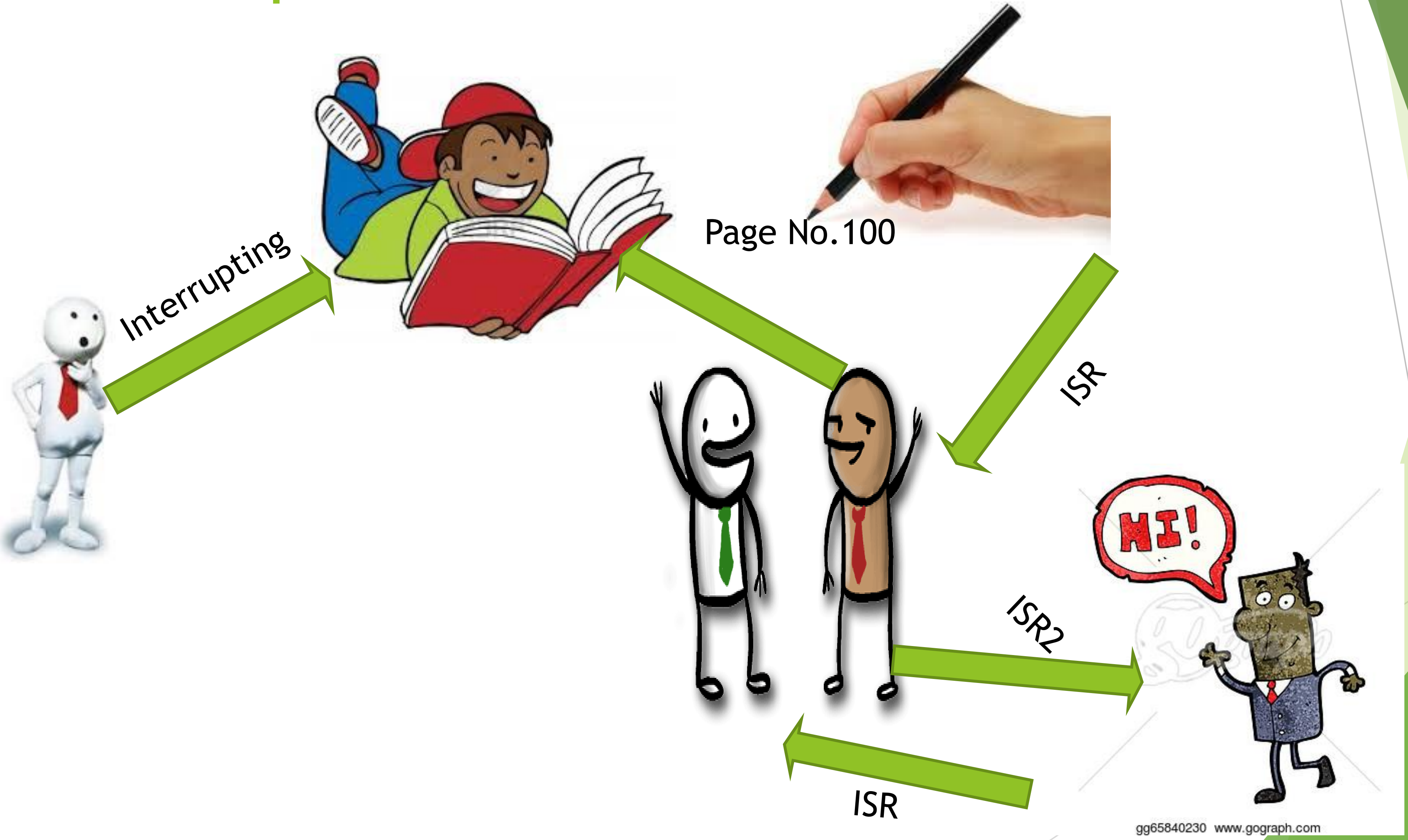
Interrupts of 8086

Vidyadhar Aski

What are Interrupts ?

- In dictionary meaning “Interrupt” is to break the sequence of operation.
- While CPU executing a program, an interrupt breaks the execution and diverts execution to some other program called ISR (Interrupt service routine)
- After ISR execution the control is transferred back to the main program.

An example



Interrupting
Page No.100
ISR
HI!
ISR2
ISR
gg65840230 www.gograph.com

Types

- Software interrupts : (When M/P interrupted by giving instruction INTn, then software interrupt)
- Hardware interrupts : (when M/P is interrupted by giving signal on hardware interrupt pin , then its hardware interrupt)

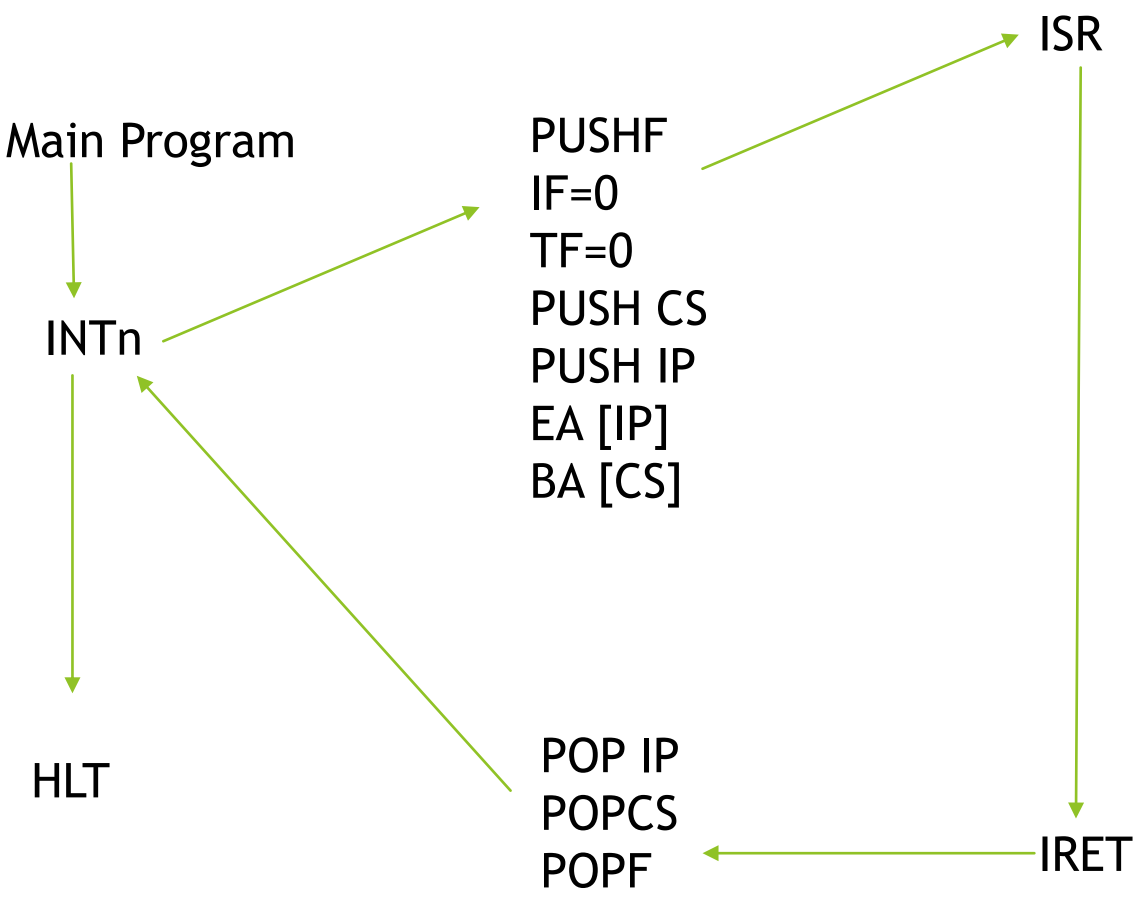
Software interrupts

- Instruction used for S/W interrupt is INTn, where n is 0,1,2,…255 decimal or 00 to ffh.
- When M/P executes INTn instruction then it performs following functions.
- Original values of flags are stored in stack
- IF Flag made 0, so hardware interrupt INTR is disabled
- TF Flag made 0, so ISR not executed in single step mode

Continued..

- The old values of EA and BA of the next instruction Y are saved from the IP and CS on to the stack
- i.e PUSH IP, PUSH CS
- Now the M/P transfers the new values of EA and BA of ISR from 4 successive memory location into IP and CS
- Starting address of these four memory locations is obtained using equation 4*n

Software interrupts



Main Program

INTn

HLT

PUSHF
IF=0
TF=0
PUSH CS
PUSH IP
EA [IP]
BA [CS]

ISR

IRET

POP IP
POPCS
POPF

Fig: S/W interrupts in 8086

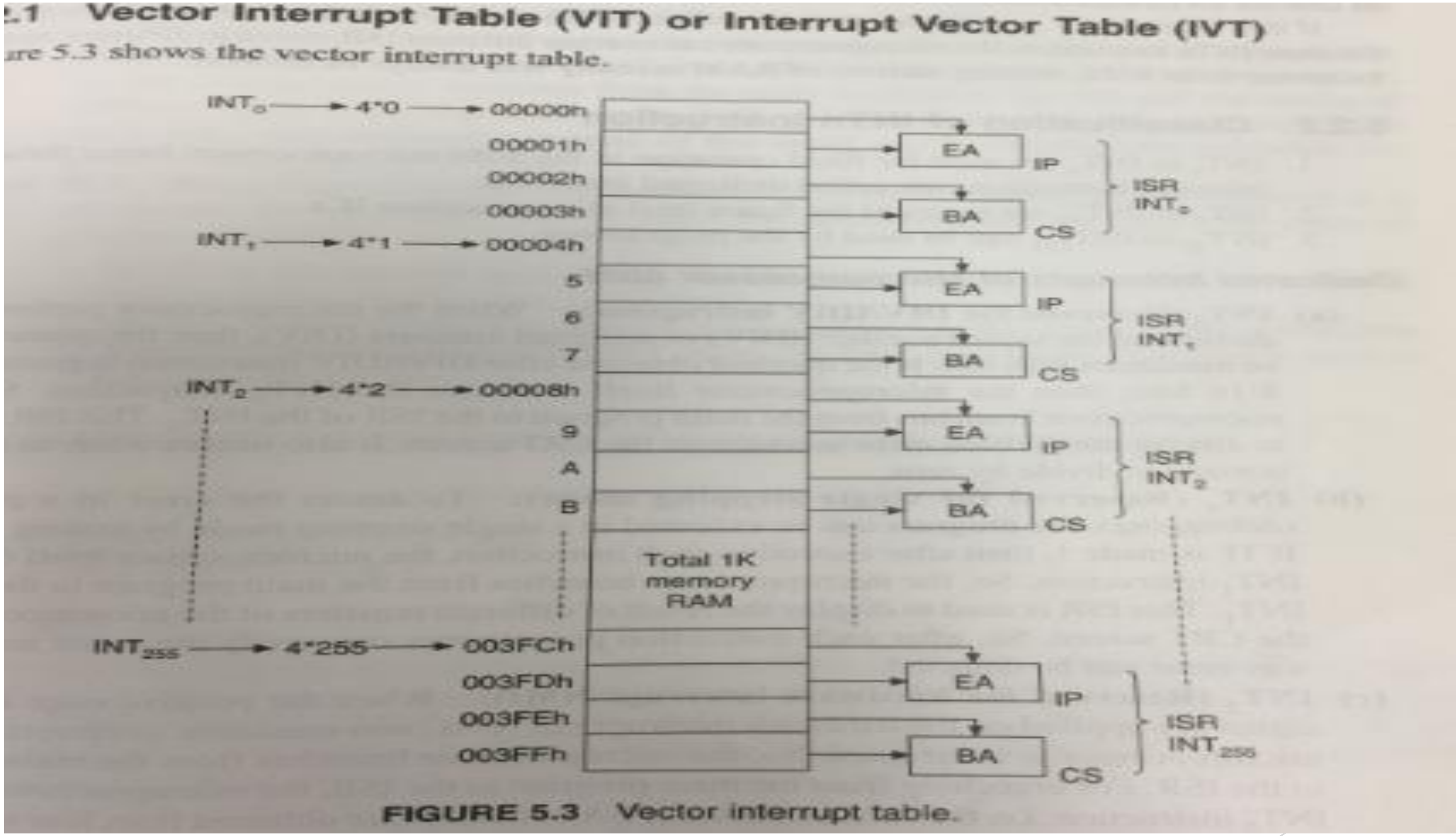
Vector Interrupt Table

- When the M/P executes INTn instruction, then it branches from main program to the ISR
- The base address and effective address of the ISR are transferred into the CS and IP from four memory locations
- The Physical address of the first memory location is obtained by using is obtained using equation 4*n
- For example,
- For INT0, n=0 hence the PA of the first memory location = 4*n => 4*0=00000h

Continued..

- From first 2 memory location new EA is transferred into IP, and the last 2 memory location new BA is transferred into CS
- For example,
- For INT0, n=0 hence the PA of the first memory location = 4*n => 4*0=00000h
- From first 2 memory location new EA is transferred into IP, and the last 2 memory location new BA is transferred into CS
- 4 Memory locations are required for each INTn instruction
- So total memory required is 256*4=1024, 1Kmemory locations required starting from 00000h to 003FFh and is reserved for storing EA and BA of ISR for INT255 instructions

IVT (interrupt vector table)



.1 **Vector Interrupt Table (VIT) or Interrupt Vector Table (IVT)**

ure 5.3 shows the vector interrupt table.

FIGURE 5.3 Vector interrupt table.

Classification of INTn instruction

- INT0 to INT4 are used for fixed operation in the 8086 microprocessor, so they are called as dedicated interrupts
- INT5 to INT31 for future intel Ics
- INT32 to INT255 can be used by programmer

Dedicated interrupts

- **INT0 (reserved for DIV/IDIV instruction):** divide by zero/if quotient greater than 8/16 bits, then this interrupt is used to print error message on CRT
- **INT1 (reserved for single stepping mode)** : used to display the content of all registers on CRT on a single step execution
- **INT2 (Reserved for hardware interrupts NMI): when +ve edge and the level signal are applied on hardware interrupt pin NMI (non-maskable Interrupt) then M/P is interrupted and branch to ISR**

Continued..

- When the power supply is made off, then interrupt signal is applied on NMI pin and M/P branches from main program to ISR , This ISR is normally used to transfer data from non battery back up RAM to battery back up RAM
- Int3 (reserved for breakpoint technique)
- Single stepping mode to detect errors will take more time
- Using breakpoint technique the time required to find out error is reduced
- ISR of the Int3 is to display the content of each register after group of instructions instead of single step execution

Continued..

- INT4 (reserved for INTO instruction: The INTO (interrupt on overflow) is a conditional interrupt instruction in which the M/P will check the value of overflow flag (OF)
- If OF = 0 then M/P will not interrupted, instead it'll execute next Instruction
- If OF = 1 then the M/P is interrupted , and M/P branches from main program to ISR using INT4 Instruction, this ISR is to display overflow message

HARDWARE INTERRUPTS

- Hardware interrupts are generated by hardware devices when something unusual happens; this could be a key-press or a mouse move or any other action.
- **Maskable Interrupts:**
  The processor can inhibit certain types of interrupts by use of a special interrupt mask bit. This mask bit is part of theflags/condition code register, or a special interrupt register. In the 8086 microprocessor if this bit is clear, and aninterrupt request occurs on the Interrupt Request input, it is ignored.

Continued..

- **Non-Maskable Interrupts:**
  There are some interrupts which cannot be masked out or ignored by the processor. These are associated with highpriority tasks which cannot be ignored (like memory parity or bus faults). In general, most processors support the Non-Maskable Interrupt (NMI). This interrupt has absolute priority, and when it occurs, the processor will finish thecurrent memory cycle, then branch to a special routine written to handle the interrupt request.