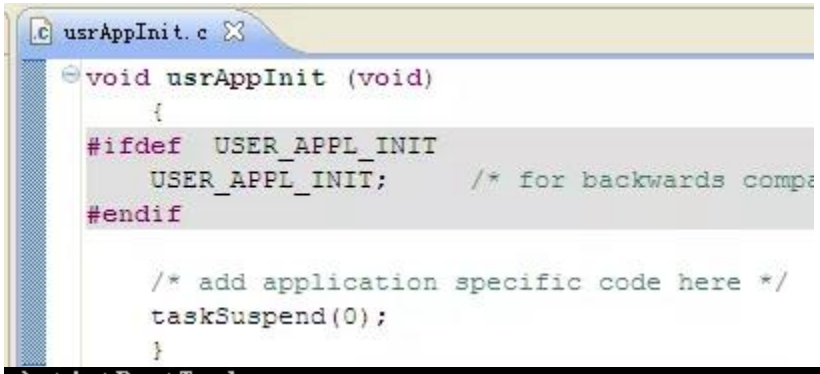
VxWorks 下 Task 中的系统任务

根据配置的不同，VxWorks 系统在启动时，会同步启动一些系统任务，其中有的任务在完成自己的工作后就会退出，而有的会一直运行下去。常见的系统任务如下：

任务名称：tRootTask
优先级：0
栈尺寸：ROOT_STACK_SIZE，具体数值与 CPU 架构有关，详见 configAll.h
触发条件：系统启动
组件：无
描述：这是内核启动的第一个任务，随后它会启动其它系统任务。任务入口是函数 usrRoot()，在这个函数里初始化了系统的大部分功能。通常情况下，在初始化完其它功能后，这个任务就自己退出了。为了查看它，我们在这个任务的最后位置添加一个 taskSuspend()，把它挂起来
注意：实际产品里不要挂起、删除，或调整这些系统任务的优先级。否则会导致系统行为不可预测




```
-> ti tRootTask

  NAME         ENTRY       TID    PRI   STATUS      PC
---------- ------------ -------- --- ---------- -------- ---
tRootTask   usrRoot        de8c90   0 SUSPEND      50ce4f

full task name : tRootTask
task entry     : usrRoot
process        : kernel
options        : 0x7
VX_SUPERVISOR_MODE  VX_UNBREAKABLE      VX_DEALLOC_STACK

 STACK      BASE     END       SP      SIZE    HIGH   MARGIN
--------- -------- -------- -------- ------- ------- -------
execution   de8c90   de6890   de8c10    921
exception   de6870   de4890              816
```

任务名称：tLogTask
优先级：0
栈尺寸：5000
触发条件：中断或任务里调用 logMsg()
组件：INCLUDE_LOGGING
描述：用于记录系统信息，不使用当前任务的 IO
最多排队消息数量：MAX_LOG_MSGS，默认值 50
格式化参数数量：6

```
-> i tLogTask

  NAME         ENTRY       TID    PRI   STATUS
---------- ------------ -------- --- ---------- --
tLogTask    logTask        e09590   0 PEND
value = 0 = 0x0
-> logMsg "%d %d %d %d %d %d %d\n", 1,2,3,4,5,6,7
0xeb8c50 (tShellRem15362348): 1 2 3 4 5 6 1543x880
value = 32 = 0x20 = ' '
```

任务名称：tExcTask
优先级：0
栈尺寸：EXC_TASK_STACKSIZE，默认值 8192
触发条件：中断里调用 excJobAdd()
组件：INCLUDE_EXC_TASK
描述：这个任务用于执行中断里不方便执行的工作，因此必须使用最高优先级。可以在这个任务上排队的工作的最多为 MAX_ISR_JOBS，取值必须为 2 的幂，默认值为 16。如果中断里需要排队的工作超过这个限制，就会有溢出警告"messages from interrupt levellost"

```
-> ti tExcTask

  NAME         ENTRY       TID    PRI   STATUS      PC       S
---------- ------------ -------- --- ---------- -------- ----
tExcTask    48a347         5c1880   0 PEND         50fcf2   5c

full task name : tExcTask
task entry     : 48a347
process        : kernel
options        : 0x3
VX_SUPERVISOR_MODE  VX_UNBREAKABLE

 STACK      BASE     END       SP      SIZE    HIGH   MARGIN
--------- -------- -------- -------- ------- ------- -------
execution   5c1880   5bf880   5c1770    8192    1332    6860
exception   5bf800   5bd880             8064
```

任务名称：tJobTask
优先级：启动时为 0，根据执行的工作而动态调整
栈尺寸：JOB_TASK_STACK_SIZE，默认 8000
触发条件：其它任务提交工作
组件：INCLUDE_JOB_TASK

描述：这个任务用于执行其它任务提交的工作。它使用优先级 0 等待工作，在执行工作时，动态调整为提交工作的任务的优先级。主要作用之一是处理任务的自我删除

```
-> ti tJobTask

  NAME         ENTRY       TID    PRI   STATUS      PC
---------- ------------ -------- --- ---------- --------
tJobTask    48ae51         e00ad0   0 PEND         50fcf2

full task name : tJobTask
task entry     : 48ae51
process        : kernel
options        : 0x9003
VX_SUPERVISOR_MODE  VX_UNBREAKABLE      VX_DEALLOC_TCB

 STACK      BASE     END       SP      SIZE    HIGH   MARGIN
---------- -------- -------- -------- ------- ------- -------
execution   e00ad0   dfeb90   e009e0    80[illegible]
exception   e02cf0   e00d70             80[illegible]
```

任务名称：tIsr*N*
优先级：0
栈尺寸：8192
触发条件：设备中断调用 isrDeferJobAdd()
组件：INCLUDE_ISR_DEFER
描述：这个(组)任务用于执行设备驱动通过 isrDeferJobAdd()提交的工作。名称中的字母 *N*表示这个任务所使用的 CPU 的序号，在单核环境里，那就只有 tIsr0 了。这组任务创建时，每个都绑定到相应序号的 CPU 上。多核模式的设备驱动将需要推迟的工作绑定到当前的 CPU 上，用于避免跨 CPU 调度

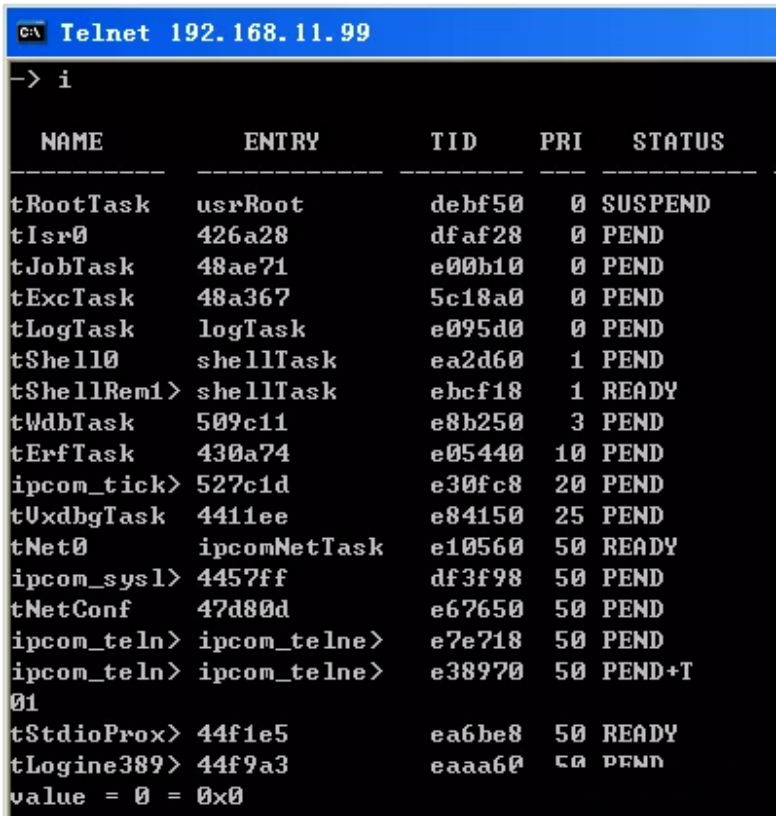
```
-> ti tIsr0

  NAME         ENTRY       TID    PRI   STATUS      PC       SP     ERRNO  DELAY
---------- ------------ -------- --- ---------- -------- -------- ------- -----
tIsr0       426a08         dfaee8   0 PEND         50fcf2   dfae50       0     0

full task name : tIsr0
task entry     : 426a08
process        : kernel
options        : 0x9003
VX_SUPERVISOR_MODE  VX_UNBREAKABLE      VX_DEALLOC_TCB      VX_DEALLOC_EXC_STACK

 STACK      BASE     END       SP      SIZE    HIGH   MARGIN
---------- -------- -------- -------- ------- ------- -------
execution   dfaee8   df8ee8   dfae50    8192     316    78
exception   dfd130   dfb1b0             8064     100    79[illegible]
```

任务名称：tNet0
优先级：NET_TASK_PRIORITY，默认值 50
栈尺寸：NET_TASK_STACKSIZE，默认值 10000
触发条件：数据包到达，传输完成，网络协议里的定时器到时，socket 应用的请求，等等
组件：INCLUDE_NET_DEAMON
描述：这是网络驱动和网络协议的守护线程

任务名称：tWdbTask
优先级：WDB_TASK_PRIORITY，默认值 3
栈尺寸：WDB_STACK_SIZE，默认值 0x2000
触发条件：无
组件：INCLUDE_WDB
描述：WDB 的 target agent 程序，用于处理 host tool 通过 target server 发送的请求

任务名称：tShell*N*
优先级：SHELL_TASK_PRIORITY，默认值 1
栈尺寸：SHELL_STACK_SIZE，默认值 0x10000
触发条件：系统启动
组件：INCLUDE_SHELL
描述：kernel shell 以任务形式存在的，可以同时启动多个，不同的 shell 使用不同的序号 *N* 为名称后缀，名称”tShell”是通过 SHELL_TASK_NAME_BASE 定义的。在 shell 里再调用的函数会使用这个 shell 的上下文。

任务名称：ipcom_telnetd
优先级：50
栈尺寸：6144
触发条件：新的 Telnet 连接
组件：INCLUDE_IPTELNETS
描述：这个守护线程允许远程用户通过 Telnet 登陆 VxWorks 的 kernel shell。它会为每个 Telnet 连接启动一组任务，包括 ipcom_telnetspawn，tStdioProxy*hexNum*，tLogin*hexNum*，tShellRem*decNum*



等介绍了任务调度之后，我们就会发现这些系统任务的优先级都是比较高的，我们自己应用程序的优先级尽量要低一些。

等介绍了 VxWorks 系统的启动流程后，我们就会知道嵌入式硬件上电后，先是执行汇编语言的初始化程序，然后跳转到 C 语言的程序，然后启动第一个任务 tRootTask，然后逐步加载其它系统任务。

这些高优先级的系统任务通常都是处于 Pend 状态，只有外界(应用程序或外设)需要它们时，它们才会提供相应的功能。它们整体对外的表现就是一个提供了很多功能的强大的实时操作系统。