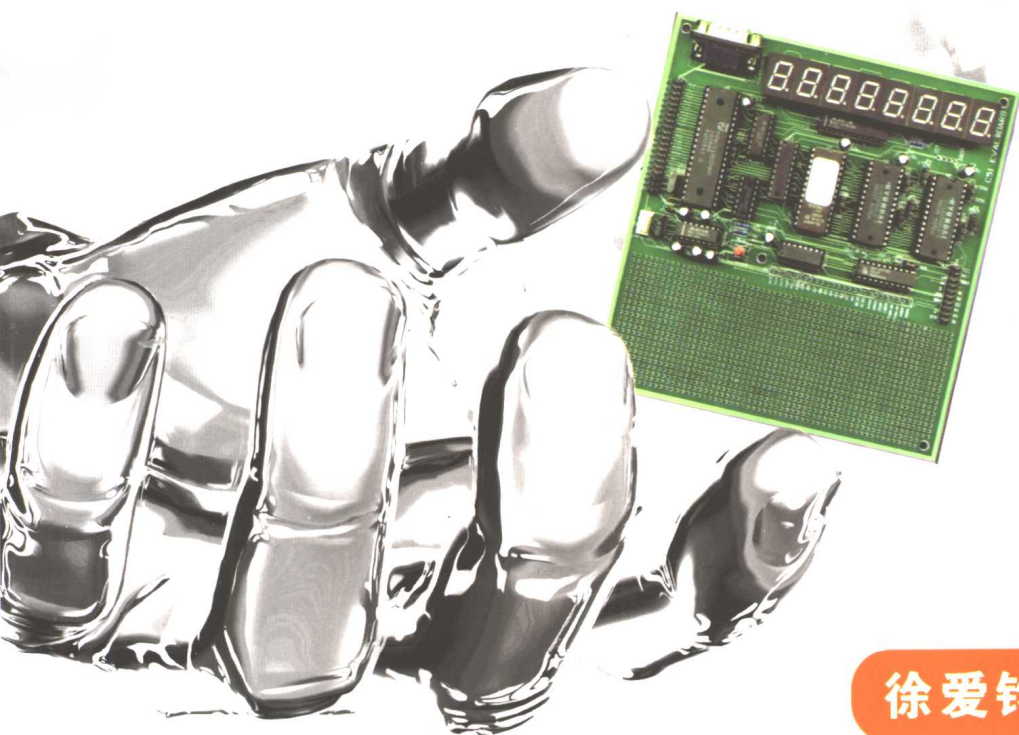


Broadview
WWW.BROADVIEW.COM.CN



徐爱钧 彭秀华 编著

Keil Cx51 V7.0

单片机高级语言编程 与 μ Vision2 应用实践



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

Keil Cx51 V7.0 单片机高级 语言编程与 μ Vision2 应用实践

徐爱钧 彭秀华 编著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书详细介绍了 Keil Cx51 V7.0 版本单片机 C 语言编译器和全新 Windows 集成开发环境 μ Vision2 的强大功能和具体使用方法。全面介绍了最新版本 Cx51 编译器新增加的控制命令, 给出了全部 Cx51 运行库函数及其应用范例, 对 Keil Cx51 软件包中各种应用工具, 如 BL51/Lx51 连接定位器、A51/Ax51 宏汇编器、LIB51 库管理程序以及 OH51 符号转换程序等都作了详细介绍, 还介绍了单片机实时多任务操作系统 RTX51 及其子集 RTX51 TINY 的具体功能与应用方法。 μ Vision2 已经将调试器功能集成于其中, 用户可以在单一环境下完成从源程序编写、编译、链接定位一直到目标文件的仿真调试等全部工作, 书中详细介绍了 μ Vision2 各种功能和应用, 包括软件模拟调试和硬件目标板实时在线仿真。

本书的特点是强调先进性和实用性, 给出了大量程序实例, 并带有一张由 Keil 公司提供的配套 CD-ROM 光盘, 其中包括 Keil Cx51 V7.0 版本全功能评估软件包, 本书各章中列出的全部程序代码, 以及 Keil Cx51 源程序仿真硬件目标板的照片和使用说明。本书适合于从事单片机应用系统开发研制的广大工程技术人员阅读, 也可以作为高等院校相关专业大学生或研究生的教学参考书。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究。

图书在版编目 (CIP) 数据

Keil Cx51 V7.0 单片机高级语言编程与 μ Vision2 应用实践 / 徐爱钧, 彭秀华编著. —北京: 电子工业出版社, 2004.6

ISBN 7-120-00057-8

I. K… II. ①徐… ②彭… III. 单片微型计算机—程序设计 IV. TP368.1

中国版本图书馆 CIP 数据核字 (2004) 第 046879 号

责任编辑: 毕 宁 bn@phei.com.cn

印 刷: 北京智力达印刷有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 45.25 字数: 1077 千字

印 次: 2004 年 6 月第 1 次印刷

印 数: 6000 册 定价: 69.00 元 (含光盘 1 张)

凡购买电子工业出版社的图书, 如有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系。联系电话: (010) 68279077。质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

前 言

8051 单片机是目前国内外工业测量控制领域内使用极为广泛的一类 8 位微控制器,它的特点是使用方便灵活, 外围硬件支持十分丰富, 世界上许多大半导体厂商, 如 Atmel、Analog Device、Dallas、Infineon、Philips、SST、TI 等公司都推出了具有各自特点的增强型 8051 系列单片机, 使用户有了更大的选择范围。另外, 世界上许多软件公司还都致力于 8051 单片机高级语言编译器的开发研究, 使用户得以采用高级语言编程, 从烦琐的汇编语言中解脱了出来。C 语言是一种特别适合于开发计算机操作系统的高级语言, 德国 Keil 公司在开发单片机 C 语言编译器方面取得了相当大的成功, 从编译器、调试器、实时操作系统到集成开发环境, 全面支持 8051、251、166 等单片机主流产品及其众多的派生系列。

本书作者曾以 Keil 公司较早期的 C51 编译器 V5.0 版本为核心编写了《单片机高级语言 C51 Windows 环境编程与应用》一书, 出版后受到广大读者的欢迎, 已经多次重印。V5.0 版的 Keil C51 开发环境实际上是从 DOS 向 Windows 的一种升级, 它在集成度和易用性方面还不尽如人意, 如不支持长文件名、编译选项的设定较为烦琐、与 Windows 环境的兼容性不够完善等。鉴于以上因素 Keil 公司于 1999 年发表了 V6.0 版本的 C51 编译器, 同时推出了一种全新的 Windows 集成开发环境 μ Vision2。最近 Keil 公司又将 C51 编译器升级到 V7.0 版本, 使之能够完全支持 Philips 公司新推出的超大容量内存单片机 80C51Mx, 并命名为 Cx51 编译器, 它是目前最高效的、灵活的 8051 开发平台, 可以支持所有 8051 的衍生产品, 同时支持第三方开发工具。

V7.0 版本的 Cx51 编译器具有如下一些优越性。

1. C51 源程序经过优化编译后生成的代码, 其效率接近于汇编语言生成的代码。
2. 支持所有 8051 系列单片机, 提供对所有外围硬件部件的操作。
3. 无论在有无工作寄存器区转换的情况下, Cx51 都能产生快速中断代码。
4. 支持 Atmel, Dallas, Infineon, Philips 和 Temic 等公司 8051 衍生产品上的双数据指针及高速算术单元 (arithmetic units)。
5. 能够在整个应用程序中执行全局寄存器优化。
6. 所有应用工具均可产生详细警告信息和错误信息, 帮助用户处理难于寻找的问题。
7. 支持再入功能和寄存器区的独立代码, 便于中断服务程序和多任务应用程序的执行。
8. 采用分组方式执行代码分组和调试, 用户程序代码可以轻松突破 64KB 空间的限制。

Keil Cx51 V7.0 版本编译器的 Windows 环境兼容性以及集成度得到极大的提升, Cx51 已被完全集成到一个功能强大的全新集成开发环境 μ Vision2 中, 它将项目 (project) 管理器、Cx51 编译器、Ax51 汇编器、BL51/Lx 连接定位器、RTX51 实时操作系统、Simulator

模拟器以及 Mornitor51 目标调试器的功能全部集成在单一而灵活的 μ Vision2 中,为用户提供了极为简便的操作环境。 μ Vision2 具有强大的项目管理功能,项目中包括源程序文件、开发工具选项以及编程说明等,一个 μ Vision2 项目能够产生一个或多个目标文件。 μ Vision2 内部集成了一个器件数据库(device database),其中储存了各种型号单片机的片上存储器和集成外围功能信息,通过器件数据库可以自动设置 Cx51 编译器、Ax51 汇编器、BL51/Lx51 连接定位器、Debug 调试器等开发工具的功能选项,充分满足用户使用特定单片机的要求。 μ Vision2 还可以为片外存储器产生必要的连接选项,确定应用程序的起始地址及代码大小规模。 μ Vision2 集成开发环境具有如下一些新的功能特点。

1. 简便易用的集成开发环境,源级浏览器(Browser)利用符号数据库使用户可以快速浏览源文件,用详细的符号信息来优化用户变量存储器。

2. 内嵌源程序文本编辑器,允许用户在编辑源程序时(甚至在未经编译和汇编之前)设置程序调试断点,便于在调试程序时快速检查和修改程序。

3. 文件查找功能,能在当前文件或全局文件中搜索指定的文本信息。

4. 集成仿真调试功能,允许在统一的 μ Vision2 环境下进入 Debug 调试器,提供纯软件模拟仿真调试(Simulator)和硬件目标板在线仿真调试(Monitor-51),可以进行应用程序代码综合性能分析和代码覆盖率分析。

5. 单一而灵活的集成开发环境,可以对所有开发工具的功能选项进行必要的设置。同时提供一个可配置的工具菜单,允许在 μ Vision2 环境下启动用户功能。

6. 提供对软件版本控制系统 SVCS 的接口、对应用程序代码进行深层语法分析的 PC-LINT 接口以及对便于生成含有 N-S 流程的块集代码 EasyCase 编辑器的接口。

本书是作者在对最新版本 Keil Cx51 专业开发软件包全面了解和熟练应用的基础上编写而成的,力求先进性和实用性,不仅完整地介绍了 V7.0 版本 Keil Cx51 开发工具的功能和使用方法,还给出了大量具体应用实例,所有程序例子都在 μ Vision2 环境下调试通过。全书共分 12 章,第 1 章阐述了 8051 单片机的存储器结构,以及在 μ Vision2 集成开发环境下进行 C51 程序设计和编译调试的基本方法。第 2 章~第 6 章阐述了 8051 单片机 C 语言编程的基础知识,以及 Keil Cx51 对 ANSI C 的扩展。第 7 章阐述了 μ Vision2 集成开发环境,详细介绍了工作环境、目标程序的仿真调试以及各种开发工具应用选项的设置方法。第 8 章阐述了在 μ Vision2 环境下进行 C51 应用程序设计的技巧,并通过大量具体实例详细讨论了许多 C51 实际应用中的常见问题和解决方法。第 9 章阐述了 V7.0 版本 Keil Cx51 编译器,详细介绍了各种编译控制命令、数据调用协议,给出了全部库函数及其应用范例。第 10 章阐述了 Ax51 宏汇编器,详细介绍了各种符号与表达式、汇编伪指令、宏处理以及汇编控制命令。第 11 章阐述了 BL51/Lx51 连接定位器与实用工具,详细介绍了 BL51/Lx51 的各种连接控制命令、符号转换程序 OH51、库管理程序 LIB51 的使用方法。第 12 章阐述了 RTX51 实时多任务操作系统及其子集 RTX51 TINY 的功能和具体使用方法。

为帮助读者更好地学习掌握 Keil Cx51 进行应用程序设计,本书带有一张由 Keil 公司提供的配套 CD-ROM 光盘,其中包含 Keil 公司最新版本的全功能 Cx51 评估软件包和本书各章所有范例程序代码。另外,我们还设计并制作了一种能与本书介绍的 μ Vision2 环境配套使用的 Keil Cx51 源程序仿真硬件目标板,可以将用户程序下载到评估板中直接进行硬件目标系统仿真调试,节省购买专用仿真器的费用。关于硬件评估板的使用方法书中作

了详细介绍，光盘中还附有 Keil Cx51 源程序仿真硬件目标板的照片。

本书在编写出版过程中得到电子工业出版社的热情支持，并得到丁福坤、燕满、张玉芳、郭邦云、王珍、邓骊、冯金栋、易金生、商杰、袁晓莉、陈召军、李进、帖伟鹏等同志的协助，在此表示衷心感谢。由于作者水平有限，书中难免会有错误和不妥之处，恳请广大读者批评指正，读者可通过电子邮件：ajxu@tom.com，ajxu41@sohu.com 直接与作者联系。

徐爱钧 2004 年 4 月

目 录

第 1 章 8051 单片机与 Keil Cx51 基础	1
1.1 8051 单片机的存储器组织结构	1
1.2 Keil Cx51 开发工具	6
1.3 Cx51 简单编程与调试	6
第 2 章 Cx51 程序设计基础	16
2.1 标识符与关键字	16
2.2 Cx51 程序设计的基本语法	18
2.2.1 数据类型	18
2.2.2 常量	21
2.2.3 变量及其存储模式	22
2.2.4 用 typedef 重新定义数据类型	26
2.2.5 运算符与表达式	27
2.3 Cx51 程序的基本语句	39
2.3.1 表达式语句	39
2.3.2 复合语句	39
2.3.3 条件语句	41
2.3.4 开关语句	43
2.3.5 循环语句	45
2.3.6 返回语句	50
第 3 章 函数	51
3.1 函数的定义	51
3.2 函数的调用	53
3.2.1 函数的调用形式	53
3.2.2 对被调用函数的说明	54
3.2.3 函数的参数和函数的返回值	56
3.2.4 实际参数的传递方式	58
3.3 函数的递归调用与再入函数	58
3.4 中断服务函数与寄存器组定义	60
3.5 函数变量的存储方式	63

3.5.1	局部变量与全局变量	63
3.5.2	变量的存储种类	65
3.5.3	函数的参数和局部变量的存储器模式	72
第 4 章	数组与指针	73
4.1	数组的定义与引用	73
4.2	字符数组	74
4.3	数组作为函数的参数	77
4.4	指针	80
4.4.1	指针与地址	80
4.4.2	指针变量的定义	81
4.4.3	指针变量的引用	87
4.4.4	指针变量作为函数的参数	88
4.5	数组的指针	89
4.5.1	用指针引用数组元素	89
4.5.2	字符数组指针	91
4.5.3	指针的地址计算	92
4.6	函数型指针	94
4.7	返回指针型数据的函数	96
4.8	指针数组与指针型指针	97
4.8.1	指针数组	97
4.8.2	指针型指针	101
4.8.3	抽象型指针	103
第 5 章	结构、联合与枚举	106
5.1	结构变量的定义与引用	106
5.2	结构变量的初值	109
5.3	结构数组	110
5.4	结构型指针	111
5.4.1	结构型指针的概念	111
5.4.2	用结构型指针引用结构元素	111
5.5	结构与函数	112
5.5.1	将结构作为函数的参数	112
5.5.2	将结构型指针作为函数的参数	115
5.6	联合变量的定义与引用	116
5.7	枚举变量的定义与引用	121
第 6 章	预处理器	124
6.1	宏定义	124

6.1.1	不带参数的宏定义	124
6.1.2	带参数的宏定义	126
6.2	文件包含	129
6.3	条件编译	129
6.4	其他预处理命令	132
第 7 章	μVision2 集成开发环境	133
7.1	μVision2 的下拉菜单	135
7.1.1	File 菜单	136
7.1.2	Edit 菜单	137
7.1.3	View 菜单	140
7.1.4	Project 菜单	143
7.1.5	Tools 菜单	154
7.1.6	SVCS 菜单	155
7.1.7	Window 菜单	157
7.1.8	Help 菜单	157
7.2	μVision2 中的调试器	158
7.2.1	Debug 状态下窗口分配与 View 菜单操作	160
7.2.2	通过 Debug 菜单进行程序代码调试	168
7.2.3	通过“Peripherals”菜单观察仿真结果	174
7.3	μVision2 的调试命令	176
7.3.1	显示和更新存储器内容命令	176
7.3.2	程序执行控制命令	181
7.3.3	断点管理命令	184
7.3.4	其他通用命令	187
7.4	μVision2 的表达式	195
7.4.1	表达式的组成	195
7.4.2	μVision2 表达式与 C 语言表达式之间的差别	201
7.4.3	μVision2 表达式应用举例	201
7.5	μVision2 的函数	204
7.5.1	内部函数	204
7.5.2	用户函数	209
7.5.3	信号函数	211
7.5.4	μVision2 函数与 Cx51 函数的差别	213
第 8 章	μVision2 环境下 Cx51 编程技巧与应用实例	214
8.1	编写 Cx51 应用程序的基本原则	214
8.2	Cx51 应用中的一些常见问题与解决方法	216
8.2.1	Cx51 程序设计中容易出错的地方	216

8.2.2	有关 Cx51 程序设计的若干实际应用技巧	218
8.3	8051 单片机的片内定时器应用编程	224
8.3.1	多模块编程	224
8.3.2	片上集成外围功能应用	246
8.3.3	在 μ Vision2 中生成应用库函数	249
8.4	在 μ Vision2 中应用硬件目标板	250
8.5	8051 单片机的片内串行口应用编程	255
8.5.1	用 8051 串行口扩展的矩阵键盘	255
8.5.2	利用 8051 串行口实现多机通信	257
8.5.3	8051 串行口的中断驱动程序	259
8.5.4	利用 8051 串行口实现人机对话命令的输入输出	263
8.6	8051 单片机串行接口扩展应用编程	267
8.6.1	5 位共阴极 LED 驱动器 MC14489 的应用	267
8.6.2	8 位共阴极 LED 驱动器 MAX7219 的应用	271
8.6.3	单总线温度传感器 DS1820 的应用	280
8.6.4	实时时钟芯片 DS1302 的应用	292
8.7	I ² C 总线驱动程序	302
8.7.1	I ² C 总线简介	302
8.7.2	I ² C 总线通用软件模拟驱动程序	305
8.7.3	I ² C 接口器件 24C04 的读写程序	308
8.8	8051 单片机存储器扩展与测试应用编程	313
8.8.1	测试 8051 应用系统总线与扩展存储器的 Cx51 程序	313
8.8.2	8051 扩展 FLASH 存储器在系统编程的 Cx51 程序	316
8.8.3	P89C51RD2 单片机片内 FLASH 在应用中编程的 Cx51 程序	329
8.9	8051 单片机并行接口扩展应用编程	341
8.9.1	打印输出接口及其驱动程序	342
8.9.2	D/A 及 A/D 转换接口及其驱动程序	345
8.9.3	用可编程芯片 8155 实现并行 I/O 接口扩展	353
8.9.4	实时日历/时钟芯片 DS12887 的 Cx51 驱动程序	369
8.9.5	用可编程芯片 8279 扩展键盘/LED 显示器接口	374
8.9.6	LCD 显示模块 EA-D20040AR 的 Cx51 应用编程	380
8.9.7	内置 T6963C 驱动器的 LCD 显示模块 Cx51 应用编程	388
8.10	80C552 单片机内部 A/D 转换器的 Cx51 驱动程序	402
8.10.1	80C552 单片机简介	402
8.10.2	80C552 内部 ADC 的应用	404
8.10.3	使用 80C552 内部 ADC 时印刷电路板的设计要点	408
8.10.4	使用 80C552 内部 ADC 的 Cx51 驱动程序	410
8.11	87C752 单片机在气流量测量中的应用	415
8.11.1	87C752 单片机简介	415

8.11.2	气流量测量仪表的硬件设计	420
8.11.3	气流量测量仪表的软件设计	423
第 9 章	Cx51 编译器	435
9.1	Cx51 编译器简介	435
9.2	Cx51 编译器控制命令详解	438
9.2.1	源控制命令	438
9.2.2	列表控制命令	441
9.2.3	目标控制命令	446
9.3	Keil Cx51 编译器对 ANSI C 的扩展	470
9.3.1	存储器类型与编译模式	470
9.3.2	关于 bit, sbit, sfr, sfr16 数据类型	471
9.3.3	一般指针与基于存储器的指针及其转换	475
9.3.4	Cx51 编译器对 ANSI C 函数定义的扩展	476
9.4	Cx51 编译器的数据调用协议	483
9.4.1	数据在内存中的存储格式	483
9.4.2	目标代码的段管理	485
9.5	Cx51 编译器的特殊支持	487
9.5.1	Analog Device 公司的 Aduc8xxB2 系列	487
9.5.2	Atmel 89x8252 及衍生产品	488
9.5.3	Dallas 80C320、C420、C520、C530	488
9.5.4	Dallas 80C390、C400、C5240 及衍生产品	488
9.5.5	Infineon 80C517、C509、C537 及衍生产品	489
9.5.6	Philips 8xC750、C751、C752	490
9.5.7	Philips 8xC51Mx	491
9.5.8	Philips 和 Atmel WM 系列的双数据指针	491
9.6	用户配置文件	491
9.6.1	启动代码文件	491
9.6.2	变量初始化文件	494
9.6.3	基本 I/O 函数文件	495
9.6.4	分组配置文件	495
9.7	与汇编语言程序的接口	497
9.8	与 PL/M51 程序的接口	507
9.9	绝对地址访问	507
9.9.1	采用扩展关键字 “_at_” 或指针定义变量的绝对地址	507
9.9.2	采用预定义宏指定变量的绝对地址	508
9.9.3	采用连接定位控制命令指定变量的绝对地址	509
9.10	Cx51 的库函数	511
9.10.1	字符函数 CTYPE.H	511

9.10.2	标准 I/O 函数 STDIO.H	517
9.10.3	字符串函数 STRING.H	525
9.10.4	标准函数 STDLIB.H	534
9.10.5	数学函数 MATH.H	540
9.10.6	绝对地址访问 ABSACC.H	546
9.10.7	内部函数 INTRINS.H	548
9.10.8	变量参数表 STDARG.H	550
9.10.9	全程跳转 SETJMP.H	551
9.10.10	计算结构体成员的偏移量 STDDEF.H	552
第 10 章	Ax51 宏汇编器	554
10.1	符号与表达式	556
10.2	汇编伪指令	561
10.2.1	段控制指令	561
10.2.2	符号定义指令	564
10.2.3	存储器初始化指令	567
10.2.4	存储器保留指令	568
10.2.5	过程声明指令 (仅用于 Ax51)	570
10.2.6	程序连接指令	571
10.2.7	地址控制指令	572
10.2.8	其他指令	573
10.3	宏处理器	574
10.3.1	标准宏处理器	574
10.3.2	嵌套宏定义	577
10.3.3	宏调用	578
10.3.4	C 宏处理器	582
10.3.5	MPL 宏处理器	584
10.3.6	MPL 函数	587
10.3.7	MPL 条件处理函数	590
10.3.8	MPL 字符串处理函数	593
10.4	汇编控制命令	595
10.4.1	汇编控制命令详解	596
10.4.2	条件汇编命令	605
第 11 章	BL51/Lx51 连接定位器与实用工具	608
11.1	BL51/Lx51 的连接定位方式	608
11.1.1	连接定位中的数据段处理	608
11.1.2	连接定位器的引用	611
11.1.3	应用程序在存储器空间中的定位	612

11.1.4	数据覆盖	614
11.1.5	代码分组	616
11.1.6	分组配置	618
11.2	连接定位控制命令详解	625
11.2.1	列表文件控制命令	626
11.2.2	输出文件控制命令	629
11.2.3	段与存储器定位控制命令	633
11.2.4	高级语言控制命令	650
11.3	符号转换工具	658
11.3.1	Intel HEX 文件格式与符号转换工具	658
11.3.2	引用符号转换工具 OH51/OHX51	659
11.3.3	引用分组目标文件转换工具 OC51	660
11.4	库管理器 LIBx51	660
第 12 章	RTX51 实时多任务操作系统	664
12.1	RTX51 一般介绍	664
12.2	RTX51 技术参数	669
12.3	使用 RTX51 TINY 的要求和限定	671
12.4	RTX51 TINY 的任务管理	673
12.5	RTX51 TINY 的配置	674
12.6	RTX51 TINY 的系统函数	676
12.7	RTX51 TINY 应用系统调试	681
附录 A	Keil Cx51 与 ANSI C 的差别	696
附录 B	Keil Cx51 不同版本的差别	698
附录 C	代码优化	702
附录 D	Cx51 编译器的限制	705
附录 E	关于配套光盘及硬件目标板	706
参考文献		707

第 1 章 8051 单片机与 Keil Cx51 基础

1.1 8051 单片机的存储器组织结构

8051 单片机最早由 Intel 公司推出，它在一块超大规模集成电路芯片上同时集成了 CPU、ROM、RAM 以及 TIMER/COUNTER，使用者只需外接少量的接口电路就可组成自己的专用微处理器系统。目前，市场上 8051 单片机的硬件支持芯片及软件应用程序十分丰富，除了 Intel 公司之外，还有 Philips、Siemens、ADM、Fujitsu、OKI、ATMEL 等公司都推出了以 8051 为核心的单片机，新一代的 8051 单片机集成度更高，在片内集成了更多的功能部件，如 A/D、PWM、PCA、WDT 以及高速 I/O 口等，在工业测量控制领域内获得极为广泛的应用，因此，有人指出 8051 单片机已成为事实上的工业标准。目前已有多个厂家生产不同型号的 8051 单片机，它们各有特点，但其基本内核相同，指令系统也完全兼容。

图 1.1 所示为 8051 基本内核的结构框图，包括中央处理器 CPU，用于执行各种指令和运算处理；内部数据存储器 RAM，用于存放可以读写的数据；内部程序存储器 ROM，用于存放程序指令或某些常数表格；四个 8 位的并行 I/O 接口 P0、P1、P2 和 P3（每个口都可以用做输入或者输出）；二个定时器/计数器，用做外部事件计数器或内部定时；中断系统具有 5 个中断源（2 个外部中断、2 个定时器中断、1 个串行口中断，采用 2 个优先级的嵌套中断结构，可实现二级中断服务程序嵌套，每一个中断源都可用软件程序规定为高优先级中断或低优先级中断）；一个串行接口电路（用做异步接收发送器）；内部时钟电路（晶体和微调电容需要外接，振荡频率可以高达 40MHz）。以上各部分通过内部总线相连接。在很多情况下，单片机还要和外部设备或外部存储器相连接，连接方式采用三总线（地址、数据、控制）方式，但在 8051 单片机中，没有单独的地址总线 and 数据总线，而是与并行 I/O 口中的 P0 口及 P2 口公用的，进行外部扩展时，P0 口分别作为低 8 位地址线和 8 位数据线，P2 口则作为高 8 位地址线用，所以也是 16 条地址线和 8 条数据线。但是一定要建立一个明确的概念，单片机进行外部扩展的地址线和数据线都不是独立的总线，而是与并行 I/O 口公用的，这是 8051 单片机结构上的一个特点。

对于采用高级语言 Keil Cx51 的用户来说，了解和熟悉 8051 单片机的存储器组织结构是十分必要的，这样在具体编程时可以合理安排各种变量，最大限度实现代码优化。从使用者的角度看，8051 单片机有如下三个存储器空间。

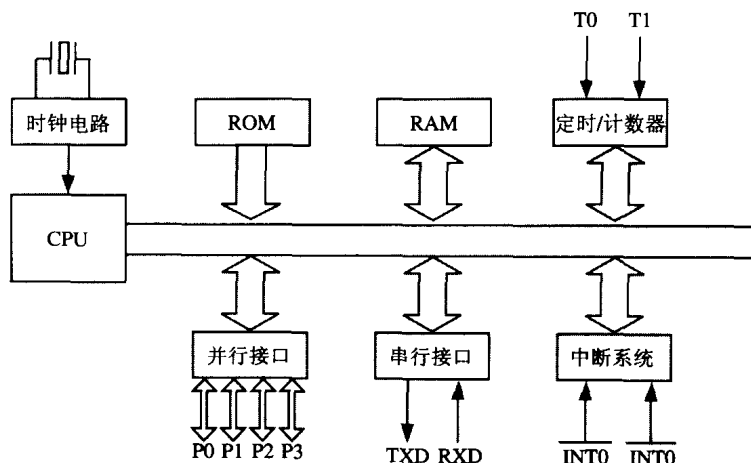


图 1.1 8051 基本内核的结构框图

程序存储器 ROM 对于普通 8051 单片机，程序存储器 ROM 空间大小为 64KB，用于存放程序代码和一些表格常数，称为 CODE 空间。普通 8051 可采用“代码分组”(CODE BANK)设计技术，将 ROM 空间扩展到 $32 \times 64\text{KB}$ ，新型 Philips 80C51Mx 单片机的 ROM 空间最大可扩展到 16MB，称为 ECODE 和 HCONST 空间。8051 单片机专门提供一个引脚“ $\overline{\text{EA}}$ ”来区分片内 ROM 和片外 ROM， $\overline{\text{EA}}$ 引脚接高电平时，单片机从片内 ROM 中读取指令，当指令地址超过片内 ROM 空间范围后，就自动地转向片外 ROM 读取指令； $\overline{\text{EA}}$ 引脚接低电平时，所有的取指操作均对片外 ROM 进行。程序存储器的某些地址单元是保留给系统使用的：0000H~0002H 单元是所有执行程序的入口地址，复位后 CPU 总是从 0000H 地址开始执行程序；0003H~002BH 单元均匀地分为 5 段，用于 5 个中断服务程序的入口，产生某个中断时，将自动进入其对应入口地址开始执行中断服务程序，一些新型 8051 单片机增加了更多的中断源，它们的中断入口地址也相应增加。

片内数据存储器 RAM 对于普通 8051 单片机，片内数据存储器 RAM 空间最大为 256B，用于存放程序执行过程的各种变量及临时数据。片内 RAM 的低 128 个字节可用直接寻址方式进行访问，也可用间接寻址方式访问，称为 DATA 区，其中，00H~1FH 地址范围平均分为 4 组，每组都有 8 个工作寄存器 R0~R7，称为工作寄存器区 (Register Banks)。20H~2FH 地址范围中，每个存储器单元的每一位都可以用位处理指令直接操作，该段地址范围称为位寻址区 (BDATA 区)，其中每一位称为一个 bit。对于 51 子系列单片机仅有上述低 128 个字节，对于 52 子系列单片机，增加了高 128 个字节的片内 RAM，地址范围为 80H~FFH，该范围只能采用间接寻址方式访问，整个片内 RAM 地址范围 00H~FFH 称为 IDATA 区。与 IDATA 空间高 128 个字节（地址范围 80H~FFH）重叠部分称为特殊功能寄存器区 (SFR SPACE)，有些特殊功能寄存器是可以位寻址的，其可寻址位称为 sbit。Philips 公司推出的新型单片机 80C51Mx，其片内 RAM 最大可扩充到 64KB，称为 EDATA 区。

片外数据存储器 RAM 对于普通 8051 单片机，片外数据存储器 RAM 空间大小为 64KB，称为 XDATA 区。在 XDATA 空间内进行分页寻址操作时，称为 PDATA 区。有些

新型 80C51 单片机的扩充片内 RAM，需要用专门的特殊功能寄存器“映像”（MAP）到 XDATA 地址空间；还有一些新型 80C51 单片机可以将片外 RAM 最大扩展到 16MB，称为 HDATA 区。

图 1.2 所示为普通 8051 单片机的存储器组织结构，其中，各部分空间说明及地址范围见表 1-1。

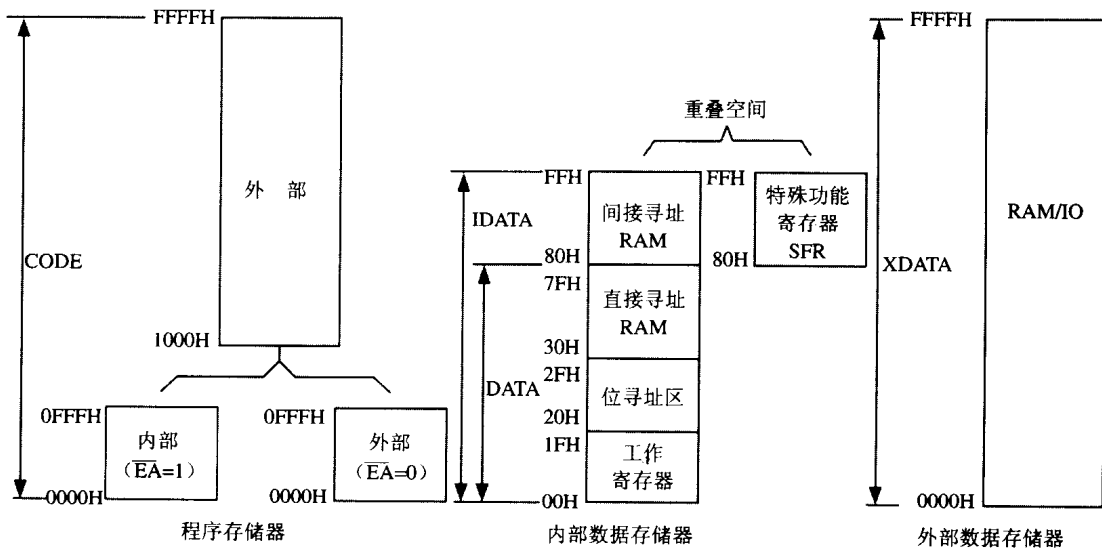


图 1.2 普通 8051 单片机的存储器组织结构

表 1-1 普通 8051 单片机存储器空间分配表

空间名称	地址范围	说明
DATA	D:00H~D:7FH	片内 RAM 直接寻址区
BDATA	D:20H~D:2FH	片内 RAM 位寻址区
IDATA	I:00H~I:FFH	片内 RAM 间接寻址区
XDATA	X:0000H~X:FFFFH	64KB 片外 RAM 数据区
CODE	C:0000H~C:FFFFH	64KB 片内外 ROM 代码区
BANK0~BANK31	B0:0000~B0:FFFFH : B31:0000~B31:FFFFH	分组代码区,最大可扩展 32×64KB ROM

图 1.3 所示为新型 80C51 单片机的扩展存储器组织结构，其中各部分空间说明及地址范围见表 1-2。

图 1.4 所示为 Philips 80C51Mx 单片机的存储器组织结构，其中，各部分空间说明及地址范围见表 1-3。

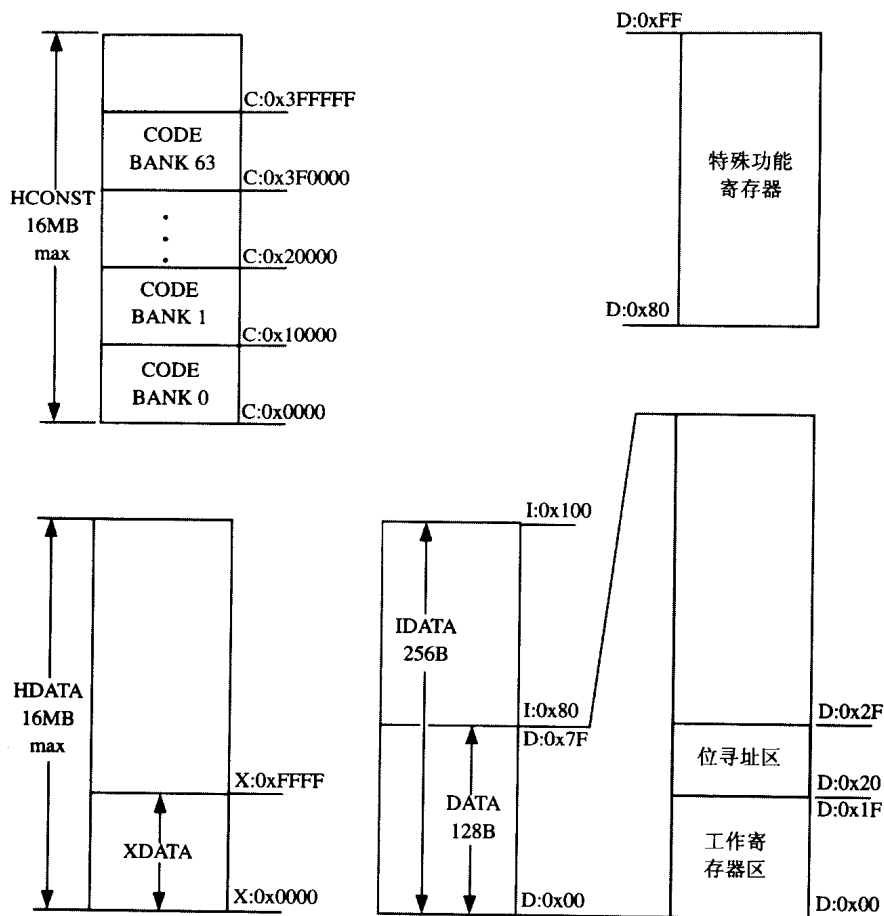


图 1.3 新型 8051 单片机的扩展存储器组织结构

表 1-2 新型 80C51 单片机扩展存储器空间分配表

空间名称	地址范围	说明
DATA	D:00H~D:7FH	片内 RAM 直接寻址区
BDATA	D:20H~D:2FH	片内 RAM 位寻址区
IDATA	I:00H~I:FFH	片内 RAM 间接寻址区
XDATA	X:0000H~X:FFFFH	64KB 常规片外 RAM 数据区
HDATA	X:0000H~X:FFFFFFFH	16MB 扩展片外 RAM 数据区
CODE	C:0000H~C:FFFFH	64KB 常规片内外 ROM 代码区
HCONST (ECODE)	C:0000H~C:FFFFFFFH	16MB 扩展片外 ROM 常数区 (对 Dallas390 可用做代码区)
BANK0~BANK31	B0:0000~B0:FFFFH ... B31:0000~B31:FFFFH	分组代码区, 最大可扩展 32×64KB ROM