

Subsystem Design

MSPM0 MCU 用作看门狗



1 说明

本示例展示了如何将 MSPM0 用作独立看门狗计时器。看门狗功能由 MSPM0 的集成计时器模块实现，用于检测程序是否失控或停止运行。GPIO 捕获喂狗输入脉冲，并向计时器提供信号。若喂狗失败，输出 IO 会触发复位引脚。下载此示例的代码。

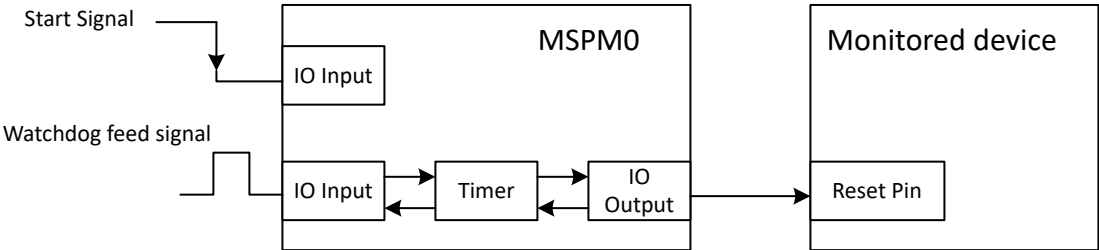


图 1-1. 系统功能方框图

2 所需外设

此应用需要两个计时器和三个器件引脚。

表 2-1. 外设

子块功能	外设	注释
捕获使能信号	(2x) GPIO	检测启动信号和看门狗喂狗信号
控制复位引脚	(1x) GPIO	输出信号
看门狗计时	(1x) 计时器	用作看门狗内部计时器

3 兼容器件

如果存在所需的外设，任何 MSPM0 器件和 EVM 或 Launchpad 都可以使用该子系统。

4 设计步骤

1. 初始化计时器。将计时器模式配置为单次触发递减计数并启用中断。
2. 确定看门狗超时窗口。此参数是确定喂狗频率的起始点。在本例中，我们选择看门狗超时窗口 = 1s。
3. 确定输出有效电压。此参数用于连接受监控器件的复位引脚。在本例中，我们选择复位有效电压为低电平。在此情况下，若喂狗失败，M0 将输出低电平信号。
4. 配置计时器时钟频率。计算公式 $F_{clock} = Sysclock / \text{计时器时钟预分频器} \times \text{超时}$ 。
5. 设置看门狗触发信号，这是用于决定是否启动看门狗功能的标志。

5 设计注意事项

- 看门狗窗口时间标度设置：建议将最小值设定在 1/32kHz 范围内（适用于低系统时钟频率场景）。
- 输出有效电压：本代码示例中，有效电压设定为低电平，且延迟时间为 10ms。有效电压和延迟时间可根据实际需求自定义，但需满足被监控器件的复位要求。
- 计时器计数器模式设置为单触发递减计数。如果需要，可以在 sysconfig 中更改此设置。

6 软件流程图

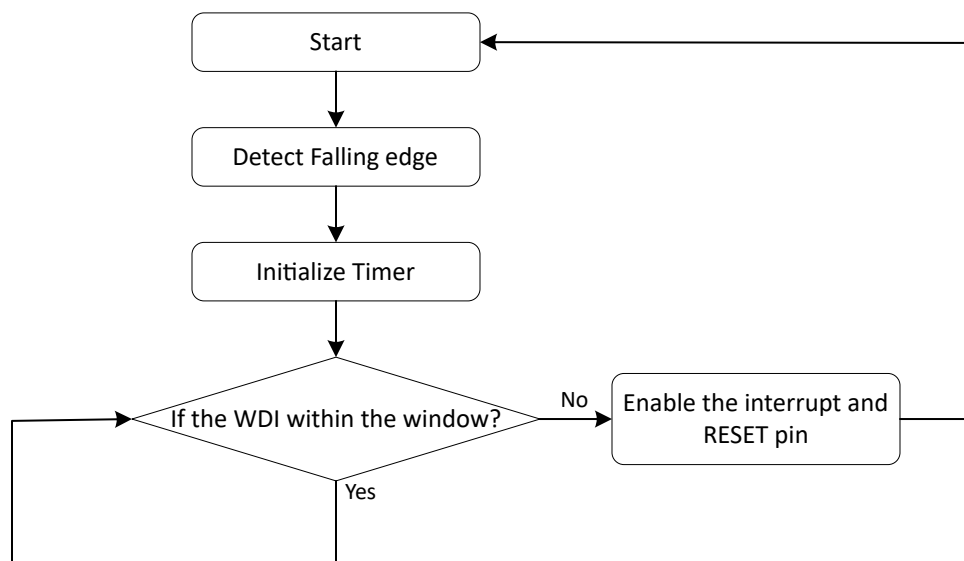


图 6-1. 应用软件流程图

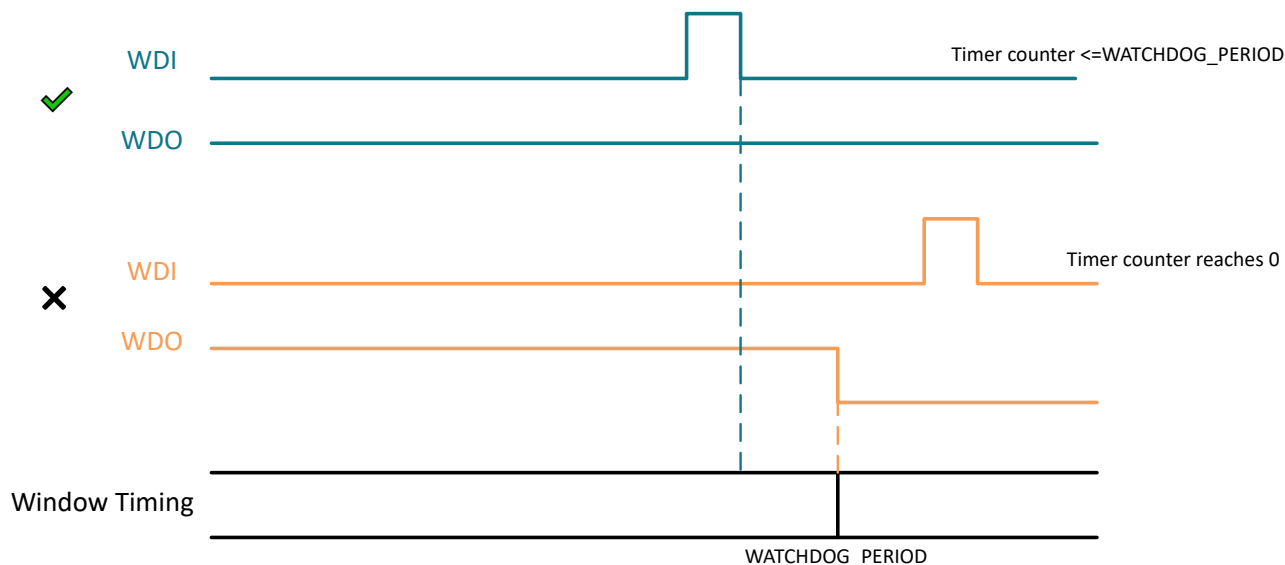


图 6-2. 看门狗计时器图

图 6-1 展示了应用程序流程图。图 6-2 展示了看门狗计时器图。在图底部是已配置的有效窗口，只有当看门狗输入信号处于该窗口内时，才视为喂狗成功。如果在计时器递减计数至零仍未收到看门狗输入信号，则输出引脚会触发被监控器件的复位引脚。

7 应用代码

在主函数中，我们启用 GPIO 中断以检测启动信号。一旦检测到启动信号，便初始化并启动看门狗。此后，每当检测到喂狗信号时，我们就重新加载计时器以防止其溢出。

```
/*
 * Subsystem example shows basic WDT functionality.
 * It enables two input pins, a WDT_Start and a WDT_WDI, as well
 * as an WDT_WDO output pin to be connected to the reset of the external device.
 *
 *
 * PA27: Input - WDT_Start
 * PA24: Input - WDT_WDI
 * PA0: Output - WDT_WDO (open-drain)
 *
 */
#include "watchdog_hal.h"

int main(void)
{
    SYSCFG_DL_init();
    DL_GPIO_disableInterrupt(GPIOA, WDT_WDI_PIN);

    /* Enable start signal capture */
    DL_GPIO_enableInterrupt(GPIOA, WDT_Start_PIN);
    NVIC_EnableIRQ(WDT_INT_IRQN);

    while (1) {
        __WFI();
    }
}

void watchdog_timer_INST_IRQHandler(void)
{
    switch (DL_TimerG_getPendingInterrupt(watchdog_timer_INST)) {
        case DL_TIMER_IIDX_ZERO:
            /* WDT Timed out */
            watchdog_timeout();
            break;
        default:
            break;
    }
}

void GPIOA_IRQHandler(void)
{
    switch (DL_GPIO_getPendingInterrupt(GPIOA)) {
        case WDT_Start_IIDX:
            /* WDT start pin detected */
            watchdog_start();
            break;

        case WDT_WDI_IIDX:
            /* WDT feed pin detected */
            watchdog_reload();
            break;

        default:
            break;
    }
}
```

以下是看门狗的初始化、重新加载及超时的处理方式。

```
#include "watchdog_hal.h"
#include <stdio.h>

void watchdog_start(void)
{
    /* disable WDT feed pin until WDT started */
    DL_GPIO_disableInterrupt(GPIOA, WDT_Start_PIN);
    DL_GPIO_clearInterruptStatus(GPIOA, WDT_WDI_PIN | WDT_Start_PIN);

    /* Enable WDT Feed pin */
    DL_GPIO_enableInterrupt(GPIOA, WDT_WDI_PIN);
    NVIC_EnableIRQ(watchdog_timer_INST_INT_IRQN);

    DL_Timer_setLoadValue(watchdog_timer_INST, WATCHDOG_PERIOD-1);
    DL_TimerG_startCounter(watchdog_timer_INST);
}

void watchdog_timeout(void)
{
    DL_TimerG_stopCounter(watchdog_timer_INST);
    DL_GPIO_disableInterrupt(GPIOA, WDT_WDI_PIN);

    /* Toggle WDT output fail signal */
    DL_GPIO_clearPins(WDT_PORT, WDT_WDO_PIN); //output low
    delay_cycles(WDO_RESET_TIME);
    DL_GPIO_setPins(WDT_PORT, WDT_WDO_PIN); //output high

    /* enable WDT start pin */
    DL_GPIO_clearInterruptStatus(GPIOA, WDT_WDI_PIN | WDT_Start_PIN);
    DL_GPIO_enableInterrupt(GPIOA, WDT_Start_PIN);
}

void watchdog_reload(void)
{
    DL_TimerG_stopCounter(watchdog_timer_INST);
    DL_Timer_setLoadValue(watchdog_timer_INST, WATCHDOG_PERIOD-1);
    DL_TimerG_startCounter(watchdog_timer_INST);
}
```

8 结果

下面内容显示了代码的执行结果。如图 8-1 所示，启动引脚的下降沿表示计时器开始。当看门狗输入信号 (WDI) 在窗口内生成信号时，看门狗输出信号保持高电压。但如果未馈送 WDI，WDO 会复位，且 WDI 延迟时间窗口为 10ms，这表示看门狗计时器处于工作状态。



图 8-1. 测试结果

9 其他资源

- 德州仪器 (TI), [下载 MSPM0 SDK](#)
- 德州仪器 (TI), [详细了解 SysConfig](#)
- 德州仪器 (TI), [MSPM0C LaunchPad™](#)
- 德州仪器 (TI), [MSPM0L LaunchPad™](#)
- 德州仪器 (TI), [MSPM0G LaunchPad™](#)
- 德州仪器 (TI), [MSPM0 Academy](#)

10 E2E

请访问 TI 的 E2E™ 支持论坛来查看讨论并发布新主题，以获得在设计中使用 MSPM0 器件的技术支持。

11 商标

LaunchPad™ and E2E™ are trademarks of Texas Instruments.

所有商标均为其各自所有者的财产。

12 修订历史记录

注：以前版本的页码可能与当前版本的页码不同

Changes from Revision * (July 2025) to Revision A (August 2025)	Page
• 删除了“兼容器件”部分.....	1

重要通知和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的相关应用。严禁以其他方式对这些资源进行复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
版权所有 © 2025，德州仪器 (TI) 公司