

Chapter 3

A Top-Level View of Computer Function And Interconnection – Part 02

Dr. Mostafa Elhosseini

Interrupts

| Agenda

 I/O module

 Port

 Interrupt

- What ? ✓
- Function ✓
- Interrupt Types
- How it work

 Short I/O wait

 Long I/O wait

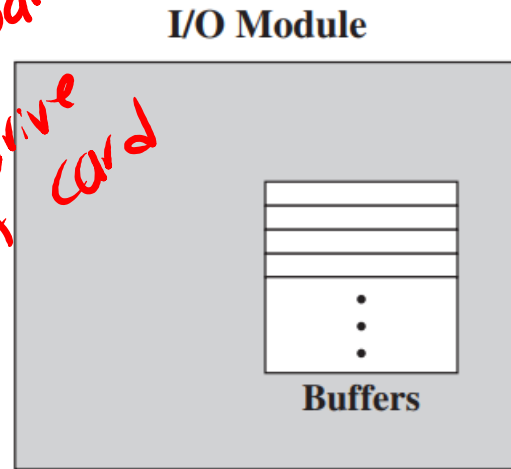
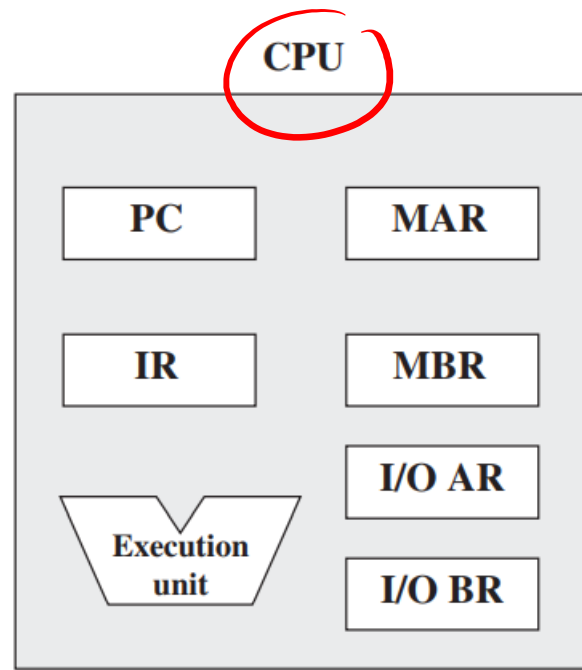
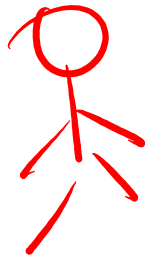
 Interrupt Vector

 Interrupt Vector table

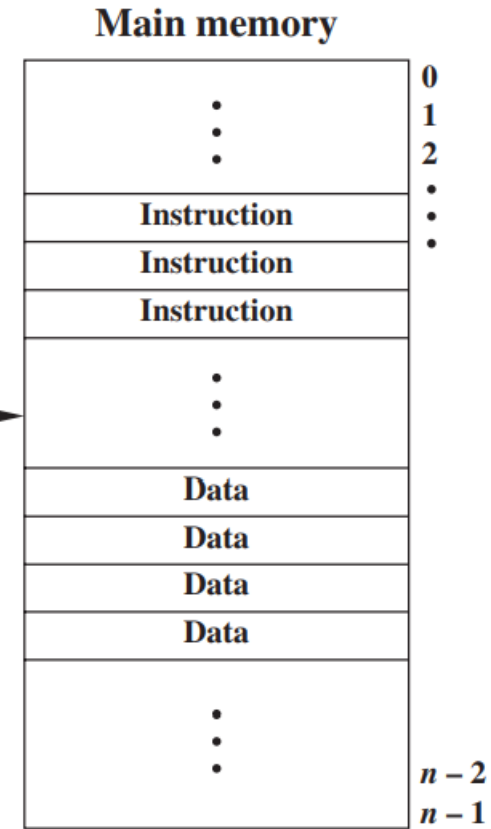
 Interrupt Service Routine ISR – Interrupt Handler

Von
Neumann

Printer
Keyboard
mouse
disk drive
VGA card



System
bus



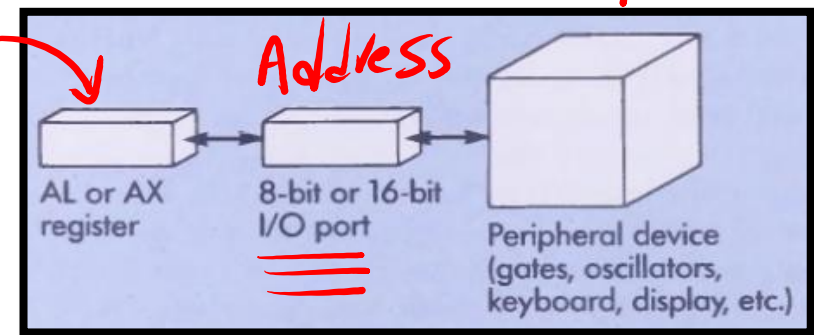
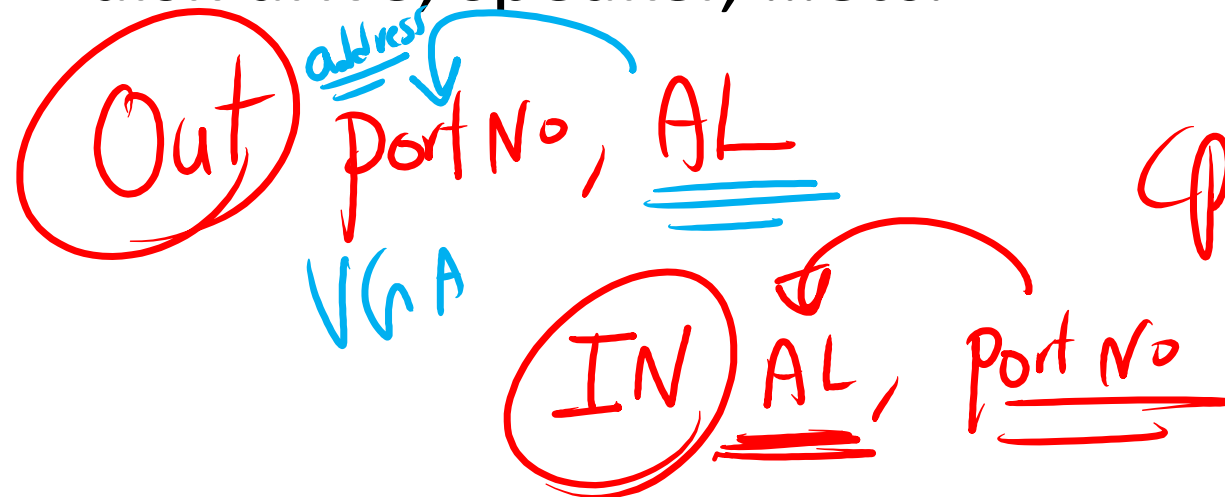
PC	=	Program counter
IR	=	Instruction register
MAR	=	Memory address register
MBR	=	Memory buffer register
I/O AR	=	Input/output address register
I/O BR	=	Input/output buffer register

Ports

cpu

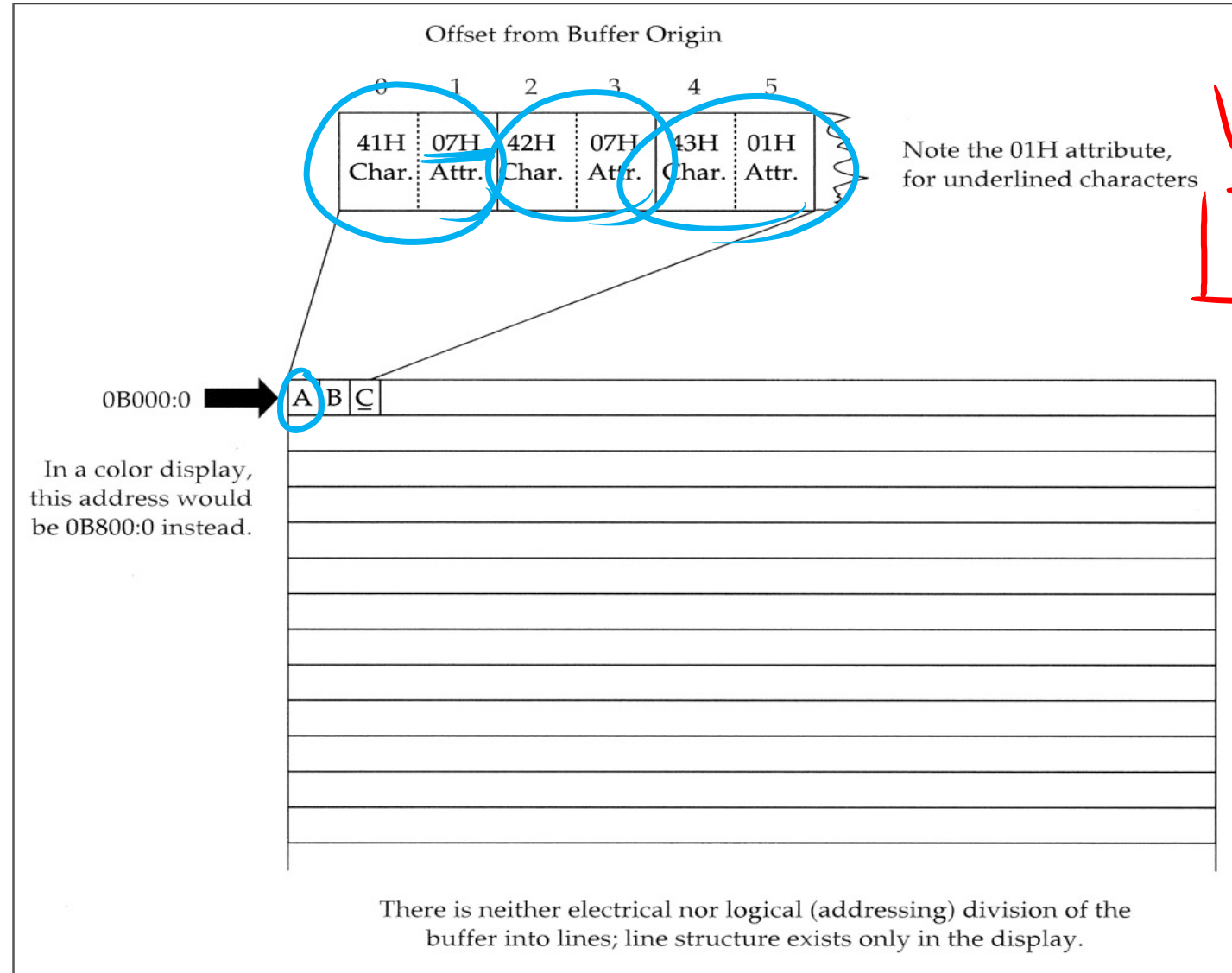
- Ports are somewhat like register, however ports are connected to physical device
- when you change something in a port, you are sending a message to some peripheral device, such as video screen, disk drive, speaker, ...etc.

I/O unit



| Ports

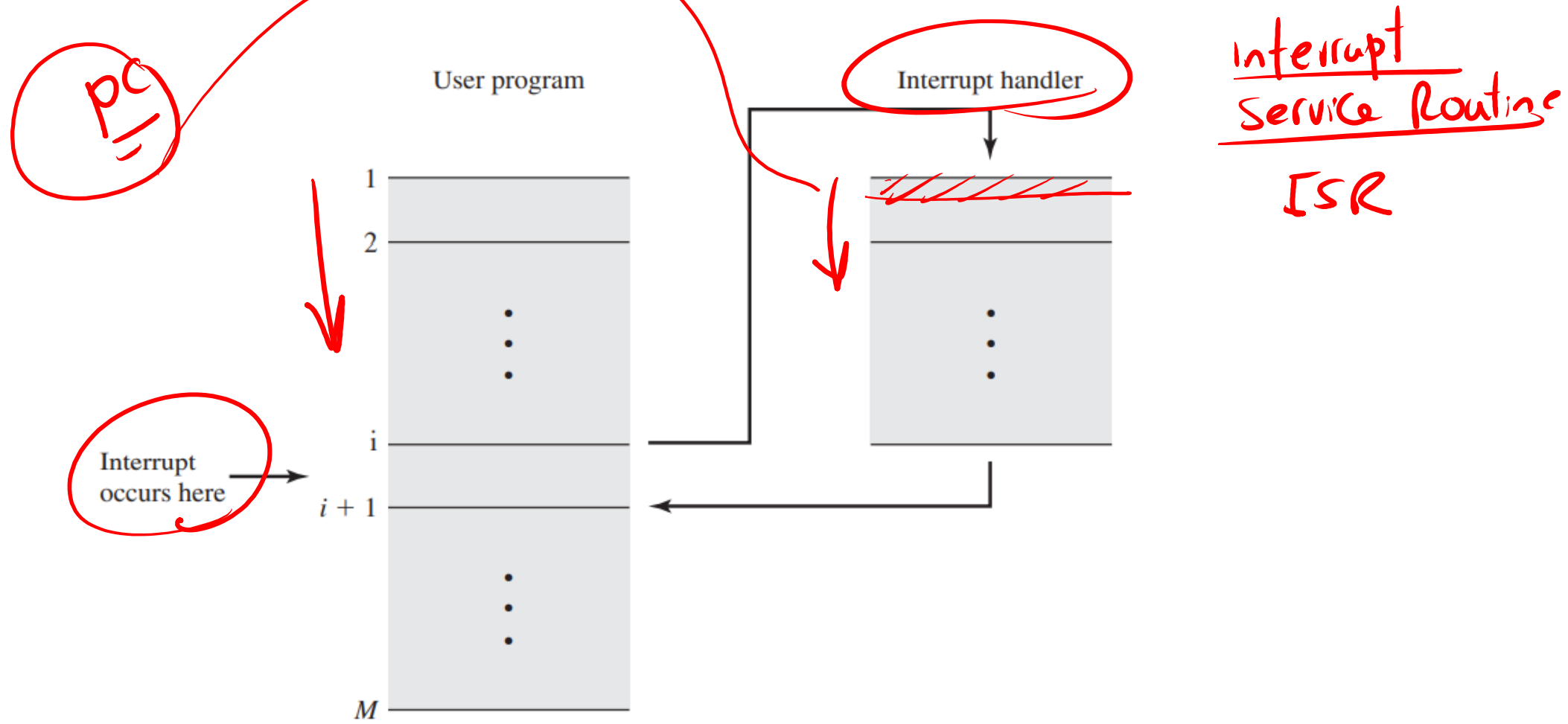
ASCII
code
41H
0100 0001
italic
Bold
blink
Color



VGA card
0B800 H

monochrome
0B000 H

| Interrupt - point of view of the user program

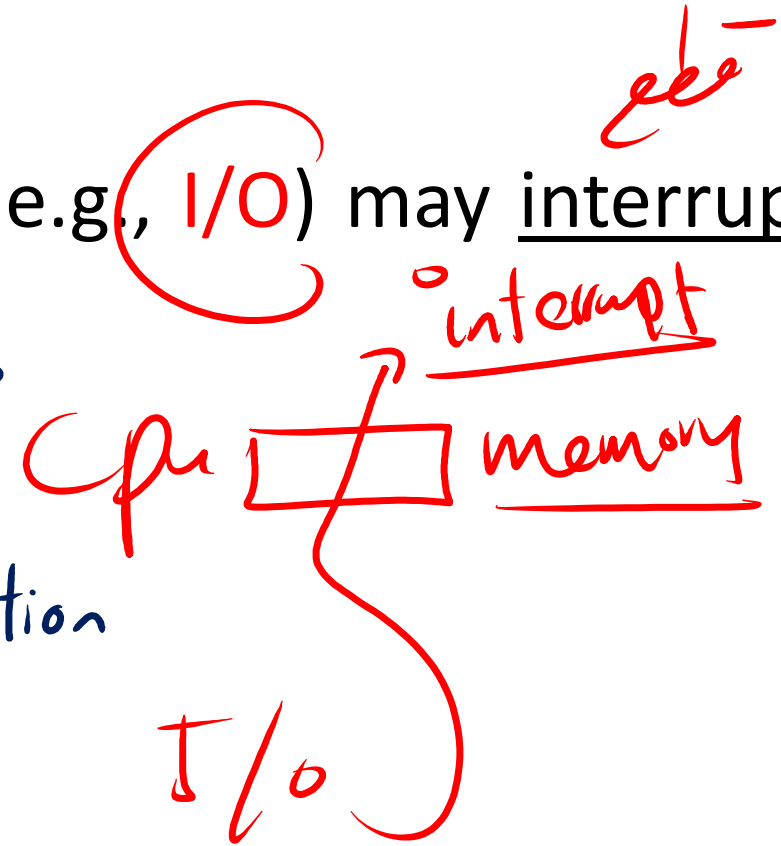


| Interrupt

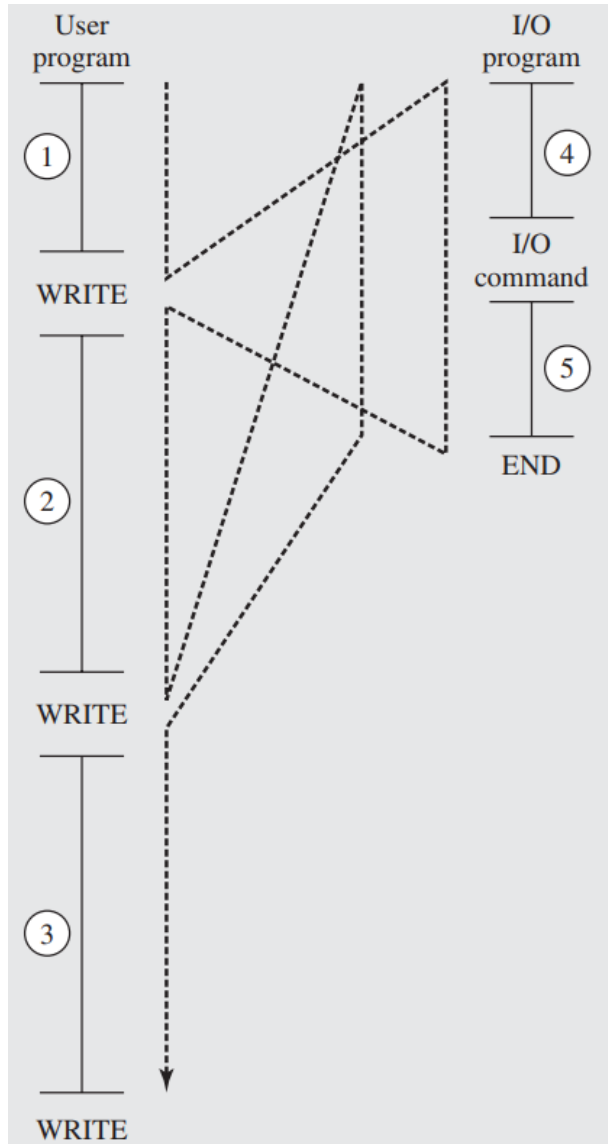
Mechanism by which other modules (e.g., I/O) may interrupt normal sequence of processing

- Program $\Rightarrow$ Overflow — division by zero
- Timer $\Rightarrow$ multi tasking
- I/O $\Rightarrow$ normal completion of an operation
- Hardware failure

memory parity
bit

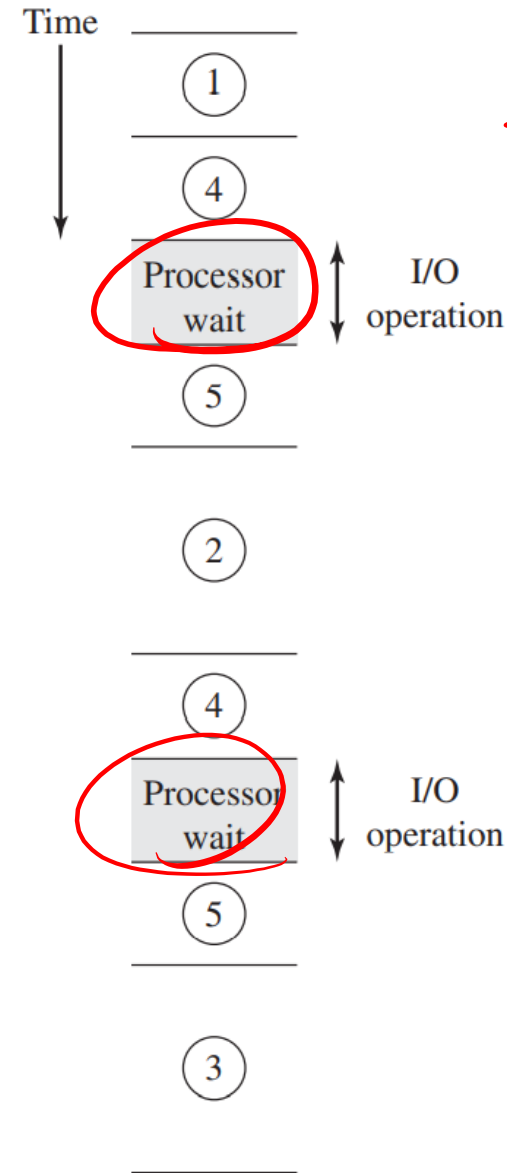


Program Flow Control



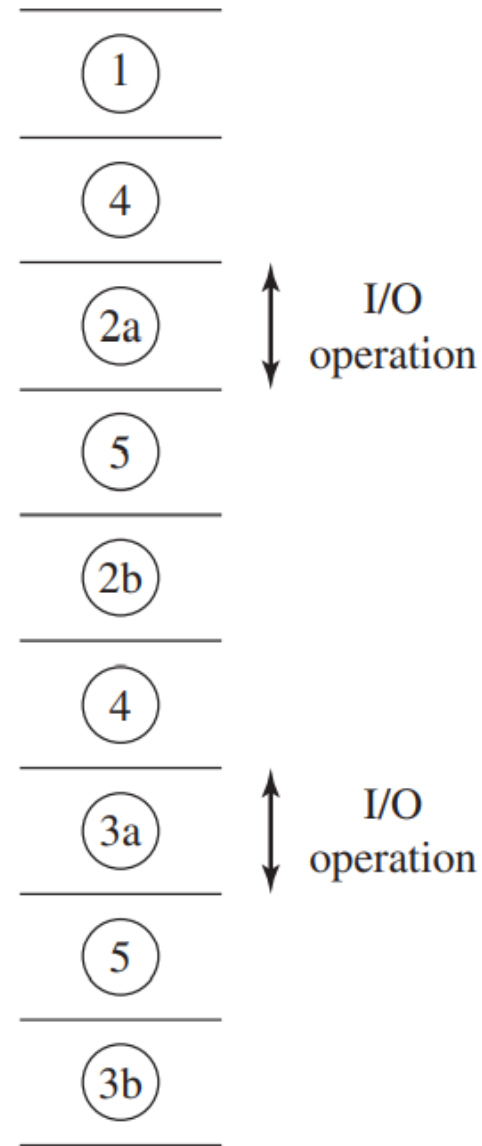
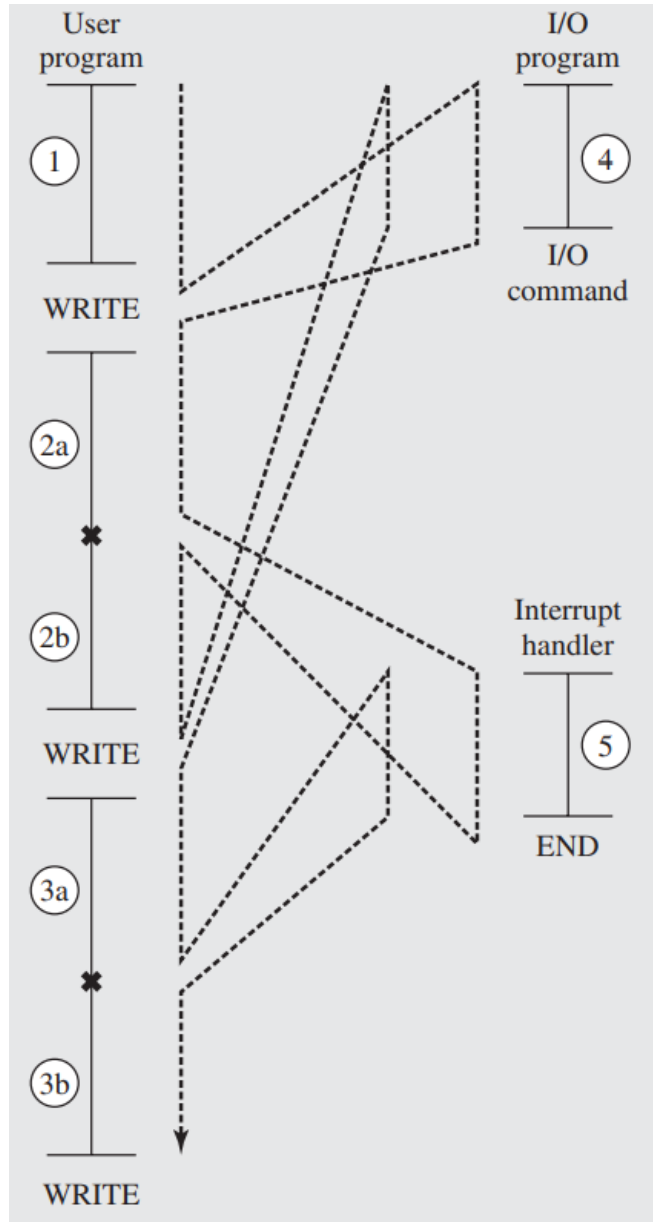
prepare

flag < success
failure

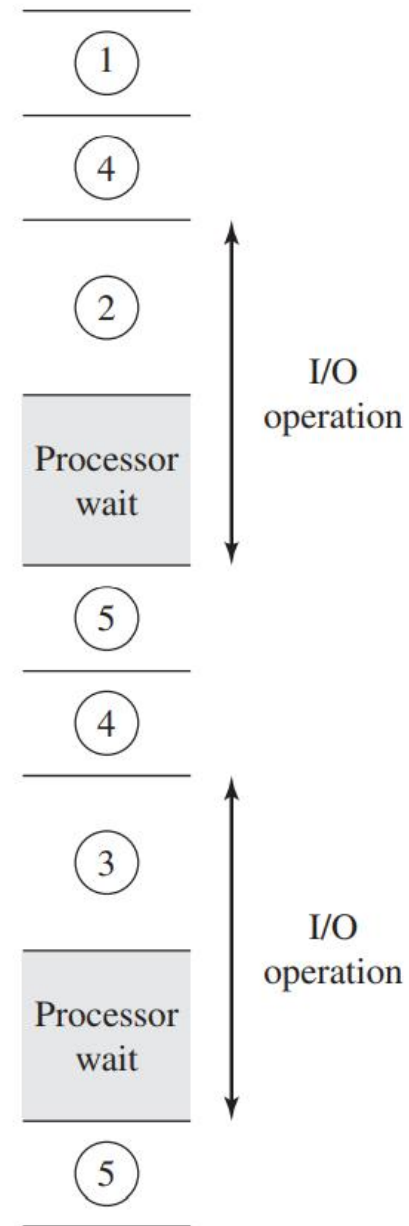
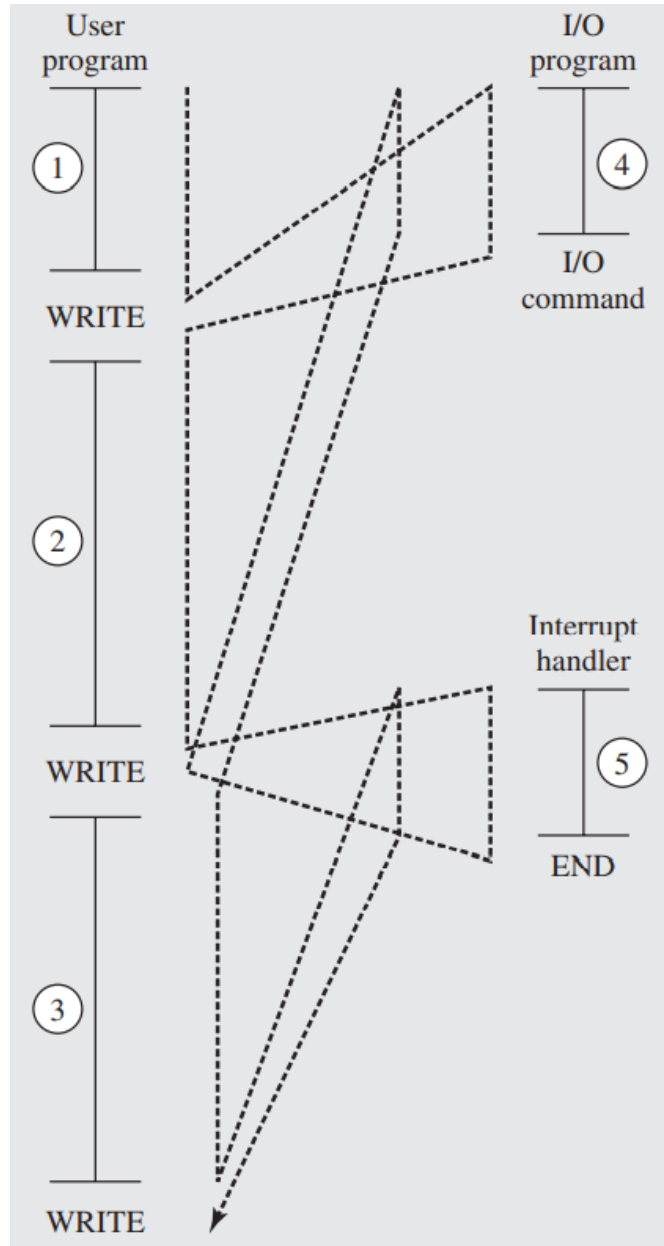


Without Interrupt

Program Flow Control – Short wait



Program Flow Control – Long wait



Interrupt – 8086 MP

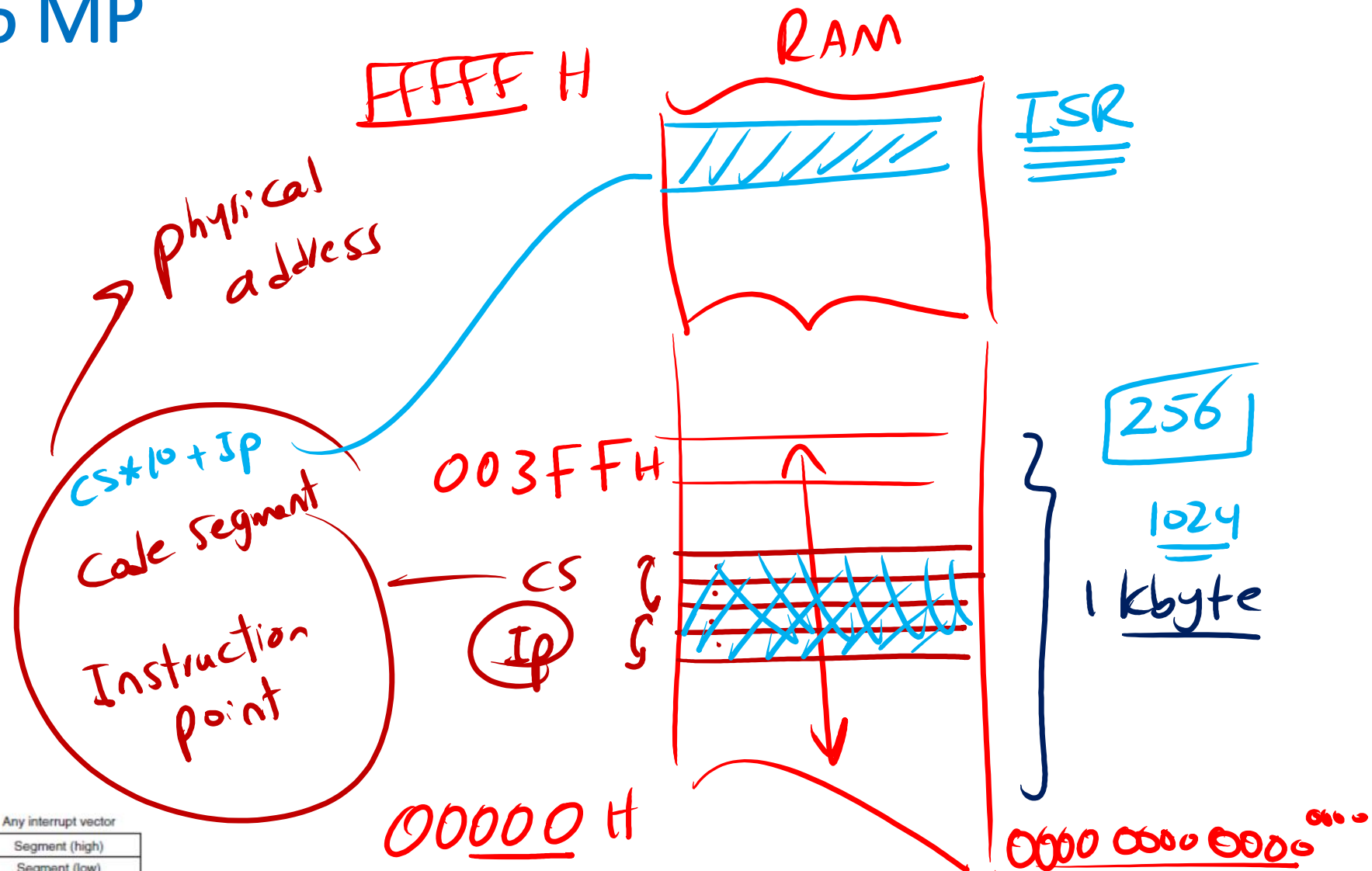
FIGURE 12-2 (a) The interrupt vector table for the microprocessor and (b) the contents of an interrupt vector.

	Type 32 — 255
	User interrupt vectors
080H	Type 14 — 31
	Reserved
	Type 16
040H	Coprocessor error
03CH	Type 15
	Unassigned
038H	Type 14
	Page fault
034H	Type 13
	General protection
030H	Type 12
	Stack segment overrun
02CH	Type 11
	Segment not present
028H	Type 10
	Invalid task state segment
024H	Type 9
	Coprocessor segment overrun
020H	Type 8
	Double fault
01CH	Type 7
	Coprocessor not available
018H	Type 6
	Undefined opcode
014H	Type 5
	BOUND
010H	Type 4
	Overflow (INTO)
00CH	Type 3
	1-byte breakpoint
008H	Type 2
	NMI pin
004H	Type 1
	Single-step
000H	Type 0
	Divide error

(a)

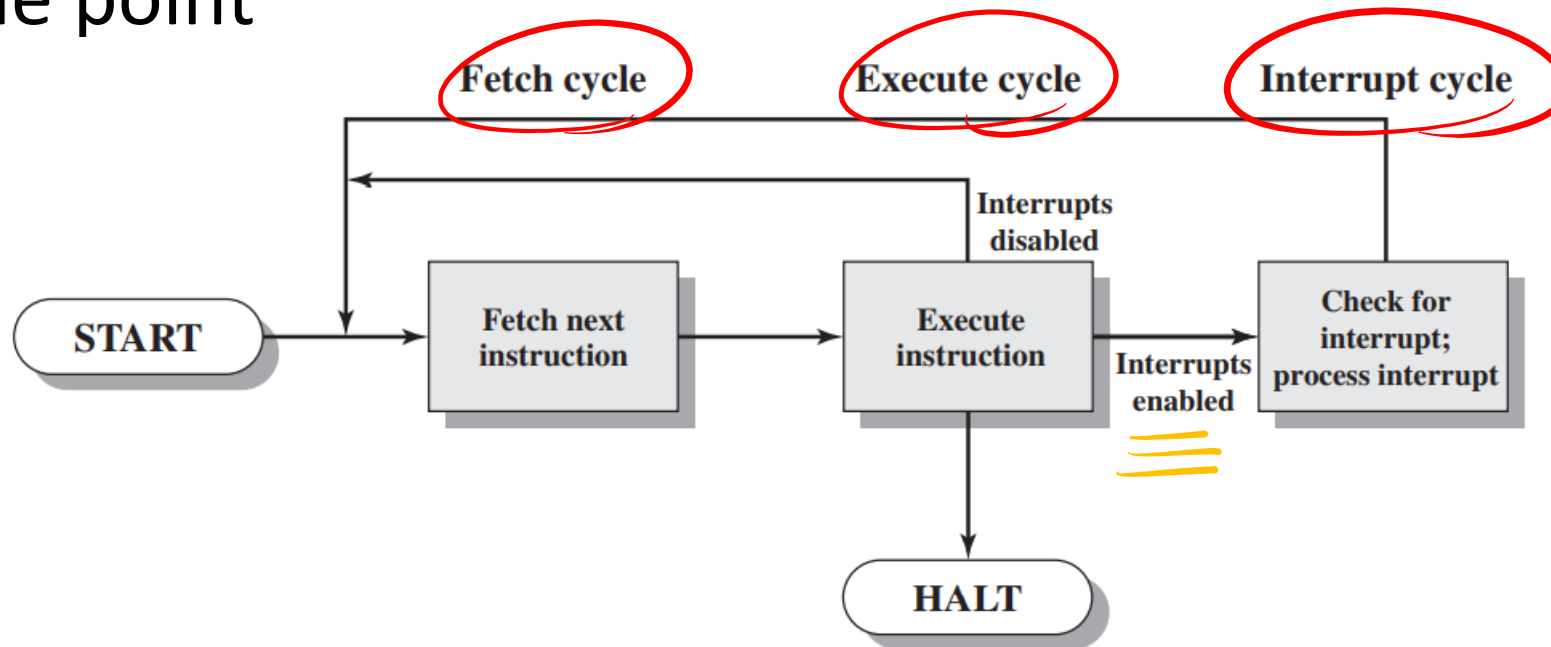
Any interrupt vector	
3	Segment (high)
2	Segment (low)
1	Offset (high)
0	Offset (low)

(b)



| Instruction Cycle with Interrupts

- The processor and the operating system are responsible for suspending the user program and then resuming it at the same point



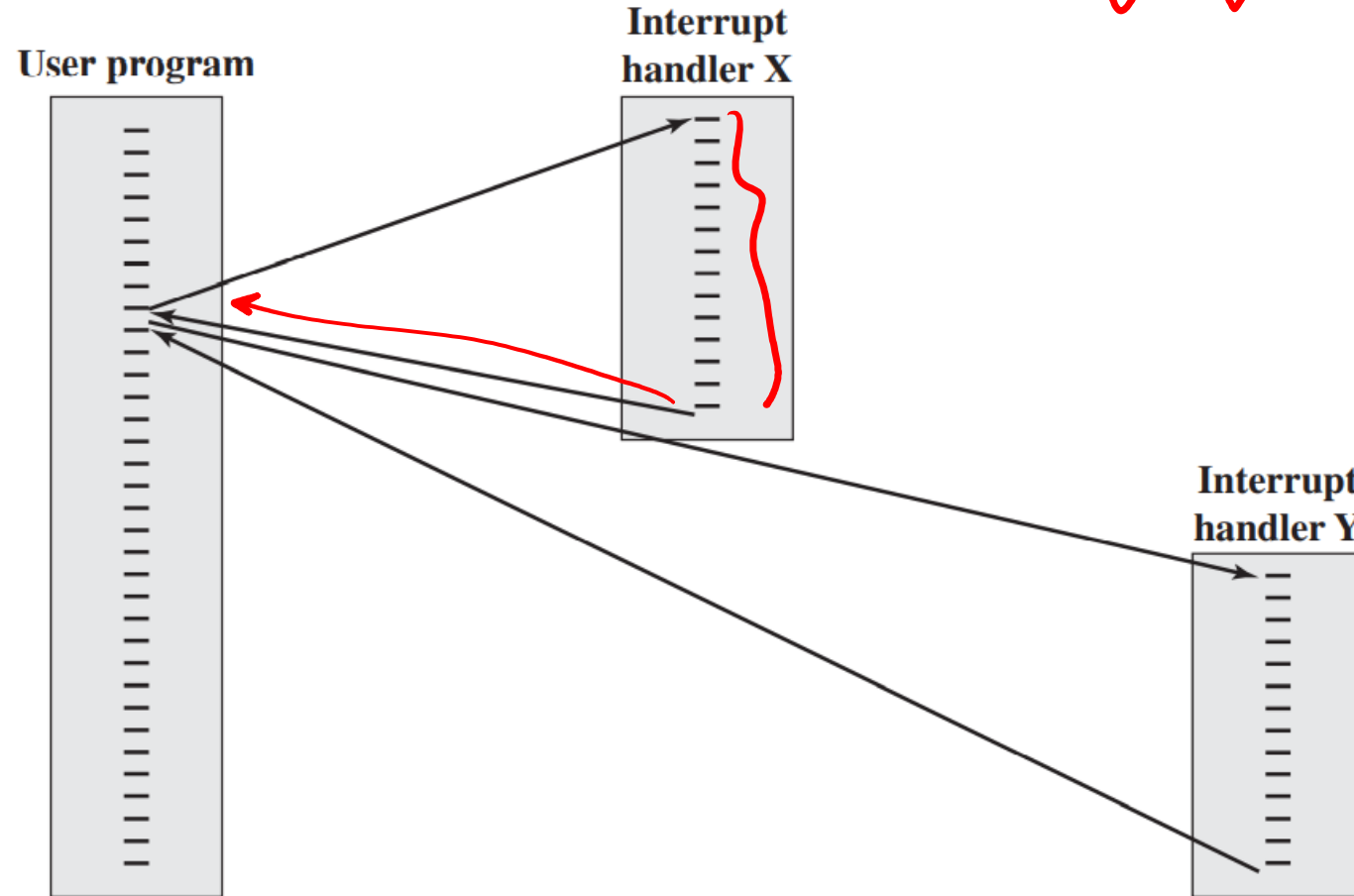
| Instruction Cycle with Interrupts

- Processor checks for interrupt
 - Indicated by an interrupt signal
- If no interrupt, fetch next instruction
- If interrupt pending:
 - Suspend execution of current program
 - Save context
 - Set PC to start address of interrupt handler routine
 - Process interrupt ⇒ ISR
 - Restore context and continue interrupted program

Register

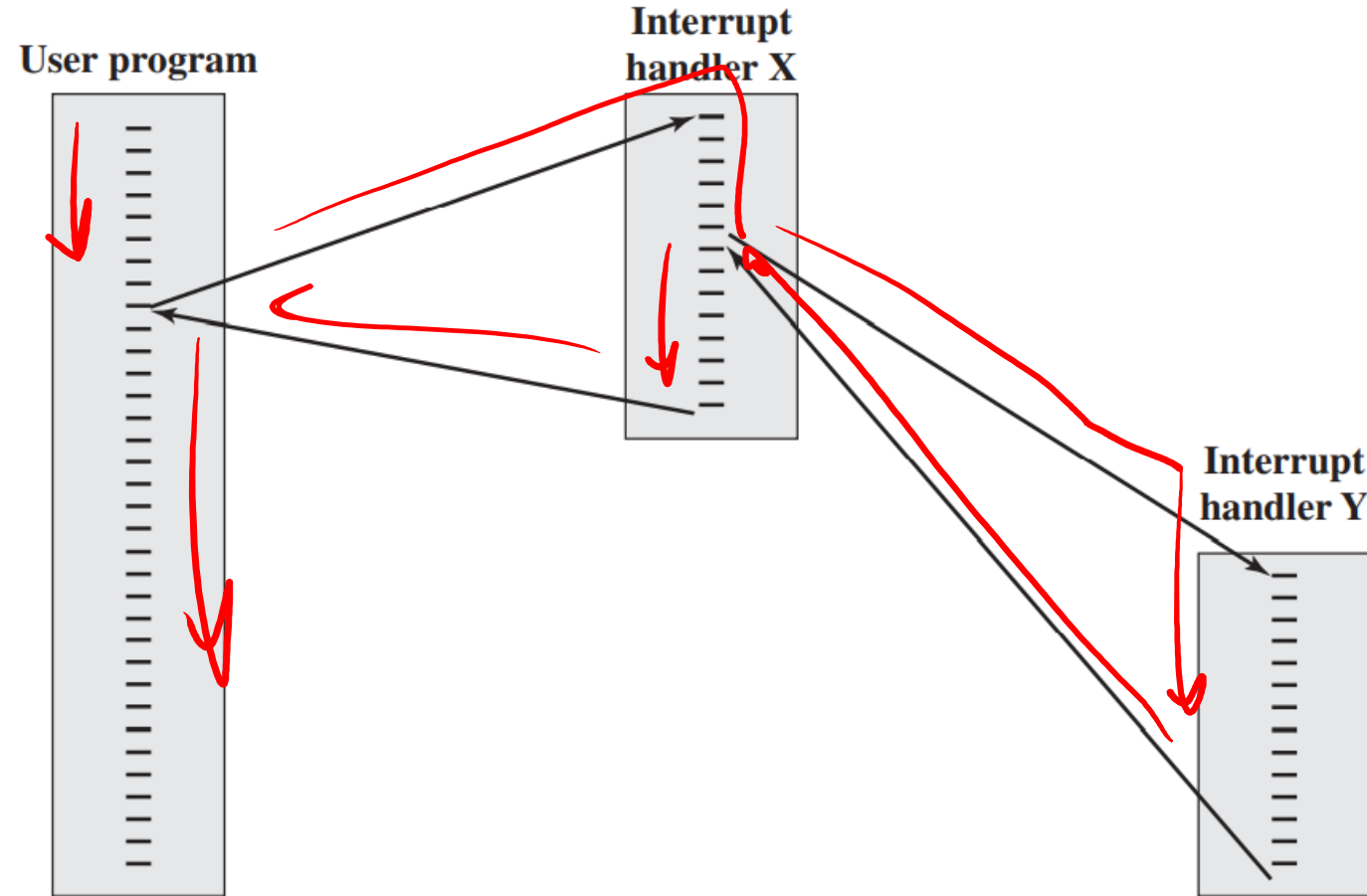
ISR

| Multiple Interrupts – Sequential



disable

| Multiple Interrupts – Nested



| Multiple Interrupts – Example

