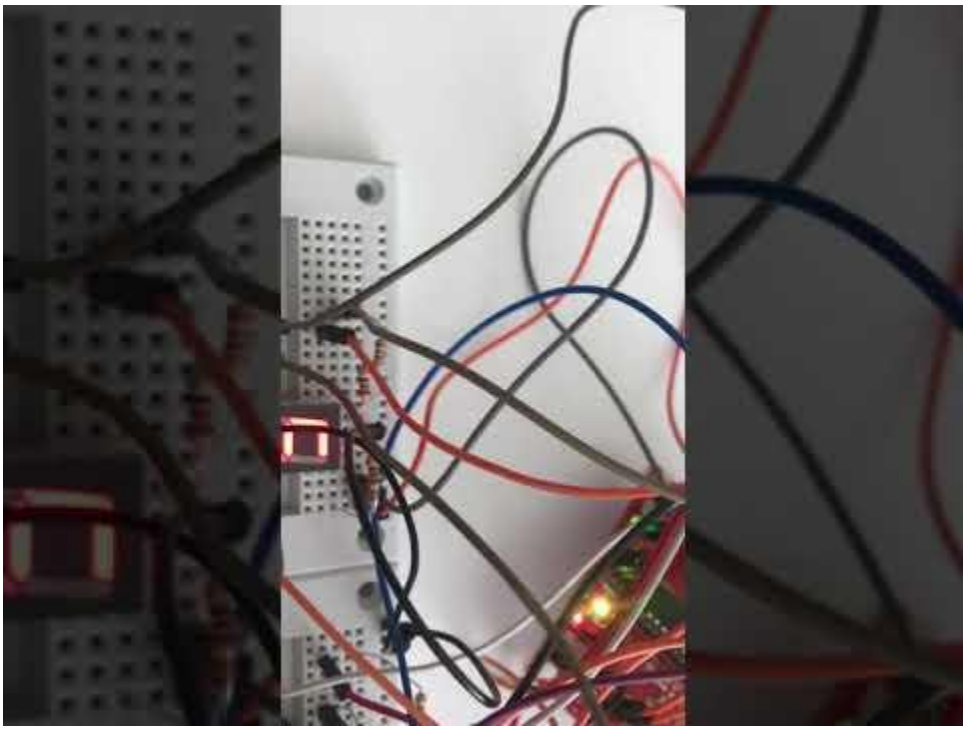
CENG 329 – Microprocessors

Project 2

Implement a two-digit counter with two 7-segment displays. Please click the below picture to watch a sample.



For submissions, upload your video to YouTube, preferably by selecting ‘Unlisted’ rather than ‘Public’. Don’t forget to record yourself about two or three seconds since we need to approve your ownership. Show your student ID card, too. Append your link to the beginning of your source code as a comment. Write also your team info (ids, names) to the start of the file.

Code for a demo version (one of the displays shows 0 if the number in other one is even, 1 otherwise) :

```
;-------------------------------------------------------------------------------
; MSP430 Assembler Code Template for use with TI Code Composer Studio
;
;
;-------------------------------------------------------------------------------
            .cdecls C,LIST,"msp430.h"       ; Include device header file

;-------------------------------------------------------------------------------
            .def    RESET                   ; Export program entry-point to
                                            ; make it known to linker.
                    .data
segm                .byte  0x7e, 0x30, 0x6d, 0x79
                    .byte  0x33, 0x5b, 0x5f, 0x70
                    .byte  0x7f, 0x73, 0x77, 0x1f
                    .byte  0x4e, 0x3d, 0x4f, 0x47

;-------------------------------------------------------------------------------
            .text                           ; Assemble into program memory.
            .retain                         ; Override ELF conditional linking
                                            ; and retain current section.
            .retainrefs                     ; And retain any sections that have
                                            ; references to current section.

;-------------------------------------------------------------------------------
RESET       mov.w   #__STACK_END,SP         ; Initialize stackpointer
StopWDT     mov.w   #WDTPW|WDTHOLD,&WDTCTL  ; Stop watchdog timer


;-------------------------------------------------------------------------------
; Main loop here
;-------------------------------------------------------------------------------
                    bic.b  #01111111b, &P1SEL
                    bic.b  #01111111b, &P1SEL2

                    bis.b  #01111111b, &P1DIR
                    bic.b  #01111111b, &P1OUT

                    bic.b  #01111111b, &P2SEL
                    bic.b  #01111111b, &P2SEL2

                    bis.b  #01111111b, &P2DIR
                    bic.b  #01111111b, &P2OUT


                    mov.w  #0, r5


inc_val:
                    mov.b r5, r7
                    and.b #1, r7
                    jnz show
```

```
                    jmp light
show:
                    jmp light

light:
                    mov.b segm(r5), r11
                    mov.b r11, &P1OUT
                    mov.b segm(r7), r11
                    mov.b r11, &P2OUT



                    call #delay
                    inc r5
                    cmp #16, r5
                    jne cont
                    mov #0, r5
cont:
                    jmp inc_val



delay               push   r6
                    mov.w  #0, r6
dloop               add.w  #1, r6
                    push   r9
                    push   r9
                    pop           r9
                    pop           r9
                    jne           dloop
                    pop           r6
                    ret



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

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

Here is the video for demo: