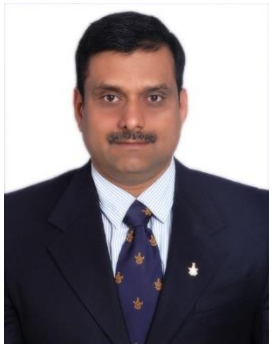
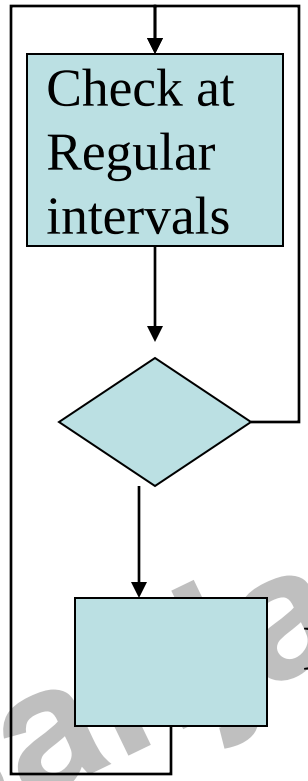
Microprocessors and Interfaces: 2021-22
Lecture 21 : Interrupts

By Dr. Sanjay Vidhyadharan



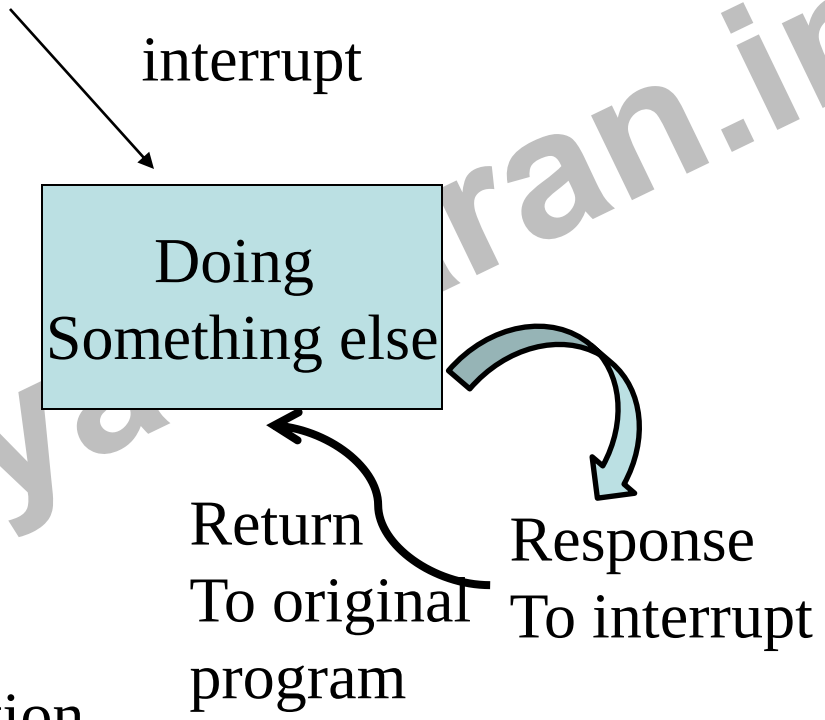
Interrupts vs. Polling

Polling

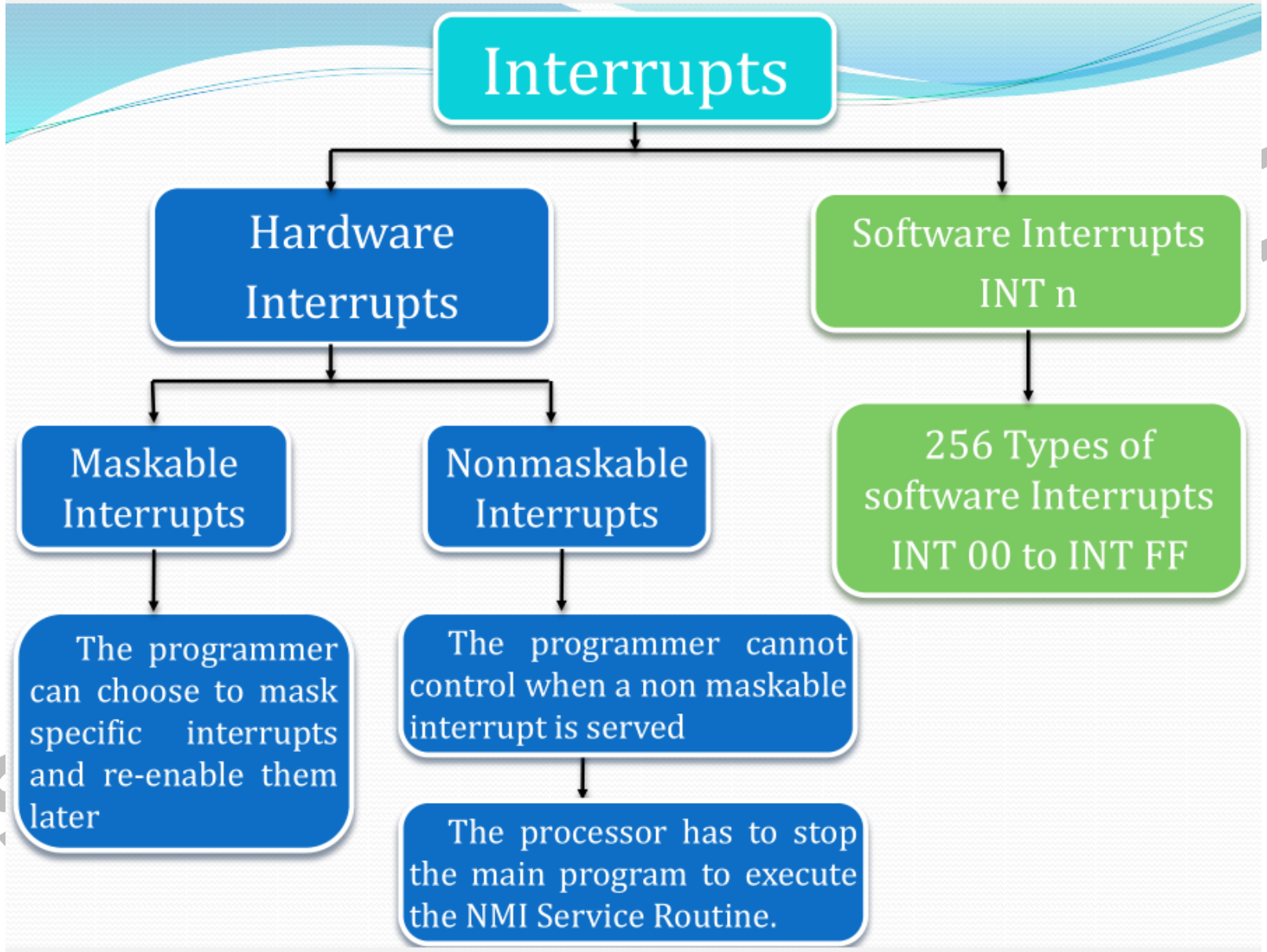


No key pressed

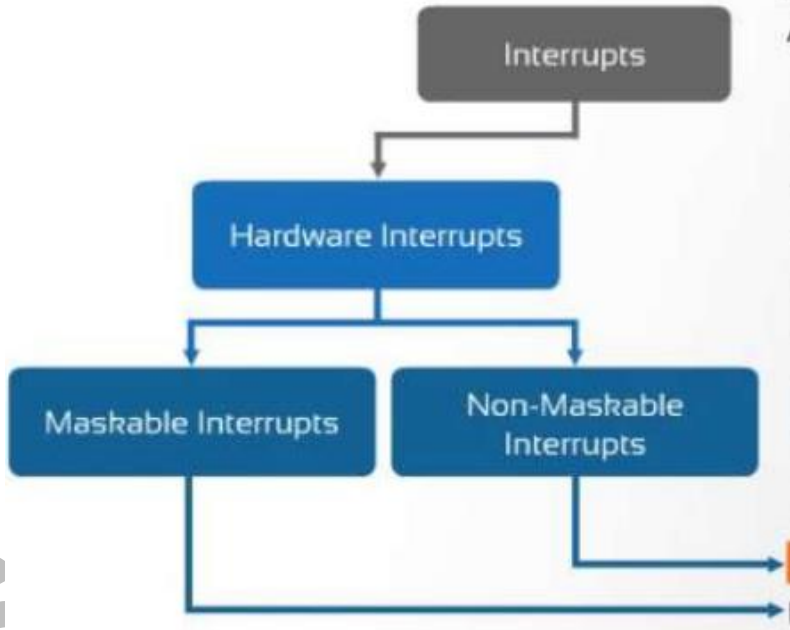
Do key pressed action

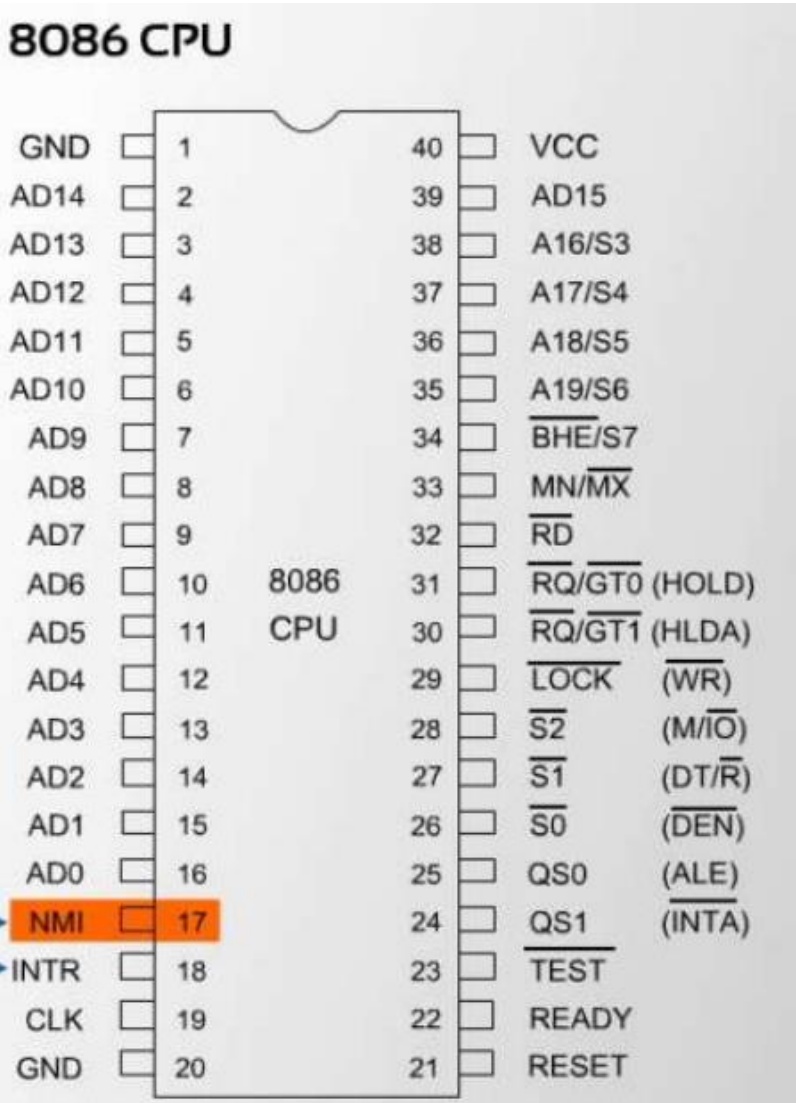


Types of 8086 Interrupts

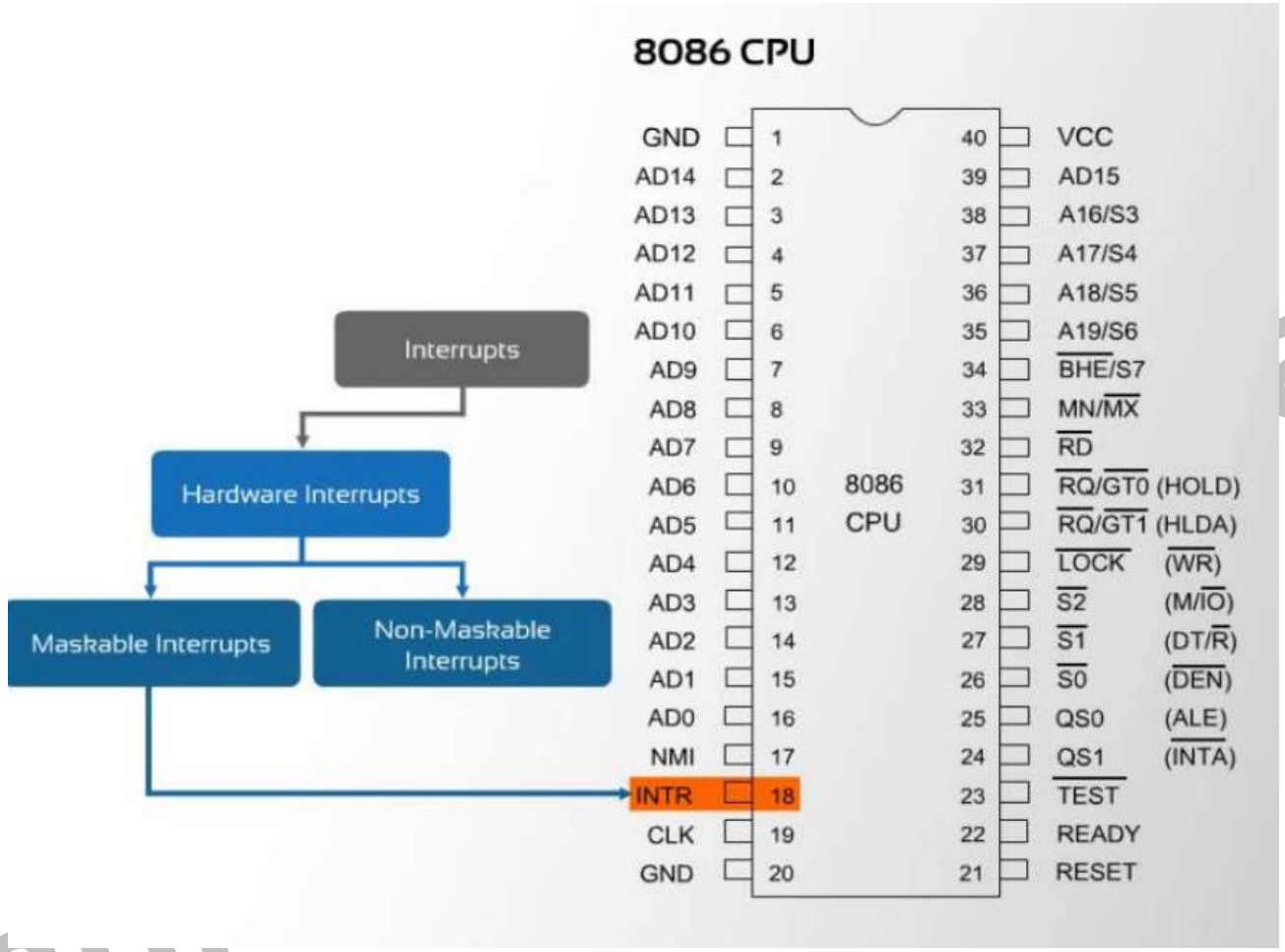


8086 Interrupts





8086 Interrupts

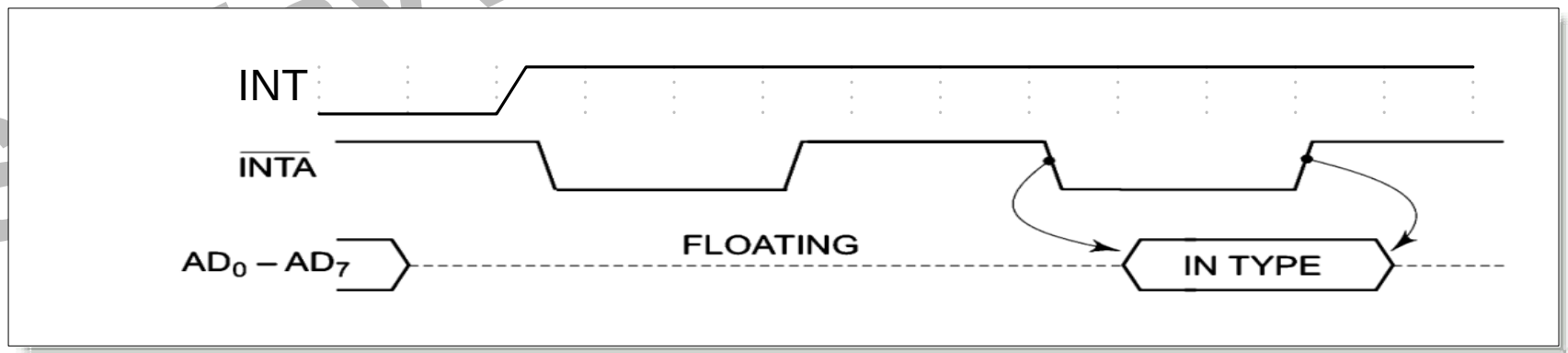


D_{15}	D_{14}	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
				O	D	I	T	S	Z		AC		P		CY

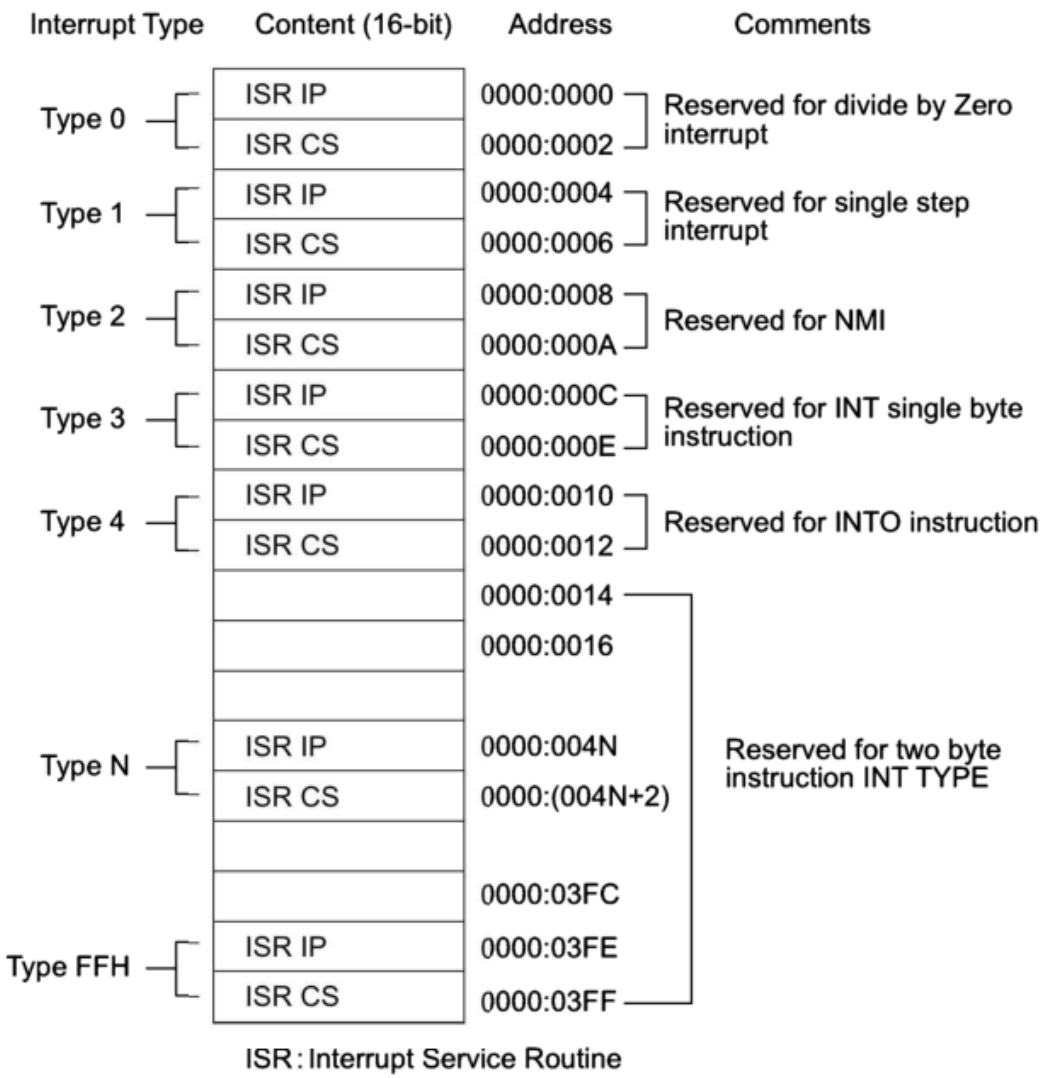
Hardware Interrupts : Interrupt pins and timing

x86 Interrupt Pins

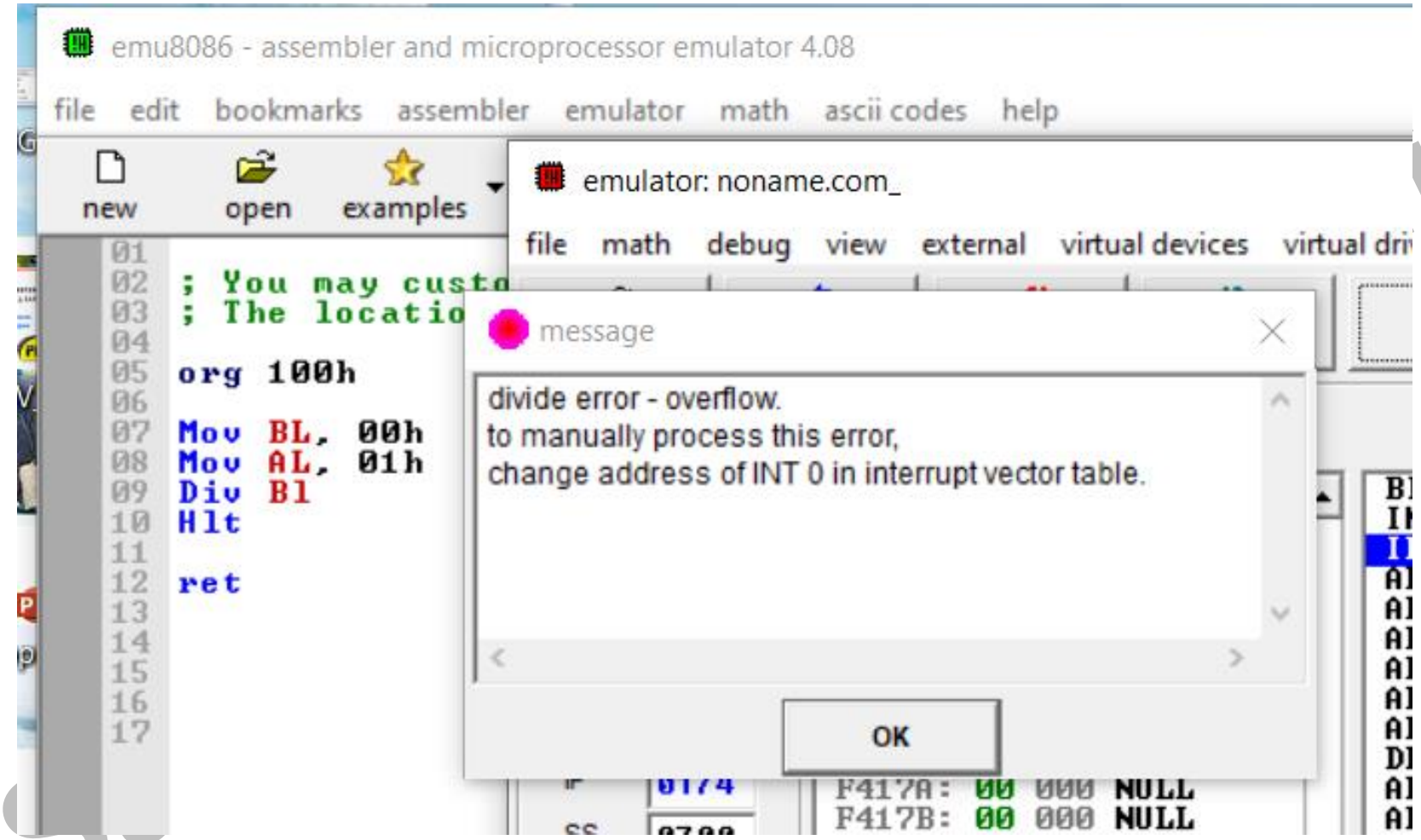
- INTR: Interrupt Request. Activated by a peripheral device to interrupt the processor.
  - Level triggered. Activated with a logic 1.
- /INTA: Interrupt Acknowledge. Activated by the processor to inform the interrupting device that the interrupt request (INTR) is accepted. **(IF Checked)**
  - Level triggered. Activated with a logic 0.
- NMI: Non-Maskable Interrupt. Used for major system faults such as parity errors and power failures.
  - Edge triggered. Activated with a positive edge (0 to 1) transition.
  - Must remain at logic 1, until it is accepted by the processor.
  - Before the 0 to 1 transition, NMI must be at logic 0 for at least 2 clock cycles.
  - **INT 02h**



8086 Vector Table



INT 0


emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples
emulator: noname.com_
file math debug view external virtual devices
; You may custo
; The locatio
org 100h
Mov BL, 00h
Mov AL, 01h
Div Bl
Hlt
ret
message
divide error - overflow.
to manually process this error,
change address of INT 0 in interrupt vector table.
OK
F417A: 00 000 NULL
F417B: 00 000 NULL

Interrupt Sub-Routine (ISR)

256 Interrupts Of 8086 are Divided in To 3 Groups

1. **Type 00 to Type 04 interrupts-**
These are used for fixed operations and hence are called dedicated interrupts

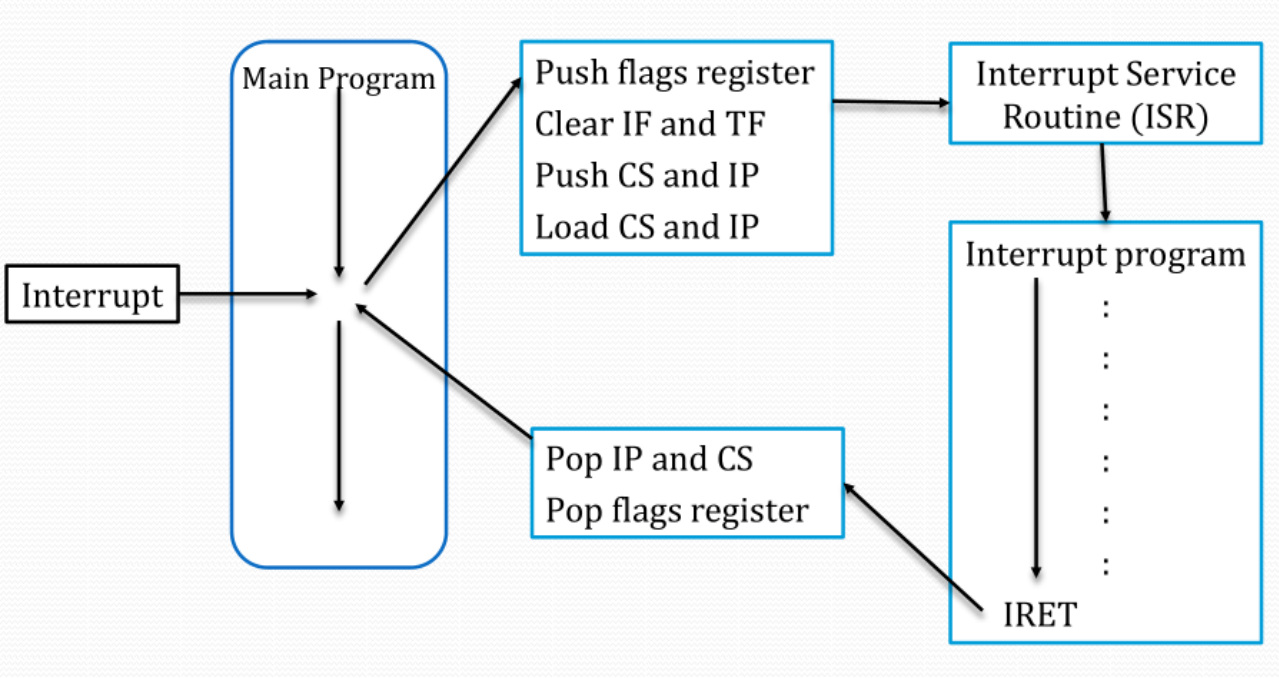
2**. Type 05 to Type 31 interrupts**
Preserved for Higher processors

3. **Type 32 to Type 255 interrupts**

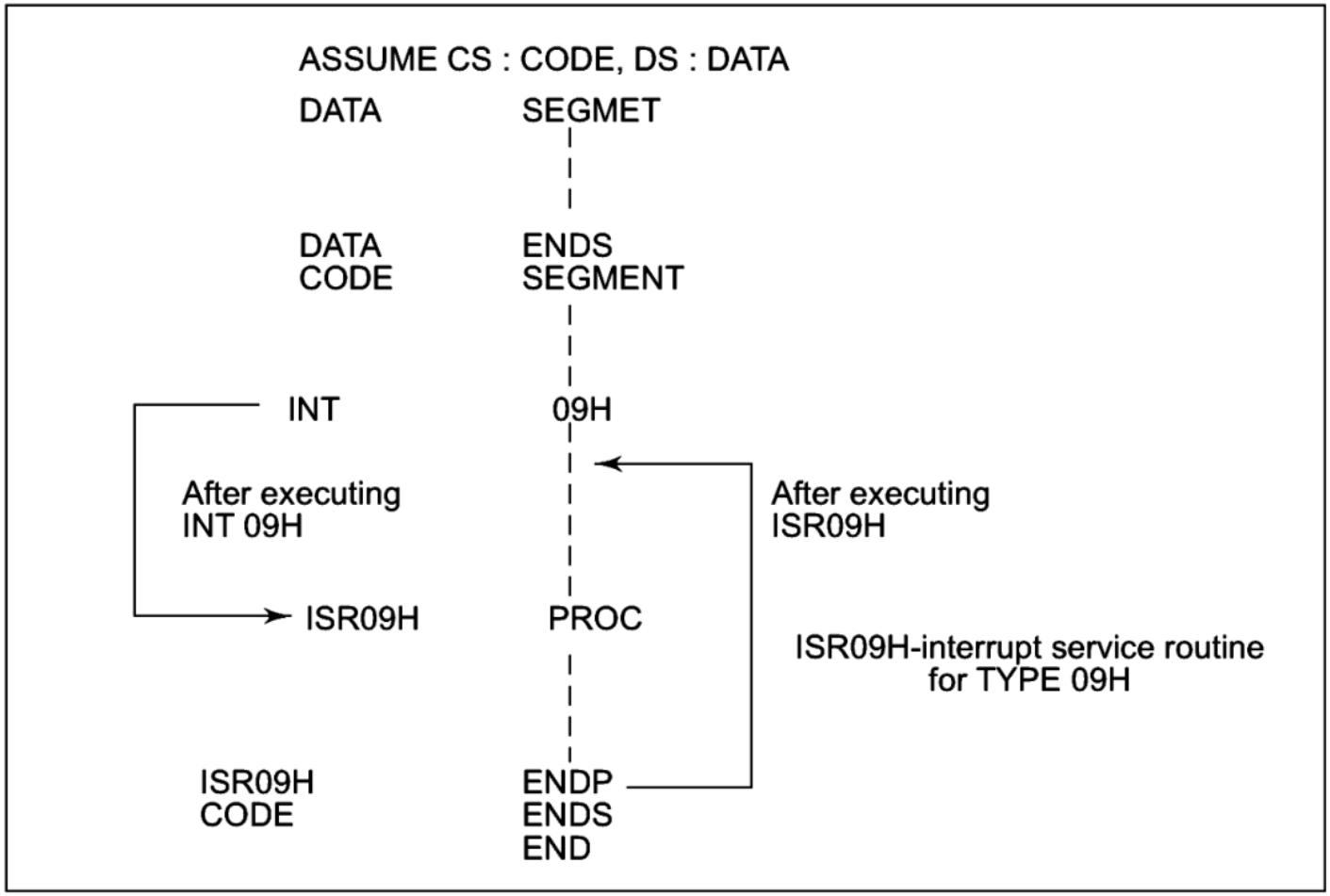
Available for user, called user defined interrupts these can be H/W interrupts and activated through INTR line or can be S/W interrupts.
[E.g. int 21 : 34^{th} interrupt in vector Table or Type 33]

Processing of an Interrupt by the 8086

1. It decrements the stack pointer by 2 and pushes the flag register on the stack.
2. It disables the 8086 INTR interrupt input by clearing the interrupt flag in the flag register.
3. It resets the trap flag in the flag register.
4. It decrements the stack pointer by 2 and pushes the current code segment register contents on the stack.
5. It decrements the stack pointer again by 2 and pushes the current instruction pointer contents on the stack.



Soft Interrupts 8086



Bus Cycle

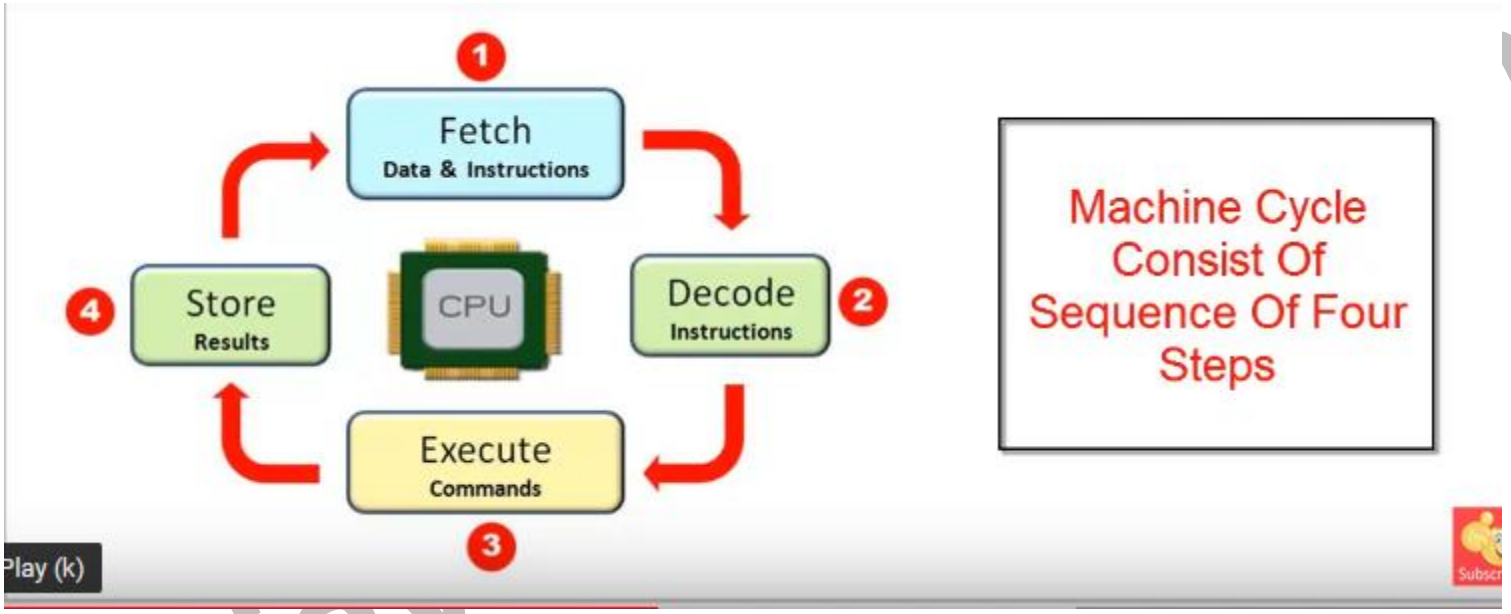
Instruction Cycle

Machine Cycle

T states

Bus Cycle

Machine Cycle



Bus Cycle

Instruction Cycle

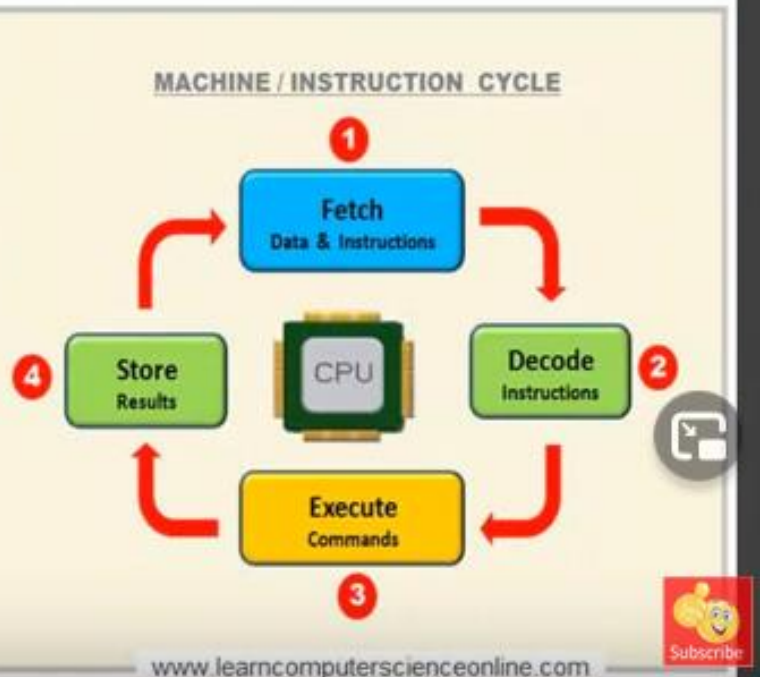
Has more than 1 Machine Cycles

CPU - Instruction Cycle / Machine Cycle.

The CPUs instruction cycle is executed sequentially and each instruction is processed by CPU which consist of following steps :

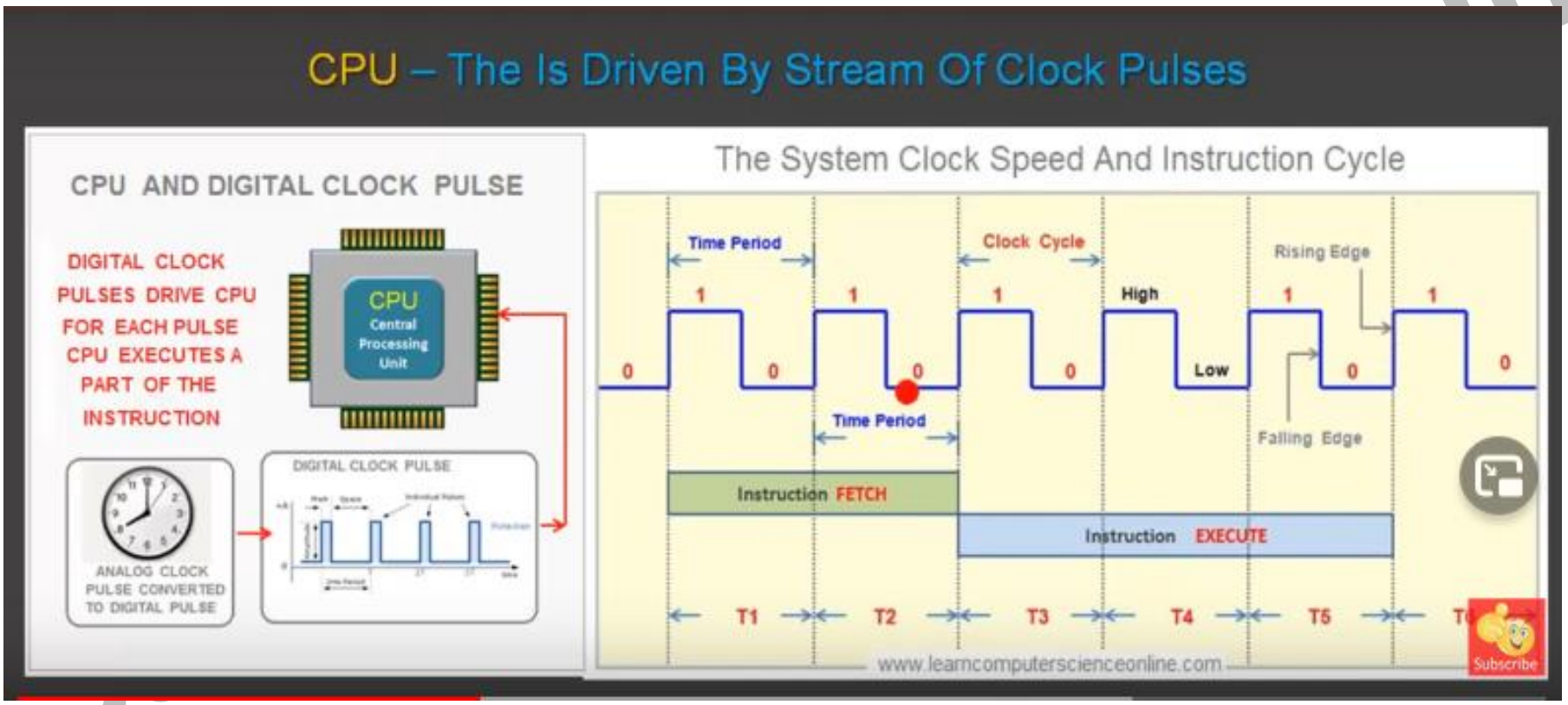
- Read an Instruction from memory .
- Decode the instruction as per OPCODE
- Find the address of operand .
- Retrieve an operand .
- Perform the desired operation
- Find the address of destination memory .
- Store the result into the destination memory

www.learncomputerscienceonline.com

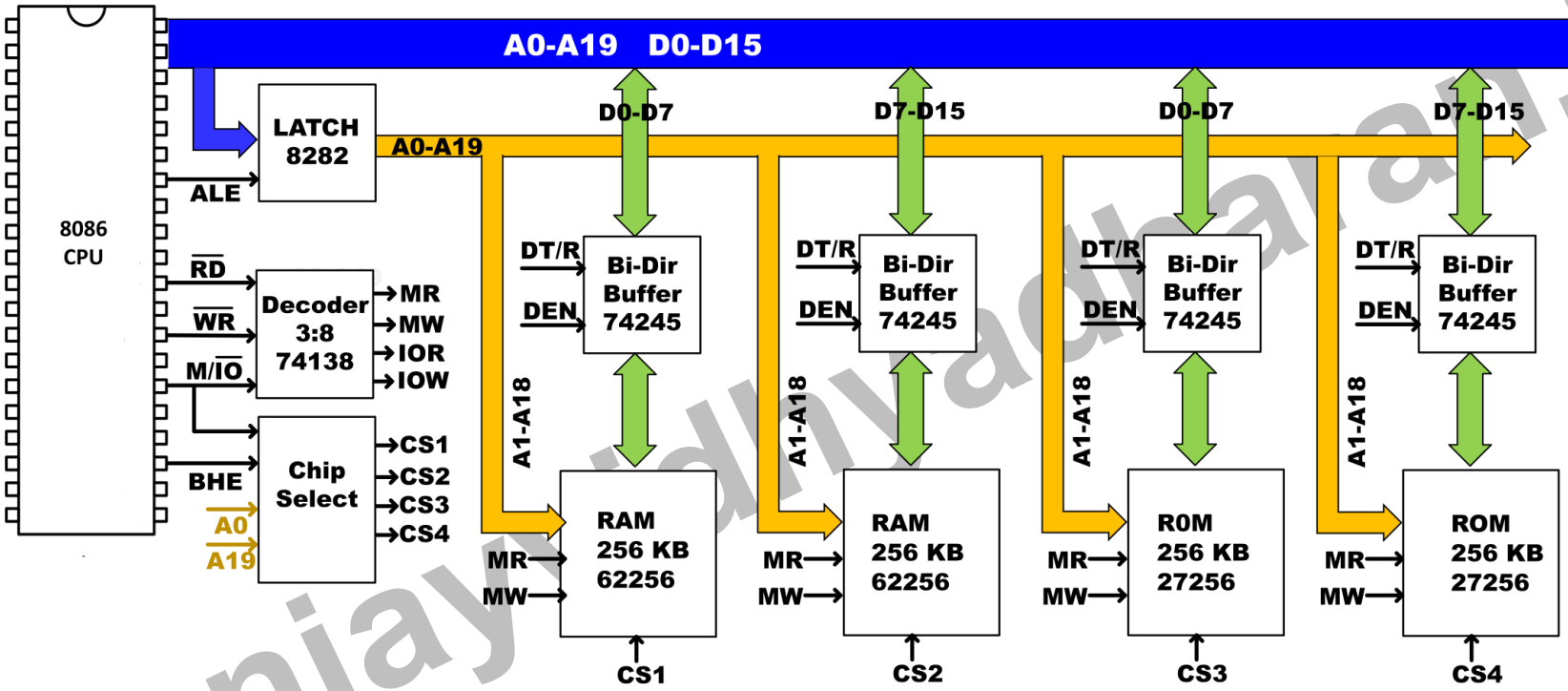


Bus Cycle

T states



8086 Memory Interface



Thankyou