



含光盘

单片机 C51

完全学习手册

兰吉昌 等编著



化学工业出版社

单片机

C51

完全学习手册

兰吉昌 等编著



化学工业出版社

·北京·

图书在版编目(CIP)数据

单片机 C51 完全学习手册/兰吉昌等编著. —北京: 化学工业出版社, 2008.10

ISBN 978-7-122-03582-0

I. 单… II. 兰… III. 单片微型计算机 IV. TP368.1

中国版本图书馆 CIP 数据核字 (2008) 第 129390 号

责任编辑: 刘 哲

装帧设计: 张 辉

责任校对: 宋 玮

出版发行: 化学工业出版社 (北京市东城区青年湖南街 13 号 邮政编码 100011)

印 刷: 大厂聚鑫印刷有限责任公司

装 订: 三河市延风装订厂

787mm×1092mm 1/16 印张 25½ 字数 679 千字 2009 年 1 月北京第 1 版第 1 次印刷

购书咨询: 010-64518888 (传真: 010-64519686) 售后服务: 010-64518899

网 址: <http://www.cip.com.cn>

凡购买本书, 如有缺损质量问题, 本社销售中心负责调换。

定 价: 68.00 元 (含光盘)

版权所有 违者必究

TP-368.1-62
157
12

前 言

使用 C 语言进行单片机开发是一个非常好的决策。C 语言具有良好的可读性，将极大地减少工程开发和维护的时间。

符合 C51 编译器的 C 语言就是 C51 语言。和汇编语言相比，用 C51 语言有很多优势，如可以不必对处理器的指令集了解甚多，这样可以使初学者能更快地上手进行开发和调试；整个应用系统的结构清晰，源代码变得可重复使用；可以运用和人的思维很接近的词汇和算法表达式等。

Keil C51 交叉编译器是一个基于 ANSI C 标准，针对 51 系列 MCU 的 C 编译器，经过多年的更新和发展，目前的版本生成的可执行代码快速、紧凑，在运行效率和速度上可以和汇编程序得到的代码相媲美。

Keil C51 的集成开发环境 μ Vision3 界面简单、友好，对于在编译过程所发生的警告和错误有一个完整的体系表，这样可以用户的编译和查错工作更加轻松自如。

本书详细介绍了 Keil C51 的 Windows 集成开发环境 μ Vision3 的强大功能和具体使用方法。在 μ Vision3 中，可以完成从源程序编写、编译、连接定位到目标文件的仿真调试等全部工作。 μ Vision3 是一种基于 Windows 的多窗口软件仿真器，它可以在完全没有 8051 单片机硬件的情况下模拟调试各种应用程序，即使对单片机内部特殊集成功能的编程，也可以找到对应的模拟窗口。

作为一本完全学习手册，本书通过大量的实例介绍了 C51 的基础知识和语法结构。第 1 章 C51 程序设计简介，简要介绍 C51 的基本知识。第 2 章 C51 程序设计基础，主要介绍 C51 语法规则。第 3 章是 C51 语句。C 语言是一种结构化的程序设计语言，它提供了非常丰富的程序控制语句。语句是组成程序的灵魂，学习掌握这些语句的使用方法是单片机 C 语言学习中的重点。第 4 章 C51 函数，主要介绍 C51 程序的主函数 main、子函数以及库函数。第 5 章 C51 数据结构，主要介绍 C51 的数据类型结构、联合和枚举等。第 6 章 C51 存储结构，主要介绍 C51 的存储结构，包括存储类型、存储模式和存储单元。第 7 章预处理包括宏定义、文件包含、条件编译等。预处理是 C 语言特有的功能，用于改进程序设计环境、提高程序设计效率。第 8 章 Keil C51 编译环境 μ Vision3，主要介绍 Keil C51 的集成编译环境 μ Vision3。第 9 章 Keil C51 编译，主要介绍在 μ Vision3 环境下进行程序的编写以及编译。第 10 章 Keil C51 调试，主要介绍 μ Vision3 环境下进行的程序调试。第 11 章 Keil C51 仿真，主要介绍 μ Vision3 环境下进行的程序仿真。第 12 章是 C51 在 RTX-51 实时操作系统的应用，主要介绍 Keil C51 软件本身就自带的操作系统 RTX-51 实时操作系统。第 13 章是 C51 程序设计综合实例，结合开发流程详细

讲解如何设计系统的硬件电路和软件系统。

本书强调先进性和实用性。全书各章列举了大量程序实例，并全部都收入到配套光盘中，光盘的源程序都已经调试并通过，可以直接应用到开发项目中去。

由于时间仓促以及水平所限，书中难免有不完善的地方，恳请广大读者给予指正。

编者

目 录

第 1 章 C51 程序设计简介	1
1.1 C51 程序特点	1
1.2 C51 开发流程	1
1.3 C51 编程规范	2
1.4 常用 C51 编程软件 Keil C51	2
1.4.1 安装 Keil C51 软件	3
1.4.2 Keil C51 软件的使用	5
第 2 章 C51 程序设计基础	9
2.1 标识符	9
2.2 关键字	9
2.3 数据类型	10
2.3.1 整型 (int)	11
2.3.2 浮点型 (float)	12
2.3.3 字符型 (char)	12
2.3.4 指针型 (*)	13
2.3.5 无值型 (void)	13
2.4 变量的作用域	13
2.4.1 自动型变量	14
2.4.2 外部型变量	14
2.4.3 静态型变量	15
2.4.4 寄存器型变量	16
2.5 const 修饰符	16
2.6 C51 的分隔符	16
2.6.1 “[]” 分隔符	17
2.6.2 “()” 分隔符	17
2.6.3 “{ }” 分隔符	17
2.6.4 “,” 分隔符	17
2.6.5 “:” 分隔符	17
2.6.6 “;” 分隔符	18
2.6.7 “...” 分隔符	18
2.6.8 “*” 分隔符	18
2.6.9 “=” 分隔符	18
2.6.10 “#” 分隔符	18
2.6.11 “^” 分隔符	18

2.7 运算符与表达式	19
2.7.1 赋值运算符	19
2.7.2 算术运算符	19
2.7.3 逻辑运算符	21
2.7.4 关系运算符	22
2.7.5 增量和减量运算符	22
2.7.6 位运算符	23
2.7.7 复合赋值运算符	24
2.7.8 逗号运算符	25
2.7.9 条件运算符	25
2.7.10 指针和地址运算符	26
2.7.11 sizeof 运算符	26
第3章 C51 语句	28
3.1 C51 语句的流程与控制	28
3.1.1 顺序结构 (sequence)	28
3.1.2 选择结构 (selection)	28
3.1.3 循环结构 (repetition)	28
3.1.4 自顶向下设计方法	29
3.2 说明语句	29
3.3 表达式语句	29
3.4 复合语句	30
3.5 条件语句	31
3.6 开关与跳转语句	33
3.6.1 开关语句	33
3.6.2 跳转语句	36
3.7 循环控制语句	38
3.7.1 for 循环	38
3.7.2 while 循环	39
3.7.3 do-while 循环	40
3.8 函数调用语句	41
3.9 空语句	41
第4章 C51 函数	43
4.1 函数的定义	43
4.2 函数说明	44
4.3 C51 函数的调用	45
4.3.1 函数的参数传递	46
4.3.2 递归函数的调用	46
4.3.3 数组作为函数参数	47
4.3.4 用函数指针调用函数	48
4.3.5 嵌套函数的调用	48
4.4 重入函数	49

4.5	main 函数	49
4.6	函数的重载	50
4.7	函数作用域	51
4.7.1	局部变量	51
4.7.2	全局变量	52
4.8	内部函数和外部函数	52
4.8.1	内部函数	52
4.8.2	外部函数	53
4.9	中断函数	53
4.9.1	内部中断函数的使用	54
4.9.2	外部中断函数的使用	55
第 5 章	C51 数据结构	56
5.1	C51 结构	56
5.1.1	结构体类型的定义	56
5.1.2	结构体变量的定义	59
5.1.3	结构体的初始化	59
5.1.4	结构体变量成员的引用	61
5.1.5	结构的嵌套	62
5.1.6	位结构	62
5.2	数组	63
5.2.1	一维数组的定义和初始化	63
5.2.2	一维数组元素的下标法引用	66
5.2.3	字符数组的输入和输出	67
5.2.4	二维数组的定义和初始化	69
5.2.5	结构体数组	71
5.3	联合与枚举	73
5.3.1	联合 (union)	73
5.3.2	枚举	75
5.4	类型说明	75
5.5	指针	76
5.5.1	指针的基本概念	76
5.5.2	数组元素的指针法引用	78
5.5.3	二维数组的指针	80
5.5.4	二级指针变量	80
5.5.5	指针数组	81
5.5.6	结构指针	81
5.5.7	特定寄存器指针和通用指针	82
第 6 章	存储结构	85
6.1	C51 存储类型	85
6.1.1	存储模式	87
6.1.2	存储空间	88

6.2	80C51 单片机的存储器结构	89
6.3	C51 支持的指针类型	93
6.4	内存分配	94
6.4.1	内存分配方法	94
6.4.2	动态分配内存	95
6.5	位操作	95
6.5.1	位操作指令	96
6.5.2	字段结构	97
6.6	访问绝对地址	98
第 7 章	预处理	100
7.1	文件包含	100
7.2	宏定义	101
7.2.1	无参宏定义	101
7.2.2	带参宏定义	103
7.2.3	#运算符	104
7.3	预处理指令	105
7.4	头文件	107
第 8 章	Keil C51 编译环境 μVision3	115
8.1	Keil μ Vision3 工作界面	115
8.1.1	Keil μ Vision3 的菜单栏	115
8.1.2	Keil μ Vision3 的工具栏	123
8.2	工程设置	124
第 9 章	Keil C51 编译	127
9.1	Keil C51 编译环境 μ Vision3	127
9.2	编译过程	127
9.2.1	编译步骤	129
9.2.2	编译过程产生的映像文件	131
9.3	Keil C51 编译器的控制命令	132
9.4	定制文件	133
9.4.1	STARTUP.A51	133
9.4.2	INIT.A51	134
9.4.3	XBANKING.A51	134
9.4.4	LIB 目录中的 C 文件	136
9.5	优化器	136
9.5.1	8051 特定优化	137
9.5.2	代码生成选项	137
9.6	C 程序和汇编的接口	137
9.6.1	函数参数	137
9.6.2	用固定存储区传递参数	137
9.6.3	通过寄存器传递参数	137

9.6.4	函数返回值	138
9.6.5	使用 SRC 控制指令	138
9.6.6	C 程序传递参数给汇编程序	141
9.7	C 程序和 PL/M-51 的接口	141
9.8	μ Vision 其他编译功能	141
9.8.1	在多个文件中查找	142
9.8.2	Source Browser (资源浏览器)	142
9.8.3	对话框工具	143
9.9	编译器限制	143
9.10	使用错误信息	144
9.10.1	语法和语义错误	144
9.10.2	警告	144
第 10 章	Keil C51 调试	146
10.1	μ Vision3 调试器	146
10.2	断点	147
10.3	在线编译	152
10.4	实例演示	152
第 11 章	Keil C51 仿真	159
11.1	Keil C51 的一些仿真窗口	159
11.1.1	模拟仿真单片机的外设接口	159
11.1.2	逻辑分析窗口	171
11.1.3	性能分析窗口	174
11.2	Monitor 51	176
11.3	利用开发板和仿真器进行仿真	177
11.3.1	SHX-51A 型开发板	177
11.3.2	SHX-C51 仿真器	188
第 12 章	C51 在 RTX-51 实时操作系统的应用	190
12.1	操作系统概述	190
12.1.1	操作系统的类型	190
12.1.2	操作系统的功能	191
12.1.3	实时操作系统 (Real Time OS)	191
12.2	RTX-51 系统概述	192
12.2.1	单任务程序	192
12.2.2	时间片轮转程序	192
12.2.3	用 RTX-51 进行循环调度	193
12.2.4	RTX-51 事件	193
12.2.5	os_wait 函数	194
12.2.6	RTX-51 的技术数据	194
12.3	RTX-51 系统函数	195
12.4	RTX-51 系统调试	195

12.5	RTX-51 系统应用实例	196
12.5.1	交通灯控制器命令	197
12.5.2	应用程序	197
12.5.3	TRAFFIC 工程	206
12.5.4	运行 TRAFFIC 程序	207
12.5.5	RTX 内核调试	208
第 13 章 C51 程序设计综合实例 209		
13.1	八位 7 段数码管动态显示	209
13.1.1	实例说明	209
13.1.2	八位 7 段数码管工作原理分析	209
13.1.3	硬件电路设计	210
13.1.4	软件设计	211
13.2	单片机实现 12864 的液晶显示	213
13.2.1	实例说明	213
13.2.2	设计思路分析	213
13.2.3	硬件电路设计	216
13.2.4	软件设计	220
13.3	单片机实现 4×4 矩阵键盘识别技术	224
13.3.1	实例说明	224
13.3.2	设计思路分析	224
13.3.3	硬件电路设计	225
13.3.4	软件设计	227
13.4	基于 MAX232 的 RS-232 串口通信	230
13.4.1	串行通信	230
13.4.2	芯片选取	231
13.4.3	芯片工作原理	232
13.4.4	硬件电路设计	233
13.4.5	软件设计	233
13.5	基于 MAX527 的串行 D/A 转换	235
13.5.1	设计思路分析	235
13.5.2	芯片选取	237
13.5.3	芯片的工作原理	238
13.5.4	硬件电路	239
13.5.5	软件设计	239
13.6	LED 流水灯显示	240
13.6.1	实例说明	241
13.6.2	流水灯及 LED 工作原理分析	241
13.6.3	硬件电路设计	241
13.6.4	程序说明	242
13.7	基于 MAX197 的并行 A/D 转换实例	242
13.7.1	实例说明	242
13.7.2	设计思路分析	243

13.7.3	硬件电路设计	245
13.7.4	软件设计	249
13.8	基于 RTL8019AS 实现以太网通讯	252
13.8.1	芯片选取	253
13.8.2	芯片工作原理	256
13.8.3	硬件电路设计	258
13.8.4	软件设计	259
13.9	单片机实现智能信号发生器	263
13.9.1	实例说明	263
13.9.2	MAX038 信号发生芯片	263
13.9.3	硬件电路设计	265
13.9.4	软件设计	266
13.10	单片机实现简单音乐发生器	268
13.10.1	实例说明	269
13.10.2	设计思路分析	269
13.10.3	硬件电路设计	271
13.10.4	软件设计	273
13.11	RS-485 总线现场监控系统	276
13.11.1	实例说明	277
13.11.2	RS-485 总线工作原理分析	277
13.11.3	硬件电路设计	277
13.11.4	软件设计	279
13.12	单片机实现语音录放	285
13.12.1	实例说明	285
13.12.2	语音芯片工作原理	286
13.12.3	硬件电路设计	286
13.12.4	软件设计	288
13.13	单片机时钟设计	291
13.13.1	实例说明	291
13.13.2	设计思路	291
13.13.3	软件设计	291
13.14	基于 T6369C 液晶显示控制器的单片机接口	299
13.14.1	实例说明	299
13.14.2	芯片介绍	299
13.14.3	硬件电路设计	305
13.14.4	软件设计	306
13.15	非接触式 IC 卡读写	319
13.15.1	实例说明	320
13.15.2	射频 IC 卡和卡片读写器原理	320
13.15.3	硬件电路设计	320
13.15.4	软件设计	322
13.16	基于 MPX4105 的数字气压计设计	331
13.16.1	实例说明	331

13.16.2	应用芯片介绍	332
13.16.3	硬件电路设计	335
13.16.4	软件设计	336
附录 A	致命错误信息	339
附录 B	语法和语义错误信息	342
附录 C	警告信息	356
附录 D	L51 连接定位器使用错误提示	359
附录 E	μ Vision3 调试命令	367
附录 F	C51 库函数	389

第 1 章 C51 程序设计简介

采用 C51 编程是目前 51 系列单片机开发的趋势所在, 一般来说, 使用 C 语言开发速度快, 代码可重复使用, 程序结构清晰、易懂、易维护, 很容易做一些比较大型的项目。C51 编译器不但可以缩短单片机控制系统的开发周期, 而且易于调试和维护。此外, C51 语言还有许多强大的功能, 如提供丰富的库函数供用户直接调用, 完整的编译控制指令为程序调试提供必要的符号信息等。总之, C51 语言是单片机开发的强有力工具。

1.1 C51 程序特点

汇编语言和 C51 是单片机应用系统开发常用的编程工具。汇编语言能直接操作硬件, 生成的机器代码简洁, 占 ROM 空间少, 指令的执行速度快, 适用于一些对时序要求特别精确的场合, 如遥控解码、步进电机的控制等。但其指令系统的固有格式受硬件结构的限制很大, 且难于编写与调试, 可移植性也差。随着单片机硬件性能的提高, 其工作速度越来越快, 因此在编写单片机应用系统程序时, 更着重于程序本身的编写效率。当编写一个较大的应用程序时, 从内部 RAM 单元的规划、堆栈的保护、ROM 中断地址的安排、PC 的维护, 到内部和外部资源的整合、系统的调试和维护, 尤其当程序中存在大量计算时, 程序就显得特别庞杂, 如果全部采用汇编语言, 则工作量非常大。故除了一些要求特殊的功能模块采用汇编语言外, 一般情况下, 都采用主要面向应用、运算符丰富、结构简洁的 C51 实现。

1.2 C51 开发流程

C51 的项目开发流程可以简单总结为下面几个步骤:

- (1) 创建一个项目, 从器件库中选择目标器件, 配置工具设置;
- (2) 创建源程序;
- (3) 用项目管理器生成应用;
- (4) 修改源程序中的错误;
- (5) 测试, 连接应用。

Keil C51 是一个标准的 Windows 应用程序, 集成了 C51 编译器、A51 汇编器、L51 连接器、实时操作系统、项目管理器及调试器等, 可以完成编辑、编译、链接、调试及仿真等整个开发流程。

针对目前应用比较广泛的单片机应用开发软件 Keil C51, 其开发流程如下:

- (1) 建立工程;
- (2) 为工程选择目标器件;
- (3) 设置工程的配置参数;
- (4) 打开/建立程序文件;
- (5) 编译和连接工程;

- (6) 纠正程序中的书写和语法错误并重新编译连接;
- (7) 对程序中某些纯软件的部分使用软件仿真验证;
- (8) 使用硬件仿真器对应用程序进行硬件仿真;
- (9) 将生成的 Hex 文件烧写到 ROM 中运行测试。

1.3 C51 编程规范

编程首先是要考虑程序的可行性, 然后是可读性、可移植性、健全性以及可测试性。下面是 C51 的一些编程规范。

- (1) 当项目比较大时, 最好分模块编程, 这样既方便修改又便于重用和阅读。
- (2) 每个文件的开头应该写明这个文件是哪个项目里的哪个模块, 是在什么编译环境下编译的, 编程者和编程日期。值得注意的是不要忘了编程日期, 因为以后再看文件时, 会知道大概是什么时候编写的、有些什么功能, 并且可能知道类似模块之间的差异。
- (3) 一个 C 源文件配置一个 h 头文件, 或者整个项目的 C 文件配置一个 h 头文件。
- (4) 一些常量或者常需要在调试时修改的参数最好用 `#define` 定义, 但要注意宏定义只是简单的替换, 因此有些括号不可缺少。
- (5) 不要轻易调用某些库函数, 因为有些库函数代码很长。
- (6) 书写代码时要注意括号对齐, 固定缩进, 一个 `{}` 占一行, `if/for/while/do` 等语句各占一行, 执行语句不能紧跟其后, 无论执行语句有多少都要加 `{}`。
- (7) 一行只实现一个功能。
- (8) 重要难懂的代码要写注释, 函数和全局变量要写注释, 一些局部变量也要写注释。注释写在代码的上方或者右方。
- (9) 变量和函数的命名最好能做到望文生义。
- (10) 函数的参数和返回值没有的话最好使用 `void`。
- (11) 一些常数和表格应该放到 `code` 中去, 以节省 RAM。
- (12) 程序编完编译后看有多少 `code` 和多少 `data`。
- (13) 程序要能方便地进行测试, 其实这也与编程的思维有关; 一般有自上而下先整体再局部、自下而上先局部再整体、结合两者往中间凑三种方法。

1.4 常用 C51 编程软件 Keil C51

Keil C51 是 Keil Software 公司出品的 51 系列兼容单片机 C 语言软件开发系统, 是专为 80C51 系列单片机设计的一种高效的 C 语言编译器, 它是众多 51 单片机开发软件中应用最广泛的一个, 集编辑、编译、仿真于一体, 支持汇编语言、PLM 语言和 C 语言的程序设计, 对 C 语言的支持也是最好的, 而且界面直观, 易学易用。使用它可以缩短开发周期, 降低开发成本, 而且开发出的系统易于维护, 可靠性高, 可移植性好, 即使在代码的使用效率上, 也完全可以和汇编语言相比, 因此目前它已成为开发 80C51 系列单片机的流行工具。

Keil C51 软件提供丰富的库函数和功能强大的集成开发调试工具, 全 Windows 界面。另外, Keil C51 生成的目标代码效率非常高, 多数语句生成的汇编代码很紧凑, 容易理解。在开发大型软件时更能体现高级语言的优势。

C51 工具包的整体结构如图 1-1 所示。

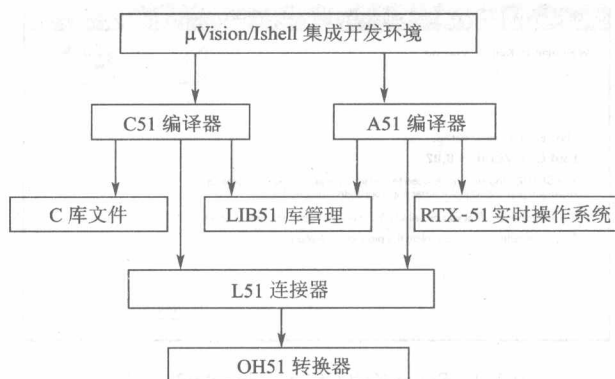


图 1-1 C51 工具包的整体结构

其中 μ Vision 与 Ishell 分别是 C51 在 Windows 和 DOS 下的集成开发环境 (IDE)，可以完成编辑、编译、连接、调试、仿真等整个开发流程。用户可用 IDE 本身或其他编辑器编辑 C 或汇编源文件，然后分别由 C51 及 A51 编译器编译生成目标文件 (.OBJ)。目标文件可由 LIB51 创建生成库文件，也可以与库文件一起经 L51 连接器定位生成绝对目标文件 (.ABS)。ABS 文件由 OH51 转换成标准的 Hex 文件，以供调试器 dScope51 或 tScope51 使用进行源代码级调试，也可由仿真器使用直接对目标板进行调试，还可以直接写入程序存储器如 EPROM 中。

此外，Keil C51 的集成开发环境 μ Vision3 还具有极其强大的软件环境、友好的操作界面和简单快捷的操作方法，其主要表现在以下几点：

- (1) 丰富的菜单栏；
- (2) 可以快速选择命令按钮的工具栏；
- (3) 一些源代码文件窗口；
- (4) 对话框窗口；
- (5) 直观明了的信息显示窗口。

1.4.1 安装 Keil C51 软件

安装 Keil C51 集成开发软件，必须满足最低的硬件和软件要求，才能确保编译器以及其他程序功能正常，必须具有：

- (1) Pentium、Pentium-II 或兼容处理器的 PC；
- (2) Windows95、Windows98、Windows NT4.0；
- (3) 至少 16MB RAM；
- (4) 至少 20MB 硬盘空间。

安装步骤如下。

(1) 在 Keil C51 Version 8.02 软件的安装目录下双击安装软件 v51v802.EXE，弹出如图 1-2 所示对话框。

(2) 单击按钮 后弹出如图 1-3 所示的对话框。在该对话框内勾选下方的确认同意安装选项。

(3) 单击按钮 后弹出如图 1-4 所示的对话框，确认安装路径。系统默认选择 C:\Keil 为安装路径。也可以自定义其他安装路径。

(4) 选择好安装路径后，单击按钮 弹出如图 1-5 所示的对话框。要求填写用户的名字、单位以及 E-mail。

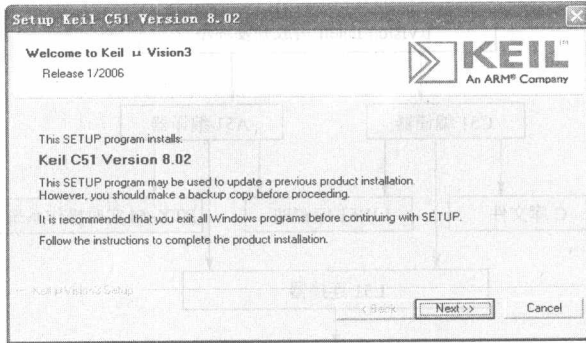


图 1-2 Setup Keil C51 Version 8.02 对话框

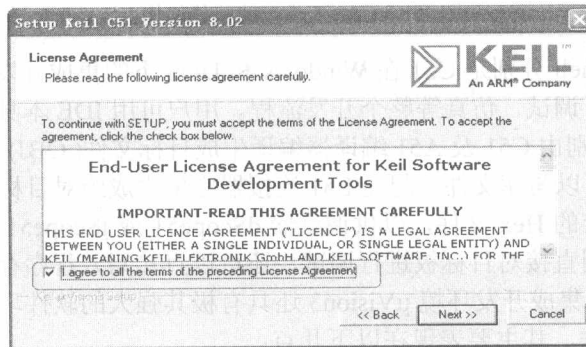


图 1-3 “同意安装”对话框

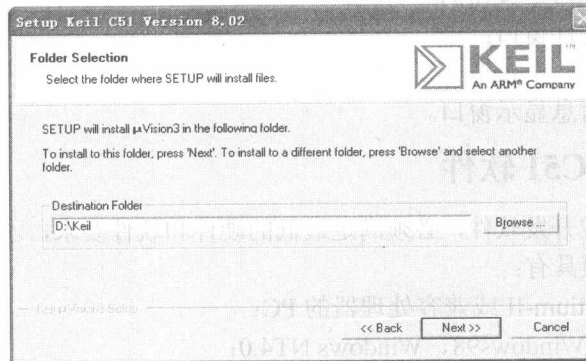


图 1-4 “选择路径”对话框

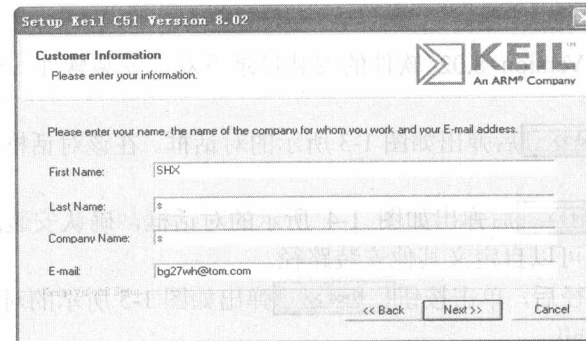


图 1-5 “用户信息”对话框