

# VxWorks 在某实时显控设备中的应用

李义民

中国电子科技集团公司第 27 研究所 郑州 450005

**摘 要** 以实时可裁剪微内核 VxWorks 为操作系统,结合某实时显控设备这一使用环境,对该设备应用软件的开发进行了较为详尽的介绍。重点讲述了实时性、任务划分、优先级设置等应用程序中较关键而又复杂的问题。

**关键词** 实时性 多任务 嵌入式 VxWorks 电子对抗 显控台

## 1 引言

多任务、抢占调度、快速上下文切换、低中断延时和快速灵活的通信机制是对现代实时操作系统内核的标准要求。而 VxWorks 正是满足了这种要求的操作系统内核的杰出代表,它由一个微内核、文件系统和 I/O 管理、网络支持及支持 C++ 的各种模块组成。在诸如通信、航天、高清晰度数字电视、军工舰船等众多领域都有出色的运用, VxWorks 具有的可伸缩性、可裁剪、高可靠性使其能适用于所有的流行 CPU 平台<sup>[1]</sup>。当然,基于 VxWorks 的嵌入式应用程序开发是离不开一个优秀的开发环境的, Tornado II 正是 Wind River 公司推出的一个高度可视化和自动化的集成开发环境,该开发环境提供目标机与宿主机之间的交叉通信连接,并且允许开发者增量地加载目标模块到目标系统,从而加速了软件设计中最为费时的编译与调试周期<sup>[2]</sup>。

实时显控设备在一个系统中最为主要的任务顾名思义就是综合显示与干预决策,它同时是各分系统(设备)的连接纽带,是整个系统的控制处理中心,它能完成系统的信息处理、综合显示、威胁告警、目标分配、资源管理、综合控制、数据相关、情报录取、目标源定位、训练模拟等战术功能。正是由于这些事件的异步性,并且要求并行执行,所以能够运

行许多并发进程和任务便成为对操作系统最为基本的要求。当然,在操作系统不能提供多任务操作时例如 DOS、WINDWOS、WINDOWS NT 等环境,就需要程序员自己手动编写相应的多线程或多进程程序,效率必定会大打折扣。VxWorks 这种实时操作系统内核利用任务级别的不同来分配 CPU 机时从而获得多任务的并发性,基于抢占调度和灵活的任务间通信与同步、任务与中断间通信等确保了 VxWorks 可以很好的满足上述特定的实时环境的实时性要求。

## 2 硬件环境和支持软件的开发

通常来说,实时显控设备其硬件核心为一台或多台增强型加固计算机,具体来说,它由以下部分组成:显示单元、操控单元、电子机箱单元及辅助设备单元等。其中电子机箱单元由人机接口计算机和任务机组成,人机接口计算机单元包括中央处理模块、显示模块、操控模块、网络单元、辅助模块。

基于以上硬件的底层支持软件由各种底层驱动程序、BSP(板级支持包)和软件中间件组成。通常,串行通信、并行通信、存储设备、图形显示、输入部件模块的驱动程序往往由硬件生产商提供(只有特定的硬件生产商提供针对 VxWorks 操作系统的驱动即 BSP);BSP 即板级支持包,主要功能是在上电时对硬件进行初始化,从而实现操作系统

和硬件的连接。大多数情况下,需要使用者自己编写基于所设计的硬件的 BSP;中间层软件提供用户一致的开发界面,它屏蔽了底层硬件和操作系统的细节,形成抽象的模型。面向用户的中间层软件由一组中间件组成,各中间件由相应的 Application Program Interface(API 函数)构成<sup>[3]</sup>。根据人机接口计算机所承担的任务和配置,采用软件层次结构如图 1 所示。

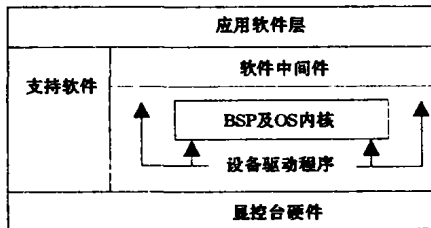


图 1 基于 VxWorks 的显控台软件结构

### 3 应用软件开发

根据某工程对显控设备的具体要求,本应用软件可以划分为以下几个功能模块:任务管理模块、初始化模块、人机交互模块、命令生成模块、显示模块、数据综合处理模块、通信模块、帮助模块,而每一个功能模块又由一个或多个任务组成。框图如图 2。

我们知道,一个系统如果是多任务的,那么它未必是实时的;但一个系统如果是实时的,那么它必定是多任务的。因此,实时系统中任务的数量是一个系统软件顶层设计中需慎重考虑的问题之一,任务数量太少,势必造成系统实时性差,反之,任务过多,CPU 的开销也必然增加。上图中系统任务的划分正是基于 H. GEMMA 原则(I/O 原则、异步事件原则、周期事件原则、功能耦合、偶然耦合等)和考虑显控台的特殊性而设计的。由框图 2 我们可以清楚地看到,电子对抗显控台应用软件的详细任务划分情况,此处的任务即是特指 VxWorks 中的 Task(我们可以把其当作一个进程来理解,针对 VxWorks 中的每

一个 Task 都有就绪态(Ready)、阻塞态(Pend)、延迟态(Delay)、挂起态(Suspend)四种状态,或者这四种状态的变换组合,当然,在这个任务执行完毕后,它便处于 DEAD 状态或者说该任务不存在了)。如此众多的任务,必须依靠一种有效的调度机制进行协调,那就是不同优先级的任务进行优先级抢占和同等优先级的任务进行时间片轮转调度,从而达到高效利用 CPU 资源,实现系统的实时性要求。

继任务划分和任务创建之后,电子对抗显控台应用软件的顶层详细设计便可分为以下几个步骤进行:

首先,确定各任务的优先级和入口函数。VxWorks 操作系统提供 0 到 255 共 256 个软件优先级,应用程序中还可以使用中断,基于硬中断的中断服务程序(ISR)高于一切优先级的任务。例如在本系统中,人机接口计算机和战术决策的任务机针对共享的扩展存储器便采用了中断方式。确定各任务的优先级主要依据各自任务的重要程度及系统分解给其的响应时间,我们知道,每个实时系统都有其特定的 Time Deadline,超过了这个时限,一个系统便不能称之为实时了。将这个时限再进行综合考虑(即某个任务的重要程度、运行时间、触发频率等)后分解到各任务,然后依据该分解指标再暂定该任务级别。一个任务可能由多个函数组成,一般来说,入口函数在一个任务中起着初始化和总控的作用,合理安排该函数将使程序结构清晰和易于阅读。图 3 为优先级示意图,图中任务优先级由高到低依次排列。

其次,确定任务间通信机制。VxWorks 操作系统提供了共享地址空间、消息队列、管道、套接字、远程过程调用等任务间的通信机制<sup>[2]</sup>,因后几种通信方式其本质仍是基于内核同步方式的,类似于消息队列,而且由于包含了更高层的抽象而造成操作系统内核扩大,开销增大,因此较少使用。所以,应用软件详细设计在这一步骤的目的便是确定各任

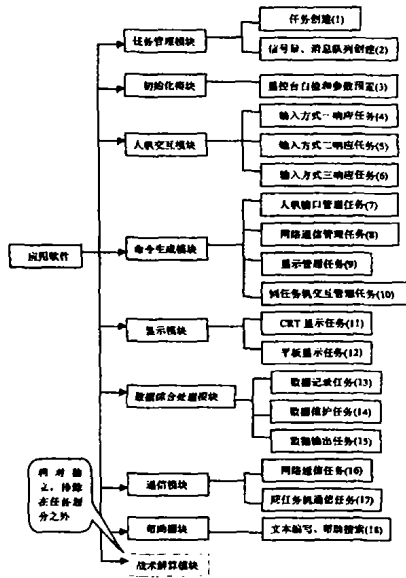


图2 应用软件详细任务划分

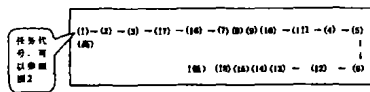


图3 任务优先级示意图

务间是使用共享地址空间还是使用消息队列。第一种通信方式适于较大数量的数据交换,不同的任务可以使用共享的数据结构指针自由的通信;相应的,通过互斥访问便成为不同任务间避免资源竞争的有效办法,实现互斥的方法可以有禁止中断、禁止任务抢占和通过信号量进行资源锁定。第二种通信方式使用更为方便,是一种适于任务间交换变长数据的较底层的通信机制。依据显控台应用软件各任务之间的数据交换量和数据特点,各任务间通信方式如图4。

最后,编写各任务代码并进行调试。通过代码编写和调试,将落实顶层设计时的各种思想,并及时发现原设计的不足和错误之处。特别的,上述第一步骤当中,任务优先级是基于静态设计(taskSpawn)的,没有考虑如何利用优先级逆转和优先级继承等问题(高低优先级的任务在同时竞争资源时,高优

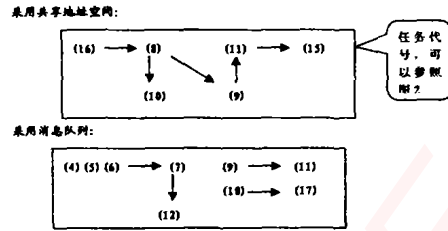


图4 任务间通信方式示意图

先级任务被强制等待一段时间,以便继承了另一个更高优先级任务的低优先级任务得到资源),因此软件各任务间有可能发生阻塞和死锁,所以软件的动态调试将显得特别重要。Tornado II 提供的交叉编译环境(宿主机和目标机之间通过网络相连,生成后的可执行程序直接下载到目标机上运行)大大方便了调试过程。

### 4 设计验证

经过顶层设计—详细设计—代码编写—动态调试等过程,VxWorks 操作系统以其独特的优势而在实时显控设备这一信息吞吐量、计算解算频繁、图形显示复杂、实时性要求高的平台上获得了无论开发者还是设备使用者的广泛认可。应用软件设计达到了预期的效果。

### 5 结束语

实时操作系统 VxWorks 和其开发环境 Tornado II 进入国内时间还很短,却已在广泛的领域获得了应用,尤其是在例如实时显控台这样高可靠性、高指标要求的复杂电子设备上获得了认可,其品质可见一斑。“它山之石,可以攻玉”,恰当地利用当今世界的高科技产品可以使我们的工程更上一个台阶。

#### 参考文献

- [1] VxWorks Programmer's Guide. Wind River Systems, Inc.
- [2] Tornado User's Guide. Wind River Systems, Inc.
- [3] BSP Reference. Wind River Systems, Inc.