

英创 ARM9 嵌入式主板看门狗的使用

英创提供的系列 ARM9 嵌入式主板，包括有 EM9000 嵌入式主板、EM9260 嵌入式主板都预装了正版 WinCE 操作系统，ARM9 嵌入式主板的所有通讯接口均配置有标准的 WinCE 驱动程序。EM9000 和 EM9260 都具有有硬件看门狗的功能，本文侧重于介绍在 WinCE 下 ARM9 嵌入式主板 Watchdog 的使用。

EM9260 直接使用了 CPU 芯片内部的独立看门狗定时器，最长定时间隔为 16 秒。系统调试模式启动时，看门狗被禁止，运行模式启动时，看门狗为激活状态，且由 WinCE 内核的 Watchdog 线程对看门狗进行刷新。此模式可以防止应用程序占用 CPU 的死循环，但对应用程序异常退出或挂起没有作用。

应用程序可通过 EM9260 的专用 API 函数来接管对看门狗的操作，使之更为全面的监管应用程序行为的有效性。应用程序接管看门狗后，建议按 8 秒的间隔对看门狗进行刷新操作。具体代码操作如下：

```
EM9XXX_WDT_INFO    WatchdogInfo;

BOOL StartEM9260WDT( )
{
    BOOL        bResult;
    LPVOID      lpParam;

    bResult = TakeoverWatchdog( &WatchdogInfo );

    // 接管操作失败，该函数返回

    if( bResult==FALSE )
        return bResult;

    if( WatchdogInfo.dwWatchDogPeriod == 0 )
    {
        // 接管操作成功，但是 dwWatchDogPeriod==0，表明系统看门狗被禁止，该
        // 函数返回。

        return FALSE;
    }
}
```

```

}

lpParam = (LPVOID)&WatchdogInfo;
// 创建 WDT 刷新线程
m_hWDTThread = CreateThread( NULL,0, WatchdogRefreshThread, lpParam,
                            0, NULL );

if( m_hWDTThread == INVALID_HANDLE_VALUE )
    return FALSE;

return TRUE;
}

// WDT 刷新线程
DWORD WINAPI WatchdogRefreshThread( LPVOID lpParam )
{
    EM9XXX_WDT_INFO* pWatchdogInfo = (EM9XXX_WDT_INFO*)lpParam;
    for(;;)
    {
        // 用户可以根据实际情况增加调用重载 WDT 函数的条件，以确保应用程序
        // 正常运行，在此例中条件为 WDTFlag
        if( WDTFlag == 0 )
            pWatchdogInfo->pfnKickWatchDog( );
        // 挂起本线程
        Sleep(pWatchdogInfo->dwWatchDogPeriod);
    }
    return 0;
}

```

EM9000 的 CPU (EP9315) 芯片内的 WDT 的定时间隔仅仅 250 ms，如果加载到内核，可能会对系统性能产生影响，因此在 EM9000 的内核中没有加载 WDT 线程。如果应用程序需要使用 WDT，可参考本文中 EM9260 看门狗的使用原则，直接调用 EM9000 提供的 API 函数：

```
//  
// 功能描述：启动看门狗定时器，本函数只需调用一次。  
//          WDT 的持续时间为 250ms，WDT 超时将产生 250ms 的复位脉冲  
// 返回值 = 1，设置成功  
int EM9000_StartWatchDog( );  
  
//  
// 重载系统看门狗定时器 (WATCH DOG TIMER)，必须在 WDT 超时前 8ms 进行的  
// 重载操作才是有效的。  
void EM9000_KicktheDog( );
```