Evolution of Microprocessors

- **1965:**

  Gordon Moore at Fairchild Semiconductor observes in an article for *Electronics* magazine that the number of transistors on a semiconductor chip doubles every year (download PDF). For microprocessors, it will double about every two years for more than three decades.

- **1968:**

  Moore, Robert Noyce and Andy Grove found Intel Corp. to pursue the business of "INTegrated ELectronics."

- **1969:**

  Intel announces its first product, the world's first metal oxide semiconductor (MOS) static RAM, the 1101. It signals the end of magnetic core memory.

- **1971:**

  Intel launches the world's first microprocessor, the 4-bit 4004, designed by Federico Faggin. The 2,000-transistor chip is made for a Japanese calculator, but a farsighted Intel ad calls it "a micro programmable computer on a chip.“It has 4 KB memory, 45 instructions and based on PMOS technology.

- **1972:**

  Intel announces the 8-bit 8008 processor. Teenagers Bill Gates and Paul Allen try to develop a programming language for the chip, but it is not powerful enough.

- **1974:**

  Intel introduces the 8-bit 8080 processor, with 4,500 transistors and 10 times the performance of its predecessor.

- **1975:** The 8080 chip finds its first PC application in the Altair 8800, launching the PC revolution. Gates and Allen succeed in developing the Altair Basic language, which will later become Microsoft Basic, for the 8080.

- **1976:**

  The x86 architecture suffers a setback when Steve Jobs and Steve Wozniak introduce the Apple II computer using the 8-bit 6502 processor from MOS Technology. PC maker Commodore also uses the Intel competitor's chip.

- **1978:**

  Intel introduces the 16-bit 8086 microprocessor. It will become an industry standard.
- **1979:**

  Intel introduces a lower-cost version of the 8086, the 8088, with an 8-bit bus.

- **1980:**

  Intel introduces the 8087 math co-processor.
- **1981:**

  IBM picks the Intel 8088 to power its PC. An Intel executive would later call it "the biggest win ever for Intel."

- **1982:**

  IBM signs Advanced Micro Devices as second source to Intel for 8086 and 8088 microprocessors. Intel introduces the 16-bit 80286 processor with 134,000 transistors.

- **1984:**

  IBM develops its second-generation PC, the 80286-based PC-AT. The PC-AT running MS-DOS will become the de facto PC standard for almost 10 years.

- **1985:**

  Intel exits the dynamic RAM business to focus on microprocessors, and it brings out the 80386 processor, a 32-bit chip with 275,000 transistors and the ability to run multiple programs at once.

- **1986:**

  Compaq Computer leapfrogs IBM with the introduction of an 80386-based PC.
- **1987:** VIA Technologies is founded in Fremont, Calif., to sell x86 core logic chip sets.
- **1989:**

  The 80486 is launched, with 1.2 million transistors and a built-in math co-processor. Intel predicts the development of multicore processor chips some time after 2000.
- **Late 1980s:** The complex instruction set computing (CISC) architecture of the x86 comes under fire from the rival reduced instruction set computing (RISC) architectures of the Sun Sparc, the IBM/Apple/Motorola PowerPC and the MIPS processors. Intel responds with its own RISC processor, the i860.
- **1990:**

  Compaq introduces the industry's first PC servers, running the 80486.
- **1993:**

  The 3.1 million transistor, 66-MHz Pentium processor with superscalar technology is introduced.
- **1994:**

  AMD and Compaq form an alliance to power Compaq computers with Am486 microprocessors.

- **1995**

  The Pentium Pro, a RISC slayer, debuts with radical new features that allow instructions to be anticipated and executed out of order. That, plus an extremely fast on-chip cache and dual independent buses, enable big performance gains in some applications.
- **1997:**

  Intel launches its 64-bit Epic processor technology. It also introduces the MMX Pentium for digital signal processor applications, including graphics, audio and voice processing.
- **1998:**

  Intel introduces the low-end Celeron processor.
- **1999:**

  VIA acquires Cyrix Corp. and Centaur Technology, makers of x86 processors and x87 co-processors.
- **2000:** The Pentium 4 debuts with 42 million transistors.
- **2003:** AMD introduces the x86-64, a 64-bit superset of the x86 instruction set.

- **2004:** AMD demonstrates an x86 dual-core processor chip.

- **2005:** Intel ships its first dual-core processor chip. Apple announces it will transition its Macintosh computers from PowerPCs made by Free scale (formerly Motorola) and IBM to Intel's x86 family of processors. AMD files antitrust litigation charging that Intel abuses "monopoly" to exclude and limit competition. (The case is still pending in 2008.)

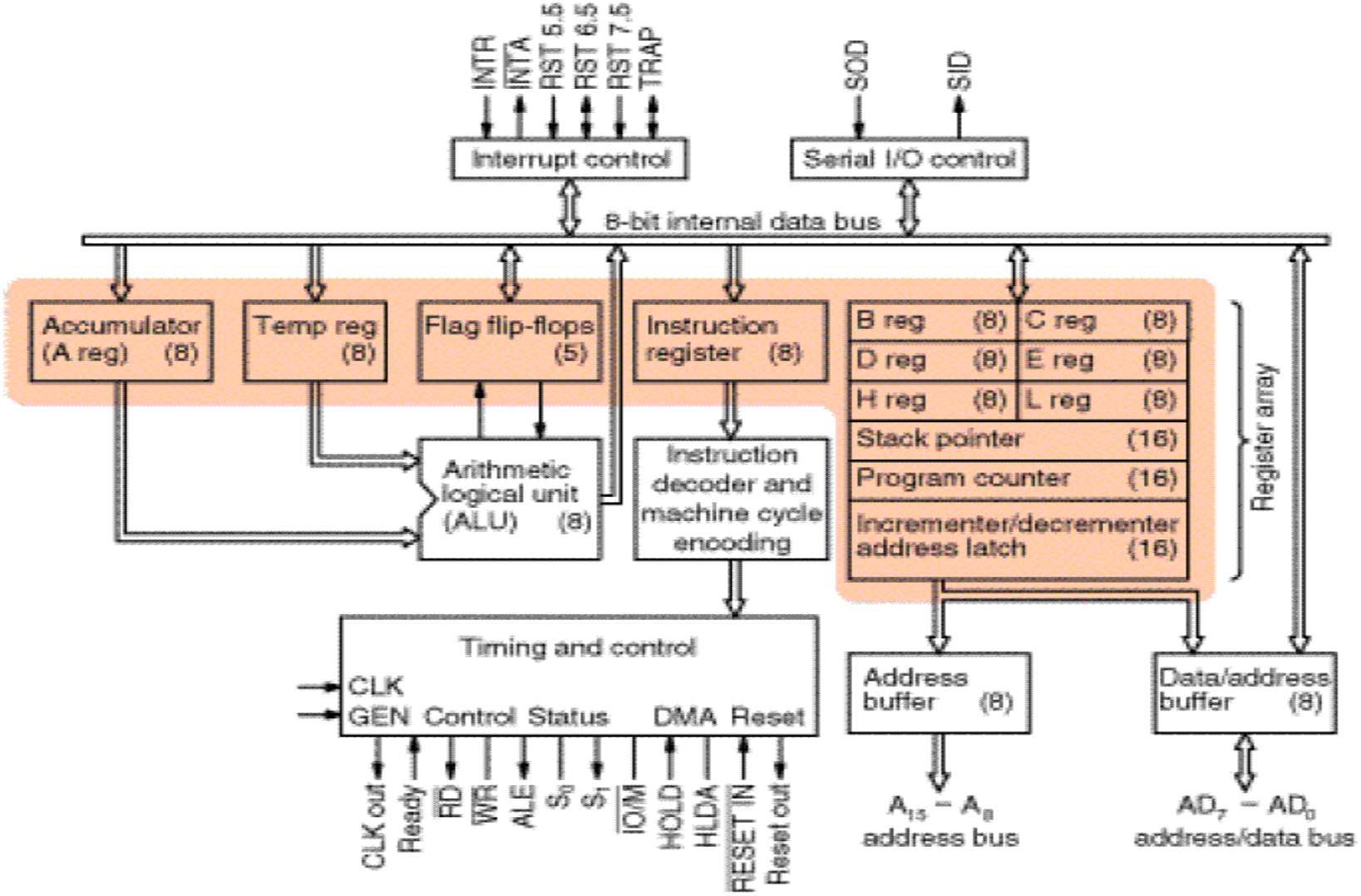
- **2006:** Dell Inc. announces it will offer AMD processor-based systems.

References

- http://www.newagepublishers.com/samplechapter/000030.pdf
- Text book: Microprocessors and its architecture by Ramesh Gaonkar
- http://www.quepublishing.com/articles/article.aspx?p=482324&seqNum=2

Architecture of 8085 Microprocessor

8085 ARCHITECTURE



Arithmetic and Logical group

Accumulator: It is 8 bit general purpose register.

- It is connected to ALU.
- So most of the operations are done in Acc.

Temporary register: It is not available for user

- All the arithmetic and logical operations are done in the temporary register but user can't access it.

Flag: It is a group of 5 flip flops used to know status of various operations done.

- The Flag Register along with Accumulator is called PSW or Program Status Word.

Arithmetic and Logical group

Flag Register is given by:

S	Z	X	AC	X	P	X	CY

S:Sign flag is set when result of an operation is negative.

Z:Zero flag is set when result of an operation is 0.

Ac:Auxiliary carry flag is set when there is a carry out of lower nibble or lower four bits of the operation.

CY:Carry flag is set when there is carry generated by an operation.

P:Parity flag is set when result contains even number of 1's.

Rest are don't care flip flops.

Register Group

- Temporary registers (W,Z):These are not available for user. These are loaded only when there is an operation being performed.
- General purpose:There are six general purpose registers in 8085 namely B,C,D,E,H,L.These are used for various data manipulations.
- Special purpose :There are two special purpose registers in 8085:

1. SP :Stack Pointer.
2. PC:Program Counter.

Register Group

Stack Pointer: This is a temporary storage memory 16 bit register. Since there are only 6 general purpose registers, there is a need to reuse them .

- Whenever stack is to be used previous values are PUSHED on stack and then after the program is over these values are POPED back.

Program Counter: It is 16 bit register used to point the location from which the next instruction is to be fetched.

- When a single byte instruction is executed PC is automatically incremented by 1.
- Upon reset PC contents are set to 0000H and next instruction is fetched onwards.

INSTRUCTION REGISTER,DECODER & CONTROL

- Instruction register:When an instruction is fetched , it is executed in instruction register.This register takes the Opcode value only.
- Instruction decoder: It decodes the instruction from instruction register and then to control block.
- Timing and control:This is the control section of μP.It accepts clock input .

INTERRUPT CONTROL

- It accepts different interrupts like TRAP INT5.5,6.5,7.5and INTR.

SERIAL IO CONTROL GROUP

- It is used to accept the serial 1 bit data by using SID and SOD signals and it can be performed by using SIM & RIM instructions.

Addressing Mode of Microprocessor 8085

ADDRESSING MODES OF 8085

Immediate addressing:

Immediate data is transferred to address or register.

Example:

MVI A,20H. Transfer immediate data 20H to accumulator.

Number of bytes:

Either 2 or 3 bytes long.

1st byte is opcode.

2nd byte 8 bit data .

3rd byte higher byte data of 16 bytes.

ADDRESSING MODES OF 8085

Register addressing:

Data is transferred from one register to other.

Example:

MOV A, C :Transfer data from C register to accumulator.

Number of bytes:

Only 1 byte long.

One byte is opcode.

ADDRESSING MODES OF 8085

Direct addressing:

- Data is transferred from direct address to other register or vice-versa.

Example:

LDA C200H .Transfer contents from C200H to Acc.

Number of bytes:

These are 3 bytes long.

1st byte is opcode.

2nd byte lower address.

3rd byte higher address.

ADDRESSING MODES OF 8085

Indirect addressing:

- Data is transferred from address pointed by the data in a register to other register or vice-versa.

Example:

MOV A, M: Move contents from address pointed by M to Acc.

Number of bytes:

These are 3 bytes long.

1st byte is opcode.

2nd byte lower address.

3rd byte higher address.

ADDRESSING MODES OF 8085

Implied addressing:

- These doesn’t require any operand. The data is specified in Opcode itself.

Example: RAL: Rotate left with carry.

No.of Bytes:

These are single byte instruction or Opcode only.

Interrupts of Microprocessor 8085

INTERRUPTS IN 8085

- Interrupt is a process where an external device can get the attention of the microprocessor.

  The process starts from the I/O device

  The process is asynchronous.

- Classification of Interrupts

  Interrupts can be classified into two types:

  - Maskable Interrupts (Can be delayed or Rejected)
  - Non-Maskable Interrupts (Can not be delayed or Rejected)

INTERRUPTS IN 8085

Interrupts can also be classified into:

- Vectored (the address of the service routine is hard-wired)
- Non-vectored (the address of the service routine needs to be supplied externally by the device)

- An interrupt is considered to be an emergency signal that may be serviced.
  - The Microprocessor may respond to it as soon as possible.

INTERRUPTS IN 8085

- The 8085 has 5 interrupt inputs.
- The INTR input.

  The INTR input is the only non-vectored interrupt.

  INTR is mask-able using the EI/DI instruction pair.

  RST 5.5, RST 6.5, RST 7.5 are all automatically vectored.
  - RST 5.5, RST 6.5, and RST 7.5 are all mask-able.

  TRAP is the only non-mask-able interrupt in the 8085
  - TRAP is also automatically vectored.

INTERRUPTS IN 8085

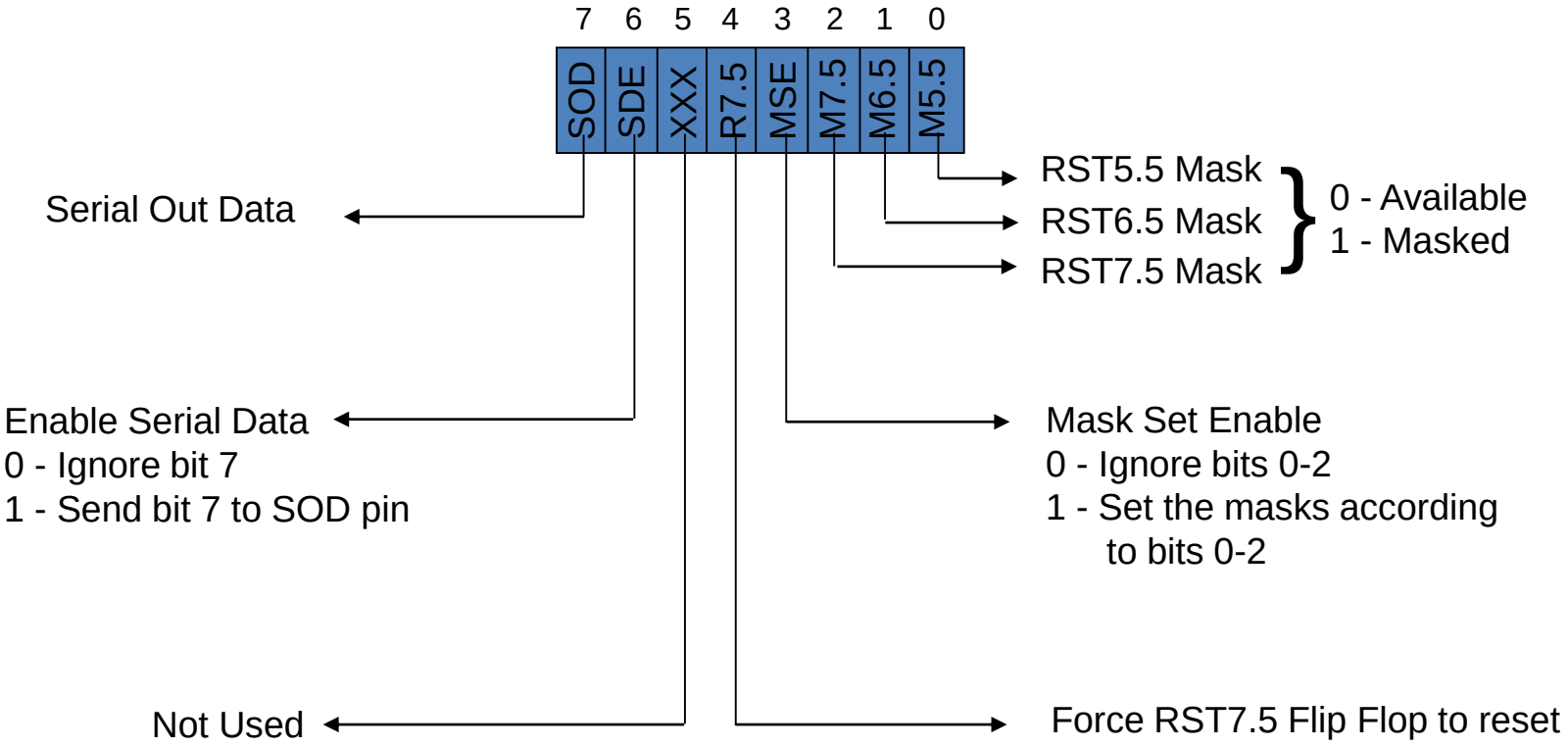
- Non vectored interrupts:
- The 8085 recognizes 8 RESTART instructions: RST0 - RST7 . Each of these would send the execution to a predetermined hard-wired memory location:

Restart Instruction	Equivalent to
RST0	CALL 0000H
RST1	CALL 0008H
RST2	CALL 0010H
RST3	CALL 0018H
RST4	CALL 0020H
RST5	CALL 0028H
RST6	CALL 0030H
RST7	CALL 0038H

INTERRUPT PRIORITY

Interrupt name	Mask-able	Vectored
TRAP	No	Yes
RST 7.5	Yes	Yes
RST 6.5	Yes	Yes
RST 5.5	Yes	Yes
INTR	YES	NO

SIM INSTRUCTION



- SIM Instruction helps activate a particular interrupt.
- It can also mask a maskable interrupt.

SIM INSTRUCTION

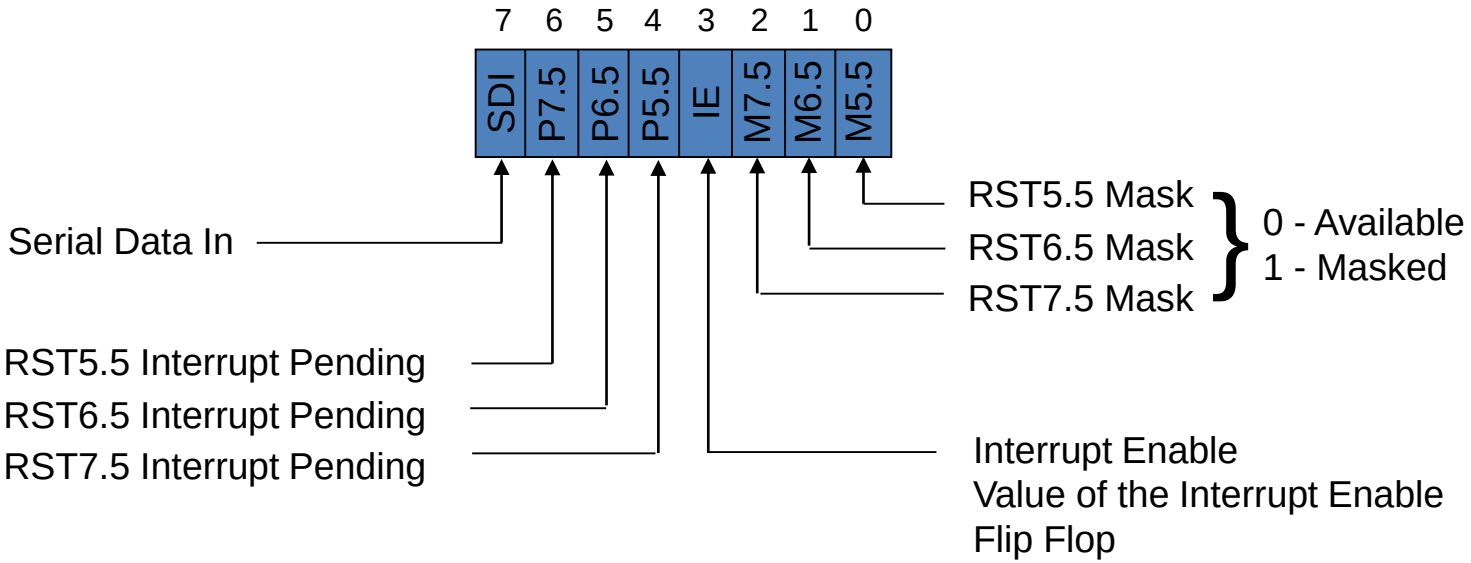
- Example: Set the interrupt masks so that RST5.5 is enabled, RST6.5 is masked, and RST7.5 is enabled.
- First, determine the contents of the accumulator.

- Enable 5.5	bit 0 = 0
- Disable 6.5	bit 1 = 1
- Enable 7.5	bit 2 = 0
- Allow setting the masks	bit 3 = 1
- Don't reset the flip flop	bit 4 = 0
- Bit 5 is not used	bit 5 = 0
- Don't use serial data	bit 6 = 0
- Serial data is ignored	bit 7 = 0

SDO	SDE	XXX	R7.5	MSE	M7.5	M6.5	M5.5
0	0	0	0	1	0	1	0

```
EI          ; Enable interrupts including INTR
MVI A, 0A   ; Prepare the mask to enable RST 7.5, and 5.5, disable 6.5
SIM         ; Apply the settings RST masks
```

RIM INSTRUCTION



- Since the 8085 has five interrupt lines, interrupts may occur during an ISR and remain pending.
- Using the RIM instruction, it is possible to can read the status of the interrupt lines and find if there are any pending interrupts.

Memory Interfacing of Microprocessor 8085

8085 Memory Interfacing

- Generally µP 8085 can address 64 kB of memory .
- Generally EPROMS are used as program memory and RAM as data memory.
- We can interface Multiple RAMs and EPROMS to single µP .
- Memory interfacing includes 3 steps :

1. Select the chip.
2. Identify register.
3. Enable appropriate buffer.

8085 Memory Interfacing

- Example: Interface 2Kbytes of Memory to 8085 with starting address 8000H.

Initially we realize that 2K memory requires 11 address lines (2^11=2048). So we use A_0-A_{10} .

- Write down A_{15} –A_0

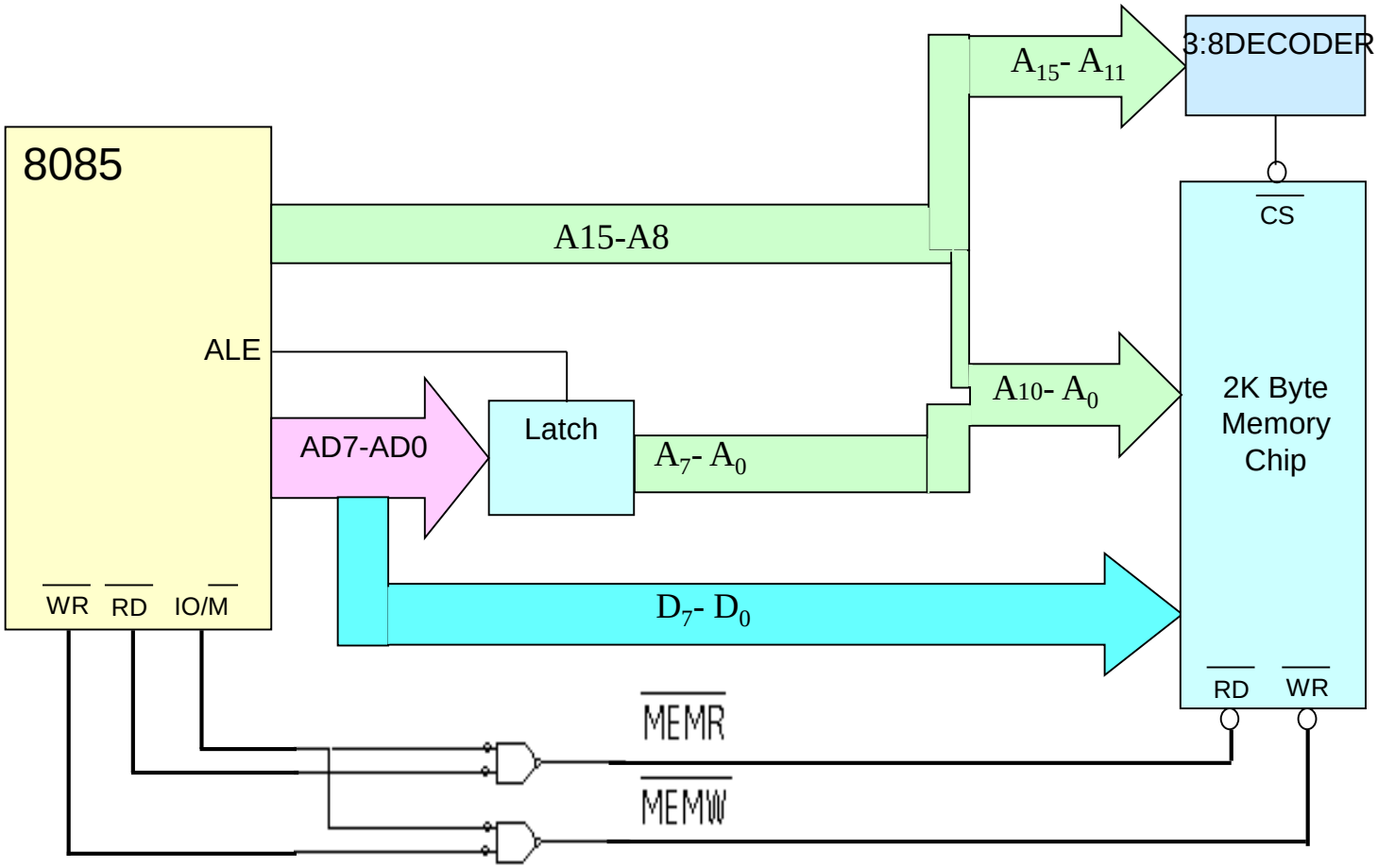
A_{15}	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	ADD
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8000H
1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	87FFH

8085 Memory Interfacing

- Address lines A_0-A_{10} are used to interface memory while A_{11},A_{12},A_{13},A_{14},A_{15} are given to 3:8 Decoder to provide an output signal used to select the memory chip $\overline{CS}$ or Chip select input.
- $\overline{MEMR}$ and $\overline{MEMW}$ are given to $\overline{RD}$ and $\overline{WR}$ pins of Memory chip.
- Data lines D_0-D_7 are given to D_0-D_7 pins of the memory chip.
- In this way memory interfacing can be achieved.

8085 Memory Interfacing

- The diagram of 2k interfacing is shown below:



8085 Memory Interfacing

•In this example we saw that some address lines are used for interfacing while others are for decoding.

•It is called absolute decoding.

•We sometimes don’t requires that many address lines.So we ignore them.But this may lead to shadowing or multiple address.

•This type of decoding is called linear decoding or partial decoding.

•In partial decoding wastage of address takes place but it requires less hardware and cost is also less as compared with absolute one.

Programming of Microprocessor 8085

PROGRAM

- Write a program to transfer a block of data from C550H to C55FH. Store the data from C570H to C57FH .

```
    LXI H ,C550H
    LXI B ,C570H
    MVI D,0FH
UP  MOV A,M
    STAX B
    INX H
    INX B
    DCR D
    JNZ  UP
     RST1
```

PROGRAM

- Find out errors in the following :
- MVI B,D =Immediate addressing doesn't have register as operand .Therefore, MVI B,80H.
- INX L=Increment operator always acts on the higher memory address in register pair .Thus ,INX H.
- JP 80H = Conditional jump instructions doesn't have any immediate operand .Thus, JP UP.

If Flag contents are AB H, what is flag status

If flag contains AB H then it's values from D_7 to D_0 are

10101011.

By comparing it with flag register we get S=1,Z=0,AC=0,

P=0,Cy=1.

PROGRAM

11. What are the instructions for the following actions?

- Load the PC with second and third byte of instruction.

LXI H, C200H

PCHL Load PC with HL content

Thus PC= L,PC +1=H.

- No change in normal execution except increment the PC.

NOP (No operation)

- This instruction has no effect on code only used to cause delay .

PROGRAM

Write a program to add 10 data bytes. Data is stored from locations C200. Store result at C300H.

```
      LXI H,C200 H
      MVI C, 0A H
UP    MVI A,00 H
      MOV B,M
      ADD B
      INX H
      DCR C
      JNZ UP
      STA C300H
      RST1.
```

Timing and State Diagram of Microprocessor 8085

TIMING AND STATE DIAGRAM

- The µP operates with reference to clock signal.The rise and fall of the pulse of the clock gives one clock cycle.
- Each clock cycle is called a T state and a collection of several T states gives a machine cycle.
- Important machine cycles are :

1. Op-code fetch.
2. Memory read.
3. Memory write.
4. I/Op-read.
5. I/O write.

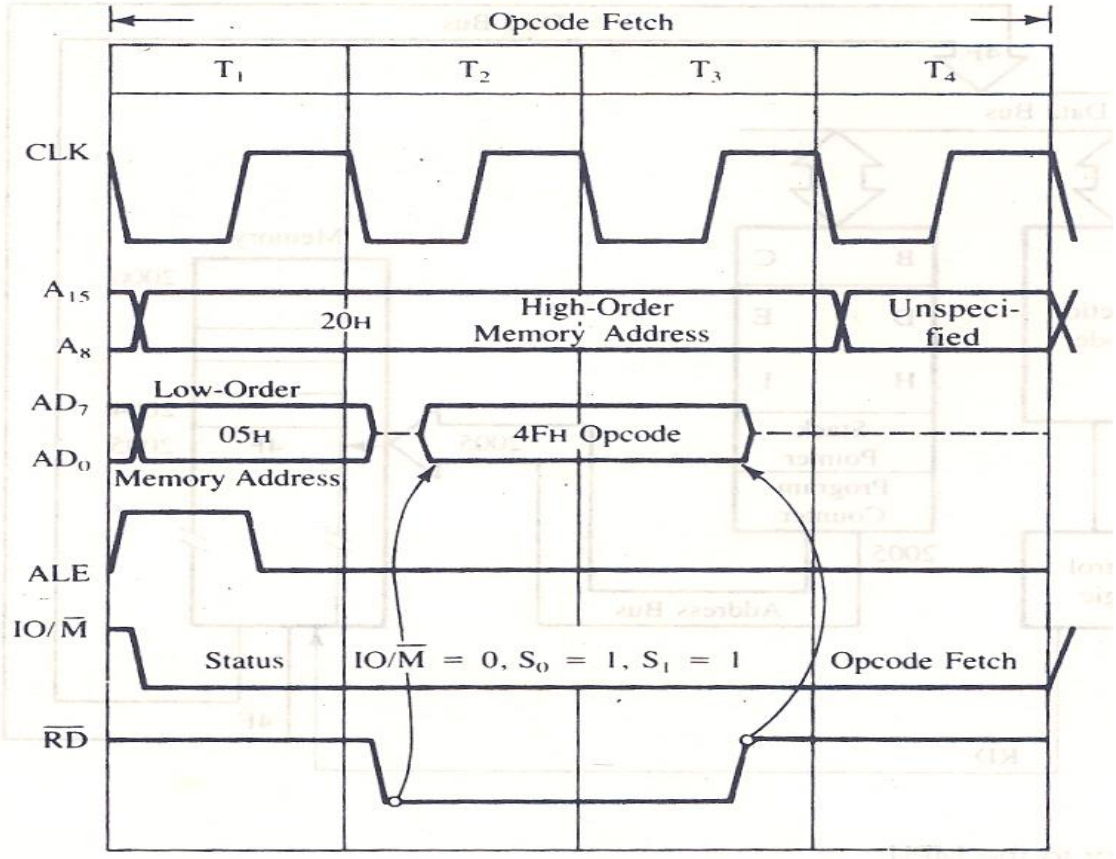
TIMING AND STATE DIAGRAM

Op-code Fetch:It basically requires 4 T states from T_1-T_4

- The ALE pin goes high at first T state always.
- AD_0-AD_7 are used to fetch OP-code and store the lower byte of Program Counter.
- A_8-A_{15} store the higher byte of the Program Counter while IO/M^- will be low since it is memory related operation.
- RD^- will only be low at the Op-code fetching time.
- WR^- will be at HIGH level since no write operation is done.
- S_0=1,S_1=1 for Op-code fetch cycle.

TIMING AND STATE DIAGRAM

Op-code fetch cycle :



TIMING AND STATE DIAGRAM

Memory Read Cycle: It basically requires 3T states from T_1-T_3 .

- The ALE pin goes high at first T state always.
- AD_0-AD_7 are used to fetch data from memory and store the lower byte of address.
- A_8-A_{15} store the higher byte of the address while IO/M^- will be low since it is memory related operation.
- RD^- will only be low at the data fetching time.
- WR^- will be at HIGH level since no write operation is done.
- S_0=0,S_1=1 for Memory read cycle.

TIMING AND STATE DIAGRAM

Memory write Cycle: It basically requires 3T states from T_1-T_3 .

- The ALE pin goes high at first T state always.
- AD_0-AD_7 are used to fetch data from CPU and store the lower byte of address.
- A_8-A_{15} store the higher byte of the address while IO/M^- will be low since it is memory related operation.
- RD^- will be HIGH since no read operation is done.
- WR^- will be at LOW level only when data fetching is done.
- S_0=1,S_1=0 for Memory write cycle.

SUBROUTINE

Calculation of Delay using 8 bit counter:

- Consider following example:

MVI C, count(8 bit) H	7 T states
UP DCR C	4 T states
JNZ UP	10/7 T
RET	10T

- Here loop UP is executed (N-1) times.
- Thus delay is

Td=M+[(count)x N) -3.

- Where M= no.of T states outside loop.

  N=no.of T states inside loop.

SUBROUTINE

- Here value of M= 17, N= 14.
- The maximum delay will occur if count is 255 or FF H.
- Thus Td max =17+[255x14]-3= 3584 T states.
- For 0.5 µsec delay for a T state, we get
- Td max=0.5 µsec x 3584= 1792 µsec or 1.792 m sec.