

嵌入式实时操作系统 BSP 的探讨

刘雪艳,张 强,张贵仓

(西北师范大学 数学与信息科学学院,甘肃兰州 730070)

[摘 要] 基于嵌入式实时操作系统的要求,分析了 VxWorks 的功能,强调 BSP(Board Support Package) 针对于不同的基本接口界面,提供不同的 BSP;同时说明 BSP 的功能是提供给操作系统一个无关的设备接口,可以让程序开发者在不熟悉具体硬件的情况下正确地配置硬件寄存器,并给出了它的主要设计方法和实现过程。

[关键词] 嵌入式;VxWorks;板级支持包(BSP)

[分类号] TP316.2 **[文献标识码]** A **[文章编号]**1008 - 8113(2003)06 - 0068 - 03

随着微电子技术的迅猛发展,数字化技术已经渗透到社会、经济、通信、医学、军事等各行各业,使得嵌入式实时操作系统成为研究的一个热点。

嵌入式实时操作系统是软硬件的紧密结合体,它的特点可分为两大块:嵌入式和实时性。嵌入式系统是以应用为中心,计算机作为一种智能部件内装于专用设备,软硬件要求具有高度可裁减性,以便适应应用系统对功能可靠性、体积、功耗等方面的要求的专用系统。实时性要求实时软件对外部事件做出反应的时间必须要快,在某些情况下还需要确定的、可重复实现的,不管当时系统内部状态如何,都是可预测的。

结合嵌入式实时操作系统的两大特点,它还具有两个重要功能:处理异步并发事件功能,快速启动、出错处理和自动复位功能。在实际环境中,嵌入式实时操作系统处理外部事件往往不是单一的,这些事件通常同时出现,且发生的时间也是随机的、异步的,这就不允许控制软件临时从盘上装入,因而嵌入式实时软件需先固化到只读存储器,开机自行启动,并在运行出错死机时能自动恢复先前的运行状态,所以嵌入式实时软件应采用特殊的出错处理措施。

1 VxWorks 操作系统

VxWorks 操作系统是美国 WindRiver 公司推出的一种嵌入式强实时操作系统(RTOS),是 Tornado II 嵌入式开发环境的关键组成部分。它是目前应用最广泛的操作系统,具有高效的实时任务调度、中断处理,实时的任务间通信,可靠的系统稳定性,高度可裁剪性,是现在所有独立于处理器的实时系统中最具特色的操作系统。

VxWorks 的核心特点有:

- (1) 高性能的可高度裁剪的实时操作系统内核 Wind:需要存储空间小,具有高度缩减性;
- (2) 任务间通信机制:支持信号量(semaphores)、消息队列(messagequeue)、管道(pipe)、信号(singal)和套接字(socket);

[收稿日期] 2003 - 09 - 22

[作者简介] 刘雪艳(1978~),甘肃兰州人,西北师范大学数学与信息科学学院助教,硕士研究生,主要研究方向为计算机图形图像处理;张强(1975~),甘肃兰州人,西北师范大学数学与信息科学学院助教,硕士研究生,主要研究方向为数据库;张贵仓(1965~),甘肃天水人,西北师范大学数学与信息科学学院副教授,硕士生导师,主要研究方向为计算机图形图像处理。

- (3) 高效的多任务调度:采用基于优先级抢占调度(Priority - base preemptive scheduling) 和轮转(round - robin) 调度算法;
- (4) 强大的网络支持:提供对其他网络和 TCP/ IP 网络系统的“透明”访问,包括与 BSD 套接字兼容的编程接口,远程过程调用,远程文件访问及 BOOTP 和 ARP 代理;
- (5) I/O 系统:提供快速灵活的与 ANSIC 兼容的 I/O 系统,包括 UNIX 标准的缓冲 I/O 和 POSIX 标准的异步 I/O;
- (6) 快速的文件系统:支持 MS - DOS ,RT11 ,rawFs 文件系统;
- (7) 板级支持包:对各种板子的硬件功能提供统一的软件接口。

2 板级支持包 BSP

2.1 板级支持包 BSP(Board Support Package)

板级支持包 BSP 是介于硬件和上层软件之间的底层软件开发包,对各种板子的硬件功能提供统一的软件接口,负责在上电时的硬件初始化,屏蔽硬件,提供操作系统的驱动及硬件驱动,将 VxWorks 的硬件相关和硬件无关的软件集成为一体。

简单地说,BSP 就是一段启动代码,通常是初始化寄存器,配置地址空间以及系统启动方式,对不同的微处理器板及不同的 RTOS 应配置各自相应的 BSP,BSP 可针对不同的硬件作不同的调整,也可根据实际的要求修改 BSP,在 BSP 中只需要对硬件相关的寄存器及与终端相关的寄存器进行正确配置即可。BSP 为程序开发人员提供了对硬件进行描述性操作的开发支撑库,同时还包含了对目标板的初始化部分以及一些简单的驱动程序单元,这样的 BSP 可以不依赖于任何操作系统和驱动程序,但可作为操作系统和驱动程序的开发支撑库,方便地移植或开发操作系统和驱动程序(BSP 在 VxWorks 体系结构中的地位如图 1 所示)。

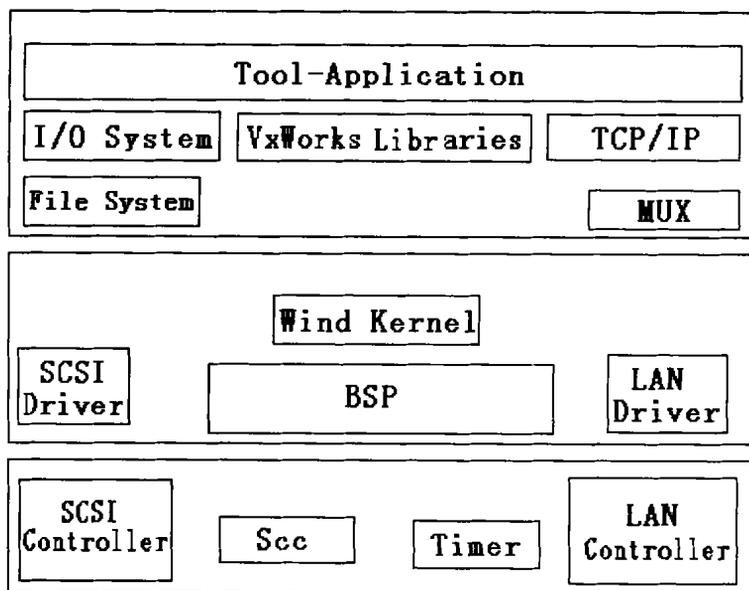


图 1 VxWorks 体系结构

2.2 BSP 运行流程

VxWorks BSP 包含以下几部分:源文件(source file);头文件(include file)和编译生成文件(make file);继承文件(derived file)和二进制驱动模块(binary)。

VxWorks 的 BSP 可以驻留在 ROM 中,也可以在线加载,通常启动代码驻留在 ROM,它的简单运行流程

如图 2 所示,其中 romInit() 为上电复位的入口函数,它禁止中断,将启动类型压栈,做硬件相关初始化且转到 romStart(),romStart() 加载 VxWorks 映像到 RAM 中,即完成必要的代码重定位,驻留映像重定位数据段。usrInit() 保存引导类型信息,然后激活 VxWorks 内核,在多任务环境建立后(Post - kernel),启动用户应用程序。

romInit()

romStart()

usrInit()

用户程序

图 2 BSP 运行流程

2.3 VxWorks BSP 设计过程

VxWorks BSP 的设计与开发过程一般可以分为以下几个步骤:

(1) 建立开发环境(Tornado),选择合适的开发工具。

(2) 选择合适的 BSP 模板:主要是以目标板 CPU 的 BSP 文件为模板,在 \ tornado \ target \ config 目录下创建用户的 BSP 目录 bspname,把 \ tornado \ target \ config \ all 下的文件和 BSP 模板文件拷贝到该目录下,根据具体情况选择合适的 VxWorks 镜像类型,修改 Makefile 文件,为特定目标系统生成 ROM 引导与 VxWorks 系统映像。

(3) 修改或添加 WIND 内核激活前的初始化代码:usrInit() 程序以调用两个内核初始化函数宣告结束,这两个函数为:

usrKernelInit() 程序(在文件 usrInit.c 中)调用适当的初始化子程序;

kernelInit() 程序初始化多任务环境,并且永不返回。

(4) 内核激活以后,连接系统中断和系统时钟,修改或添加所需要的设备驱动程序等。在任务 usrInit() 中首先初始化 VxWorks 时钟,通过调用 sysClkConnect() 将系统时钟中断向量与中断处理程序 usrClock() 建立连接;然后根据情况初始化 I/O 系统和设备驱动。

(5) 测试与验证。

3 结束语

以上基于嵌入式实时操作系统的要求,介绍了 VxWorks 的功能,强调 BSP 针对于不同的基本接口界面,提供不同的 BSP;同时说明 BSP 的功能是提供给操作系统一个无关的设备接口,可以让程序开发者在不熟悉具体硬件的情况下正确地配置硬件寄存器,并简单给出了它的设计和实现过程。

[参考文献]

[1] Tornado BSP Developer Kitfor VxWorks User 's Guide, Tornado2.0 Edition1,1999.

[2] VxWorks Programmer 's Guider5.3.1 Mar97 WindRiver Systes Inc.

[3] 孔祥营,柏桂枝编著. 嵌入式实时操作系统 VxWorks 及其开发环境 Tornado[M]. 中国电力出版社,2003.

[4] 汤子瀛,汤小丹,哲凤屏著. 计算机操作系统[M]. 西安:西安电子科技大学出版社,1996.

【责任编辑:邓崇亮】