Edward Ghazarossian
Christian Garcia

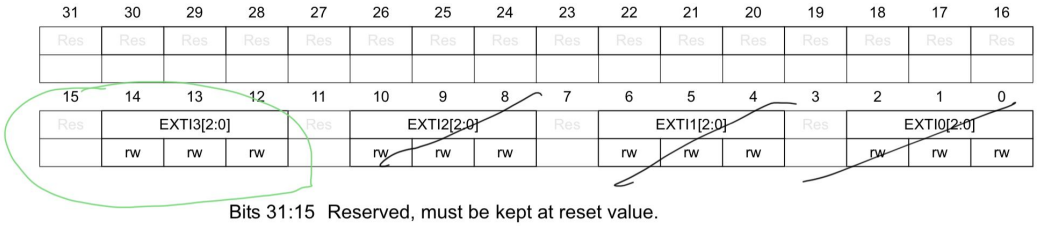
Lab 5

Register maps:

10.2.3 SYSCFG external interrupt configuration register 1 (SYSCFG_EXTICR1)

Address offset: 0x08

Reset value: 0x0000 0000



Bits 31:15 Reserved, must be kept at reset value.

Bits 14:12 **EXTI3[2:0]**: EXTI 3 configuration bits
These bits are written by software to select the source input for the EXTI3 external interrupt.
000: PA[3] pin
001: PB[3] pin
010: PC[3] pin
011: PD[3] pin
100: PE[3] pin
101: PF[3] pin
110: PG[3] pin
111: Reserved

Bit 11 Reserved, must be kept at reset value.

Problem 1:

Main:

```
        INCLUDE core_cm4_constants.s        ; Load Constant Definitions
        INCLUDE stm32l476xx_constants.s
        INCLUDE jstick.h
        INCLUDE leds.h

                AREA    main, CODE, READONLY
                EXPORT      __main
                ENTRY

__mainPROC
                ldr         r0,=RCC_AHB2ENR_GPIOBEN
                bl          portclock_en
                ldr         r0,=RCC_AHB2ENR_GPIOEEN
                bl          portclock_en
```

```
                ldr             r0,=GPIOB_BASE
                ldr             r1,=GPIO_MODER_MODER2_0
                bl              port_bit_pushpull
                ldr             r0,=GPIOE_BASE
                ldr             r1,=GPIO_MODER_MODER8_0
                bl              port_bit_pushpull

                bl              porta_init
                bl              exti3_init
                bl              exti0_init

endlessb                endless
                ENDP


EXTI3_IRQHandler PROC
                EXPORT      EXTI3_IRQHandler
                push    {lr}
                bl              red_tog
                pop             {lr}
                ldr             r2,=(EXTI_BASE+EXTI_PR1)
                mov             r1,#EXTI_PR1_PIF3
                str             r1,[r2]
                dsb
                bx              lr
                ENDP

EXTI0_IRQHandler PROC
                EXPORT      EXTI0_IRQHandler
                push    {lr}
                bl              green_tog
                pop     {lr}
                ldr     r2,=(EXTI_BASE+EXTI_PR1)
                mov             r1,#EXTI_PR1_PIF0
                str     r1, [r2]
                dsb
                bx              lr
                ENDP
```

```
        ALIGN
        AREA    myData, DATA, READWRITE

        ALIGN


    END
```

Jstick.s

```
exti0_init      PROC
                EXPORT      exti0_init
                ldr         r2,=(RCC_BASE+RCC_APB2ENR)
                ldr         r1,[r2]
                orr         r1,#RCC_APB2ENR_SYSCFGEN
                str         r1,[r2]
                ldr         r2,=(SYSCFG_BASE+SYSCFG_EXTICR0)
                ldr         r1,[r2]
                bic         r1,#0x00000007
                str         r1,[r2]
                ldr         r2,=(EXTI_BASE+EXTI_RTSR1)
                ldr         r1,[r2]
                orr         r1,#EXTI_RTSR1_RT0
                str         r1,[r2]
                ldr         r2,=(EXTI_BASE+EXTI_FTSR1)
                ldr         r1,[r2]
                bic         r1,#EXTI_FTSR1_FT0
                str         r1,[r2]
                ldr         r2,=(EXTI_BASE+EXTI_IMR1)
```

```
        ldr     r1,[r2]
        orr     r1,#EXTI_IMR1_IM0
        str     r1,[r2]
        ldr     r2,=(NVIC_BASE+NVIC_ISER0)
        ldr     r1,=(1<<6)
        str     r1,[r2]
        bx      lr
        ENDP

        ALIGN
        END
```

Jstick.h

```
        IMPORT  porta_init
        IMPORT  read_jstick
        IMPORT  exti3_init
        IMPORT  exti0_init

        END
```

Problem 2:

Main:

```
    INCLUDE core_cm4_constants.s
    INCLUDE stm32l476xx_constants.s
    INCLUDE jstick.h
    INCLUDE leds.h

            AREA    main, CODE, READONLY
            EXPORT      __main
            ENTRY

__mainPROC
        ldr     r0,=RCC_AHB2ENR_GPIOBEN
        bl      portclock_en
        ldr     r0,=RCC_AHB2ENR_GPIOEEN
        bl      portclock_en

        ldr     r0,=GPIOB_BASE
        ldr     r1,=GPIO_MODER_MODER2_0
        bl      port_bit_pushpull
```

```
                ldr             r0,=GPIOE_BASE
                ldr             r1,=GPIO_MODER_MODER8_0
                bl              port_bit_pushpull

                bl              porta_init
                bl              exti3_init
                bl              exti5_init

endlessb                endless
                ENDP

EXTI3_IRQHandler PROC
                EXPORT      EXTI3_IRQHandler
                push    {lr}
                bl              red_tog
                pop             {lr}
                ldr             r2,=(EXTI_BASE+EXTI_PR1)
                mov             r1,#EXTI_PR1_PIF3
                str             r1,[r2]
                dsb
                bx              lr
                ENDP

EXTI9_5_IRQHandler PROC
                EXPORT      EXTI9_5_IRQHandler
                ldr             r3,=(EXTI_BASE+EXTI_PR1)
                ldr             r4, [r3]
                tst             r4, #EXTI_PR1_PIF5
                beq     return
                push    {lr}
                bl              green_tog
                pop     {lr}
                ldr     r2,=(EXTI_BASE+EXTI_PR1)
                mov             r1,#EXTI_PR1_PIF5
                str     r1, [r2]
                dsb
return  bx              lr
                ENDP

                        ALIGN
```

```
        AREA    myData, DATA, READWRITE
        ALIGN
    END
```

Jstick.s:

```
exti5_init      PROC
                EXPORT      exti5_init
                ldr         r2,=(RCC_BASE+RCC_APB2ENR)
                ldr         r1,[r2]
                orr         r1,#RCC_APB2ENR_SYSCFGEN
                str         r1,[r2]
                ldr         r2,=(SYSCFG_BASE+SYSCFG_EXTICR0)
                ldr         r1,[r2]
                bic         r1,#0x00000070
                str         r1,[r2]
                ldr         r2,=(EXTI_BASE+EXTI_RTSR1)
                ldr         r1,[r2]
                orr         r1,#EXTI_RTSR1_RT5
                str         r1,[r2]
                ldr         r2,=(EXTI_BASE+EXTI_FTSR1)
                ldr         r1,[r2]
                bic         r1,#EXTI_FTSR1_FT5
                str         r1,[r2]
                ldr         r2,=(EXTI_BASE+EXTI_IMR1)
```

```
        ldr     r1,[r2]
        orr     r1,#EXTI_IMR1_IM5
        str     r1,[r2]
        ldr     r2,=(NVIC_BASE+NVIC_ISER0)
        ldr     r1,=(1<<23)
        str     r1,[r2]
        bx      lr
        ENDP
```

Jstick.h:

```
        IMPORT  porta_init
        IMPORT  read_jstick
        IMPORT  exti3_init
        IMPORT  exti5_init

        END
```

Problem 3:

```
tim2_init   PROC        ;initialize Timer 2 for this program and setup its interrupt
            EXPORT      tim2_init
            ldr         r2,=(RCC_BASE+RCC_APB1ENR1)             ;enable timer 2
clock
            ldr         r1,[r2]
            orr         r1,#RCC_APB1ENR1_TIM2EN
            str         r1,[r2]

            ldr         r2,=(TIM2_BASE+TIM_PSC)             ;Setup the prescaler.
Assuming a 4MHz clock, this gives 1ms timer ticks
            ldr         r1,=39
            str         r1,[r2]

            ldr         r2,=(TIM2_BASE+TIM_ARR)             ;Setup the reload.
Assuming a 1ms tick, this gives 1s overflows
            ldr         r1,=49
            str         r1,[r2]
```

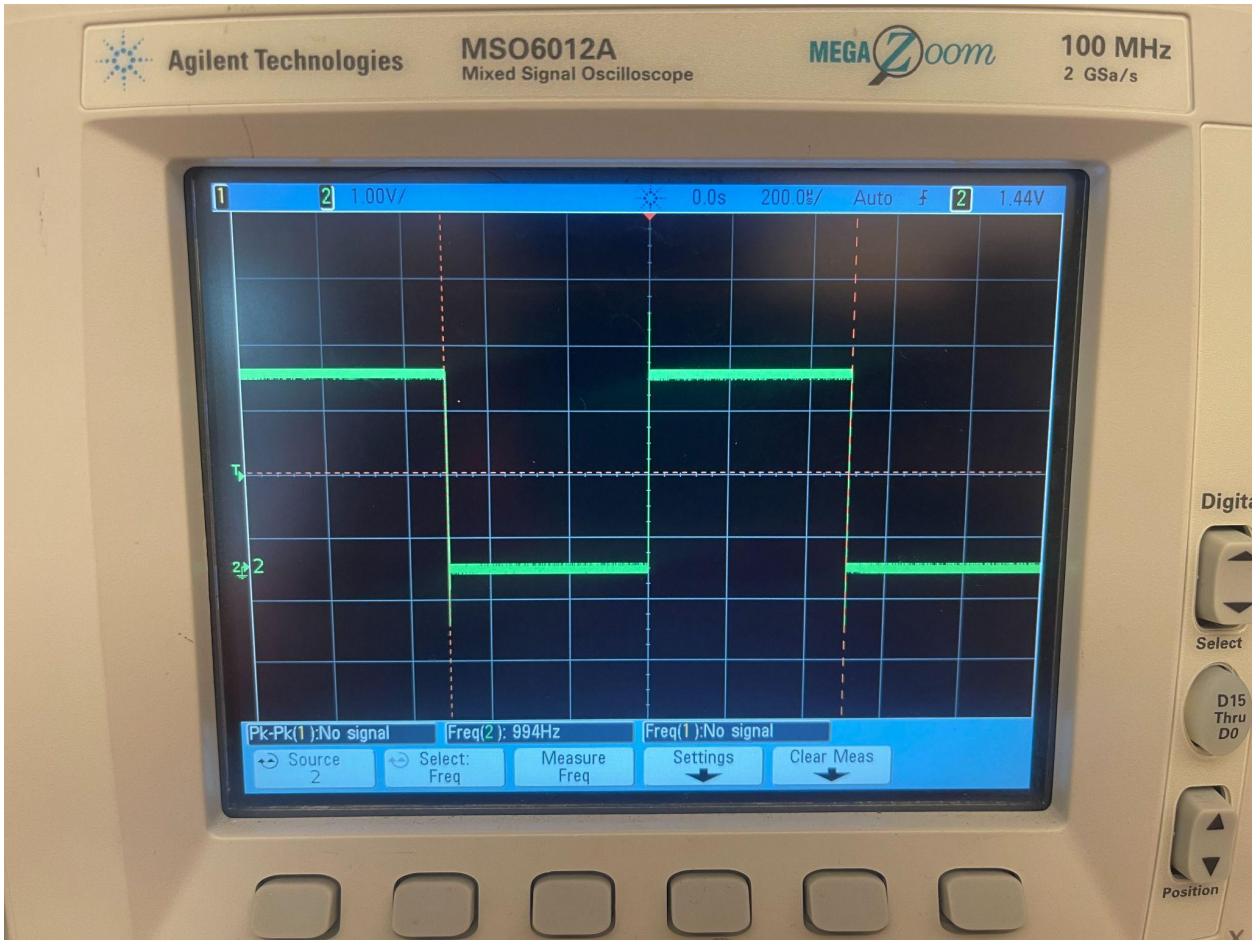
```
        ldr     r2,=(TIM2_BASE+TIM_CR1)         ;enable the counter in
control register 1
        ldr     r1,[r2]
        orr     r1,#TIM_CR1_CEN
        str     r1,[r2]

        ldr     r2,=(TIM2_BASE+TIM_DIER)        ;enable the timer
update interrupt
        ldr     r1,[r2]
        orr     r1,#TIM_DIER_UIE
        str     r1,[r2]

        ldr     r2,=(NVIC_BASE+NVIC_ISER0)  ;enable the TIM2 interrupt in
NVIC_ISER0
        ldr     r1,=(1<<28)
        str     r1,[r2]
        bx      lr

        ENDP
        ALIGN

        END
```


Agilent Technologies
MSO6012A
Mixed Signal Oscilloscope
MEGA Zoom
100 MHz
2 GSa/s
1.00V/
0.0s
200.0ns/
Auto
1.44V
Pk-Pk(1):No signal
Freq(2): 994Hz
Freq(1):No signal
Source 2
Select: Freq
Measure Freq
Settings
Clear Meas
Select
D15 Thru D0
Position