

VxWorks 下 Task 的状态

在介绍“Task 是啥”时，我们提到过，在多任务系统里，任务可能有不同的状态，例如 Waiting 或 Ready。事实上，VxWorks 里的任务可能有更多的状态，主要有 Ready(就绪态)、Pend(阻塞态)、Delay(延时态)和 Suspend(挂起态)。还有一种特殊的状态 Stop，以及多种组合状态。

状态	描述
Pend	任务在等待某个系统资源(例如信号量、IO)
Delay	任务在等待一段时间，也算是一种资源吧
Ready	除了可能等待CPU，任务不需要等待任何资源
Suspend	任务被挂起，不能继续执行。挂起的原因有两种： 1手动挂起，例如调用taskSuspend()； 2任务出现异常后，被系统挂起，例如调用除零操作
Stop	任务被Debugger或ED&R机制或SIGSTOP信号停下
Delay+S	任务延时并被挂起
Pend+S	任务阻塞并被挂起
Pend+T	任务阻塞并延时
Stop+P	任务阻塞并被停下
Stop+T	任务延时并被停下
Stop+S	任务停下并被挂起
PEND+S+T	任务阻塞并延时并被挂起
STOP+P+S	任务阻塞并被挂起并被Debugger停下
STOP+P+T	任务阻塞并延时并被Debugger停下
STOP+T+S	任务被挂起并延时并被Debugger停下
ST+P+S+T	任务阻塞并延时并被挂起并被Debugger停下
以上任意状态+I	任务的优先级是继承的

在 VxWorks 的 Shell 里通过命令 i 可以看到任务的状态：

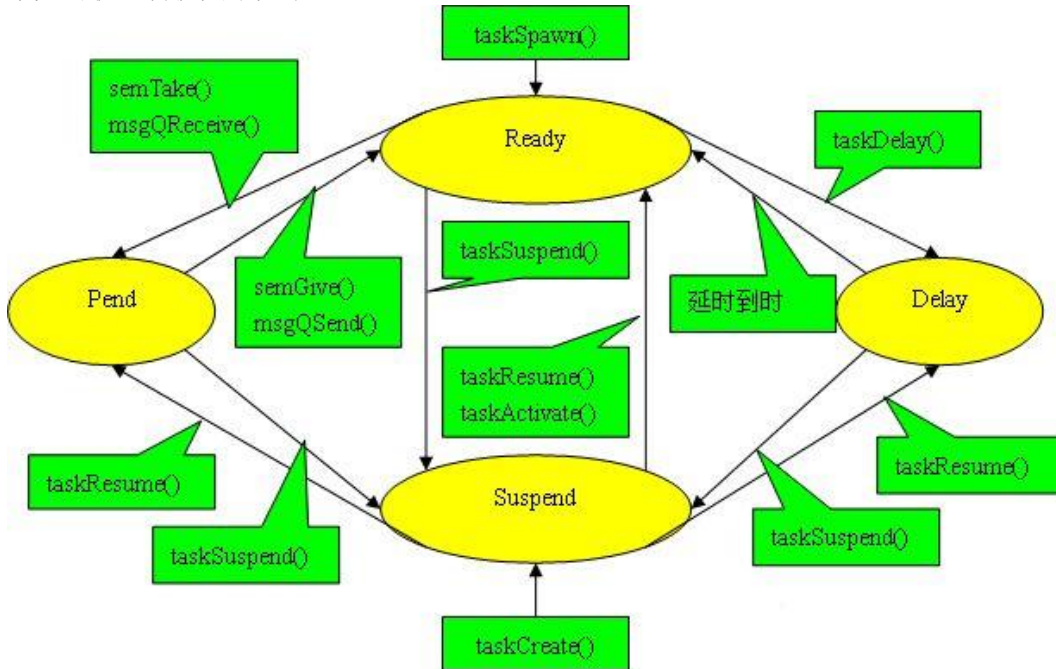
```
-> i
```

NAME	ENTRY	TID	PRI	STATUS
tIsr0	4269a8	e69cb0	0	PEND
tJobTask	4848b1	e6dea0	0	PEND
tExcTask	483da7	5aa1a0	0	PEND
tLogTask	logTask	e74790	0	PEND
tShell0	shellTask	eefa00	1	READY
tWdbTask	4f9731	eda740	3	PEND
tErfTask	4309f4	e71150	10	PEND
ipcom_tick>	51773d	e98450	20	PEND
tUxdbgTask	44116e	ed4540	25	PEND
tNet0	ipcomNetTask	e7a1e8	50	SUSPEND
ipcom_sysl>	44577f	e63810	50	PEND
tNetConf	47724d	ec75e0	50	PEND
t1	test	e9f520	100	SUSPEND

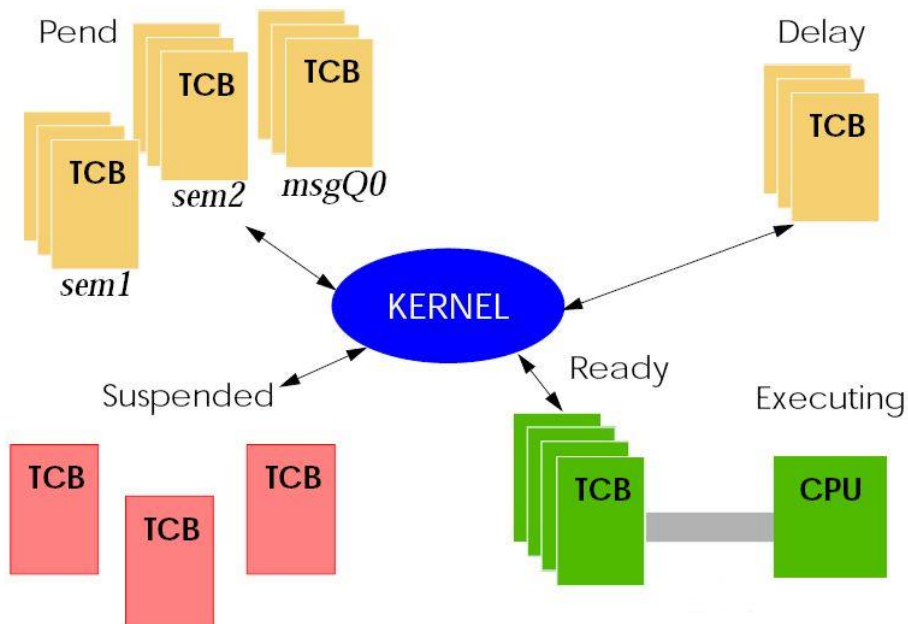
```
value = 0 = 0x0
```

应用程序或开发工具可以通过调用系统函数来改变任务的状态，例如通过 `taskSpawn()` 创建的任务会立即进入 Ready 队列，而通过 `taskCreate()` 或 `taskOpen(VX_TASK_NOACTIVATE)` 创建的任务只是在 Suspend 队列中实例化，然后需要通过 `taskActivate()` 才能进入 Ready 队列。

常见的状态转换关系如下：



VxWorks 的内核维护着当前每个任务的状态，将它们放入不同的队列。而处于 Ready 队列最前端的任务就是正在使用 CPU 执行的任务，因此把这个任务处于的状态又可以叫做 Executing (执行态)。



从上图可以看到，在不同队列里排队的，其实就是每个任务的 TCB。