

嵌入式技术与应用

符意德 陆阳 编著

嵌入式系统原理 及接口技术

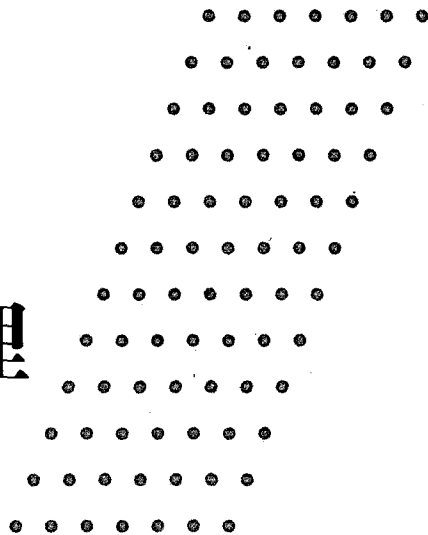
清华大学出版社



嵌入式技术与应用

符意德 陆阳 编著

嵌入式系统原理 及接口技术



清华大学出版社
北京

内 容 简 介

本书以三星公司的 S3C2410 CPU 为平台,系统地描述了基于 ARM9 微处理器核的嵌入式系统体系结构及其接口技术。书中采用原理加设计实例的方法来介绍嵌入式系统硬件平台设计,并且从具体的实例中归纳出了具有普遍指导意义的嵌入式系统硬件平台设计原理和方法。

本书适合作为高等院校计算机、电子信息及相关专业的本科教材,也适于从事嵌入式系统研究与开发的技术人员及普通读者参考。

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 13501256678 13801310933

图书在版编目(CIP)数据

嵌入式系统原理及接口技术/符意德,陆阳编著. —北京:清华大学出版社,2007.10

(嵌入式技术与应用)

ISBN 978-7-302-15646-8

I. 嵌… II. ①符… ②陆… III. ①微型计算机—系统设计 ②微型计算机—接口
IV. TP36

中国版本图书馆 CIP 数据核字(2007)第 103413 号

责任编辑:付弘宇

责任校对:时翠兰

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编:100084

c-service@tup.tsinghua.edu.cn

社 总 机:010-62770175 邮购热线:010-62786544

投稿咨询:010-62772015 客户服务:010-62776969

印 刷 者:北京市清华园胶印厂

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:185×260 印 张:23 字 数:557 千字

版 次:2007 年 10 月第 1 版 印 次:2007 年 10 月第 1 次印刷

印 数:1~4000

定 价:29.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:022637-01

前 言

嵌入式系统目前已广泛应用到了信息家电、通信设备、医疗仪器、军事装备等众多领域,它将作为后 PC 时代的核心,成为一种新的计算平台。虽然传统的嵌入式系统起源于 20 世纪 70 年代初,至今已有很长时间,以 MCS-51 单片机为基础平台的“单片机原理及应用”课程也早已在许多高校开设,但是,随着时代的进步、技术的发展,对嵌入式系统功能的要求越来越复杂,传统的以 MCS-51 系列单片机为代表的嵌入式系统硬件平台,以及传统的嵌入式系统程序设计方法已不能满足快速、高效地设计复杂嵌入式系统的要求。因此,系统地开设嵌入式系统原理及设计方法的相关课程,培养计算机科学与技术、通信工程、电子工程等相关专业的本科生及研究生,使其能熟练掌握复杂嵌入式系统的设计方法,是十分必要的。基于 32 位嵌入式微处理器的系统其硬件构件较复杂,用户应用软件的复杂度也成倍增长。因此,要完整地学习嵌入式系统的设计知识,需要进行多门课程的学习。“嵌入式技术与应用”系列教材就是基于这样的观点来组织编写的。

嵌入式系统涉及的知识点非常多,因此,对于初学者来说,结合自己的目标,找准学习嵌入式系统设计知识的切入点,是非常必要的。狭义地说,学习嵌入式系统设计知识可以从两个不同的层面进行切入:第一层面,针对将来只是应用嵌入式系统硬件、软件平台来进行二次开发的学生而言,应侧重学习基于某个嵌入式系统平台(包括硬件平台和软件平台)进行应用系统设计和开发的能力,即主要是学习在某个嵌入式操作系统(如嵌入式 Linux)环境下的应用程序的编写、调试,学习其 API 函数的使用及 I/O 接口部件的驱动程序编写等;第二层面,针对将来从事嵌入式系统平台设计或者需要结合应用环境设计专用硬件平台的学生而言,需要重点学习嵌入式系统体系结构及接口设计原理,即主要学习某个具有代表性的嵌入式 CPU(如 ARM 系列)的内部寄存器结构、汇编指令系统、中断(异常)管理机制及常用的外围接口,同时要学习无操作系统下的编程技术。更进一步地,还需要学习启动程序的编写和操作系统移植等方面的知识。

本书是“嵌入式技术与应用”系列教材之一,重点介绍嵌入式系统的硬件平台组成原理及其接口技术。书中没有局限于某个具体的嵌入式微处理器,而是用了大量的篇幅来介绍其原理。在介绍原理的同时,又列举了许多基于三星 S3C2410 芯片的设计实例,从而使原理、概念更为具体化。

本书由符意德、陆阳主编,王丽芳参加了第 1 章的编写,卢波参加了第 5 章的编写,孙晔参加了第 7 章的编写,朱剑峰参加了第 8 章的编写,陈文娟参加了第 9 章的编写。在本书的编写过程中,参考了许多专家学者的成果,在此向他们表示感谢!感谢本书责任编辑的支持!感谢家人的关心和支持!

嵌入式系统目前正处于一个快速发展的阶段,新的技术和应用成果不断地涌现,囿于编者的水平,书中的疏漏和错误之处在所难免,希望广大读者批评、指正。如果读者在本书的使用中有何问题或建议,请发邮件至: fuhy@tup.tsinghua.edu.cn。

编 者
2007 年 3 月
于南京紫金山麓

目 录

第 1 章 绪论	1
1.1 嵌入式系统的发展	1
1.1.1 嵌入式系统硬件平台的发展	1
1.1.2 嵌入式系统软件平台的发展	3
1.2 嵌入式计算的特点	5
1.2.1 嵌入式系统的要求	6
1.2.2 为什么以微处理器为核心	6
1.2.3 嵌入式系统设计所面临的问题	7
1.3 嵌入式系统的应用	8
1.4 嵌入式系统软件开发环境	9
1.4.1 集成开发环境简介	10
1.4.2 建立单工程项目	10
1.4.3 配置生成目标	18
1.4.4 编译和连接工程项目	28
1.4.5 嵌入式系统调试	30
1.5 本书内容简介	32
习题	36
第 2 章 ARM9 体系结构	37
2.1 ARM 处理器系列	37
2.1.1 ARM7 系列	37
2.1.2 ARM9 系列	38
2.1.3 ARM9E 系列	39
2.1.4 ARM10E 系列	39
2.1.5 SecurCore 系列	40
2.1.6 StrongARM 系列	40
2.1.7 Xscale 系列	41
2.2 ARM9 微处理器	42
2.2.1 ARM9 的结构特点	42
2.2.2 ARM9 指令集的特点	44
2.2.3 ARM9 的工作模式	45
2.3 ARM9 的存储组织结构	46
2.3.1 大端存储和小端存储	46

2.3.2	I/O 端口的访问方式	47
2.3.3	内部寄存器	48
2.4	ARM9 的异常	51
2.4.1	异常的类型及向量地址	51
2.4.2	异常的优先级	53
2.4.3	进入异常和退出异常	53
2.5	ARM9 的寻址方式	54
	习题	56
第 3 章	ARM9 指令系统	58
3.1	ARM 指令集	58
3.1.1	寄存器装载及存储指令	59
3.1.2	算术和逻辑指令	62
3.1.3	比较指令	65
3.1.4	分支指令	66
3.1.5	软件中断指令	67
3.1.6	汇编器伪指令	68
3.2	Thumb 指令集	70
3.3	汇编程序设计	71
3.3.1	汇编编程规则	71
3.3.2	汇编指令的特点	74
3.3.3	子程序调用	78
3.4	汇编程序实例——系统引导程序	81
3.4.1	系统引导程序的功能	81
3.4.2	系统引导程序的代码	82
	习题	85
第 4 章	存储系统机制及存储器接口	86
4.1	微处理器总线及接口控制方式	86
4.1.1	总线定时	86
4.1.2	数据读写	92
4.1.3	中断	93
4.1.4	DMA	94
4.2	存储器及存储系统机制	95
4.2.1	存储器接口方式	95
4.2.2	高速缓存机制	98
4.2.3	存储管理单元	100
4.3	存储系统地址分配	106
4.3.1	SRAM 的寻址	106

4.3.2	DRAM 的寻址	108
4.3.3	NAND Flash 的寻址	109
4.3.4	I/O 端口的寻址	111
4.4	存储系统实例	113
4.4.1	S3C2410 存储空间	113
4.4.2	SDRAM 存储器接口	119
4.4.3	NAND Flash 存储器接口	122
4.4.4	NOR Flash 存储器接口	125
	习题	126
第 5 章	中断机制	127
5.1	中断方式原理	127
5.1.1	中断操作	127
5.1.2	中断源及其识别方法	128
5.1.3	中断优先级仲裁方式	128
5.2	S3C2410 中断系统	131
5.2.1	概述	131
5.2.2	中断优先级	132
5.2.3	中断控制寄存器	134
5.3	中断编程实例	142
5.3.1	IRQ 异常处理过程	142
5.3.2	FIQ 异常处理过程	143
5.3.3	中断编程模式	144
5.3.4	实例	144
	习题	149
第 6 章	DMA 机制	150
6.1	S3C2410 芯片的 DMA 方式	150
6.1.1	DMA 请求源	150
6.1.2	DMA 的模式	151
6.1.3	DMA 操作过程	152
6.1.4	DMA 时序	152
6.2	S3C2410 芯片的 DMA 寄存器	155
6.2.1	传输控制寄存器	155
6.2.2	状态寄存器	158
6.3	DMA 操作编程	158
6.3.1	DMA 操作初始化	159
6.3.2	实例程序	159
	习题	160

第 7 章 定时部件	161
7.1 定时部件的原理	161
7.2 看门狗定时器	162
7.2.1 看门狗定时器概述.....	162
7.2.2 看门狗控制寄存器.....	162
7.2.3 使用实例.....	164
7.3 RTC 部件	165
7.3.1 概述.....	165
7.3.2 RTC 控制寄存器	166
7.3.3 编程实例.....	171
7.4 Timer 部件	173
7.4.1 Timer 部件概况	173
7.4.2 Timer 部件的操作	174
7.4.3 Timer 部件内部寄存器	178
7.4.4 应用实例.....	181
习题.....	182
第 8 章 I/O 端口	183
8.1 I/O 端口功能	183
8.1.1 端口功能定义.....	183
8.1.2 端口控制寄存器.....	187
8.1.3 端口其他寄存器.....	197
8.1.4 端口的使用实例.....	205
8.2 IIS 接口	206
8.2.1 IIS 总线格式	206
8.2.2 控制寄存器.....	209
习题.....	210
第 9 章 异步串行接口	211
9.1 异步串行通信概念	211
9.1.1 基本术语.....	211
9.1.2 RS-232C	213
9.1.3 RS-485	216
9.2 S3C2410 的串行接口	218
9.2.1 概述.....	218
9.2.2 UART 的操作	219
9.2.3 波特率的计算.....	221
9.2.4 UART 接口寄存器	222

9.3 串行通信实例	226
9.3.1 RS-232 接口电路	226
9.3.2 初始化程序	226
9.3.3 发送/接收程序	227
习题	229
第 10 章 网络接口技术	230
10.1 概述	230
10.1.1 分布式嵌入式系统结构	230
10.1.2 嵌入式网络通信方式	232
10.2 I ² C 总线技术	233
10.2.1 I ² C 协议结构	233
10.2.2 S3C2410 的 I ² C 接口	236
10.2.3 I ² C 总线专用寄存器	239
10.2.4 I ² C 总线的编程	241
10.3 CAN 总线接口技术	246
10.3.1 CAN 总线协议	247
10.3.2 CAN 总线接口实例	249
10.4 以太网接口技术	250
10.4.1 基本原理	251
10.4.2 以太网接口电路	254
10.4.3 网络协议软件实现	256
10.5 USB 接口技术	270
10.5.1 USB 的概念及特点	270
10.5.2 USB 主机	274
10.5.3 USB 设备	284
10.5.4 实例——S3C2410 的 USB 接口	290
习题	302
第 11 章 人机接口技术	303
11.1 键盘接口	303
11.1.1 按键的识别方法	303
11.1.2 键盘接口实例	305
11.2 LCD 显示器接口	308
11.2.1 LCD 显示原理	308
11.2.2 LCD 的控制器	309
11.2.3 S3C2410 芯片的 LCD 寄存器	316
11.2.4 S3C2410 芯片 LCD 寄存器的设置	322
11.3 LED 显示器接口	324

11.3.1	LED 显示控制原理	324
11.3.2	LED 显示接口实例	327
习题	330
第 12 章	嵌入式系统整体设计实例	331
12.1	设计要求	331
12.1.1	系统需求	331
12.1.2	规格说明	332
12.2	体系结构设计	335
12.2.1	软件结构	335
12.2.2	硬件组成	336
12.3	构件设计与集成	337
12.3.1	构件设计	337
12.3.2	系统集成	338
习题	338
附录	常用头文件代码	339
参考文献	356

第 1 章 绪 论

嵌入式系统是后 PC 时代被广泛使用的计算机平台。在人们日常生活、学习和工作中所接触到的仪器或设备里,都将嵌入具有强大计算能力的嵌入式系统。它不仅广泛地应用于众多成熟领域,如工业控制、家用电器、通信设备、医疗仪器、军事设备等,而且,嵌入式计算技术的不断发展还创造了许多新的应用,如 PDA、智能手机、MP4 等。可以预见,嵌入式系统将越来越深入地影响人们的生活、学习和工作。

那么,什么是嵌入式系统呢?对于嵌入式系统,目前还没有一个被广泛接受的定义。传统的定义是:嵌入在其他设备中、起智能控制作用的专用计算机系统。也可以说,它是任意包含一个可编程计算机的设备,但是这个设备不是作为通用计算机而设计的。一台通用个人电脑不能称为嵌入式系统,尽管有时会把它嵌入到某些设备中。而一台包含有微处理器的打印机、数码相机就可以称为嵌入式系统。目前较流行的定义是:嵌入式系统是以应用为中心,以计算机技术为基础,并且软硬件可裁剪,适用于应用系统对功能、可靠性、成本、体积、功耗有严格要求的专用计算机系统。它一般由嵌入式微处理器、外围硬件设备、嵌入式操作系统以及用户的应用程序等四个部分组成,用于实现对其他设备的控制、监视或管理等功能。

1.1 嵌入式系统的发展

嵌入式系统的概念很早就已经存在,早在 20 世纪 70 年代初、微处理器诞生的同时,就出现了嵌入式系统。到目前,据统计每年只有 10%~20% 的微处理器芯片用于台式机或笔记本电脑,而 80% 以上的微处理器芯片是为嵌入式系统设计和制造的。嵌入式系统已对信息技术(IT)产业产生了强有力的影响,它的出现已经改变了以台式个人计算为主的计算模式,使计算无处不在。

1.1.1 嵌入式系统硬件平台的发展

嵌入式系统硬件的核心部件是各种类型的嵌入式处理器,嵌入式系统硬件平台是随着嵌入式处理器芯片的发展而发展的。在微处理器诞生的早期阶段,微处理器的数据位是 8 位的,可直接寻址的存储器容量通常为 64KB 或 1MB。以这类微处理器为核心的嵌入式系统,其硬件平台构件相对较简单,外围电路多采用 74 系列的芯片及独立的晶体管等组成。这个阶段的嵌入式系统硬件平台所用的 CPU 主要有 Z80、Intel8080、MC6800 等。后来,随着应用系统的需要及集成电路技术的发展,出现了外围接口电路与微处理器核集成在一片芯片上的单片机,典型的如 Intel 的 MC-51 系列单片机。而今天组成嵌入式系统的基本硬件构件已较复杂,如 16 位/32 位 CPU 或特殊功能的微处理器、特定功能的集成芯片、FPGA 或 CPLD 等。

目前据不完全统计,全世界嵌入式处理器的品种总量已经超过 1000 多种,流行体系结构有 30 多个系列。嵌入式处理器的寻址空间一般从 64KB 到几个 GB,处理速度从 0.1MIPS 到 2000MIPS。根据不同的应用状况,嵌入式处理器可以分成下面几类。

1. 嵌入式微处理器(Embedded Microprocessor Unit, EMPU)

嵌入式微处理器的基础是通用计算机中的 CPU。在应用中,将微处理器装配在专门设计的电路板上,板上只保留和嵌入式应用有关的功能电路,这样可以大幅度减小系统体积和降低功耗。为了满足嵌入式应用的特殊要求,嵌入式微处理器虽然在功能上和标准微处理器基本是一样的,但在工作温度、抗电磁干扰、可靠性等方面一般都有增强。但是由于在电路板上还必须包括 ROM、RAM、总线接口、各种外设等器件,从而降低了系统的可靠性。嵌入式微处理器及存储器、总线、外设等安装在一块电路板上,因此称为单板计算机。典型的如 PC104、STD-BUS 等。近年来,德国、日本的一些公司又开发出了类似“火柴盒”式名片大小的嵌入式计算机系列 OEM 产品。嵌入式处理器目前主要有 Am186/88、386EX、SC-400、PowerPC、68000、MIPS 系列等。

2. 嵌入式微控制器(Microcontroller Unit, MCU)

嵌入式微控制器又称单片机,顾名思义,就是将整个计算机系统集成到一块芯片中。嵌入式微控制器一般以某种微处理器内核为核心,芯片内部集成 ROM/EPROM、RAM、总线、总线逻辑、定时/计数器、WatchDog、I/O、串行口、脉宽调制输出、A/D、D/A、FlashRAM、EEPROM 等各种必要功能和外设。为适应不同的应用需求,一个系列的单片机具有多种衍生产品,每种衍生产品的处理器内核都是一样的,不同的是存储器容量、外设的配置和引脚封装。这样可以使单片机最大限度地和应用需求相匹配,功能不多不少,从而降低功耗和成本。

和嵌入式微处理器相比,微控制器的最大特点是单片化,体积大大减小,从而使功耗和成本下降、可靠性提高。微控制器是目前嵌入式应用的主流。微控制器的芯片上外设资源一般比较丰富,适合于控制,因此称微控制器。

嵌入式微控制器目前的品种和数量最多,比较有代表性的通用系列包括 8051、P51XA、MCS-251、MCS-96/196/296、C166/167、MC68HC05/11/12/16、68300、ARM 系列等。另外还有许多半通用系列,如支持 USB 接口的 MCU,支持 I²C、CAN-Bus、LCD 等的 MCU。

在本书中并未把微处理器与微控制器进行严格区分,而是统称为微处理器。

3. 嵌入式数字信号处理器(Digital Signal Processor, DSP)

DSP 处理器对系统结构和指令进行了特殊设计,使其适合于执行 DSP 算法,编译效率较高,指令执行速度也较高。在数字滤波、FFT、谱分析等方面 DSP 算法正在大量进入嵌入式领域,DSP 应用正从在通用单片机中以普通指令实现 DSP 功能,过渡到采用嵌入式 DSP 处理器。嵌入式 DSP 处理器有两个发展来源,一是 DSP 处理器经过单片化、EMC 改造、增加片上外设成为嵌入式 DSP 处理器,TI 的 TMS320C2000/C5000 等属于此范畴;二是在通用单片机或 SoC 中增加 DSP 协处理器,例如 Intel 的 MCS-296 和 Infineon(Siemens)的 TriCore。

推动嵌入式 DSP 处理器发展的另一个因素是嵌入式系统的智能化,例如各种带有智能逻辑的消费类产品、生物信息识别终端、带有加解密算法的键盘、ADSL 接入、实时语音压缩解压系统、虚拟现实显示等。这类智能化算法一般都是运算量较大,特别是向量运算、指针

线性寻址等较多,而这些正是 DSP 处理器的长处所在。

嵌入式 DSP 处理器比较有代表性的产品是 TI 的 TMS320 系列和 Motorola 的 DSP56000 系列。TMS320 系列处理器包括用于控制的 C2000 系列,移动通信的 C5000 系列,以及性能更高的 C6000 和 C8000 系列等。DSP56000 目前已经发展成为 DSP56000, DSP56100, DSP56200 和 DSP56300 等几个不同系列的处理器。另外 Philips 公司今年也推出了基于可重置嵌入式 DSP 结构低成本、低功耗技术上制造的 R. E. A. L DSP 处理器,特点是具备双 Harvard 结构和双乘/累加单元,应用目标是大批量消费类产品。

4. 嵌入式片上系统(System on Chip, SoC)

随着 EDA(电子设计自动化)的推广和 VLSI(超大规模集成电路)设计的普及化及半导体工艺的迅速发展,在一个硅片上实现一个更为复杂系统的时代已来临。各种通用处理器内核将作为 SoC 设计公司的标准库,和许多其他嵌入式系统外设一样,成为 VLSI 设计中一种标准的器件,用标准的 VHDL(超高速集成电路硬件描述语言)等语言描述,存储在器件库中。用户只须定义出其整个应用系统,仿真通过后就可以将设计图交给半导体工厂制作样品。这样除个别无法集成的器件以外,整个嵌入式系统大部分均可集成到一块或几块芯片中去,应用系统电路板将变得很简洁,对于减小体积和功耗、提高可靠性非常有利。

SoC 可分为通用和专用两类。通用系列包括 Infineon 的 TriCore、Motorola 的 M-Core、某些 ARM 系列器件、Echelon 和 Motorola 联合研制的 Neuron 芯片等。专用 SoC 一般专用于某个或某类系统中,不为一般用户所知。一个有代表性的产品是 Philips 的 Smart XA,它将 XA 单片机内核和支持超过 2048 位复杂 RSA 算法的 CCU 单元制作在一块硅片上,形成一个可加载 Java 或 C 语言的专用 SoC,可用于公众互联网安全方面。

1.1.2 嵌入式系统软件平台的发展

嵌入式系统的软件部分包括系统软件(即嵌入式操作系统)和应用软件,通常设计人员把这两种软件组合在一起,作为一个有机的整体存在。应用程序控制着系统的动作和行为,而操作系统控制着应用程序与嵌入式系统硬件的交互作用。嵌入式系统软件的要求与台式通用计算机有所不同,其特点主要有以下几点。

(1) 软件要求固化存储

为了提高执行速度和系统可靠性,嵌入式系统中的软件(包括系统软件和应用软件)一般都固化在外部存储器芯片或单片机内部存储器中,而不是存储于磁盘等载体中。

(2) 软件代码要求高效率、高可靠性

尽管半导体技术的发展使微处理器速度不断提高、单片存储器容量不断增加,但在大多数应用中,存储空间仍然是宝贵的,还存在实时性的要求。为此要求程序编写和编译工具的效率要高,以减少程序二进制代码长度、提高执行速度。较短的代码同时也提高了系统的可靠性。

(3) 系统软件有较高的实时性要求

在多任务嵌入式系统中,对重要性各不相同的任务进行统筹兼顾的合理调度是保证每个任务及时执行的关键,单纯通过提高微处理器速度是无法完成和没有效率的,这种任务调度只能由优化编写的系统软件来完成,因此对于许多嵌入式系统而言,其软件的实时性是基本要求。

虽然嵌入式系统是以应用为核心的,其应用软件是实现嵌入式系统功能的关键。在嵌入式系统发展的初期,嵌入式系统的软件是一体化的,即软件中没有把系统软件和应用软件独立开来,整个软件是一个大的循环控制程序,设备控制功能模块、人机操作模块、硬件接口模块等通常在这个大循环中。但是,随着应用要求的越来越复杂,例如,需要嵌入式系统能连接 Internet 网,需要嵌入式系统具有多媒体处理功能,需要嵌入式系统具有丰富的人机操作界面等。若再按照传统方法把嵌入式系统设计成一个大的循环控制程序,不仅费时、费力,而且设计的程序可能不能满足需求。因此,嵌入式系统的系统软件平台(即嵌入式操作系统)也得到了迅速的发展。

从 20 世纪 80 年代起,国际上就有一些 IT 组织和公司,开始进行商用嵌入式操作系统的研发。到了 21 世纪初,嵌入式操作系统在嵌入式系统的开发中得到了广泛应用,并在开发中起到了关键的作用。这其中比较著名和流行的嵌入式操作系统有以下几种。

(1) Windows CE

Microsoft Windows CE 是从整体上为有限资源的平台设计的多线程、完整优先权、多任务的操作系统。它的模块化设计允许它对于从掌上电脑到专用的工业控制器的用户电子设备进行定制。操作系统的基本内核需要至少 200KB 的 ROM。

(2) VxWorks

VxWorks 是目前嵌入式系统领域中使用最广泛、市场占有率最高的系统。它支持多种处理器,如 x86、i960、Sun Sparc、Motorola MC68xxx、MIPS RX000、POWER PC 等。大多数的 VxWorks API 是专有的,采用 GNU 的编译和调试器。

(3) pSOS

ISI 公司已经被 WinRiver 公司兼并,现在 pSOS 属于 WindRiver 公司的产品。这个系统是一个模块化、高性能的实时操作系统,专为嵌入式微处理器设计,提供一个完全多任务环境,在定制的或是商业化的硬件上提供高性能和高可靠性。可以让开发者根据操作系统的功能和内存需求定制成每一个应用所需的系统。开发者可以利用它来实现从简单的单个独立设备到复杂的、网络化的多处理器系统。

(4) QNX

QNX 是一个实时的、可扩充的操作系统,它部分遵循 POSIX 相关标准,如 POSIX.1b 实时扩展。它提供了一个很小的微内核以及一些可选的配合进程。其内核仅提供 4 种服务:进程调度、进程间通信、底层网络通信和中断处理,其进程在独立的地址空间运行。所有其他 OS 服务,都实现为协作的用户进程,因此 QNX 内核非常小巧(QNX4.x 大约为 12KB)而且运行速度极快。这个灵活的结构可以使用户根据实际的需求,将系统配置成微小的嵌入式操作系统或是包括几百个处理器的超级虚拟机操作系统。

(5) Palm OS

3Com 公司的 Palm OS 在 PDA 市场上占有很大的市场份额,它有开放的操作系统的应用程序接口(API),开发商可以根据需要自行开发所需要的应用程序。

上述嵌入式操作系统均为商业化的产品,价格昂贵。而且由于它们各自的源代码不公开,使得每个系统上的应用软件与其他系统都无法兼容。并且,由于这种封闭性还导致了商业嵌入式系统在对各种设备的支持方面存在很大的问题,使得对它们的软件移植变得很困难。而近几年来,由于 Linux 自身拥有诸多优势,吸引了许多开发者的目光,成为嵌入式操

作系统的新宠。它的出现无疑为我国发展嵌入式操作系统事业提供了一个极有吸引力的机会。

(6) 嵌入式 Linux

Linux 最初是由 Linus Torvalds 编写及发布的、源代码公开、可免费使用的操作系统。后来,又通过 Internet 上成百上千的程序员们的加入,使 Linux 成为一个几乎支持所有主流 32 位 CPU 的操作系统。其特点主要有:内核高效稳定、公开源代码、可移植性、可裁剪、支持多任务等。

嵌入式 Linux 是指对 Linux 经过裁剪小型化后,可固化在存储器或单片机中,应用于特定嵌入式场合的专用 Linux 操作系统。Linux 具备一整套工具链,容易自行建立嵌入式系统的开发环境和交叉运行环境,可以跨越嵌入式系统开发中仿真工具的障碍。传统的嵌入式开发的程序调试和调试工具是用在线仿真器(ICE)实现的。它通过取代目标板的微处理器,给目标程序提供一个完整的仿真环境,完成监视和调试程序,但一般价格比较昂贵,只适合做非常底层的调试。使用嵌入式 Linux,一旦软硬件能够支持正常的串口功能,即使不用仿真器,也可以很好地进行开发和调试。嵌入式 Linux 提供的工具链(tool chain)有:利用 GNU 的 gcc 做编译器,用 gdb、kgdb、xgdb 做调试工具,能够很方便地实现从操作系统到应用软件各个级别的调试。另外,Linux 也符合 IEEE POSIX.1 标准,使应用程序具有较好的可移植性。

1.2 嵌入式计算的特点

嵌入式系统本身就是计算机系统,因此,它设计中所面临的问题有许多是计算机系统设计中面临的共性问题。但由于嵌入式系统是以应用为中心的,通常它并不是独立的,与其所嵌入的设备紧密关联,因此,与通用台式计算机比较而言,它的设计还是具有许多特殊性,归纳起来有以下几点。

(1) 嵌入式系统通常是面向特定应用的。嵌入式系统大多工作在为特定用户群设计的系统中,它通常都具有低功耗、体积小、集成度高等特点,能够把通用台式计算机中许多由板卡完成的任务集成在芯片内部,从而有利于嵌入式系统设计趋于小型化,移动能力大大增强,跟网络的耦合也越来越紧密。

(2) 嵌入式系统是将先进的计算机技术、半导体技术和电子技术与各个行业的具体应用相结合后的产物。这一点就决定了它必然是一个技术密集、资金密集、高度分散、不断创新的知识集成系统。

(3) 嵌入式系统的硬件和软件都必须高效率地设计,根据应用需求量体裁衣、去除冗余,力争在同样的硅片面积或单板面积上实现更高的性能,这样才能在具体应用中更具有竞争力。

(4) 嵌入式系统和具体应用有机地结合在一起,它的升级换代也是和具体产品同步进行,因此嵌入式系统产品一旦进入市场,具有较长的生命周期。

(5) 为了提高执行速度和系统可靠性,嵌入式系统中的软件一般都固化在存储器芯片或单片机本身中,而不是存储于磁盘等载体中。

(6) 嵌入式系统本身不具备自举开发能力,即使设计完成以后用户通常也不能对其中的程序功能进行修改,必须有一套开发工具和环境才能进行开发。

1.2.1 嵌入式系统的要求

嵌入式计算技术所面临的挑战源于基础技术的迅猛发展及用户需求的不断提高。在设计中,系统的功能性对于通用台式计算机系统和嵌入式系统来说都是非常重要的,但是,与通用台式计算机系统的设计相比较,嵌入式系统的设计有其许多特殊的要求,主要体现在以下几方面。

(1) 实时性

许多嵌入式系统需要工作在实时方式下。如果数据或控制信息在某段时限内不能到达,系统将会引起错误。在某些嵌入式系统中,实时性能得不到满足是不能接受的,超过时限会引发危险甚至对生命造成伤害,如高速列车控制器,控制信息超时会引起列车运行故障,甚至翻车。而在另一些嵌入式系统中,超过时限虽然不会引发危险,但也会引发一些不愉快的结果,如打印机在打印时,控制信息的响应时间若超时,就会使打印机发生混乱。

(2) 多速率

许多嵌入式系统不仅有实时性要求,而且还需要同时运行多个实时性任务,即使这些动作有些速度慢,有些速度快,系统必须同时控制这些动作。多媒体应用系统就是多速率行为的典型例子,多媒体数据流的音频和视频部分以不同的速率播放,但是它们必须保持同步。只要音频数据或视频数据不能在有限时间内准备好,就会影响整体效果。

(3) 功耗

在通用台式计算机系统中,功耗已不是一个主要的考虑因素,但在嵌入式系统中,尤其是在用电池供电的嵌入式系统中,这是一个主要考虑的因素。大耗电量直接影响到硬件费用,并影响电源寿命以及带来散热问题。

(4) 低成本

多数情况下,嵌入式系统都希望是低成本的。制造成本由许多因素决定,其中包含硬件成本和软件成本。硬件成本主要决定于所使用的微处理器、所需的内存及相应的外围芯片;软件成本通常难于预测,但一个好的设计方法有利于降低软件成本。

(5) 环境相关性

嵌入式系统不是独立的,而是与其被嵌入的设备紧密相关联的。因此,嵌入式系统设计时,必须考虑模拟量信号、数字量信号及开关量信号的输入输出,系统抗干扰性,环境温度、湿度等。

1.2.2 为什么以微处理器为核心

嵌入式系统是以微处理器为核心的数字系统。而设计一个数字系统可以用很多种方法,如定制逻辑、现场可编程门阵列(FPGA)等,那么,为什么要用微处理器呢?这主要有以下两种原因。

(1) 微处理器是实现数字系统的一种十分便捷、有效的方法;

(2) 微处理器使设计不同价位、不同特性的产品系列变得容易,并容易扩充新功能以满足飞速变化的市场需求。

在数字电路设计方面,人们总是认为,微处理器取指令、译码以及执行指令的开销很大,因此,某功能用微处理器执行应用程序实现,比用定制的逻辑电路实现要慢,其实不然,有两个因素一起作用使基于微处理器的设计更快。首先,微处理器能非常高效地执行程序,现代RISC处理器在大多数情况下可用每个时钟周期执行一条指令,虽然取指令、解释指令必须有开销,但可以通过CPU内部并行、流水处理使这些开销不大。第二,微处理器制造商采用最先进的生产技术和工艺,并投入了相当大的财力和人力来优化微处理器电路,以使他们设计的微处理器具有高性能,能高速运行,同时通过大批量生产以降低成本。而那些想自己设计定制逻辑电路的用户,首先必须掌握新的超大规模集成电路技术,然后才能定制逻辑电路。而嵌入式系统设计团队往往是小规模的,甚至是一个人,他们对超大规模集成电路新技术的了解及电路的优化能力,比微处理器制造商相差甚远,因而建立在过时技术上的定制逻辑电路就不可能存在什么性能优势。

另外,人们通常还认为微处理器的通用性以及需要独立的存储器会使得基于微处理器设计的嵌入式系统体积比基于定制逻辑电路设计的大得多。然而,在许多情况下就所使用的逻辑门电路单元而言,微处理器的尺寸是比较小的。一个设计好的定制逻辑电路不能用于执行其他的功能,但微处理器却不是这样,只需要更换微处理器执行的程序就能让它完成不同的功能。由于现代的嵌入式系统需要复杂的算法和用户界面,因此如果是使用定制逻辑电路,就需要设计多个执行不同任务的逻辑电路,因而,系统的尺寸会变大。

不使用微处理器很少或没有优势,微处理器的优点使它在许多领域内成为首选。微处理器的可编程能力在嵌入式系统开发过程中是最宝贵的,它使程序设计可以与硬件系统的设计分开进行,当一组人员在设计包含微处理器、存储器、输入输出设备等电路板时,另一组人员可以同时编写程序。同样重要的是,可编程能力使厂商可以很容易使自己的产品系列化,在许多情况下,高端产品设计可以在不改变原来硬件的情况下,仅仅通过升级软件来实现,这样可以降低生产成本。即使当硬件必须重新设计时,原有的一些软件也可重用,从而大大节约了时间和开销。

1.2.3 嵌入式系统设计所面临的问题

外部约束是嵌入式系统设计遇到困难的一个重要来源。下面列出了嵌入式系统设计过程中所面临的一些主要问题。

(1) 需要多少硬件

在设计嵌入式系统时不仅需要选择何种CPU,同样需要考虑存储器容量、I/O设备及其他外围电路。在满足系统性能要求的前提下,满足系统经济性要求。即系统硬件太少,将不能达到性能要求;硬件太多,又会使产品变得过于昂贵,并降低了可靠性。

(2) 如何满足实时性

用提高CPU速度的方法来使程序运行的速度加快从而解决实时性问题的方法是不可取的。因为这会使系统的价格上升;同时,仅仅提高CPU的时钟频率有时并不能提高程序执行速度,因为程序执行速度还会受存储器速度的限制。因此应精确设计程序以满足实时性要求。

(3) 如何减少系统的功耗

对于电池供电的嵌入式系统,功耗是一个十分重要的问题。对于非电池供电的嵌入式