

WATCHDOG 设备

WATCHDOG 简介

硬件看门狗 (watchdog timer) 是一个定时器，其定时输出连接到电路的复位端。在产品化的嵌入式系统中，为了使系统在异常情况下能自动复位，一般都需要引入看门狗。

当看门狗启动后，计数器开始自动计数，在计数器溢出前如果没有被复位，计数器溢出就会对 CPU 产生一个复位信号使系统重启（俗称“被狗咬”）。系统正常运行时，需要在看门狗允许的时间间隔内对看门狗计数器清零（俗称“喂狗”），不让复位信号产生。如果系统不出问题，程序能够按时“喂狗”。一旦程序跑飞，没有“喂狗”，系统“被咬”复位。

一般情况下软件可配置该模块使能或去使能。

访问看门狗设备

应用程序根据提供的库函数接口来访问看门狗硬件，相关接口如下所示：

函数名称	描述
<code>extern void wdt_init(WDG_TYPE* WDG, uint32_t period_ms);</code>	初始化看门狗设备
<code>extern void wdt_enable(WDG_TYPE *WDG);</code>	使能看门狗设备
<code>extern void wdt_disable(WDG_TYPE *WDG);</code>	关闭看门狗设备
<code>extern void wdt_feed(WDG_TYPE *WDG);</code>	喂食看门狗设备

初始化看门狗

使用看门狗设备之前，必须执行初始化操作，使用下列函数来初始化看门狗设备：

```
void wdt_init(WDG_TYPE* WDG, uint32_t period_ms)
{
    CHECK_PARAM(PARAM_WDG(WDG));

    WDG->WIV = 0xFFFFFFFFU - period_ms*32768U/1000;

    /*wdt reload initial counter*/
    WDG->WFD |= WDG_FEED_MASK;
}
```

喂食看门狗

定期喂食看门狗，让系统正常运行，使用下列函数来初始化看门狗：

```
void wdt_feed(WDG_TYPE *WDG)
{
    CHECK_PARAM(PARAM_WDG(WDG));

    WDG->WFD |= WDG_FEED_MASK;
}
```

使能看门狗

当应用程序初始完看门狗后，使用下列函数来使能看门狗：

```
void wdt_enable(WDG_TYPE *WDG)
{
    CHECK_PARAM(PARAM_WDG(WDG));

    WDG->CTR |= (WDG_ENABLE_MASK);
}
```

关闭看门狗

当应用程序完成看门狗操作后，可以关闭看门狗设备，通过下列函数完成：

```
void wdt_disable(WDG_TYPE *WDG)
{
    CHECK_PARAM(PARAM_WDG(WDG));

    WDG->CTR &= ~(WDG_ENABLE_MASK);
}
```

看门狗设备使用实例

1. 初始化设备并设置看门狗溢出时间。
2. 使能看门狗。
3. 喂狗一次:系统在初始化后需要喂狗一次才能正常运行。
4. 喂狗:在溢出时间之前进行操作。

```
#include <stdio.h>
#include "pulpino.h"
#include "watchdog.h"
```

```
void check_ok()
{
    printf("check_ok");
}
int main(int argc, char **argv)
{
    printf("begin\n");
    WDG_Cmd(UC_WATCHDOG, ENABLE);
    WDG_SetReload(UC_WATCHDOG, 0xFFFF0000);
    wdt_enable();
    WDG_FEED(UC_WATCHDOG);
    printf("end\n");

    while(1){
        delay_ms(300);
        WDG_FEED(UC_WATCHDOG);
    }
    return 0;
}
```

注意事项:

1. UC8088 看门狗计数器使用32.768KHz时钟为计数器源时钟，因此，即使系统处于SLEEP状态也依然会保持计数。
2. 最大计数周期为 $2^{32} * \text{period}(32.768\text{KHz})$ 约等于 131072 秒。