



# 实验报告

课程名称: 机电一体化

课程编号: ME333

实验题目: 独立看门狗实验

学 号: 12313215

姓 名: 梁家源

专 业: 机器人工程

指导教师: 曾千里 柯文德

实验成绩:

实验日期: 2025 年 3 月 19 日

# 独立看门狗实验

## 一、实验目的

1. 掌握 STM32 独立看门狗 (IWDG) 的工作原理及配置方法。
2. 通过按键喂狗操作验证看门狗的复位功能。
3. 设置看门狗溢出时间不少于 2 秒；串口显示累计喂狗次数；每次喂狗后触发 LED 灯 (DS1) 闪烁一次。

## 二、实验仪器和用具

主要仪器设备：

1. 计算机
2. Keil uVision 软件
3. XCOM/SSCOM 串口助手

## 三、实验方法与步骤

### 1. 独立看门狗的作用

在 MCU 系统中，外界干扰可能导致程序跑飞或陷入死循环。独立看门狗通过定时器复位电路，在程序未及时“喂狗”时强制复位系统，确保系统稳定性。

### 2. 溢出时间计算

公式：Tout=预分频系数×重装载值/LSI 频率

### 3、程序初始化

配置系统时钟 (216MHz)、延时、串口 (115200bps)。初始化 LED (DS0、DS1) 和按键 (KEY\_UP)。配置看门狗：预分频系数 128，重装载值 500，溢出时间 2 秒。启动看门狗。

```
Cache_Enable();           //打开L1-Cache
HAL_Init();               //初始化HAL库
Stm32_Clock_Init(432, 25, 2, 9); //设置时钟, 216Mhz
delay_init(216);          //延时初始化
uart_init(115200);        //串口初始化
LED_Init();               //初始化LED
KEY_Init();               //初始化按键
delay_ms(100);            //延时100ms再初始化看门狗, LED0的变化“可见”
IWDG_Init(IWDG_PRESCALER_128, 500); //分频数为128, 重载值为500, 溢出时间为2s
```

#### 4、主循环逻辑

检测 KEY\_UP 按键按下，执行喂狗操作。每次喂狗后：串口输出累计喂狗次数；DS1 闪烁一次（亮 100ms 后熄灭）。

```
while(1)
{
    if(KEY_Scan(0)==WKUP_PRES)      //如果WK_UP按下，喂狗
    {
        IWDG_Feed();           //喂狗
        count++;
        printf("已喂狗 %d 次\r\n", count);
        LED1(0);
        delay_ms(100);
        LED1(1);
    }
    delay_ms(10);
}
```

## 四、实验分析及结论

### 一、实验结果分析

#### 1. 功能实现验证

正常喂狗时：DS0 保持常亮，系统正常运行。按下 KEY\_UP 按键后，DS1 闪烁一次，串口输出累计喂狗次数（如“已喂狗 5 次”）。并且在按下重置键后计数归零。

未喂狗时：看门狗溢出时间 2 秒后，系统复位，DS0 短暂熄灭后重新点亮（复位后重新初始化）。



## 二、实验结论

本实验通过配置 STM32 独立看门狗（IWDG），验证了其在程序异常时的复位功能。设置预分频系数 128 与重装载值 500，实现 2 秒溢出时间，确保系统稳定性。通过按键喂狗触发 LED（DS1）闪烁及串口累计次数显示，协调多外设控制，成功模拟看门狗的工作流程。实验表明，合理设置喂狗间隔可有效避免系统死机，提升抗干扰能力。该机制在嵌入式系统中对保障可靠性具有重要价值，为复杂环境下的程序容错设计提供了实践基础。