

面向 21 世纪

高等学校系列教材

计算机辅助机械设计

Computer Aided Mechanical Design

秦汝明 主编 董建国 主审

西安电子科技大学出版社
<http://www.xduph.com>



面向 21 世纪高等学校系列教材

计算机辅助机械设计

Computer Aided Mechanical Design

秦汝明 主编

董建国 主审

西安电子科技大学出版社

2005

内 容 简 介

本书主要介绍了跨平台组合式机械设计应用软件的开发技术。全书分为两部分,共6章。第一部分为机械设计软件开发基础,包括第1~4章。第1章概括介绍了计算机辅助机械设计的概念。第2、3章详细介绍了用可视化开发工具 Visual Basic 6.0 和 Visual FoxPro 6.0 处理机械设计资料中的图线、区域和数表的方法。第4章系统介绍了小型机械设计软件的开发技术,重点介绍了用不同平台开发的程序之间组合的方法和数据传递技术。第二部分为实践部分,包括第5、6章。第5章介绍了用 Visual Basic 6.0 开发普通 V 带传动设计软件的方法。第6章较全面地介绍了跨平台开发渐开线圆柱齿轮传动设计软件的方法。书中有相当多的经上机验证过的例程及较完整的小型机械设计软件源程序,可供读者上机调试与学习参考。

本书可作为高等院校机械专业教材,也可作为机械 CAD 工程技术人员的参考书。

本书配有电子教案,需要的教师可与出版社联系,免费提供。

图书在版编目(CIP)数据

计算机辅助机械设计/秦汝明主编. —西安:西安电子科技大学出版社,2005.3

ISBN 7-5606-1463-9

I. 参… II. 秦… III. 机械设计:计算机辅助设计—应用软件—高等学校:技术学校—教材 IV. TH122

中国版本图书馆 CIP 数据核字(2004)第 105846 号

责任编辑 王晓杰 云立实

出版发行 西安电子科技大学出版社(西安市太白南路2号)

电 话 (029)88242885 88201467 邮 编 710071

http://www.xduph.com E-mail: xdupfxb@pub.xaonline.com

经 销 新华书店

印刷单位 陕西华沐印刷科技有限责任公司

版 次 2005年3月第1版 2005年3月第1次印刷

开 本 787毫米×1092毫米 1/16 印张 17.25

字 数 410千字

印 数 1~4 000册

定 价 19.00元

ISBN 7-5606-1463-9/TH·0044(课)

XDUP 1734001-1

*** 如有印装问题可调换 ***

本社图书封面为激光防伪覆膜,谨防盗版。

前 言

可视化计算机操作系统及各种各样支撑软件的迅猛发展,使跨平台组合式可视化机械设计软件的开发成为现实。实践证明,这种开发方法具有高效率、低成本的优点。掌握这种方法,是进行计算机辅助机械设计的有效途径之一。然而,目前能够采用的教材往往难以满足这样的教学要求。为此,编者把多年来不断完善的计算机辅助机械设计教学讲义和一些自编软件综合整理成书,以填补目前教材之空缺。

本书以培养学生对机械设计图表资料的处理能力为基础,以培养学生综合应用可视化语言(如 Visual Basic 6.0、Visual FoxPro 6.0 和 Visual Lisp)开发小型机械设计应用软件的能力为目的。书中所附例程和较完整的程序,均以常用机械传动的计算机辅助设计为开发对象,并突出对图表数据处理能力、模块化组合式软件设计理念和方法的培养,具有较强的实用性。

全书共分为 6 章,内容主要包括两部分。第一部分为机械设计软件开发基础,包括第 1~4 章。第 1 章概括介绍了计算机辅助机械设计的概念。第 2、3 章详细介绍了用 Visual Basic 6.0 和 Visual FoxPro 6.0 处理机械设计资料中的图线、区域和数据表的方法。第 4 章系统介绍了跨平台组合式小型机械设计软件的开发技术,重点介绍了用不同平台开发的程序组合方法和接口技术。第二部分为实践部分,包括第 5、6 章。第 5 章介绍了用 Visual Basic 6.0 开发普通 V 带传动设计软件的方法,目的是强化学生对机械设计图表的处理能力。第 6 章介绍了用跨平台组合式技术开发渐开线圆柱齿轮传动设计软件的方法。

本书由秦汝明、孟逵、赵艳霞编写。秦汝明担任主编,董建国担任主审。在本书的编写过程中,得到了河南工业大学机械工程学院及教务处有关领导和老师的热情帮助和支持,在此一并表示衷心的感谢!

由于作者水平有限,书中难免有错误与不足之处,敬请读者批评指正。

编 者

2005.1

目 录

第一部分 机械设计软件开发基础

第 1 章 计算机辅助机械设计概述.....	3
1.1 机械设计过程	3
1.2 软件技术概述	5
1.3 软件危机	6
1.4 软件工程概述	8
1.5 计算机辅助机械设计概述.....	10
第 2 章 机械设计数表处理	13
2.1 非函数表处理	14
2.1.1 规则非函数表处理.....	14
2.1.2 不规则非函数表处理.....	26
2.2 函数表处理	49
2.2.1 函数表的插值法处理.....	49
2.2.2 函数表的公式法处理.....	70
2.3 数值处理.....	80
2.3.1 舍去小数部分.....	80
2.3.2 角度与弧度的转换.....	81
习题	83
第 3 章 机械设计图线处理	84
3.1 直线处理	84
3.1.1 直角坐标系中的直线处理.....	84
3.1.2 双对数坐标系中的直线处理.....	86
3.2 曲线处理	89
3.3 区域处理	91
第 4 章 组合式软件设计	94
4.1 组合式软件设计的基本概念	94
4.2 接口技术	97
4.2.1 程序接口.....	97

4.2.2 可执行程序的正确调用	97
4.2.3 可执行文件之间的数据传递	106
4.3 主程序的确定	107
习题	108

第二部分 实践部分

第5章 普通V带传动计算机辅助设计	111
5.1 用户需求	111
5.2 系统结构设计	113
5.2.1 普通V带传动的手工设计过程	113
5.2.2 软件结构设计	117
5.3 数表的规则化	120
5.3.1 单根带额定功率的查询	121
5.3.2 V带轮轮辐结构的查询	121
5.4 选型图的处理	124
附录 普通V带传动计算机辅助设计 VB 6.0 源程序	126
第6章 渐开线标准圆柱齿轮传动计算机辅助设计	158
6.1 用户需求	158
6.2 系统结构设计	162
6.2.1 渐开线标准圆柱齿轮传动的手工设计过程分析	162
6.2.2 确定软件功能模块与调用关系	167
6.3 渐开线标准圆柱齿轮传动的设计计算	169
6.3.1 输入设计	169
6.3.2 设计计算模块的设计	169
6.4 渐开线标准圆柱齿轮传动的公差查询	171
6.4.1 齿轮工作图标注内容及 VB 到 VL 之间的参数传递	171
6.4.2 公差查询 VF 6.0 数据库设计	176
6.5 渐开线标准圆柱齿轮自动绘图程序设计	185
6.6 设计结果的输出	187
附录 A 渐开线标准圆柱齿轮传动计算机辅助设计软件 VB 6.0 源程序	188
附录 B 渐开线标准圆柱齿轮自动绘图 VL 源程序	244
参考文献	270

第一部分

机械设计软件开发基础

第 1 章 计算机辅助机械设计概述

本课程以研究常用机械传动设计(以下简称机械设计)软件的开发方法为主要目的。通常,一项计算机辅助机械设计任务,就是开发一套合格的应用软件。它的处理对象是机械设计内容。运行它,可以实现机械设计的参数化。所谓机械设计的参数化,就是在软件运行中,只需要改变为数不多的输入参数(如传递功率、转速),即可获得相应的机械设计结果(如主要设计参数、零件工作图等)。

为了便于学习,我们需要明确计算机辅助机械设计课程的学科范畴。从机械设计的观点来看,计算机辅助机械设计是众多机械设计方法中的一种,它利用计算机系统,辅助完成机械设计任务,应该属于机械设计范畴;换一个角度来看,计算机辅助机械设计不是去研究常规机械设计方法本身(常规机械设计方法已经成熟),而是研究机械设计软件的开发方法,也就是研究计算机系统中的一个组成部分——软件开发方法,所以它应当属于软件开发范畴。这两种划分方法都有一定的道理。因此,计算机辅助机械设计应该属于一种交叉学科或边缘学科,是机械设计学科与软件设计学科相互交叉而形成的一门学科。我们可以认为,计算机辅助机械设计是研究常用机械设计应用软件的一门学科。

交叉学科的特点不仅反映在学科划分时的模棱两可,最重要的是,学科之间通过交叉,产生了区别于相互交叉学科的一些新的内容与规律。计算机辅助机械设计也不例外,它通过机械设计方法与软件开发方法的相互交叉,产生了诸如“组合式设计”技术、“数据接口”技术、“参数传递”技术等一些新技术内容。

由于计算机辅助机械设计课程既兼容了多种学科的原有知识,又产生了一些新的内容,因此在学习时,需要运用多种学科知识与能力,尤其是机械设计与软件设计的知识与能力。下面就机械设计与软件设计作一些必要的概述。

1.1 机械设计过程

机械的种类很多,但其设计的基本要求大致相同,主要有以下几个方面:

(1) 实现预定功能。设计机械的基本出发点是实现预定的功能要求。功能要求是指被设计机械的功用和性能指标。一般机械的预定功能要求包括运动性能、动力性能、基本技术指标及外形结构等方面。

(2) 满足安全可靠与强度、刚度和寿命等要求。安全可靠是机械正常工作的必要条件,因此,设计的机械必须保证在预定的工作期限内能够可靠地工作。为此,应使所设计的零件结构合理并满足强度、刚度、耐磨性、耐热性、振动稳定性及其寿命等方面的要求。

(3) 满足经济性要求。经济性是一个与设计、制造和使用等各方面有关的综合性要求。良好的工艺性、合理的选材、尽可能实现三化(零件标准化、部件通用化、产品系列化)等,

可获得较好的性能价格比,取得较好的经济效益。

(4) 满足操作和使用要求。设计要力求使机械的操作、维护、修理方便,有利于减轻使用人员的劳动强度。考虑好安全防护装置和措施,避免人身及设备事故的发生。另外还需要满足产品造型美观,噪声较低,污染较轻等要求。

机械设计是一项复杂、细致和科学性很强的工作。机械设计过程也不是一成不变的。随着科学技术的发展,对设计的理解在不断地深化,设计方法也在不断地发展。就目前来说,机械设计的一般过程通常可分为以下几个阶段:

(1) 产品规划。根据市场需求提出设计任务、明确设计要求。

(2) 方案设计。在分析设计任务和要求的条件下,确定机械的工作原理和技术要求,拟定机械的总体布置、传动方案等。由设计人员构思出多种可行方案并进行比较评价,从中选取最佳方案。

(3) 技术设计(总体设计、结构设计)。在既定设计方案的基础上,完成机械产品的总体设计、部件设计、零件设计等,设计结果为工程图、计划书等技术文件。

(4) 试制及鉴定。经过加工、安装及调试制造出样机。对样机进行试运行或在生产现场试用,将试验过程中发现的问题反馈给设计人员。经过反复修改完善,从技术上、经济上作出全面的评价,最后定型生产。下面给出机械设计的一般过程框图(图 1-1),以便读者更清楚地了解机械设计的一般过程。

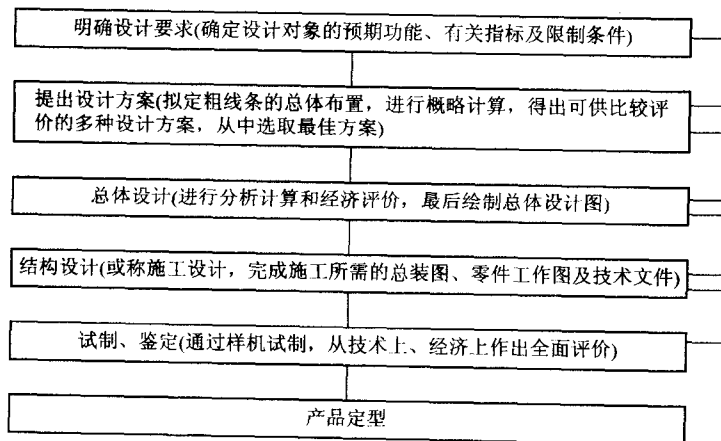


图 1-1 机械设计的一般过程

机械设计过程是一个复杂细致的过程,整个设计过程中的各个阶段是互相紧密联系的,也是一个不断反复、不断完善、逐渐接近最优结果的过程。

与设计机械一样,设计机械零件时,也常需拟定出几种不同方案,经过认真比较来选用其中最好的一种。为此,必须对所有零件(标准件除外)进行结构设计,并对主要零件的工作能力进行计算,即进行机械零件设计。设计机械零件的一般步骤如下:

(1) 根据机械零件的使用要求,选择零件的类型与结构。

(2) 根据机械的工作要求,分析零件的工作情况,确定作用在零件上的载荷。

(3) 根据零件的工作条件(包含对零件的特殊要求,例如耐高温、耐腐蚀等),考虑材料的性能、供应情况、经济因素等,合理选择零件的材料。

(4) 根据零件可能出现的失效形式, 确定其计算准则, 并通过计算, 确定零件的主要尺寸。

(5) 根据零件的主要尺寸及工艺性、标准化等要求, 进行零件的结构设计。

(6) 绘制零件工作图, 制定技术要求。

对于不同的零件和工作条件, 以上这些设计步骤可以有所不同。此外, 在设计过程中, 这些步骤又是相互交错、反复进行的。

应当指出, 在设计机械零件时往往是将较复杂的实际工作情况进行一定的简化, 才能应用力学等理论解决机械零件的设计计算问题, 因此, 这种计算或多或少带有一定的条件性或假定性, 故称之为条件性计算。机械零件设计基本上是按条件性计算进行的, 如果注意公式的适用范围, 计算结果一般都具有一定的可靠性和安全性。为了使计算结果更符合实际情况, 必要时可进行模型试验或实物试验。

1.2 软件技术概述

1. 软件的概念与特点

“软件”一词是 20 世纪 60 年代出现的, 其定义是计算机程序及其说明程序的各种文档。在该定义中, 程序是一系列指令序列的集合, 它能被计算机理解和执行。文档是指用自然语言或者形式化语言所编写的文字资料和图表, 用来描述有关计算机软件的功能、内容、设计、测试结果、使用方法, 如设计说明书、流程图、用户使用手册等。软件与硬件一起构成完整的计算机系统, 它们是相互依存的, 缺一不可的。软件是一种特殊的产品, 它具有下列一些特点:

(1) 软件是一种逻辑产品, 它与物质产品有很大的区别。软件产品是看不见摸不着的, 因而具有无形性。它以程序和文档的形式出现, 保存在计算机存储器的磁盘和光盘等介质上, 通过计算机的运行才能体现它的功能和作用。

(2) 软件产品的生产主要是研制, 其成本主要体现在软件的开发上。软件开发完成后, 通过复制就产生了大量软件产品。

(3) 软件产品不会用坏, 不存在磨损、消耗问题。

(4) 软件产品的生产主要是脑力劳动, 目前还未完全摆脱手工开发方式, 大部分产品是“定做”的。

(5) 软件费用不断增加, 软件成本相当昂贵。软件的研制工作需要投入大量的、复杂的、高强度的脑力劳动, 它的成本非常高。

(6) 开发周期长。大型软件从立项到交付使用, 需几十人、几百人经过几个月甚至几年的时间。

(7) 开发、维护和使用人员不同。

(8) 多学科的综合。软件开发人员除了具有必备的软件知识外, 还应该具有多方面的专业知识和经验。

2. 计算机软件技术

计算机软件技术是指开发计算机软件所需的所有技术的总称。按照软件分支学科的内

容划分,计算机软件技术主要有以下几个领域:

(1) 软件工程技术。其中包括软件开发的原则与策略、软件开发方法与软件过程模型、软件标准与软件质量衡量、软件开发的组织与项目管理和软件工程工具及环境等。

(2) 程序设计技术。其中包括程序的结构与算法设计、程序设计的风格、程序设计语言、程序设计方法和程序设计自动化以及程序的正确性证明和程序的变换。

(3) 软件工具环境技术。其中包括人机接口技术、软件自动生成、软件工具的集成、软件开发环境和软件的复用等。

(4) 系统软件技术。其中包括操作系统、编译方法、分布式系统的分布处理与并行计算、并行处理技术和多媒体软件技术。

(5) 数据库技术。其中包括数据模型、数据库与数据库管理系统、分布式数据库、面向对象的数据库、工程数据库、多媒体数据库以及数据仓库和数据挖掘等。

(6) 网络软件技术。其中包括网络软件技术、网络管理、局域网技术、网络互连技术和智能网络等。

3. 软件复用

软件复用是指在构造新的软件系统的过程中,对已存在的软件产品(设计结构、源代码、文档等)重复使用的技术。软件复用有三个层次:知识的复用、方法的复用和软件成分的复用,前两个属于知识工程的范畴,这里只讨论软件成分的复用。软件成分的复用包括以下三个级别:

(1) 代码的复用,可以采用源代码剪贴、源代码包含和继承来实现。

(2) 设计结果的复用,是指复用某个软件系统的设计模型,适用于软件系统的移植。

(3) 分析结果的复用,是指复用某个软件系统的分析模型,适用于用户需求未改变,而系统体系结构变化的场合。

Microsoft 等公司提出了 OLE/COM (Object Linking Embedding/Component Object Model, 对象链接和嵌入/组件对象模型)概念,并开发出了各种独立的标准组件。用户使用这些组件集成自己的软件,提高了软件的质量,软件维护变得更加容易,同时也降低了软件开发成本。

由于软件复用是利用已有的软件成分来构造新的软件,因此大大缩减了软件开发所需的人力、物力、财力和开发时间,并且能提高软件的可靠性和维护性。

1.3 软件危机

1. 软件危机的概念

计算机出现后的相当长一段时间内,人们在编制程序或开发软件系统时,往往一个人或几个人按照各自的习惯和经验来进行,这实际上是一种手工作坊式的生产方式。以这种方式研制的软件,其周期长、费用高,而且质量低下、可靠性得不到保证。到 20 世纪 60 年代中期,计算机进入集成电路时代,大批功能强大的计算机进入市场,高级语言蓬勃兴起,计算机应用迅速地渗透到各个科学技术领域。同时,程序的规模和复杂程度也急剧地增长,而程序设计仍停留在个体手工方式上。到 20 世纪 70 年代初,形成了所谓的“软件危

机”。软件危机是指在计算机软件的开发和维护过程中所遇到的一系列严重的问题，其主要表现在以下几个方面：

(1) 对软件成本、开发成本和开发进度的估计不准确；软件成本在计算机系统总成本中所占的比例逐年上升。

(2) 用户对“已完成的”软件系统不满意的现象时常发生。

(3) 软件产品的质量往往靠不住。

(4) 软件通常没有适当的文档资料，维护困难。

(5) 软件开发生产率的提高速度跟不上计算机应用的迅速普及和深入。

2. 软件危机产生的原因

在 1946 年第一台计算机“诞生”以后的很长一段时间里，人们都是用计算机来编制一些小程序，解决一些“小问题”。随着计算机软硬件的发展，人们用计算机来解决的问题越来越多，程序规模也越来越大，而开发大型软件与编制小程序有一定的区别，具体体现在以下几个方面：

(1) 人员：小程序从确定要求、设计、编制、使用，直到维护，通常由一个人来完成；而大型软件则必须由用户、项目负责人、分析员、初级程序员、资料员、操作员等组成一支开发队伍来共同完成。

(2) 文档：小程序是编制者脑中的“产品”，很少有书面文档；而大型软件是集体劳动的“产物”，必须有规范化的文档，便于开发和维护。

(3) 产品：小程序通常是一次性的，如果需要大的修改，可舍弃旧程序而重新编写；而大型软件的开发耗费了大量的人力与物力，轻易抛弃将造成很大的浪费，因此总是在旧软件的基础上对其进行改动，以延长它的使用期，所以其“版本”不断升级。

大型软件的开发提出了许多新的问题，而开发方法却还停留在编制小程序的方法上，经验和技能已不能满足开发大型软件的需要，从而导致软件开发过程混乱。使用的开发方法和技术不当，没有适当的文档，不易交流，维护困难，开发成本高，软件质量低等等，这些问题是造成软件危机的主要原因。

3. 软件危机的解决方法

软件危机使人们认识到，当研制的软件规模日趋庞大时，问题变得越来越复杂。软件的研制和开发再也不能像以前那样了，必须立足于科学理论的基础，像生产产品、研制机器或建造楼房那样，以“工程化”的思想来指导软件开发，解决软件研制中面临的困难和混乱，从根本上解决软件危机。在技术上，以软件工程技术、程序设计方法和技术为基础，力求将软件工程与知识工程、人工智能技术结合起来，以构造基于知识的软件开发环境；在管理上，以管理学为依托，对开发人员、成本、项目、文档等加强管理，对软件开发全过程进行控制。

大型程序与小型程序相比，二者有本质的区别。小型程序规模小，程序不太长，一个人或几个人就能管理和控制其复杂性，因此可以在较短的时间内完成程序的编制并达到预期的效果。但大型程序的程序规模大，可由几十个、几百个模块组成，模块间的接口较为复杂，常常需要许多程序员工作很长时间才能完成。因此，大型程序具有程序设计工作量大、复杂程度高、研制周期长、可靠性低等特点。这样的程序往往是难以维护的，而且许多

程序中的错误也难以纠正。人们对正确可靠的、可维护的软件系统的迫切需求与软件研制的手工作坊方式的矛盾日益尖锐化。为了解决这一矛盾，人们提出了软件工程的观念。

1.4 软件工程概述

软件开发不是某种个体劳动的神秘技巧，而是一种组织良好、管理严密、各类人员协同配合、共同完成的工程项目。为了解决软件危机，既要有技术手段(方法和工具)，又要有必要的组织管理和措施。软件工程正是从管理和技术两个方面，研究如何更好地开发和维护计算机软件的一门新兴学科。

软件工程是指导计算机软件开发和维护的工程科学。它采用工程的概念、原理、技术和方法来开发与维护软件。因此，在开发与维护软件过程中，把经过时间考验而证明正确的管理技术和当前能够得到的最好的技术方法结合起来，这就是软件工程。软件工程强调使用软件生命周期方法学和各种结构分析及结构设计技术。所谓软件生命周期是指一个软件从定义、开发、使用和维护，直到最终被废弃所经历的一段漫长的时期。由于软件生命周期分为若干个阶段，每个阶段的任务相对独立和简单，因此软件研制可以用分工协作的方法来共同完成。在每个阶段采用科学的管理技术和良好的技术方法，结束时从管理和技术两个角度进行严格审查，合格后进入下一个阶段这样可以保证软件质量，提高软件的可维护性。

软件生命周期可分为三个时期，每个时期又可分成若干个阶段。软件生命周期模型见表 1-1。

表 1-1 软件生命周期模型

时 期	阶 段	
软件定义	软件计划	
	软件需求分析	
软件开发	软件设计	总体设计
		详细设计
	软件编码	
	软件测试	单元测试
		综合测试
软件维护	软件使用	

对表 1-1 中软件生命周期各阶段说明如下：

(1) 软件计划：在设计任务确立前，首先要进行调研和可行性研究，理解工作范围和所花费的代价，然后作出软件计划。

(2) 软件需求分析：对用户要求进行具体分析，确定用户要求软件系统做什么，并用软件需求规格说明书表达出来，作为用户和软件人员之间共同的约定。

(3) 软件设计：根据需求说明建立软件系统的“结构”，包括数据结构和模块结构。它又分为总体设计和详细设计两个阶段。其中，总体设计决定系统的模块结构，给出模块的相互调用关系，产生软件概要说明书；详细设计则给出模块内部的细节及功能说明，产生

详细的设计说明书。

(4) 软件编码：按软件设计的要求为每个模块编写程序。

(5) 软件测试：发现和排除程序中留存的错误，经过测试并排除错误，得到可以交付运行的软件。软件测试又分为单元测试和综合测试两个阶段。

(6) 软件维护：经过测试的软件仍然可能有错误，用户需求和系统的操作环境也有可能发生变化。因此，交付运行的软件仍然需要继续排查错漏、修改错误和扩充内容，这就是软件的维护。

软件生命周期各阶段的任务、应完成的工作如表 1-2 所示。

表 1-2 软件生命周期各阶段的任务

阶 段	基本任务	产生文档
软件计划	理解工作范围	计划任务书
需求分析	定义用户要求	需求规格说明书
软件设计	建立软件结构	软件概要设计说明书
		软件详细设计说明书
软件编码	编写程序	程序
软件测试	发现和排除错误	可运行的程序
软件维护	运行和管理	改进的系统

软件开发的手段，经历了从手工编码到使用支撑软件产品的自动化软件工具的变迁。现在，从软件的开发、运行到维护各阶段都有软件工具，这些工具形成了现代化软件环境的基础。

软件工具是指可以用来帮助开发、测试、分析、维护其它计算机程序的程序以及文档资料的集合，它可以实现软件生产过程的自动化，提高软件的生产率、可靠性，降低软件的生产成本。软件工具在各种状况下都能被简单、方便地使用，能给软件的开发带来极大的方便。大型软件生产所使用的软件工具是一种自动化系统，包括需求分析工具、设计工具、编码工具、确认工具、维护工具等。比如，需求分析工具能够辅助系统分析员把用户所提出的含糊的用户说明，经过分析及一致性、完备性检查后，快速生成指导系统设计用的“需求规格说明书”及其相应的文档资料。设计工具能够依据输入的需求规格说明，自动设计出一系列软件设计文档，自动生成特定语言编制的程序，如各种应用程序生成器等。

尽管软件工具种类繁多、形式多样，但都只是用于软件生存周期中的某一个阶段或某一个环节，而不能对整个生命周期有效。为了能够给软件整个生命周期提供支持，于是出现了新课题：软件工程环境。软件工程环境(Software Engineering Environment, SEE)是指用以支持需求定义、程序生成以及软件维护等整个软件生命周期全部活动的，并把方法、规模和计算机程序集成在一起的整个体系。软件工程环境又称为软件开发环境、软件支撑环境、自动开发环境等。

软件工程环境的全部需求可以概括为：

- (1) 集成化的系统；
- (2) 通用的系统；

(3) 既可剪裁又可扩充的系统;

(4) 实用的、经济合算的系统。

近几年来,软件工程领域中出现了一种新趋势,即将软件工程方法、工具与环境方面的新技术同形式化语义理论有机地结合起来,形成高水平的计算机辅助软件工程系统(Computer Aided Software Engineering, CASE)。这标志着软件开发技术的发展进入到了一个新阶段。

1.5 计算机辅助机械设计概述

计算机辅助机械设计的任务是开发集设计计算、资料处理、数据管理及自动绘图等多项功能于一体的应用软件。

1. 计算机辅助机械设计软件开发需要解决的主要问题

- (1) 软件功能的确定;
- (2) 功能模块的划分;
- (3) 软件开发平台(计算机语言)的选择;
- (4) 功能模块的连接;
- (5) 功能模块之间的数据传递;
- (6) 设计资料的处理;
- (7) 参数的输入、分析、判断和调整;
- (8) 程序运行的安全保障。

2. 计算机辅助机械设计软件的主要设计步骤

通常情况下,先列出手工设计步骤或整个详细的设计过程,据此确定软件设计步骤,具体过程包括以下几个方面:

- (1) 熟悉设计的内容、要求、任务、适用范围和功能;
- (2) 熟悉设计的计算准则、计算方法和数学模型;
- (3) 熟悉设计的计算步骤、计算公式、参数符号及变量;
- (4) 根据设计的计算内容、方法、步骤,绘出程序结构流程图;
- (5) 确定功能模块的结构图;
- (6) 逐个设计每个功能模块。

3. 计算机辅助机械设计软件的设计方法

由于目前可视化软件已经占主导地位,因此,计算机辅助机械设计软件也应该是可视化的应用软件。可视化软件的设计与传统的DOS软件相比有许多优点,此处不再赘述。这里主要介绍可视化机械设计应用软件开发的一些要点。

1) 选择开发平台

选择开发平台,需要考虑具体的机械设计任务内容。一方面,机械设计任务内容比较复杂,比如,需要大量计算、查询大量的数表和线图,绘制零件工作图等;另一方面,每种计算机开发语言都有特长与不足,而且不同开发人员的基础各有差别。因此选择开发平台要综合考虑上述三方面因素。就开发语言来说,VC(Visual C)语言的功能最强大,适合开

发操作系统软件及一些综合功能非常强大的软件，但学习这种语言要达到实战的程度，所需要的学习时间也最多。VB(Visual Basic)语言的功能不如 VC 的强大，但在计算功能方面最优秀，语法也贴近生活，简单易学，达到实战所需要的学习时间最少。VF(Visual Foxpro)的综合功能也不如 VC，简单易学性虽说不如 VB，但它的数据库管理功能却是最强大的。VL(Visual Lisp)在自动绘图方面则最优秀。这些可视化语言都比较流行，用这些语言开发的软件当然也会容易受到用户的欢迎。为了便于选择，现把这几种语言的主要特点列于表 1-3，供读者参考。

表 1-3 几种可视化语言特点比较

语言名称	易学易用性	计算功能	数据库管理功能	工程绘图功能	综合评价
VC	4	2	2	2	3
VB	1	1	3	3	1
VF	2	3	1	4	2
VL	3	4	4	1	4

注：1—优；2—良；3—中；4—差。

实际上，根据具体的机械设计任务的不同、开发语言特点的不同、开发人员基础的不同，选择的开发语言都会有所区别。对于非软件专业的技术人员来说，出于解决面临的机械设计问题之需要，可优先选用简单易学的语言。而且，如果一种语言不够，可多采用几种语言，分模块分别开发。最后再把各模块组合起来，这样可以取长补短，“多、快、好、省”地开发出自己所需的应用软件，本课程的重点也就是介绍这种开发方法。

2) 在单平台上开发软件时的设计要点

在一种开发平台(比如 VB)上完成全部设计内容，主要考虑以下几点：

(1) 在弄清手工完成机械设计过程的基础上，确定软件的输入/输出参数及全局变量、全局过程或函数。

(2) 分配窗体功能及其接收的输入/输出参数，并进一步确定所需窗体的总数及窗体变量。

(3) 从第一个窗体开始，根据窗体功能、窗体的输入/输出参数，进行窗体的外观设计，然后逐个确定并编写该窗体的每个事件过程以及每个一般过程或函数的代码，进行调试，排除所有错误后，再设计第二个窗体。在设计第二个窗体时，要确定第一、第二两个窗体的连接交替方法。将第一、第二两个窗体合起来调试，排除一切错误后，再设计第三个窗体，把这三个窗体合起来调试，依此类推，直至全部调试完毕。

3) 在多平台上开发软件时的设计要点

(1) 在弄清手工完成机械设计过程的基础上，分配每个开发平台上应完成的功能，并确定所需的软件输入/输出参数。

(2) 按照要点 2)，完成单平台上的开发任务，并通过编译，生成可在 Windows 环境下独立运行的可执行文件(扩展名为 exe 的文件)。根据功能模块结构图中确定的前后运行(调用)关系，在前后两个开发平台上分别编写程序时，都要协调考虑并确定在这两个平台上开发的可执行文件之间的数据相互传递的方法。

(3) 将前后两个不同平台上编译的可执行文件合起来调试。