

嵌入式系统 课程设计

贾世祥 俞建新 马小飞 肖建 编著

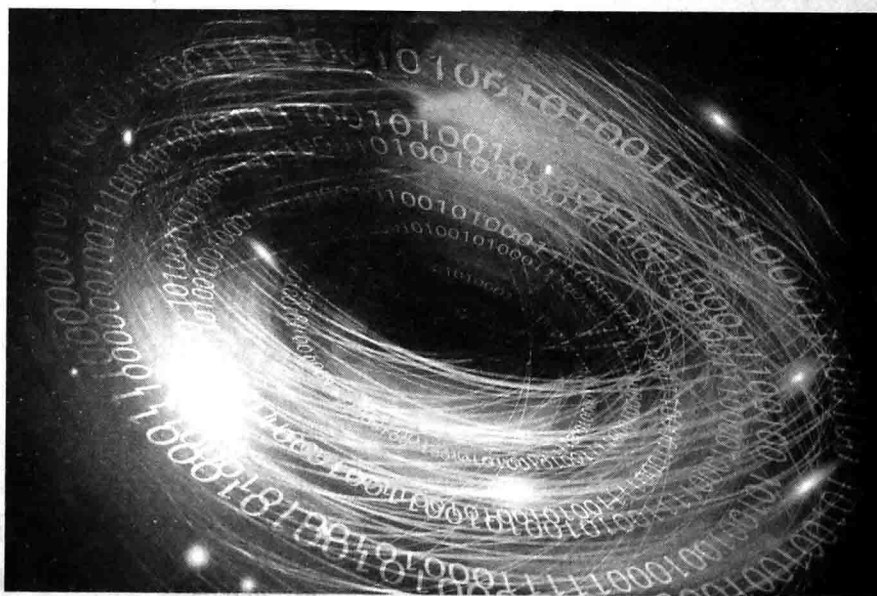


机械工业出版社
China Machine Press

高等院校计算机课程设计指导丛书

嵌入式系统 课程设计

贾世祥 俞建新 马小飞 肖建 编著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

嵌入式系统课程设计 / 贾世祥等编著. —北京: 机械工业出版社, 2015.3
(高等院校计算机课程设计指导丛书)

ISBN 978-7-111-49637-3

I. 嵌… II. 贾… III. 微型计算机—系统设计—课程设计—高等学校—教材 IV. TP360.21-41

中国版本图书馆 CIP 数据核字 (2015) 第 050573 号

本书是基于 VxWorks 操作系统的嵌入式系统课程设计参考书。首先介绍 VxWorks/Tornado 以及 WindML 的基本理论和应用知识, 概述嵌入式系统课程设计的一般定义、特点、分类、基本结构和开发步骤。随后介绍控制类、模拟器类、网络游戏类课程设计的基本解决方案、特色技术和实验要点。本书重点讲解了汉字显示可编程逻辑扩展板、列车自动售票机模拟器和联网跳棋这三个嵌入式综合课程设计的实验案例, 详细说明了这三个案例的选题、功能设计和运行原理、数据流设计、任务划分、函数流程分析、IPC 设计、详细代码分析、测试设计、测试用例和测试结果报告、运行结果展示等。为了方便读者的阅读和实验应用, 本书附有教辅资料 (可通过华章网站 www.hzbook.com 免费下载), 内含正文阐述的 VxWorks 课程设计案例的工程文件。

本书可作为高等院校研究生、本科生嵌入式系统教学或者案例教学的教材, 也可作为工程技术人员和嵌入式系统爱好者的参考读物。

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 余 洁

责任校对: 董纪丽

印 刷: 北京诚信伟业印刷有限公司

版 次: 2015 年 4 月第 1 版第 1 次印刷

开 本: 185mm × 260mm 1/16

印 张: 16

书 号: ISBN 978-7-111-49637-3

定 价: 39.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

前 言

嵌入式系统课程的教学内容大致包括：嵌入式处理器、现场可编程逻辑器件设计、嵌入式操作系统、嵌入式调试技术、外部设备与驱动、软硬件协同设计、典型应用项目、高级语言程序设计等。这门课程的教学内容多，系统性和综合性的特点较强。只从课堂上学习这些理论知识是远远不够的，需要通过各种课程内和课程外的实验活动来深化、夯实、巩固对嵌入式理论与知识的理解，进而掌握嵌入式系统设计和开发的主要方法和技能。因此，实验是嵌入式系统课程的重要环节。

一般而言，嵌入式系统教学实验活动可以分为单项实验、课程设计和自主创新实践三个类别。

1. 单项实验

单项实验是基础实验，属于软硬件工作原理的验证性实验。单项实验的典型代表有**硬件实验**、**软件实验**和**软硬件实验**。硬件实验主要指单个外部设备的裸板驱动程序和实验板的引导加载程序实验。软件实验主要指操作系统的特别函数功能实验，还包括高级语言的特别函数功能实验。软硬件实验主要涉及外设驱动和控制程序的综合实验，例如在开发板的 LCD 模块上显示表达式和计算结果的计算器实验，以及多个开发板联网的黑白棋实验。

通过单项实验，学生可以在嵌入式实验平台上验证理论知识、工作原理、编程技巧和技术解决方法。以基于 ARM 处理器和 VxWorks 操作系统的实验平台为例，单项实验可以包括：Tornado 集成开发环境实验、WindSh 命令实验、VxWorks 组件裁剪实验、模拟器 VxSim 应用实验、软件逻辑分析仪 WindView 实验、实时多任务实验、字符设备驱动实验、网络驱动实验、IPC 实验、闪存实验、WindML 图形用户界面实验等。

单项实验的特点是面向技术细节，引导学生逐步掌握硬件器材的运作机制和函数编程原理，训练学生掌握嵌入式开发过程中的一个单项作业。单项实验所需要的实验时间长短不一。举例来说，工具和操作界面类型的实验借助实验指导书可以在几分钟内完成，含有源代码的实验则需要学生投入较多的时间理解和消化。例如，一个 LCD 驱动程序实验大约需要学生一到两天的时间进行源代码阅读，这样才能完全理解该 LCD 控制器各个寄存器的作用，以及 LCD 驱动程序的架构和编程方法。

单项实验有许多不足。首先，如果学生没有足够的时间通读源代码，就容易造成即使得到了正确实验结果，但也不理解程序内部流程的“知其然不知其所以然”现象。此外，单项实验的程序多半是实验平台供应商已经开发好并且经过调试、排错的。直接阅读经过他人调试好的程序，而不经调试和排错，实验者对现有程序的认识就比较肤浅，容易停留在表面，无法深入理解程序代码的细节与技巧。为了克服单项实验的局限性，需要让学生掌握单项实验编程的要领，如让实验者对于每一个单项实验至少完成一个替换练习或者修改练习。例如，变换高分辨率 LCD 输出画面的外观，让学生理解图形用户界面相关函数的作用、功能和编程

方法。又例如，对包含三个外设驱动的组合单项实验程序，添加另外两个独立的外设驱动，做到在集成开发环境下编译通过，并且要求这5个外设都能够在调度程序的协调控制下正常动作。这样就让学生理解了开发板上裸板运行系统的生成原理，它是经过对多个单独外设驱动程序进行集成和控制而获得的。

2. 课程设计

一般认为，嵌入式系统**课程设计**是一种学生在学习期间，结合在嵌入式系统课程中学习得到的理论知识，自主独立完成的较大编程规模的应用实验。所以也可称之为嵌入式系统**课程设计实验**，或者嵌入式系统**综合课程设计实验**。上述这三个术语的内涵基本是一致的，都表达了一个意思，即课程设计是嵌入式系统教学中的一个重要实验环节。

参考百科知识门户网站（例如百度百科、互动百科）的定义，嵌入式系统课程设计是“为掌握嵌入式系统课程教学内容所进行的实验设计或者实践”。它的含义大致与嵌入式系统综合实践工程（或者嵌入式系统综合实验工程）的含义相当。

我们认为，对于高等院校IT专业的学生而言，嵌入式系统课程设计实验就是让学生综合在校期间已经学习到的各门课程知识和技能，针对一个简单的实际项目解决方案或者一个简单的应用原理，开动脑筋，完成项目构思以及实验方案设计、开发、测试和文档制作的全套作业，以提高应用能力和动手操作能力。由于嵌入式系统课程设计（实验）或者综合实践工程都应该在一个学期的一门课程里完成，两者的含义非常接近，可以认为一门课的课程设计就是这门课的综合实践工程，反之亦然，因此，在本书里，我们不加区别地使用这两个名称。

课程设计实验是单项实验项目的集成实验或者综合应用实验。例如开发一个在嵌入式开发板上运行的图形数字时钟，就需要在高分辨率的液晶屏上绘制一个具有交互功能的指针式时钟和数字时钟，它是键盘驱动、LED七段数码管驱动、LCD驱动、触摸屏驱动、RTC编程、GUI编程、控制程序编程等多个单项实验的无缝衔接与集成。

嵌入式综合课程设计实验具有一定的难度，而且经常具有工程性质和原型开发生质。多数情况下课程设计实验不是单项实验的简单堆砌，可以认为它是介于高等院校的实验项目与专业团队研发项目之间的过渡性实验工程项目。学生在学校经受了课程设计实验的实训之后，到了工作岗位就能够较快地熟悉所面临的工作，独立地完成工作任务。

在课程设计实验的开始阶段，学生通常得不到现成的解决方案和答案。因此让学生自己思考、查阅参考资料、最终完成可行性分析和设计报告是比较好的做法。这要求学生投入较多的时间和精力。在许多情况下，由于课程设计的功能综合性强，工作量比较大，经常需要两人或者更多人组成团队才能够完成。这样就提供了一个团队工作的实验环境，学生可以从中学体验嵌入式开发团队的工作方法。综合课程设计实验一般要求学生在学校课程学习的中间阶段着手进行，并且在全课程结束之前完成。

3. 自主创新实践

自主创新实践是比课程设计或者综合实践层次更高的实验活动，它需要更加综合的理论和实践技能，这往往超越了一个专业课程群的知识范畴，与多个专业的课程相关。以嵌入式系统的自主创新实践为例，它涉及处理器技术、软件技术、电子工程、自动控制技术、通信技术等，工程开发性质更加突出，要求的技术水平更高。

自主创新实践强调应用和实用,要求实验成果能够产生一定的经济效益或者社会效益,至少能够产生一个好的教学案例。例如设计开发一个嵌入式处理器的指令集模拟器就是一个很好的自主创新实践活动。如果开发成功,就有可能投入实际的科研或教学运用中,也有可能作为嵌入式软件工具产品出售。

自主创新实践具有提高开发水平、深入关键技术和注重细节的特点。一个自主创新实践项目往往就是一个产品原型的正式开发。在实践过程中,通常由教师和学生共同组成一个研发团队。这样的团队会提出不低于当前技术水平的解决方案,最终给出一个完整、可运行的工程原型,而且在设计、编码、测试、试运行的各个阶段提供质量合格、文笔流畅、图表规范的文档。

自主创新实践还常常具有产、学、研相结合的特点。企业和科研院所的技术人员也可以加入以教师和学生为主的团队中,参与实践项目开题、项目咨询、项目评估、项目测试和项目鉴定等活动。业界专业技术人员的参与对提升学生的专业技能意义重大,使他们能够接触到嵌入式专业工作者的观点和做法,从更加广阔的视野来考虑实践活动中所遇到的技术创新问题。产、学、研相结合能够使培养出的毕业生具有较强的适应能力,毕业后能够更快地适应自己的工作岗位。

本书的编写背景

本书署名作者当中的前三位是南京大学的校友,第4位是南京邮电大学的EDA实验室高级实验师。从2004年到2013年,四位作者都先后参加过嵌入式系统课程设计的实验活动。

在南京大学计算机系历时近10年的本科生和研究生嵌入式系统课程教学中,使用的硬件嵌入式实验平台之一是CVT2410(创维特公司的ARM9实验箱产品),使用的典型操作系统之一是VxWorks 5.5/Tornado 2.2,使用的与VxWorks配套的图形用户界面是WindML 3.0。在参与嵌入式系统教学和实验的过程中,作者们一再强调实验环节的重要性,并对修课学生参加实验环节给予力所能及的指导和帮助。

在VxWorks的实验教学活动中,我们整理出50多个单项实验,并在此基础上先后提出了近十个课程设计让优秀学生实践,包括:自动洗衣机模拟器、实时时钟、华容道、飞行棋、联网跳棋、自动售票机模拟器等。

由于我国目前嵌入式系统综合课程设计的教材还比较少,而在实际教学过程中学生对这一类图书又有较大的需求,作为嵌入式系统课程的任课教师,我们有责任帮助学生解决综合课程设计的实验用书问题。为了给学生提供一个开发步骤合格、流程正确、能启迪思维的教学用书,我们从2010年起就开始着手制订编写计划,准备对其中的若干个课程设计的工作原理、任务功能、任务划分、任务设计方案和测试方案做出较为全面的总结,写作一本实验项目案例教材。基于这个计划,我们一边致力于嵌入式系统课堂教学,一边致力于实验环节的实践、指导和管理。在陆续积累VxWorks教学和实验的基础上,我们优选了3个比较完善的课程设计作为重点加以剖析和描述,逐步地编写出本书的各个章节。

通过阅读本书,读者可以加深对VxWorks重要知识点的理解,掌握嵌入式系统课程设计的概貌与过程,并借鉴本书的三个课程设计对书中提出的其他典型课程设计项目加以尝试,

提高嵌入式系统设计与开发的技能。

致谢

在本书的编写过程中，参考和借鉴了许多专家学者的研究成果，我们在列举参考文献时可能有疏漏，在此向相关所有者和机构表示诚挚的谢意。

由于作者专业教学水平有限及编写时间紧，书中难免有不妥甚至错误之处，恳请读者批评、指正和评价，与我们一起对书中内容进行完善。我们对来自读者的评价、批评和建议信息表示欢迎和致谢，将认真地考虑并酌情处理。

反馈信息请通过以下方式与作者联系：

E-mail: jiashixiang@sohu.com, yujianxin@nju.edu.cn

编 者

目 录

前言

第 0 章 引言 1

0.1 对读者基本知识和技能的要求 1

0.2 对读者自行开展课程设计的期望 2

第 1 章 VxWorks/Tornado 概述 3

1.1 VxWorks 基本组成 3

1.2 VxWorks 特点 4

1.3 Tornado 概述 5

1.4 VxWorks 任务管理 6

1.4.1 任务状态转换 7

1.4.2 任务框架 8

1.4.3 任务相关 API 函数 9

1.4.4 任务相关 API 函数使用 范例 10

1.5 VxWorks 任务间通信 12

1.5.1 信号量 13

1.5.2 信号量语句编程实验 14

1.5.3 消息队列 16

1.5.4 消息队列语句编程实验 19

1.5.5 管道 19

1.5.6 信号 (软中断) 20

1.5.7 进程间通信函数综合使用 范例 22

1.6 VxWorks 中断机制 27

1.7 VxWorks 设备驱动 28

1.7.1 VxWorks 设备描述符 29

1.7.2 VxWorks 的 I/O 系统 29

1.7.3 基本 I/O 接口 30

1.7.4 安装 VxWorks 驱动程序 31

1.8 VxWorks 图形组件 32

1.8.1 WindML 3.0 32

1.8.2 通用图形库 33

1.8.3 WindML 基本知识 34

1.8.4 UGL 获得输入设备信息的 详细解释 35

1.8.5 典型 WindML 绘图程序 结构 36

1.8.6 UGL 常用的窗口函数 38

1.9 小结 40

第 2 章 嵌入式系统课程设计概述 41

2.1 嵌入式课程设计的基本特点 41

2.2 嵌入式课程设计的教学目标 42

2.3 嵌入式课程设计的基本分类 42

2.3.1 控制类课程设计 42

2.3.2 模拟器类课程设计 43

2.3.3 通信类课程设计 43

2.3.4 地理位置服务类课程设计 44

2.3.5 游戏类课程设计 46

2.3.6 实用工具类课程设计 46

2.4 嵌入式课程设计的基本步骤 46

2.4.1 选题说明书 46

2.4.2 可行性分析 48

2.4.3 概要设计 49

2.4.4 任务划分 58

2.4.5 常用编码规则 58

2.4.6 头文件编程范例 60

2.4.7 代码优化 62

2.4.8 测试方案设计和测试用例 65

2.4.9 测试报告 68

2.4.10 实验文档编写指导 69

2.5 小结 70

第 3 章 控制类课程设计 71

3.1 LED 点阵汉字滚行显示扩展板 71

3.1.1 软硬件开发环境 71

3.1.2 硬件功能分析与设计 71

3.1.3 扩展板电路设计	77	4.6.2 触摸屏任务的处理流程.....	135
3.1.4 汉字显示原理	80	4.6.3 计算任务的处理流程	137
3.1.5 软件功能分析与设计	82	4.6.4 输入任务的处理流程	142
3.1.6 驱动程序设计	87	4.6.5 输出任务的处理流程	146
3.1.7 测试方案设计	90	4.6.6 键盘任务的处理流程	150
3.1.8 测试应用程序设计	90	4.6.7 LED 数码管任务的处理 流程	150
3.1.9 CPLD 程序注解	92	4.6.8 自动售票机模拟器的主要 函数	152
3.1.10 VxWorks 驱动程序注解 ..	95	4.7 测试方案设计	154
3.1.11 驱动程序测试步骤和 测试结果	109	4.8 异常处理测试方案设计	155
3.2 其他实现方案	110	4.9 测试结果和故障排除	159
3.3 思考题	111	4.10 思考题	160
3.4 替换练习	111	4.11 联网自动售票机模拟器	161
3.5 小结	111	4.12 小结	161
第 4 章 模拟器类课程设计	112	第 5 章 网络游戏类课程设计	162
4.1 列车自动售票机模拟器	112	5.1 跳棋溯源	162
4.1.1 概述	113	5.2 联网跳棋电子游戏简介	163
4.1.2 自动售票机软硬件环境 需求	113	5.2.1 软硬件环境	163
4.1.3 自动售票机用户需求	113	5.2.2 主要功能	163
4.1.4 功能分析和设计	115	5.3 跳棋游戏运行指南	164
4.2 数据分析	116	5.3.1 运行前的准备工作	164
4.2.1 主要结构体数据变量定义	117	5.3.2 游戏操作说明	164
4.2.2 数据流分析	118	5.4 自然人走棋版本系统设计	168
4.3 任务划分和定义	119	5.4.1 主界面设计	168
4.4 人机交互设计	120	5.4.2 棋位极坐标系设计	170
4.4.1 主函数 progStart 设计	125	5.4.3 绘图坐标系设计	171
4.4.2 usrAppInit 函数	126	5.4.4 主要数据结构和数组赋值 ..	173
4.4.3 模拟钱币输入时间限制 处理	127	5.4.5 颜色确定算法	175
4.5 进一步理解 TTVM 程序	128	5.4.6 旋转映射算法	176
4.5.1 完整源代码清单阅读指南 ..	128	5.4.7 走棋规则	178
4.5.2 任务处理流程的着色说明 ..	128	5.4.8 光标改变及走棋设计	178
4.5.3 二进制信号量使用分析	128	5.4.9 可达棋位搜索算法	179
4.5.4 消息队列使用分析	129	5.4.10 游戏胜利的判断及排名 和记录步数算法	180
4.5.5 其他 IPC 语句使用	131	5.4.11 判断当前和下一步该谁 走棋的算法	180
4.6 任务执行流程分析	131	5.4.12 程序模块划分	181
4.6.1 LCD 任务的处理流程	131		

5.4.13	任务划分	181	5.7.4	改造后的程序测试及 算法评估	215
5.4.14	数据包结构描述	181	5.8	编码过程中的故障排除	217
5.4.15	网络通信处理流程	182	5.9	设计测试方案并进行测试	217
5.4.16	光标闪烁的实现及信号 的使用	182	5.9.1	测试方案	217
5.4.17	程序间通信	183	5.9.2	测试实施	218
5.4.18	图形界面绘制	183	5.9.3	6 个自然人玩家的实测 记录	218
5.4.19	函数	184	5.9.4	4 个自然人玩家的实测 记录	219
5.5	任务函数处理流程	187	5.9.5	2 个自然人玩家和 1 个 机器玩家的实测记录	221
5.5.1	progStart 任务	187	5.9.6	2 个机器玩家的实测记录	221
5.5.2	taskGame 任务	188	5.9.7	2 个自然人玩家和 4 个 机器玩家的实测记录	222
5.5.3	taskGlint 任务	189	5.9.8	6 个机器玩家的实测记录	222
5.5.4	taskRecv 任务	190	5.10	机器棋手博弈走棋算法	223
5.5.5	taskTick 任务和 taskTimeShow 任务	192	5.10.1	博弈走棋的图例说明	224
5.6	机器走棋版本系统设计	192	5.10.2	红色玩家的优选走棋 路径	225
5.6.1	跳棋游戏平台描述	193	5.10.3	黄色玩家的优选走棋 路径	226
5.6.2	可判优坐标系设计	193	5.10.4	绿色玩家的优选走棋 路径	227
5.6.3	机器走棋的棋盘坐标系 变换	194	5.10.5	浅蓝色玩家的优选走棋 路径	229
5.6.4	智能走棋策略	196	5.10.6	深蓝色玩家的优选走棋 路径	230
5.6.5	数据结构改动	198	5.10.7	紫色玩家的优选走棋 路径	231
5.6.6	程序模块改动	198	5.10.8	当前玩家敏感走法案例 图解	232
5.6.7	taskCMP 任务工作原理 及流程图	198	5.10.9	博弈算法基本处理流程	233
5.6.8	开局阶段走棋函数说明	199	5.11	功能扩展练习和替换练习	237
5.6.9	中局阶段走棋函数说明	200	5.12	小结	239
5.6.10	收尾阶段走棋函数说明	202	附录	WindML 绘图程序例子	240
5.6.11	循环走棋判断	203	参考文献		246
5.6.12	程序间通信	204			
5.6.13	每步走棋等待时间设置	205			
5.6.14	增加的函数	205			
5.7	机器走棋版本编程指南	206			
5.7.1	跳棋 B 版本的工程文件	206			
5.7.2	处理流程总控制结构	206			
5.7.3	对模板程序中智能走棋 算法的改造	207			



第0章 引言

读者可以从第1章获得 VxWorks 操作系统任务管理、任务间通信、中断机制、设备驱动以及 WindML 图形组件的原理性知识；可以从第2章获得嵌入式系统课程设计的基本特点、基本分类、基本开发步骤等理论知识。

第3章到第5章分别介绍了3个嵌入式系统实验案例。其中第3章介绍了一个控制类课程设计方案，即在一款 ARM 9 实验平台上，借助系统总线插拔槽，自行开发一个 CPLD 扩展板插件，实现 LED 点阵的驱动；LED 点阵显示板插在 CPLD 扩展板上，能够滚动显示 VxWorks 应用程序输出的单行汉字。第4章介绍了一个机械装置模拟器类课程设计方案，它是一个用于铁路售票的自动售票机模拟器，属于 VxWorks 的原型软件。该章描述了这个模拟器的基本设计、数据流图、任务划分、处理流程、人机对话界面、绘图程序流程、IPC 设计以及测试结果。第5章介绍了一个联网运作的电子游戏软件。它是一个可供6人博弈的电子跳棋，含有两个程序。第一个程序是供自然人玩家操作的跳棋程序，可实现自然人玩家之间的跳棋博弈。如果该联网的实验板上没有自然人玩家，就自动地运行第2个机器棋手操作的跳棋程序，实现机器棋手与其他棋手之间的跳棋博弈。

0.1 对读者基本知识和技能的要求

为了很好地理解本书的前两章，读者应该具备下列知识和技能：

- 1) ARM 9 处理器的结构、汇编语言程序设计。
- 2) VxWorks 操作系统的基本概念、任务管理函数、IPC 函数等。
- 3) C 语言程序设计的基本规则。
- 4) 课程设计的性质、分类以及各种类型课程设计的基本特点和基本应用。

读者具备较强的理解力和动手能力是透彻理解本书后三章所必需的。需要具备的知识和技能以下分别叙述。

为了很好理解第3章控制类课程设计，读者应该具备 CPLD 和 FPGA 的设计技能、汉字字库芯片的选型和使用方法、VxWorks 驱动程序设计、VxWorks 应用程序设计、ARM 实验平台的操作和调试，等等。

为了深刻理解第4章模拟器类课程设计，读者应该具备网络应用系统设计、数据组织和访问、系统设计、WindML 绘图程序设计、VxWorks 应用程序设计、ARM 实验平台的操作、嵌入式测试和故障排除，等等。

为了掌握第5章网络游戏类课程设计，读者应该具备棋牌类电子游戏程序的框架设计、

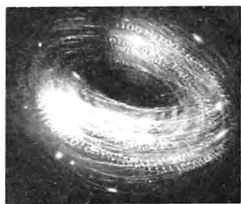
网络应用系统设计、棋类电子游戏的坐标系统设计、人机博弈系统设计、WindML 绘图程序设计、VxWorks 应用程序设计、ARM 实验平台的操作、嵌入式测试和故障排除，等等。

0.2 对读者自行开展课程设计的期望

阅读本书之后，高年级学生读者应该以书中的三个课程设计案例为参照物，与其他学生组成一个开发团队，启迪自己或者他人的创意思维，提出项目模型，进行交流讨论。用演绎、模仿、借鉴的方式完成一个类型相同、工作量相当、具有新意的课程设计。

例 1 可以根据北京—沈阳或者上海—杭州的高速铁路，尝试编写一个高铁自动售票机模拟器。要求完成源代码设计和调试、文档编写、使用说明等。

例 2 可以尝试设计一个自动洗衣机模拟器，用图形方式给出洗衣机的控制面板、工作状态，用户使用小键盘或者触摸屏控制洗衣机的动作，步进电机模仿洗衣机转鼓的快慢速旋转和正转 / 反转 / 定时转动，程序调通之后，再编写一个合格的设计报告、含注释的源代码清单，以及用户说明书。



第 1 章

VxWorks/Tornado 概述

VxWorks 是美国风河公司 (Wind River Inc.) 于 20 世纪 80 年代中期投放市场的嵌入式实时操作系统。它是一个强实时的嵌入式操作系统, 经过多年的实际使用没有发生过重大事故, 其高可靠性得到业界的公认。在航空航天、军用、通信和医疗等领域, VxWorks 拥有最大的市场份额, 成为这些领域事实上的操作系统标准。

VxWorks 操作系统能够在多种主流的 CPU 硬件平台上向开发者提供统一的 API 接口。1997 年风河公司为 VxWorks 提供了可视化的集成开发环境 Tornado。这样, VxWorks 程序员就有了一个 API 接口丰富、操作方便、高度专业化的开发环境。现在, Tornado 成为 VxWorks 的标准开发工具。借助 Tornado 集成开发环境, 程序员能够高效快速地开发出高质量的嵌入式应用程序。

WindML 是风河公司为 VxWorks 操作系统配套研发的多媒体组件, 它能为嵌入式系统提供图形用户界面。WindML 已经成为 VxWorks 嵌入式研发项目常用的图形工具。

2009 年 6 月上旬, Intel 公司全资收购了风河公司。这个行动表明 Intel 的市场战略从传统的 PC 和服务器的延伸到了嵌入式系统和便携手持设备。目前, 风河公司作为 Intel 公司的子公司仍然从事嵌入式操作系统开发, 包括嵌入式 Linux 和 VxWorks/Tornado 的技术维护与技术研发。风河公司于 2014 年推出了 VxWorks 7, 对物联网应用提供了更完备的支持。风河公司还推出了新的集成开发环境 WorkBench 来取代 Tornado。WorkBench 以开放的 Eclipse 平台为框架, 开发环境可由客户自由定制。例如, 用户想使用自己熟悉的配置管理工具或者编辑器, 就可以找到这样一个插件并集成到 WorkBench 中。

本书介绍的 VxWorks / Tornado 版本分别是 VxWorks 5.5 和 Tornado 2.2。除此之外, 还介绍了 WindML 3.0 图形组件的函数功能。

1.1 VxWorks 基本组成

VxWorks 属于大型操作系统, 是 UNIX 内核改良型操作系统。这意味着 VxWorks 的内核模块、设备驱动、I/O 管理、Shell 命令和 Makefile 命令与 UNIX 和 Linux 相兼容。在 I/O 管理方面, 它与 ANSI C 兼容, 包括 UNIX 标准的缓冲 I/O 和 POSIX 标准的异步 I/O。

VxWorks 由以下 4 大部分组成。

- 1) Wind 内核。VxWorks 的核心模块通常简称为 Wind, 包括任务管理、抢占式多任务调度、进程间通信机制、中断处理、看门狗定时器和内存管理机制。
- 2) I/O 系统。VxWorks 的输入 / 输出系统由兼容 ANSI C 的代码编写, 包含了 UNIX 的 I/O

系统和 POSIX 的 I/O 系统，还包含了网络驱动和各种外设驱动。

3) 文件系统。VxWorks 支持的文件系统有：

- dosFS，与 MS-DOS 兼容的文件系统。
- rawFs，将整个盘作为一个文件，允许根据字节偏移读写磁盘的一部分。
- TrueFFS，闪存文件系统。

4) 板级支持包 (BSP)。对各种嵌入式电路板提供一个统一的软件接口库。

1.2 VxWorks 特点

VxWorks 具有以下特点：

(1) 具有强实时性能

VxWorks 在 80486 处理器 66MHz 主频下的任务切换时间 (Time of Context, TC)、系统调用时间 (Time of System, TS) 和中断响应时间仅为几微秒。

(2) 支持 POSIX 标准

POSIX 即可移植操作系统接口 (Portable Operating System Interface)，是 IEEE 提出的技术标准。POSIX 的目标是使应用程序源代码可以在兼容 POSIX 的操作系统上移植。理想的目标是应用程序移植到另外一个操作系统只需要重新编译就可以了。VxWorks 正在不断完善对 POSIX 的支持。

(3) 微内核设计方法

基本的操作系统功能由 Wind 内核提供，其他系统功能以系统组件形式存在。

(4) 可裁减性

提供高度的可裁减性，可供裁减的组件多达 300 个以上。

(5) 可移植性

VxWorks 操作系统分为两部分，即硬件相关部分和硬件无关部分。硬件相关部分程序由 BSP 提供。由于 BSP 的单独存在，移植 VxWorks 的开发工作变得十分明确。目前，中小型 VxWorks 应用开发商都能够移植 BSP。

(6) 抢占式任务调度

VxWorks 具有抢占式任务调度机制，支持基于优先级的任务调度和循环优先级的任务调度。

(7) 可靠性

VxWorks 经过了近 30 年的市场应用验证，核心代码很长时间没有修改，即没有发现错误。这证明 VxWorks 操作系统是高度可靠的。

VxWorks 技术一直在不断发展之中。VxWorks 6.8 版能向客户提供完整的移动 IP 支持，能为新一代网络通信设备提供包括 LTE (长期演进) 和 WiMAX 技术在内的 4G 通信或者准 4G 通信的处理能力；此外，还提供了对多核处理器支持，包括 Intel Core i7 处理器、ARM Cortex-A9 处理器、Freescall QorIQ P2020 处理器、Cavium 54xx/55xx/56xx/57xx/58xx 系列处理器和 RMI XLR/XLS 系列处理器等，客户能够根据实际需求灵活选择最合适的处理器。2014 年，风河公司推出最新的 VxWorks 7 来帮助嵌入式设备制造商应对物联网 (Internet of

Things, IoT) 时代的挑战, 满足其对连通性、可伸缩性和安全性的要求。

1.3 Tornado 概述

Tornado 是 VxWorks 的可视化集成开发环境。与 VxWorks 5.5 相配套的 Tornado 版本是 2.2。它集编辑器、编译器、调试器于一体, 为嵌入式系统开发人员提供了一个不受目标机资源限制的开发调试环境。Tornado 主要运行在 PC 的 Windows 操作系统之上, 具备 Windows 风格特点的友好人机交互界面。

Tornado 集成了多种开发工具, 可在不同的应用开发阶段使用, 并且对于各种目标机连接策略(以太网、串行口、片上调试)和不同的目标机内存都适用。Tornado 包含 3 个高度集成部分: ①强有力的交叉开发工具和实用程序; ②在目标机上执行的高性能、可裁减的 VxWorks 实时操作系统; ③连接宿主机和目标机的多种通信工具, 如以太网、串口线、ICE 或 ROM 仿真器。

Tornado 的主要工具包由以下几项构成:

(1) 集成的源代码编辑器

Tornado 的源代码编辑器具有以下特点:

- 标准的文档处理功能。
- C 和 C++ 语法关键字的突出显示。
- 调试程序时追踪代码执行。
- 编译链接程序时错误及警告信息显示。
- 不支持汉字输入, 但支持汉字显示。

(2) 工程管理工具

Tornado 工程管理工具提供一个可视化的操作界面。程序员使用它建立工作空间, 在工作空间中建立可启动或者可加载的 VxWorks 工程。有了工程管理工具, 组织、配置、构造 VxWorks 的应用开发变得更为方便。

(3) 集成的 C/C++ 编译器

Tornado 的 C/C++ 编译器有两种: GNU 和 Diab。

GNU 编译器是以 GPL 以及 LGPL 许可证发行的自由软件套件, 来源于自由软件基金会(Free Software Foundation, FSF)。包括: cpp, C 预处理程序; gcc, C/C++ 编译器; ld, 目标代码链接器; make, 程序编译链接批命令工具。

Diab 是风河公司自主研发的商用 C/C++ 编译器。某 VxWorks 程序员在因特网上做出了这样的评价: Diab 的编译速度慢一些, 生成的汇编代码效率比 gcc 高一些。

(4) 仿真器 VxSim

在主机上仿真目标机的运行, 属于指令集模拟器工具。

(5) 调试器 CrossWind

CrossWind 是 Tornado 的源码级调试器, 提供图形和命令行两种调试方式, 可进行符号反汇编、任务或系统级的断点设置、单步执行及调试异常处理等。CrossWind 是 GNU 源代码调试器(GDB)的一个扩充版本, 来自自由软件基金会的可移植符号调试器。CrossWind 对 GDB 的主要扩充是图形接口。

(6) 软件逻辑分析仪 WindView

在调试程序时，提供图形化的中断、上下文切换和任务阻塞等信息显示。

(7) 行命令窗口 WindShell

WindShell 是 Tornado 统一的命令解释器接口，也简写成 WindSh。WindSh 允许用户与目标机交互。它可以直接解释执行几乎所有的 C 语句表达式，包括函数调用和函数名称在系统符号表变量中的引用。WindSh 的符号调试使下面的操作更容易：任务断点；任务单步；符号反汇编；符号产生和变量观察；内存观察和修改；异常陷入；栈检查。

(8) 系统对象检查工具 Browser

Browser 是 Tornado 命令解释器 WindSh 的图形化工具，可以显示目标机中的系统对象，如任务、信号灯、消息队列、内存分区、定时器、模块、变量、堆栈等系统信息，也可以显示内存的使用信息。

(9) Auto Scale 功能按钮

Tornado 集成开发环境具有自动增减组件功能。在菜单上按下 Auto Scale 功能按钮就开始执行对可启动工程的用户应用代码进行分析。它能够辨识可清除的组件并自动裁剪，或者加入当前还没有包含进去的必要组件。参看图 1-1。这个向导功能为程序员提供了方便，缩短了程序员配置 VxWorks 的操作过程。

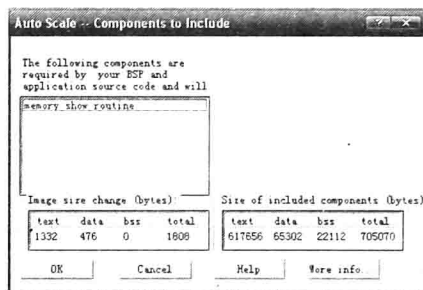


图 1-1 Tornado 的组件自动增减对话框

Tornado 集成环境可以提供上述所有工具，不受目标机资源约束。Tornado 主机工具与目标机系统的通信关系如图 1-2 所示。Target Server (目标服务器) 管理所有与目标机交互的工具、主机与目标机的通信、目标模块的加载和卸载以及主机上的目标机符号表。

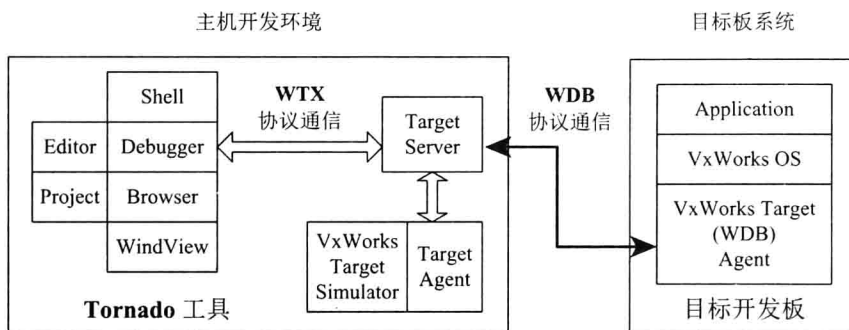


图 1-2 Tornado 主机工具与目标板系统的通信关系

1.4 VxWorks 任务管理

VxWorks 实时内核 Wind 提供基本的多任务管理和调度。在 VxWorks 中，每一个任务拥有任务名称、标识 (ID)、优先级、任务控制块 (TCB) 和栈等私有资源。

任务控制块用来描述一个任务，每一个任务都与一个 TCB 关联。任务控制块中包含：

- 1) 任务的程序计数器。

- 2) 处理器的通用寄存器、浮点寄存器。
- 3) 局部变量和调用函数时使用的栈。
- 4) 标准输入 / 输出和错误输出的 I/O 重定向。
- 5) 一个延时定时器。
- 6) 一个时间片定时器。
- 7) 优先级。
- 8) 等待的事件或资源。
- 9) 任务程序码的起始地址。
- 10) 初始堆栈指针。

此外，任务控制块还包括：

- 任务“上下文”。任务调度器根据 TCB 中的内容来对任务进行管理和调度。此外，TCB 还被用来存放任务的上下文 (context)。任务上下文就是当一个执行中的任务被停止时，所要保存的所有信息。通常，上下文就是计算机当前的状态，即处理器中各个寄存器的内容。
- 任务名称和任务 ID。VxWorks 在创建一个任务时一般要指定一个任务名称，任务名称可以是任意长度的 ASCII 字符串。如果不指定任务名称的话，VxWorks 会按命名规则分配一个默认的任务名：所有从目标机启动的任务以字母 t 开头，后面跟一个自增的序号；而从主机启动的任务以字母 u 开头。

任务 ID 是 TCB 的地址指针，通常是 4 字节的无符号整数。当创建任务时如果创建成功，VxWorks 将返回一个指向 TCB 的任务 ID。另外也可通过 taskNameToId 和 taskName 两个函数实现任务 ID 和任务名称之间的相互转换。

1.4.1 任务状态转换

任务一旦被创建，其状态可以在有限范围内迁移，从而构成一个简单的有限状态机 (Finite State Machine, FSM)。Wind 内核负责维护每个任务的当前状态。VxWorks 的基本任务状态包括：挂起 (Suspend)、就绪 (Ready)、阻塞 (Pend)、延时 (Delay) 和运行 (Running)。

Wind 内核任务状态说明如表 1-1，对应的状态转换如图 1-3 所示。

表 1-1 Wind 内核任务状态

状态符号	描 述
就绪 (Ready)	处于这种状态的任务除了等待 CPU 外，无须等待其他资源
阻塞 (Pend)	由于一些资源不可用而阻塞的任务状态
延时 (Delay)	处于延时计数之中的任务状态
挂起 (Suspend)	此状态主要用于调试，不会约束状态转换，仅仅约束任务的执行
DELAY+S	既处于延时又处于挂起的任务状态
PEND+S	既处于阻塞又处于挂起的任务状态
PEND+T	带有超时值 (timeout) 处于阻塞的任务状态
PEND+S+T	带有超时值处于阻塞和挂起的任务状态
state+I	state 指定的任务状态加上一个继承优先级