
本章教程通过串口调试助手打印显示程序运行状态，具体现象如下：

- 若计数器值在上窗口值和下窗口值0X40之间的时候，进行喂狗操作，计数器重新计数，程序正常运行，串口打印显示：The program runs normally;
- 若程序在快要减到0X3F时候通过中断函数进行喂狗操作，则串口打印显示：Execute interrupt function The program runs normally;
- 若程序没有在窗口时间内进行喂狗操作，则程序会不停进行复位。

1、WWDG简介及相关函数介绍

窗口看门狗（WWDG）通常用来监测系统运行的软件故障，例如外部干扰、不可预见的逻辑错误等情况。其需要在一个特定的窗口时间（有上下限）内进行计数器刷新（喂狗），否则早于或者晚于这个窗口时间看门狗电路都会产生系统复位。

CH32V103的窗口看门狗具有可编程的7位自减型计数器；具有双条件复位，当计数器值小于0X40，或者计数器值在窗口时间外被重装载；具有唤醒提前通知功能（EWI），用于及时喂狗动作防止系统复位。

CH32V103的窗口看门狗运行基于一个可编程的7位的自减型计数器，其挂载在 APB1 总线下，计数时基 WWDG_CLK 来源（PCLK1/4096）时钟的分频，分频系数在配置寄存器 WWDG_CFGR 中的 WDG TB[1:0]域设置。递减计数器处于自由运行状态，无论看门狗功能是否开启，计数器一直循环递减计数。关于CH32V103窗口看门狗具体信息，可参考CH32V103应用手册。

本章教程通过调用窗口看门狗库函数中相关函数进行程序编写，关于库函数相关介绍如下：

1. void WWDG_DeInit(void);
2. void WWDG_SetPrescaler(uint32_t WWDG_Prescaler);

3. void WWDG_SetWindowValue(uint8_t WindowValue);
4. void WWDG_EnableIT(void);
5. void WWDG_SetCounter(uint8_t Counter);
6. void WWDG_Enable(uint8_t Counter);
7. FlagStatus WWDG_GetFlagStatus(void);
8. void WWDG_ClearFlag(void);

复制代码

1.1、void WWDG_DeInit(void)

功 能：将WWDG外围寄存器初始化为其默认重置值

输 入：无

1.2、void WWDG_SetPrescaler(uint32_t WWDG_Prescaler)

功 能：设置WWDG预分频器

输 入：WWDG_预分频器：指定WWDG预分频器。

1.3、void WWDG_SetWindowValue(uint8_t WindowValue)

功 能：设置WWDG窗口值。

输 入：WindowValue：指定要与downcounter比较的窗口值，该值必须小于0x80

1.4、void WWDG_EnableIT(void)

功 能：启用WWDG早期唤醒中断（EWI）

输 入：无

1.5、void WWDG_SetCounter(uint8_t Counter)

功 能：设置WWDG计数器值

输 入：Counter：指定看门狗计数器值，该值必须是介于0x40和0x7F之间的数字。

1.6、void WWDG_Enable(uint8_t Counter)

功 能：启用WWDG并加载计数器值

输 入：Counter：指定看门狗计数器值，该值必须是介于0x40和0x7F之间的数字。

1.7、FlagStatus WWDG_GetFlagStatus(void)

功 能：检查是否设置了早期唤醒中断标志

输 入：无

1.8、void WWDG_ClearFlag(void)

功 能：清除提前唤醒中断标志

输 入：无

2、硬件设计

窗口看门狗属于CH32V103内部资源，无需进行任何硬件设计，只需

通过串口调试助手打印显示观察其运行状态即可。

3、软件设计

本教程中窗口看门狗主要通过窗口时间内喂狗或在中断服务函数内喂狗保证系统程序正常运行，其具体程序如下：

wwdg.h文件

```
1. #ifndef __WWDG_H
2. #define __WWDG_H
3.
4. #include "ch32v10x_conf.h"
5. #define WWDG_CNT 0x7f
6.
7. void WWDG_Config(uint8_t tr, uint8_t wr, uint32_t prv);
8. void WWDG_NVIC_Config(void);
9. void WWDG_IRQHandler(void);
10. void WWDG_Feed(void);
11.
12. #endif
```

复制代码

wwdg.h文件主要是相关定义及函数声明。

wwdg.c文件

```
1. #include "wwdg.h"
2.
3. void WWDG_IRQHandler(void) __attribute__((interrupt("WCH-
   Interrupt-fast")));
4.
5. //初始化窗口看门狗， tr：递减计数器值； wr：上窗口值； prv：分频系
   数
6. void WWDG_Config(uint8_t tr, uint8_t wr, uint32_t prv)
7. {
8.     RCC_APB1PeriphClockCmd(RCC_APB1Periph_WWDG, ENABLE);
   //WWDG时钟使能
9.
10.    WWDG_SetCounter( tr );    //设置WWDG递减计数器值
11.    WWDG_SetWindowValue( wr ); //设置WWDG上窗口值
12.    WWDG_SetPrescaler( prv ); //设置WWDG预分频器值
13.    WWDG_Enable(WWDG_CNT);    //设置计数器初始值并使能看门狗
14.    WWDG_ClearFlag();         //清除提前唤醒中断标志
15.    WWDG_NVIC_Config();       //初始化窗口看门狗NVIC
16.    WWDG_EnableIT();          //开启窗口看门狗中断
17. }
18.
19. //初始化看门狗NVIC，配置优先级
20. void WWDG_NVIC_Config(void)
21. {
22.     NVIC_InitTypeDef NVIC_InitStructure;
```

```

23. NVIC_InitStructure.NVIC_IRQChannel = WWDG_IRQn;
    //WWDG中断
24. NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0; //抢占优先级 0
25. NVIC_InitStructure.NVIC_IRQChannelSubPriority = 0;      //响应优先级 0
26. NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;        //使能通道1中断
27. NVIC_Init(&NVIC_InitStructure);                        //NVIC初始化
28. }
29.
30. //WWDG中断服务函数
31. void WWDG_IRQHandler(void)
32. {
33.     WWDG_Feed();
34.     WWDG_ClearFlag();      //清除提前唤醒中断标志位
35.     printf("Execute interrupt function\r\n");
36. }
37.
38. //喂狗
39. void WWDG_Feed(void)
40. {
41.     WWDG_SetCounter( WWDG_CNT );//喂狗，刷新递减计数器的值，
    设置成最大WWDG_CNT=0X7F
42. }
43.

```

复制代码

wwdg.c文件主要是WWDG相关配置，其具体配置流程如下：

1、进行WWDG配置，包括开启WWDG时钟、设置计数器值、窗口值和预分频器值等；

```

1. //初始化窗口看门狗，tr：递减计数器值；wr：上窗口值；prv：分频系数
2. void WWDG_Config(uint8_t tr, uint8_t wr, uint32_t prv)
3. {
4.     RCC_APB1PeriphClockCmd(RCC_APB1Periph_WWDG, ENABLE);
    //WWDG时钟使能
5.
6.     WWDG_SetCounter( tr );    //设置WWDG递减计数器值
7.     WWDG_SetWindowValue( wr ); //设置WWDG上窗口值
8.     WWDG_SetPrescaler( prv ); //设置WWDG预分频器值
9.     WWDG_Enable(WWDG_CNT);    //设置计数器初始值并使能看门狗
10.    WWDG_ClearFlag();          //清除提前唤醒中断标志
11.    WWDG_NVIC_Config();        //初始化窗口看门狗NVIC
12.    WWDG_EnableIT();           //开启窗口看门狗中断
13. }

```

复制代码

2、WWDG NVIC初始化配置，设置优先级分组；

```
1. //初始化看门狗NVIC，配置优先级
2. void WWDG_NVIC_Config(void)
3. {
4.     NVIC_InitTypeDef NVIC_InitStructure;
5.     NVIC_InitStructure.NVIC_IRQChannel = WWDG_IRQn;
        //WWDG中断
6.     NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0; //抢占优先级 0
7.     NVIC_InitStructure.NVIC_IRQChannelSubPriority = 0;      //响应优先级 0
8.     NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;        //使能通道1中断
9.     NVIC_Init(&NVIC_InitStructure);                        //NVIC初始化
10. }
```

复制代码

3、编写WWDG喂狗函数；

```
1. //喂狗
2. void WWDG_Feed(void)
3. {
4.     WWDG_SetCounter( WWDG_CNT );//喂狗，刷新递减计数器的值，
        设置成最大WWDG_CNT=0X7F
5. }
```

复制代码

4、编写中断服务函数

```
1. //WWDG中断服务函数
2. void WWDG_IRQHandler(void)
3. {
4.     WWDG_Feed();
5.     WWDG_ClearFlag();      //清除提前唤醒中断标志位
6.     printf("Execute interrupt function\r\n");
7. }
```

复制代码

wdg.c文件主要进行窗口看门狗相关函数配置

main.c文件

```
1. int main(void)
2. {
3.     u8 wwdg_tr,wdg_wr;
4.
5.     NVIC_PriorityGroupConfig(NVIC_PriorityGroup_2);
6.     Delay_Init();
7.     USART_Printf_Init(115200);
8. }
```

```

9.   printf("SystemClk:%d\r\n",SystemCoreClock);
10.  printf("WWDG Test\r\n");
11.
12.  WWDG_Config(0x7f,0x5f,WWDG_Prescaler_8);    /* 36M/8/4096
    */
13.  wwdg_wr=WWDG->CFGR & 0x7F;
14.  while(1)
15.  {
16.      Delay_Ms(70);
17.      printf("The program runs normally\r\n");
18.      wwdg_tr=WWDG->CTLR & 0x7F;
19.      if(wwdg_tr<wwdg_wr)
20.      {
21.          WWDG_Feed();
22.      }
23.  }
24. }

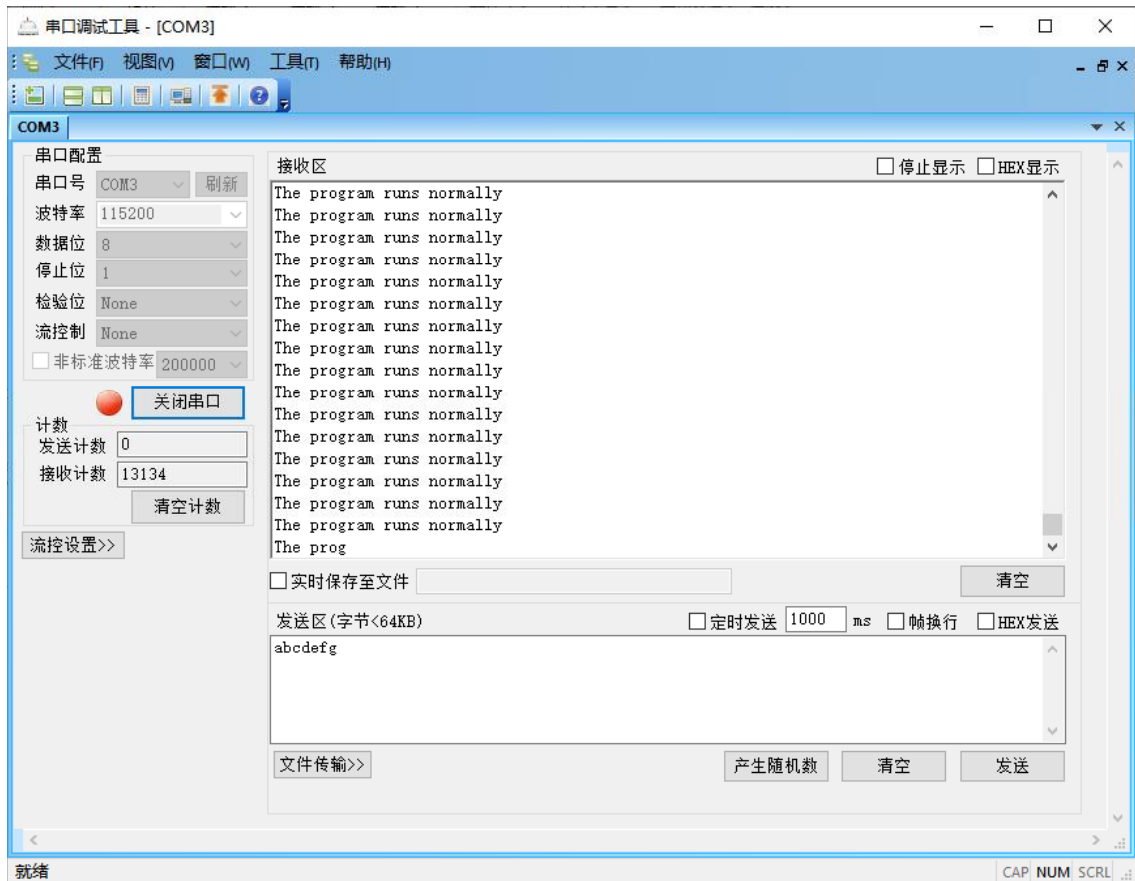
```

复制代码

main.c文件主要进行相关函数初始化以及根据判断进行喂狗操作。其中，while循环中延时函数值不宜设置过大，其值根据计数器值、上窗口值和下窗口值进行设置。若设置过大，则会进入中断，本程序在中断中调用喂狗函数以保障程序正常运行，若无喂狗函数，则程序会不停复位。

4、下载验证

将编译好的程序下载到开发板并复位，打开串口调试助手，可看到串口调试助手一直在打印输出：The program runs normally，如图所示：



将main函数while循环中延时函数值增大大，函数将会进入中断通过中断服务函数进行喂狗，串口调试助手打印显示如下：

