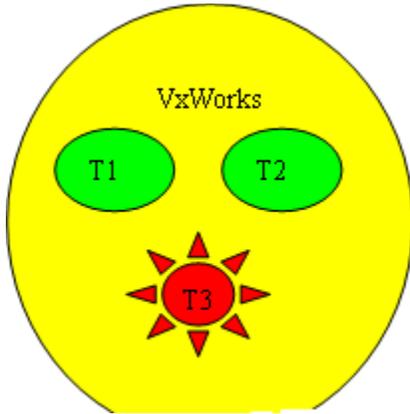


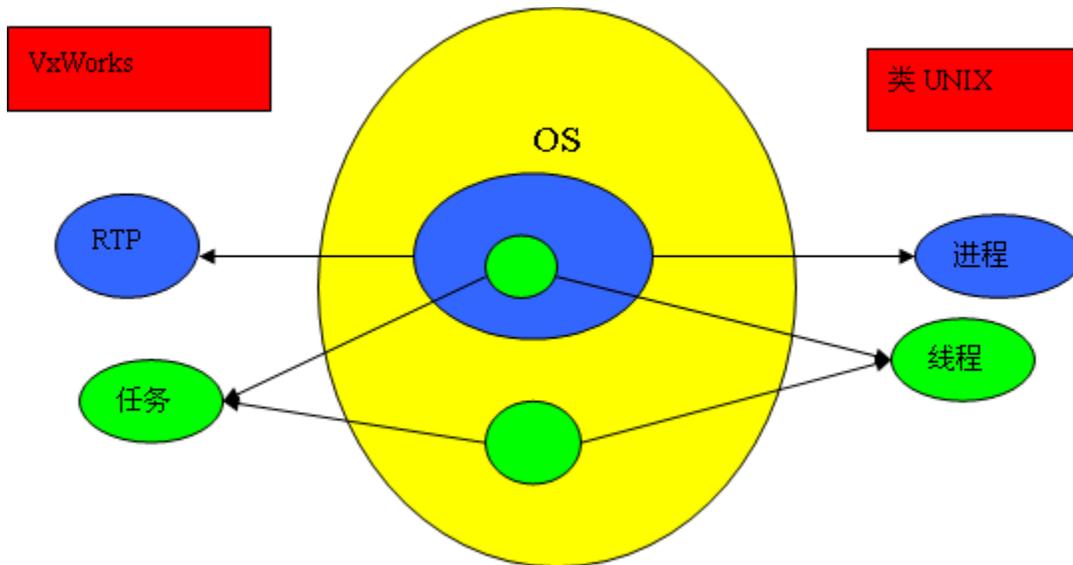
VxWorks 下的 RTP 是什么

VxWorks 的 RTP，全称是 Real-Time Process，可以翻译为实时进程。

在 6.0 之前，VxWorks 使用的是 single 的内存空间，操作系统与应用程序是不分离的。这种模型在开发应用程序时有更好的灵活性和性能，但对开发者的要求也更高一些，因为错误的应用可以轻易导致其它应用或系统崩溃，程序员需要确保内核设施和应用程序在同一内存空间中和平共处。例如任务 T3 的栈溢出了，就有可能破坏其它任务的地址空间，如果被破坏的是系统任务 T1，那很可能结果就是死机。而此时的 T3，有可能已经退出了，T1 就成了 T3 的背锅侠了。



从 6.0 开始，VxWorks 引入了 RTP。这个 RTP 在许多地方都与其它操作系统的进程差不多，例如对 POSIX 的兼容性。所以了解 UNIX/Linux 进程模型的程序员，很快就可以熟悉 RTP 的创建、执行或者终止。



不过 RTP 是专门为 RTOS 设计的，为了满足实时性的需求，它与其它系统的进程还是有很多不同的。

RTP 用于在用户模式里执行应用程序。每个进程使用独立的地址空间，将自己的程序、数据、栈、堆，以及其它相关资源保护起来。系统中可以同时存在多个进程，每个进程内可以有多个任务(类似于其它系统的线程)。

有了 RTP，就可以在用户模式下执行应用程序和操作系统的其它功能，这些功能在内核和应用程序之间具有清晰的划分。这种体系结构通常被称为进程模型。

同时，6.x 与 5.5 保持了高度的兼容。基本上不需要任何修改，就可以将早期 VxWorks 的内核应用程序迁移到 6.x 内核空间(在大多数情况下，只是重新编译)。

www.vxbus.com