Keegan Vaz
SECMPN B
42
192120

To Display All Flag Registers

Aim: Write an assembly language program to display all flag registers

Prerequisite: TASM assembler

Theory:

To display the contents of flag register pushf and pop instruction. Each bit of flag register is then masked off with 1 and all 0's (i.e. 1000 0000 0000 0000(16 bit) à 8000h) and based on the result of masking either 0 (30h) or 1 (31h) is get displayed on the screen. Each bit of the above 16 bit number gets shifted in the right direction by 1 position before masking to obtain the next bit position of the flag register. This whole procedure gets repeated 16 times.

Algorithm to display all flag registers:

1. Start

2. Initialize data segment through AX register in the DS register.

3. Display the flag bit names as "X X X X O D I T SF ZF x AF X PF X CF "

4. Push the contents of flag register to a stack

5. Pop the contents of stack to register to any 16 bit register (say BX =0000 0100 1000 1001)

6. Move the contents of BX to temporary variable say t

7. Move the 8000h number to AX.(AXß 8000h)

8. Move the count as 16(in decimal) to CX register (as 16 bit flag register)

9. Move the contents of temporary variable t to BX.

10. And the contents of BX and AX.

11. If zero flag is set then goto step no 14 otherwise goto step no. 12

12. Move the 31h to DL register.

13. Make the unconditional jump to a step no. 15

14. Move the 30h to DL register.

15. Preserve the (8000h) number from AX in t1 temporary variable. (As while displaying a 30h or 31 h AH register gets modified as the 02h function is moved to INT 21h).

16. Display the contents of the DL register.

17. Move the contents of t1 to AX register back (As while displaying 30h or 31 h AH register get modified as 02h function is moved to INT 21h).

18. Rotate the contents of AX by 1 position in the right direction.

19. Repeat step no 5 to 17 till count CX reaches 0.

20. Stop.

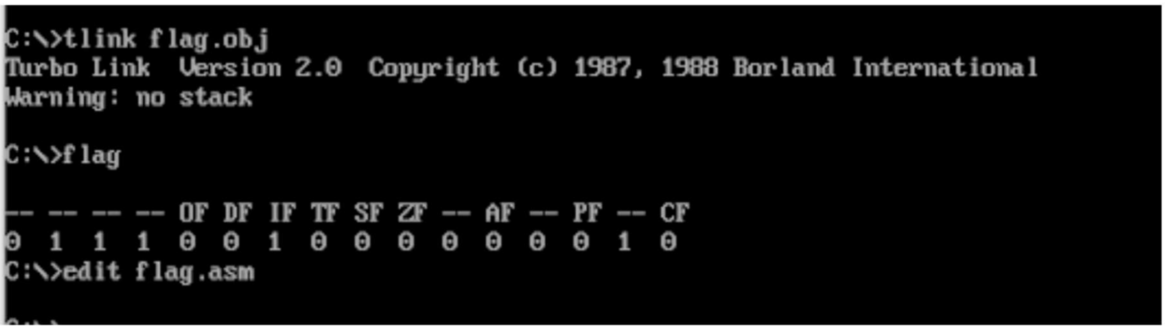
CODE:

```
 Data Segment
 msg db 0dh,0ah,"-- -- -- -- OF DF IF TF SF ZF -- AF -- PF -- CF $"
 newl db 0dh,0ah,"$"
 flag dw ?
Data ends
Code Segment
assume CS:Code,DS:Data
start:
 mov ax,Data
 mov DS,ax
 mov dx,offset msg
 mov ah,09h
 int 21h
 mov dx,offset newl
 mov ah,09h
 int 21h

 cli
 stc
 std
 pushf
 pop bx
 mov flag,bx

 mov cx,16
 mov bx,8000h

loops:
 mov ax,flag
 and ax,bx
 jz zero
 mov dl,31h
 mov ah,02h
 int 21h
 jmp space

zero: mov dl,30h
 mov ah,02h
 int 21h

space: mov dl,' '
 mov ah,02h
 int 21h

 mov ah,02h
 int 21h
 ror bx,1
 loop loops
 mov ah,4ch
 int 21h
Code ends
```

end

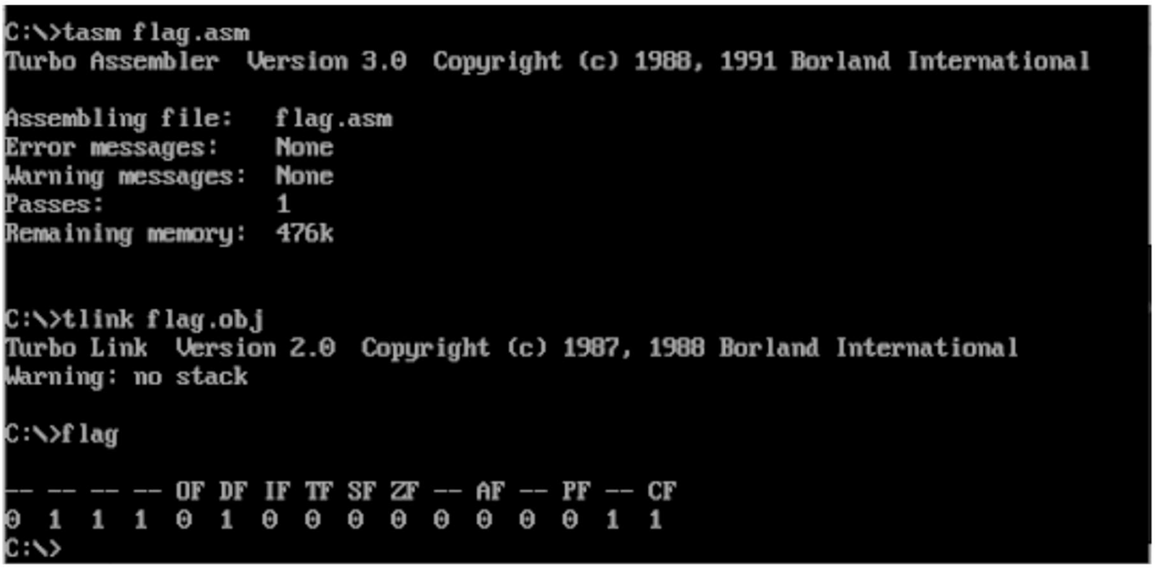
OUTPUT:

(Without using CLI, STD, STC)-



```
C:\>tlink flag.obj
Turbo Link  Version 2.0  Copyright (c) 1987, 1988 Borland International
Warning: no stack

C:\>flag

-- -- -- -- OF DF IF TF SF ZF -- AF -- PF -- CF
0  1  1  1  0  0  1  0  0  0  0  0  0  0  1  0
C:\>edit flag.asm
```

(Using CLI, STD, STC)-



```
C:\>tasm flag.asm
Turbo Assembler  Version 3.0  Copyright (c) 1988, 1991 Borland International

Assembling file:   flag.asm
Error messages:    None
Warning messages:  None
Passes:            1
Remaining memory:  476k


C:\>tlink flag.obj
Turbo Link  Version 2.0  Copyright (c) 1987, 1988 Borland International
Warning: no stack

C:\>flag

-- -- -- -- OF DF IF TF SF ZF -- AF -- PF -- CF
0  1  1  1  0  1  0  0  0  0  0  0  0  0  1  1
C:\>
```

start

Conclusion:

From the experiment above we are able to learn the two types of flags i.e the status flag and control flags. The status flag contains a carry flag, auxiliary flag, parity flag, sign flag, zero flag and overflow flag. The status flags are basically set by the ALU to show the status of execution of an instruction. The control flags are set by the programmer to control the execution. The control flags present are direction flag, trap flag and interrupt flag.