



嵌入式 VxWorks 系统开发与应用



王学龙 编著

6.2

 人民邮电出版社
POSTS & TELECOM PRESS



嵌入式 VxWorks 系统开发与应用



王学龙 编著

人民邮电出版社

图书在版编目 (C I P) 数据

嵌入式 VxWorks 系统开发与应用 / 王学龙编著. —北京: 人民邮电出版社, 2003.10
ISBN 7-115-11575-3

I. 嵌... II. 王... III. 实时操作系统, VxWorks IV. TP316.2

中国版本图书馆 CIP 数据核字 (2003) 第 081796 号

内容提要

本书详细介绍了当今流行的嵌入式操作系统 VxWorks, 首先概括 VxWorks 操作系统的基本知识, 如任务管理、任务间通信机制、内存管理以及定时管理等内容, 说明了嵌入式操作系统的实现关键。然后, 结合作者多年的嵌入式系统的开发应用经验, 详细阐述了 VxWorks 系统中 BSP 和应用的开发技巧, 并提供了多个应用实例及分析设计。

本书适用于嵌入式系统开发人员作为参考手册使用。

嵌入式 VxWorks 系统开发与应用

- ◆ 编 著 王学龙
责任编辑 刘 浩
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67132692
北京汉魂图文设计有限公司制作
北京隆昌伟业印刷有限公司印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 21
字数: 509 千字 2003 年 10 月第 1 版
印数: 1-5 000 册 2003 年 10 月北京第 1 次印刷

ISBN7-115-11575-3/TP · 3589

定价: 36.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

前 言

当前企业应用最广泛的三大嵌入式操作系统是 pSOS、VxWorks 和 Embedded Linux。目前, 大批大型高科技企业以及为数众多的中小型企业, 甚至高等院校的实验室, 都将使用重点放在 pSOS、VxWorks 系统上。尤其是 VxWorks 系统, 已经得到了非常广泛的应用, 利用嵌入式 VxWorks 系统开发出来的大型移动通信设备、ATM 交换设备、IP 交换设备、甚至医药仪器, 已经在市场上稳定运行。国内, 这方面的开发起步较晚, 大多数开发人员处于自己摸索阶段, 因此笔者想通过对自己多年开发经验的总结, 帮助开发人员更好地理解 VxWorks 系统, 更快地开发产品。

本书包含 7 章。

● 第 1 章介绍 VxWorks 及其开发环境 Tornado 的特点。

● 第 2 章介绍 VxWorks 系统的基本理论, 全面描述 VxWorks 系统的任务管理、内存机制、中断机制、定时机制、时间机制、系统调用, 并且结合自己的开发经验将描述的重点放在常用的内核机理上, 便于读者迅速掌握 VxWorks 系统的基本知识。

● 第 3 章介绍 VxWorks 系统 BSP 的基本概念, 开发 BSP 是开发 VxWorks 系统的关键。本章讲述如何在 VxWorks 系统上开发 BSP, 并且详细探讨 BSP 与嵌入式系统开发的关系。这是开发 BSP 的入门知识。

● 第 4 章详尽介绍一个具体的 BSP 开发实例。考虑到开发一个具体的 BSP 及其驱动程序具有相当难度, 笔者在本章介绍自己曾开发过的一块硬件主板的 BSP 的开发全过程。向读者展示设计 BSP 的过程, 以及此 BSP 的部分代码, 同时对其做详细的分析。

● 第 5 章简单介绍 VxWorks 系统的开发环境 Tornado, 此环境是 VxWorks 系统的应用程序的开发环境, 是组建 VxWorks 系统工程的管理中心。此部分的内容是为第 7 章阐述 VxWorks 系统应用开发实例做准备的。

● 第 6 章介绍 VxWorks 支持 GNU 的 Gcc 编译器, 为了让读者对 Makefile 以及组建工程有较好的认识, 这里举例向读者介绍 Makefile 的编写方法。

● 第 7 章用较大的篇幅向读者展示 VxWorks 系统的应用开发, 介绍诸多应用程序的开发范例, 这些开发技巧在用户开发过程中是非常实用的, 完全可以直接使用或者移植到用户的实际开发中。

希望读者能够在透彻理解嵌入式系统原理的基础上, 把握当前嵌入式系统应用的巨大商机, 尽早加入到开发我国具有自主知识产权的嵌入式系统中, 为我国的科技发展和繁荣富强作出更大贡献。

由于本人水平有限, 再加上时间仓促, 书中难免有不当之处, 敬请读者见谅并提出批评指正意见。读者在学习本书的过程中, 如有意见或发现问题, 请发信至 ridgel@sina.com。

作者

2003 年 9 月

目 录

第 1 章 概述	1
1.1 嵌入式实时操作系统 VxWorks	1
1.1.1 VxWorks 的应用领域	1
1.1.2 VxWorks 系统的特点	2
1.1.3 VxWorks 的可用主机 / 目标机	4
1.2 Tornado 开发环境	5
1.2.1 Tornado 核心工具	6
1.2.2 WindPower 工具	8
1.3 Tornado 嵌入式开发系统可选组件	10
1.3.1 板级支持包 BSP Developer's Kit	11
1.3.2 虚拟内存接口 VxVMI	12
1.3.3 支持紧耦合共享内存多处理器结构的 VxMP	13
1.3.4 支持紧耦合分布式多处理器结构的 VxDCOM	13
1.3.5 支持松耦合分布式多处理器结构的 VxFUSION	14
1.3.6 闪存文件系统 TrueFFS for Tornado	15
第 2 章 VxWorks 系统基本理论	16
2.1 VxWorks 系统概述	16
2.2 VxWorks 系统内核及组件	17
2.2.1 任务管理	19
2.2.2 任务间通信和同步机制	25
2.2.3 中断机制	32
2.2.4 定时管理机制	33
2.2.5 内存管理	34
2.2.6 I/O 与文件系统	35
2.3 VxWorks 系统开发经验	37
2.3.1 正确划分任务	37
2.3.2 防止任务异常	38
2.3.3 正确运用函数的可重入性	38
2.3.4 使用名称访问资源	39
2.3.5 用户任务优先级确定	39
2.4 VxWorks 系统开发模型概述	39
2.4.1 系统启动	40
2.4.2 应用系统配置	42

第 3 章 VxWorks 系统 BSP 基本概念	47
3.1 BSP 基础	47
3.2 BSP 文件结构	49
3.3 VxWorks 系统的 BSP 开发过程	50
3.3.1 建立 BSP 开发环境	50
3.3.2 编辑修改 BSP 文件	50
3.3.3 生成目标文件 bootrom 和 VxWorks 映像	56
3.3.4 基于 ROM 映像的初始化	57
3.4 BSP 中设备驱动程序的开发	58
第 4 章 VxWorks 系统 BSP 开发实例	60
4.1 MPC8260 处理器的组成与结构	60
4.1.1 基本功能模块	60
4.1.2 内核 603e 的组成	62
4.1.3 SIU 的结构	63
4.1.4 CPM 的模块结构	64
4.2 MPC8260 通信处理模块	66
4.2.1 内部存储空间	66
4.2.2 缓冲描述符 BD	68
4.2.3 参数 RAM	71
4.2.4 快速以太网控制器的功能	72
4.2.5 快速以太网控制器的接收过程	74
4.2.6 快速以太网控制器的发送过程	74
4.3 MPC8260 编程特点	75
4.3.1 数据格式和指令格式	75
4.3.2 指令分类	77
4.3.3 特殊功能寄存器	79
4.3.4 高速缓存控制	80
4.4 BSP 最小系统设计	81
4.4.1 BOOT ROM 配置编程	82
4.4.2 程序存储区 Flash 配置	84
4.4.3 SDRAM 初始化	93
4.4.4 CPU 初始化	96
4.4.5 系统软复位	97
4.5 接口驱动设计	97
4.5.1 MPC8260 SCC1-Ethernet 接口的设计	97
4.5.2 MPC8260 SMC1-RS232 接口的设计	126
4.6 BSP 的调试和测试	136
4.6.1 测试内容	136
4.6.2 测试项目及结果	136

第 5 章 VxWorks 系统开发环境 Tornado	140
5.1 Tornado 开发环境概述	140
5.2 Tornado 开发环境的安装	141
5.2.1 安装 Tornado 开发环境	141
5.2.2 注册 Tornado 开发环境	148
5.3 初步使用 Tornado 环境	152
5.3.1 Tornado 工程的类型	153
5.3.2 启动 Tornado 环境	154
5.3.3 创建工作区和工程	155
5.3.4 添加文件到工作区和工程	158
5.3.5 编译工程	160
5.3.6 下载工程到 VxWorks 目标模拟器	162
5.3.7 在 Tornado Shell 下运行应用程序	165
5.4 监视与调试	166
5.4.1 检查内存消耗	166
5.4.2 软件逻辑分析	167
5.4.3 应用程序调试	168
第 6 章 VxWorks 系统编译器	171
6.1 Make 管理项目概述	171
6.2 编写 Makefile 的规则	172
6.2.1 虚拟目标	173
6.2.2 Makefile 的变量	173
6.2.3 make 的变量	174
6.2.4 隐式规则	175
6.2.5 模式规则	176
6.3 Make 命令	176
6.4 Makefile 实例分析	177
6.5 Gcc 的基本概念	187
6.6 Gcc 命令	189
6.7 Gcc 扩展	192
第 7 章 VxWorks 系统应用实例	194
7.1 VxWorks 系统中的任务划分	194
7.2 任务间通信机制	195
7.3 Wind 内核功能	198
7.4 中断处理	203
7.5 Sockets 通信	210
7.6 任务多实例应用	218
7.7 C++应用	241
7.8 数据报应用	252

7.9 虚拟内存设备驱动	263
7.10 RamDisk 驱动.....	277
7.11 WDB 应用	284
7.12 任务软调度实例一.....	296
7.13 任务软调度实例二.....	310

第 1 章 概 述

在 16 位嵌入式系统应用中，由于 CPU 资源量较少，任务比较简单，程序员可以在应用程序中自己管理 CPU 资源，而不一定需要专用的系统软件。如果嵌入式系统比较复杂并且采用 32 位 CPU 时，情况就不同了。32 位 CPU 的资源量非常大，寻址可以达到 4GB 空间，处理能力也非常强大，可以实现实时多任务并发处理，如果仍然沿用手工编制 CPU 管理程序，面对复杂应用，就很难发挥出 32 位 CPU 的处理能力，开发出高效、可靠的应用系统。

操作系统是整个系统应用的基础，它既要管理硬件又要为软件提供应用接口，所以操作系统的任何缺陷都可能引起严重而且不可预知的问题，如果每个嵌入式应用系统，从系统软件到应用软件都需要开发者自己完成，势必造成人力资源的浪费，延长开发周期，增加开发成本。如果基于嵌入式系统直接开发应用，则可很好地解决这些问题。

1.1 嵌入式实时操作系统 VxWorks

VxWorks 是专门为实时嵌入式系统设计开发的操作系统软件，为程序员提供了高效的实时任务调度、中断管理、实时的系统资源以及实时的任务间通信。程序员可以将尽可能多的精力放在应用程序本身，而不必再去关心系统资源的管理。1983 年，VxWorks 成功推出以来，已顺利应用到航空、航天、医疗、通信等领域，并且在嵌入式领域占有一定的市场份额。目前，VxWorks 已经成为实际上的工业标准和军用标准，大量软硬件厂家都提供基于 VxWorks 的扩展组件，因此，VxWorks 几乎可以在各种 CPU 硬件平台上提供统一的接口和一致的运行特征，应用程序不用做太多的改动就可以运行在各种 CPU 上，为程序员提供了一致的开发、运行环境，避免了重复劳动。

VxWorks 是一种功能强大而且复杂的操作系统，仅仅依靠人工编程调试，很难发挥它的功能并设计出可靠、高效的嵌入式系统，必须有与之相适应的开发工具。Tornado 就是为开发基于 VxWorks 的应用系统而提供的集成开发环境，Tornado 中包含的工具管理软件，可以将用户自己的代码与 VxWorks 的核心系统有效地组合起来，从而轻松、可靠地完成嵌入式应用开发。

1.1.1 VxWorks 的应用领域

嵌入式 VxWorks 系统的主要应用领域如表 1-1 所示。

表 1-1 嵌入式 VxWorks 系统的主要应用领域

应用领域	示 例
数据网络	以太网交换机、路由器、网桥、网络集线器、远程接入服务器、异步传输模式交换机、帧中继交换机

续表

应用领域	示 例
远程通信	电信专用分组交换机 (PBX) 和自动呼叫分配器 (ACDs), CD 交换系统, 蜂窝电话系统、xDSL 和电缆调制解调器
医疗设备	核磁共振扫描仪 (MRI)、正电子成像扫描仪、放射理疗设备、床头监护器
消费电子	个人数字助理、机顶盒、数字电话机、应答机、可视电话、声频设备、交互式设备
交通运输	汽车发动机控制系统、汽车导航系统、交通信号控制系统、高速火车控制系统、防滑测试系统、机载娱乐设备
计算机外围设备	网络计算机、X 终端、冗余磁盘阵列数据存储系统 (RAID)、I/O 控制系统
数字图像	打印机、数字复印机、传真机、多功能外围设备、数字相机
工业	机器人、测试测量设备、过程控制系统、计算机数值控制设备
航空	飞行仿真、航空机舱管理系统、卫星跟踪系统、航空电子设备
多媒体	专业视频编辑系统、电视会议

1.1.2 VxWorks 系统的特点

用户在开发包含复杂的 32 位嵌入式处理器的产品时,需要一个用来连接产品应用程序和底层硬件的操作系统。这种操作系统应该具有以下重要特点:

- 可靠性
- 高实时性
- 可裁剪性
- 可协同工作

VxWorks 操作系统是美国 WindRiver 公司于 1983 年设计开发出的一种嵌入式实时操作系统 (RTOS), 是 Tornado 嵌入式开发环境的关键组成部分。VxWorks 具有多达 1800 多个应用程序接口 (API), 适用范围较广, 可用于简单的应用开发也可用于复杂的产品设计。它可靠性高, 可以用于从防抱死刹车系统到星际探索的关键任务, 而且具有高度的适用性, 几乎可以用于所有流行的 32 位 CPU 平台。

VxWorks 操作系统的基本体系结构如图 1-1 所示。

VxWorks 实时操作系统包括微内核 WIND、高级网络支持、强有力的文件系统和 I/O 管理、C++ 和其他标准等核心功能。这些核心功能还可以与 WindRiver 公司的其他产品以及 WindRiver 公司 320 个合作伙伴的产品联合使用。

1. 高性能的微内核设计

微内核的主要特点 (Wind Microkernel) 有如下几点。

- 高效的任务管理: 数目无明显限制的多任务, 具有 256 个优先级; 具有优先级抢占和时间片轮转调度; 快速、准确的上下文切换; 快速灵活的任务间通讯; 3 种信号量, 包括二进制、计数、有优先级继承特性的互斥信号量; POSIX 管道、消息队列和信号
- 高度可裁剪性
- 动态连接和模块加载

- 高效的中断和异常事件处理
- 优化浮点支持
- 动态内存管理
- 时钟和记时工具

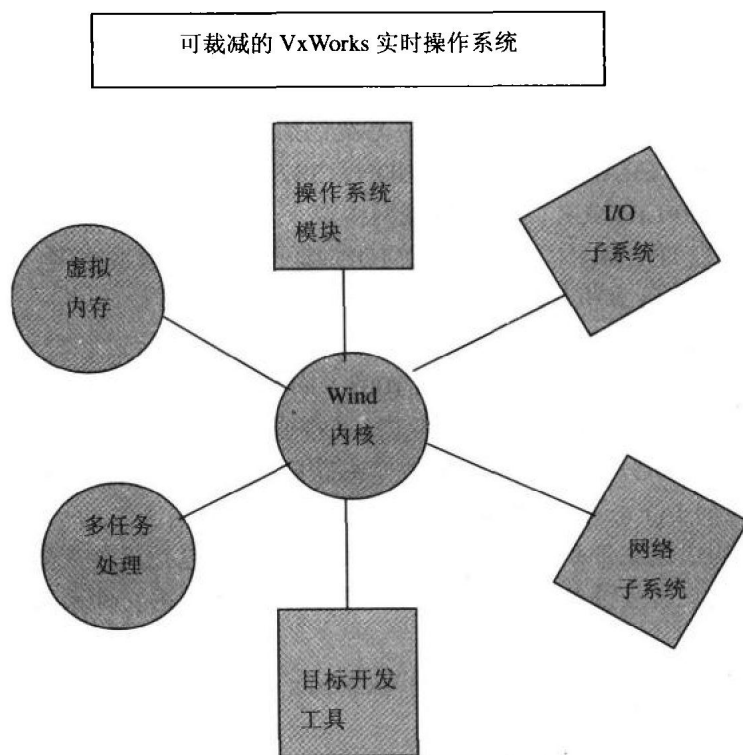


图 1-1 VxWorks 基本体系结构

2. 良好的可移植性

嵌入式应用通常需要将操作系统和应用程序方便地移植到目标硬件中，如果能够将依赖于硬件的低级代码和高级的应用程序和操作系统区分开来，移植工作将会变得非常简单。VxWorks 将依赖于硬件的低级代码设计成板级支持包（BSP）。任何一个要运行 VxWorks 的硬件主板都需要相应的板级支持包，有了板级支持包的支持，移植高级代码时，只要改变相应的依赖于硬件的板级支持包即可，无须修改操作系统和应用程序。

3. 良好的可裁剪性

VxWorks 的可裁剪性，可以使开发者根据自己的应用程序需要，而不是根据操作系统的需要来分配系统资源。从需要几 KB 字节的深层嵌入式设计到需要更多的操作系统功能的复杂的高端系统，开发者可以在 300 多个独立的模块中选择。而且，这些模块本身也是可裁剪的，所以 VxWorks 可以为用户提供粒度极小的运行环境配置。

所有这些配置选项可以通过 Tornado 的项目工具图形接口轻易地选择，或者使用 Tornado 的自动裁剪特性，自动地分析应用程序代码并选择相应的选项。

4. 网络组件

VxWorks 是支持工业标准 TCP/IP 网络协议族的实时操作系统。包括：

- IP、IGMP、CIDR、TCP、UDP、ARP、RARP、RIPV1/V2
- Standard Berkeley Sockets
- Z-bufs、NFS、RPC
- PPP、BOOTP、DNS、DHCP、TFTP、FTP
- RLOGIN、TELNET、RSH

WindRiver 还支持可选的 WindNet 产品, 包括 SNMP、OSPF、STREAMS 等。WindRiver 通过提供工业级最广泛的网络开发环境来加强这些核心技术, 这主要是通过 WindLink for Tornado 伙伴计划来实现的。高级的网络解决方案包括:

- ATM、SMDS、FRAME RELAY、ISDN、SS7、X.25、V5 等广域网协议
- IPX/SPX、APPLETALK、SNA 局域网协议
- 分布式网络管理 RMON、CMIP/GDMO、基于 Web 的解决方案
- CORBA 分布式计算机环境

5. 针对 POSIX1003.1b 标准的兼容性

VxWorks 支持 POSIX 1003.1b 的规定和 1003.1 中有关基本系统调用的规定, 其中包括进程初始化、文件和目录、I/O 初始化、语言服务、目录管理。而且 VxWorks 还支持 POSIX1003.1b 的实时扩展, 包括异步 I/O、记数型信号量、消息队列、信号、内存管理和调度控制等。

6. 灵活和快速的 I/O 文件特性

VxWorks 提供的快速文件系统适合于实时系统应用。它包括几种支持使用块设备 (如磁盘) 的本地文件系统。这些设备都使用一个标准的接口从而使得文件系统能够被灵活地移植到设备驱动程序上。

VxWorks 也支持 SCSI, 还支持在单独的 VxWorks 系统上同时并存几个不同的文件系统。这些文件系统包括:

- POSIX 异步 I/O 和目录管理
- SCSI 支持
- 兼容 MS-DOS 文件系统
- RAW Disk 文件系统
- TrueFFS 闪存文件系统
- ISO9660 CD-ROM 文件系统
- PCMCIA 支持

1.1.3 VxWorks 的可用主机 / 目标机

1. 支持的主机

- Sun-4: Sun OS 4.1.x、Solaris 2.4/2.5、Solaris 2.5.1/2.6、Solaris 2.7
- HP 9000/700: HP-UX 9.0.7、HP-UX 10.10、HP-UX 10.20
- PC: Windows 95、Windows 98、WindowsNT 3.51、WindowsNT 4.0、Windows 2000, Windows XP

2. 支持的目标机

- Motorola 68K: 68000、68010、68020、68030、68040、68060
- Motorola ColdFire: MCF51xx、MCF52xx、MCF53xx

- Motorola/IBM PowerPC: PPC4xx、PPC6xx、PPC7xx、MPC5xx、MPC8xx、MPC82xx
- Motorola M-Core
- Intel X86: i386、i486、Pentium、Pentium-Pro
- Intel ARM、StrongARM: SA-110、SA-1100、SA-1110
- NEC: V86x、V83x
- i960: KA/KB/CA/JX/PR、RP/RD
- NEC/LSI MIPS: R3K、R4K、Vr41xx、R4700、CW400x、CW4011、R5K
- HITACHI SH: SH-1、SH-2、SH-3、SH-4、SH-DSP、SH3-DSP
- SUN SPARC: UltraSPARC、SPARC
- ST-20
- TriCore

1.2 Tornado 开发环境

Tornado 是 Windriver 公司开发的嵌入式软件开发环境。Tornado 开发环境的最新版本是 2.2，它在延续了 Tornado 2.0 开发环境的基础上，增加了更多易于使用、性能优异的工具，因此在商业上取得了较大成功，获得了用户的好评。

Tornado 开发环境是嵌入式实时领域里最新的开发调试环境，是编写嵌入式实时应用程序的完整的软件开发平台。它给嵌入式系统开发人员提供了一个不受目标机资源限制的超级开发和调试环境。它包含 3 个高度集成的部分：

- 运行在宿主机和目标机上的强有力的交叉开发工具和实用程序
- 运行在目标机上的高性能、可裁剪的实时操作系统 VxWorks
- 连接宿主机和目标机的多种通讯方式，如以太网、串口线、ICE 或 ROM 仿真器

Tornado 嵌入式集成开发环境的结构图如图 1-2 所示。

嵌入式产品的复杂度越来越高，而同时生产周期也越来越紧张，这种情况给嵌入式开发人员带来了巨大的压力。用户们则希望以更低的价格获得更多的性能。现在市场上最热门的产品应该是快速而且更加智能化、可以运行更多代码的产品。所以开发人员必须能够快速启动并且快速开发。

Tornado 是 WindRiver 公司推出的嵌入式开发环境。它保持了原来版本的特色，同时又继续不断创新。Tornado 是一个开放的，而且独立于硬件的软件解决方案，它可以运行在 90% 的 32 位嵌入式处理机体系结构上，可以加速设计和开发的进程。Tornado 的开放特性使得它非常易于与第三方产品集成，而且在 WindLink 合作计划的推动下，Tornado 的工具愈加丰富和成熟，并且可以专注于互连网络和 Java 上。

Tornado 是用于实时嵌入式应用的完整的软件开发平台，可以用于 UNIX 和 Window 95/98/NT/XP，它包括 3 个高度集成的部件：

- Tornado 工具，一整套完整的交互式开发工具和实体
- VxWorks 嵌入式实时操作系统
- 主机和目标机之间全方位的通信选择

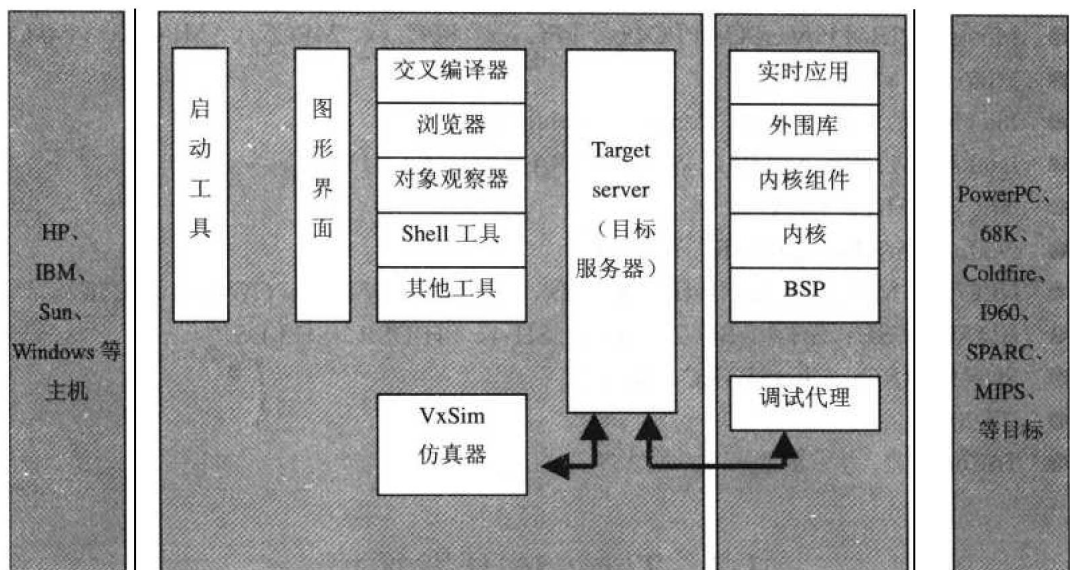


图 1-2 Tornado 嵌入式集成开发环境的结构图

Tornado 开放的体系结构以及符合企业标准的特性，使得开发人员可以轻松地开发不同厂家的系统，并且以最小的代价移植到不同的处理器上。

1.2.1 Tornado 核心工具

Tornado 核心工具包 (CORE TOOLS) 主要包括以下几项。

1. 集成仿真器 VxSim

许多嵌入式的开发人员发现商用的硬件并不能够满足应用的要求，所以需要开发自己的、定制的硬件。在硬件开发和调试的过程中，如果能够同时开始软件的开发和调试，那么就可以有效地缩短产品的开发周期。集成仿真器 (VxSim) 就是这样的产品。VxSim 可以在主机的环境中模拟 VxWorks 环境，用户可以在这个模拟的环境下开发软件程序，使用 Tornado2 IDE，从而实现软硬件的同步开发。

VxSim 是完整的 Tornado2 和 VxWorks 应用的原型模拟工具。它可以允许硬件开发完成之前先行开发软件，在开发早期实现软件的测试。VxSim 还具有一定的模拟通用 I/O 设备的功能，如串口，而且可以模拟文件系统和网络。

2. 软件逻辑分析仪 WindView

嵌入式的开发已经变得越来越复杂，实时任务也越来越多，如果没有复杂的系统级可视化工具支持，最终的系统几乎是不可读的。由于系统内部的运行不可知，许多开发者可能会非常困惑“为什么要用 200ms 才能接受一个键盘中断？”。

WindView 正是解决这类困惑的工具，它是一个软件逻辑分析仪，可以深入地了解任务的工作状态，同时跟踪系统的有用信息，如中断、上下文切换和任务阻塞等。所以，WindView 可以向开发者提供目标机硬件上实际应用程序的运行情况并且可以过滤掉无关的信息。

WindView 在 1994 年一经推出，立即获得了计算机界的一致好评。这是世界上第一个可以让开发人员看到正在目标机上运行的任务、中断、对象的运行情况的工具。开发人员可以在 WindView 的屏幕上看到系统中各种对象，如信号灯、消息队列、信号、任务、计时器和

用户事件的发生序列及时间。WindView 有以下 3 个重要的特性：

- 快速集中：可以帮助用户快速地分离应用中的关键区域
- 深入分析：分析事件内部的特性
- 随处可用：无须考虑目标机的类型就可以使用 WindView

3. 强有力的命令行执行方式 WindShell

Tornado 的命令执行工具 WindSh 是 Tornado 所独有的功能强大的命令解释器，可以直接解释执行 C 语句表达式、调用目标机上 C 的函数、查看符号表中的变量，还可以直接执行 TCL 语言。

4. 系统对象检查工具 Browser

Tornado 的目标机系统浏览器 Browser 是 Tornado 的一个图形化组件，它可以提供目标机中系统对象，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。

5. 集成调试器 CrossWind

CrossWind 是一个源代码集成调试器，支持任务级和系统级调试，支持混合代码和汇编代码显示，支持多目标机同时调试。

CrossWind 沿用了 GNU Debugger 4.17 版本，并且进行了 VxWorks 定制。在 CrossWind 中，开发者可以成组地观察表达式的观察窗口，可以在调试器的图形用户界面中迅速改变变量、寄存器和局部变量的值，可以为不同组的元素设定根值数，并且可以用分类的方法提供信息，使得调试工作效率更高。在提供 GUI 调试方法的同时，也提供了命令行方式，开发者可以根据自己的使用背景予以选择。

CrossWind 中有两种调试方式：任务级和系统级。在任务级调试中，程序员可以在单个任务中设置断点，这种断点只对该任务有效，其他任务和系统仍然持续运行。系统级断点可以在整个系统的运行时间轴上设置断点，一个断点可以将整个系统，包括任务和操作系统全部停下来，可用于调试多任务或者是中断处理程序。

6. C/C++交叉编译工具

Tornado 可以为用户提供支持 C 语言和 C++语言的交叉编译器和类库。

编译器包括 GNU 的 C/C++编译器和 Diab C/C++编译器。类库包括 VxWorks 类库和 tool.h 类库。

7. 工程管理和配置工具 (Project facility/Configuration)

VxWorks 中的工程管理工具是非常方便而且有用的图形化管理工具。它可以对 VxWorks 操作系统及其组件进行配置。组织、配置和建立 VxWorks 应用程序时，根据应用程序进行依赖性分析后，可以自动裁剪和配置 VxWorks。Project 中的检查功能还可以对开发人员定义的配置选项进行检查，并以醒目的方式标明冲突的配置。

更为有用的是，Project 工具还提供了生成不同 VxWorks 映像的功能。开发人员可以在 Project 中指定生成 VxWorks 的映像并且加入应用程序模块，从而为产品阶段的开发提供非常便利的手段。

8. 源代码分析仪 (WindNavigator)

如果想透彻地了解大型 C 或 C++程序的结构是一件非常困难的工作，如果该程序来自第三方，难度就更大了。一般而言，这类程序会包含上千个文件、函数和类。但是，开发人员

又不得不对这样的程序进行升级、维护和调试的工作。所以在 Tornado 中包含一个源代码分析仪 (WindNavigator)，它可以浏览源代码，提取出有用的信息后生成一个高效率的数据库，并且用图形来显示函数调用关系，快速进行代码定位。在几分钟之内，可以扫描数千行的程序。

9. 动态上载/卸载器 (Incremental Loader)

Tornado 的动态上载工具 (Incremental Loader) 可以动态地上载或卸载新增模块并且与目标机上的内核实现动态链接，所以开发人员可以不必重新下载内核以及未改动的模块。

1.2.2 WindPower 工具

Tornado 中的 WINDPOWER 工具包括：

- ScopePak
- PerformancePak
- CodeTest/Coverage
- CodeTest/Memory
- VisualSlickEdit
- Wind Foundation Classes
- Look! For Tornado

1. 软件“示波器” (ScopePak)

在开发实时多任务系统程序时，开发人员非常希望可以跟踪任务的运行以便更好地掌握系统的运行，但是传统的调试工具不能实时地显示任务的运行轨迹，所以开发者只好不停地在程序中增加诸如 printf 一类的语句来跟踪任务的运行。这种方法给任务的运行增加了巨大的开销而且跟踪的效果并不理想。Wind Power 工具——ScopePak 就可以在几乎不增加系统开销的情况下，提供系统实时的运行轨迹。ScopePak 中包含两个工具：StethoScope——图形化显示系统内变量的监视工具和 TraceScope——跟踪程序运行序列的工具。

- StethoScope StethoScope 是一个实时数据收集、图形显示、归档和调试工具，主要功能是以最小的干扰跟踪并发现程序中的性能问题、失灵或错误等。它可以利用多窗口环境图形化显示变量，就好像用数字示波器显示电路信号一样；可以在最小干扰的情况下捕捉问题，调整参数，观察系统运行轨迹；可以允许开发者动态地检查和分析正在目标机上运行的实时应用程序，并且可以提供历史数据图形和直角坐标图形；可以允许开发者在程序运行的过程中改变系统的变量；可以支持多种数据类型，以高采样速率收集事后分析的数据并在产生数据的同时显示数据，也可以根据特定的时间收集数据；可以发现噪音、失灵和系统中任何信号的问题；可以为图形和数据进行归档并打印。
- TraceScope TraceScope 对于多任务环境是一个非常有用的调试工具，它可以为开发者提供函数级的运行顺序、函数在什么时候被调用、顺序和调用参数如何等。如果与 WindView 结合使用，任何一个函数的入口和出口都可以成为 WindView 的事件予以显示。TraceScope 对系统的运行几乎没有影响。

2. 性能检测包 PerformancePak

实时系统的开发者总是希望获得最高的系统性能，但是又很难发现系统运行的薄弱环节。

开发者经常会遇到系统被挂起的情况，可是又不能确定是程序代码的原因还是操作系统或第三方工具的原因。在实际应用中，实时系统需要长时间稳定地工作，此时对内存使用的要求就非常高，然而内存有时是最难以控制的环节，如大量的内存被浪费，内存使用过程中发现的问题有时很隐蔽，经常是系统运行很长时间后，才能发现重大而且致命的内存问题。性能检测包(PerfomencePak)就是解决这个问题的工具，它包括一个 CPU 的分析工具 ProfileScope 和内存的分析工具 MemScope。

- ProfileScope ProfileScope 是一个分层的 CPU 运行轮廓分析仪。它可以分析 CPU 在开发者的代码、第三方库函数或操作系统中花费了多少时间。无须任何特殊的编译或硬件就可以提供全方位的精确的树型轮廓图。利用 ProfileScope，开发者可以逐个函数地分析 CPU 的运行轨迹，而且 ProfileScope 还可以指出系统效率低的环节，帮助开发者将系统的性能提高到最佳，并且发现系统的瓶颈所在。
- MemScope MemScope 可以用图形化的方式分析内存的使用情况而且无须任何特殊的编译。MemScope 可以为开发者提供实时的详细的内存使用图，包括每一个内存块的历史调用过程。它还可以帮助开发者发现内存漏洞，全面理解内存的使用情况，从而使系统的运行更加高效和可靠。

3. 代码测试器 CodeTest/Coverage

在实时嵌入式系统中，很难预料程序的运行序列，开发者难以确定系统内部的运行轨迹，而软件更是其中最难以预料的部件。在以往的嵌入式开发中，开发者通常会利用自行开发的工具或手工检查代码是否运行，这样做的结果当然不能令人满意。

WindPower 工具 CodeTest/Coverage 就是为嵌入式开发提供的嵌入式软件验证工具，有了 CodeTest/Coverage，开发者和测试者可以在开发早期验证软件的模块和集成特性，无须任何硬件的支持。CodeTest/Coverage 可以在代码运行的过程中检查代码的运行序列，准确地显示代码在特定的测试条件下的运行轨迹，为开发者提供有用的信息。

CodeTest/Coverage 可以在程序的运行过程中，交互地显示程序、函数和源代码的语句覆盖情况。CodeTest/Coverage 的独特之处还在于它可以在测试进行过程中以 XY 时间轴的方式显示覆盖趋势，如果发现覆盖峰值，可以及时终止测试，因此可以有效地缩短测试的时间。

CodeTest/Coverage 所生成的分析结果可以让开发者在实时系统的运行过程中跟踪覆盖率并观察测试的效率。开发者可以用多种方法观察覆盖信息：总结、函数、源代码及趋势。开发者可以获得真实的覆盖测量结果而不是统计采样，因此开发者获得的永远是最真实的测量结果。CodeTest/Coverage 可以跟踪数以百万计的分支，所以非常适合于当前的嵌入式应用。CodeTest/Coverage 所产生的信息有助于开发者发现无用的代码，减少死码，因此可以缩小程序的大小和复杂度，提高程序的性能。

4. CodeTest/Memory

在实时嵌入式开发过程中，跟踪并标识内存的动态分配和错误是非常重要而且非常困难的工作。一个小小的内存故障可能会引起系统严重的故障。所以，在系统失效之前及时地发现内存的漏洞是必要而且必须的。CodeTest/Memory 可以让开发者和测试者在开发早期验证软件的模块和集成特性，无须任何硬件的支持。CodeTest/Memory 可以解决嵌入式程序中最难解决的动态内存跟踪问题。它不仅可以在程序的运行过程中报告每一个申请语句所分配到的内存字节，更可以发现 20 多种分配错误，如“释放空指针”等。