

普通高等教育“十一五”国家级规划教材



嵌入式系统技术

—基于ARM的嵌入式系统

王晓春 主编

TP323.2



高等教育出版社
HIGHER EDUCATION PRESS

普通高等教育“十一五”国家级规划教材

嵌入式系统技术

——基于 ARM 的嵌入式系统

Qianrushi Xitong Jishu

——Jiyu ARM de Qianrushi Xitong



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

内容提要

本书是普通高等教育“十一五”国家级规划教材。

本书针对目前高职高专嵌入式系统及相关专业中有关嵌入式系统技术编写。全书共 11 章,分为三个部分。第一部分介绍嵌入式系统的基础知识及 ARM 微处理器的相关内容,对嵌入式系统中的常见硬件做了剖析,是开发嵌入式产品的硬件基础。第二部分介绍嵌入式系统软件,通过大量实例讲解了 ARM 的指令系统及嵌入式系统软件的设计方法。第三部分介绍嵌入式系统的调试,根据嵌入式系统应用开发过程中的具体问题而编写,内容实用,实例针对性强。

本书通俗易懂,其中很多内容是作者在多年教学实践中积累下来的,可作为技能型、应用型教育相关专业“嵌入式系统技术”课程的教材,也可作为专业人员的参考书。

图书在版编目(CIP)数据

嵌入式系统技术:基于 ARM 的嵌入式系统/王晓春主编. —北京:高等教育出版社,2010.7

ISBN 978-7-04-030169-4

I. ①嵌… II. ①王… III. ①微处理器, ARM-系统设计-高等学校:技术学校-教材 IV. ①TP332

中国版本图书馆 CIP 数据核字(2010)第 110237 号

策划编辑 冯 英 责任编辑 郭福生 封面设计 于 涛 责任绘图 尹 莉
版式设计 范晓红 责任校对 杨凤玲 责任印制 尤 静

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100120

经 销 蓝色畅想图书发行有限公司
印 刷 北京宏信印刷厂

开 本 787×1092 1/16
印 张 16.25
字 数 390 000

购书热线 010-58581118
咨询电话 400-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
网上订购 <http://www.landaco.com>
<http://www.landaco.com.cn>
畅想教育 <http://www.widedu.com>

版 次 2010 年 7 月第 1 版
印 次 2010 年 7 月第 1 次印刷
定 价 25.80 元(含光盘)

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究
物料号 30169-00

前 言

1. 关于嵌入式系统

嵌入式系统是嵌入式计算机系统的简称，包括软件系统和硬件系统。关于嵌入式系统的定义有多种说法，但其最主要特征可以归纳为：嵌入式系统是一个“看不见的计算机”系统。嵌入式系统的主要应用领域有网络通信、消费类电子及工业控制和测试。尤其是消费类电子产品，呈现出强劲的发展态势和活力，各种各样的新产品层出不穷。

嵌入式系统的开发需要研发人员熟悉软件开发工具、硬件平台和相关调试工具。嵌入式应用领域的逐渐拓展，使得社会对嵌入式开发人员的需求日渐增多。

2. 本书的编写特点

1) 以一个典型的嵌入式产品为主线

为突出高等职业教育特点，实现真正意义上的工学结合，经过广泛的调研和筛选，本教材确定以 PDA 这一典型的嵌入式产品为主线，将嵌入式系统中软硬件相关知识进行分解，并通过相关实训项目完成对知识点的讲解和训练。需要强调的是，本书并不是一本关于 PDA 的专业书籍，PDA 在书中只是一个载体，通过它来贯穿嵌入式软硬件的内容，书中内容要比 PDA 本身更广泛，也更深入。

2) 反映嵌入式系统的最新发展

近年来，随着嵌入式系统的发展，嵌入式系统的应用已进入 32 位时代。尤其是以 ARM 为代表的 RISC 产品，占据了嵌入式应用领域很大一部分市场。各种以 ARM 技术为核心的嵌入式产品广泛进入人们的生活。本书力求反映 ARM 技术的最新发展，同时还将嵌入式系统中的最新技术引入到书中来。

3) 以实用为目的，适于高校教学需求

目前基于 ARM 的嵌入式系统应用类书籍很多，但并不适合作为高等职业教育教材使用。其原因在于很多书籍内容技术性太强，其主要读者对象是开发工程师，内容偏重于芯片，手册的功能较强。本书的编写原则是依据嵌入式系统开发过程中的内容，将相关重点突出，知识点重新整理规划后形成的，是一本内容清晰、结构合理、使用方便的教材。

书中内容的编撰更适应目前高等教育中教学方法的改变，即突出学生学习能力的培养。学生在教师的重点指导下，可以独立完成课程的学习，更好地适应以学生为主体的教学方式。书中对重点难点进行标注，案例也更接近于工程应用的实际情况。各章练习题不仅仅有助于理解和巩固相关知识点，还有查阅资料、调研市场、撰写报告等类型的作业，促进学习者更深入地了解嵌入式系统的技术发展。

4) 不依赖于某一家公司的硬件平台

在学习过程中，经常要用到嵌入式开发硬件平台，而目前很多书籍的内容都会依赖于

一家公司的硬件平台，相关的实验也都是在相应的硬件平台上完成的。书籍与硬件开发平台太过紧密的捆绑，降低了书籍的适用程度。本书侧重于嵌入式硬件开发中的一般规律和方法，不依赖某一家公司的硬件平台。在不涉及实时问题时，相关实例都是在模拟环境（ARMulator）下实现的，从而突出了学习过程的重点内容；在涉及异常等实时性问题时，例子更侧重相关问题的处理过程，突出其中共性的问题。这样的话，不论在什么样的硬件平台下，相关的实例都可以经过少量移植后很方便地使用，使教材的适用面更广。

3. 嵌入式系统的学习方法

嵌入式应用程序的调试有着其特殊的方法。在嵌入式系统中开发程序相对于在桌面系统下开发要困难很多，虽然代码生成的过程并无很大差异，但程序的调试却有很大差异。因为嵌入式系统的编程是在宿主机上完成的，但代码的运行却是在目标平台上。所以程序编译链接完成后必须从宿主机上下载到目标平台上运行，而代码的调试必须借助各式各样的工具与仪器来完成。

嵌入式系统的应用环境与硬件密切相关，代码的运行受制于现场硬件的真实状态。所以在调试代码的过程中会经常出现各种各样的情况，其原因也五花八门：可能是代码本身的原因，也可能是硬件的错误连接；可能是时序的错误，也可能仅仅是旁边一台电风扇的启动造成的。面对这种情况，开发人员经常会感到一筹莫展。要找出问题，需要的不仅仅是编程知识。所以在嵌入式系统学习过程中，掌握程序的调试方法是非常重要的，也需要积累经验才能提升调试的水平。

4. 本书的内容组织

本书的编写直面嵌入式系统的核心问题。第1~6章介绍嵌入式系统的硬件部分，介绍了常见的嵌入式硬件，并对嵌入式系统中的电源管理做了介绍。第7章针对嵌入式系统应用中的实际需求，介绍了嵌入式系统软件的设计方法及功能划分原则，尤其是在应用系统中如何实现中断及中断处理程序的安装等内容。第8章介绍ARM的编译器和链接器，对嵌入式软件的编译链接过程做了剖析。第9章对嵌入式系统的启动代码做了分析。嵌入式应用的特殊性导致很难像PC一样有一个通用的启动代码，本章介绍了启动代码应该关注的问题和实现方法。第10章对编译器和链接器的参数做了详细分析。第11章中介绍了嵌入式系统的调试方法。全书内容围绕嵌入式应用中的常见问题，具有很强的实用性。

本书可作为高职高专院校嵌入式系统、电子工程、工业自动化、仪器仪表等相关专业的专业课程教材。课程安排学时为54~72学时，在实际使用时可根据具体要求调整。

本书由王晓春任主编，并负责编写第1~8章；聂丽文编写了第9~11章。

在本书的编写过程中，得到深圳职业技术学院孙涌老师、孙宏伟老师、潘焕成老师的大力支持和帮助，深圳职业技术学院的刘兴东高级工程师和深圳市英蓓特信息技术有限公司的徐光峰先生审阅了文稿，并提出了宝贵的修改意见，在此深表感谢！

由于编者水平有限，书中不妥或错误之处在所难免，希望读者批评指正，我们对此深表感谢。作者的E-mail地址：szwince@gmail.com。

编 者

2010年5月

目 录

第一部分 嵌入式系统硬件

第 1 章 一个典型的嵌入式系统	2	练习题	25
1.1 嵌入式系统	2	第 2 章 ARM 微处理器	26
1.1.1 嵌入式系统的定义	3	2.1 PDA 中的 ARM 微处理器	26
1.1.2 嵌入式微处理器	3	2.2 ARM 微处理器的工作模式	27
1.1.3 嵌入式系统的发展历程	5	2.2.1 ARM 微处理器的工作模式	27
1.1.4 嵌入式系统的特点	5	2.2.2 微处理器工作模式的切换	28
1.1.5 嵌入式系统开发的特殊要求	6	2.3 ARM 微处理器的工作状态	29
1.1.6 嵌入式系统对人才的能力要求	7	2.3.1 ARM 微处理器的工作状态	29
1.1.7 嵌入式系统行业对人才的需求	7	2.3.2 ARM 微处理器工作状态下的切换	30
1.2 ARM	8	2.4 ARM 微处理器的寄存器	30
1.2.1 ARM 公司简介	8	2.4.1 ARM 状态下的寄存器	30
1.2.2 ARM 体系结构的发展	9	2.4.2 Thumb 状态下的寄存器	35
1.2.3 ARM 系列产品	10	2.5 ARM 微处理器的异常	35
1.2.4 基于 ARM 技术的典型芯片	12	2.5.1 ARM 微处理器的异常种类	35
1.3 嵌入式硬件与软件系统	13	2.5.2 进入异常处理	36
1.3.1 嵌入式硬件系统	13	2.5.3 退出异常处理	37
1.3.2 嵌入式软件系统	14	2.5.4 异常向量	39
1.4 ADS 1.2 的使用	14	2.5.5 异常的优先级	40
1.4.1 ADS 集成开发环境 CodeWarrior	14	2.5.6 工程实例	40
1.4.2 编译和链接	15		
1.4.3 ADS 1.2 调试器 AXD	18		
综合实训	20		
本章小结	24		

2.6 ARM 系统中的存储器	41	4.3.4 电容式触摸屏	74
2.6.1 字节、半字与字	41	综合实训	75
2.6.2 大端与小端方式	41	本章小结	76
2.6.3 对齐	42	练习题	76
综合实训	42	第 5 章 通信接口	77
本章小结	43	5.1 PDA 中的通信接口	77
练习题	43	5.2 串行接口	78
第 3 章 存储器	44	5.2.1 串行通信的相关概念	78
3.1 PDA 中的存储器	44	5.2.2 RS-232C 串行接口	80
3.2 存储器使用的一般规则	45	5.2.3 串行接口的相关 寄存器	82
3.2.1 常用术语	45	5.3 USB 接口	84
3.2.2 存储器的三级结构	45	5.3.1 USB 接口的相关概念	84
3.2.3 存储器的分类	46	5.3.2 USB 互连	86
3.2.4 存储器的性能指标	47	5.3.3 USB 主机	86
3.3 动态随机存储器	47	5.3.4 USB 设备	87
3.3.1 DRAM	47	5.3.5 USB 的不同接口与 信号线	87
3.3.2 SDRAM	48	5.4 蓝牙	89
3.4 闪速存储器	51	5.4.1 蓝牙的相关概念	89
3.4.1 NOR 闪存	51	5.4.2 蓝牙通信的原理	91
3.4.2 NAND 闪存	51	5.4.3 蓝牙核心系统	91
3.5 存储卡	51	5.5 IEEE 802.11	93
3.5.1 CF 卡	52	5.5.1 IEEE 802.11 标准的相关 概念	93
3.5.2 SD 卡	54	5.5.2 IEEE 802.11 系列标准	94
3.5.3 记忆棒	56	5.5.3 无线网卡	96
综合实训	58	综合实训	96
本章小结	58	本章小结	96
练习题	58	练习题	97
第 4 章 输入输出设备	60	第 6 章 电源管理	99
4.1 PDA 中的输入输出设备	60	6.1 电源管理的意义和作用	99
4.2 液晶显示器	61	6.2 电源管理的方法	100
4.2.1 显示器的分类	61	6.2.1 嵌入式硬件系统的电源 管理	100
4.2.2 液晶的工作原理	62	6.2.2 嵌入式操作系统的电源 管理	103
4.2.3 液晶显示器	63	本章小结	104
4.2.4 字符型液晶显示器	66	练习题	104
4.2.5 图形液晶显示器	67		
4.3 触摸屏	69		
4.3.1 触摸屏的分类	69		
4.3.2 四线电阻式触摸屏	71		
4.3.3 五线电阻式触摸屏	73		

第二部分 嵌入式系统软件

第 7 章 ARM 的指令系统	106	8.2.1 嵌入式系统软硬件的 划分	145
7.1 ARM 汇编语言的基本 组成	106	8.2.2 嵌入式系统软硬件的协 同设计	146
7.1.1 汇编程序的基本结构 ..	106	8.3 嵌入式软件系统设计	149
7.1.2 条件执行	107	8.3.1 嵌入式软件的开发	149
7.1.3 数据处理指令的寻址 ..	109	8.3.2 汇编语言与 C/C++ 语言的 混合编程	150
7.1.4 存储器寻址	111	8.3.3 ARM 指令与 Thumb 指令 的混合编程	160
7.1.5 堆栈寻址	114	8.4 ARM 异常处理程序	162
7.2 ARM 指令系统	115	8.4.1 复位异常处理程序	162
7.2.1 分支跳转指令	115	8.4.2 未定义指令异常处理 程序	162
7.2.2 数据处理指令	116	8.4.3 软件中断处理程序	162
7.2.3 乘法指令	123	8.4.4 预取中止与数据中止异常 处理程序	166
7.2.4 内存数据访问指令	126	8.4.5 IRQ 和 FIQ 异常处理 程序	167
7.2.5 批量内存数据访问 指令	127	综合实训	174
7.2.6 状态寄存器访问指令 ..	128	本章小结	177
7.2.7 信号量操作指令	129	练习题	177
7.3 Thumb 指令系统	129	第 9 章 嵌入式系统的启动代码	178
7.3.1 Thumb 指令的几点 说明	129	9.1 产品中的启动代码	178
7.3.2 数据处理指令	130	9.2 启动代码	180
7.3.3 内存数据访问指令	131	9.2.1 启动代码简介	180
7.4 伪指令	133	9.2.2 启动代码中常用的通信 协议	181
7.4.1 地址定义伪指令	133	9.3 ARM 应用系统中的启动 代码	181
7.4.2 数据定义伪指令	134	9.3.1 典型启动代码	181
7.4.3 符号定义伪指令	135	9.3.2 关于启动代码的几个 问题	187
7.4.4 汇编控制伪指令	136	本章小结	189
7.4.5 其他伪指令	137	练习题	189
综合实训	138	第 10 章 ARM 编译器与链接器	190
本章小结	139	10.1 ARM 编译器与链接器 概述	190
练习题	140		
第 8 章 嵌入式系统软件设计	141		
8.1 嵌入式操作系统	141		
8.1.1 嵌入式操作系统的 产品	142		
8.1.2 嵌入式操作系统的主要 构成	144		
8.2 硬件系统的协同设计 ..	145		

10.2 ARM 编译器	191	10.4 ARM 链接器	213
10.2.1 ARM C/C++编译器	191	10.4.1 ARM C/C++链接器	213
10.2.2 ARM 编译器的 关键字	194	10.4.2 ARM 的映像文件	220
10.2.3 语言扩展	200	10.4.3 链接器对代码的优化	222
10.2.4 C 和 C++程序实现 细节	202	10.4.4 几个简单的链接实例	223
10.3 ARM 中的 C/C++库	206	10.4.5 链接器使用的符号	225
10.3.1 C/C++的运行时库	206	10.4.6 使用另一个映像文件中 的符号	226
10.3.2 使用 C 运行时库生成 应用程序	207	10.4.7 链接器对 C/C++库的 访问	228
10.3.3 不使用 C 运行时库生成 应用程序	210	10.5 分散加载	229
10.3.4 裁减 C 运行时库以适应 新的运行环境	212	10.5.1 分散加载简介	229
		10.5.2 分散加载描述文件	229
		本章小结	235
		练习题	235
第三部分 嵌入式系统的调试			
第 11 章 嵌入式系统的调试工具	238	11.3 ARM 集成开发环境及调 试解决方案	247
11.1 嵌入式系统的调试	238	11.3.1 ARM 集成开发环境	247
11.1.1 调试器和仿真器	238	11.3.2 ARM 嵌入式系统调试 解决方案	248
11.1.2 调试器或仿真器的选择	240	本章小结	248
11.2 ARM 的调试工具	241	练习题	249
11.2.1 ARMulator	241		
11.2.2 JTAG	241		
11.2.3 ARM 仿真器	244		
参考文献	250		

第一部分

嵌入式系统硬件

第 1 章 一个典型的嵌入式系统

学习目标

- 熟悉嵌入式系统的基本概念。
- 熟悉基于 ARM 体系结构的产品。
- 熟悉 ADS 1.2 的使用方法。

本章重点

- 嵌入式系统的概念。
- ADS 1.2 的使用。

本章难点

- ADS 1.2 的使用。

学习指导

- 了解嵌入式系统的相关概念、嵌入式系统的特点和应用领域会对掌握嵌入式系统的研发技术有积极的帮助，所以要花一定的时间全面了解整个嵌入式系统行业的发展现状，并积极思考，定位自己将来在嵌入式系统行业中的位置。
- ADS 1.2 是常用的嵌入式系统开发工具，需要熟悉其使用方法和特点。

1.1 嵌入式系统

PDA 是一个典型的嵌入式系统，其硬件系统包含微处理器、内存、串口（即串行接口）、无线网卡、蓝牙、SDIO、USB 接口、声卡、照相机接口、LCD 控制器和 LCD 显示器、触摸屏等。软件系统的核心是嵌入式操作系统。

如图 1-1 所示是 PDA 的硬件框图，从中可以了解该典型嵌入式产品硬件的基本构成。其中核心 SoC（System on Chip）芯片是 i.MX515 及相关外围电路（电池、电源管理、时钟电路等），周边是一些基本的接口电路以实现功能扩展。这样一款芯片不仅可以用来做 PDA，还可以用来做上网笔记本电脑、电子书、移动互联网设备（MID）、便携式媒体播放器（PMP）、便携式导航设备（PND）及游戏机等。

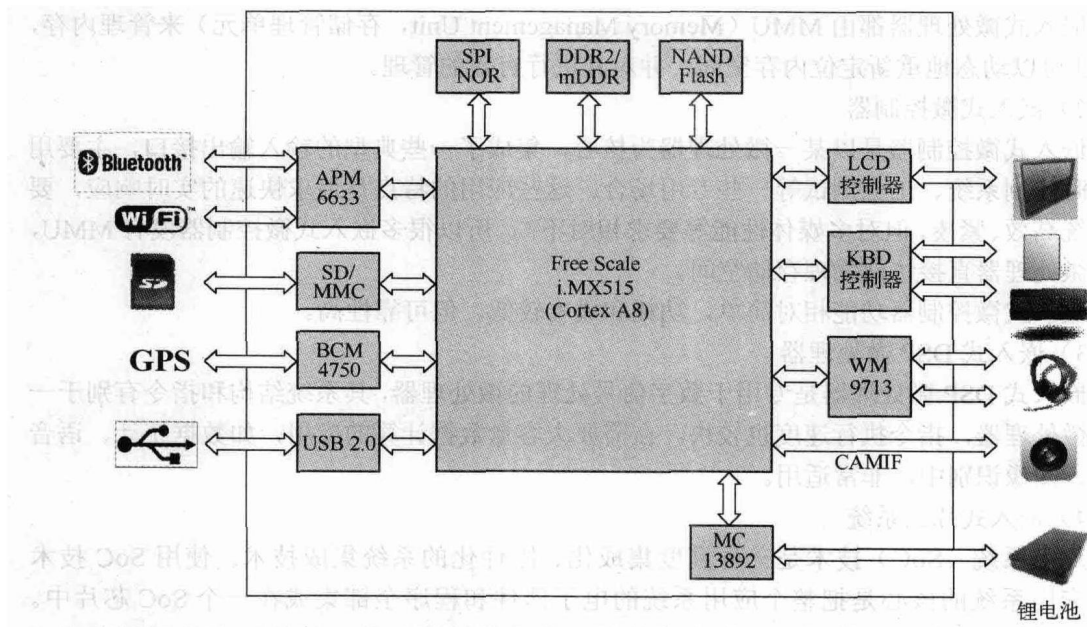


图 1-1 PDA 的硬件框图

1.1.1 嵌入式系统的定义

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，适应应用系统对功能、可靠性、成本、体积、功耗等有严格要求的专用计算机系统。

(美国)电气和电子工程师学会 (IEEE) 对嵌入式系统的定义为：“Device used to control, monitor, or assist the operation of equipment, machinery or plants (用于控制、监视或支持设备、机器或工厂运行的器件)”。

嵌入式系统的全称是嵌入式计算机系统，所以其具有计算机系统的基本特性，包含硬件系统和软件系统。硬件系统同样由处理器、存储器、输入设备和输出设备组成。有的嵌入式微处理器还集成有存储器。

基于其是一个专用的计算机系统，嵌入式系统也常常被称为“看不见的计算机”，既有通用接口，也有专用接口；既有一些通用的输入输出 (I/O) 设备，也有一些专用的输入输出设备。

1.1.2 嵌入式微处理器

嵌入式系统使用的微处理器与 PC (个人计算机) 上的微处理器有很大的不同，这是由嵌入式系统的专用特性所决定的。从应用角度来看，嵌入式微处理器主要分为 4 类，各自的应用领域及对软硬件资源的要求迥异，在结构设计上的差别也很大。

1. 嵌入式微处理器的类型

1) 嵌入式微处理器

嵌入式微处理器是功能比较完善的微处理器，常用于比较复杂的应用。这些应用要求微处理器提供足够高的性能和灵活的存储系统，系统中的各类接口和资源相对丰富。

嵌入式微处理器都由 MMU (Memory Management Unit, 存储管理单元) 来管理内存, MMU 可以动态地重新定位内存空间, 并对其进行有效的管理。

2) 嵌入式微控制器

嵌入式微控制器是以某一微处理器为核心, 集成了一些典型的输入输出接口, 主要用于实时控制系统、工业测试等一些专用场合。这些应用的特点是追求快速的实时响应, 要求系统高效、紧凑, 但对多媒体性能等要求相对不高。所以很多嵌入式微控制器没有 MMU, 而由微处理器直接访问物理存储空间。

嵌入式微控制器功能相对简单, 功耗和成本较低, 但可靠性高。

3) 嵌入式 DSP 微处理器

嵌入式 DSP 微处理器是专用于数字信号处理的微处理器, 其系统结构和指令有别于一般的微处理器, 指令执行速度也较快。在需要大容量数据计算的应用, 如数据统计、语音处理、图像识别中, 非常适用。

4) 嵌入式片上系统

片上系统 (SoC) 技术是一种高度集成化、固件化的系统集成技术。使用 SoC 技术设计应用系统的核心是把整个应用系统的电子器件和程序全部集成在一个 SoC 芯片中。在使用 SoC 技术时, 除了无法集成的外部电路或机械部分外, 其他所有的系统电路一般均由 SoC 芯片来实现。

目前, 根据应用的需求, 很多嵌入式 DSP 微处理器与嵌入式微处理器集成在一起, 形成一个双核的 SoC 芯片。

2. 嵌入式微处理器的选择

在为嵌入式系统选择微处理器时, 主要考虑以下因素。

1) 性能与功能

微处理器必须具备足够的能力来保障产品的性能, 并能支持产品的生存周期。嵌入式微处理器的品种非常多, 针对不同的应用领域有不同的芯片类型。而且很多芯片是 SoC 芯片, 芯片上除集成微处理器核外, 还集成有 LCD 控制器、A/D 转换器、多媒体音频控制器、以太网控制器等各类功能模块, 有些芯片甚至是双核的。所以, 选择适合自己产品的一种芯片是有一定难度的, 要求研发人员熟悉整个嵌入式微处理器的产品情况, 并对各类芯片的性能有足够的了解。

除考虑芯片的性能外, 还需要考虑芯片的成本和功耗, 这些因素都会对嵌入式产品的选取产生重要影响。

2) 开发工具

要考虑已有的开发工具是否支持所选芯片。如果不支持, 就要考虑新购开发工具, 包括参考设计、编译环境、测试器、代码库、评估版等。这时需要考虑开发工具的价格及性能, 包括集成开发环境 (IDE) 的适用性、调试能力、代码优化能力、硬件工具的性能表现及今后可以扩展的性能等。

3) 操作系统支持

嵌入式系统越来越复杂, 所以采用嵌入式操作系统来管理整个系统也越来越普遍。在选择微处理器芯片时, 要综合考虑嵌入式操作系统的性能。因为操作系统不仅取决于微处理器芯片, 还会影响应用软件的编制、调试工具的选择以及产品性能等。

目前嵌入式操作系统非常多, 而主要的几款产品, 如 VxWorks、Windows CE、Linux

及 Android 等，都支持基于 ARM 微处理器的芯片。

4) 应用软件

应用软件的多少取决于嵌入式操作系统的类型。这也是一个会直接影响产品性能的重要因素。如果产品属于消费类电子，则应该优先考虑在市场上认可度好、应用较广而且应用软件丰富的嵌入式操作系统。而对于专用于工业控制或测试的嵌入式系统，由于没有应用软件的扩展问题，所以并不需要过多考虑这类问题。

除了以上因素外，在实际应用时还有许多其他可能需要考虑的因素，如市场因素、商业因素等。

1.1.3 嵌入式系统的发展历程

嵌入式系统的发展与计算机系统的发展基本同步，任何在计算机领域出现的新技术都会很快进入嵌入式系统领域。嵌入式系统中一些新技术的使用，也会对其他计算机应用领域产生影响。

1. 嵌入式系统的出现和兴起（1960—1970）

20 世纪 60 年代，以晶体管、磁芯存储为基础的计算机开始用于航空及军事领域。在军事领域中，为了满足可靠性、体积及重量等方面的严格要求，为各类武器系统设计出了五花八门的专用嵌入式计算机系统。

2. 嵌入式系统开始走向繁荣（1971—1989）

嵌入式系统大发展是在微处理器问世之后。集成电路制造工艺水平越来越高，芯片制造商开始把嵌入式应用所需要的微处理器、I/O 接口、A/D 转换器、D/A 转换器集成到一个芯片中，制造出面向应用的各种微控制器。而软件技术的进步使嵌入式系统也日臻完善。

3. 嵌入式系统应用走向纵深（1990—现在）

进入 20 世纪 90 年代后，在分布控制、柔性制造、数字化通信和消费类电子等巨大需求的牵引下，嵌入式系统的硬件、软件技术进一步加速发展、应用领域进一步扩大。手机、MP4、数码相机、DVD 播放机、数字电视、路由器、交换机等都是典型的嵌入式系统。

而今，嵌入式系统的应用已经覆盖了不同的工业领域，人们的生活也越来越离不开嵌入式产品。

1.1.4 嵌入式系统的特点

1. 嵌入式系统通常是形式多样、面向特定应用的软硬件综合体

嵌入式系统一般针对特定的应用，其硬件和软件都必须高效率地设计，量体裁衣、去除冗余。每种嵌入式微处理器大多专用于某个或几个特定的应用，工作在为特定用户群设计的系统中。而且通常都具有低功耗、体积小、集成度高等特点，能够把通用微处理器中许多由板卡完成的任务集成在芯片内部。

嵌入式系统的软件是嵌入式操作系统和应用程序两种软件一体化的程序。在具体的产品中，很难分清哪些是操作系统的程序，哪些是应用程序。

2. 嵌入式系统得到多种类型的处理器和处理器体系结构的支持

通用计算机的处理器和体系结构类型较少，而且主要掌握在少数几家大公司手里，而嵌入式系统可采用多种类型的处理器和处理器体系结构。

在嵌入式微处理器产业链上, IP 设计、面向应用的特定嵌入式微处理器的设计、芯片制造已各自形成巨大的产业。大家分工协作, 形成多赢模式。目前, 在嵌入式微处理器市场上, 有上千种嵌入式微处理器和几十种嵌入式微处理器体系结构可以选择。

3. 嵌入式系统通常极其关注成本

成本是嵌入式产品竞争的关键因素之一, 尤其是消费类电子产品。嵌入式系统成本主要包括开发成本和产品成本。开发成本包括开发软件及开发工具的投入、开发人员的培训投入等; 产品成本包括硬件成本、外壳包装和软件版税等, 再加上销售、公司的各项费用。成本对于嵌入式产品的影响常常会决定产品的生存, 有效地控制成本是嵌入式研发人员必须牢记的一条原则。例如, 代码的长度和执行效率会直接影响内存使用的多少, 所以如何在保证性能不变的前提下, 尽量减少代码存储空间和执行空间是降低成本的重要手段。

4. 嵌入式系统有实时性和可靠性的要求

大多数实时系统都是嵌入式系统, 而嵌入式系统多数也有实时性的要求。嵌入式系统的软件一般是直接从内存中运行或将程序从外存加载到内存中运行, 而且一般都要求快速启动。

嵌入式系统一般要求具有出错处理和自动复位功能, 特别是对于一些在极端环境下运行的嵌入式系统而言, 其可靠性设计尤其重要。大多数嵌入式系统的软件中一般都包括一些可靠性机制, 比如硬件的看门狗定时器、软件的内存保护和重启机制等, 以保证系统在出现问题时能够重新启动, 保障系统的健壮性。

5. 嵌入式系统使用的操作系统能适应多种类型处理器、可剪裁、轻量型、实时可靠、可固化

基于嵌入式系统应用的特点, 与嵌入式微处理器类似, 嵌入式操作系统也呈现百花齐放的局面。大多数商用嵌入式操作系统可同时支持不同类型的嵌入式微处理器, 而且用户可根据应用的具体情况进行剪裁和配置。

6. 嵌入式系统开发需要专门工具和特殊方法

多数嵌入式系统开发意味着软件与硬件的并行设计和开发, 其开发过程一般分为几个阶段: 产品定义, 软硬件设计与实现, 软件与硬件集成, 产品测试与发布, 维护与升级, 运行。

1.1.5 嵌入式系统开发的特殊要求

嵌入式系统的应用领域实在是太大了, 其开发所需要的知识不仅包含微处理器、计算机结构、微型计算机原理、计算机接口技术、电路与电子技术、集成电路技术、操作系统、数字信号处理与人工智能、固件及中间件的设计与集成, 还包括传感器技术、数据采集检测、控制技术及相关专业的知识。

1. 嵌入式系统既是软件又是硬件

嵌入式系统的开发不仅需要编制软件, 更需要软件与硬件的协调来实现其功能。这就要求嵌入式系统开发人员要有一定的硬件知识背景, 熟悉嵌入式硬件系统的工作原理和工作过程; 熟悉相关寄存器的设置、时序及数据的传送协议; 熟悉中断的产生, 以及中断的处理方法和处理流程。所以, 开发嵌入式应用程序, 要熟悉硬件平台的功能及使用, 才能编制出实用、高效的程序。

2. 嵌入式应用程序的调试过程有特殊的方法

嵌入式系统的编程是在宿主机上完成的, 但代码的运行却是在目标平台上。所以程序编译链接完成后必须从宿主机下载到目标平台运行, 而代码的调试必须凭借各种不同的工

具与仪器来完成。

即使是最好的调试工具，也不能完全真实地反映代码的运行情况，因为嵌入式系统的应用环境与硬件密切相关，代码的运行受限于现场硬件的真实状态，而这些是任何仿真器都无法再现的。

3. 嵌入式系统开发工具的差异

目前嵌入式系统应用开发工具种类繁多，彼此之间有很多不同。在开发产品之前，不仅要考虑所使用的嵌入式操作系统的性能，还要考虑开发工具的性能和开发人员过去使用的平台。协调开发小组人员使用平台间的差异，就可能需要很长时间。

1.1.6 嵌入式系统对人才的能力要求

1. 系统集成能力

嵌入式系统开发人员所面对的是五花八门的开发工具，性能高低不一的操作系统，以及各种不同的独特应用需求。如何在面对任何一类特殊应用需求时，都能够从现有的产品中挑选合适的平台以配合开发产品最终的应用；如何有效地规划、设计以及合理地划分软硬件功能，以缩短产品开发周期及提高产品可靠性；如何集成现有的技术并与其他协作单位有效合作；这些都是对嵌入式系统研发人员的最高要求。

2. 编程能力

对嵌入式系统开发人员的基本要求是掌握汇编语言、C 或 C++ 语言和 Java 语言。要和硬件打交道，汇编语言和 C 语言是必须掌握的。

在程序调试过程中也一样会遇到汇编指令，因为调试中开发人员所面对的指令，一般是 C 程序编译链接后产生的汇编代码，熟悉汇编代码可大大提高程序的调试效率。

1.1.7 嵌入式系统行业对人才的需求

1. 系统集成与设计人员

要求熟悉各类操作系统和相关工具，对硬件、操作系统及应用软件设计均有较深研究，参与并主持过产品开发，熟悉开发过程，熟练使用调试工具，对解决调试中出现的问题较有心得。

具有较强的专业基础知识，解决问题的思路清晰。而这一切应该是在专业学习过程中，通过大量的实践训练逐步培养出来的。

2. 驱动程序开发人员

编写设备驱动程序会耗费嵌入式开发人员的很多时间，这是由于嵌入式系统是一种量身定制的系统，应用可能会面对不同的硬件环境。对不同的配置需要，应用要求也就不同。系统中每增加一种设备，就需要为其编写相应的驱动程序，所以其工作量也比较大。

有些公司在开发程序过程中并不设置专门的驱动程序开发人员，而是把相关工作交由其他公司来做，其好处是编写的驱动程序更专业，与操作系统的配合更理想，但为此也要花费不菲的费用。

3. 应用程序开发人员

应用程序开发人员的工作是在嵌入式系统开发平台上，编写应用程序。这类人员是嵌入式系统应用开发的主力。产品的设计和功能的实现都是由他们来完成的。他们常常困惑

于不知如何使产品有新的创意，而不是如何去实现它，所以需要能够紧盯市场。从实际意义上来讲，他们不仅仅是做应用，更多的是做市场。

1.2 ARM

1.2.1 ARM 公司简介

1. ARM 公司

ARM (Advanced RISC Machines) 公司是一家总部位于英国、专门从事嵌入式微处理器芯片设计的公司。该公司只提供芯片的设计，并不进行芯片的生产和销售。其业务模型是把自己的知识产权授权给合作伙伴进行相关产品的设计、生产及销售。这一方式使 ARM 公司在发展过程中，与全球各大芯片厂商成为合作伙伴，而不是竞争对手，赢得了极好的发展机会。经过近 20 年的发展，ARM 公司成为目前全球领先的芯片设计公司，应用遍及嵌入式系统的各个领域，成为嵌入式行业内的亮点。与此同时，ARM 体系的产品还得到了第三方公司的支持，提供大量调试工具和其他产品。

ARM 公司的产品包括 16/32 位 RISC 微处理器、数据引擎、软件、芯片物理 IP、开发工具，以及模拟和高速连接产品等。

2. CISC 和 RISC

1) CISC

CISC (Complex Instruction Set Computer, 复杂指令集计算机) 是通过不断在微处理器的指令系统中引入新的复杂指令集，来提升计算机的性能。为支持这些新增的指令，计算机的结构也越来越复杂。但是，在 CISC 指令集的各种指令中，绝大多数指令并不经常使用。这就造成 CISC 微处理器结构越来越复杂，指令系统越来越庞大。

CISC 的微处理器主要有以下特点：

- 由硬件完成部分软件功能，硬件复杂性增加，芯片成本高。
- 代码长度相对较短，但指令的执行周期数较大。
- 存在大量的混杂型指令集，有简单、快速的指令，也有复杂的多周期指令。
- 对高级语言的支持由硬件完成。
- 复杂的寻址模式，支持内存到内存寻址。
- 控制单元由微码实现。
- 寄存器较少。

2) RISC

RISC (Reduced Instruction Set Computer, 精简指令集计算机) 于 1979 年由美国加州大学伯克利分校提出。RISC 不仅减少了指令系统的代码长度，而且也使计算机的结构更加简单、合理，运算速度也明显提高。

RISC 的微处理器主要有以下特点：

- 选取使用频率最高的简单指令，避免复杂指令。
- 指令长度固定。
- 指令格式和寻址方式种类减少。
- 使用流水线降低指令的执行周期数，但代码长度相对较大。