Assignment

Day 8

Monday, 12 September 2023

By: Ibrahim Tarek

SV G2 / Intake #3

Table of Contents

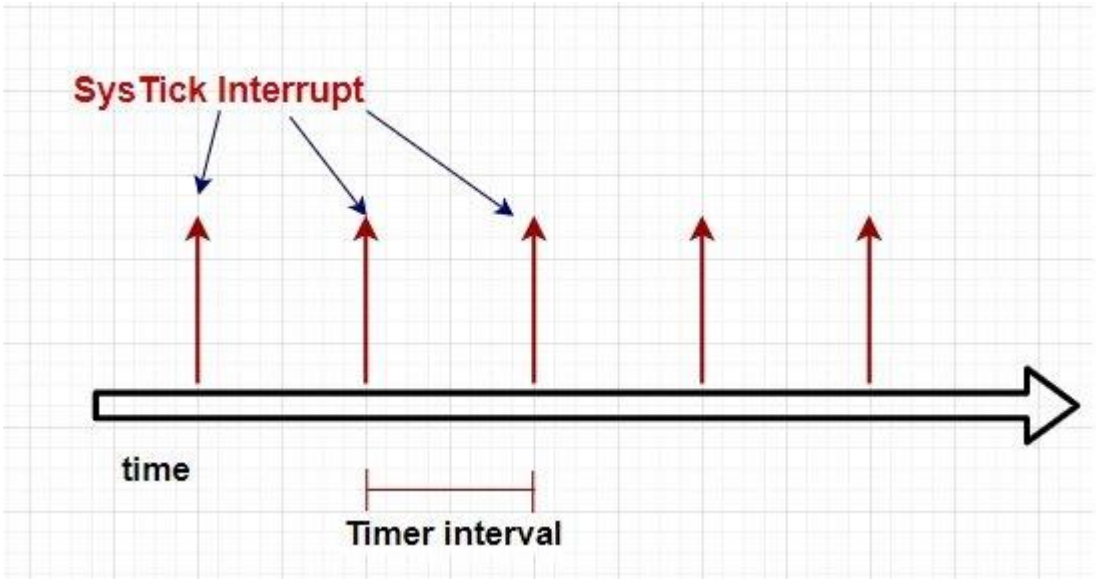
1. Executive Summary

This report discusses each of the following topics:

1- The main features system timer.
2- Delay functions (milli – micro) seconds.

2. System Timer

System timer (systick) timer is a dedicated hardware-based timer which is built inside the ARM Cortex M4 CPU and can be used to generate an interrupt at a fixed interval of time. As shown in the figure below:



The systick timer will generate interrupts after a specified time and time settings can be done using the Systick control register.

2.1. Applications

Following are the some of the applications of systick timer:

1- Generate delays
2- Introducing time delays between two events
3- Calling periodic tasks after a specified time e.g. RTOS tick timer
4- Used to implements time-share scheduling in RTOS

2.2. How does the Systick timer work?

TM4C123GH6PM provides a 24 bit timer. Therefore, the maximum value that can be loaded to the load register of system timer is 2^(24-1) which is . Hence, it starts decrement value by one from its initial set value and generates an interrupt when values reaches zero.

In all ARM cortex M4 microcontrollers, the nested vectored interrupt controller manages interrupts or exceptions generated by peripherals or GPIO pins. Hence, whenever a Systick timer interrupt occurs, the nested interrupt vector controller transfers the control of CPU to the related interrupt service routine.

As we have seen in the TM4C123G microcontroller startup file tutorial that the startup file contains a weak definition of all interrupts and exception routines. These ISR routines are defined with weak attributes that means we can redefine them inside our main code according to our requirement. But the name of a specific ISR routine should remain the same.

3. Delay functions (m – u) seconds

3.1. Delay function (milli second)

```
void MSYSTICK_voidDelayms(uint32_t Copy_u32Delay)
{
    uint32_t Local_u32PreloadValue = Copy_u32Delay * (SYSTICK_FREQ / 1000UL);
    SYSTICK -> VALUE = 0;
    SYSTICK -> LOAD = Local_u32PreloadValue & SYSTICK_RES;
    SET_BIT(SYSTICK -> CTRL, ENABLE);
    while(GET_BIT(SYSTICK -> CTRL, COUNT_FLAG) == 0);
    CLR_BIT(SYSTICK -> CTRL, ENABLE);
}
```

3.2. Delay function (micro second)

```
void MSYSTICK_voidDelayus(uint32_t Copy_u32Delay)
{
    uint32_t Local_u32PreloadValue = Copy_u32Delay * (SYSTICK_FREQ / 1000000UL);
    SYSTICK -> VALUE = 0;
    SYSTICK -> LOAD = Local_u32PreloadValue & SYSTICK_RES;
    SET_BIT(SYSTICK -> CTRL, ENABLE);
    while(GET_BIT(SYSTICK -> CTRL, COUNT_FLAG) == 0);
    CLR_BIT(SYSTICK -> CTRL, ENABLE);
}
```

4. References

4.1. SysTick Timer (System Timer) TM4C123G ARM Cortex M4 Microcontroller (microcontrollerslab.com)