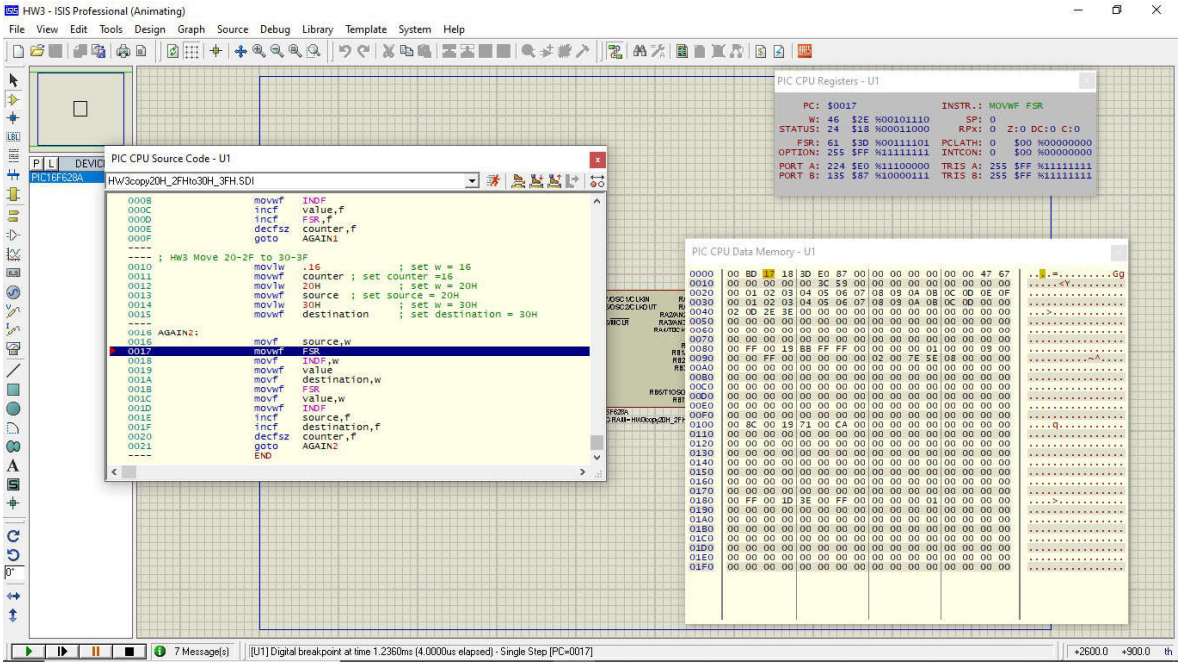
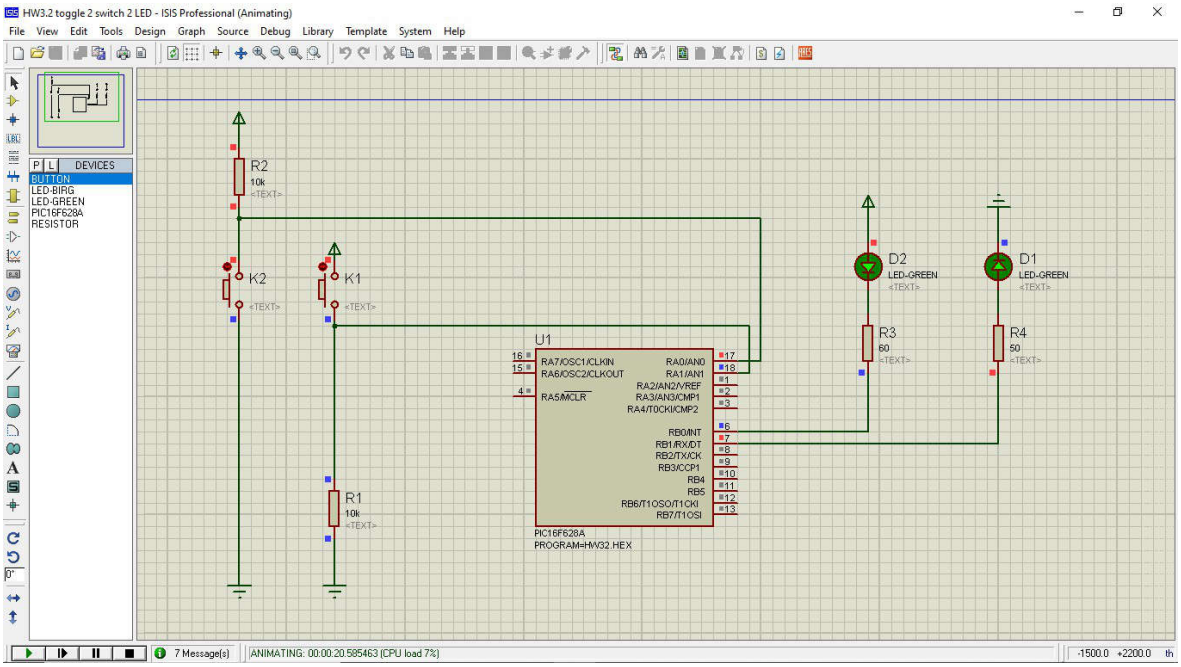



```
;*************************************************************
;**** Program title: HW3.1 COPY 20H-2FH to 30H-3FH ***********
;**** Programmer: SUPPAKORN HENGPRASITH 5913370    ***********
;*************************************************************
            PROCESSOR PIC16F628
            #include <P16F628.INC>
            __CONFIG    _CP_OFF & _MCLRE_ON & _INTRC_OSC_NOCLKOUT &
_LVP_OFF & _WDT_OFF

;Declare File register
counter     EQU   40H
value EQU   41H
source      EQU 42H
destination EQU   43H
            ORG   0x00  ; reset vector
            GOTO START  ; jump to start of the program

            ORG   0x04  ; Interrupt vector

START:
; SET 20-2F to 0-F
            movlw 20H
            movwf FSR
            clrw                ; clrw = movlw 0H
            movwf value ; set value = w = 0
            movlw .16           ; set w = 16
            movwf counter       ; set counter =16

AGAIN1:  movf       value,w
            movwf INDF
            incf  value,f
            incf  FSR,f
            decfsz      counter,f
            goto  AGAIN1

; HW3 Move 20-2F to 30-3F
            movlw .16           ; set w = 16
            movwf counter       ; set counter =16
```

```
                movlw 20H          ; set w = 20H
                movwf source       ; set source = 20H
                movlw       30H          ; set w = 30H
                movwf       destination ; set destination = 30H

AGAIN2:
                movf  source,w
                movwf FSR
                movf  INDF,w
                movwf value
                movf  destination,w
                movwf FSR
                movf  value,w
                movwf INDF
                incf  source,f
                incf  destination,f
                decfsz      counter,f
                goto  AGAIN2
                END
```




```
;*****************************************
;HW32 2switch2led toggle
;*****************************************
            PROCESSOR PIC16F628
            #include <P16F628.INC>
            __CONFIG    _CP_OFF & _MCLRE_ON & _INTRC_OSC_NOCLKOUT &
_LVP_OFF & _WDT_OFF

;Declare File register
SW1_STATE   EQU   40H
SW2_STATE   EQU 41H
            ORG   0x00  ; reset vector
            GOTO START  ; jump to start of the program

            ORG   0x04  ; Interrupt vector


START:      ; Port Configuration
            MOVLW .7
            MOVWF CMCON       ; CMCON=7 turn-off analog comparator inputs
            BCF         STATUS,RP1  ; or 'BCF 0x03,0x06'
            BSF         STATUS,RP0  ; Select Bank1
            BSF         TRISA,0           ; Port RA0 is an Input pin
            BSF         TRISA,1           ; Port RA1 is an Input pin
            BCF         TRISB,0           ; Port RB0 is an output pin
            BCF         TRISB,1           ; Port RB1 is an output pin
            BCF         STATUS,RP0  ; Back to Bank0

            ; Start your program here
            ; K1 ACTIVE HIGH K2 ACTIVE LOW
            ; LED1 ACTIVE HIGH LED2 ACTIVE LOW

            CLRW
            MOVWF SW1_STATE   ;set SW1_STATE=0
            MOVWF SW2_STATE   ;set SW2_STATE=0
Inf_loop:
            BTFSC PORTA,1           ; K1 ACTIVE HIGH use BTFSC
            GOTO  TOGGLE_SW1  ;
```

```
RETURN_TOGGLE_SW1:
            BTFSS       PORTA,0           ; K2 ACTIVE LOW use BTFSS
            GOTO  TOGGLE_SW2  ;
RETURN_TOGGLE_SW2:
            BTFSC SW1_STATE,0
            GOTO  LED1_ON
            GOTO  LED1_OFF
RETURN_LED1_ONOFF:
            BTFSC SW2_STATE,0
            GOTO  LED2_ON
            GOTO  LED2_OFF

LED1_ON:
            BSF         PORTB,1           ; LED1 ACTIVE HIGH >> on use BSF
            GOTO  RETURN_LED1_ONOFF

LED1_OFF:
            BCF         PORTB,1           ; LED1 ACTIVE HIGH >> off use BCF
            GOTO  RETURN_LED1_ONOFF

LED2_ON:
            BCF         PORTB,0           ; LED2 ACTIVE LOW >> on use BCF
            GOTO  Inf_loop


LED2_OFF:
            BSF         PORTB,0           ; LED2 ACTIVE LOW >> off use BSF
            GOTO  Inf_loop

TOGGLE_SW1:
            MOVLW .1
            XORWF SW1_STATE,f ; TOGGLE SW1_STATE
            GOTO  RETURN_TOGGLE_SW1

TOGGLE_SW2:
            MOVLW       .1
            XORWF SW2_STATE,f ; TOGGLE SW2_STATE
            GOTO  RETURN_TOGGLE_SW2

            END
```