(Pulse- Width Modulation)·····120	13.3 S3C2410 处理器 I ² C 接口·····144
11.1.2 S3C2410 的脉宽调制 和 PWM 控制·····120	13.3.1 S3C2410 I ² C 接口简介·····144
11.1.3 S3C2410 定时器特性·····122	13.3.2 使用 S3C2410 I ² C 总线读/写方法·····146
11.1.4 定时器操作示例·····123	13.4 S3C2410 I ² C 总线读/写 参考程序编写·····146
11.1.5 死区生成器·····123	13.5 I ² C 实验程序·····147
11.2 PWM 输出电平控制·····124	13.6 习题·····151
11.2.1 PWM 工作原理·····124	第 14 章 I²S 介绍和 S3C2410 的 I²S 控制·····152
11.2.2 PWM 输出控制·····125	14.1 数字音频信号(I ² S)介绍·····152
11.3 PWM 定时器控制寄存器·····125	14.2 数字音频计算机处理·····153
11.3.1 定时器配置寄存器 0·····125	14.2.1 采样频率和采样精度·····153
11.3.2 定时器配置寄存器 1·····126	14.2.2 音频编码·····154
11.3.3 减法缓冲寄存器 和比较缓冲寄存器·····126	
11.3.4 定时器控制寄存器·····127	

14.2.3	IIS 数字音频接口	154	15.2.7	SPI 接口操作	178
14.3	音频芯片 UDA1341TS		15.2.8	SPI 接口编程	178
	介绍	154	15.2.9	SPI 口的传输格式	178
14.3.1	硬件结构	154	15.2.10	SPI 通信模式	180
14.3.2	S3C2410 和 UDA1341TS		15.3	参考程序	180
	的连接	156	15.4	习题	184
14.3.3	UDA1341TS 的软件		第 16 章	S3C2410 的人机界面设计	185
	编程	156	16.1	汉字和西文字符存储	
14.3.4	UDA1341TS DATA0			与显示原理	185
	编程	158	16.1.1	ASCII 码	185
14.3.5	UDA1341TS DATA1		16.1.2	英文字符的显示	186
	编程	160	16.2	汉字在计算机中的表示	
14.3.6	UDA1341TS 控制寄存器			和显示	187
	STATUS 编程	161	16.2.1	汉字的内码和区位码	187
14.4	S3C2410 中 I ² S 总线控制		16.2.2	汉字的显示	188
	寄存器	162	16.2.3	其他西文字符在计算机中	
14.5	WAV 声音格式文件	164		的存储和显示	189
14.6	IIS 实验参考程序	164	16.2.4	屏幕上“打点”	190
14.7	习题	173	16.2.5	字模提取与建立小字库	
第 15 章	串行外设接口(SPI)介绍	174		概述	190
15.1	SPI 接口及操作	174	16.3	字模提取与建立小字库	190
15.1.1	SPI 接口原理	174	16.3.1	用 C 语言提取字模	
15.1.2	SPI 接口特性	176		和建立小字库	191
15.2	SPI 接口控制寄存器	176	16.3.2	用 Delphi 提取字模	
15.2.1	SPI 控制寄存器			和建立小字库	195
	(SPICONn)	176	16.3.3	通用字模提取程序	
15.2.2	SPI 状态寄存器			MinFonBase 使用说明	205
	(SPSTAn)	176	16.4	S3C2410 显示控制特点	206
15.2.3	SPI 引脚控制寄存器		16.4.1	STN LCD 显示器	206
	(SPPINn)	177	16.4.2	TFT LCD 显示器	206
15.2.4	SPI 波特率预分频寄存器		16.4.3	LCD 控制器特点	206
	(SPIPREn)	177	16.5	S3C2410 的 LCD 控制信号	
15.2.5	SPI 发送数据寄存器			和外部引脚	207
	(SPTDATn)	177	16.5.1	LCD 专用控制寄存器	208
15.2.6	SPI 接收数据寄存器		16.5.2	LCD 专用控制寄存器	
	(SPRDATn)	178		的设置	213

16.5.3 LCD 屏幕“打点” 程序.....	216	16.7.3 HD66421 与微处理器 接口及驱动程序.....	235
16.6 S3C2410 的 LCD 驱动程序.....	220	16.8 在 LCD 屏上按一定 格式显示汉字和曲线.....	247
16.6.1 S3C2410LCD 驱动程序 编写步骤.....	220	16.9 S3C6410 (ARM11)的汉字 和曲线显示.....	248
16.6.2 利用 S3C2410 显示汉字 与曲线.....	221	16.9.1 S3C6410 (ARM11) 简介.....	248
16.7 S3C2410 在 LCD 驱动 方面的其他应用.....	230	16.9.2 S3C6410(ARM11) 的汉字和曲线显示.....	250
16.7.1 HD66421 的硬件简介.....	230	16.10 习题.....	257
16.7.2 HD66421 的软件编程.....	232	参考文献.....	258

第1章 嵌入式控制系统简介

嵌入式控制系统设计是当前 IT 行业最热门的话题之一。什么是嵌入式控制系统？它们如何分类？各类系统的设计方法有什么不同？本章都将给出答案，并对嵌入式控制系统中最常用的 ARM 嵌入式微处理器做简要介绍。

1.1 单片机和嵌入式控制系统的定义和分类

在现有的不同文献中，嵌入式控制有着不同的定义，最常见的一种定义是：嵌入式系统是以应用为中心、以计算机技术为基础、软硬件可裁剪的，对功能、可靠性、成本、体积和功耗有严格要求的专用计算机系统。还有一种定义是：嵌入式系统就是一个具有特定功能或用途的计算机软硬件结合体。这些说法虽然在一定程度上对嵌入式进行了描述，但都不够全面或确切。

实际上，嵌入式控制系统是和单片机的产生和发展分不开的。本节将结合单片机对嵌入式系统进行定义，并对嵌入式控制系统的设计方法进行介绍。

1.1.1 单片机和嵌入式控制系统的定义

单片机就是在一片半导体硅片上集成了中央处理器单元(CPU)、存储器(RAM/ROM)和各种 I/O 接口的微型计算机。这样的一块集成电路芯片具有一台微型计算机的功能，因此被称为单片微型计算机，简称单片机。

单片机主要应用在测试和控制领域。由于单片机在使用时通常处于测试和控制领域的核心地位并嵌入其中，因此人们常把单片机称为嵌入式微控制器(Embedded Microcontroller Unit)，把嵌入某种微处理器或单片机的测试和控制系统称为嵌入式控制系统(Embedded Control System)。

嵌入式控制系统在航空航天、机械电子、家用电器等各个领域都有广泛的应用，其中家用电器领域是嵌入式控制系统最大的应用领域。MP3、MP4、数码相机、扫描仪、个人 PC、车载电视、DVD、PDA 等，到处都可以看到嵌入式控制系统的应用。

随着超大规模集成电路工艺和集成制造技术的不断完善，单片机的硬件集成度也在不断提高，已经出现了能满足各种不同需要、具有各种特殊功能的单片机。在 8 位单片机得到广泛应用的基础上，16 位单片机和 32 位单片机也应运而生，特别是以 ARM 技术为基础的 32 位精简指令集系统单片机(RISC Microprocessor)的出现，由于其性能优良、价格低廉，因此大有取代 16 位单片机而成为高档主流机型的趋势。

嵌入式控制系统由于其内核嵌入的微处理器不同，在应用上大致可分为两个层次，在系统简单、要求不高以及成本低的应用领域，大多采用以 MCS-51 为代表的 8 位单片机。

随着嵌入式控制系统与 Internet 的逐步结合, PDA、手机、路由器、调制解调器等复杂的高端应用, 对嵌入式控制器提出了更高的要求, 在少数高端应用领域以 ARM 技术为基础的 32 位精简指令系统单片机得到越来越多的青睐。嵌入式控制系统在高端应用领域又分为带嵌入式操作系统支持和不带嵌入式操作系统支持两种情况。

1.1.2 嵌入式控制系统的设计方法

作为嵌入式控制器的单片机, 不管是 8 位、16 位还是 32 位, 由于受其本身资源限制, 其应用程序都不能在其自身上开发。开发其应用程序, 都需要一台通用计算机, 如常用的 IBM-PC 机或兼容机, Windows 95/98/2000 或 XP 操作系统, 256MB 以上内存, 1GB 以上硬盘存储空间(运行交叉编译环境 ADS1.2 最低配置)。这台通用计算机称为“宿主机”, 嵌入式控制器的单片机称为“目标机”。应用程序在“宿主机”上开发, 在“目标机”上运行。“目标机”和“宿主机”之间利用计算机并口通过一台叫“仿真器”的设备相连, 程序可以从“宿主机”传到“目标机”, 这也叫程序下载, 也可以从“目标机”传到“宿主机”, 这叫程序上传。应用程序通过“仿真器”的下载和上传, 在“宿主机”上反复修改, 这个过程叫做“调试”。调试好的应用程序, 在“宿主机”上编译成“目标机”可以直接执行的机器码文件, 通过一台叫“固化器”的设备下载并固化到“目标机”的程序存储器中, 整个下载过程, 称为烧片, 也称为程序固化。

程序固化是单片机开发的最后一步, 以后“宿主机”和“目标机”就可以分离, “宿主机”任务完成, “目标机”就可以独立执行嵌入式控制器的任务。

1.1.3 嵌入式控制系统各种设计方法的特点

1. 目标机上安装某种嵌入式操作系统

随着嵌入式系统的发展, 应用程序变得越来越复杂, 例如应用程序与 Internet 的结合、多线程、复杂的数据处理、高分辨率图形图像显示等, 如果没有操作系统支持, 应用程序的编写将变得非常困难。因此, 人们在目标机上嵌入某种功能较强且占用内存较少的操作系统, 用户程序在该操作系统支持下运行, 这种操作系统叫做嵌入式操作系统。嵌入式操作系统有多种, 如比较著名的 Windows CE、Linux、 μ C/OS-II 等。特别是 Linux 操作系统, 由于其具有代码简练、功能强大、内核公开等优点, 获得了广泛应用。

采用 Linux 操作系统来开发嵌入式系统, 首先要在“宿主机”上建立 Linux 开发环境, 这有两种作法: 一是“宿主机”放弃原来的 Windows 操作系统, 改装 Linux 操作系统, 如安装 Linux Red Hat 9.0; 二是在原来的 Windows 操作系统上安装一个虚拟机, 在该虚拟机中安装 Linux 操作系统, 如 Cygwin 1.5.10(可以从 <http://www.cygwin.com> 下载并安装最新版本)。

接着要根据应用程序的需要编写一个驱动程序, 把该驱动程序和 Linux 操作系统一起编译, 形成一个包含此驱动程序的 Linux 内核可执行文件 image, 将此文件下载到“目标机”。今后, 实现应用程序的功能, 只需对内核中相应的函数进行调用即可。

由于“宿主机”和“目标机”之间文件的下载和上传是以文件形式进行的, 所以在两个机器上都要有相应的文件管理系统, 在“宿主机”上, 可以使用 TFTP Server for Windows,

在“目标机”上则还要下载 Cramfs 文件管理系统。

为了实现上电时系统能自动按一定顺序启动,如系统的硬件初始化,包括时钟的设置、存储区的映射、设置堆栈指针、应用程序入口等,还必须有一个系统引导程序,即 Boot Loader,常用的 Boot Loader 是由韩国 Mizi 公司开发的 VIVI 软件,该软件特别适合 ARM 9 处理器,需要将 VIVI 下载到“目标机”上。

此外,还要为每一个项目的驱动程序和应用调试程序各编写一个工程管理文件 Makefile。

在 Linux 操作系统下,对应用程序和驱动程序的编辑和调试还需要一个交叉编译工具,要在 busybox 工具集中选择需要的部分进行编辑,形成可执行文件,下载到“目标机”上。

2. 目标机上不安装操作系统

在这种情况下,把 ARM9 只当成是 32 位单片机,使用 Code Warror IDE 对其进行开发,整个开发过程和开发 MCS-51 单片机一样,非常简单。

ADS(ARM Developer Suite)是 ARM 公司推出的新一代 ARM 开发工具,目前的最新版本是 ADS1.2。ADS 使用 Code Warror IDE 替代了旧的开发工具,使用 AXD 作为调试工具,现代集成开发环境的一些特点,如源文件编辑器、语法高亮和窗口驻留等功能都有体现。

ADS 使用并口通过 JTAG 仿真器与“目标机”相连,实现在线调试与仿真。

3. 两种设计方法的特点

带操作系统的嵌入式控制系统,在编制较复杂和高端应用程序时,例如上面提到的与 Internet 的结合、多线程、复杂的数据处理、高分辨率图形图像显示等,用户程序就会比较简单,但整个工程研制的时间开销不会少,因为要把很多时间放在对 Linux 操作系统的安装和熟悉上,虽然 Linux 操作系统是免费的,其内核可以根据用户需要进行剪裁,但要达到随意剪裁的水平,需要花费很多时间去熟悉和研究。此外,程序员还要学会驱动程序和 Makefile 文件的编写,特别是驱动程序,每一个设备都要有一个,它要和内核结合到一起,形成操作系统的一部分。也就是说,在开发嵌入式控制系统时,还要完成一部分操作系统的内核工作,难度较大,会花费很多时间。

系统在调试程序时,要占用“宿主机”较多资源,如使用并口连接 JTAG 仿真器、使用串口与“宿主机”通信、使用网口来传输文件。

如果在目标机上不安装嵌入式操作系统,把 ARM9 只当成是 32 位单片机来开发,那么,整个开发过程和开发 MCS-51 单片机一样,特别简单。这样就可以把主要时间放在对 ARM9 单片机软件和硬件的熟悉上,充分发挥 32 位单片机本身资源优势;把主要精力放在控制系统的稳定性和可靠性上,在较短时间内开发出高品质的嵌入式产品。

嵌入式控制系统大多具有小、巧、轻、灵、薄的特点,需要与 Internet 结合、多线程的系统等“高端应用”只占非常少的一部分,因此不采用嵌入式操作系统,也可以满足系统需要。

如果系统需要网络连接(连接 Internet 会使系统易受病毒攻击,导致系统稳定性下降,同时运行数据易泄密,因此工程上基本只使用局域网),可以采用串行通信代替,点对点且距离不长,可以采用 232 标准,多点通信或距离较长,可以采用 485 标准。如遇多线程问题,可以采用多微处理器分级分布控制。

1.2 ARM 处理器简介

ARM 有 3 个含义：一是从事嵌入式微处理器开发的高科技公司的名字；二是代表一种低功耗、高性能的 32 位 RISC(精简指令系统)处理器的技术；三是代表一种微处理器产品。

本节介绍 ARM 微处理器系列的几种产品，从中可以看到 ARM 技术的发展和现状。

1.2.1 ARM 体系结构的发展

ARM 处理器是一种低功耗、高性能的 32 位 RISC(精简指令系统)处理器。本章将从其结构入手，分析目前流行的 ARM920T 硬件结构和编程。

ARM 处理器共有 31 个 32 位寄存器，其中 16 个可以在任何模式下看到。它的指令为简单的加载与存储指令(从内存加载某个值，执行完操作后再将其放回内存)。ARM 的特点有：所有的指令都带有条件；可以在加载数值的同时进行算术和移位操作。它可以在几种模式下操作，包括使用 SWI(软件中断)指令从用户模式进入系统模式。

ARM 处理器是一个综合体。ARM 公司自身并不制造微处理器，它们是由 ARM 的合作伙伴(Intel 或 LSD)制造的。ARM 还允许将其他处理器通过协处理器接口进行紧耦合。它还包括几种内存管理单元的变种，包括简单的内存保护到复杂的页面层次。

ARM 微处理器系列包括 ARM7、ARM9、ARM9E、ARM10E、SecurCore 等系列和 Intel 的 Xscale。其中，ARM7、ARM9、ARM9E 和 ARM10E 为 4 个通用处理器系列，每一个系列都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。

1. ARM7 系列微处理器

ARM7 系列微处理器是低功耗的 32 位 RISC 处理器，适用于对价位和功耗要求较高的消费类产品。ARM7 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产品，如便携式产品。
- 能够提供 0.9 MIPS(MIPS，每秒百万条指令)/MHz 的三级流水线结构。
- 对操作系统的支持广泛，如 Windows CE、Linux、PalmOS(最流行的掌上电脑操作系统)等。
- 指令系统与 ARM9、ARM9E、ARM10E 系列兼容，便于用户的产品升级换代。
- 主频最高可达 130MHz，高速的运算处理能力可胜任绝大多数的复杂应用。

ARM7 系列微处理器主要应用于工业控制、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，ARM7TDMI 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 的基本含义如下：

- 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

2. ARM9 系列微处理器

ARM9 系列微处理器在高性能和低功耗方面有着非常突出的特点。具体如下:

- 5 级流水线结构, 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 WindowsCE、Linux、PalmOS 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(高速缓存)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 系列微处理器包括 ARM920T、ARM922T 和 ARM940T 共 3 种类型, 以适用于不同的应用场合。

3. ARM9E 系列微处理器

ARM9E 系列微处理器的主要特点如下:

- 支持 DSP 指令集, 适用于需要高速数字信号处理的场合。
- 5 级流水线, 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 VFP9 浮点处理协处理器。
- 全性能的 MMU, 支持众多主流嵌入式操作系统。
- 支持数据 Cache 和指令 Cache, 具有更高的处理能力。
- 主频最高可达 300MHz。

ARM9E 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、存储设备和网络设备等领域。

ARM9E 系列微处理器包含 ARM926EJ-E、ARM946E-S 和 ARM966E-S 共 3 种类型, 以适用于不同的应用场合。

4. ARM10E 系列微处理器

ARM10E 系列微处理器的主要特点如下:

- 支持 DSP 指令集, 适用于需要高速数字信号处理的场合。
- 6 级流水线, 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 浮点协处理器。
- 全性能的 MMU，支持众多主流嵌入式操作系统。
- 支持数据 Cache 和指令 Cache，具有更高的处理能力。
- 主频最高可达 400MHz。
- 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 系列微处理器包括 ARM1020E、ARM1002E 和 ARM1026JE-S 共 3 种类型，以适用于不同的应用场合。

5. ARM920T

ARM920T 高缓存处理器是 ARM9 Thumb 系列中高性能的 32 位单片系统处理器。

ARM920TDMI 系列微处理器包含如下几种类型的内核。

- ARM9TDMI：只有内核。
- ARM940T：由内核、高速缓存和内存保护单元(MPU)组成。
- ARM920T：由内核、高速缓存和内存管理单元(MMU)组成。

ARM920T 提供完善的高性能 CPU 子系统，包括以下几个方面：

- ARM9TDMI RISC CPU。
- 16K 字节指令缓存与 16K 字节数据缓存。
- 指令与数据存储管理单元(MMU)。
- 写缓冲器。
- 高级微处理器总线架构(AMBA)总线接口。
- ETM(内置跟踪宏单元)接口。

ARM920T 中的 ARM9TDMI 内核可执行 32 位 ARM 及 16 位 Thumb 指令集。ARM9TDMI 处理器是哈佛结构，有包括取指、译码、执行、存储及写入的 5 级流水线。

ARM920T 处理器包括 CP14 和 CP15 两个协处理器。

- CP14：控制软件对调试通道的访问。
- CP15：系统控制处理器，提供 16 个额外寄存器来配置与控制缓存、MMU、系统保护、时钟模式以及其他系列选项。

ARM920T 处理器的主要特征如下：

- ARM9TDMI 内核，ARM v4T 架构。
- 两套指令集：ARM 高性能 32 位指令集和 Thumb 高代码密度 16 位指令集。
- 5 级流水线结构，即取指(F)、指令译码(D)、执行(E)、数据存储访问(M)和写寄存器(W)。
- 16K 字节数据缓存，16K 字节指令缓存。
- 写缓冲器：16 字的数据缓冲器。
- 标准的 ARMv4 存储器管理单元(MMU)：区域访问许可，允许以 1/4 页面大小对页面进行访问，16 个嵌入域，64 个输入指令 TLB 以及 64 个输入数据 TLB。