### SysTick嘀嗒定时器实现精确延时

1. SysTick 定时器介绍

Systick定时器就是系统滴答定时器，一个24 位的倒计数定时器，计到0 时，将从RELOAD 寄存器中自动重装载定时初值。只要不把它在SysTick 控制及状态寄存器中的使能位清除，就永不停息，即使在睡眠模式下也能工作。

1. 定时器/计数器 计时原理

定时器从本质上来讲其实就是一个计数器，先给它设置一个初始计数值，它每收到一个脉冲，计数器就会减1，当减到0时，产生一个“叮”的信号，告知我们定时时间到。

定时时间 = 接收到的脉冲个数 \* 单个脉冲的周期



STM32F103的工作时钟为72MHz，经过8分频后给SysTick 计数器用。此时计数器接收到的脉冲信号的频率为72/8=9Mhz, 单个脉冲的周期= 1 / （9MHz）= 1/9 us

如果给SysTick 计数器设置初始计数值RELOAD为9，则计数器每接收9个计数脉冲所需要的时间为9 \* （1/9 us）= 1 us.

如果给SysTick 计数器设置初始计数值RELOAD为9\*1000，则计数器每接收9\*1000个计数脉冲所需要的时间为9 \*1000\* （1/9 us）= 1000 us = 1 ms.

Systick定时器是24 位的倒计数定时器，RELOAD最大为2^24 = 16777216

16777216 \* （1/9 us）=1864135 us = 1.86 秒

1. SysTick 寄存器结构

typedef struct

{

 uint32\_t CTRL; /\*SysTick 控制和状态寄存器\*/

 uint32\_t LOAD; /\*SysTick 重装载值寄存器\*/

 uint32\_t VAL; /\*SysTick 当前值寄存器\*/

 uint32\_t CALIB; /\*SysTick 校准值寄存器\*/

} SysTick\_Type;

SysTick 控制和状态寄存器SysTick->CTRL的关键信息位:



1. SysTick实现精确延时程序: Delay.c

\*\*

 \* 功能：初始化Systick定时器

 \* 参数：None

 \* 返回值：None

 \*/

void initSysTick(void)

{

 SysTick\_CLKSourceConfig(SysTick\_CLKSource\_HCLK\_Div8); //设置时钟源8分频

 SysTick->CTRL |= SysTick\_CTRL\_ENABLE\_Msk; //开定时器

 SysTick->LOAD = 9; //随意设置一个重装载值

}

/\*\*

 \* 功能：us级别延时

 \* 参数：xus：要延时的时间

 \* 返回值：None

 \*/

void Delay\_us(u32 xus)

{

 SysTick->LOAD = 9 \* xus; //计数9次为1us，则重装载值xus\*9

 SysTick->VAL = 0; //计数器归零

 while (!(SysTick->CTRL & SysTick\_CTRL\_COUNTFLAG\_Msk)); //空循环,等待计数完成

}

/\*\*

 \* 功能：ms级别延时

 \* 参数：xms：要延时的时间

 \* 返回值：None

 \*/

void Delay\_ms(u32 xms)

{

 SysTick->LOAD = 9000; //计9次为1us，1000次为1ms

 SysTick->VAL = 0; //计数器归零

 while (xms--)

 {

 while (!(SysTick->CTRL & SysTick\_CTRL\_COUNTFLAG\_Msk)); //等待单次计数完成

 }

}