

零死角玩转STM32—M4系列



SysTick—系统定时器

淘宝：fire-stm32.taobao.com

野火论坛：www.firebbs.cn

主讲内容



01

SysTick简介

02

SysTick功能框图讲解

03

SysTick定时实验讲解

参考资料: 《零死角玩转STM32》

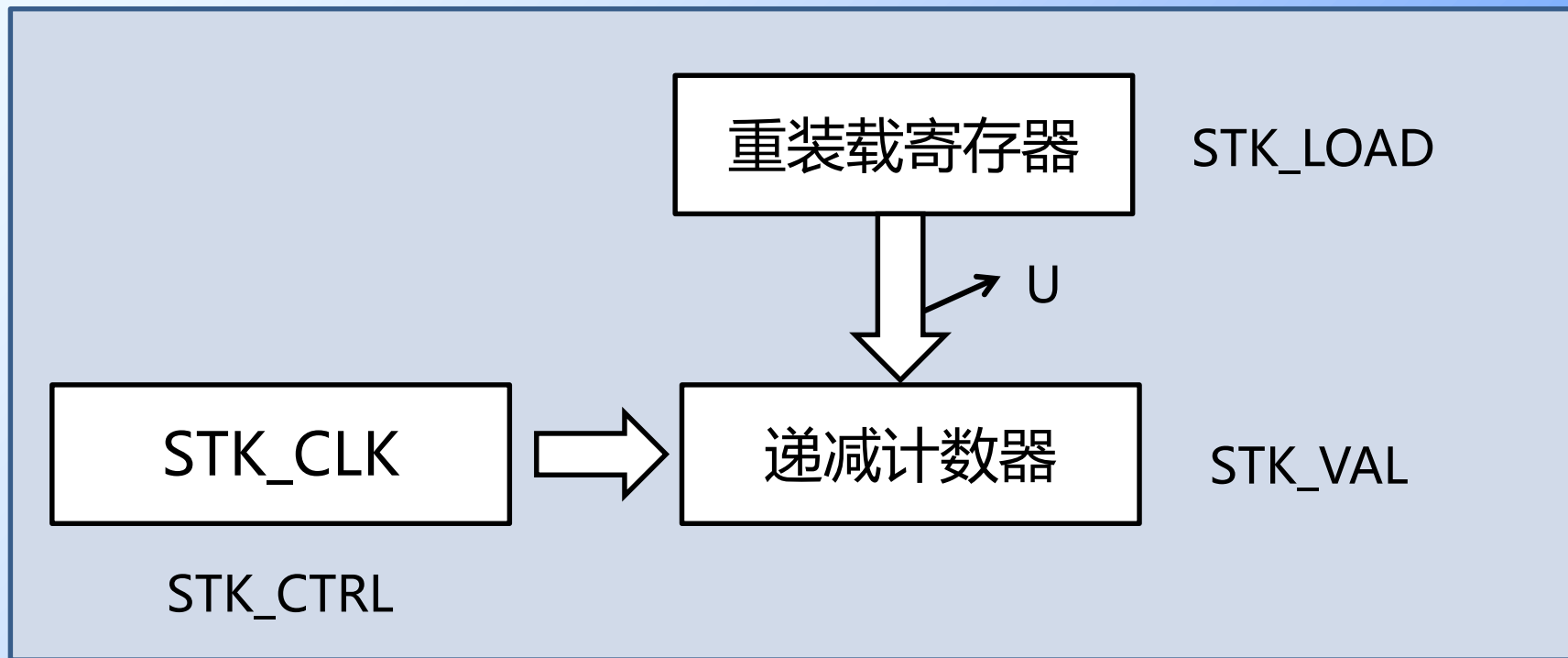
“SysTick—系统定时器” 章节

SysTick简介



SysTick：系统定时器，24位，只能递减，存在于内核，嵌套在NVIC中，所有的Cortex-M内核的单片机都具有这个定时器。

SysTick功能框图



SysTick寄存器



表 18-2 SysTick 控制及状态寄存器

位段	名称	类型	复位值	描述
16	COUNTFLAG	R/W	0	如果在上次读取本寄存器后， SysTick 已经计到了 0， 则该位为 1。
2	CLKSOURCE	R/W	0	时钟源选择位， 0=AHB/8， 1=处理器时钟 AHB
1	TICKINT	R/W	0	1=SysTick 倒数计数到 0 时产生 SysTick 异常请求， 0=数到 0 时无动作。也可以通过读取 COUNTFLAG 标志位来确定计数器是否递减到 0
0	ENABLE	R/W	0	SysTick 定时器的使能位

表 18-3 SysTick 重装载数值寄存器

位段	名称	类型	复位值	描述
23:0	RELOAD	R/W	0	当倒数计数至零时， 将被重装载的值

表 18-4 SysTick 当前数值寄存器

位段	名称	类型	复位值	描述
23:0	CURRENT	R/W	0	读取时返回当前倒计数的值， 写它则使之清

SysTick库函数



SysTick唯一的固件库函数

```
/*  
- STATIC_INLINE uint32_t SysTick_Config(uint32_t ticks)  
{  
    if ((ticks - 1) > SysTick_LOAD_RELOAD_Msk) return (1); /* Reload value impossible */  
  
    SysTick->LOAD = ticks - 1; /* set reload register */  
    NVIC_SetPriority (SysTick_IRQn, (1<<__NVIC_PRIO_BITS) - 1); /* set Priority for SysTick Interrupt */  
    SysTick->VAL = 0; /* Load the SysTick Counter Value */  
    SysTick->CTRL = SysTick_CTRL_CLKSOURCE_Msk |  
                    SysTick_CTRL_TICKINT_Msk |  
                    SysTick_CTRL_ENABLE_Msk; /* Enable SysTick IRQ and SysTick Timer */  
    return (0); /* Function successful */  
}
```

实验设计

让SysTick产生1s的定时，让LED亮灭。



1-初始化LED相关的GPIO

2-配置SysTick (STK_CTRL寄存器)

3-编写中断服务函数

4-main函数

零死角玩转STM32—M4系列



THANKS

野火论坛 : www.firebbs.cn

淘宝 : fire-stm32.taobao.com