

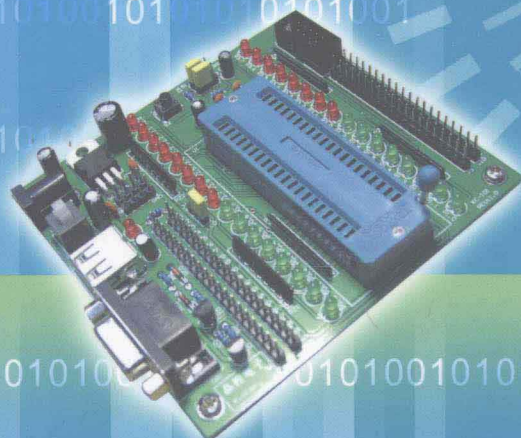


新世纪应用型高等教育
电子信息类课程规划教材

MCS-51 单片机应用实验教程

新世纪应用型高等教育教材编审委员会 组编

主编 陈育斌



大连理工大学出版社



新世纪应用型高等教育
电子信息类课程规划教材

MCS-51

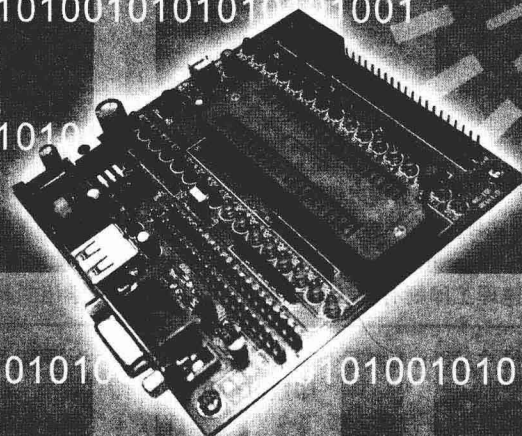
单片机应用实验教程

新世纪应用型高等教育教材编审委员会 组编

主编 陈育斌

副主编 秦晓梅

0101001010101010100101010010101010101001
0101001010101
01010010101010101001
0101001010101010100101010
010100101010101



01010 01010010101001010101010100
010100101010
0101001010101010100
010100101010100101010010101010100
0101001010101010

大连理工大学出版社

图书在版编目(CIP)数据

MCS-51 单片机应用实验教程 / 陈育斌主编. — 大连 :
大连理工大学出版社, 2011. 3

新世纪应用型高等教育电子信息类课程规划教材

ISBN 978-7-5611-6055-8

I. ①M… II. ①陈… III. ①单片微型计算机—高等学校—教材 IV. TP368.1

中国版本图书馆 CIP 数据核字(2011)第 023848 号

大连理工大学出版社出版

地址:大连市软件园路 80 号 邮政编码:116023

发行:0411-84708842 邮购:0411-84703636 传真:0411-84701466

E-mail: dulp@dulp.cn URL: <http://www.dulp.cn>

大连理工印刷有限公司印刷 大连理工大学出版社发行

幅面尺寸:185mm×260mm 印张:15.75 字数:364 千字
印数:1~2500

2011 年 3 月第 1 版 2011 年 3 月第 1 次印刷

责任编辑:潘弘喆

责任校对:章婉霞

封面设计:张莹

ISBN 978-7-5611-6055-8

定 价:31.00 元

前 言

单片机应用技术是高等学校理工科电类学生必须掌握的课程,实践又是学习、掌握单片机应用技术的重要环节。在积累多年教学经验的基础上,我们将理论课、基础实验和综合设计三个环节有机结合起来,建立了以培养学生独立、创新能力为宗旨的单片机综合设计课程,探索一种实践性教学的新模式,并为此编写了本教程。本教程适用于高等学校高年级本科生的综合设计性实验教学,也适用于电类专业的技术人员自学参考。

本教程所使用的实验硬件平台是 DP-51PROC 单片机综合仿真实验台。调试软件采用 KeilC51 μ Vision2(简称 Keil),该软件是基于 80C51 内核的单片机开发平台,内嵌多种标准的开发工具,是目前被广泛使用的调试工具之一。本教程对 Keil 软件的使用给出简单而实用的描述。

引入新型模块及串行接口技术是当前单片机应用领域的发展趋势,也是本教程所描述的重点,如具有同步串行接口 SPI 的 ADC 和 DAC 模块编程原理、I²C 总线协议及接口器件编程、单总线接口原理、128 $\times$ 64 LCD 点阵显示编程等,这些内容的引入使教材更具实用性。

本教程虽以实验内容为主,但每一章节包含相关模块基本原理介绍、相关 SFR 初始化方法、实验台上的应用电路及编程思想描述。读者可参考实验台上的实验电路及对应程序进一步加深对内容的理解和掌握。当然,读者可按教程内容在自己设计的 PCB 板上进行实践。

教程中每一实验都分为验证性实验和思考性编程两部分,前者给出实验程序以便读者学习,后者则需要读者在验证性实验基础上进行适当修改,独立编写所要求的程序,这种方法有利于读者更好地掌握和灵活运用单片机相关知识。

这里我们应提醒每位读者:(1)实践是学习单片机知识的重要环节,只有通过不断编程、调试和查找错误才能提高自己;(2)在软件调试过程中应强调调试方法,利用单步、断点和观察变量等方法查找错误,实现程序的“透明化”调试;(3)初学者编

程中常出现的错误有两种:语法错误和逻辑错误。语法错误可通过编译器自动查找、指出出错的指令行和错误类型;逻辑错误只能靠编程者自己解决。避免和减少逻辑错误最有效的方法是养成先画流程图,再根据流程图编写代码的好习惯。如出现逻辑错误可在程序关键部位设置断点,程序运行到断点时,通过观察变量查找错误,修改设计;(4)对于比较大或比较复杂的程序,应采用模块化设计方法。在结构上主程序尽可能简单,其他所有操作都以“子程序”或“子函数”的方式调用。

本教程的内容分以下几个部分:第一章对实验系统的硬件平台和运行模式进行描述;第二章从实用角度介绍 KeilC51 的使用步骤及方法;第三章、第四章分别介绍了 DP-51PROC 实验系统在线调试和脱机运行两种操作模式;第五章介绍 MCS-51 系列单片机的主要特征、最小系统的概念及组成;第六章分别以 51 单片机内部各功能模块为基础,描述片内基本模块的组成、初始化方法以及对应的编程实验。本章节后半部分介绍了 SPI 接口的外围芯片编程实验实例;第七章、第八章以 I²C 总线为重点,介绍总线的结构、通讯协议、工作时序与模拟编程以及三种具有 I²C 总线接口的外围芯片编程实验;第九章介绍了 128×64LCD 点阵式液晶显示模块编程与应用;第十章介绍单总线智能传感器(18B20)的接口编程实验等;第十一章介绍在 DP-51PROC 硬件平台进行各项综合类设计的设计题目。

最后以附录形式提供了 I²C 总线的通讯协议子程序库(汇编和 C 语言)、MCS-51 单片机的指令系统表,这些资料可以为读者在不具备 I²C 总线接口的 MCS-51PROC 单片机编程时提供参考。

教程的基础实验中所给出的参考程序以汇编语言为主。从汇编语言入门是比较科学的方法。汇编语言与机器语言一一对应,采用汇编指令编程可准确掌握单片机的内部结构与工作过程。汇编语言实时性好、代码利用率高,特别是在一些要求实时性较强的场合其特点更明显。当然,从工程角度出发,用 C 语言编程具有编程速度快、算法丰富、可移植性好和实用性强等特点,因此教程中也给出与汇编语言相对应的 C 语言参考程序,只要具备 C 语言基础就可方便地掌握 C51 编程方法。我们提倡在熟练掌握汇编语言的基础上再用 C 语言编程或将两者结合起来,这样效果更好。

单片机是硬件工程师必须掌握的专业工具。如何在较短时间内学习、掌握单片机知识一直是读者关心的问题。总结多年教学、科研经验,我们认为学习内容分两部分:一是单片机本身的内容,包括硬件模块、汇编指令、C51 语言编程;另一方面是接口设计,这需要读者尽量多地了解新型接口器件、多实践、多编程。从另一角度讲,单片机学习是无止境的,随着新器件、新知识的不断出现,我们必须不断学习才能适应发展、提高自身水平。

对于无实验室条件的读者,建议自己构造一个单片机最小系统,添置必要设备(仿真器、编程器等),应当提醒读者:在选购、使用仿真器前一定要先了解它所支持的运行软件、语言以及仿真器对各端口、引脚使用的限制(端口的开放程度)等。编程器是用于将调试好的目标文件(.HEX)烧写到单片机内部 ROM 的工具。当然,读者也可以选择廉

价的“在线编程”小模块实现程序的调试、下载和在线编程。只要具备程序调试环境,成功只是时间问题。

本书由陈育斌任主编,秦晓梅任副主编,具体分工如下:第一章、第二章由贾凤英编写,第三章、第四章、第五章由秦晓梅编写,第六章、第七章、第八章、第九章、第十章和第十一章由陈育斌编写。全书由陈育斌总体规划和统稿。

本教程经过多个学期的试用,给出的实验参考程序在 DP-51PROC 综合实验台上均已调试通过。由于水平有限,编写中如有错误敬请读者谅解。

所有意见和建议请发往: dutpbk@163.com

欢迎访问我们的网站: <http://www.dutpgz.cn>

联系电话:0411-84707492 84706104

编 者

2011年3月



第 1 章 单片机实验系统简介	1
1.1 实验系统的构成	1
1.2 实验系统的三种运行模式	5
第 2 章 KeilC51 使用简介	6
2.1 创建一个 KeilC51 调试环境	6
2.2 新建一个工程项目	7
2.3 程序文件的编译和链接	12
第 3 章 在线调试仿真功能	16
3.1 在线调试仿真功能的特点	16
3.2 进入调试状态	16
3.3 调试前的准备工作	18
3.4 实例应用	21
第 4 章 脱机 Flash 运行模式	28
4.1 脱机 Flash 运行模式的特点	28
4.2 脱机 Flash 运行模式的存储器配置	29
4.3 进入脱机 Flash 运行状态	29
第 5 章 MCS-51 的基本结构及最小系统	31
5.1 MCS-51 单片机内部基本结构及特点	31
5.2 MCS-51 单片机常用型号及规格	37
5.3 MCS-51 单片机的最小系统	38
第 6 章 MCS-51 基础知识与实验	40
6.1 MCS-51 单片机的存储器读写实验	40
6.2 MCS-51 单片机的并行输入输出端口实验	42
6.3 MCS-51 单片机中断系统及外部中断/INT0 实验	50
6.4 定时/计数器实验	58
6.5 串行接口 SBUF 实验	85
6.6 MCS-51 与 TLC549 接口芯片编程实验	93
6.7 MCS-51 与 TLC5620 编程实验	100
第 7 章 单片机模拟编程	108
7.1 单片机的同步串行接口及标准	108
7.2 I ² C 总线的主要特点及结构	110

7.3	I ² C 总线的工作过程与原理	111
7.4	I ² C 总线的信号时序	113
7.5	I ² C 总线的时钟同步与总线仲裁	113
7.6	I ² C 总线的工作时序与 AT89C51 单片机的模拟编程	115
7.7	芯片内部的单元寻址	119
7.8	I ² C 通讯子程序/子函数	124
第 8 章	I²C 外围器件编程	125
8.1	EEPROM 芯片原理及实验	125
8.2	ZLG7290B 结构、原理及实验	132
8.3	PCF8563T 低功耗时钟芯片简介	160
第 9 章	ZY12864D 液晶模块编程	175
9.1	ZY12864D 液晶模块内部结构框图	176
9.2	ZY12864D 液晶模块外部引脚定义	176
9.3	以 ZY12864D 为核心的显示系统接口框图	177
9.4	ZY12864D 显示模块的工作时序	178
9.5	ZY12864D 显示模块的命令	179
9.6	ZY12864D 显示系统与单片机的接口	182
9.7	LCD 模块编程	184
第 10 章	DS18B20 智能温度传感器编程	203
10.1	DS18B20 元件介绍	203
10.2	单总线系统的通信协议	204
10.3	DS18B20 内部存储器结构	206
10.4	DS18B20 的操作流程及指令说明	208
10.5	单点 DS18B20 温度采集编程实验	210
第 11 章	单片机综合设计选题	221
附 录		225
附录 1	由汇编语言编写的 I ² C 通讯子程序	225
附录 2	由 C 语言编写的 I ² C 通讯子函数	230
附录 3	MCS-51 单片机指令系统一览表	234
附录 4	DP-51PROC 单片机综合实验台模块资源一览表	238
附录 5	综合设计报告书样板示例	240
参考文献		244

第 1 章

单片机实验系统简介



知识导入

学习单片机知识离不开单片机实验系统,通过单片机实验系统将书本上的理论知识付诸实践。因此,单片机实验系统是学习、掌握单片机知识的一个重要工具。

1.1 实验系统的构成

单片机实验系统由硬件调试平台和调试软件两大部分构成。

1.1.1 硬件调试平台

单片机实验系统的硬件调试平台由以下三个部分构成:

(1) 计算机系统

计算机系统也称“上位机”,计算机系统对机器资源的最低要求如下:

- 具备 COM 口(9 针 RS-232 串行通信接口)的台式机或便携式笔记本电脑;
- 具备 Pentium、Pentium- II 或兼容的微处理器;
- Windows 95、Windows 98、Windows 2000 或 Windows XP 操作系统;
- 至少 16 MB 的 RAM 空间;
- 至少 20 MB 的硬盘空间。

(2) DP-51PROC 单片机综合仿真实验台

该实验台是可以实现单片机各种外围接口实验的硬件调试平台(如图 1-1(a)所示)。实验台共有约 30 个外围模块,可以实现并行接口、各种同步串行接口的编程实验。实验台设有 128×64LCD 点阵显示、I²C 接口、一线制(温度传感器)接口和 SPI 等多种接口芯片的编程实验,同时还具备直流电机、步进电机、继电器、语音芯片、接触式和非接触式 IC 卡、USB 接口等实验模块。实验台通过 40 脚插座(U13)与仿真器连接,实现对程序各种不同模式的调试和运行。功能模块的详细描述见教程后面的附录 4。

(3) TKSMonitor 51 仿真器

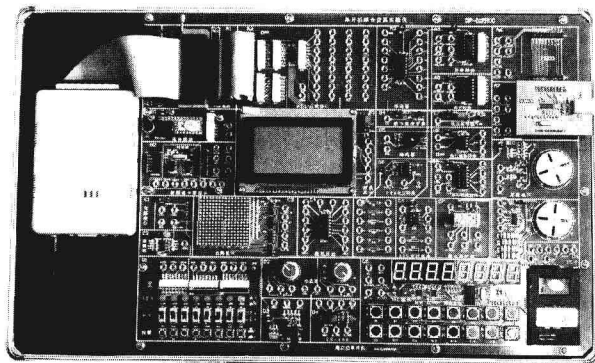
下面介绍基于 Keil 环境下的 TKSMonitor 51 仿真器(参见图 1-1(b)所示)。

仿真器单独供电(+9 V),仿真器内部由一个三端稳压器 7805 为其内部提供+9 V 电源。仿真器内部由一个可在线编程的单片机 P87C52X2BN 和一个 64 K 的 SRAM(静态

存储器)构成。与 DP-51PROC 实验台(目标板)的连接、运行,可以按两种工作方式操作:

①DP-51PROC 的在线调试、运行模式

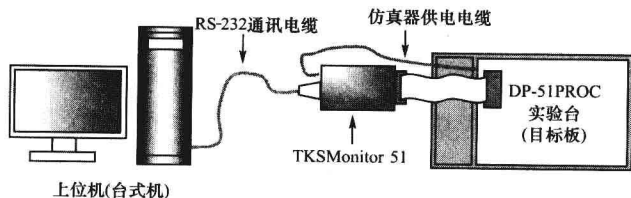
在这种模式下,上位机、仿真器和 DP-51PROC 实验台连接在一起(如图 1-1(c)所示),在这种工作模式中,TKSMonitor 51 仿真器上行与上位机 COM 口连接(采用 RS-232 信号标准),下行由 40 线的扁平电缆与 DP-51PROC 实验台上具有锁定功能的 40 脚 U13 插座连接。其电源由 DP-51PROC 实验台提供。在 Keil 软件的环境下实现对实验台上相应模块程序的动态调试、运行。这种模式,上位机与 DP-51PROC 实验台之间的信息交换是通过 TKSMonitor 51 仿真器中的监控程序 MON51 进行协调工作的。当初次使用 TKSMonitor 51 仿真器时,先要通过微机系统运行一个 DPFlash 程序(仿真器厂家提供),将监控程序 MON51 下载到 TKSMonitor 51 的 Flash 中(地址从 0000H 开始)。这样,在上位机的 Keil 软件环境下,通过调用 TKSMonitor 51 仿真器的 Flash 中监控程序 MON51 实现对目标模块的调试操作。在这个过程中可以运用各种方法(单步、断点或连续运行),通过对各种变量(寄存器、内存单元)的监控完成程序的调试工作。



(a)DP-51PROC 单片机综合仿真实验台



(b)TKSMonitor 51 仿真器



(c)DP-51PROC 的在线调试、运行模式示意图

图 1-1 DP-51PROC 和 TKSMonitor 51 介绍

在这种工作方式中,仿真器中的存储资源如图 1-2 所示,应当注意的是,用户的目标程序起始地址不是 0000H,而应临时改为 8000H,数据区从 C000H 开始对应。中断矢量的位置也要做相应的调整:INT0 中断的矢量单元 0003H 要改为 8003H(其他的矢量入口类同),这是因为在 TKSMonitor 51 仿真器的 Flash 中,0000H~7FFFH 的空间已经被 MON51 监控程序所占用。所以,当实验系统处于在线调试、运行模式时,用户的程序代码中的起始地址都要由原来的 0000H,修改为 8000H,当整个程序调试成功后,在需要真正下载到单片机(或采用脱机 Flash 模式)时,将用户的目标程序的起始地址(包括中断向量等)恢复到从 0000H 开始对应的位置上。

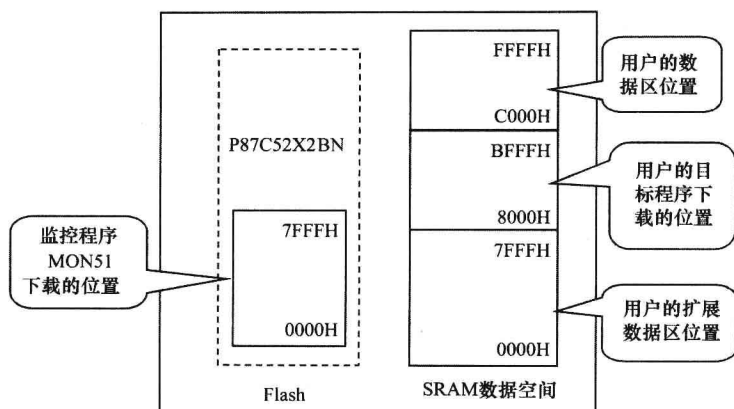


图 1-2 在线调试模式下仿真器内部存储器资源分配示意图

应当说明的是,监控程序 MON51 是被下载到仿真器内部的 Flash 中,所以,一旦下载成功后监控程序会一直保留(掉电不丢失),除非需要刷新 Flash 中的程序;而用户程序会因为存储在 SRAM 中,一旦掉电,便会丢失。

②DP-51PROC 的脱机下载 Flash 模式

这种工作模式是在已经通过上述 DP-51PROC 在线调试、运行模式操作后,程序的功能正常时,对用户程序的一种“验证性”运行方式。

在这种工作方式中,首先利用上位机的 DPFlash 程序将用户的目标程序直接装载到 TKSMonitor 51 仿真器中的 Flash 中,取代仿真器原来的监控程序 MON51 而实现对目标板的控制。此时,将仿真器与上位机脱离,只与目标系统相连接(如图 1-3 所示)。

注意 用户程序是下载到仿真器 Flash 中的 0000H~7FFFH(32 K)单元,所以用户的目标程序起始地址和中断矢量必须恢复到程序只读存储器的以 0000H 开始的单元(如图 1-4 所示)。

在物理位置上,脱机 Flash 运行实际上是直接借用了仿真器中单片机和存储器的资源来临时取代实验台(目标板)上的单片机,完成程序的运行。由 TKSMonitor 51 仿真器和实验台(目标板)临时构成一个用户产品的样机,这也是单片机开发的最后一个验证调试过程。

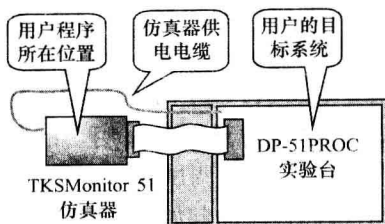


图 1-3 脱机下载 Flash 模式示意图

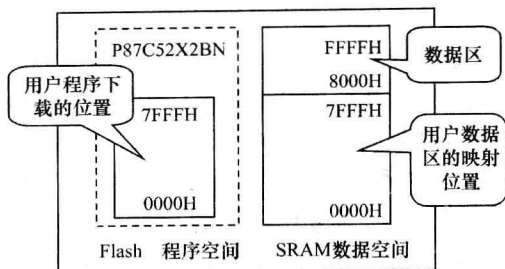


图 1-4 脱机模式存储器资源分配示意图

注意仿真器在调试过程中对于目标系统的限制:对于单片机的 I/O 端口引脚 P0、P2 和 P_{3.0}、P_{3.1}、P_{3.6}、P_{3.7} 已被系统占用,内部为 11.0592 MHz 的晶体做系统时钟。在线调试时目标板上不能接有低于 11.0592 MHz 的晶体,如果目标板上的晶体等于或高于 11.0592 MHz,实际频率按仿真器内部的频率运行(当然最好目标板上暂时不接晶体)。如果用户要使用 P_{3.0}、P_{3.1},只能采用脱机模式,但无法进行在线调试。

1.1.2 调试软件

与实验系统相关的调试软件有两个:KeilC51、DPFlash,这里着重介绍 KeilC51 的性能和使用方法,DPFlash 软件将在第 4 章中作以介绍。

KeilC51 μ Vision2(简称 Keil)集成开发软件是基于 80C51 内核的单片机开发平台,内嵌多种标准的开发工具,可以实现从工程建立、编译、链接、目标代码的生成、软件仿真、硬件的在线调试等完整的开发过程。其中,内嵌的 C 编译器是目前在生成代码的准确性和效率方面都是处于较高水平的。系统支持汇编和 C 语言编程,可了解 Keil 的启动画面(如图 1-5 所示)。

Keil 软件的运行方式有两种,模拟仿真(不需要仿真器,程序从 ROM 的 0000H 单元开始)和在线调试(借助于仿真器调试运行)。可以通过 Keil 程序菜单栏中的“Project”项,在其下拉菜单中选择“Option for Target ‘Target 1’”,在 Target 界面下选择 Debug(如图 1-6 所示),其中“Use Simulator”为纯软件模拟仿真模式(也是系统运行的默认模式)，“Use”为在线调试模式。

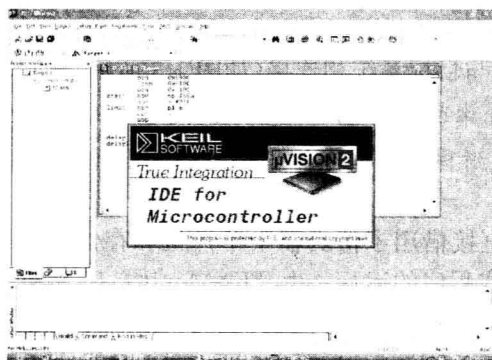


图 1-5 Keil 的启动画面

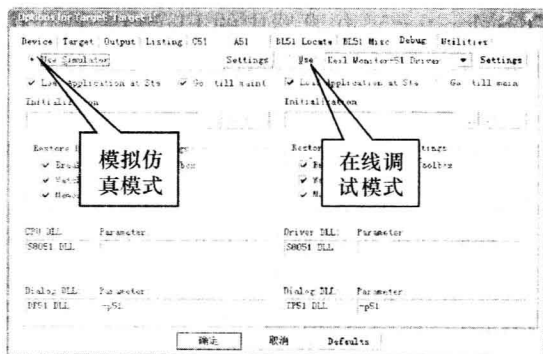


图 1-6 Target 下的两种运行方式选择

1.2 实验系统的三种运行模式

实验系统具有不同的三种运行模式。按照其开发顺序与过程分别介绍模拟仿真(Use Simulator)模式、在线调试(Use)模式以及脱机运行模式。

(1) 模拟仿真(Use Simulator)模式

在这种模式下利用上位机运行 Keil 软件,实现用户程序的编辑、编译(包含语法检查),不需要仿真器和目标系统(实验台),因此,这种方式适合于工程设计的准备阶段,由于它不需要仿真器和硬件目标平台环境的支持,因此,可以在任何一个具备 Windows 操作系统的微机系统上运行 Keil 软件,以实现对工程的建立、管理,源程序的编辑和编译,通过对源程序的编译完成对程序的语法检查。

(2) 在线调试(Use)模式

在这种模式下,上位机(具有 COM 口的台式机或笔记本电脑)、仿真器(TKSMonitor 51)、目标系统(DP-51PROC 实验台)构成一个整体(如图 1-1(c)所示)。在程序的调试过程中,Keil 软件不仅可以完成对目标程序的编辑、编译(语法检查)等,还可以实现用户程序在目标系统上的运行,而且可以利用“单步”、“断点”和“观察变量”等方法 and 手段对程序进行跟踪调试,在上位机的屏幕上将程序的各种状态、变量数据进行显示,使程序运行的整个过程“透明化”。

应当注意,与模拟仿真模式不同,在线调试模式下,使用者必须按照仿真器的要求来设定调试环境的参数,否则仿真器无法正常进行工作,调试环境参数如用户程序的起始地址、数据区的地址等。使用 TKSMonitor 51 仿真器进行在线调试时,用户的程序是下载到仿真器中 SRAM 存储器 8000H 开始的单元,而不是 0000H,所以要对用户程序相关的 ORG 伪指令进行修改(用户原有的程序起始地址应临时修改为 8000H),而仿真器中的监控程序(MON51)占据着 0000H 开始的单元(如图 1-2 所示)。

在线调试模式下,用户的目标程序是在监控程序的控制下运行,并将程序的各种参数传至上位机的屏幕上显示,为编程者提供运行、调试的信息。

当用户的程序调试工作结束,并要将最终程序代码烧写到用户单片机芯片之前,不要忘记将调试时的程序入口地址恢复到正常值(ROM 的 0000H 开始的单元),并重新编译,将编译后的十六进制文件(.HEX)烧写到用户单片机芯片中。

在线调试的系统连接见图 1-1(c) (DP-51PROC 的在线调试、运行模式示意图)。

(3) 脱机运行模式

此模式脱离上位机,由仿真器和目标系统构成(参见图 1-3)。由于脱离了上位机,系统非常方便被带到应用现场进行程序功能的进一步验证。因为是在线调试模式中调试通过的程序代码文件(.HEX)直接下载到仿真器的 Flash 中从 0000H 开始的单元(取代原有的监控程序 MON51),所以用户程序的起始地址和中断向量应恢复为 0000H 单元开始的地址。工作时,仿真器全速运行 Flash 中的用户程序,即由仿真器临时取代目标板上的单片机,通过这种方法来进一步验证程序的运行状态。脱机 Flash 模式的系统结构参见图 1-3(脱机下载 Flash 模式示意图)。

第 2 章

KeilC51 使用简介



知识导入

Keil 调试软件具有非常强大的调试功能,这里通过使用 Keil 软件对一个开发过程做一个较详细的描述,以便于读者快速了解、掌握 Keil 软件的特点和使用步骤,建立正确使用 KeilC51 集成调试软件的基本概念。

2.1 创建一个 KeilC51 调试环境

KeilC51 集成开发软件是采用“工程”方法而不是使用单一程序文件的形式来管理文件的。所有的文件(包括源程序,如 C 语言或汇编语言程序、头文件甚至说明性的技术文件)都是包含在一个“工程项目”文件里进行统一管理。

使用“工程”方法来调试程序可以简化操作、方便调试,如在初次建立工程时所设置的调试参数在工程管理上是自动保存的,在后续的操作中节省了重新建立调试参数的麻烦,只要直接打开以前建立的工程(注意,不是直接打开用户文件)就可以直接进行调试了。使用 KeilC51 的读者应当适应和习惯这种文件的管理方法。

在调试一个程序的时候,往往要设置相关的调试参数(详见后续部分)。当编程者需要临时退出调试时,工程管理器会将对应的参数一同保存下来,当编程者重新调试程序时不是打开程序文件,而是直接打开该工程,这样打开的工程中依然会保留上次调试环境的各项参数,给调试者带来了很大的方便,这一特点需要读者注意。

注意

- 一个用户程序要单独由一个工程项目来管理(单独的一个工程名);
- 每一个工程要单独占用一个文件夹;
- 工程所在的文件夹,建议不要使用中文命名并避免长字符做文件名。

使用 KeilC51 集成调试软件来建立自己的一个工程项目要经过如下几个步骤:

- (1) 建立一个工程项目;
- (2) 为工程选择一个目标器件(如选择 AT89C51);
- (3) 为工程项目设定相关的软件和硬件的调试环境(如纯软件仿真或在线调试等);

- (4)创建源程序文件并输入、编辑源程序代码(汇编格式或 C 语言格式);
- (5)保存所创建的源程序项目文件;
- (6)把源程序文件添加到项目中(同时指明程序文件的格式:汇编格式或 C51 格式),具体方法详见后续内容。

2.2 新建一个工程项目

2.2.1 运行 μ Vision2 软件

双击桌面上的 KeilC51 快捷图标,运行 KeilC51 调试软件。注意,不同情况下打开 KeilC51 程序时的界面往往是不同的,一般总是启动该软件前一次所处理的工程(如图 2-1所示)。在这种情况下可以选择工具栏中的“Project”选项中的“Close Project”命令,关闭该工程(如图 2-2 所示)。

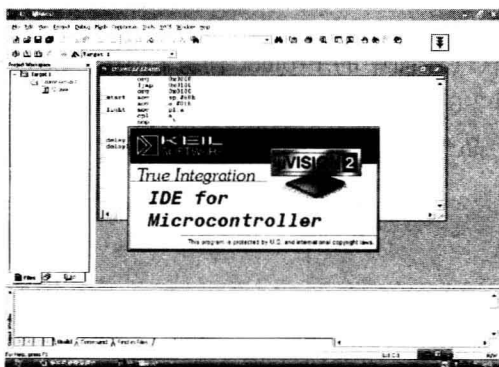


图 2-1 KeilC51 运行时打开前一次的工程

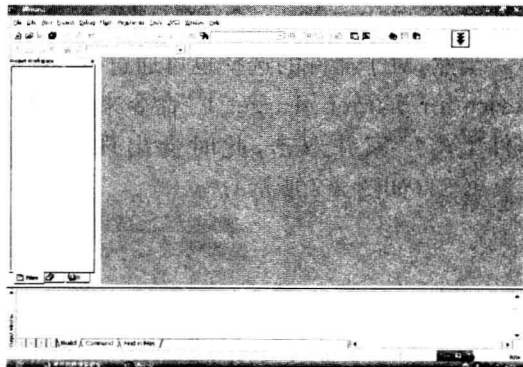


图 2-2 利用“Close Project”命令关闭工程

2.2.2 建立新工程

选择工具栏中的“Project”选项中的“New Project”命令,建立一个新的 μ Vision2 工程(如图 2-3 所示)。

这里需要完成下列操作:

- (1)为工程起一个名字,具有一定的意义(不要使用中文名),不要太长(保存类型可按默认处理);
- (2)选择本工程所存放的路径,注意在某一个盘符中为该工程单独建立一个文件夹(不要使用中文名),以便于将本工程所需的所有文件都保存在该文件夹中且不与其它工程的文件相混淆。

【举例】 选择工程目录为:D:\51\Led_Light,输入项目名称:Led_Light 后单击“保存”按钮后返回(如图 2-4 所示)。

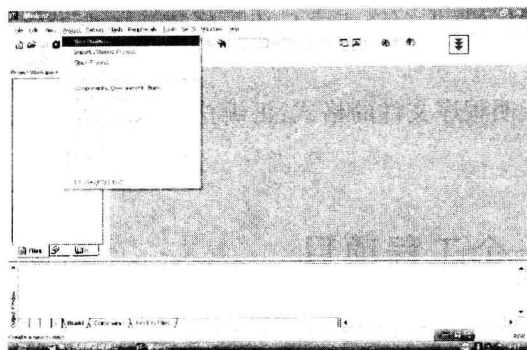


图 2-3 利用“New Project”命令建立新工程

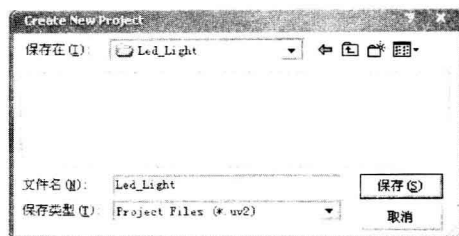


图 2-4 新建并选择合适的路径保存该工程

2.2.3 为工程选择目标器件

当完成保存工程项目后,会弹出器件选择窗口(如图 2-5 所示),该操作是为工程指明最终所使用的单片机型号。尽管都是 51 系列单片机,但不同厂家、不同型号的单片机其内部硬件资源、相关硬件参数是不同的,μVision2 软件会根据所选择的器件来调用、管理硬件资源,协调程序的运行。

当然,对于器件的选择也可通过 μVision2 软件界面中的 Project 任务栏中的“Select Device for Target 'Target 1'”命令来完成(如图 2-6 所示)。首先在“Data base”窗口选择器件生产厂家并双击,此时会出现该厂家产品的列表,用鼠标单击即可。如选择 AT89C51(如图 2-7 所示)。



图 2-5 为 μVision2 选择工程器件

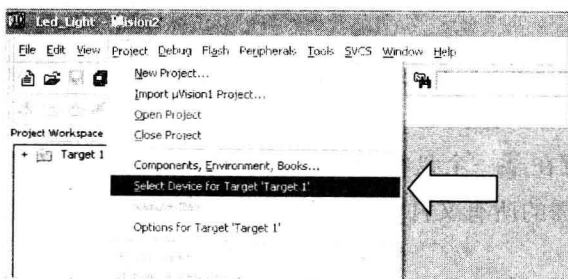


图 2-6 利用“Select Device for Target 'Target 1'”选择器件

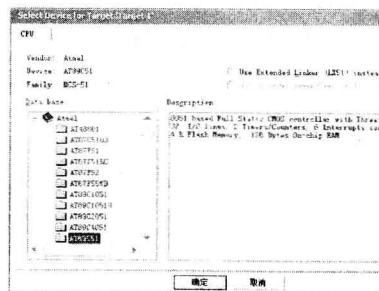


图 2-7 选择 AT89C51 单片机

 **注意** 在新建的工程中会自动产生一个文件 STARTUP.A51, 这是一个 C51 编译器的设置文件。

(1) 如果使用汇编语言编程要将其删除(使用鼠标右键点击后选择“Remove File”项移除即可);

(2) 若使用 C51 编程(且采用在线调试模式时)应将其保留并要对该文件的一条语句做修改(与仿真器有关,可参见第3章内容),将该文件第91行,CSEG AT 0 修改为 CSEG AT 8000H,即将程序代码段定义为 8000H 开始的单元。如果采用脱机模式则不用修改,按照 CSEG AT 0 处理。

2.2.4 为所创建的工程建立程序文件

到现在为止,我们已经建立了一个空的工程项目“Led_Light.uv2”,并为该工程项目选择了目标器件。但是,现在这个工程项目还是一个空的,必须将程序文件建立起来。

选择 File 任务栏中的 New 命令(如图 2-8 所示),窗口便会出现新文件窗口 Text1 (如图 2-9 所示)。

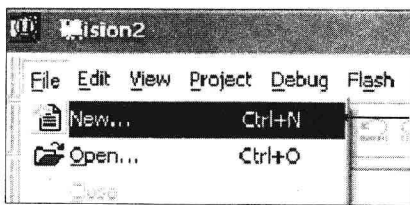


图 2-8 利用 File 任务栏中的 New 命令建立程序文件

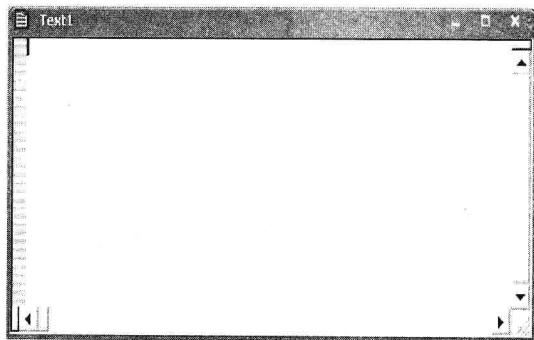


图 2-9 执行 New 命令后弹出的 Text1 新文件窗口

2.2.5 编辑程序源代码

在 μ Vision2 中的 Text1 窗口中,与其他文本编辑软件一样,用户可通过拷贝、粘贴、输入、删除、选择等基本的文字处理命令实现对程序源代码文件的编辑(如图 2-10 所示)。应当注意, μ Vision2 支持汇编语言(*.asm)和 C 语言(*.c),所以在后面的存盘操作和向工程添加文件时应当通过文件类型指明。现以 led_light.asm 为例输入该汇编语言源程序的文件,该程序清单如下:

```
org    0x8000
ljmp   0x8100
org    0x8100
start: mov    sp, #60h          ;主程序
      mov    a, #0fh
light: mov    p1, a
      cpl    a
```