

Experiment 6

Name:Pratik Chavan

Div/Batch:A/A1 Roll No.07

.MODEL SMALL

.STACK 100H

.DATA

MSG DB 'Flag Register: \$' ; *Message to display*

HEX_CHARS DB '0123456789ABCDEF' ; *Lookup table for hex digits*

FLAGS DW ? ; *Variable to store flag register value*

.CODE

MAIN PROC

MOV AX, DGROUP

MOV DS, AX

PUSHF ; *Push flag register onto the stack*

POP FLAGS ; *Pop flag register into FLAGS variable*

MOV DX, OFFSET MSG

MOV AH, 09H

INT 21H ; *Print "Flag Register: "*

MOV AX, FLAGS ; *Load flag register value into AX*

CALL PRINT_HEX ; *Print the flag register in hexadecimal format*

MOV AH, 4CH

```

    INT 21H    ; Exit program
MAIN ENDP

; -----
; Print 16-bit Hex Procedure
; -----

PRINT_HEX PROC
    MOV CX, 4    ; We have 4 hex digits (16-bit / 4-bit each)
    MOV BX, 12   ; Bit shift amount (12, 8, 4, 0)

HEX_LOOP:
    MOV DX, AX   ; Copy AX value
    MOV CL, BL   ; Move shift count into CL (Fix for SHR error)
    SHR DX, CL   ; Shift right to isolate one hex digit
    AND DX, 0FH  ; Mask the lower 4 bits
    MOV SI, DX   ; Move index to SI
    MOV DL, [HEX_CHARS + SI] ; Convert to ASCII hex character
    MOV AH, 02H
    INT 21H     ; Print the hex digit
    SUB BX, 4    ; Move to the next hex digit
    LOOP HEX_LOOP ; Repeat until all digits are printed

    RET
PRINT_HEX ENDP

END MAIN

```

OUTPUT: