Christian Garcia

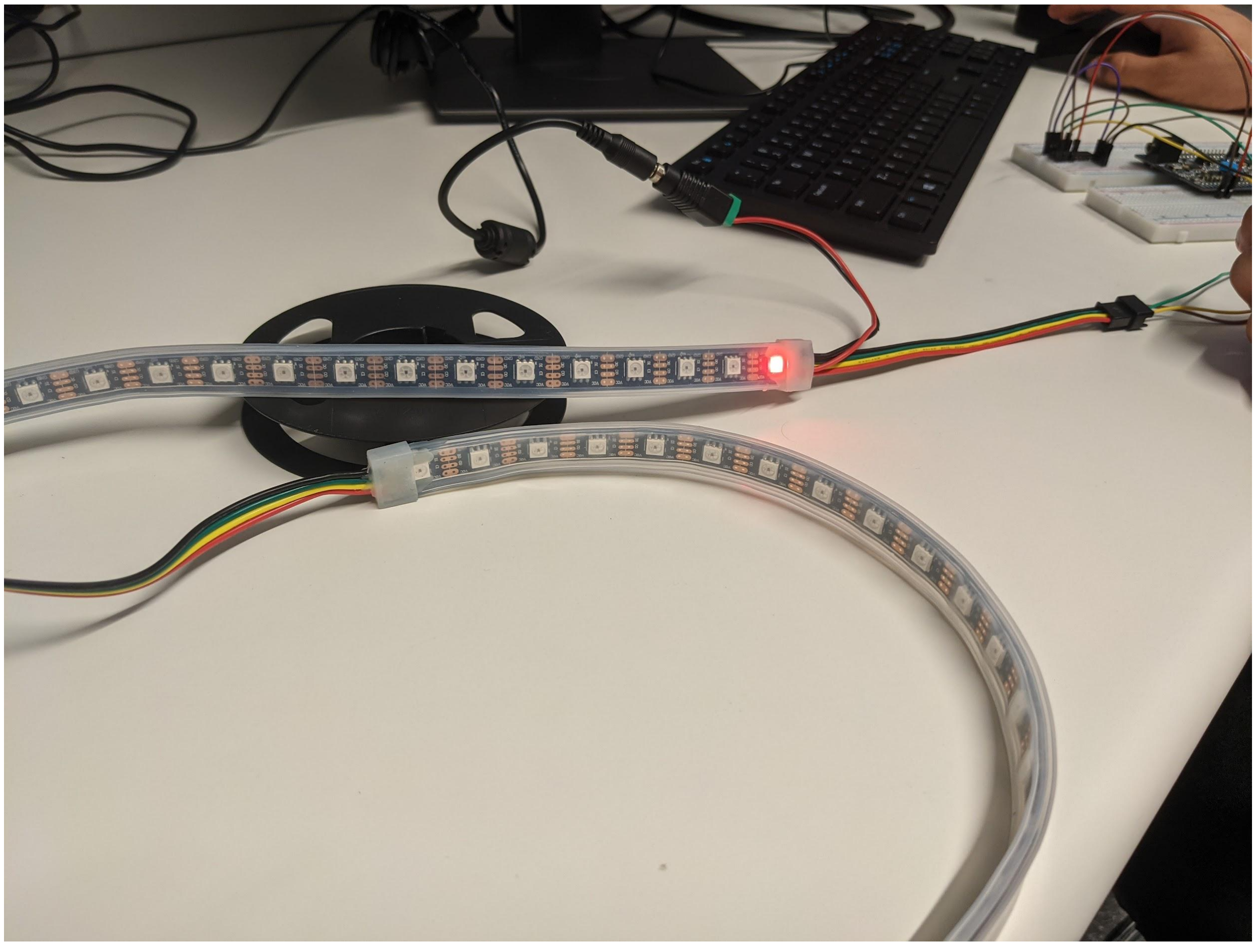
Edward Ghazarossian

ELEN 120 Lab

9 November 2021

Lab 6: Serial Communications

Problem 1:



Main:

```
__mainPROC

                    IMPORT spi32
```

```
                IMPORT spisw_init
                bl              spisw_init
                mov     r0, #0
                bl              spi32
                ldr             r0, =0xF00000FF
                bl              spi32
                mov             r0, #0xFFFFFFFF
                bl              spi32
                bx              lr
endless         b               endless




Init:
spisw_init      PROC
                EXPORT          spisw_init
                ldr     r0, =(RCC_BASE+RCC_AHB2ENR)
                ldr     r1, [r0]
                orr     r1, #RCC_AHB2ENR_GPIOEEN
                str     r1, [r0]
                ldr     r0, =(GPIOE_BASE+GPIO_MODER)
                ldr     r1, [r0]
```

```
bic     r1, r1, #(0x03<<(2*15))
orr     r1, r1, # (1<<(2*15))
str     r1, [r0]
ldr     r0, =(GPIOE_BASE+GPIO_MODER)
ldr     r1, [r0]
bic     r1, r1, #(0x03<<(2*13))
orr     r1, r1, # (1<<(2*13))
str             r1, [r0]
bx              lr
ENDP
```

Problem 2:

Main:

```
__mainPROC


            IMPORT spi32

            IMPORT spisw_init



            bl          spisw_init

            mov     r0, #0

            bl          spi32
```

```
                    mov     r4, #61
loop        ldr             r0, =0xF00000FF
                    bl              spi32
                    subs    r4, #1
                    beq     loop
                    ldr     r0, =0xF000FF00
                    bl              spi32
                    subs    r4, #1
                    beq     loop
                    ldr     r0, =0xF0FF0000
                    bl              spi32
                    subs    r4, #1
                    beq     loop
                    ldr             r0, =0xFFFFFFFF
                    bl              spi32
                    subs    r4, #1
                    beq     loop
                    bne     loop
                    bx              lr

endless     b               endless
```

Problem 3:

Main:

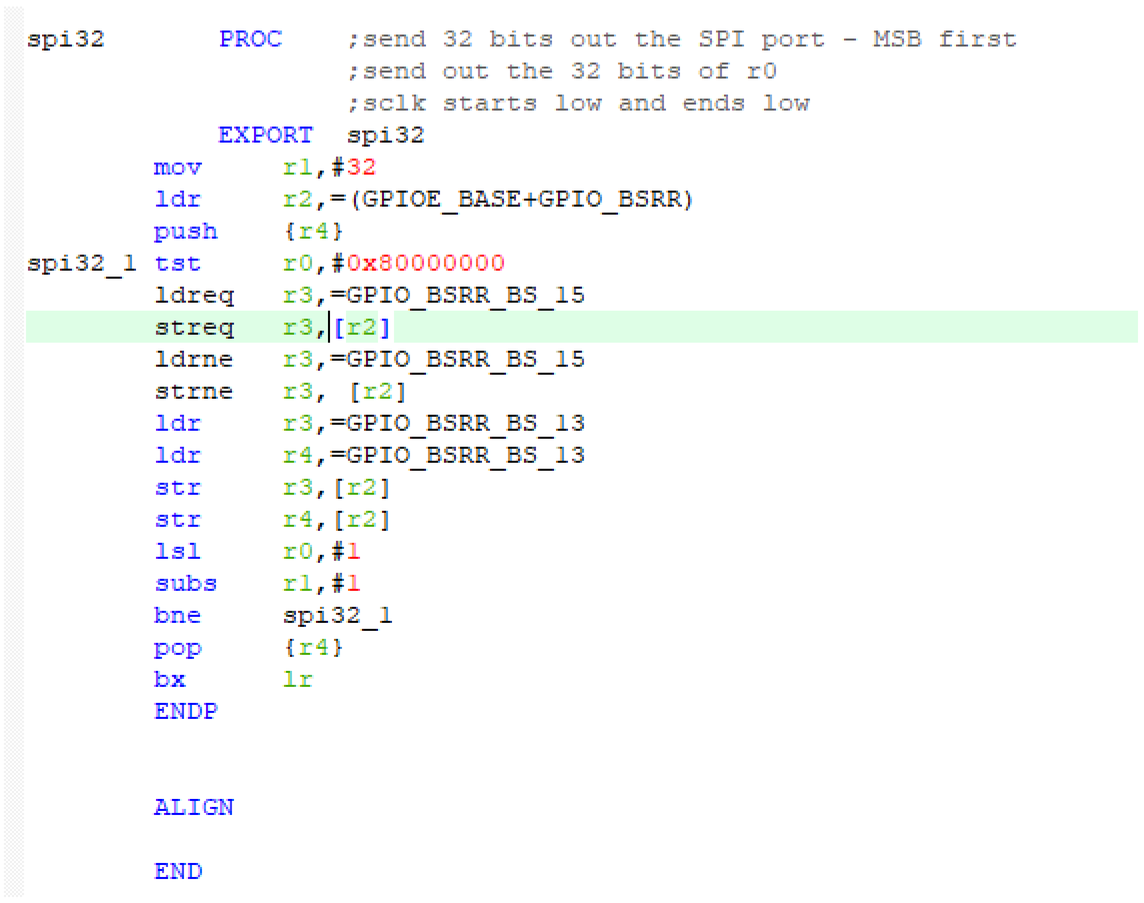
```
__mainPROC


                    IMPORT spi32

                    IMPORT spisw_init


                    bl          spisw_init

loop1       mov     r0, #0

                    bl          spi32

                    mov     r4, #61


loop2       mov             r0, #0xe4000000

                    ldr         r0, =0xF00000FF

                    ldr         r0, =0xF000FF00

                    bl          spi32


                    mov         r0, #0xe4000000

                    ldr     r0, =0xF000FF00

                    ldr         r0, =0xF0FF0000

                    bl          spi32


                    mov         r0, #0xe4000000

                    ldr     r0, =0xF0FF0000
```

```
                ldr             r0, =0xF00000FF
                bl              spi32


                ldr             r0, =0xe4FFFFFF
                bl              spi32


                subs    r4, #1
                bne     loop2


                mov     r0, #0xFFFFFFFF
                bl              spi32


                ldr     r8, =500000
loop3           subs    r8, #1
                cmp     r8, #0
                bne             loop3


                mov     r0, #0
                bl              spi32
                mov     r4, #61


loop4
                ldr             r0, =0xe4FFFFFF
```

```
bl              spi32


mov             r0, #0xe4000000

ldr     r0, =0xF0FF0000

ldr             r0, =0xF00000FF

bl              spi32


mov             r0, #0xe4000000

ldr     r0, =0xF000FF00

ldr             r0, =0xF0FF0000

bl              spi32


mov             r0, #0xe4000000

ldr             r0, =0xF00000FF

ldr             r0, =0xF000FF00

bl              spi32


subs    r4, #1

bne     loop4


mov     r0, #0xFFFFFFFF

bl              spi32
```

```
                    ldr       r8, =500000

loop5     subs      r8, #1

                    cmp       r8, #0

                    bne                 loop5


                    b loop1


endless   b                   endless


          ENDP

                    ALIGN

                    AREA    myData, DATA, READWRITE

                    ALIGN


counter   DCD                 10


     END
```

Problem 4:



```
spi32       PROC    ;send 32 bits out the SPI port - MSB first
                    ;send out the 32 bits of r0
                    ;sclk starts low and ends low
            EXPORT  spi32
        mov     r1,#32
        ldr     r2,=(GPIOE_BASE+GPIO_BSRR)
        push    {r4}
spi32_1 tst     r0,#0x80000000
        ldreq   r3,=GPIO_BSRR_BS_15
        streq   r3,[r2]
        ldrne   r3,=GPIO_BSRR_BS_15
        strne   r3, [r2]
        ldr     r3,=GPIO_BSRR_BS_13
        ldr     r4,=GPIO_BSRR_BS_13
        str     r3,[r2]
        str     r4,[r2]
        lsl     r0,#1
        subs    r1,#1
        bne     spi32_1
        pop     {r4}
        bx      lr
        ENDP


        ALIGN

        END
```