

SYSTEM文件夹介绍

1, sys文件夹介绍（掌握）

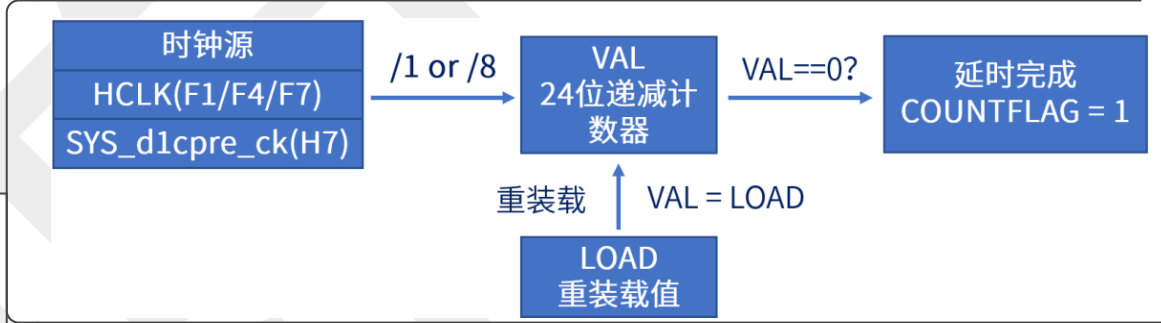
函数分类	相关函数	功能
中断类函数	sys_nvic_set_vector_table()	设置中断向量表地址
	sys_intx_enable()	开启所有中断
	sys_intx_disable()	关闭所有中断(但是不包括fault和NMI中断)
低功耗类函数	sys_wfi_set()	执行: WFI指令(执行完该指令进入低功耗状态)
	sys_standby()	进入待机模式
	sys_soft_reset()	系统软复位
设置栈顶地址函数	sys_msr_msp()	设置栈顶地址
系统时钟初始化函数	sys_stm32_clock_init()	设置系统时钟
Cache配置函数(F7/H7)	sys_cache_enable()	使能I-Cache和D-Cache, 开启D-Cache强制透写

2, deley文件夹介绍（掌握）

2.1, deley文件夹函数简介

函数分类	相关函数	功能
使用OS
不使用OS	delay_init()	初始化系统滴答定时器
	delay_us()	用系统滴答定时器实现微秒延时
	delay_ms()	用微秒延时函数实现毫秒延时

SysTick，即系统滴答定时器，包含在M3/4/7内核里面，核心是一个24位的递减计数器



每次VAL到0时，VAL自动从LOAD重载！开始新一轮递减计数！

2.2, SysTick工作原理

2.3, SysTick寄存器介绍

- SysTick控制及状态寄存器(CTRL)
- SysTick重载数值寄存器(LOAD)
- SysTick当前数值寄存器(VAL)
- SysTick校准数值寄存器(CALIB)

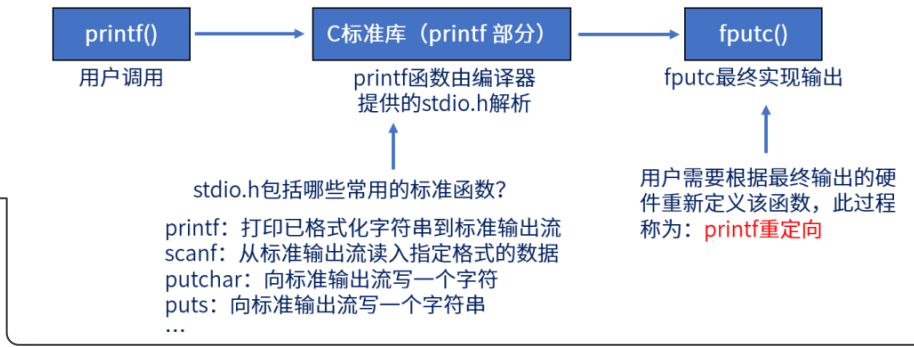
2.4, delay_init()函数

2.5, delay_us()函数

2.6, delay_ms()函数

3, usart文件夹介绍（掌握）

3.1, printf函数输出流程



3.2, printf的使用

- 1, printf("字符串\\n");
- 2, printf("输出控制符", 输出参数);
- 3, printf("输出控制符1输出控制符2...", 输出参数1, 输出参数2, ...);
- 4, printf("非输出控制符 输出控制符 非输出控制符", 输出参数);
- 5, 如何输出%d、\和双引号

3.3, printf函数支持

- 1, 避免使用半主机模式 两种方法：微库法、代码法
- 2, 实现fputc函数 实现单个字符输出

4, 课堂总结（了解）