

İTÜ



Department of Computer Engineering

BLG 351E Microcomputer Laboratory Experiment Report

Experiment No : 6
Experiment Date : 02.12.2016

Group Number : Friday - 3
Group Members :

ID	Name	Surname
150130109	Güllü	Katık
150130136	Mehmet Barış	Yaman
150140142	Gamze	Akyol

Laboratory Assistant : Mustafa Ersen

1 INTRODUCTION

In this experiment, a chronometer is designed with centisecond precision and displayed through 7-segment display. In part 1, numbers 0-3 are demonstrated in 7-segment display side by side via an infinite loop. In part 2, the chronometer is designed.

2 EXPERIMENT

2.1 PART 1

In this part of the experiment we showed four digits on 7-segment display. It is not possible to show four digits at the same time, so we showed digits by turns. We observed that all the digits lit at the same time because human eye cannot detect rapid changes.

The code of this part given below:

```

;-----
; Main loop here
;-----

Setup      mov     #array2, R7           ;R7 points first element of array2
           mov     #array, R8           ;R8 points first element of array
           bis.b   #11111111b, &P1DIR   ;initializing P1 as output
           bis.b   #11111111b, &P2DIR   ;initializing 4 bit of P2 as output

reset      mov     #array2, R7           ;R7 initialize again
           mov     #array, R8           ;R8 initialize again
           ;0, 1, 2 and 3 are shown on display by turns

mainLoop   mov.b   #00000000b, &P2OUT
           mov.b   @R8, &P1OUT
           mov.b   @R7, &P2OUT
           inc     R7
           inc     R8
           cmp.w   #lastElement2, R7
           jne     mainLoop
           jmp     reset                ;when R7 equals to last element of array2, jump reset

;array that contains all digits
array      .byte   00111111b, 00000110b, 01011011b, 01001111b, 01100110b, 01101101b, 01111101b, 00000111b, 01111111b, 01100111b
lastElement
;array that used for select the digit of 7-segment display
array2     .byte   00000001b, 000000010b, 00000100b, 00001000b
lastElement2

;-----
; Stack Pointer definition
;-----
           .global __STACK_END
           .sect   .stack

;-----
; Interrupt Vectors
;-----
           .sect   ".reset"             ; MSP430 RESET Vector
           .short  RESET

```

2.2 PART 2

In the second part of the experiment, a chronometer is designed with a button which stops the chronometer whenever it is pushed. To do this, interrupt mechanism is used. In addition, given timer interrupt subroutine is used to count the centiseconds and convert them to second in every 100 loops. Also, the given BCD conversion subroutine is used to convert numbers to 7-segment display numbers.

The code of this part given below:

BLG 351E Microcomputer Laboratory – Experiment Report

```
-----
; Main loop here
-----
Initialvalues    bis.b #040h,&P2IE ;for interrupt P2.6
                  and.b #0BFh,&P2SEL ;setting values for P2SEL
                  and.b #0BFh,&P2SEL2
                  bis.b #040h,&P2IES ;high-low interrupting
                  clr &P2IFG ;clear interrupt flag

Setup            eint
                  bis.b #0FFh,&P1DIR ;initializing P1 as output
                  bis.b #00Fh,&P2DIR ;initializing 4 bit of P2 as output
                  mov.w #0000001000010010b,&TA0CTL
                  mov.w #0010011100010000b,&TA0CCR0
                  mov.w #0000000000010000b,&TA0CCTL0
                  mov #array,R8
                  mov.b #000h,R5 ;centiseconds reset
                  mov.b #000h,R6 ;seconds reset
                  mov.b #000h,R7 ;R7 is used for stop/start/continue the timer

mainLoop         push R5
                  call #bcd
                  pop R11 ;units digit of centisecond
                  pop R12 ;tens digit of centisecond
                  push R6
                  call #bcd
                  pop R9 ;units digit of second
                  pop R10 ;tens digit of second
                  ;units and tens digits of centisecond and second are shown on 7-segment display by turns
                  add.w R8,R11
                  mov.b @R11,&P1OUT
                  mov.b #008h,&P2OUT
                  mov.b #000h,&P2OUT
                  add.w R8,R12
                  mov.b @R12,&P1OUT
                  mov.b #004h,&P2OUT
                  mov.b #000h,&P2OUT
                  add.w R8,R9
                  mov.b @R9,&P1OUT
                  mov.b #002h,&P2OUT
                  mov.b #000h,&P2OUT

                  add.w R8,R10
                  mov.b @R10,&P1OUT
                  mov.b #001h,&P2OUT
                  mov.b #000h,&P2OUT
                  jmp mainLoop ;jump mainLoop

;subroutine for convert seconds or centiseconds to BCD form
bcd              pop R4 ;R4 keeps return address
                  pop R15
                  mov.b #000h,R14 ;initially tens digit is 0
new_mod          cmp #00Ah,R15 ;if R15 smaller than 10
                  jl return ;jump return
                  sub #00Ah,R15 ;else R15=R15-10
                  inc R14 ;increment tens digit
                  jmp new_mod ;jump new_mod
return           push R14
                  push R15
                  push R4
                  ret

;interrupt subroutines
ISR              dint
                  xor.b #001h,R7
                  cmp #001h,R7
                  jeq stop
                  mov.w #0000001000010010b,&TA0CTL

clear            clr &P2IFG
                  eint
                  reti

stop             mov.w #00000010000000010b,&TA0CTL
                  jmp clear

TISR             dint
                  inc R5 ;increment centisecond
                  cmp #064h,R5 ;
                  jnz return2 ;jump return2
                  inc R6 ;if centiseconds equal to 100, increment seconds
                  mov.b #000h,R5 ;centiseconds reset
                  mov.b #080h,&P1OUT
```

BLG 351E Microcomputer Laboratory – Experiment Report

```
return2                eint                bic #0001h,&TA0CTL
                                bic #0001h,&TA0CCTL0
                                reti

array .byte 00111111b,00000110b,01011011b,01001111b,01100110b,01101101b,01111101b,00000111b,01111111b,01101111b
lastElement

.data
seconds .byte 00h
centiseconds .byte 00h
;-----
; Stack Pointer definition
;-----
        .global __STACK_END
        .sect   .stack

;-----
; Interrupt Vectors
;-----
        .sect   ".reset"                ; MSP430 RESET Vector
        .short  RESET

        .sect   ".int09"
        .short  TISR
        .sect   ".int03"
        .short  ISR
```

3 CONCLUSION

As a result, usage of 7-segment display was understood deeply. In this experiment, timer of processor was used and firstly we had a difficulty to understand its mechanism, but then we managed to solve this.