;-----------------------------------------------
; MSP430 Assembler Code Template for use with TI Code Composer Studio
;
;-----------------------------------------------
.cdecls C,LIST, "msp430.h" ; Include device header file
.data
result1: .space 2
result2: .space 2
.text ; Assemble into program memory
.retain ; Override ELF conditional linking
; and retain current section
.retainrefs ; Additionally retain any sections
; that have references to current
; section
RESET mov.w #_STACK_END,SP ; Initialize stack pointer
StopWDT mov.w #WDTPW|WDTHOLD,&WDTCTL ; Stop watchdog timer

;-----------------------------------------------
; Main loop here
mov.w #111000001000011b, R12 ; prepare subroutine input
call #OddFunction
mov.w R11, &result1

mov.w #111111111111111b, R12 ; prepare subroutine input
call #OddFunction
mov.w R11, &result2

loop: jmp loop

;-----------------------------------------------
; subroutine: OddFunction
; Tests if the number of set bits in a number are odd or even
; Input R12: modified
; Output R11: (R11=1 if number of set bits is odd, otherwise 0)
; Local Variable R10: modified, not preserved
;-----------------------------------------------

OddFunction:
call #NoOfOnes ; call subroutine
mov.w R11, R12 ; copy output of NoOfOnes to input of IsOdd
call #IsOdd
ret

;-----------------------------------------------
; subroutine: IsOdd
; Tests if a number is odd or even
; Input R12: not modified
; Output R11: R11 = 1 if number is odd, 0 otherwise
;-----------------------------------------------

IsOdd:
mov.w #0, R11
bit.w #0000000000000001b, R12
jnc Even ; bit not set
mov.w #1, R11 ; bit set
Even:
ret
Subroutine: NoOfOnes
Counts the number of ones in the binary representation of a number
Input R12: (not modified)
Output R11:
Local variable R10: modified, not preserved

NoOfOnes:
mov.w #0000000000000001b, R10 ; R10 holds bit for testing, start with the lsb
mov.w #0, R11 ; R11 holds sum of set bits in the number

MoreBits:
bite.w R10, R12 ; test the bit
jnc BitNotSet ; if bit is not set then go to label NotSet
ince.w R11 ; if you are here then bit is set, increase sum of bits

BitNotSet:
rla.w R10 ; if you are here then bit is not set
jnz MoreBits ; Go to label L1 if there are more bits to test
; loop is exited when R11 becomes zero and there are no more bits to test

ret

Stack Pointer definition

Interrupt Vectors

.MP430 RESET Vector
.reset