

1.分区看门狗

1.1 概述

看门狗机制，是计算机可靠性领域中极为简单而又重要的机制，一般被用来监测系统此时的运行情况，是否出现卡死在某一个任务或者代码里，是否发生故障。

被监测的目标，周期性的重置计数值，如果计数值溢出，说明被监测目标在某一处卡住，或者发生故障，用户可以安装自己的超时处理回调函数，去处理该故障。

每个应用分区 **vm** 可以配置一个分区看门狗，超时后，会调用用户给分区看门狗安装的超时处理的回调函数。

1.2 编程接口

为分区看门狗安装超时触发的回调，并设置回调超时：

原型：

```
T_VMK_ReturnCode TTOS_WatchdogInstallCallback(  
T_WORD timeout, T_BOOL (*callback)(T_VOID * ctx), T_VOID * ctx  
)
```

参数：

timeout：看门狗回调超时时间

callback：看门狗回调函数，该函数返回值决定是否立即重启，TRUE 为立即执行用户策略，否则超时执行用户策略

返回值：

其他：失败。

TTOS_OK：成功。

使用说明：

本口在调用 **TTOS_StartWatchdog()** 之前使用。

启动分区看门狗，并设置超时时间：

原型：

```
T_VMK_ReturnCode TTOS_StartWatchdog(  
T_UWORD maxFeeddogTime, T_BOOL timeout_stop  
)
```

参数：

maxFeeddogTime：看门狗超时时间，时间为：该参数值 * 系统每 tick 的秒数

timeout_stop: 超时后的执行策略，为 **TRUE** 时超时后停止 TTOS，为 **FALSE** 时超时后重启 TTOS

返回值:

其他: 失败。

TTOS_OK: 成功。

使用说明:

该函数会为当前分区创建一个看门狗，可以配置超时时间，即多长时间不喂狗会造成超时，可以配置超时策略，即超时后执行的方式。

停止看门狗:

原型:

T_VMK_ReturnCode **TTOS_StopWatchdog**(T_VOID)

参数:

无

返回值:

其他: 失败。

TTOS_OK: 成功。

使用说明:

该函数会自动寻找当前分区配置的看门狗，无需传入参数。

喂看门狗

原型:

T_VMK_ReturnCode **TTOS_FeedWatchdog**(T_VOID)

参数:

无

返回值:

其他: 失败。

TTOS_OK: 成功。

使用说明:

该函数会自动寻找当前分区配置的看门狗，无需传入参数。