

21ic社区重磅力推!



# 单片机 编程魔法师

## 高级裸编程思想

- 深度解剖单片机裸编程的奥秘
- 亲密接触数据驱动、分时多任务、面向对象思想

余灿基 主编

张玮 张志柏 苏永刚 编著



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
http://www.phei.com.cn

014059435

TP368.1  
869

# 单片机 编程魔法师

## 高级裸编程思想

余灿基 主编

张玮 张志柏 苏永刚 编著



TP368.1  
869



北航

C1745706

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

010202432

## 内 容 简 介

本书以单片机裸环境为基础，为编程者定义了一个微操作系统（MOS）的编程环境，并面向应用中不断提高的需求对编程策略进行了深度剖析与研究，从而分离出数据驱动、并行多任务、面向对象等重要编程思想。这些思想既可独立运用，又可有机结合成一个体系，是我们实践中解决问题的致胜法宝。本书以实例为基础，分6章对这一思想体系进行了阐述。阐述通常以提出问题开始，然后针对解决问题的现状，从心理学的角度对问题展开讨论，力求将容易遇见的问题一网打尽。本书通过一些列的优化过程对思想要点进行完整描述，然后通过软件仿真手段给读者一个清晰的认识，并在最后进行归纳总结。

本书既介绍了思想，又介绍了使用思想的方法。无论您是单片机自动化领域的初行者，还是资深的单片机自动化领域的工程师，本书都将成为您的得力帮手。希望这种既有理论又有方法论的阐述方式能帮助读者在事业上更上一层楼。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有，侵权必究。

### 图书在版编目（CIP）数据

单片机编程魔法师之高级裸编程思想 / 余灿基主编；张玮，张志柏，苏永刚编著. —北京：电子工业出版社，2014.9

ISBN 978-7-121-23972-4

I. ①单… II. ①余… ②张… ③张… ④苏… III. ①单片微型计算机—程序设计 IV. ①TP368.1

中国版本图书馆 CIP 数据核字（2014）第 177667 号

责任编辑：陈晓猛

印 刷：三河市双峰印刷装订有限公司

装 订：三河市双峰印刷装订有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：787×980 1/16 印张：17.5 字数：390 千字

版 次：2014 年 9 月第 1 版

印 次：2014 年 9 月第 1 次印刷

印 数：3 000 册 定价：59.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn)，盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

服务热线：（010）88258888。

# 序 言

我认识余工已经有好多年了，他做事认真、富有思想、坚持己见的个性是我非常喜欢的地方。他有很多奇思妙想，他可以对宇宙的起源、相对论等深奥话题滔滔不绝。他在单片机开发方面有很多丰富的经验与体会，对程序有独特的理解与观点。他是我难得可以交流思想的朋友之一。

由于我们彼此工作都调整的原因，大概有近一年的时间没有联系。突然有一天接到余工的电话，告诉我他要出一本关于单片机编程的书，告诉我他在几个知名的网站社区上的热帖很受追捧。我感到有些诧异，我从事教育教学工作近 20 年，自认为对单片机的教与学还是有些发言权的，见过的单片机的教材、书籍不下 100 种，目前市面上关于单片机的书籍可以用琳琅满目、大同小异来概括。所以我的第一感觉这不是一个好的主意，如果是为了出一本“多一本不多、少一本不少”的书去浪费精力就有些可惜了。

两天后，余工专程“闯”到我的办公室，和我就单片机编程有哪些非说不可的思想观点，有哪些非同凡响的效果，用什么样的表达方式等问题进行了深入的沟通。后来我又看了部分章节的内容，我感觉这是件无论如何都应该支持的事。余工作为一个工程师，为成千上万的单片机专业教师填补了关于单片机编程思想方面著作的空白。余工站在一个全新的高度来分析单片机与单片机的应用。如果作为教科书，可以避免学生在学习单片机课程时最容易陷入的一种“盲人摸象”状态。我深深地钦佩余工身上的那种执着。

很荣幸能为余工这本凝聚了十多年心血，花费了巨大心思的处女作写序。这是一本探索单片机学习新模式的书，这是一本提升单片机开发思维层次的书，这是一本有思想能实践的书。期待余工的这本书尽快与大家见面，期待更多的学子、工程师能有缘遇到这本书，期待余工更多的奇思妙想。

秦益霖

教授、研究员 高级工程师、硕导

2014 年 6 月 1 日

# 前 言

时下，很多人在设计智能产品时，喜欢为产品装备高档芯片，那架势仿佛即使什么代码都不用写，他的产品就已经是很先进的了。

我们的需求经常很有限，即使是使用单片机都会觉得资源浪费，又怎么会需要更高配置的硬件资源呢？也许在一些应用中我们会遇到一些难题，我们害怕它们，从而指望通过技术升级来解决问题，但是站在应用角度的我们，真的就无能为力了吗？很多盲目升级芯片档次的人不是真正为了提升技术含量的档次，更多的是为了掩饰自己在编程技术上的不足。其实，我们只要稍微修炼一下，一切问题都会迎刃而解。

一些介于单片机与微机之间的高档处理机拥有强大的硬件与软件资源，这似乎让单片机望尘莫及，因为高档处理机往往会在硬件上集成更多的单元来武装自己。然而，很多所谓技术的进步，都只是一些技巧的进步。

高档处理机的另一个优势便是它们拥有强大的软件支持，它们会固化一些软件包，或者支持操作系统。通过技术的向下移植，高档处理机可以做一些原来只有计算机系统才能做到的事情，比如代码的内存调度，并行多任务运行，等等。但是只要你有足够的编程策略，利用单片机来实现那些功能，其实也是不在话下的。

单片机作为一个五脏俱全的小麻雀，高档处理机能做到的，单片机也完全能做到。

但是如何做呢？这就是本书我们要探讨的问题。

本书第 1 章通过对 4 支方波并行输出方案的探究引入数据驱动编程的理念。第 2 章则通过三个互不相关，但要同时运行的并行任务提出并行多任务编程思想，并引入了微操作系统（MOS）编程环境的理念。为了强化这一思想，本书在第 3 章直接针对我们在实际工

作中经常遇到的问题——多定时器、多延时器问题进行多线程编程实现，并在实现过程中引入消息处理机制。通过前3章的技术准备，在第4章正式提出面向对象的编程思路。第5章为这种编程思路（上层建筑）给予一个具体的实践形态（物质基础），同时对实践形态中的一些本质问题花絮也进行了讨论。最后，第6章通过对宝贝车面向对象编程的实践来对全书的裸编程思想进行一次完整而简明的演练，以期让思想这种抽象的东西变得实实在在！

关于本书的源代码，读者可以在电子工业出版社的官网（[www.phei.com.cn](http://www.phei.com.cn)）的“在线资源”中下载。

下面，就让我们一起来看看见证编程技巧给单片机带来的神奇吧。

余灿基

# 目 录

第 1 章 数据驱动程序	1
1.1 数据驱动程序	1
【导读】本节通过对数据驱动程序的概念的界定 (1.1.1 节) 与数据驱动程序的简介 (1.1.2 节), 让读者初步了解数据驱动程序。	
1.1.1 数据驱动程序的定义	1
1.1.2 数据驱动程序简介	2
1.2 4 支方波问题与测试模型	4
【导读】本节对 4 支方波同步实现的问题 (本书的问题 1) 进行了分析 (1.2.1 节), 并建立一个数字测试模型 (1.2.2 节)。本节只是数据驱动程序的引子, 具体的数据驱动程序的编写需要经历一个漫长的演变才能完成 (在后续章节中逐步实现)。	
1.2.1 问题 1 与分析	4
1.2.2 测试模型	5
1.3 一支峰谷等宽方波的实现	6
【导读】为了获得 4 支方波同步实现的解决方法, 先简化问题, 从一支有规律的峰谷等宽的特例方波的实现入手来寻求答案。本节先对这支特例方波进行特性分析 (1.3.1 节), 然后利用常规方法进行实现 (1.3.2 节), 最后通过 1.2.2 节建立的仿真测试模型对这支特例波的实现效果进行验证。	
1.3.1 问题 1-1 与分析	6
1.3.2 实现	8
1.3.3 仿真	8
1.4 一支峰谷不等宽方波的实现	9
【导读】本节对解决问题的任务进行了深入分析, 将这支等宽的特例方波改成不等宽的普通方波, 并对这支普通方波进行分析 (1.4.1 节) 与实现 (1.4.2 节), 最后进行仿真验证 (1.4.3 节)。为了向终极目标迈进, 本节对这次实现的代码进行了亮点分析 (1.4.4 节), 从而引导读者从平凡的代码中去寻求那些宝贵的关键点。	
1.4.1 问题 1-2 与分析	9



1.4.2 实现	10
1.4.3 仿真	11
1.4.4 亮点分析	11
1.5 两支波的实现	12
【导读】通常，多支波的难度系数远远大于1支波的难度系数，而多支波的波数对难度系数的影响不大。但是波数越小，复杂度就越小。所以本节从多支波集合中的2支波向终极问题挺进。本节在对问题进行分析（1.5.1节）之后很轻松地就进行了实现（1.5.2节），似乎困难根本就没有存在过。接下来的仿真（1.5.3）继续为研究的深入提供了信心，而亮点分析（1.5.4节）则向读者暗示：这里的代码很平凡，但是里头却隐藏着什么。	
1.5.1 问题1-3与分析	12
1.5.2 实现	14
1.5.3 仿真	15
1.5.4 亮点分析	15
1.6 4支波的实现	16
【导读】本节通过对问题的分析（1.6.1节）、实现（1.6.2节）与仿真，呈现给读者一个完整的解决方案，但这个方案只是一个普通的方案。至此读者需要思考的是：这个普通的方案与数据驱动有着什么样的关联呢？	
1.6.1 问题1-4与分析	16
1.6.2 实现	17
1.6.3 仿真	19
1.7 冗余代码的一次简化	20
【导读】烦则思变，一个常规的方法往往是不完美的，但是我们所期待的完美却又离不开一个不完美的开始。所以我们绝不能盲目抛弃一个不完美的方案，本节通过对这个常规方案的亮点分析（1.7.1节），然后进行合理简化（1.7.2节），从而寻求一种科学的编程思路。	
1.7.1 亮点分析	21
1.7.2 代码简化	22
1.8 冗余代码的二次简化	23
【导读】随着简化进程的推进，数据驱动程序的思想已经跃然于代码之上（1.8.2节）。	
1.8.1 亮点分析	24
1.8.2 代码简化	24
1.9 冗余代码的三次简化	26
【导读】优化后的代码（1.8.2节）带给我们新的启示（1.9.1节），数据可以从代码中分离出去了，分离出去的数据称为数据脚本（1.9.2节），我们可以用播放器来播放这种脚本（1.9.3节），到	



此我们将对编程耳目一新,也到了思考的时候了(1.9.4节)。	
1.9.1 数码分离的启示	26
1.9.2 数据脚本	27
1.9.3 数据驱动的实现	28
1.9.4 回顾与思考	29
1.10 4支波数据驱动程序应用	29
【导读】我们获得了一种新的思想,现在可以小试牛刀了,将4支方波的形式转换一种面目(1.10.1节),这看似迥然不同的问题,是不是会让我们有顾虑?其实不必担心,我们轻松就能解决这种问题(1.10.2节)。	
1.10.1 问题2与分析	30
1.10.2 实现	32
1.10.3 仿真	33
1.10.4 回顾与思考	33
1.11 总结	34
【导读】继续将问题异样化(1.11.1节),更复杂的现象依然遵循一个简单的道理。至此我们将见识数据驱动的强大力量,所以我们得让这种程序衣冠楚楚地出现在我们的面前(1.11.2节至1.11.4节),我们得从方法论上掌握这种思想(1.11.5节)。	
1.11.1 问题3与分析	34
1.11.2 规范脚本	35
1.11.3 规范播放器	36
1.11.4 规范实现	37
1.11.5 回顾与思考	38
<b>第2章 并行多任务程序</b>	<b>42</b>
2.1 初识并行多任务程序	42
【导读】什么是并行多任务程序(2.1.1节)?单片机能不能胜任并行处理工作(2.1.2节)?本节将给出答案。	
2.1.1 释义	42
2.1.2 单片机能力的评估	43
2.2 并行三任务问题与测试平台	45
【导读】为了见识并行多任务程序的庐山真面目,本节同样利用典型问题(2.2.1节)作为开篇,并建立了一个新的测试模型(2.2.2节)。	
2.2.1 问题4与分析	45

2.2.2	测试模型	46
2.3	并行三任务问题的顺序编程	47
	【导读】我们是不是习惯于使用常规思维来思考问题？我们可以用这种方法来做做看（2.3.2），然后仔细审视一下实现的效果（2.3.3节、2.3.5节、2.3.6节），是不是看似正确，又存在问题（2.3.7节）？这就是常常出现的所谓的不稳定现象（2.3.8节）。	
2.3.1	问题与分析	47
2.3.2	实现	48
2.3.3	仿真一	50
2.3.4	测试分析	50
2.3.5	仿真二	52
2.3.6	仿真三	53
2.3.7	仿真四	53
2.3.8	回顾与思考	54
2.4	运行时序	55
	【导读】问题的根源是什么？是我们需要掌握单片机运行时的时序真相（2.4.1节），要真正了解一个中央处理单元应该如何运行多任务（2.4.2节、2.4.3节）。	
2.4.1	时序分析	55
2.4.2	并行多任务	56
2.4.3	并行多任务的可行性	57
2.5	我们的微操作系统	59
	【导读】裸编程时，我们常常习惯无视环境的存在，这也正是裸编程的典型特征，但是新的需求让这种思想窘态百出，我们需要自己的微操作系统（MOS；2.5.3节），需要在MOS中编写全新理念的程序。	
2.5.1	操作系统与并行多任务	59
2.5.2	单片机的优劣分析	60
2.5.3	微操作系统	62
2.6	任务的生与死	63
	【导读】在MOS中编程，我们必须遵守一定的约束，把任务看作一个个的生命体，并让它们生活在MOS中，而我们则掌握着它们的生杀大权（2.6.4节、2.6.5节）。	
2.6.1	问题5与分析	63
2.6.2	问题5的实现	63
2.6.3	暗点分析	64
2.6.4	任务的生死状态	65

2.6.5 “生”与“死”的实现	65
2.6.6 回顾与思考	66
2.7 一个任务的线程	67
【导读】简化问题，各个击破，是解决问题的绝佳方法，所以本节先实现一个任务（2.7.2节），然后探究其中的奥秘（2.7.3节）。	
2.7.1 问题与分析	67
2.7.2 实现	69
2.7.3 回顾与思考	71
2.8 并行多任务线程	73
【导读】为了实现并行多任务，我们先引入一个与硬件无关的纯代码问题（2.8.1节），这样便于对代码规律进行研究（2.8.3节）。	
2.8.1 问题与分析	73
2.8.2 实现	74
2.8.3 回顾与思考	76
2.9 并行多任务多线程的数据与代码分离	78
【导读】数据与代码分离的编程策略再次登场（2.9.1节），由此可见这种策略是一种必不可少的思想基础（2.9.3节）。	
2.9.1 问题与分析	78
2.9.2 实现	81
2.9.3 回顾与思考	84
2.10 任务的生命	86
【导读】也许我们认为生死与生命是一个东西，但是严格来说，二者并不是一个东西。本节将揭示二者的差别所在。	
2.10.1 问题与分析	86
2.10.2 实现	87
2.10.3 回顾与思考	90
2.11 任务的复活	92
【导读】本节将赋予裸编程者以造物主的思想，不但可以决定一个任务的生死，还可以任意地让某个任务重生。	
<b>第3章 定时器与延时器</b>	<b>94</b>
3.1 并行多任务多线程的等待方案	94
【导读】等待（3.1.1节）是让顺序编程思想崩塌的火山，如果没有这种耗时大户，顺序编程思想	

仍能苟延残喘。所以我们要必须要接触这两个传奇英雄(3.1.2节)。

3.1.1 概述	94
3.1.2 软件定时器与软件延时器	95
3.2 一个软件定时器	96
【导读】学会使用软件定时器,能让我们不再为有限的硬件定时器而苦恼,所以需要先实现(3.2.4节)一个软件定时器(3.2.1节),以便获得定时器的新概念。	
3.2.1 问题6与分析	96
3.2.2 测试模型	97
3.2.3 问题与分析	97
3.2.4 实现	99
3.2.5 仿真	100
3.3 8个软件定时器	100
【导读】我们知道,1与多迥然不同,但是2与多则差异不大。所以本节我们直接从1个软件定时器的实现变成8个软件定时器(3.3.1节)的实现(3.3.2节)。	
3.3.1 问题与分析	100
3.3.2 实现	102
3.3.3 仿真	107
3.3.4 回顾与思考	107
3.4 软件定时器代码优化	109
【导读】代码优化是编程者一项重要的工作内容,而不论这种优化能进行到什么程度。软件定时器的代码该如何优化呢(3.4.1节)?请看这里的实现(3.4.2节)。	
3.4.1 问题与分析	109
3.4.2 实现	110
3.4.3 回顾与思考	114
3.5 时基中断的时序与主程序的关系	114
【导读】尽管软件定时器解决了单片机定时器资源的不足,但是软件定时器到底能有多少呢?本节将回答这个问题。	
3.5.1 时序分析	114
3.5.2 回顾与思考	116
3.6 一个延时器	116
【导读】虽然软件定时器使用起来已经很方便了,但它总是在挤兑其他任务,它具有很高的优先级,而有时我们并不希望耗时短的任务进行过长的等待,所以我们需要一种优先级低的延时工具——延时器。	

3.6.1	问题7与分析	116
3.6.2	实现	118
3.6.3	回顾与思考	119
3.7	8个延时器	120
	<b>【导读】</b> 与软件定时器一样, 延时器也会多个并存而呈现为并行多任务形态, 这里引入8个延时器并发运行的问题(3.7.1节)并提出解决方案(3.7.2节)。	
3.7.1	问题与分析	120
3.7.2	实现	120
3.7.3	回顾与思考	126
3.8	延时器的优化	126
	<b>【导读】</b> 延时器的实现代码如何优化? 本节将会详细讨论(3.8.2节)。	
3.8.1	问题与分析	126
3.8.2	实现	128
3.9	任务代码的初步改造	132
	<b>【导读】</b> 问题4提出已久, 但是我们并未正面解决它, 而是插入了并行多任务的编程思想, 这些思想与问题4有什么关联呢(3.9.1节)? 请看本节是如何处理的(3.9.2节)。	
3.9.1	问题与分析	132
3.9.2	实现	132
3.9.3	回顾与思考	135
3.10	消息处理	136
	<b>【导读】</b> 在一个MOS中编程, 往往会产生不少新的编程策略, 消息机制就是一个典型。本节将重点介绍消息机制(3.10.1节)并举例说明消息处理代码编写的方法(3.10.2节)。	
3.10.1	问题与分析	136
3.10.2	实现	137
3.10.3	问题分析	139
3.10.4	实现	139
3.10.5	回顾与思考	142
3.11	广播消息	142
	<b>【导读】</b> 尽管消息机制只是一个概念, 但是对应消息的处理却十分丰富。本节将重点探讨消息是如何广播的。	
3.11.1	问题与分析	142
3.11.2	实现	143
3.11.3	回顾与思考	147

3.12 任务代码的最终改造 .....	148
【导读】万事俱备，现在我们只需要利用并行多任务思想、消息机制、任务嵌套手段来分析问题 4(3.12.1 节)与解决问题 4(3.12.2 节)就可以了，对于新程序的正确性我们可以进行仔细验证(3.12.3 节)，您一定会发现，原来一个好的思想是多么的神奇，您甚至不用担心会存在 BUG。	
3.12.1 问题与分析 .....	149
3.12.2 实现 .....	149
3.12.3 仿真 .....	154
3.12.4 回顾与思考 .....	154
3.12.5 亮点分析 .....	155
3.13 状态指示灯 .....	155
【导读】这是一个常见的问题，也是并行多任务思想的一个典型综合应用。为了节省资源，设备只有一个指示灯，它必须同时完成不同状态的指示（冲突时靠后者指示优先），如何做到这一点呢？本节将给出完整的实现策略。	
3.13.1 问题 8 与分析 .....	155
3.13.2 测试模型 .....	157
3.13.3 实现 .....	157
3.13.4 回顾与思考 .....	161
<b>第 4 章 面向对象的程序 .....</b>	<b>162</b>
4.1 计算机的语言特征 .....	162
【导读】发现新思想，从用正确的视角看 C 语言开始(4.1.1 节)；了解智能芯片的心情，从了解自己的心情开始(4.1.2 节)。本节将从思想角度来引导读者走出常规思维的盲区。	
4.1.1 正视 C 语言 .....	162
4.1.2 以对象看世界 .....	163
4.2 兔类的传奇 .....	164
【导读】代码中的对象理解起来十分生涩，但是我们可以从生活中的对象开始(4.2.1 节)，一只兔子能带给我们什么样的启迪(4.2.2 节)？一只兔子与兔子的种群有着什么样的关系(4.2.3 节、4.2.4 节)？本节将要展示单片机内心世界中的兔子种群是什么样的(4.2.5 节)。	
4.2.1 兔类浅说 .....	164
4.2.2 单片机中的兔类 .....	165
4.2.3 兔类的结构 .....	166
4.2.4 兔类的属性成员 .....	167
4.2.5 兔类的实现 .....	169



4.3 兔子的传奇	170
【导读】我们可以像上帝一样设计一个种群，然后根据这个种群的特点创造一只只的兔子对象。本节将深刻探讨兔子对象有着什么样的传奇。	
4.3.1 问题9与分析	170
4.3.2 实现	171
4.3.3 回顾与思考	173
4.4 面向对象编程的书写规范	174
【导读】如何让思想成为方法论，这也是本书的重要任务之一。本节打开从思想通向实践的时空隧道，指引读者一步步将面向对象思想落到实处。	
4.4.1 问题与分析	174
4.4.2 实现	180
4.5 方波对象	184
【导读】单片机中几乎不会有兔子，但会有方波。本节从一个走进单片机内心世界的兔子模型开始讲解，逐渐引导读者进入单片机的内心世界去看一支方波。因为只有方波才是单片机内心世界的常见对象。	
4.5.1 问题10与分析	184
4.5.2 测试模型	186
4.5.3 综合分析一	186
4.5.4 增补的面向对象编程的头文件	189
4.5.5 综合分析二	190
4.5.6 实现	198
4.5.7 仿真	207
4.5.8 回顾与思考	207
<b>第5章 对象的归宿</b>	<b>209</b>
5.0 引言	209
【导读】思想与方法论的改变对我们的产品和工作会带来哪些影响呢？本节简单地引出问题。	
5.1 解密对象魔法	209
【导读】生命是一个对象的本质特征，掌握对象的生命特征是灵活运用对象思想的重要保证。	
5.1.1 对象的生命特征	210
5.1.2 对象生命特征的含义	210
5.2 项目管理	211
【导读】对象思想的项目化是面向对象编程进入实践的基本途径，本节将探讨如何用—个 Keil C	



的 Project 文档来组织一个对象思想的程序。	
5.2.1 项目的内容	211
5.3 项目的实现	214
【导读】本节组织文档的内容。	
5.3.1 文档的落实	214
5.3.2 文档的分包	222
5.4 对象文档与项目分离	224
【导读】为了增加代码的可再用性，我们得让常用代码与某一个具体的项目分离。对象就是这种典型的常用代码，我们得让它从物质基础上独立于整个项目。	
5.4.1 任务与分析	224
5.4.2 面向对象编程的层次关系	225
5.5 源码的商业保护	226
【导读】商业保护是日常工作的一个重要事务，保护源码，发布二进制码是一个基本做法，而库制作则是实现二进制码发布的一个重要工具，本节具体介绍库的相关知识。	
5.5.1 库文件	226
5.5.2 库中的模块	227
5.5.3 制作库	228
5.5.4 使用对象库	229
5.5.5 库、模块与对象的关系	230
5.5.6 库的操作	230
5.5.7 创库计划	232
5.6 对象的花絮	233
【导读】一个思想上的对象，属于上层建筑的内容，所以它依赖很多物质基础（5.6.2 节、5.6.3 节），任何一个离散的物质基础都会对它产生或多或少的影响，我们要掌握这些虚虚实实的内在关联（5.6.3 节），并能抽象与总结这种客观的存在（5.6.1 节、5.6.4 节）。	
5.6.1 对象分析的观点	233
5.6.2 内存统筹的观点	234
5.6.3 虚拟与现实相通的观点	234
5.6.4 常见的对象	238
<b>第 6 章 宝贝车的综合应用</b>	<b>241</b>
6.1 宝贝车简介	241
【导读】通过一个具体的宝贝车应用（6.1.1 节）来将前面所述思想进行一个综合的实践。	

6.1.1 问题 11	241
6.2 对象分析	242
【导读】对象与对象之间既独立，又合作，本节通过宝贝车的对象分析来介绍对象关系的处理方法（6.2.1 节），并构造一个宝贝车的对象资源库（6.2.2 节）。	
6.2.1 组合对象与实现	242
6.2.2 宝贝车的库项目	246
6.3 实现对象	248
【导读】本节完成宝贝车对象的各种行为特征的全部代码设计。	
6.3.1 车轮的定义	248
6.3.2 脉冲发生器	249
6.3.3 宝贝车的控制	252
6.3.4 宝贝车的创建	253
6.3.5 实现	254
6.3.6 修成正果	259
6.4 对象的使用	259
【导读】本节将演示在一个具体的项目中如何很方便地使用一个共享的独立对象代码资源库中的对象——宝贝车。	
参考文献	261