8086/ Interrupts

10.11.12

Objectives

- Introduction to interrupts
- Interrupt Vectors
- Interrupt instructions
- Interrupt control

Introduction to Interrupts

- current executing program ---TO--→ an ISR/ISP/INT handler.
  - ISR: Routine, which performs whatever functions necessary is servicing the interrupt
- A transfer is initiated by
  - the hardware in response to special internal or external condns
  - A software by means of INT instruction
- when an interrupt occurs,
  - Executes ISR
  - Address of the associated interrupt service routine needs to specify.
  - each vector contains the address of an interrupt service procedure

Interrupt Vector

- Each vector contains a value for IP and CS that forms the address of the ISR/ISP.
  - the first 2 bytes contain IP;
  - the last 2 bytes CS
- 256 different interrupt vectors
- Interrupt vectors (32–255) are available to users
  - such as the divide error interrupt

Reserved Interrupt Vectors

- First 32 interrupt vectors reserved for the present and future products.
- Some reserved vectors are for errors that occur during the execution of software
- Some vectors are reserved for the coprocessor.
  - others occur for normal events in the system
- In a personal computer, reserved vectors are used for system functions

INTs

- 256 different software interrupt instructions (INTs) available to the programmer.
  - each INT instruction has a numeric operand whose range is 0 to 255 (00H–FFH)
- For example, INT 100 uses interrupt vector 100, which appears at memory address 190H–193H.
- Each INT instruction is 2 bytes long.
  - the first byte contains the opcode
  - the second byte contains the vector type number

- Address of the interrupt vector is determined by multiplying the interrupt type number by 4.
  - INT 10H instruction calls the interrupt service procedure whose address is stored beginning at memory location 40H (10H × 4) in the mode
- In protected mode, the interrupt descriptor is located by multiplying the type number by 8
  - because each descriptor is 8 bytes long

Interrupt types

- Software interrupt
  - Type of the interrupt is specified in the INT instruction.
- Hardware interrupts
  - Type of the interrupt is supplied by the interrupting hardware.
  - Internal Hardware Interrupts
    - hardware interrupts that are generated internally to the processor
    - on the occurrence of an error condition.

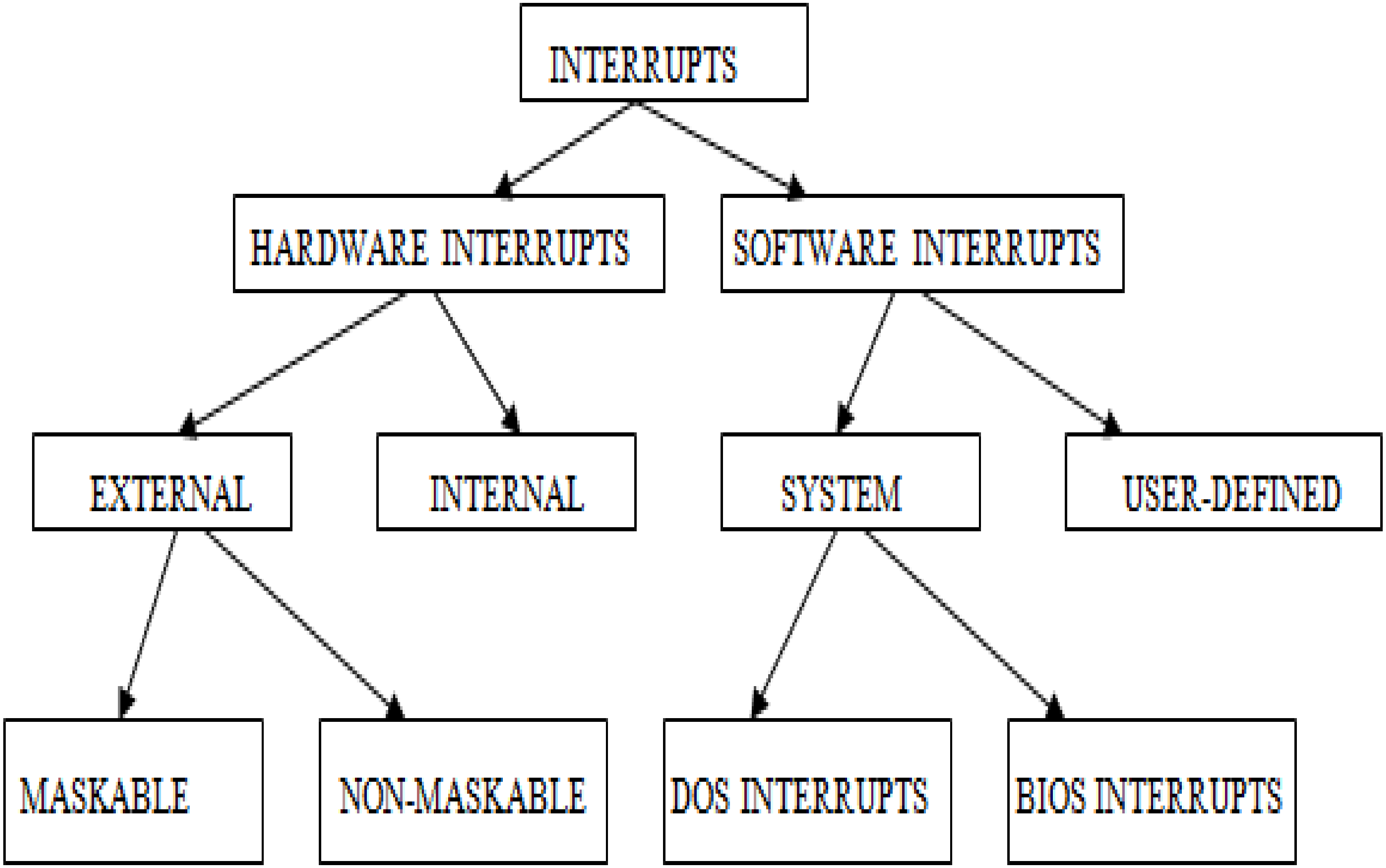
Interrupt

- External Hardware Interrupts
  - Used to alert processor that peripheral device needs the attention
  - Generated by peripheral devices
  - Are the main mechanism used by these devices to get the attention of the processor.
  - Maskable/Non-maskable
  - Non-maskable interrupts are hardware events that must be responded to immediately by the CPU.

- Two control lines that can signal interrupts.
  - INTR (Interrupt Request)
  - NMI (Non-maskable Interrupt).

Interrupt Priority

- INT s are serviced based on priority basis
- Priority level assigned by TYPE NUMBER
- INT interrupted by higher level INT


INTERRUPTS
HARDWARE INTERRUPTS
SOFTWARE INTERRUPTS
EXTERNAL
INTERNAL
SYSTEM
USER-DEFINED
MASKABLE
NON-MASKABLE
DOS INTERRUPTS
BIOS INTERRUPTS

Interrupt Instructions

- Three different interrupt instructions available:
  - INT, INTO, and INT 3
- In real mode, each fetches a vector from the vector table, and then calls the procedure stored at the location addressed by the vector.
- In protected mode, each fetches an interrupt descriptor from the interrupt descriptor table.
- Similar to a far CALL instruction because it places the return address (IP and CS) on the stack.

When Software INT occurs……….

- When a software interrupt executes, it:
  - pushes the flags onto the stack
  - clears the T and I flag bits
    - T =1, microp interrupts the flow o program on conditions as indicated by the debug registers and control registers.
  - pushes CS onto the stack
  - fetches the new value for CS from the interrupt vector
  - pushes IP onto the stack
  - fetches the new value for IP from the vector
  - jumps to the new location addressed by CS and IP/EIP

INT vs far CALL

- INT performs as a far CALL
  - not only pushes CS & IP onto the stack, also pushes the flags onto the stack
- The INT instruction performs the operation of a PUSHF, followed by a far CALL instruction.
- Software interrupts are most commonly used to call system procedures because the address of the function need not be known.
- The interrupts often control printers, video displays, and disk drives.
- INT replaces a far CALL that would otherwise be used to call a system function

IRET/IRETD

- Used only with software or hardware interrupt service procedures.
- IRET instruction will
  - pop stack data back into the IP
  - pop stack data back into CS
  - pop stack data back into the flag register
- Accomplishes the same tasks as the POPF followed by a far RET instruction.

INT 3

- A special software interrupt designed to function as a breakpoint.
  - a 1-byte instruction, while others are 2-byte
- Common to insert an INT 3 in software to interrupt or break the flow of the software.
  - function is called a breakpoint
  - breakpoints help to debug faulty software
- A breakpoint occurs for any software interrupt, but because INT 3 is 1 byte long, it is easier to use for this function.

INTO

- Interrupt on overflow (INTO) is a conditional software interrupt that tests overflow flag (O).
  - if O = 1 and an INTO executes, an interrupt occurs via vector type number 4
  - If O = 0, INTO performs no operation
- JO or INTO instructions detect the overflow.
- The INTO instruction appears in software that adds or subtracts signed binary numbers.
  - with these operations, it is possible to have an overflow

- In software ISR, hardware interrupts are enabled as one of the first steps.
    - accomplished by the STI instruction
- Interrupts are enabled early because just about all of the I/O devices in the personal computer are interrupt-processed.