```
;-------------------------------------------------------------------------------
; 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.
;-------------------------------------------------------------------------------
            .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
;-------------------------------------------------------------------
```

```
------------

; Configure RED LED, LED1 (Port 1, Pin 0)
            bis.b   #BIT0, &P1OUT
            bis.b   #BIT0, &P1DIR           ; Set P1.0 as output pin

; Configure Green LED, LED2 (Port 9, Pin 7)
            bis.b   #BIT7, &P9OUT
            bis.b   #BIT7, &P9DIR           ; Set P9.7 as output pin

; Configure Push Button S1 (Port 1, Pin 1)
            bis.b   #BIT1,  &P1REN          ; Enable Resistor
            bis.b   #BIT1,  &P1OUT          ; Pull-up resistor, since S1 is active low
            bis.b   #BIT1,  &P1IE           ; Enable P1.1 interrupts
            bis.b   #BIT1,  &P1IES          ; Enable Loweing Edge

; Configure Push Button S2 (Port 1, Pin 2)
            bis.b   #BIT2,  &P1REN          ; Enable Resistor
            bis.b   #BIT2,  &P1OUT          ; Pull-up resistor, since S2 is active low
            bis.b   #BIT2,  &P1IE           ; Enable P1.2 interrupts
            bis.b   #BIT2,  &P1IES          ; Enable Loweing Edge

            bic.w   #LOCKLPM5,&PM5CTL0      ; Disable the GPIO power-on default
                                            ; high-impedance mode

            nop
            bis.w  #GIE|LPM3,     SR        ; sleep and wait for interrupts
            nop

; ------------------------------   ISR ----------------------------------------

PORT1_ISR:

            bit.b   #BIT1,  &P1IFG
            jnc IF1_EXIT
```

```
                xor.b     #BIT0,         &P1OUT     ; Toggle P1.0
                ;bic.b     #BIT1, &P1IFG
IF1_EXIT:

            bit.b   #BIT2,  &P1IFG
            jnc IF2_EXIT
                xor.b     #BIT7,         &P9OUT     ; Toggle P9.7
                ;bic.b     #BIT2, &P1IFG
IF2_EXIT:

            ;clr.b    &P1IFG                 ; multi-sourced, therefore clear IF (Interrupt Flag)
            bic.b     #BIT1, &P1IFG
            bic.b     #BIT2, &P1IFG
            reti

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

;-------------------------------------------------------------------------------
; Interrupt Vectors
;-------------------------------------------------------------------------------
; Connect PORT1 interrupts to Interrupt Service Routine
            .sect   ".int37"                ; PORT1 Vector
            .short  PORT1_ISR

            .sect   ".reset"                ; MSP430 RESET Vector
            .short  RESET


```