

# FLAG REGISTER OF 80386 MICROPROCESSOR

Presented by:

Name: Nikesh Tamang

Name: Sandesh Limbu

# INTRODUCTION

Flag Register of 80386:

- ❖ The Flag register of 80386 is a 32 bit register.
- ❖ Intel has reserved bits D18 to D31, D15, D5 and D3, while D1 is always set at 1. Two extra new flags are added to the 80286 flag to derive the flag register of 80386.
- ❖ They are VM and RF flags

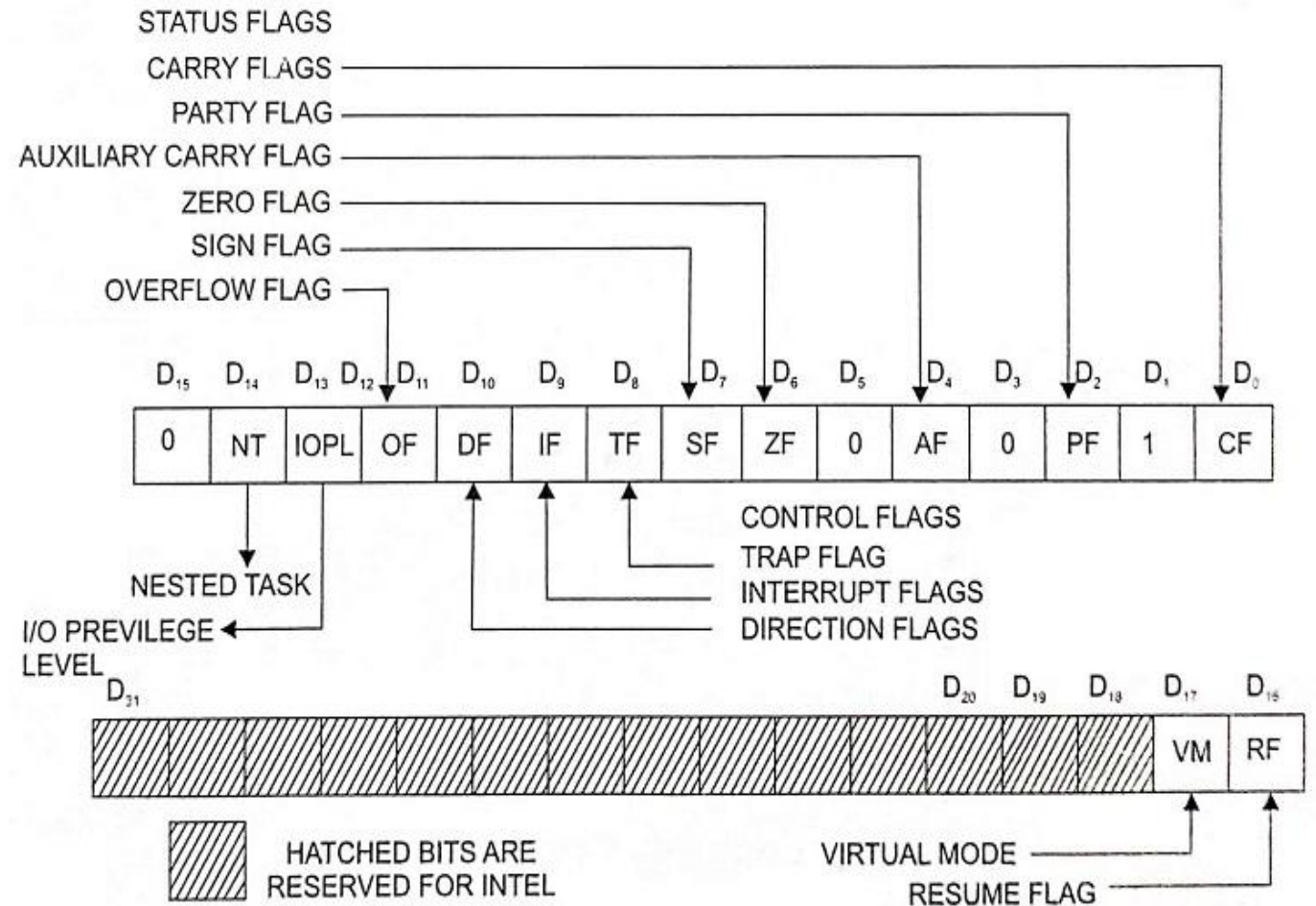


Fig. 11.35 FLAG register of the 80386

# Flag register

## 1. Status Flags

- I. CF (Carry Flag)
- II. PF (Parity Flag )
- III. AF (Auxiliary Carry Flag)
- IV. ZF (Zero Flag)
- V. SF (Sign Flag)
- VI. OF (Overflow Flag)

# FLAG REGISTER

## 2. Control Flags

- I. DF ( Direction flag)
- II. IF ( Interrupt Flag)
- III.TF (Trap Flag)

# FLAG REGISTER

## 3. System Flags

- I. VM (Virtual Memory) flag
- II. R (Resume) flag
- III. NT (Nested flag)
- IV. IOPL (I/O Privilege level)

# Status Flags

## I. SF (Sign Flag)

- SF is set if the high-order bit of the result is set, otherwise it is reset.
- For negative number , sign flag is set to 1 otherwise 0.

## II. CF (Carry Flag)

- CF is set if the operation resulted in a carry out of (addition), or a borrow into (subtraction) the high-order bit. Otherwise CF is reset.
- This bit can be set , cleared or inverted with the STC, CLC, or CMC instructions respectively.

## III. PF (Parity Flag )

- PF is set if the low-order eight bits of the operation contains an even number of ``1's" (even parity).
- PF is reset if the low-order eight bits have odd parity. PF is a function of only the low-order eight bits ,regardless of operand size.

# Status Flags

## IV. AF (Auxiliary Carry Flag)

- The Auxiliary Flag is used to simplify the addition and subtraction of packed BCD quantities.
- AF is set if the operation resulted in a carry out of bit 3 (addition) or a borrow into bit 3 (subtraction). Otherwise AF is reset.
- AF is affected by carry out of, or borrow into bit 3 only, regardless of overall operand length: 8, 16 or 32 bits.

## V. ZF (Zero Flag)

- ZF is set if all bits of the result are 0. Otherwise it is reset.

# Status Flags

## VI. OF (Overflow Flag)

- OF is set if the operation resulted in a signed overflow.
- Signed overflow occurs when the operation resulted in carry/borrow into the sign bit (high-order bit) of the result but did not result in a carry/borrow out of the high order bit, or vice-versa.
- For 8/16/32 bit operations, OF is set according to overflow at bit7/15/31, respectively.



# Control Flags

## I. DF (Direction flag):

- The direction flag controls the direction of string operations. When the D flag is cleared these operations process strings from low memory up to high memory.
- If the D flag is set, then SI and DI (Offset pointers) are decremented by 1 after each operation to process strings from high to low memory

## II. IF (Interrupt Flag):

- When the interrupt flag is set, the 80386 recognizes and handles external hardware interrupts on its INTR pin.
- If the interrupt flag is cleared, 80386 ignores any inputs on this pin.
- The IF flag is set and cleared with the STI and CLI instructions, respectively.

# Control Flag

- III. TF (Trap Flag):
  - When an 80386 detects that this flag is set, it executes one instruction and then automatically generates an internal exception 1.
  - After servicing the exception, the processor executes the next instruction and repeats the process.
  - This single stepping continues until the program code resets this flag for debugging programs' single-step facility is used.

# System Flags

## I. VM (Virtual Memory) flag:

- This flag indicates the operating mode of 80386.
- When the VM flag is set, 80386 switches from protected mode to virtual 8086 modes.

## II. R (Resume) flag:

- This flag, when set allows selective masking of some exceptions at the time of debugging.

## III. NT (Nested flag):

- This flag is set when one system task invokes another task.

# System Flags

## IV. IOPL (VO Privilege level):

- The two bits in the IOPL are used by the processor and the operating system to determine your application's access to I/O facilities.
- It holds a privilege level, from 0 to 3, at which the current code is running in order to execute any I/O-related instruction.

**THE END.**