

21 世纪高等院校规划教材 · 计算机类

8086/8088 和 ARM 核
汇编语言程序设计

8086/8088 和 ARM 核 汇编语言程序设计

主 编 李敬兆
副主编 丁 刚 张 雷 谢晓东 詹 林

中国科学技术大学出版社

21 世纪高等院校规划教材 · 计算机类

TP332
76

8086/8088 和 ARM 核汇编 语言程序设计

主 编 李敬兆

副主编 张 雷 谢晓东

丁 刚 詹 林

中国科学技术大学出版社

2006 · 合肥

内 容 简 介

本书以 8086/8088 系列 16 位微型计算机和基于 ARM 的 32 位嵌入式微处理器为背景,系统地介绍了这两类微处理器的汇编语言程序设计。全书共 20 章,分两个部分。第一部分 8086/8088 汇编语言程序设计,共 14 章,以 8086/8088 作为背景系统,介绍微机原理的基础知识及汇编语言设计的方法和技术。对 80286 以上微处理器及其汇编语言进行了简单介绍。第二部分基于 ARM 核的汇编程序设计,共 6 章,以目前最为流行的嵌入式系统—基于 ARM 核的微处理器为背景,介绍了基于 ARM 的嵌入式系统的基础知识及汇编语言程序设计的方法和技术。

本书可作为高等学校计算机及其相关专业“汇编语言程序设计”等课程的教材或参考书,主要读者为计算机、电子工程和自动控制等相关学科的本科生和研究生,也适用于计算机应用开发人员、希望深入学习微型计算机和基于 ARM 嵌入式技术的普通读者和培训班学员。

图书在版编目(CIP)数据

8086/8088 和 ARM 核汇编语言程序设计/李敬兆主编. —合肥:中国科学技术大学出版社, 2006.9

ISBN 7-312-01980-3

I .8… II.李… III.①微处理器-系统设计-高等学校-教材 ②汇编语言-程序设计-高等学校-教材 IV.①TP332②TP313

中国版本图书馆 CIP 数据核字(2006)第 086317 号

中国科学技术大学出版社 出版发行

(安徽省合肥市金寨路 96 号, 邮政编码: 230036)

中国科学技术大学印刷厂印刷

全国新华书店经销

开本: 787×1092/16

印张: 19.75

字数: 490 千

2006 年 9 月第 1 版

2006 年 9 月第 1 次印刷

印数: 1—4000 册

ISBN7-312-01980-3/TP·370

定价: 29.00 元

编委会名单

主 任：汪光阳

副主任：周鸣争 李敬兆 方潜生

委 员：（以姓氏笔划为序）

丁 刚 方潜生 纪 平

汪光阳 李敬兆 陈 辉

张 雷 周鸣争 宗欣欣

夏 巍 谢晓东 詹 林

前 言

汇编语言是计算机能够提供给用户使用的最快而又最有效的语言,也是能够利用计算机所有硬件特性并能直接控制硬件的唯一语言。因而,对程序的空间和时间要求很高的场合,汇编语言的应用是必不可少的。至于很多需要直接控制硬件的应用场合,则更是非用汇编语言不可。由于汇编语言如此地接近计算机硬件,因此,它可以最大限度地发挥计算机硬件的性能。用汇编语言编写的程序速度通常要比高级语言快几倍、几十倍、甚至成百上千倍。解释语言其程序速度更无法与汇编语言程序同日而语。

汇编语言课程是高等教育计算机专业本科学生的必修课,是计算机专业语言类最基础的课程之一。学好这门课程,可为学习相关的专业课打下基础。

目前,除了 80X86 系列 PC 机外,嵌入式技术尤其是基于 ARM 核微处理器的嵌入式系统发展迅猛,成为后 PC 时代的佼佼者。本书就是以 8086/8088 系列微机和基于 ARM 核的嵌入式微处理器为背景,介绍这两类微处理器的汇编语言程序设计。

全书共 20 章,分两个部分。第一部分 8086/8088 汇编语言程序设计,共 14 章,以 8086/8088 作为背景系统,介绍微机原理的基础知识及汇编语言程序设计的方法和技术。第 1 章和第 2 章介绍微机基础知识以及 80x86 机型的系统结构。这两章为后面的讲解提供必要的微机系统软件、硬件知识。第 3 章详细介绍 8086/8088 的寻址方式和指令系统。第 4 章介绍伪指令、汇编语言程序格式等知识。第 5 章至第 10 章讲述基本程序设计、输入/输出程序设计、循环程序设计、分支程序设计、子程序设计、数据运算程序设计的方法。第 11 章讲述高级汇编技术,包括宏汇编、条件汇编、重复汇编等。第 12 章讲述 DOS 功能调用和 BIOS 功能调用知识,并通过程序设计实例分析系统功能调用的实现。第 13 章介绍汇编语言与 C/C++ 的混合编程技术,第 14 章介绍了关于 80x86 及 Pentium 的增强和扩展指令内容。

第二部分 基于 ARM 核的汇编语言程序设计,共 6 章,以目前最为流行的嵌入式系统——基于 ARM 核的微处理器为背景,介绍了基于 ARM 核的嵌入式系统的基础知识及汇编语言程序设计的方法和技术。第 15 章介绍了目前常用的 ARM CPU,第 16 章介绍了 ARM7TDMI 的系统结构,第 17 章讲解了 ARM/Thumb 汇编语言(格式),第 18 章和第 19 章分别讲解了 ARM 指令系统和 Thumb 指令系统,第 20 章给出了 ARM 汇编程序设计实例。

本书由安徽理工大学李敬兆教授担任主编,安徽工业大学张雷老师、安徽工程科技学院谢晓东老师、安徽建筑工程学院丁刚老师、安徽理工大学詹林老师担任副主编。第 1 章、第 2 章、第 9 章、第 10 章由安徽工业大学张雷、纪平两位老师编写,第 3 章、第 4 章、第 11 章、第 12 章由安徽工程科技学院谢晓东老师编写,第 5 章、第 6 章、第 11 章、第 12 章由安徽建筑工程学院丁刚、夏巍两位老师编写,其余各章由安徽理工大学李敬兆、詹林、陈辉、宗欣欣四位老师编写。

由于编者水平有限及时间仓促,书中错漏之处,敬请读者批评指正。

编 者

2006 年 6 月

目 录

第一部分 8086/8088 汇编语言程序设计

第1章 汇编语言基础知识	(3)
1.1 汇编语言概述	(3)
1.1.1 汇编语言基本概念	(3)
1.1.2 汇编语言的特点	(4)
1.2 进位计数制及其转换	(5)
1.2.1 数与数制	(6)
1.2.2 不同数制之间的转换	(7)
1.3 计算机中数与字符的表示方法	(11)
1.3.1 数值型数据的编码	(11)
1.3.2 非数值型数据的编码	(17)
小 结	(20)
习 题	(21)
第2章 IBM-PC 系统结构	(22)
2.1 微型计算机发展概述	(22)
2.2 微型计算机系统	(23)
2.2.1 硬件	(23)
2.2.2 软件	(25)
2.3 8086/8088CPU 的内部结构	(25)
2.3.1 8086/8088CPU 功能结构	(25)
2.3.2 8086/8088CPU 的寄存器组	(27)
2.4 8086/8088 的存储器结构	(31)
2.4.1 存储单元的地址和内容	(31)
2.4.2 存储器的分段	(31)
2.4.3 物理地址的形成	(32)
习 题	(33)
第3章 8086/8088 寻址方式和指令系统	(34)
3.1 指令的基本格式	(34)
3.2 8086/8088 寻址方式	(34)
3.2.1 与数据有关的寻址方式	(35)
3.2.2 与转移地址有关的寻址方式	(40)

3.3 8086/8088 的指令系统	(42)
3.3.1 数据传送指令	(42)
3.3.2 算术运算指令	(46)
3.3.3 逻辑指令	(51)
3.3.4 串处理指令	(54)
3.3.5 控制转移指令	(57)
3.3.6 处理器控制指令	(60)
习 题	(61)
第4章 8086/8088 汇编语言(格式)	(65)
4.1 汇编语言和汇编程序的基本概念	(65)
4.1.1 汇编语言	(65)
4.1.2 汇编程序	(65)
4.2 汇编语言源程序书写格式	(66)
4.2.1 汇编语言源程序的分段结构	(66)
4.2.2 汇编语言源程序的语句类型	(66)
4.3 伪指令语句	(66)
4.3.1 数据定义及存储器分配伪指令	(66)
4.3.2 符号定义伪指令	(67)
4.3.3 段定义伪指令	(69)
4.3.4 过程定义伪指令	(71)
4.3.5 汇编结束伪指令 END	(71)
4.4 8086/8088 汇编语言程序格式	(72)
4.4.1 名字项	(72)
4.4.2 操作数项	(73)
4.4.3 表达式	(76)
4.5 汇编语言的上机过程	(77)
4.5.1 汇编语言的工作环境及上机步骤	(77)
4.5.2 汇编语言程序运行实例	(77)
习 题	(81)
第5章 基本程序设计	(84)
5.1 程序设计方法概述	(84)
5.1.1 编写汇编语言程序的步骤	(85)
5.1.2 判断程序质量的标准	(85)
5.1.3 汇编语言程序的开发过程	(85)
5.1.4 完整的汇编语言程序框架	(86)
5.2 流程图	(87)
5.2.1 流程图的功能及意义	(87)

5.2.2 流程图的画法规定	(88)
5.3 结构化程序设计	(88)
5.4 顺序结构的程序设计	(90)
习 题	(92)
第6章 输入/输出程序设计	(93)
6.1 输入/输出指令	(93)
6.1.1 I/O 端口编址方式	(93)
6.1.2 输入/输出指令	(94)
6.1.3 I/O 端口寻址方式	(94)
6.2 CPU 与外设传送数据的控制方式	(94)
6.3 程序查询传送方式	(97)
6.4 中断传送方式	(99)
6.4.1 中断的概念	(99)
6.4.2 8086/8088 中断系统	(100)
6.4.3 中断传送方式程序举例	(106)
习 题	(107)
第7章 8086/8088 循环程序设计	(109)
7.1 循环程序结构	(109)
7.2 循环程序设计	(111)
习 题	(115)
第8章 8086/8088 分支程序设计	(118)
8.1 分支程序的结构形式	(118)
8.2 分支结构程序设计	(119)
8.2.1 测试法分支程序设计	(119)
8.2.2 跳跃表法	(120)
8.2.3 举例	(121)
习 题	(126)
第9章 子程序设计	(129)
9.1 子程序设计方法	(129)
9.1.1 子程序的定义	(129)
9.1.2 子程序调用与返回	(130)
9.1.3 寄存器内容的保护与恢复	(132)
9.2 子程序的参数传递	(132)
9.2.1 用寄存器传递参数	(133)
9.2.2 用存储单元传递参数	(135)
9.2.3 用堆栈传递参数	(136)
9.3 子程序嵌套与递归	(138)

9.4 子程序库	(139)
习 题	(142)
第 10 章 数据运算程序设计	(144)
10.1 多精度的加减运算	(144)
10.1.1 二进制加减运算	(144)
10.1.2 压缩型 BCD 码加减运算	(146)
10.1.3 非压缩型 BCD 码(ASCII 码)加减运算	(147)
10.2 乘除运算	(149)
10.2.1 一般整数二进制乘除运算	(149)
10.2.2 多字节整数十进制乘除运算	(151)
小 结	(154)
习 题	(154)
第 11 章 高级汇编语言技术	(156)
11.1 宏汇编	(156)
11.1.1 宏定义	(156)
11.1.2 宏调用和宏扩展	(157)
11.1.3 宏定义和宏调用中参数的使用	(158)
11.1.4 宏嵌套	(162)
11.2 重复汇编	(163)
11.2.1 使用 REPT 伪指令的重复汇编结构	(163)
11.2.2 使用 IRP 伪指令的重复汇编结构	(164)
11.2.3 使用 IRPC 伪指令的重复汇编结构	(165)
11.3 条件汇编	(166)
11.3.1 条件汇编的概念及条件汇编结构	(166)
11.3.2 条件汇编伪指令	(166)
11.4 库的使用	(171)
11.4.1 库的建立	(171)
11.4.2 库的使用	(171)
习 题	(171)
第 12 章 BIOS 和 DOS 中断调用	(173)
12.1 BIOS 中断调用	(173)
12.1.1 显示输出管理(INT 10H)	(174)
12.1.2 键盘输入管理(INT 16H)	(178)
12.1.3 打印输出管理(INT 17H)	(180)
12.1.4 时间中断调用(INT 1AH)	(180)
12.2 DOS 功能调用	(182)
12.2.1 常用 DOS 操作系统中断调用	(183)

12.2.2 常用 INT 21H 系统功能中断	(184)
习 题	(189)
第 13 章 汇编语言与 C/C++ 的混合编程	(191)
13.1 C/C++ 语言参数调用协议	(191)
13.1.1 入口参数传递规则	(191)
13.1.2 返回参数传递规则	(192)
13.2 汇编语言与 C/C++ 语言的连接	(193)
13.2.1 模块连接法	(193)
13.2.2 嵌入式汇编法	(198)
小 结	(200)
习 题	(201)
第 14 章 80286/386/486/Pentium	(202)
14.1 80x86 CPU 特点及其寄存器组	(202)
14.1.1 80x86CPU 特点	(202)
14.1.2 80x86 寄存器组	(204)
14.2 80x86 工作模式	(206)
14.3 80286、80386、80486 和 Pentium 处理器的指令	(207)
14.3.1 数据传送类指令	(207)
14.3.2 算术运算类指令	(209)
14.3.3 位操作类指令	(209)
14.3.4 比较类指令	(210)
14.3.5 串操作类指令	(210)
14.3.6 逻辑运算类指令	(211)
14.3.7 堆栈操作类指令	(212)
14.3.8 条件设置和控制转移类指令	(213)
14.3.9 类型转换类指令	(214)
14.3.10 I/O 类指令	(214)
习 题	(215)

第二部分 基于 ARM 核的汇编语言程序设计

第 15 章 ARM CPU 概述	(219)
15.1 ARM-Advancei RISC Machines	(219)
15.2 ARM 微处理器的应用领域及特点	(220)
15.2.1 ARM 微处理器的应用领域	(220)
15.2.2 ARM 微处理器的特点	(220)
15.3 ARM 微处理器系列	(220)
15.3.1 ARM7 微处理器系列	(221)

15.3.2	ARM9 微处理器系列	(221)
15.3.3	ARM9E 微处理器系列	(222)
15.3.4	ARM10E 微处理器系列	(222)
15.3.5	SecurCore 微处理器系列	(223)
15.3.6	StrongARM 微处理器系列	(223)
15.3.7	Xscale 处理器	(223)
15.4	ARM 微处理器结构	(224)
15.4.1	RISC 体系结构	(224)
15.4.2	ARM 微处理器的寄存器结构	(224)
15.4.3	ARM 微处理器的指令结构	(225)
15.5	ARM 微处理器的应用选型	(225)
小 结		(226)
习 题		(226)
第 16 章	ARM 系统结构	(227)
16.1	ARM 微处理器的工作状态	(227)
16.2	ARM 微处理器的工作模式	(228)
16.3	ARM 微处理器的寄存器组织	(229)
16.3.1	ARM 状态下的寄存器组织	(229)
16.3.2	Thumb 状态下的寄存器组织	(231)
16.3.3	程序状态寄存器	(233)
16.4	ARM 异常处理	(234)
16.4.1	ARM 中的异常类型	(235)
16.4.2	ARM 对异常的响应	(235)
16.4.2	ARM 从异常的返回	(236)
16.4.3	异常向量地址及优先级	(236)
16.5	ARM 体系中的存储系统	(236)
16.5.1	ARM 体系中的存储空间	(236)
16.5.2	ARM 存储器格式	(237)
小 结		(237)
习 题		(238)
第 17 章	ARM/Thumb 汇编语言	(239)
17.1	ARM 指令的分类及格式	(239)
17.1.1	ARM 指令的分类	(239)
17.1.2	ARM 指令一般编码格式	(239)
17.1.3	ARM 指令基本语法格式	(240)
17.2	ARM 指令的寻址方式	(242)
17.3	ARM 汇编语言的语句格式	(244)

17.3.1 在汇编语言程序中常用的符号	(244)
17.3.2 汇编语言程序中的表达式和运算符	(245)
小 结	(248)
习 题	(248)
第 18 章 ARM 指令系统	(249)
18.1 概 述	(249)
18.2 ARM 指令的格式与条件码	(250)
18.2.1 ARM 指令的格式与条件码	(250)
18.2.2 ARM 指令的条件码	(252)
18.3 单寄存器存取指令	(253)
18.4 多寄存器存取指令	(255)
18.5 数据交换指令	(256)
18.6 数据处理指令	(256)
18.7 乘法指令与乘加指令	(260)
18.8 ARM 分支指令	(262)
18.9 软件中断指令	(263)
18.10 程序状态寄存器访问指令	(263)
18.11 ARM 伪指令	(264)
小 结	(266)
习 题	(266)
第 19 章 Thumb 指令系统	(267)
19.1 概 述	(267)
19.2 Thumb 存储器访问指令	(268)
19.2.1 单寄存器加载/存储指令 LDR 和 STR	(268)
19.2.2 寄存器入栈及出栈指令 PUSH 和 POP	(270)
19.2.3 多寄存器加载/存储指令 LDMIA 和 STMIA	(270)
19.3 Thumb 数据处理指令	(271)
19.3.1 Thumb 数据处理指令——数据传送指令	(271)
19.3.2 Thumb 数据处理指令——算术逻辑运算指令	(272)
19.4 Thumb 分支指令	(277)
19.5 Thumb 杂项指令 SWI	(278)
19.6 Thumb 伪指令	(279)
小 结	(280)
习 题	(280)
第 20 章 ARM 汇编程序设计	(281)
20.1 ARM 汇编器所支持的伪指令	(281)
20.1.1 符号定义(Symbol Definition)伪指令	(281)

20.1.2	数据定义(Data Definition)伪指令	(283)
20.1.3	汇编控制(Assembly Control)伪指令	(285)
20.1.4	其他常用的伪指令	(287)
20.2	汇编语言的程序结构	(291)
20.2.1	汇编语言的程序结构	(291)
20.2.2	汇编语言的子程序调用	(292)
20.2.3	汇编语言程序示例	(293)
20.2.4	汇编语言与 C/C++ 的混合编程	(297)
小 结		(299)
习 题		(299)



第一部分

8086/8088 汇编语言程序设计



第1章 汇编语言基础知识

本章介绍学习汇编语言程序设计所必须具备的基本知识,主要包括汇编语言的基本概念及计算机中数据的表示方法。通过本章的学习,读者应能了解汇编语言概念及其使用的进位计数制、不同进位计数制之间的转换、计算机编码以及基本数据类型。

本章内容要点:

- 📖 汇编语言的概念
- 📖 汇编语言的特点
- 📖 不同进位计数制之间的转换
- 📖 计算机编码

1.1 汇编语言概述

1.1.1 汇编语言基本概念

自然语言是具有特定语音和语法等规范的、用于人类表达思想并实现相互交流的工具。人与人之间只有使用同一种语言才能进行直接交流,否则就必须通过翻译。要使计算机为人类服务,人们就必须借助某种工具,告诉计算机“做什么”甚至“怎么做”,这种工具就是程序设计语言。

程序设计语言通常分为三类:机器语言、汇编语言和高级语言。而前两种语言与机器密切相关,统称为低级语言。

1. 机器语言

机器语言是计算机第一代语言,它全部由0、1代码组成,是能够直接被机器所接受的语言,是最底层的计算机语言。

机器语言不容易记忆,程序编写难度大,调试修改繁琐,且不易移植,现在程序员很少用。但机器语言执行速度最快,它是一种面向机器的程序设计语言。

2. 汇编语言

为了克服机器语言难以记忆、表达和阅读的缺点,人们采用具有一定含义的符号作为助记符,用指令助记符、符号地址等组成的符号指令称为汇编格式指令(或汇编指令)。例如,用ADD表示加法指令, SUB表示减法指令, MOV表示传送指令等。汇编语言是汇编指令集、伪指令集和使用它们规则的统称。伪指令的概念将在第4章介绍。

汇编语言比机器语言直观,容易记忆和理解,用汇编语言编写的程序也比机器语言程序易读、易检查、易修改。对于不同的计算机,针对同一问题所编写的汇编语言源程序是互不通用的。



用汇编语言编写的程序执行效率比较高,但通用性与可移植性仍然比较差。计算机不能直接识别用汇编语言编写的程序,必须由一种专门翻译程序将汇编语言程序翻译成机器语言程序,计算机才能执行。

例如,在 8086 机器下,分别用汇编语言和机器语言计算 $10+20$ 的程序代码如下:

汇编语言	机器语言
MOV AL, 10	B0 0A
ADD AL, 20	04 14

显然,使用汇编语言编写的程序要比机器语言更容易理解。

3. 高级语言

机器语言和汇编语言以外的程序设计语言统称高级语言。其特点是更加接近自然语言和惯用的数学表达形式,与计算机硬件结构无关,因而便于使用,便于交流和推广。例如,可以在程序中直接使用表达式 $10+20$ 。目前,常用的高级语言数十种,如 C、C++、Pascal、Basic、Java 等。

高级语言可分成编译型和解释型高级语言,需要分别使用编译程序和解释程序将源程序翻译成机器语言程序,才能交计算机执行。总之,高级语言编程效率高,但运行效率低。

1.1.2 汇编语言的特点

汇编语言相对机器语言而言好记好用,但远不如高级语言方便、实用,而且编写同样的程序,使用汇编语言比使用高级语言花费的时间更多,调试和维护更困难,在计算机速度大大提高和存储器容量大大增加的今天,高级语言的使用更为广泛和普遍(特别是编写大型程序)。既然如此,为什么还要使用汇编语言呢?主要有两个原因:性能和对计算机的完全控制。一般而言,汇编语言具有如下特点:

(1) 执行速度快

一个汇编语言程序,要比高级语言程序执行得更快。程序的执行速度对于某些应用来说是至关重要的。对于这些应用,单纯使用高级语言往往达不到要求,单纯使用汇编语言编写程序也并不是最好的方案,许多成功的大型应用程序往往使用的是混合编程。首先使用高级语言编写整个程序,然后测试程序的执行时间,再使用汇编语言重写其中最费时间的部分。这样做的依据是在实际使用中,通常程序的大部分执行时间都花费在一小部分代码上。例如编写图像处理程序,就往往使用汇编语言编写软件中的关键部分。

(2) 程序短小

一个汇编语言程序,要比高级语言程序更小。在某些情况下,设备中的嵌入式处理器往往只有很少的内存,使用汇编语言可能是唯一的方法。如智能卡中有 CPU,但是智能卡中很难有 1 MB 以上的内存,也不可能有带分页的硬盘,但智能卡又必须执行复杂的加密解密计算。个人数字助理(PDA)和其他使用电池作为能源的无线电子设备,为了节省电池的电力,往往也只有很少的内存,它们也需要使用短小精悍而且具有高效率的机器代码。

(3) 可以直接控制硬件

某些应用程序要求能够完全控制计算机硬件,这也必须使用汇编语言。如操作系统中的低级中断和陷阱处理程序,以及许多嵌入式实时系统中的设备控制程序都属于这一类应用。