









cortex-M0+的CMSIS 提供的函数

1. CPSIE i void \_\_enable\_irq(void): 这个函数用于启用中断（IRQ）。通过调用这个函数，可以允许中断在处理器中被触发和处理。
2. CPSID i void \_\_disable\_irq(void): 这个函数用于禁用中断（IRQ）。通过调用这个函数，可以阻止中断在处理器中被触发和处理。
3. ISB void \_\_ISB(void): 这个函数用于插入指令同步栅栏（Instruction Synchronization Barrier），确保处理器按照指定的顺序执行指令。
4. DSB void \_\_DSB(void): 这个函数用于插入数据同步栅栏（Data Synchronization Barrier），确保内存中的数据同步。
5. DMB void \_\_DMB(void): 这个函数用于插入数据内存栅栏（Data Memory Barrier），确保内存中的数据同步和一致性。
6. NOP void \_\_NOP(void): 这个函数用于插入空操作指令（No Operation），在程序中不执行任何操作，通常用于延时或调试目的。
7. REV uint32\_t \_\_REV(uint32\_t int value): 这个函数用于反转（翻转）32位无符号整数的字节顺序。
8. REV16 uint32\_t \_\_REV16(uint32\_t int value): 这个函数用于反转（翻转）16位无符号整数的字节顺序。
9. REVSH uint32\_t \_\_REVSH(uint32\_t int value): 这个函数用于反转（翻转）16位有符号整数的字节顺序，并符号扩展。
10. SEV void \_\_SEV(void): 这个函数用于发送事件（Send Event），用于唤醒处于休眠状态的处理器。
11. WFE void \_\_WFE(void): 这个函数用于等待事件（Wait For Event），让处理器进入休眠状态等待事件唤醒。
12. WFI void \_\_WFI(void): 这个函数用于等待中断（Wait For Interrupt），让处理器进入休眠状态等待中断唤醒。

NVIC\_ISPR强制中断进入挂起状态，并显示哪些中断处于挂起状态

NVIC\_ICER禁用中断，并显示启用了哪些中断

NVIC\_ISER启用中断，并显示启用了哪些中断

NVIC\_IPR0-NVIC\_IPR7寄存器为每个中断提供一个 8 位优先级字段。这些寄存器只能通过文字访问

挂起的中断将保持挂起状态，直到出现以下情况之一：

• 处理器输入中断的 ISR。这会将中断的状态从挂起更改为活动。然后：

– 对于电平敏感的中断，当处理器从 ISR 返回时，NVIC 对中断信号进行采样。如果信号被置位，中断的状态将变为挂起，这可能会导致处理器立即重新进入 ISR。否则，中断的状态将变为非活动状态。

– 对于脉冲中断，NVIC 继续监控中断信号，如果这是脉冲信号，则中断状态将变为挂起和活动。在这种情况下，当处理器从 ISR 返回时，中断的状态将更改为挂起，这可能会导致处理器立即重新进入 ISR。如果当处理器位于 ISR 中时中断信号没有脉冲，则当处理器从 ISR 返回时，中断的状态将变为非活动状态

• 软件写入相应的中断清除待处理寄存器位。对于电平敏感的中断，如果中断信号仍处于置位状态，则中断的状态不会改变。否则，中断的状态将变为非活动状态。对于脉冲中断，中断的状态将更改为：

• 如果状态为挂起，则为非活动。

• 活动，如果状态为活动且待处理

SCR 控制进入和退出低功耗状态的功能。

SEVEONPEND 在挂起位上发送事件

0：只有启用的中断或事件才能唤醒处理器，禁用的中断被排除在外。

1 = 启用事件和所有中断（包括禁用的中断）都可以唤醒处理器。当事件或中断变为挂起时，事件信号将从 WFE 中唤醒处理器。如果处理器未等待事件，则会注册该事件并影响下一个 WFE。处理器还会在执行 SEV 指令或外部事件时唤醒。

SysTick 计数器的正确初始化顺序为： 1. 程序重新加载值。2.明确电流值。3. 程序控制和状态寄存器

Data Fault Status Register（DFSR）中，以下标志位代表了不同的数据访问异常类型：

1. **HALTED**：表示处理器在执行指令时遇到断点（breakpoint）或单步调试（single-step debug）操作导致暂停。这个标志位用于指示处理器是否因为调试操作而停止。
2. **VCATCH**：表示处理器遇到了向量捕获（Vector Catch）异常，即在执行向量表中的异常处理程序时发生了另一个异常。这个标志位用于指示异常处理程序的执行过程中出现了异常。
3. **BKPT**：表示处理器遇到了断点（breakpoint）异常，即在执行过程中遇到了设置的断点。这个标志位用于指示程序中的断点触发了异常。
4. **EXTERNAL**：表示处理器遇到了外部触发的数据访问异常，可能是外设或外部情况导致的异常。这个标志位用于指示外部因素导致的异常。
5. **DWTTRAP**：表示处理器执行了DWT（Data Watchpoint and Trace）触发的数据访问异常，即在监视特定数据地址访问时触发了异常。这个标志位用于指示数据监视功能引起的异常。