

教育部-IBM高校合作项目
精品课程系列教材



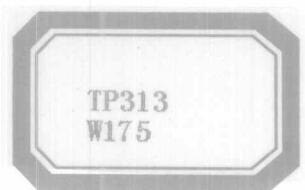
大型主机 汇编语言程序设计

王 芬 马光志 编著



清华大学出版社





郑州大学 *04010747729+*

教育部-IBM高校合作项目
精品课程系列教材



大型主机 汇编语言程序设计

王 芬 马光志 编著



清华大学出版社
北京

TP313
W175

内 容 简 介

大型主机汇编语言是一门和大型主机系统架构结合紧密的程序设计语言。本书系统地介绍了基于大型主机的汇编语言程序设计方法和技术。主要包括主机汇编语言的寻址方式、汇编语言格式与汇编语言程序结构、输入输出程序设计、十进制运算、分支程序设计、定点二进制计算、循环程序设计、位操作程序设计、模块化编程等内容。本书能帮助读者进一步认识和了解大型主机技术,增强应用开发的能力,并进一步掌握使用主机汇编语言解决实际问题的能力。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

大型主机汇编语言程序设计/王芬,马光志编著. —北京:清华大学出版社,2012.5

ISBN 978-7-302-28004-0

I. ①大… II. ①王… ②马… III. ①大型计算机—汇编语言—程序设计 IV. ①TP313

中国版本图书馆 CIP 数据核字(2012)第 019877 号

责任编辑:焦虹 李玮琪

封面设计:傅瑞学

责任校对:梁毅

责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印装者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:19.5

字 数:486 千字

版 次:2012 年 5 月第 1 版

印 次:2012 年 5 月第 1 次印刷

印 数:1~3000

定 价:33.00 元

产品编号:035663-01

序 言

从早期的人类登月计划到现今繁荣的银行业,大型主机都是不可或缺的功臣。

从 360 时代“360 度全方位服务”的大型机,到 390 年代“面向企业”的大型服务器,直至如今电子商务时代的“不中断、不宕机的 z”主机,IBM 公司在其主机发展的历史长河中,始终秉承了 On Demand(按需应变)的宗旨,以“虚拟化、开放与创新”为指导,以超乎想象的能力面对各种严峻的挑战,甚至是生存危机,从而不断焕发出新的生命力。这就是大型主机永葆青春的秘诀。

当今,大型主机在降低基础设施复杂性,降低数据中心占地空间,确保信息安全以及降低数据中心能源和电力成本方面又有新的创新与发展,它已成为全新企业级数据中心的理想引擎,大型主机的用户数量也在不断增多。大型主机的应用不仅在传统的政府、金融、海关、税务、铁路等领域继续发挥着威力,而且在保险、医疗、制造、物流、网络、手机、游戏娱乐等领域也展现了新的魅力。

我们已经看到,市场对那些有能力承担大型主机开发、编程或创新任务的学生有着巨大的需求。对这些学生而言,大型主机俨然已成为他们职业生涯的基础。

1997 年,IBM 公司与国家教育部签署了捐赠四台大型主机的协议,并选择北京大学、复旦大学、华中科技大学(原华中理工大学)和华南理工大学作为受赠高校。四所高校在获得了当时最先进的计算机系统后,借鉴国外高校的经验,在国内高校的计算机学院设置了大型主机专业方向,并在大型主机的人才培养和科研上进行了探索。在课程体系的建设上,四所高校根据各自的特点,编写了相关教材,对本科生、研究生开设了专业课程。

随着 IBM 公司技术的创新发展、全球经济化的持续深入以及我国软件产业和全球外包业的发展,需要更多的大型主机专门人才,教材更新的需求也越来越明显。2005 年,IBM 公司再次与华中科技大学、同济大学、华南理工大学、成都电子科技大学、四川大学、大连理工大学、大连交通大学及北京大学八所高校的计算机学院与软件学院联合启动了新的大型主机人才培养

合作项目,并在上海、广州、成都、大连和武汉设立了五个区域性的系统教育中心,配备了先进的大型主机和配套设备,从而使我国大型主机教学环境紧跟大型主机技术的发展前沿;2008年又新发展了云南大学、重庆大学、东北大学、东南大学及深圳大学等五所大学。预计今后两年,该项目将推广到20所大学。另外,参加合作项目的高校共同成立了“大型主机专业方向教学指导工作组”,定期开展大型主机教学交流活动,推进大型主机专业方向的教学改革。

令人高兴的是,IBM大学合作部组织各校教学骨干教师编写、出版了该套系列教材,其特色如下:

系统性:该套教材针对大型主机的基本概念、硬件体系结构、主机系统管理基础、主机应用系统开发工具及环境等几个主要方面进行较为系统、全面的阐述,并对最新的大型主机技术进行了详细的介绍,同时也介绍了主机操作的基本知识和主机操作系统的相关概念。

适用性:大型主机代表着一种信息处理方式,因此,大型主机技术不仅是计算机专业的研究方向,也是金融信息工程专业的研究课题。该套教材紧密结合当今大型主机应用的现状,紧紧围绕大型主机的最新关键技术进行重点介绍,同时注重系统操作实践,在向读者阐明相关基本概念和理论的同时,也介绍了主机操作的基本知识和操作实例。因此它不仅适于作为高校计算机专业课程的教材,而且是大型主机系统操作、管理人员和程序设计人员不可缺少的学习资料,对于从事主机外包业务的专业人员来说,也有很好的指导意义。

时代性:随着网络的发展,大型主机也在不断发展。现今的时代是知识经济时代,是信息爆炸的时代,更是大型主机推陈出新的时代。随着大型主机的发展,越来越多的新技术、新方案、新名词层出不穷。该套教材紧随时代发展的潮流,将大型主机相关的新技术、新概念、新方案融合在教材中,使读者能够快速了解目前大型主机的最新发展态势,成为时代的弄潮儿。

本套教材包括了学习大型主机技术的一些基础和入门课程,可以满足高等院校及企事业单位的开发工程师和维护管理人员教学与自学的需要。

我们相信,这套教材的出版,将进一步推动大型主机系统的产学研互动,提升大型主机从业人员的专业技能,提高大型主机的推广应用水平,为我国各项社会事业的繁荣发展做出更大的贡献。

中国计算机用户协会常务理事

中国计算机用户协会 IBM 用户大会组委会主席

中国机械工业联合会副会长

国家机械工业局经济信息中心主任



前言

自 2000 年以来,计算机人才市场逐渐开始有了一个新的方向的人才需求:大型主机专业人才需求。这种专项人才需求源于全国范围内的软件外包产业的兴起和不断增长。随着市场对大型主机人才需求的增加,国内进行主机人才培养的高校在不断增加,目前已有包括华中科技大学、同济大学、华南理工大学、成都电子科技大学、大连理工大学等高校在内的十数所高校开设了主机课程。主机教学的课程体系和教材也在不断完善中。大型主机汇编语言课程作为主机教学课程体系中的基础课程,一直缺乏合适的中文课程教材。为了更好地开展大型主机汇编语言课程的教学工作,特编写本教材。

本书共分 14 章,具体内容组织如下:

第 1 章 概述。从语言级别的角度介绍汇编语言的功能、特点及常用应用;讲述了汇编语言程序的运行过程,并对与主机汇编语言关系紧密的主机硬件部件进行介绍。

第 2 章 数据表示。简要介绍基本的进制系统和进制数字的运算方法,并进一步介绍主机中的字符数据和数值数据的表示方法。

第 3 章 汇编语言的编码和结构。详细介绍主机汇编语句基本字符集、编码规范以及表达式的构成。

第 4 章 汇编语言程序结构。以一个简单的主机汇编语言程序为例,介绍主机汇编语言程序的组成部分和运行过程。该章中还对定位计数器、寻址方式、地址处理指令、打印控制语句进行较详细的介绍。

第 5 章 数据及存储区域定义。通过多个例子详细介绍主机汇编语言中用于数据及存储区域定义的指令。

第 6 章 寻址方式及基本指令。介绍主机汇编语言指令的寻址方式及常用指令类型。针对不同的指令类型,详细介绍其指令格式和编码格式及常用的指令。该章中对指令的介绍使用了多个例子进行讲述。

第 7 章 输入输出程序设计。详细介绍使用主机汇编语言进行输入输出的编程方式以及相应的宏指令。讲述过程中以一个例子贯穿全章,最终给出一个完整的输入输出程序实例。

第 8 章 十进制运算。以十进制的应用问题为引子,详细介绍数值的十

进制表示格式、十进制运算指令、十进制格式的输出编辑指令以及十进制与二进制相互转换的指令。本章还给出了使用十进制指令解决应用需求的两个程序实例。

第9章 分支程序设计。介绍实现分支程序设计的比较指令和分支指令,并进一步介绍分支和循环结构编程方法。

第10章 数据操作指令。通过多个实例详细介绍数据载入和存储指令以及数据传递指令。

第11章 定点二进制计算。详细介绍定点二进制运算的加、减、乘、除指令,并给出了使用定点二进制运算指令解决应用问题的案例。

第12章 循环程序设计。介绍迭代循环程序设计方法、地址运算指令以及数组处理的方法。

第13章 位操作程序设计。介绍常用的布尔运算指令、移位操作指令以及位测试指令。

第14章 模块化编程。通过实例详细介绍模块化编程的实现方法以及虚拟段的概念及使用。

本书附录中包含 IBM EBCDIC 编码表、扩展助记符表、常用指令表以及书中多个程序的汇编清单。各章后均附有丰富的习题,习题包括填空题、选择题、问答题以及编程题等多种题型,供读者进行复习回顾。

本书的第4、8、13章由王芬和马光志共同编写,其他章节由王芬编写。

大型主机汇编语言是主机程序设计和系统管理的基础语言之一。本书即适合作为普通高校、高职计算机专业课程和金融信息专业课程的教材,也可作为主机系统管理员、主机应用开发人员的参考资料。

本书编写过程中得到了 IBM 大学合作部的邱晓平女士、李晶晖女士、万泽春先生等的大力支持;还有华中科技大学教务处、计算机学院、软件学院以及 IBM 技术中心领导和各位老师的配合和鼓励。王亮、陈岩、黄伟明、蔡斌、童围、秦明康、张风雷、王柳峰、赵恩、杨超、梁永雄、沈兴华、周昌凤、李杨瑞、黄肖群等同学作了大量的翻译和程序测试工作。借此机会向所有支持和参与此项工作的同志们表示衷心的感谢!

特别感谢黄晓涛老师,本书的最终成稿得益于她的不断鼓励和支持。

特别感谢黄小平先生,在本书的编写过程中给予了专业的建议和详尽的指点,在此深表感谢!

特别感谢孙肖连老师、吴驰老师,感谢他们始终如一的支持与鼓励。衷心感谢我的家人和朋友,他们的支持使得本书得以顺利完成。

本书获得了“IBM 大学合作项目书籍出版资助计划”的资助。

由于编者水平有限、时间仓促,书中难免存在不足之处,恳请读者批评指正。

编 者

2011 年 11 月

华中科技大学

目 录

第 1 章 概述	1
1.1 汇编语言的级别	1
1.1.1 机器语言	1
1.1.2 汇编语言	1
1.1.3 高级语言	2
1.2 汇编语言的特点	3
1.3 汇编语言的应用	4
1.4 汇编语言程序模块	4
1.5 汇编程序与汇编语言程序	5
1.5.1 汇编程序	5
1.5.2 汇编语言程序的运行过程	5
1.6 硬件结构	6
1.6.1 存储系统	6
1.6.2 寄存器	7
习题	8
第 2 章 数据表示	10
2.1 数字进制系统	10
2.1.1 数字进制系统概述	10
2.1.2 基数	10
2.1.3 二进制数	11
2.1.4 十六进制数	11
2.1.5 进制的转换	11
2.2 进制数字的运算	13
2.2.1 二进制和十六进制加法	13
2.2.2 二进制和十六进制减法	14
2.3 数据表示	14

2.3.1 字符数据的表示形式	14
2.3.2 数值数据的表示形式	15
习题	17
第3章 汇编语言的编码和结构	19
3.1 基本字符集	19
3.2 汇编语句的编码规范	19
3.2.1 语句域与汇编指令语法	20
3.2.2 续行指示域和续行	21
3.2.3 标识序列域	22
3.2.4 注释的使用	22
3.3 符号	22
3.3.1 符号的格式及使用	22
3.3.2 使用符号的优点	23
3.3.3 符号等价语句 EQU	24
3.4 项及表达式	25
习题	26
第4章 汇编语言程序结构	27
4.1 汇编语言程序例子	27
4.2 汇编语言程序的组成部分	28
4.2.1 控制段	28
4.2.2 汇编语言程序结构	28
4.2.3 程序的开始和结束	30
4.2.4 程序段语句	31
4.3 汇编语言程序的运行过程	32
4.4 定位计数器	33
4.5 寻址方式	34
4.5.1 操作数寻址方式	34
4.5.2 建立寻址能力	35
4.5.3 静态重定位和动态重定位	35
4.6 地址处理指令	36
4.6.1 USING 语句	37
4.6.2 DROP 语句	38
4.6.3 ORG 语句	38
4.6.4 CNOP 语句	38
4.7 打印控制语句	39
4.7.1 标题语句 TITLE	39
4.7.2 换页语句 EJECT	39

4.7.3 空行语句 SPACE	39
4.7.4 打印方式语句 PRINT	40
习题	40
第5章 数据及存储区域定义	41
5.1 存储区域定义	41
5.1.1 DS 语句的格式	41
5.1.2 长度修饰符与长度属性	42
5.1.3 地址计数器	43
5.1.4 重复因子的使用	44
5.2 常量定义	45
5.2.1 常量定义指令(DC)	45
5.2.2 字符常量的定义	45
5.2.3 十六进制常量的定义	45
5.2.4 二进制常量的定义	46
5.2.5 分区十进制常量的定义	46
5.2.6 压缩十进制常量的定义	47
5.2.7 定点数的定义	47
5.2.8 地址常量的定义	48
5.3 自由量	49
习题	51
第6章 寻址方式及基本指令	53
6.1 寻址及机器指令	53
6.1.1 寻址方式	53
6.1.2 指令格式	53
6.1.3 逻辑地址	54
6.1.4 S/390 指令类型	56
6.2 RR 指令格式	57
6.2.1 指令格式	57
6.2.2 编码格式	57
6.2.3 AR 指令	57
6.2.4 SR 指令	58
6.2.5 LR 指令	58
6.3 RX 指令格式	59
6.3.1 指令格式	59
6.3.2 编码格式	59
6.3.3 L 指令	59
6.3.4 A 指令	60

6.3.5	S 指令	60
6.4	RS 指令	61
6.4.1	指令格式	61
6.4.2	编码格式	61
6.4.3	LM 指令	61
6.4.4	STM 指令	62
6.5	SI 指令	62
6.5.1	指令格式	62
6.5.2	编码格式	62
6.5.3	移动立即数指令 MVI	63
6.5.4	立即数逻辑比较指令 CLI	63
6.6	SS 指令	64
6.6.1	等长的 SS 指令格式和编码	64
6.6.2	移动字符串指令 MVC	65
6.6.3	逻辑比较指令 CLC	66
6.6.4	不等长的 SS 指令格式和编码	66
	习题	68

第 7 章 输入输出程序设计 71

7.1	输入输出处理	71
7.2	宏指令	72
7.2.1	宏指令简介	72
7.2.2	输入输出系统宏指令	73
7.3	数据集定义	73
7.3.1	DCB 宏的格式	73
7.3.2	DCB 宏的关键字参数	74
7.3.3	DCB 宏指令示例	75
7.4	数据集的打开和关闭	77
7.4.1	OPEN 宏指令	77
7.4.2	CLOSE 宏指令	78
7.5	数据集的移动	78
7.5.1	MOVE 模式	78
7.5.2	GET 宏	79
7.5.3	PUT 宏	80
7.5.4	样例程序	80
7.6	其他的宏指令	81
7.6.1	CNTRL 宏指令	81
7.6.2	RELSE 宏指令	81
7.6.3	FEOV 宏指令	82

7.7 一个完整的输入输出操作程序·····	82
习题·····	83
第8章 十进制运算·····	85
8.1 十进制应用问题·····	85
8.1.1 应用问题·····	85
8.1.2 十进制运算的优点·····	86
8.2 分区和打包十进制格式·····	86
8.2.1 分区十进制格式·····	86
8.2.2 打包十进制格式·····	90
8.3 打包和解包指令·····	92
8.3.1 问题引入·····	92
8.3.2 指令格式·····	93
8.3.3 打包指令 PACK·····	94
8.3.4 解包指令 UNPK·····	95
8.3.5 使用 PACK 和 UNPK 指令解决问题·····	97
8.4 十进制加减运算·····	99
8.4.1 问题引入·····	99
8.4.2 十进制加法指令 AP·····	101
8.4.3 十进制减法指令 SP·····	102
8.4.4 清零并加十进制指令 ZAP·····	102
8.4.5 比较十进制指令 CP·····	103
8.5 十进制乘除运算·····	104
8.5.1 十进制乘法指令 MP·····	104
8.5.2 十进制除法指令 DP·····	105
8.6 输出编辑·····	107
8.6.1 ED 指令·····	107
8.6.2 EDMK 指令·····	116
8.7 十进制移位操作·····	122
8.8 十进制与二进制之间的转换·····	123
习题·····	125
第9章 分支程序设计·····	131
9.1 分支程序设计——应用问题·····	131
9.2 条件转移的实现·····	131
9.3 比较指令·····	133
9.3.1 算术比较和逻辑比较·····	133
9.3.2 算术比较指令·····	133
9.3.3 逻辑比较·····	137

9.4 分支指令	141
9.4.1 条件分支指令 BC	141
9.4.2 寄存器条件转移指令 BCR	143
9.4.3 扩展助记符指令	144
9.5 分支和循环结构编程	145
9.5.1 IF-THEN 结构的实现	145
9.5.2 IF-THEN-ELSE 结构的实现	146
9.5.3 DO-WHILE 结构的实现	146
习题	147
第 10 章 数据操作指令	149
10.1 载入和存储指令	149
10.1.1 载入全字指令 L	149
10.1.2 载入半字指令 LH	149
10.1.3 多寄存器载入指令 LM	150
10.1.4 寄存器载入指令 LR	150
10.1.5 载入测试指令 LTR	151
10.1.6 存储全字指令 ST	151
10.1.7 存储半字指令 STH	152
10.1.8 多寄存器存储指令 STM	152
10.1.9 插入字符指令 IC 和存储字符指令 STC	153
10.1.10 屏蔽插入字符指令 ICM 和屏蔽存储字符指令 STCM	153
10.2 传递指令	155
10.2.1 传递字符指令 MVC	155
10.2.2 传递立即数指令 MVI	155
10.2.3 长移动指令 MVCL	156
习题	158
第 11 章 定点二进制计算	161
11.1 定点运算的数据表示	161
11.2 定点加法	162
11.2.1 寄存器加指令 AR	162
11.2.2 全字加指令 A	162
11.2.3 半字加指令 AH	164
11.3 定点减法	165
11.3.1 寄存器减指令 SR	165
11.3.2 全字减指令 S	166
11.3.3 半字减指令 SH	166
11.4 定点乘法	166

11.4.1	寄存器乘法指令 MR	166
11.4.2	全字乘指令 M	167
11.4.3	半字乘指令 MH	168
11.5	定点除法	171
11.5.1	寄存器除法指令 DR	172
11.5.2	全字除指令 D	173
11.5.3	应用实例	174
	习题	176
第 12 章	循环程序设计	182
12.1	迭代循环程序设计	182
12.1.1	计数循环指令 BCT	182
12.1.2	BCTR 指令	183
12.2	地址运算	184
12.3	数组处理	185
12.3.1	数组的定义	185
12.3.2	数组的处理技术	186
	习题	188
第 13 章	位操作程序设计	191
13.1	布尔运算	191
13.1.1	RX 型的布尔操作	192
13.1.2	RR 型的布尔操作	194
13.1.3	SI 型的布尔操作	195
13.1.4	SSI 型布尔操作	197
13.2	移位操作	198
13.2.1	移位类型	198
13.2.2	逻辑移位指令	199
13.2.3	算术移位指令	201
13.3	位测试指令	206
	习题	208
第 14 章	模块化编程	210
14.1	主程序与子程序	210
14.1.1	主程序与子程序的概念	210
14.1.2	主程序和子程序的连接过程	211
14.1.3	外部子程序的编译连接过程	211
14.2	子程序设计	212
14.2.1	子程序的结构	212

14.2.2	子程序的调用与返回·····	213
14.2.3	主程序和子程序的实现·····	216
14.3	与高级语言编程模块间的连接·····	231
14.4	虚拟段·····	233
14.4.1	虚拟段的定义·····	233
14.4.2	虚拟段的使用·····	234
14.4.3	虚拟段程序示例·····	235
习题	·····	238
附录 A	IBM EBCDIC 编码表·····	243
附录 B	扩展助记符表·····	245
附录 C	常用指令表·····	246
附录 D	程序 8-1 汇编后的程序列表·····	249
附录 E	程序 8-2 汇编后的程序列表·····	258
附录 F	程序 11-1 汇编后的程序列表·····	269
附录 G	程序 11-2 汇编后的清单·····	276
附录 H	程序 13-1 的汇编清单·····	289

概 述

汇编语言是计算机所提供的一种最快且最有效的语言,也是能够利用计算机所有硬件特性并能直接控制硬件的低级语言,它随计算机结构的不同而不同。大型主机汇编语言是一门和主机硬件结构结合紧密的程序设计语言。本章从大型主机汇编语言程序设计的角度出发,介绍相关的基础知识。

1.1 汇编语言的级别

计算机是由程序控制且受人类操控,能完成结账、文本处理、游戏、医疗诊断等多种功能的设备。程序是指令的集合。程序开发人员将解决问题的步骤编写成由一条条指令组成的程序,输入到计算机的存储设备中。计算机执行指令序列并完成预定任务。

人类使用计算机语言与计算机进行交流并对计算机进行控制。计算机语言按照级别由低到高分三类:机器语言、汇编语言和高级语言。

1.1.1 机器语言

计算机是由机器语言构成的程序控制。机器语言由二进制的 1 和 0 构成的字符串组成。机器语言指令包括表示要执行的操作的代码(即操作码)以及程序要执行的数据的地址(即地址码)两个部分,如图 1-1 所示。

操作码	地址码
-----	-----

图 1-1 机器语言指令的组成

机器语言是由二进制代码组成、完全面向机器的指令序列。用机器语言编写的程序称为机器语言程序,又称为目标程序。程序由指令和含有不同数据类型的数据区域组成。

图 1-2 展示了机器语言程序执行的过程。当程序执行时,首先是第一条指令被计算机的 CPU 进行处理,然后,后续指令顺次由 CPU 执行。一旦指令被执行,数据可能从一个数据区域被复制或者移动到另一个数据区域。

虽然机器语言便于计算机识别,但对于人类却晦涩难懂,所以后来又出现了汇编语言。

1.1.2 汇编语言

在早期程序员用机器语言编写程序。这是个缓慢、枯燥且容易出错的过程,仅限于小而简单的程序。为了简化编程任务并使编写大的程序成为可能,汇编语言诞生了。汇编语言程序允许编程人员使用汇编语言。虽然汇编语言程序有与其对应的机器语言程序基本相同的指令个数,但编写汇编语言程序更加简单。

汇编语言用自然符号来代替二进制指令代码。机器指令的操作码部分在汇编语言中由助记符(例如助记符 MVC 是 MOVE 的含义)表示,地址是由符号代码——标志符(例如 PLACE1)表示。同时,汇编语言还允许在指令前冠以标号,代表该指令的存放地址。汇编语言的主要操作与机器指令基本是一一对应的。

由于计算机只能执行机器语言程序,因此,汇编语言程序是不能直接被计算机所识别的。它被执行之前,必须将汇编语言程序翻译为由机器指令组成的机器语言程序后才能被执行。这个功能是由汇编器完成的。汇编器是一个机器语言程序。它将由自然符号表示的汇编语言程序转换成由二进制 0 和 1 组成的机器语言程序。转换后的机器语言程序方可执行。

虽然编写汇编语言程序是个繁琐的过程,但是相对于编写机器语言,还是会简单很多。汇编语言程序必须精确而详细地指出计算机需要做什么,以完成整个编程任务的每个部分。例如在 COBOL 语言中,两个数相加的过程可以用一条命令表示:

```
ADD A TO B
```

两个数相加的过程往往被看做一个单步操作。但是,这个过程在汇编语言编程者看来是一个操作序列:将第一个值写累加器;加第二个值至累加器;然后将累加器的值存储至第三个域。相对应的大型主机汇编语句序列如下:

```
PACK X,A (机器语言指令 11110010...)
PACK Y,B (机器语言指令 11110010...)
AP X,Y (机器语言指令 11111010...)
UNPK B,X (机器语言指令 11110011...)
```

也就是说,使用 COBOL 语言的一条指令完成的两数相加功能,在大型主机汇编语言中需由多条指令组成的序列来实现。

1.1.3 高级语言

高级语言相比汇编语言更进一步地简化了编程任务。在高级语言中,程序由更高级的抽象语言表达。同样的一个程序,用汇编语言编写所需的指令比用高级编程语言编写的指令多五倍以上甚至十倍。

如图 1-3 所示的 COBOL 语言为高级语言,在 COBOL 语言中将两个数相加的过程被看做为一个单步操作,由一条 COBOL 指令实现。但是,这个过程在汇编语言编程者看来是一个操作序列:将第一个值写累加器,加第二个值至累加器,然后将累加器的值存储至第三个域。这个操作序列在本样例中使用了 4 条汇编语句才得以实现。

计算机硬件只能执行机器语言,因此,高级语言需要通过编译器转换为机器语言程序后才可以被执行。编译器是一种机器语言程序。

下面是一些高级语言:

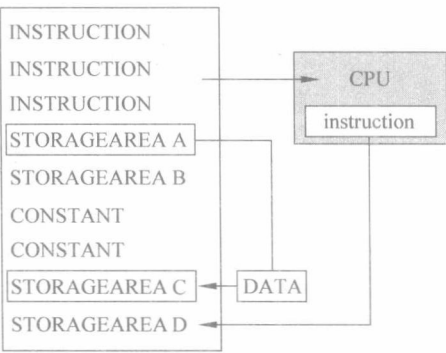


图 1-2 机器语言程序的执行