第一图书网, tushu007.com

<<8位单片机开发>>

图书基本信息

书名：<<8位单片机开发>>

13位ISBN编号：9787111247722

10位ISBN编号：7111247728

出版时间：2008-9

出版时间：机械工业出版社

作者：何宗彬

页数：265

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

第一图书网, tushu007.com

<<8位单片机开发>>

前言

在后PC时代，单片机发展之迅速超乎想象。
今天的单片机已经是嵌入式领域最为重要的分支之一，其应用领域日趋广泛。
从日常生活用品到航天航空，无处不见其踪影。
在全球范围内，单片机的出货量持续增长，2007年约为90亿片。
在我国，单片机技术人员数以百万计。
在某种程度上，单片机的开发相比于DSF(数字信号处理器)、ARM、FPGA(现场可编程门阵列)等其他嵌入式处理器的开发相对简单。
但事实上，我国的单片机开发者却在实践中遭遇着更多的迷茫，然而令人振奋的是他们也在试图寻找答案。

当前，很多参考书都对众人皆知的知识重复讲述，但在重要的细节与概念上避而不谈，也就不能为开发者拨开迷雾。
作者以自己从事单片机开发的亲身经历，尽量对每一个知识点都能为读者解答疑问。
这些知识有些已经被忽视，而有些已经被许多开发者作为疑问在网络等途径上提出。

您是否适合本书　　首先感谢您能打开本书。

作为众多讲述单片机开发书籍中的一本，我希望《8位单片机开发——技术、技巧与规范》一书能真正地给我的读者带来惊喜与收获。
本书不是可以被膜拜的圣典，更不是高手手册，但它也许可以让我的读者阅读并试验后说：“哦，原来是这样……”。

第一图书网, tushu007.com

<<8位单片机开发>>

内容概要

《8位单片机开发：技术、技巧与规范》介绍了完整的单片机开发流程与系统设计方法；利用Multisim、Proteus等工具软件进行开发前的仿真评估；使用Protel绘制原理图及PcB图，并进行信号完整性分析；使用Keil单片机开发工具进行工程开发与组织；单片机开发中重要的C语言特性；Cx5l高级语言特性以及如何使用RTX51。
高质量的单片机开发首先需要开发者的经验与技巧，但也需要正确的开发规范与流程保证。《8位单片机开发:技术、技巧与规范》也介绍了如何使用工具软件（如使用CVS、PC—L,int等）来保证代码质量，同时在附录中给出了遵循，MISRAC：2004的Cx51编程规范。

第一图书网, tushu007.com

<<8位单片机开发>>

书籍目录

前言第1章 单片机开发基础1.1 单片机基础1.1.1 单片机的发展过程1.1.2 单片机的发展趋势1.1.3 单片机主流产品系列1.2 8051内核结构1.3 8051存储器1.3.1 8051存储器分类1.3.2 典型8051单片机存储器1.3.3 扩展80C51单片机存储器1.3.4.Philips80C5lMx单片机存储器1.3.5 数据存储器1.3.6 程序存储器1.4 单片机系统设计方法1.4.1 单片机系统的生命周期1.4.2 单片机系统开发过程1.5 单片机系统硬件设计1.5.1 硬件设计内容1.5.2 硬件设计要点1.6 本章回顾第2章 电路仿真2.1 Multisim简介2.2 Multisim功能简介2.3 Mulltisim电路仿真实例2.3.1 绘制仿真电路图2.3.2 仿真电路2.3.3 分析电路2.3.4 生成报告2.4 Protel99电路仿真2.5 本章回顾第3章 单片机系统仿真3.1 Proteus简介3.2 ProteusISIS功能简介3.3 ProteusISIS单片机仿真实例3.3.1 绘制仿真原理图3.3.2 程序设计3.3.3 程序调试3.4 其他仿真方式3.4 1Muhisim10单片机仿真3.4.2 KeilixVision3单片机仿真3.5 本章回顾第4章 原理图与PCB设计4.1 Protel99简介4.2 原理图设计4.2.1 原理图设计步骤4.2.2 原理图设计规则4.2.3 Protel99设计环境4.2.4 Protel99原理图设计4.3 PCB设计4.3.1 PCB设计步骤4.3.2.Protel99PCB设计4.4 PCB信号完整性分析4.4.1 信号完整性简介4.4.2 Protel99信号完整性分析工具4.4.3 Protel99信号完整性分析步骤4.5 本章回顾第5章 Keil开发工具5.1 Keil开发工具简介5.2 KeilixVision3IDE5.2.1 使用Keil~~Vision3IDE的开发步骤5.2.2 利用KeiltxVision3IDE进行工程组织5.2.3 KeiltxVision3IDE其他功能5.2.4 KeiltrVision3.IDE使用技巧5.2.5 Keil“ Vision3IDE支持的文件类型5.3 CxSl优化编译器5.3.1 使用e语言开发的优势5.3.2 Cx51控制命令5.4 Ax51宏汇编器5.5 BL51／Lx51链接／重定位器5.5.1 链接／重定位器的作用5.5.2 代码分组5.5.3 数据覆盖5.6 LIBx51库管理器5.7 目标HEX转换器5.7.1 OH5I／OHx5l目标HEX转换器5.7.2 0C51分组目标文件转换器5.8 uVision3调试器5.9 Monitor-515.10 RTX-51实时操作系统5.11 本章回顾第6章 CxSl语言扩展及预处理6.1 存储区。
6.1.1 ROM-6.1.2 片上RAM6.1.3 片外RAM6.1.4 Far存储区6.1.5 SFR存储区6.2 存储模式6.2.1 Small6.2.2 CompactI6.2.3 Large6.3 存储类型6.3.1 明确指定存储类型6.3.2 使用默认存储类型6.4 Cx51扩展的关键字6.5 类型限定词6.5.1 const6.5.2 volatile6.6 数据类型6.7 指针6.7.1 通用指针6.7.2 指定存储区的指针6.8 函数声明6.8.1 参数与堆栈6.8.2 参数与寄存器6.8.3 返回值6.8.4 存储模式6.8.5 寄存器组6.8.6 访问寄存器组6.8.7 qp断函数6.8.8 可重人函数6.8.9 alien（PIJM-51）函数6.8.10 实时任务函数6.9 预处理器6.9.1 预处理命令6.9.2 宏定义6.9.3 文件包含6.9.4 条件编译6.10 本章回顾第7章 重要的C语言特性7.1 跳转辅助手段7.1.1 continue7.1.2 break7.1.3 goto7.2 关jptypedef7.2.1 区别于#define7.2.2 使用typedef7.3 指针7.3.1 基本概念7.3.2 指针运算7.3.3 指针与数组7.3.4 指针与结构体7.3.5 指针与函数7.4 变量的作用域7.5 变量的生存期7.6 关于static7.7 理解复杂的C语言声明7.8 位操作7.8.1 位运算符7.8.2 位操作的作用7.8.3 位段7.8.4 位段与位运算符的比较7.9 本章回顾第8章 Cx5l高级编程8.1 文件定制8.1.1 启动代码8.1.2 变量初始化文件8.1.3 低级流I／O处理文件8.1.4 存储器分配文件8.2 Cx51优化器8.2.1 指定优化等级8.2.2 使用绝对寄存器寻址8.2.3 利用寄存器传递函数参数8.2.4 代码优化等级8.2.5 链接器代码封装8.3 编写适宜的代码8.3.1 选择合适的存储模式8.3.2 显式声明存储类型8.3.3 使用最小长度类型8.3.4 使用无型号数据类型8.3.5 尽量使用局部变量8.3.6 其他优化方式8.4 与汇编程序的接口8.5 与PL／M-51的接口8.6 数据存储格式8.6.1 处理器大小端8.6.2 bit、char、int、long存储格式8.6.3 浮点数存储格式8.6.4 指针存储格式8.7 绝对地址访问8.7.1 使用可访问绝对地址的宏8.7.2 使用_at_关键字8.7.3 使用链接控制命令8.8 8051堆栈8.9 本章回顾第9章 RTX51实时操作系统9.1 RTX51基础9.1.1 RTOS简介9.1.2 前后台系统9.1.3 多任务系统9.1.4 RTX51多任务系统9.2 RTX51基本概念9.2.1 RTX51任务管理9.2.2 RTX51i扣断管理9.2.3 RTX51任务间通信9.2.4 信号9.2.5 邮箱9.2.6 信号量9.2.7 动态存储器管理争2.8 时间管理9.3.RqN51技术参数9.4 RTX51编程参考9.4.1 函数总览9.4.2 系统初始化9.4.3 任务管理9.4.4 扣断管理9.4.5 等待函数9.4.6 信号函数9.4.7 消息函数9.4.8 信号量函数9.4.9 存储器管理9.4.10 系统时钟管理9.5 使用RTX51Tiny9.5.1 要求与限定9.5.2 配置RTX51tiny9.5.3 RTX51Tiny配置原则9.5.4 RTX51Tiny例程9.6 使用RTX51Full9.6.1 配置RTX51Full9.6.2 RTX51FuU例程9.7 本章回顾第10章 编写高质量的Cx51代码10.1 一定需要高质量的Cx51代码吗10.2 有ANSIC就足够了吗10.2.1 K&RC与ANSIC10.2.2 K＆RC与ANSIC的区别10.2.3 不仅仅需要ANSIC10.3 不得不提的MISRA10.3.1 关于MISRAC10.3.2 MISRAC规范的主要内容10.3.3 对MISRAc的思考10.4 利用PC-Lint进行代码检查10.4.1 PC-Lint简介10.4.2 安装与设置PC-Lint10.4.3 集成PC-Lint到特定的工具10.4.4 对PC-Lint的思考10.5 代码质量的保证10.5.1 代码的正确性10.5.2 代码的可靠性10.5.3 代码的可维护性10.5.4 代码的高效性10.5.5 代码

第一图书网, tushu007.com

的可测试性10.5.6 代码的可移植性10.5.7 代码的可扩展性10.6 软件版本控制系统10.6.1 软件版本控制10.6.2 CVS简介10.6.3 CVS与TortoiseCVS的获取及安装10.6.4 CVS操作的基本概念10.6.5 配置CVS服务器端10.6.6 测试CVS服务器端10.6.7 TortoiseCVS客户端操作10.6.8 使用KeiluVision的CVS配置接口10.7 本章回顾附录附录A Cx51编程规范A.1 导言A.1.1 编写目的A.1.2 规范的主要内容A.1.3 缩写和术语A.2 命名规则A.3 文件组织A.3.1 头文件的结构A.3.2 源文件的结构A.3.3 目录结构A.3.4 其他A.4 程序版式A.4.1 空行A.4.2 代码行A.4.3 代码行内的空格A.4.4 对齐A.4.5 长行拆分A.4.6 修饰符的位置A.4.7 注释A.5 编程规则和建议A.5.1 表达式A.5.2 声明和定义A.5.3 整数数据类型及操作A.5.4 字符操作A.5.5 浮点型A.5.6 语句A.5.7 类型转换A.5.8 常量A.5.9 指针A.5.10 数组A.5.11 函数与宏A.5.12 控制语句A.5.13 内存分配A.5.14 错误处理A.5.15 断言A.5.16 标准库A.5.17 通用规则和建议附录B Cx51运行库参考B.1 宏B.2 本征库函数B.3 缓冲区处理库函数B.4 字符类库函数B.5 数据转换类库函数B.6 数学计算类库函数B.7 存储器分配类库函数B.8 I／O流类库函数B.9 字符串库函数B.10 变长参数库函数B.11 其他库函数参考文献

第一图书网, tushu007.com

<<8位单片机开发>>

章节摘录

第1章　单片机开发基础　　1.1　单片机基础　　单片机是计算机与大规模集成电路（Integrated Circuit，IC）技术高速发展的产物，是微型计算机的重要分支之一。

单片机常被称为单片微型计算机（Single Chip Microcomputer，SCM），或微控制器单元（Micro-Controller Unit，MCU）。

在有些领域，它也被称为嵌入式控制器（Embedded-Controller Unit，ECU），甚至直接被称为中央处理器（Central Processing Unit，CPU）。

自20世纪80年代单片微型计算机引入我国以来，“单片机”这一称谓已经被广大的技术开发者与教学工作者所接受，本书也将延用这一称谓。

如果没有特殊说明，本书中所指的单片机均是指8位单片机。

1.1.1　单片机的发展过程单片机是后PC（个人计算机）时代随着电子计算机的发展而产生的。

在20世纪70年代，电子计算机在数字逻辑运算、推理、自动控制等方面显露出非凡的功能后，在工业控制领域也对计算机技术发展提出了新的要求，如面对控制对象、面对各种传感器信号、面对人机交互操作控制能方便地嵌入到工业控制应用系统中等。

此时，单片机应运而生。

纵观单片机的发展历史，它大致经历了SCM、MCU与片上系统（System on Chip，SoC）三大阶段：

1.SCM阶段　　SCM阶段也是单片机的探索与完善阶段。

这一阶段主要是寻求单片机最佳的体系结构并进行完善，“单片机”一词也由此而来。

在这期间，“创新模式”获得了成功，它完全按嵌入式应用要求设计全新的、满足嵌入式应用要求的体系结构、微处理器、指令系统、总线方式、管理模式等。

“创新模式”奠定了SCM与通用计算机完全不同的发展道路。

Intel公司的MCS-48、MCS-51就是按照“创新模式”发展起来的单片嵌入式系统。

……

第一图书网, tushu007.com

<<8位单片机开发>>

编辑推荐

《8位单片机开发：技术、技巧与规范》适合于单片机工程技术人员、有志于学习单片机开发的其他技术人员或高等院校相关专业学生使用。

第一图书网, tushu007.com

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com