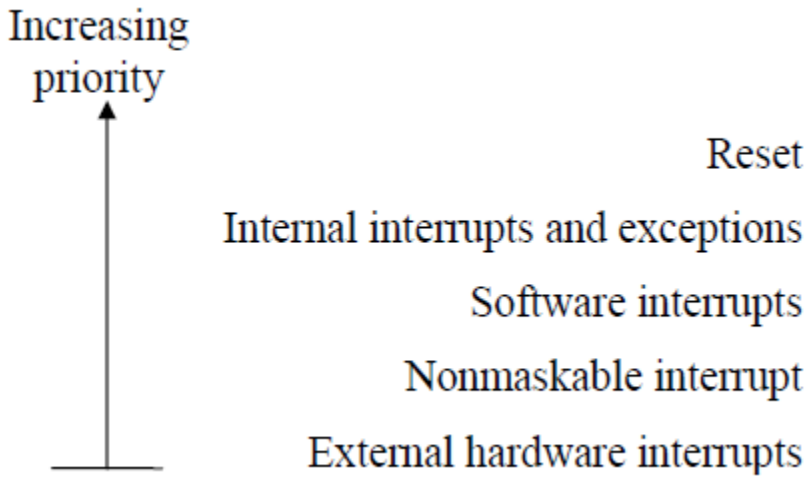
8086 Interrupt Types and Interface

Interrupt Mechanism Types, and Priority

Interrupts provide a mechanism for quickly changing program environment. Transfer of program control is initiated by the occurrence of either an event to the microprocessor or an event in its external hardware. The 8088 and 8086 microcomputers are capable of implementing any combination of up to 256 interrupts. As Fig. below shows, they are divided into five groups.



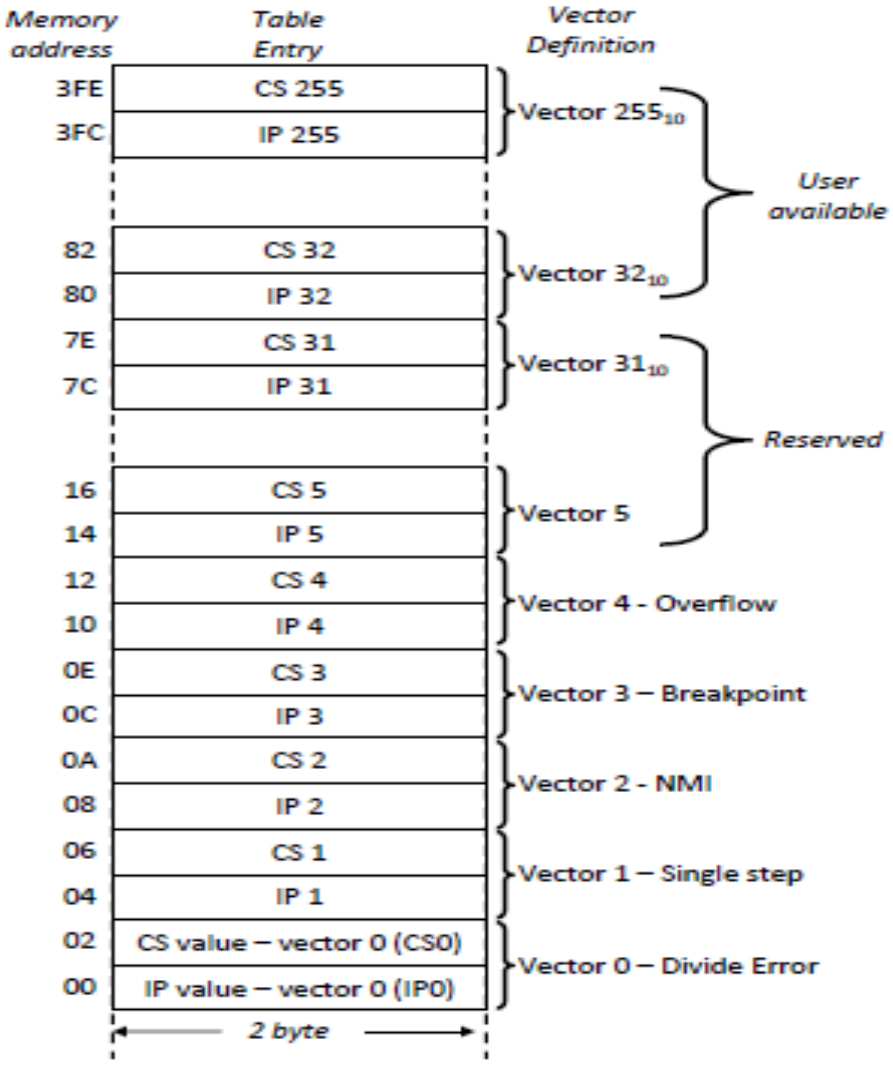
Types of interrupts and their priority

The user defines the function of the external hardware, software, and non maskable interrupt. For instance, hardware interrupts are often assigned to devices such as the keyboard, printer, and timers. On the other hand, the functions of the internal interrupts and reset are not user defined. They perform dedicated system functions. An example of a high-priority service routine that should not be interrupted is that for a power failure. Once initiated, this routine should be quickly run to completion to assure that the microcomputer goes through an orderly power-down. A keyboard should also be assigned to a high-priority interrupt. This will

assure that the keyboard buffer does not get full and lock out additional entries. On the other hand, devices such as the floppy disk or hard disk controller are typically assigned to a lower priority level.

Interrupt Vector Table

An address pointer table is used to link the interrupt type numbers to the locations of their service routines in the program-storage memory. Figure below shows a map of the pointer table in the memory of the 8086 microcomputer.



Interrupt vector table of the 8086.

Looking at this table, we see that it contains 256 address pointers (vectors). Which are identified as vector 0 through vector 255. That is, one pointer corresponds to each of the interrupt types 0 through 255. These address pointers identify the starting location of their service routines in program memory.

Example:

At what address are CS_{50} and IP_{50} stored in memory?

Solution:

Address= 4 × 50 = 200

and expressing it as a hexadecimal number results in

Address= $C8_{16}$

Therefore, IP^{50} is stored at $000C8^{16}$ and CS^{50} at $000CA^{16}$.

Interrupt Instructions

A number of instructions are provided in the instruction set of the 8086 microprocessors for use with interrupt processing. Figure below lists these instructions.

Mnemonic	Meaning	Format	Operation	Flags affected
CLI	Clear interrupt flag	CLI	0 → (IF)	IF
STI	Set interrupt flag	STI	1 → (IF)	IF
INT n	Type n software interrupt	INT n	(Flags)→((SP)-2) 0→TF,IF (CS)→((SP)-4) (2+4*n)→CS (IP)→((SP)-6) (4*n)→(IP)	TF, IF
IRET	Interrupt return	IRET	((SP))→(IP) ((SP)+2)→ (CS) ((SP)+4)→(Flags) (SP)+ 6→(SP)	All
INTO	Interrupt on overflow	INTO	INT 4 steps	TF , IF
HLT	Halt	HLT	Wait for an external interrupt or reset to occur	None
WAIT	Wait	Wait	Wait for $\overline{TEST}$ input to go active	None

Interrupt instructions.

- **STI** enables the external interrupt request (INTR) input for operation by setting IF, while **CLI** disable the external interrupt input by resetting IF.
- **INT n** instruction is used to initiate a vectored call of a subroutine.
- **IRET** instruction must be included at the end of each interrupt service routine.
- **INTO** is the interrupt-on-overflow instruction. This instruction must be included after arithmetic instructions that can result in an overflow condition, such as divide. It tests the overflow flag, and if the flag is found to be set, a **type 4** internal interrupt is initiated.

External Hardware Interrupt Interface Signals

- When an interrupt request has been recognized on the **NMI** pin, the 8086 initiate **type 2** interrupt (CS_2:IP_2).
  - ✓ It cannot be masked by **IF**.
  - ✓ The NMI input is positive edge triggered. Therefore, a request for service is automatically latched internal to the MPU.

- When an interrupt request has been recognized on the **INTR** pin, then ○ If **IF**= 0 then the interrupt request is ignored.

✓ If **IF**= 1 then 8086

1. saves the flag register on the stack,

2. saves the old program context on the stack,

3. and clears TF and IF.

4. respond with two pulses at INTA during interrupt acknowledge bus cycle .

▪ The first pulse signals the external circuitry that the interrupt request has been acknowledged and to prepare to sent the number to the 8086.

▪ The second pulse tells the external circuitry to put the type number on the data bus.

- **RESET** :
  - ✓ The reset input of the 8086 MPU provides a hardware means for initializing the microcomputer.
  - ✓ After reset the MPU start execution at address:

- CS : IP = FFFFH : 0000H
- This mean the physical address is FFFF0
- What instructions should be written in this address?

Internal interrupt function

It is involve four types: divide error, overflow error, single step, and breakpoint.

Single Step

The single-step function relates to an operation option of the 8086. If the trap flag (TF) is set, the single-step mode of operation is enabled. When TF is set, the MPU initiates a type 1 interrupt to the service routine defined by IP_1 and CS_1 at addresses 00004_{16} and 00006_{16}, respectively, at the completion of every instruction of the user program.