Pre Revisited-Zeynepnur Şahinel

Part 1

```
;****************************
;delay.s
;****************************
five_sec_delay EQU 26666665 ; 5*16M/(3) 5sec delay
;LABEL DIRECTIVE VALUE COMMENT
                        AREA delay_SR , CODE, READONLY
                        THUMB
                        EXPORT delay
delay PROC
                PUSH{R0}

                LDR R0,=five_sec_delay

loop        SUBS R0,#1
                BNE loop
                NOP

                POP{R0}

                BX LR
                ENDP

                ALIGN
                END
```

```
;************************************************
; InitSysTick.s
;************************************************
NVIC_ST_CTRL EQU 0xE000E010
NVIC_ST_RELOAD EQU 0xE000E014
NVIC_ST_CURRENT EQU 0xE000E018
SHP_SYSPRI3 EQU 0xE000ED20
; 0x7D0 = 2000 -> 2000*250 ns = 500mus

RELOAD_VALUE_ADDR EQU 0x20000400
;LABEL DIRECTIVE VALUE COMMENT
                        AREA init_isr , CODE, READONLY, ALIGN=2
                        THUMB
                        EXPORT InitSysTick
InitSysTick PROC
                LDR R1 ,=NVIC_ST_CTRL
                MOV R0,#0
                STR R0,[R1]

                LDR R1,=NVIC_ST_CURRENT
                STR R0,[R1]
```

```
                        LDR R1,=SHP_SYSPRI3
                        MOV R0,#0x40000000
                        STR R0,[R1]

                        LDR R1,=NVIC_ST_CTRL
                        MOV R0,#0x03
                        STR R0,[R1]

                        BX LR
                        ENDP
ALIGN
END
```

```
. *******************************************************
;
;My_ST_ISR.s
GPIO_PORTB_DATA_WRITE EQU 0x400053C0 ; data address to write pins B4-B7
0011_1100_0000
.*******************************************************
;
; Register R8 is used for motor drive
; SysTick ISR area
.*******************************************************
;
;LABEL DIRECTIVE VALUE COMMENT
                                AREA my_st_isr , CODE, READONLY, ALIGN=2
                                THUMB
                                EXPORT My_ST_ISR

My_ST_ISR PROC

                        CMP R9,#0x01
                        BEQ ccw
                        BNE is_cw

ccw             LDR R1,=GPIO_PORTB_DATA_WRITE
                        CMP R8,#8
                        MOVEQ R8,#128
                        STR R8,[R1] ;write 1000_0000 to B7B6B5B4_0000 IN1: HIGH
                        LSR R8,#1
                        BX LR

is_cw           CMP R9,#0x02
                        BEQ cw
                        BX LR

cw                      LDR R1,=GPIO_PORTB_DATA_WRITE
                        CMP R8,#256
                        MOVEQ R8,#16
                        STR R8,[R1] ;write 1000_0000 to B7B6B5B4_0000 IN1: HIGH
                        LSL R8,#1
                        BX LR
```

```
                ENDP

                ALIGN
                END
```

```
;*****************************************************************
;PortBInit.s
GPIO_PORTB_DIR EQU 0x40005400
GPIO_PORTB_AFSEL EQU 0x40005420
GPIO_PORTB_PUR_R EQU 0x40005510 ; PUR actual address
GPIO_PORTB_DEN EQU 0x4000551C
GPIO_PORTB_AMSEL EQU 0x40005528
SYSCTL_RCGCGPIO EQU 0x400FE608
PUB EQU 0x0F
IOB EQU 0xF0
;*****************************************************************
; Program section
;*****************************************************************
;LABEL DIRECTIVE VALUE COMMENT
                        AREA port_b_init , READONLY, CODE
                        THUMB
                        EXPORT PortBInit ; Reference external subroutine

PortBInit PROC ;B0-B3 input, B4-B7 output

        LDR R1 ,=SYSCTL_RCGCGPIO
        LDR R0,[R1]
        ORR R0,R0,#0xFF
        STR R0,[R1]
        NOP
        NOP
        NOP

        LDR R1,=GPIO_PORTB_AMSEL
        LDR R0,[R1]
        BIC R0,#0xFF
        STR R0,[R1]

        LDR R1,=GPIO_PORTB_DIR
        LDR R0,[R1]
        BIC R0,#0xFF
        ORR R0,#IOB
        STR R0,[R1]

        LDR R1,=GPIO_PORTB_AFSEL
        LDR R0,[R1]
        BIC R0,#0xFF
        STR R0,[R1]
```

```
                LDR R1,=GPIO_PORTB_DEN
                LDR R0,[R1]
                ORR R0,#0xFF
                STR R0,[R1]

                LDR R1,=GPIO_PORTB_PUR_R
                MOV R0,#PUB
                STR R0,[R1]

                BX LR
                ENDP

                ALIGN
                END
```

```
;*****************************************************************
; Program_Directives.s
RELOAD_VALUE_ADDR EQU 0xE000E014
;*****************************************************************
; Program section
;*****************************************************************
;LABEL DIRECTIVE VALUE COMMENT
                        AREA main, READONLY, CODE
                        THUMB
                        EXTERN PortBInit
                        EXTERN InitSysTick
                        EXTERN delay
                        EXPORT __main ; Make available
__main PROC

                BL PortBInit

                LDR R0,=RELOAD_VALUE_ADDR

                MOV R1,#8000

                STR R1,[R0]

                BL InitSysTick ; initialize system timer

                CPSIE I ; enable interrupts

                BIC R8,#0xFF

                MOV R8,#128;

start   MOV R9,#0x01

                BL delay

                MOV R9,#0x02
```

```
        BL delay

        B start

        ENDP

        ALIGN
        END
```

Part 2.

```
;*****************************************************************
; Program_Directives.s
; Two buttons from 4*4 Keypad
RELOAD_VALUE_ADDR EQU 0xE000E014
GPIO_PORTB_DATA_READ EQU 0x4000503C ; data address to read pins B0-B3
0000_0011_1100
GPIO_PORTB_DATA EQU 0x400053FC ; data address to all pins
;*****************************************************************
; Program section
;*****************************************************************
;LABEL DIRECTIVE VALUE COMMENT
                AREA main, READONLY, CODE
                THUMB
                EXTERN PortBInit
                EXTERN InitSysTick
                EXTERN delay

EXPORT __main ; Make available
__main PROC
BL PortBInit
LDR R0,=RELOAD_VALUE_ADDR
MOV R1,#8000
STR R1,[R0]
BL InitSysTick ; initialize system timer
CPSIE I ; enable interrupts
BIC R8,#0xFF
MOV R8,#128;

start_line1 LDR R1,=GPIO_PORTB_DATA
LDR R0,[R1]
BIC R0,#0xFF
MOV R0,#0x70;
STR R0,[R1] ;write 0111_0000 to B7B6B5B4_0000 B7 line is activated

loop1 LDR R1,=GPIO_PORTB_DATA_READ
BIC R2,#0xFF
LDR R2,[R1]
PUSH {R2,R1}
BL delay
POP {R2,R1}
```

```
BIC R3,#0xFF
LDR R3,[R1]
CMP R3,R2 ;R3 = R2 = read_data = 0000_B3B2B1B0
BNE loop1 ;debouncing
CMP R3,#0x07
MOVEQ R4,#1 ;button 1 is pressed
BEQ button1
CMP R3,#0x0B
MOVEQ R4,#2 ;button 2 is pressed
BEQ button2
B start_line1

button1
loop5 LDR R1,=GPIO_PORTB_DATA_READ
BIC R2,#0xFF
LDR R2,[R1]
CMP R2,#0x0F
BNE loop5 ;debouncing
PUSH {R2,R1}
BL delay
POP {R2,R1}
BIC R3,#0xFF
LDR R3,[R1]
CMP R3,R2 ;R3 = R2 = read_data = 0000_B3B2B1B0
BNE loop5 ;debouncing
MOV R9,#1
B start_line1

button2
loop6 LDR R1,=GPIO_PORTB_DATA_READ
BIC R2,#0xFF
LDR R2,[R1]
CMP R2,#0x0F
BNE loop6 ;debouncing
PUSH {R2,R1}
BL delay
POP {R2,R1}
BIC R3,#0xFF
LDR R3,[R1]
CMP R3,R2 ;R3 = R2 = read_data = 0000_B3B2B1B0
BNE loop5 ;debouncing
MOV R9,#2
B start_line1

ENDP
ALIGN
END
```

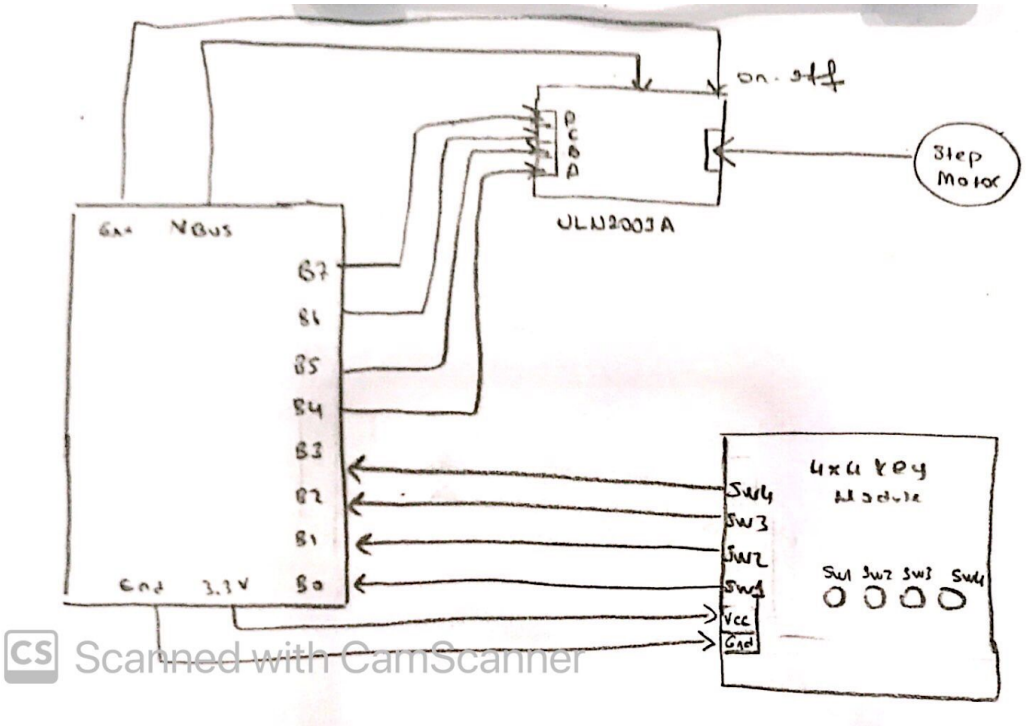
Part3

```
;*************************************************************
```

```
; Program_Directives.s
; Two buttons from single buttons, not 4*4 keypad
RELOAD_VALUE_ADDR EQU 0xE000E014
GPIO_PORTB_DATA_READ EQU 0x4000503C ; data address to read pins B0-B3
0000_0011_1100
;*****************************************************************
; Program section
;*****************************************************************
;LABEL DIRECTIVE VALUE COMMENT
AREA main, READONLY, CODE
THUMB
EXTERN PortBInit
EXTERN InitSysTick
EXTERN delay
EXPORT __main ; Make available
__main PROC
BL PortBInit
LDR R0,=RELOAD_VALUE_ADDR
MOV R1,#8000
STR R1,[R0]
BL InitSysTick ; initialize system timer
CPSIE I ; enable interrupts
BIC R8,#0xFF
MOV R8,#128;
debounce LDR R1,=GPIO_PORTB_DATA_READ
BIC R2,#0xFF
LDR R2,[R1]
BL delay
BIC R3,#0xFF
LDR R3,[R1]
CMP R2,R3 ;R3 = R2 = read_data = 0000_B3B2B1B0
BNE debounce
B release_check
release_check
debounce2 LDR R1,=GPIO_PORTB_DATA_READ
BIC R4,#0xFF
LDR R4,[R1]
BL delay
BIC R5,#0xFF
LDR R5,[R1]
CMP R4,R5 ;R4 = R5 = read_data = 0000_B3B2B1B0
BNE debounce2
CMP R2,R4; R2 x= R4 = read_data = 0000_B3B2B1B0
BNE released
BEQ release_check
released ;ccw if R9==1, cw if R9==2; 1110 for 1; 1101 for 2
;MOV R4,#8000
LDR R0,=RELOAD_VALUE_ADDR
LDR R1,[R0]
CMP R3,#14
MOVEQ R9,#0x01
BEQ debounce
CMP R3,#13
MOVEQ R9,#0x02
```

```
BEQ debounce
B debounce
ENDP
ALIGN
END
```

Part4



Part5

```
;*****************************************************************
; Program_Directives.s
;LABEL DIRECTIVE VALUE COMMENT
OFFSET EQU 0x10
FIRST EQU 0x20000400
GPIO_PORTB_DATA_READ EQU 0x4000503C ; data address to read pins B0-B3
0000_0011_1100
RELOAD_VALUE_ADDR EQU 0xE000E014
;LABEL DIRECTIVE VALUE COMMENT
                                    AREA main, READONLY, CODE
                                    THUMB
                                    EXTERN PortBInit
                                    EXTERN InitSysTick
                                    EXTERN delay
                                    EXPORT __main ; Make available
__main PROC
start   BL PortBInit
                CPSIE I ; enable interrupts
```

```
                LDR R0,=RELOAD_VALUE_ADDR
                MOV R1,#8000
                STR R1,[R0]
                BL InitSysTick ; initialize system timer
                BIC R8,#0xFF
                MOV R8,#128;

debounce LDR R1,=GPIO_PORTB_DATA_READ
                BIC R2,#0xFF
                LDR R2,[R1]
                BL delay
                BIC R3,#0xFF
                LDR R3,[R1]
                CMP R2,R3 ;R3 = R2 = read_data = 0000_B3B2B1B0
                BNE debounce
                B release_check

release_check

debounce2 LDR R1,=GPIO_PORTB_DATA_READ
                  BIC R4,#0xFF
                  LDR R4,[R1]
                  BL delay
                  BIC R5,#0xFF
                  LDR R5,[R1]
                  CMP R4,R5 ;R4 = R5 = read_data = 0000_B3B2B1B0
                  BNE debounce2

                  CMP R2,R4; R2 x= R4 = read_data = 0000_B3B2B1B0
                  BNE released
                  BEQ release_check

released ;ccw if R9==1, cw if R9==2; 1110 for 1; 1101 for 2
                LDR R0,=RELOAD_VALUE_ADDR
                LDR R1,[R0]

                CMP R3,#14
                MOVEQ R9,#0x01
                BEQ debounce

                CMP R3,#13
                MOVEQ R9,#0x02
                BEQ debounce

                CMP R3,#11 ;yavaslama
                BEQ yavaslama


                CMP R3,#7 ;hizlanma
                BEQ hizlanma
                B debounce

yavaslama ADD R1,#2000
                  STR R1,[R0]
```

```
                B debounce

hizlanma CMP R1,#8000
                SUB R1,#2000
                MOVEQ R1,#8000
                STR R1,[R0]
                B debounce

               ENDP
               ALIGN
               END
```