

轻松学习 C 程序设计

——揭开计算机与程序设计的奥秘

何 勤 著

从梦想到实现，你还需要多久的等待？

成功之路，就在手中！



中国电力出版社
www.cepp.com.cn

TP312/2984

2008

轻松学习 程序设计

——揭开计算机与程序设计的奥秘

何 勤 著

责任编辑：何勤
封面设计：何勤
版式设计：何勤

本书可作为高等院校计算机专业及相关专业的教材，也可供从事计算机工作的工程技术人员参考。

中国电力出版社出版

地址：北京市西城区三里河路
邮编：100044
电话：(010) 68459011

发行：中国电力出版社

开本：185mm×260mm

印张：12.5

字数：200千字

印次：2008年8月

印数：0001—4000

定价：32.00元



中国电力出版社

www.cepp.com.cn

北京 三里河路

内 容 提 要

本书是一本能让大家真正学会编程的程序设计入门书。全书通过对难度逐渐递增的大量例题的深入讲解,介绍了计算机的编程思想、基本算法和基本技巧。本书通过引入一个理想的厨房系统进行类比,系统地讲解了计算机硬件和操作系统中一些极重要的核心知识,具体包括计算机硬件的基本结构、指令和程序执行的全过程、中断、进程、进程调度、输入/输出、虚拟内存等。通过对这些知识的学习,使读者对计算机运行的机制和奥秘有个清楚的了解。

本书内容丰富、独具特色,非常适合各高等院校计算机及其相关专业C语言程序设计入门课的教材或教学参考书,也适合作为各类程序设计辅导班的培训教材。

图书在版编目(CIP)数据

轻松学习C程序设计:揭开计算机与程序设计的奥秘/何勤著.—北京:中国电力出版社,2008
ISBN 978-7-5083-7432-1

I. 轻… II. 何… III. C语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2008)第113299号

责任编辑:夏华香

责任校对:丁秋慧

责任印制:郭华清

书 名:轻松学习C程序设计——揭开计算机与程序设计的奥秘

作 者:何 勤

出版发行:中国电力出版社

地址:北京市三里河路6号 邮政编码:100044

电话:(010) 68362602 传真:(010) 68316497

印 刷:汇鑫印务有限公司

开本尺寸:185mm×260mm 印 张:24.5 字 数:599千字

书 号:ISBN 978-7-5083-7432-1

版 次:2008年8月北京第1版

印 次:2008年8月第1次印刷

印 数:0001—4000册

定 价:38.00元

敬 告 读 者

本书封面贴有防伪标签,加热后中心图案消失

本书如有印装质量问题,我社发行部负责退换

版 权 专 有 翻 印 必 究

前言

针对不少大学生编程能力很差以及对程序设计课感到非常厌烦和恐惧的现状，笔者在三年前下定决心，动手创作本书。本书的初稿从去年下半年就开始试用，深受大多数学生的喜爱。

对于大学生来说，学习程序设计这门课，能够得到的最大收益是什么呢？难道就是学会一门高级程序设计语言的大量语法知识，甚至包括很多容易过时的、枯燥乏味、令人厌烦的细节语法知识吗？笔者认为不是的。因为毕业以后，大多数大学生并不从事程序设计的工作。即使将来从事编程工作，使用 C 语言进行软件开发的可能性现在也变得较小了（虽然学的人很多）。由此可以得出重要结论，在 C 程序设计入门课中，C 语言所特有的语法细节讲解应当大幅度缩减。

那么，对于大学生而言，程序设计这门课是否本来就不太重要呢？笔者认为不是不重要，而是至关重要！非常重要！极为重要！

究竟什么才是大学生最应该在本课程中学到的重要知识呢？笔者认为应当是 C 程序设计语言的基本语法要素（这是不容易过时的、各种常用的命令型高级语言的公共部分）、基本编程思想、基本编程技巧和基础算法知识，仅运用这部分知识就可以解决大多数常见的编程问题（虽然有时效率不高），而不是大量的语法细节。本书在后面常常把基本编程思想、基本编程技巧和基础算法知识简称为基本编程技术。

究竟什么是基本编程技术？笔者把需要学习《面向对象程序设计》、《数据结构》和《算法分析与设计》这些高级课程后，才能够掌握的那一部分（面向对象的）编程思想、（需要以链表为基础的）中高级编程技巧和（需要以离散数学为基础的那一部分）算法排除在外。剩下的那部分就属于基本编程技术的范畴。

牢固掌握基本编程技术，对于任何一位当代大学生都极为重要。这是因为这些知识的学习、积累和掌握，不仅能够大大提高学生分析问题和解决问题的能力，其作用决不亚于《高等数学》这门课，而且还能够极大地提高学生的严谨、细致、勇于克服困难、考虑问题全面和理论联系实际的现代科学和工程技术的素养。

结论是任何一位当代大学生，尤其是理工科的学生，即使将来不想成为程序员或编程高手，都应当认真学习和努力掌握基本编程技术。这是当代社会和现代科学技术对大学生能力和素质的一项最为基本的要求。

基本编程技术的掌握真的非常困难吗？回想一下笔者自己掌握基本编程技术的经历，发现学会任何命令型高级程序设计语言的语法知识仅需一个月就足够了。但是，真正掌握基本编程技术实在是太漫长了，经历了大约 5 年的时间，认真读了十几本各种各样的书，编写了不少的程序，才感觉到自己渐渐地掌握了基本编程技术。笔者发现，除了少数程序设计竞赛的辅导书以外，在一般的程序设计初级教科书中难以系统地学习到的竟然是编程思路（这一般是由于作者没有把学会编程作为教材的重要目标）；然而是否具有编程思路和技巧，又是一个学生会不会编程的关键因素所在，就像是否具有写作思路是一个人会不会写文章的关键因素所在一样。

难道普通的大学生都得要经过几年时间，独自去钻研、消化大量的相关书籍，才能积累足够多的编程思路并掌握对他们至关重要的基本编程技术（即学会编程，但不是学会高级编程技术）吗？能否把这个学习周期和范围缩短或缩小到仅需要学习一本不算太厚的程序设计入门教

科书呢?对于非计算机专业的大学生而言,这一点太重要了!

编写一本让大学生在很短时间内,就能够积累大量编程思路 and 掌握基本编程技术的程序设计入门教材,成为历史和当代社会赋予高校计算机教师的一个义不容辞的艰巨任务。在意识到重任在肩时,笔者虽然能力有限,但仍义无反顾地投身其中,经过历时两年多呕心沥血般的艰苦创作,本书终于得以与读者见面了。

本书可使读者用一种几乎全新的方式,来轻松有趣而又深入系统地学习基本编程技术。之所以能够做到这一点,是因为本书主要有以下 4 点创新。

(1) 用少量篇幅,通过首创的、极受学生喜爱的理想厨房系统,简明而又系统地(而不是支离破碎地)介绍了计算机和操作系统的基本工作原理,使学生从此对计算机不再感到非常神秘和困惑,为学习高级程序设计语言和本书的其余内容奠定了牢固的硬件和系统软件的基础知识。

(2) 对 C 语言的全部语法内容进行了三级划分。第一级是语法要素,主要是所有命令型高级语言的基本公共语法部分,本书对此级语法内容作了系统详细的讲解;第二级是次常用语法,被有机地结合在各例题后的问题中,处于从属于例题的次要地位(但这样讲解变得生动多了);第三级是较少用到的语法部分,被放在提高部分或干脆被舍弃不讲。这样,除了语法要素和提高部分外,令很多学生深感厌烦和枯燥的对语法细节的大段生硬的讲解,在本书中基本不见了。

(3) 在本书中,凡属于××讲的例题不再是语法讲解的附属品,而是与语法要素并列,处于同等重要的核心地位。例题选用与否的主要标准是,是否有利于讲解各种基本编程技术。在例题的选择上注意兼顾趣味性、实用性和知识性,还特别注意例题的数量和难度的逐步递增。重视使用逐步求精的伪代码来构造求解问题的算法。对基本编程思路和编程技巧深入详细地讲解,在本书中处于核心地位。笔者认为,这才是学生最需要得到教材(和教师)指导的,本课程中比较灵活和难以掌握的重点知识。这也是本书独具的特色和亮点所在。

(4) 在书中最后一章通俗易懂、简明、独创性地讲解了编译程序对高级程序设计语言的要素如何进行编译处理的大致过程,使学生不必经过对《编译原理》这门深奥难懂的高级课程的学习,就可以对高级程序设计语言程序为何能够命令计算机工作有一个比较清楚地认识,消除同学们用高级语言编程时的那种如在沙滩上盖高楼般的、挥之不去的不安感和漂浮感。

本书实际上只讲述了三个相关课题。第一,计算机硬件是如何与操作系统密切配合,来运行机器语言程序的(第 1 章、第 9 章);第二,应当如何编写高级语言源程序(第 2 章~第 8 章),这是本书的主体部分;第三,高级语言源程序是如何被编译程序转换为机器语言程序的(第 10 章)。这三者之间由程序作为主线,相互之间密切配合、融为一体;为初学者编织出了一幅计算机科学比较生动有趣的完整画面。

本书后面章节的少量例题并没有给出完整的 C 程序,而只是给出了伪代码表示的算法,希望大家自己完成。

对于教师,在决定是否选用此书作为教材前,可先浏览第 11 讲、32 讲、35 讲、21 讲、29 讲和第 46~50 讲,然后再看一看第 1 章、第 7 章和第 9.4 节、9.5 节,这些内容最能体现出本书所独具的特色。

如果将本书作为非计算机专业的大专院校程序设计课的教材使用,最后两章到三章,可以作为感兴趣学生的课外读物。书中较难一些的例题可以略去不讲。第 2~5 章的授课时数,要比用通常教材多安排一些(第 4 章和第 5 章的例题可只选讲三分之一左右),才能取得较好的教学效果。第一章也可以仅选少量重要内容进行讲解,而把大部分内容作为学生的自学材料。

感谢中国电力出版社朋友们的大力支持和关心,感谢本书的参与者雷金娥老师,感谢看过此书并提出过宝贵意见的专家和老师。

其他参与本书部分编写工作和辅助工作的人员有雷金娥负责上机指导,王芸负责两个附录,楼明珠负责每章后面的习题,李璠负责对所有例题的调试和测试,刘坤负责全书插图的草稿。

如果读者对本书有什么宝贵的意见和建议,也可以直接与笔者联系。笔者的电子邮箱为:
heqin3785@yahoo.com.cn。

作 者

2008 年 7 月

目 录

前言

第 1 章 从理想厨房到计算机的基本工作原理 (初)	1
1.1 理想厨房系统	1
1.2 理想厨房系统的一个炒菜实例	5
1.3 理想厨房系统的特点	8
1.4 理想厨房系统与计算机系统术语对照表	11
1.5 计算机的基本工作原理 (初)	12
1.6 计算机的基本特点	19
1.7 理想厨房系统与计算机之间的重要区别	22
1.8 本章要点	25
本章习题	26
本章的进一步读物	27
第 2 章 C 语言部分要素简介与顺序结构程序设计	28
2.1 引言	28
2.2 术语简介	31
2.3 C 语言简介	32
2.4 C 语言程序的主要构成成分	32
2.5 C 语言的基本要素	33
2.6 部分表达式语句简介	46
2.7 C 语言中的注释	53
2.8 用计算机求解问题的步骤	53
2.9 逐步求精的算法	54
2.10 一个小型 C 程序的开发过程	54
2.11 用 C 语言编写科学计算类程序的常用格式	63
2.12 常见的编程错误	64
2.13 编程错误分类	64
2.14 提高部分	65
2.15 本章小结 (不包含提高部分)	73
本章习题	74
第 3 章 选择结构程序设计	76
3.1 两种 if 语句	76

3.2 布尔表达式之一: 关系表达式	79
3.3 空语句	82
3.4 复合语句	82
3.5 if 语句的嵌套及其用法	86
3.6 布尔表达式之二: 逻辑表达式	87
3.7 switch 语句	96
3.8 提高部分	101
3.9 选择结构的常见错误	102
本章习题	103
第4章 循环结构程序设计	107
4.1 while 语句	107
4.2 数列求和算法知识简介	111
4.3 do...while 循环语句	122
4.4 break 语句和 continue 语句	129
4.5 for 循环语句	134
4.6 多重循环	138
4.7 常见错误小结	154
本章习题	155
第5章 数组	157
5.1 一维数组引论	157
5.2 一维数组	158
5.3 一维字符数组和字符串	180
5.4 二维数组	189
5.5 编程综合应用例题	201
本章习题	219
第6章 函数	221
6.1 引言	221
6.2 函数概念的概述	222
6.3 函数的编写	224
6.4 用数组作为函数参数	236
6.5 函数的形式参数与实际参数的比较	238
6.6 函数的嵌套与递归	239
6.7 提高部分	248
本章习题	251
第7章 指针	253
7.1 引言	253

7.2 指针变量的定义和初始化	254
7.3 指针变量的运算符	256
本章习题	261
第 8 章 C 语言的其他内容概述	263
8.1 结构	263
8.2 编译预处理	275
8.3 C 语言中的输入/输出	279
本章习题	290
第 9 章 计算机的基本工作原理 (续)	292
9.1 最简单的模型计算机	292
9.2 用机器语言编程举例	297
9.3 与输入/输出有关的概念简介	300
9.4 中断和操作系统进程等概念简介	304
9.5 提高部分	313
本章习题	323
第 10 章 高级语言要素的编译过程简介	324
10.1 引言	324
10.2 用机器语言编程的特点和难点	324
10.3 用汇编语言编程的特点和难点	326
10.4 用高级程序设计语言编程	327
10.5 本章要点	344
10.6 与本章有关的读物	345
附录 A Microsoft Visual C++ 6.0 集成开发环境介绍	347
附录 B 常用字符与 ASCII 码对照表	354
附录 C 常用库函数	356
附录 D 运算符的优先级与结合性	373
附录 E 波利亚的求解方法列表	374
附录 F 计算机在当代社会中的作用	375
后记	380

第1章 从理想厨房到计算机的 基本工作原理（初）

通过研究一本好的菜谱，一个计算机程序员也能学到许多东西（事实上，作者好不容易才没有将本书的书名定为“程序员的烹调技巧”。也许某一天，作者将试图写一本叫作“厨房的算法”的书）。

——Donald E.Knuth 《计算机程序设计艺术》

本章是全书的重要基础。通过本章的学习，可为读者建立起计算机究竟如何工作的整体概念，纠正读者以前对计算机的一些错误认识和看法，并为第2章、第9章、第10章的学习打下坚实的基础。

本章不需要任何预备知识，只要你认得字就行，还有一个比较高的要求：只要你有足够的耐心，这个耐心能确保你把本章认真读两遍。你就能够真正懂得神秘的、到目前为止人类有史以来最伟大、最奇妙而又最复杂的发明——电子数字计算机（当然这个伟大发明汇集了几代人中很多杰出人士的聪明才智和研究成果）是如何工作的基本原理。为达此目的，笔者付出了极大的努力，找到了一种比较好的比喻方法——理想厨房系统。由此开始真正理解计算机工作原理的、激动人心的探索旅程。将要探索的是一个看起来很不起眼的，在有些场合被称为计算机（或电脑）的电子设备，是如何具有如此神奇的魔力和智能的奥秘所在。这种神奇的魔力和智能正在彻底地、快速地、默默无闻或者令人吃惊地改变和丰富人们所生活的大千世界。

本章将从一个理想厨房系统的例子入手，讲解计算机工作的基本原理和特点。这一叙述比较长，但和厚厚的整本计算机原理教材比起来，这里的叙述为了突出重点、减少理解难度而做了极大的简化，也是值得读者认真学习和细心领会的。

但如果你是一个急性子，可以快速地浏览下面用小号字编排的几节，重点要仔细领会第1.2节和第1.3节的内容，即理想厨房的炒菜实例和理想厨房系统的特点。

1.1 理想厨房系统

理想厨房系统是一个通过执行菜谱，把原料加工成菜肴的系统。

理想厨房系统由三个主要部件构成，包括理想厨房、自动冰箱和三条传送带。这三条传送带位于理想厨房和自动冰箱之间，分别为地址传送带、控制传送带和材料传送带（传送原料、菜肴或加工步骤）。三条传送带负责理想厨房与自动冰箱之间的通信控制及物品传送。

注意 术语“理想厨房系统”与“理想厨房”是有区别的，理想厨房只是理想厨房系统中的一部分。

理想厨房系统的构成如图 1.1 所示。



图 1.1 理想厨房系统的构成

1.1.1 自动冰箱

1. 自动冰箱的作用

自动冰箱分为大小相等的多个格子。在这些格子中，一格一格地从前往后依次存放某道菜的全部加工步骤。每一个加工步骤占据冰箱的一格，这道菜要用到的所有原材料也都事先存放在冰箱的格子中，每种原材料也都占用一格（是不是很奇妙？菜谱竟然和原材料统一存放在冰箱里）。此外，还要预先为将要炒好的菜预留存放的格子。为了准确定位在哪一格存取物品，冰箱的每格都有一个唯一的编号，称为冰箱（格子）的地址，这个编号由 0 开始，如果冰箱总共有 1000 格，那么此冰箱（格子）的地址编号就是从 0 到 999 号。

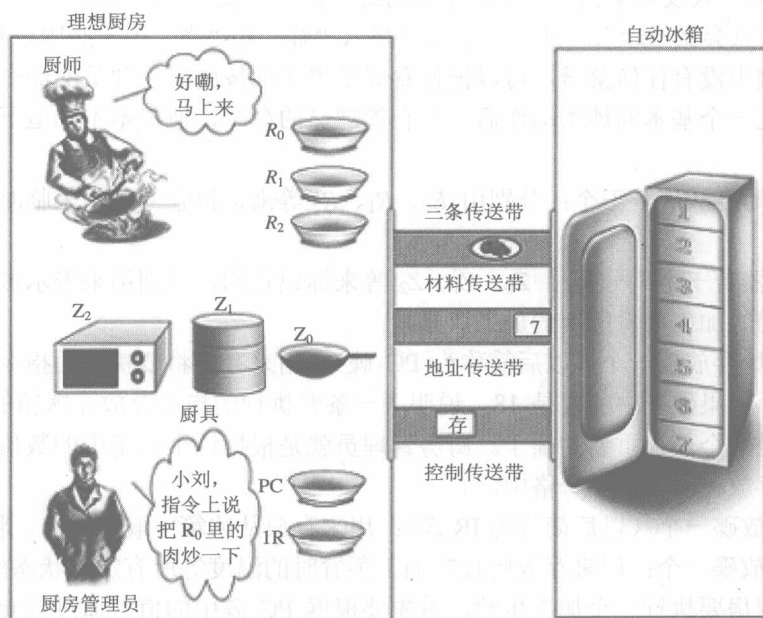
注意 所有的加工步骤是从小到大依次连续存放在冰箱的格子中。例如，4 号格中存放的加工步骤，一般来说，就是紧接着 3 号格中存放的加工步骤做完后，马上要做的下一个加工步骤。冰箱的所有格都一样大，每一格都可以存放一种食物或原料，也可以用来存放任何菜谱的一个加工步骤。

2. 自动冰箱的工作机制

自动冰箱从控制传送带上接收由理想厨房传到冰箱的“存”或“取”命令信号，然后从地址传送带接收由理想厨房传来的要存取的冰箱格子的地址，由此确定存取物品的位置。自动冰箱根据这两个信号进行相应地址的冰箱格子中的物品存取操作。例如，如果得到的是来自理想厨房的“存”信号和地址信号 5，则自动冰箱要从材料传送带上取下从理想厨房传来的物品，并送到相应地址为 5 的冰箱格子中存放；如果得到的是来自理想厨房的“取”信号和地址信号 3，则自动冰箱将地址为 3 的格子中的物品取出，放到材料传送带上，利用此传送带将物品传送到理想厨房。

注意 术语“物品”既可以是原料或食物，也可以是菜谱的一个加工步骤。

理想厨房系统的运行示意图如图 1.2 所示。



1.1.2 三条传送带

(1) 地址传送带：负责将理想厨房要存取冰箱哪一格的地址信号，由理想厨房传送到自动冰箱，这是一条单向传送带。

(2) 控制传送带：负责将理想厨房要到冰箱“存”或“取”的信息，由理想厨房传送到自动冰箱，目前这也是一条单向传送带（但在第 9.4 节要进行改进）。

(3) 材料（以及加工步骤）传送带：这是一条双向传送带，它负责从自动冰箱到理想厨房传送加工步骤（即指令）或要加工的原材料，以及从理想厨房到自动冰箱传送炒好的菜或半成品。

1.1.3 理想厨房

理想厨房本身具有加工功能（由于有厨师）和控制功能（由于有厨房管理员）。理想厨房是理想厨房系统的核心部件。

注意 为了叙述简洁，后面有时会用指令这个术语来代替加工步骤，因为加工步骤就是指示理想厨房如何工作的命令。

(1) 理想厨房由以下几部分构成。

1) 厨房管理员一人：负责总体控制（发出相关命令取一条指令，分析此指令，然后为了执行此指令，向自动冰箱或厨师等发出相关的控制命令）。

2) 炒菜厨师一人及炒菜厨具一套: 在得到厨房管理员发来的命令后, 负责执行炒菜的具体加工步骤。厨师只会做“炒”、“煎”、“蒸”、“烤”、“炸”和“煮”等一些固定数量的基本操作, 炒菜厨师的大脑中没有任何菜谱, 可以把他看成是忠实地按厨房管理员的命令一步一步进行工作的厨师, 做完一个基本的炒菜动作后, 没有管理员的命令, 他根本不知道下一步炒菜动作该如何进行。

3) 通用临时存放碟若干个: 分别用 R_0 、 R_1 、 R_2 等来标记它们, 用来临时存放原料、半成品或成品。

4) 专用容器若干个: 分别用 Z_0 、 Z_1 、 Z_2 等来标记它们, 分别用来表示炒菜的炒菜锅、蒸菜的蒸笼、烤箱等加工菜肴所用的加工炊具。

5) 指令地址存放碟一个(以后简称为 PC 碟): 用来存放将要执行的指令在冰箱中所存放的地址。例如, 如果碟子中的数是 18, 说明下一条要执行的指令存放在冰箱的地址为 18 的那个格子中。这是一个最为重要的碟子。厨房管理员就是根据这个碟子中的数值, 才知道下一条要执行的指令存放在冰箱的哪一格中。

6) 指令存放碟一个(以后简称为 IR 碟): 用来存放从冰箱中取出来的、将要执行的指令。

7) 状态存放碟一个: 用来存放厨具在加工菜肴时的温度等所有加工状态的数据。

(2) 理想厨房要执行一个加工步骤, 首先要根据 PC 碟中的值, 通过三套传送带, 到自动冰箱的那个格子中去取这个加工步骤; 取到后将其放到理想厨房的 IR 碟中, 然后分析此加工步骤, 并根据此加工步骤的要求, 做以下 5 类加工步骤中的一种。

1) 通过三套传送带, 命令冰箱把指定地址格子中的(炒菜马上要用到的)原材料传送到理想厨房中, 即取物品。

2) 命令厨师按照指令(即加工步骤)的要求, 对原材料(做“炒”、“蒸”、“煮”等基本操作步骤中的一个炒菜动作)做一个加工动作, 即加工。

3) 通过三套传送带, 把成品(或半成品)送回到冰箱指定格子中存放, 即存物品。

4) 根据某个状态条件, 不按正常顺序取并且执行下一条指令, 而是根据此指令中给出的(冰箱格子的)附加值, 修改 PC 碟中的值, 然后跳转到那里去取, 并且执行下一条指令, 即跳转。

5) 在厨房的各个碟子或炊具之间传送原料和半成品, 即内部传送。

1.1.4 理想厨房系统的工作方式

理想厨房执行一条指令(即一个加工步骤)的整个周期可划分为两个阶段, 并按如下的方式工作。

(1) 第一阶段——取指令阶段: 厨房管理员根据指令地址存放碟(PC 碟)中的那个地址号, 将这个号码复制, 并将此地址号放到地址传送带上传向冰箱; 然后厨房管理员立刻从控制传送带向冰箱发一个“取”命令信号, 冰箱得到这两个信号后, 把即将要执行的指令从冰箱的指定格子中复制出来, 将复制的指令放到材料传送带上; 到达理想厨房后, 此指令被存放到指令存放碟(IR 碟)中, 要执行的指令就这样取到理想厨房; 然后指令地址存放碟中的地址号自动增加 1, 为取下一条指令做好准备。这一阶段又被称为取指周期。任何指令的执行, 在这一阶段都是相同的。

(2) 第二阶段——分析、执行指令阶段: 这一阶段与上一阶段不同, 不同类型的指令, 其工作方式是不同的。由于有 5 大类指令(加工步骤), 分别描述其工作机制比较烦琐。这里不再

详细介绍，留给读者作习题。

注意 这里暂时不考虑如何输入菜谱和原料进入理想厨房系统的冰箱，也暂时不考虑如何从理想厨房系统的冰箱输出菜肴等问题（这些问题参见第9章）。

下面通过一个实例来讲解理想厨房系统的工作机制，这是本章的一个重点。

1.2 理想厨房系统的一个炒菜实例

下面编写一道香菇炒菜心的特殊菜谱，并把它放在自动冰箱中，把经过初加工的香菇和菜心放在自动冰箱合适的位置，然后启动理想厨房进行炒菜工作。通过这个实例查看是否可以命令这个看起来很奇怪的理想厨房系统炒好这道菜。

首先，把香菇放在冰箱地址为5的格子中，把菜心放在冰箱地址为6的格子中，冰箱地址为7的格子预留给炒好的菜使用。

菜谱从冰箱地址0号格开始存放，编写的香菇炒菜心的菜谱如下：

- (1) 在地址0的格子中，取地址5（中的物品）到 R_0 碟。
- (2) 在地址1的格子中，取地址6（中的物品）到 R_1 碟。
- (3) 在地址2的格子中，将 R_0 碟和 R_1 碟倒入炒锅中炒好，倒回 R_0 碟。
- (4) 在地址3的格子中，送 R_0 碟到地址7中。

一共有4个加工步骤。

开始时理想厨房系统的状态如图1.3所示。

注意 冰箱格子中的物品以及理想厨房碟中的物品都用斜体字。

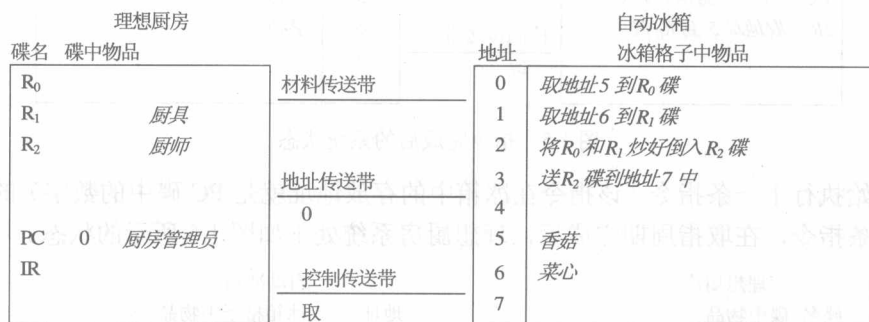


图1.3 开始时理想厨房的系统状态

厨房管理员根据PC碟子中的数字0，知道要到冰箱地址为0的格子中取指令。厨房管理员向控制传送带上送出一个“取”信号，然后马上将PC碟中的数字0复制，然后放到地址传送带上，这两个信号都会到达冰箱。冰箱收到这两个信号后，将0号格的内容副本“取地址5到 R_0 碟”放到材料传送带上送往理想厨房，理想厨房收到后自动将其放到IR碟中，然后厨房管理员将PC碟中的值增加1（以便为取下一条指令做好准备）。取指令周期完成后，理想厨房系统处于如图1.4所示的状态。



图 1.4 取指令周期完成后的系统状态

管理员分析指令存放碟中的加工步骤后, 知道要到冰箱地址为 5 的格子中去取原材料, 并且放到 R_0 碟中。因此, 管理员向控制传送带上发一个“取”信号, 然后马上将 5 这个数放到地址传送带上。

冰箱收到“取”信号后, 知道理想厨房要取物品, 然后冰箱从地址传送带上得到 5, 于是将地址为 5 的格子中的物品“香菇”取出, 放到材料传送带上。

材料传送带上的物品“香菇”被传到理想厨房后, 按照指令的要求通过厨房内部的传送带被自动送到 R_0 碟中。第一条指令执行完后, 理想厨房系统处于如图 1.5 所示的状态:

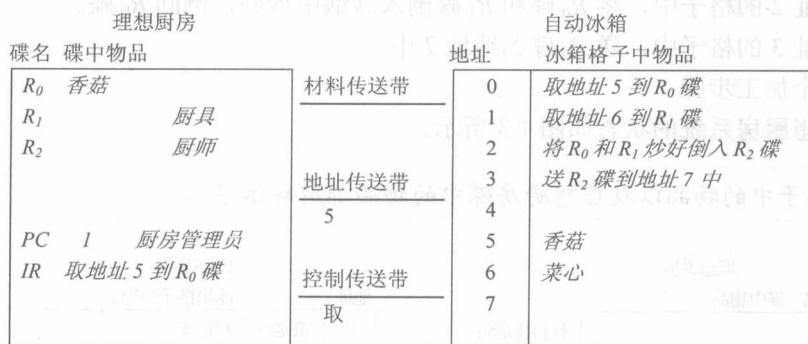


图 1.5 指令完成后的系统状态

下面开始执行下一条指令(该指令在冰箱中的存放地址就是 PC 碟中的数字)的取指周期, 类似于前一条指令, 在取指周期完成后, 理想厨房系统处于如图 1.6 所示的状态。

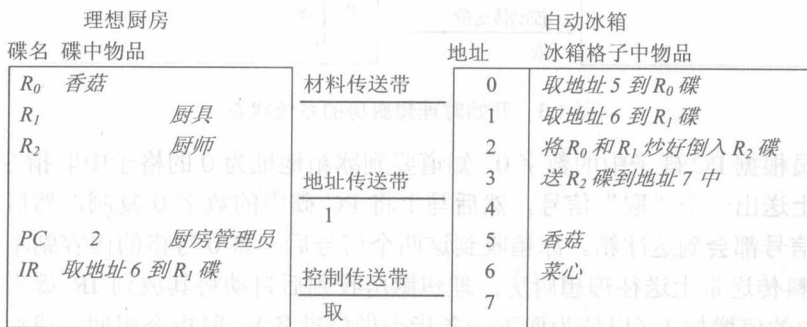


图 1.6 第二条指令完成后的系统状态

管理员分析指令存放碟中的加工步骤后，知道要到冰箱地址为 6 的格子中去取原材料，并且放到 R_1 碟中。因此，管理员向控制传送带上发一个“取”信号，然后马上将 6 这个数放到地址传送带上。

冰箱收到“取”信号后，知道理想厨房要取东西，然后冰箱从地址传送带上得到 6，于是将地址为 6 的格子中的物品“菜心”取出，放到材料传送带上。

材料传送带上的物品“菜心”传到理想厨房后，按照指令的要求通过厨房内部的传送带被自动送到 R_1 碟中。第二条指令执行完后，理想厨房系统处于如图 1.7 所示的状态。



图 1.7 第二条指令取指完成后的系统状态

下面开始执行下一条指令（该指令在冰箱中的存放地址就是 PC 碟中的数字）的取指周期，类似于前一条指令，在取指周期完成后，理想厨房系统处于如图 1.8 所示的状态。

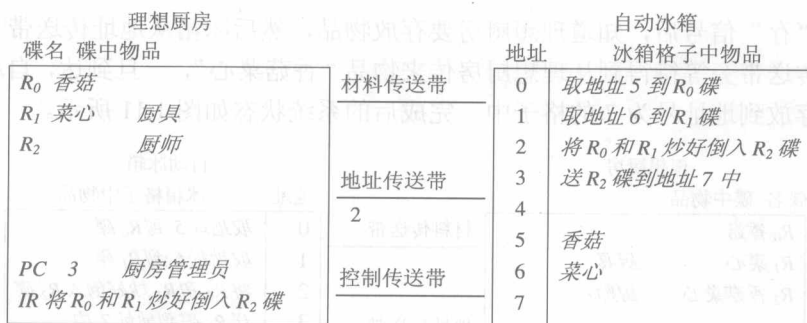


图 1.8 第三条指令取指完成后的系统状态

厨房管理员看到 IR 碟中的内容，于是命令厨师将 R_0 碟和 R_1 碟中的原材料倒入锅中炒好后，倒入 R_2 碟中。完成后如图 1.9 所示。

下面开始执行下一条指令的取指周期，类似于前一条指令的取指令阶段，在取指令周期完成后，理想厨房系统处于如图 1.10 所示的状态。

下面开始执行“送 R_2 碟到地址 7 中”这条指令。厨房管理员分析指令存放碟中的加工步骤后，知道要将 R_2 碟中的物品送到冰箱地址为 7 的格子中存放，于是，管理员向控制传送带上发一个“存”信号，然后马上将 7 这个数放到地址传送带上，最后将 R_2 碟中的物品“香菇菜心”放到材料传送带上，送往冰箱。

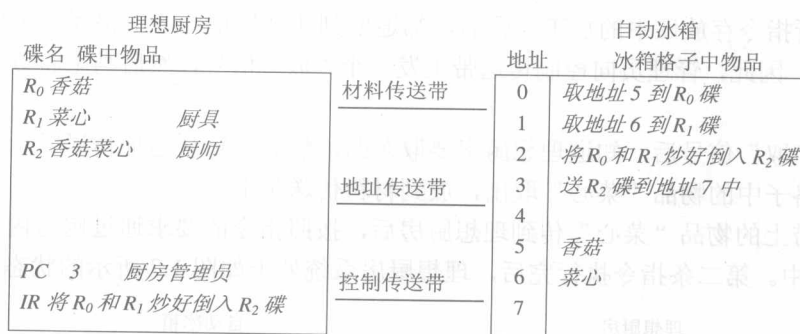


图 1.9 第三条指令执行完后的系统状态

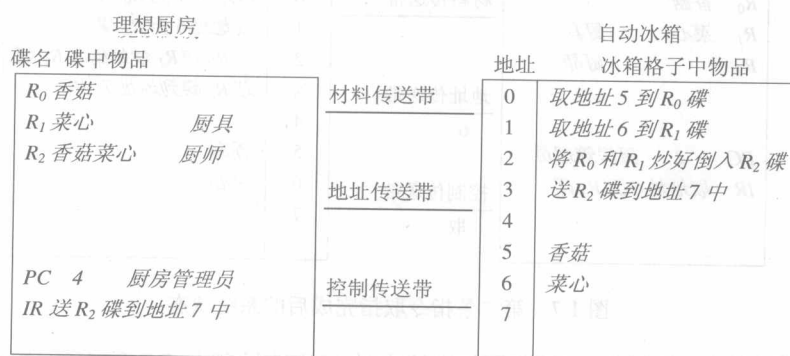


图 1.10 第 4 条指令取指完成后的系统状态

冰箱收到“存”信号后，知道理想厨房要存放物品，然后冰箱从地址传送带上得到 7，于是冰箱在材料传送带旁等待得到从理想厨房传来物品“香菇菜心”，一旦到达，自动冰箱就将其取下，并将其存放到地址号为 7 的格子中。完成后的系统状态如图 1.11 所示。

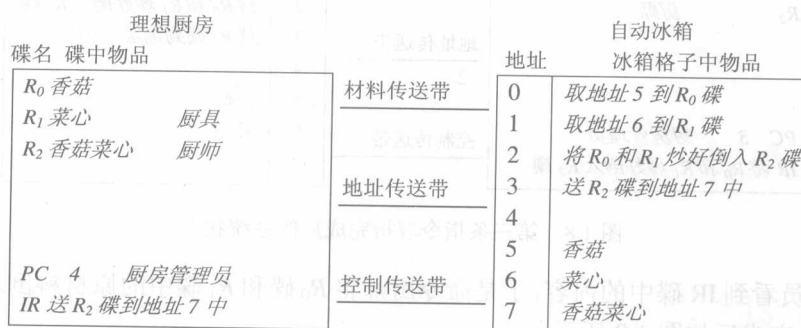


图 1.11 第 4 条指令执行完后的系统状态

至此香菇炒菜心终于大功告成了。

1.3 理想厨房系统的特点

下面通过一些问题来进一步归纳、总结这个有点古怪的理想厨房系统的一些重要特点。先