# Microprocessors

## Microprocessor Basics

1. What are the differences between microprocessors and microcontrollers?
2. For each of the following devices, would you use a microcontroller or a microprocessor?
    a. Keyboard
    b. Mouse
    c. Headphone
    d. Computer
    e. TV Remote
    f. Smart TV
    g. Regular fridge
    h. Mobile phone
3. Draw the basic block diagram of a microprocessor based system and briefly describe each component.
4. Briefly describe which component of a microprocessor based system is used for accessing data from a harddisk?
5. Briefly describe the 3 steps involved in executing instructions inside a microprocessor
6. What are the differences between RAM and ROM?
7. For each of the following, mention which memory device is used to store the information:
    a. Storing movies/pictures on a computer
    b. Viewing movies/pictures on a computer
    c. Computer BIOS
8. What are the 3 types of bus inside a microprocessor? Briefly describe the purpose of each bus.
9. ATMega328p is an 8 bit microcontroller. It has 32KB on-board flash memory (ROM)
    a. What is the data bus size of the microcontroller?
    b. What is the minimum address bus required for accessing the on-board flash memory?
10. Calculate the maximum supported RAM size for the following address bus lengths:
    a. 16 bit  b. 20 bit  c. 24 bit  d. 32 bit  e. 64 bit
11. What is an assembler? Explain in short.

# Microprocessor Architecture

1. What are the two functional units of the 8086 microprocessor? Explain why we need the two units.
2. What does the Bus Interface Unit do?
3. What does the Execution Unit do?
4. Draw the 16-bit 8086 processor architecture
5. Which functional unit does the following register belong to?
    a. Segment Register
    b. Instruction Pointer
    c. Index Register
    d. General Purpose Register
    e. Instruction Queue
    f. Flags
6. What is the advantage of an instruction queue in a processor?
7. With example, show how instruction queuing can speed up time required for processes in the 8086 microprocessor.
8. What is the purpose of segment registers in the 8086 microprocessor?
9. What is the role of flag register in the 8086 microprocessor

# Memory Partitioning and Segmentation

1. What is the bit length of the internal registers of the 8086 processor?
2. Why cannot the 8086 processor access the full memory without memory partitioning?
3. What do you understand by the term “Memory Segmentation”?
4. What are the advantages of overlapping segmentation compared to non-overlapping segmentation?
5. In the 8086 processor, what is the minimum and maximum segment size with the overlapping segmentation scheme?
6. What is the minimum and maximum number of segments supported by the 8086 microprocessor?
7. Assume the segments are non-overlapping. Find which segment the following physical addresses belong to in the 8086 processor. Then for each of those segments, find the lowest and highest addresses.
    a. *12345h*
    b. *10h*
    c. *10010h*
    d. *FFFFEh*
8. Explain why *8AB3Fh* cannot be the starting physical address of an 8086 memory segment.
9. Find the largest and smallest possible segment address for each of the following physical addresses. Also, mention the logical addresses for each case. (Maximum segment size can be 64KBytes)
    a. *12345h*
    b. *10h*
    c. *10010h*
    d. *FFFFEh*
10. A memory location has a physical address of 9A7B1 H. Determine the offset address if the segment number is 40FF H.
11. You are given the following instruction >>> **ADD AX, [10h].** You are provided the following data: **DS = AB12h; SS = 2567h; CS= 29C1h.** Find the effective address location for the given instruction.
12. Find the smallest and the second largest segment address for each of the following physical addresses. Also, mention the logical addresses (**segment: offset pair**) for each case:
    a. FFFEFh
    b. 2h
13.

| Register | DS | SS | CS | DI | SI | BX | SP |
|---|---|---|---|---|---|---|---|
| Data | 1000h | 5000h | 6730h | 1500h | 2000h | 4000h | 0900h |

| Address | 12370h | 2370h | 12371h | 2371h | 1710h | 1711h | 11710h | 11711h |
|---|---|---|---|---|---|---|---|---|
| Data | 12h | 56h | 34h | 78h | 89h | ABh | 43h | 21h |

The following instructions have been executed.

- MOV AH, [SI + 370h]
- MOV AL, [SI + 371h]
- MOV BH, [DI + 210h]
- MOV BL, [DI + 211h]
- ADD AX, BX

a. Calculate the values of AH and AL respectively after you execute instructions (i) and (ii).
b. Calculate the values of BH and BL respectively after you execute instructions (iii) and (iv).

14.

| Address | 10600h | 10601h | 20600h | 20601h | 30600h | 30601h |
|---|---|---|---|---|---|---|
| Data | 12h | 34h | 56h | 78h | 10h | 20h |

Given DS = 1000h, SS = 2000h, CS = 3000h, BP = 0500h, SI = 0100h. Now **deduce** what data will be stored in BX if the instruction MOV BX, [BP + SI] is executed.

15.

| Address | 10600h | 10601h | 20600h | 20601h | 30600h | 30601h |
|---|---|---|---|---|---|---|
| Data | 12h | 34h | 56h | 78h | 10h | 20h |

Given DS = 2000h, SS = 1000h, CS = 3000h, BP = 0400h, DI = 0200h. Now **deduce** what data will be stored in BX if the instruction MOV BX, [BP + SI] is executed.

16. Instead of 64KB, assume the size of each segment for an 8086 is 256 bytes. In that case, deduce the 2nd last address of a segment whose starting address is 10000h.

17. Instead of 64KB, assume the size of each segment for an 8086 is 4KB. In that case, deduce the 2nd last address of a segment whose starting address is 10000h.

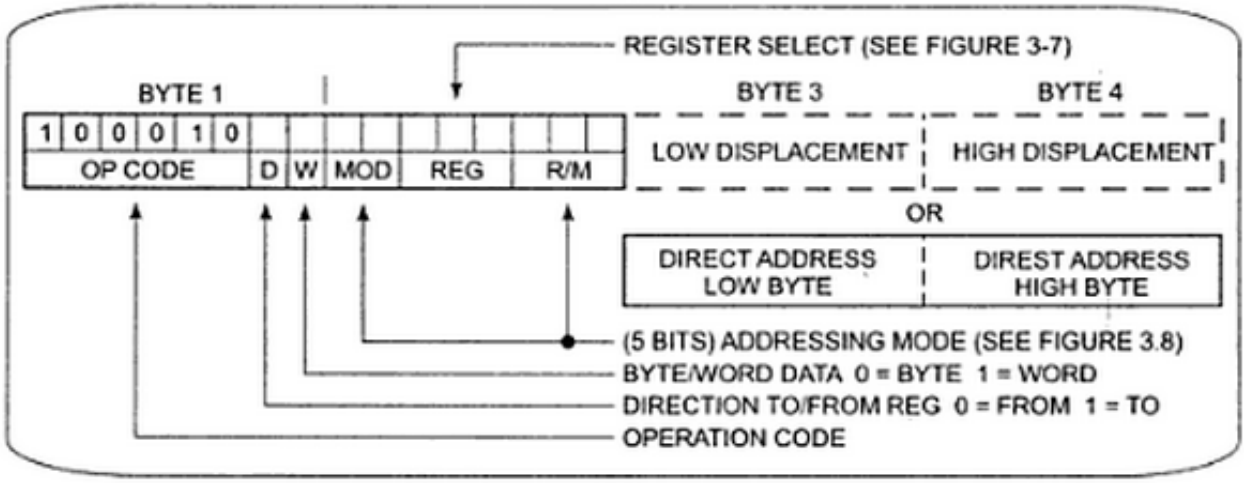
# Flag Register

1. Write down the purpose of each of the following flag bit and give on example of its use:
    a. Carry flag
    b. Zero flag
    c. Parity flag
    d. Sign flag
    e. Overflow flag
    f. Auxiliary flag
    g. Trap flag
    h. Interrupt flag
    i. Direction flag
2. Suppose two hexadecimal numbers (i) **96C9** and (ii) **99XY** are to be added by an Intel 8086 microprocessor. Here X and Y represent two unknown hexadecimal digits.
    a. After the 8086 **adds** the numbers (i) and (ii), deduce the minimum values of X and Y needed to get the value of AF = 1
    b. Using the deduced values of X and Y obtained from (a), find the values of PF, SF, OF and CF for the given **addition** operation.
3. Calculate the OF, PF, AF, SF, CF after the execution of the given instructions. Explain your reasoning for deducing the flag values.:

    ```
    >>> MOV AX, 0AB56h
    >>> MOV BX, 3965h
    >>> ADD AX, BX
    ```
4. Calculate the OF, PF, AF, SF, CF after the execution of the given instructions. Explain your reasoning for deducing the flag values.:

    ```
    >>> MOV AX, 7EBFh
    >>> MOV BX, 4A52h
    >>> ADD AX, BX
    ```

# Addressing Modes and Machine Code




| MOD / RM | 00 | 01 | 10 | 11 | |
|---|---|---|---|---|---|
| | | | | W = 0 | W = 1 |
| 000 | [BX] + [SI] | [BX] + [SI] + d8 | [BX] + [SI] + d16 | AL | AX |
| 001 | [BX] + [DI] | [BX] + [DI] + d8 | [BX] + [DI] + d16 | CL | CX |
| 010 | [BP] + [SI] | [BP] + [SI] + d8 | [BP] + [SI] + d16 | DL | DX |
| 011 | [BP] + [DI] | [BP] + [DI] + d8 | [BP] + [DI] + d16 | BL | BX |
| 100 | [SI] | [SI] + d8 | [SI] +d16 | AH | SP |
| 101 | [DI] | [DI] + d8 | [DI] +d16 | CH | BP |
| 110 | d16 (direct address) | [BP] + d8 | [BP] +d16 | DH | SI |
| 111 | [BX] | [BX] + d8 | [BX] + d16 | BH | DI |

1. Suppose you are given the following machine code for an instruction: **88160080h.** Deduce the original assembly language instruction denoted by the above mentioned machine code.
2. Consider the instruction >>> ***MOV C5A4h, CX***
   Deduce the corresponding machine code
3. Consider the instruction >>> ***MOV [AX], [BX+SI+23FFh]***
   What will be the length of the corresponding machine code in bytes?
4. Convert **89806910h** from machine language to its corresponding assembly language. Show all of your work.
5. Suppose the instruction **MOV DI, [BP+42h]**, appears in a program. What is its machine language for the given instruction?
6. Given DS = **1000h**, CS = **3000h**, SS = **8A40h**, SI = **2000h**, CX = **43AEh**, DI = **030Fh**. We also execute the **JMP [ SI + 1000h ]** instruction. Now, answer the questions based on the given table and data:

a. **Explain** to what addressing mode the instruction **IN 05h, DX** belongs.

b. **Deduce** the physical address of the instruction that 8086 will jump to.

c. "**If we are using overlapping segmentation, the physical address "00015h" can be a part of 3 different segments**". **Explain** with logic any arguments you have for or against this statement.

| Physical Address | 8 bit hex data |
|---|---|
| 14001h | 78h |
| 14000h | 56h |
| 13001h | 34h |
| 13000h | 12h |
| 03000h | ABh |

7. Suppose you are given the following values; DS = 1234h, CS = 2345h, SS = 3456h, BX = 0010h, DI = 0021h. Assume we are using overlapping segmentation here. Additionally, you are also given two instructions:

   Instruction 01 >>> **MOV AX, [BP]**
   Instruction 02 >>> **MOV [BX+DI+1019H], AX**

   Now based on these values answer the following questions:
   a. Explain with proper reasoning to which addressing mode does Instruction 01 belong.
   b. Calculate what physical address is the destination of Instruction 02 referring to? Your final answer should be in hexadecimal.
   c. Using the physical address obtained from "b", identify the largest segment the obtained physical address can be a part of and deduce the corresponding logical address.
8. The OPCode for MOV = 100010. Deduce the machine code (in hexadecimal) for the following instruction:

   >>> MOV [BX+DI+46DCh], SI

9. For each of the following statements, state the addressing modes.

   (i) MOV BX, 1000H

   (ii) MOV DI, 1008H

   (iii) MOV AX, B [BX+SI+2]; B is an array

   (iv) MOV AX, B [BX]

10. Now consider the following instruction " MOV J, [ BX + K + L ] " whose equivalent Machine code is XY A8 86 02 h. Here XY is an unknown 8-bit hex value. Additionally, J represents a specific general purpose register and K is a specific index register. The given instruction basically moves data from the address 474D2 h and copies it to the "J" register.

| CS | DS | BX | SI | DI | IP |
|---|---|---|---|---|---|
| 286A h | 458E h | 0A2E h | N/A | N/A | N/A |

a. Deduce what are the J and K registers. Give reasons behind your answer.
b. Deduce the values of L and XY
c. Deduce the value that resides inside the K register.

11.

a. Convert the following machine codes to their corresponding assembly language instructions:
   - 8BD0
   - 8BDA
b. If, before the execution of the above lines of code, the values of the general-purpose registers were AX = 7, BX = 6, CX = 5 and DX = 3. Calculate the values of these mentioned registers after the codes you found in part a are executed.

12. The following information is given for a program in the 8086 microprocessor.
CS = 8000h, IP = 0000h. Consider the following 2 lines of assembly language program
>>> MOV AX, [BX]
>>> MOV 1234h[SI], AX

a. **Explain** how many bytes are required in total to store the two instructions in memory.
b. **Find** the machine code (in hexadecimal) for the two instructions and **fill** the following table to indicate which byte is stored in which memory location of the RAM. Leave the unused memory spaces empty. (Hint: A memory location in RAM can store a single byte of information)

| Logical Address | Content (Hex) |
|---|---|
| 8000:0000 | |
| 8000:0001 | |
| 8000:0002 | |
| 8000:0003 | |
| 8000:0004 | |
| 8000:0005 | |
| 8000:0006 | |

13. Explain to which addressing mode does the instruction “HLT AX” belong

14. Explain to which addressing mode does the instruction “RET AX” belong

15.

| Address | 589B6h | 589B7h | 589B8h | 650FBh | 650FCh | 650FDh |
|---|---|---|---|---|---|---|
| Data | 15h | 74h | 36h | C1h | DAh | FEh |

Assume DS = 5555h, SS = 3200h, CS = 2000h, SI = 3467h, BP = 2766h, BX = FBACh.

**a.** Based on the above-given registers and table, deduce which data will be accessed by executing the instruction CALL [BX].
**b.** Explain with reasoning CALL [BX] falls under which addressing mode.
**c.** Deduce the hex machine code of the instruction MOV [BP], BL.

16.

| Address | 576BDh | 576BEh | 576BFh | 5A7B9h | 5A7BAh | 5A7BBh |
|---|---|---|---|---|---|---|
| Data | 75h | 42h | BAh | F2h | 4Eh | AEh |

Assume CS = 6666h, SS = A200h, DS = 4AC0h, SI = FBBAh, BP = 8877h, BX = CABDh.

**a.** Based on the above given registers and table, deduce which data will be accessed by executing the instruction CALL [BX].
**b.** Explain with reasoning CALL [BX] falls under which addressing mode.
**c.** Deduce the hex machine code of the instruction MOV AX, [BP+2000h].

# Memory Banking

1. For the instruction, MOV ‘Destination Register’, [117X h], the $A_0$ and $\overline{BHE}$ values were both found to be 0. Assuming the memory is divided into even and odd banks, answer the following:
    a. Find a possible value of X. Explain why you chose this value.
    b. Suggest a possible general-purpose register as a Destination for the MOV instruction above. Explain whether it’ll be an 8-bit or a 16-bit register.
2. Assuming the memory is divided into even and odd banks, an instruction requires one bus cycle for which $A_0$ = 1 and $\overline{BHE}$ = 0. If the instruction is a MOV instruction from a memory to a destination register,
    a. Deduce a possible value of the memory location.
    b. Deduce a possible general-purpose register as a Destination for the MOV instruction above.
3. An 8086 cpu has a RAM of 1MB split into two equal memory banks. The $A_0$ and $\overline{BHE}$ pins of the cpu are used to access the two memory banks. Assuming DS=0000h, consider the following instructions-
    a. MOV AH, [8086h]
    b. MOV AL, [8086h]
    c. MOV AX, [8086h]
    d. MOV BH, [8085h]
    e. MOV BL, [8085h]
    f. MOV BX, [8085h]

    For each of the given instructions, determine the following information:
    a. Size of the data being transferred (byte/word)
    b. Which portion of the data bus was used to transfer the data from source location to destination register
    c. Value of $A_0$ and $\overline{BHE}$ required to make the transfer
    d. Memory bank from which the data is fetched
    e. Total number of data cycles used to move data
    f. The internal address of the selected memory bank from which the data is being transferred
4. Why is memory banking done in the 8086 system? What would happen if memory banking is not implemented?
5. What do you understand by aligned word and unaligned word? Explain with an example.

# Pin Specification and Timing Diagram

1. Suppose you have an Intel 8086 which is operating at a Duty Cycle of 60% and for each clock pulse assume $T_{off}$ = 40ns. The 8086 is now going to execute the instruction MOV [1235h], AX. Based on this, answer the following questions:
    a. Estimate the frequency at which the 8086 is operating.
    b. Calculate the total time for one Instruction Cycle of the given instruction.
    c. Calculate the values of the $A_0$ and BHE' pins during the execution of the given instruction.

2. The diagram below shows 1 MACHINE/BUS CYCLE of an 8086 microprocessor operating at a particular frequency.


    a. Calculate the percentage of Duty Cycle of the waveform. Additionally, calculate the frequency (in Mhz) at which the 8086 is operating if each clock pulse is 'ON' for exactly 55.55ns.
    b. During a READ cycle, suppose an 8086 is operating at the frequency you got from (I) and is now reading from a Memory that takes 300ns to upload data on the data bus. Explain with reasoning, whether the microprocessor will insert a Wait State (Tw) or not.

3. Suppose an 8086 is operating in a way such that $T_{ON}$ is 1/4th of the total time required for one clock pulse. Consider $T_{ON}$ is 30ns. Now the 8086 is going to execute the instruction MOV AX, [2315h] i.e. 16 bits of data will be read from memory.
    a. Calculate the frequency in MHz at which the 8086 is operating.
    b. Calculate the time required for 1 Machine / Bus Cycle.
    c. Deduce the total time required to execute the given instruction MOV AX, [2315h].
    d. Explain with proper reasoning the values of pins $A_0$ and BHE' during the execution of the given instruction MOV AX, [2315h].
    e. Suppose from the start of the T3 clock pulse, the memory took 240 ns to finalize uploading data on the data bus. Explain with reason whether the processor will exert a Wait State or not.

4. A Microprocessor has a duty cycle of 40%.
    a. Calculate the frequency (in Mhz) at which the 8086 operates if each clock pulse is 'ON' for exactly 80 ns.
    b. Construct the timing diagram of a Write cycle.

5. For the following instructions, mention the states of $\overline{RD}$, $\overline{WR}$, $M/\overline{IO}$, $\overline{BHE}$ pins during the T2 cycle.
    a. MOV AL, [34h]
    b. MOV [33h], BL
    c. OUT 82h, AL
    d. IN AL, 82h

6. a. Suppose the time required to complete 1 instruction cycle for the instruction MOV AX, [1235h] is 200ns. And for each bus cycle TON = 0.2T where T represents the time required for 1 clock pulse. Now deduce the values of TON, TOFF, and the frequency at which this 8086 is operating.

b. Deduce the values of the Ao and BHE pins during 1 instruction cycle of the above given mov operation.

c. Illustrate the timing diagram for the given mov operation showcasing only the CLK, M/IO', and RD'/WR' pins [use either RD' or WR' depending on the operation].

7. a. Suppose the time required to complete 1 instruction cycle for the instruction MOV [1235h], AX is 400ns. And for each bus cycle TON = 0.4T where T represents the time required for 1 clock pulse. Now deduce the values of TON, TOFF, and the frequency at which this 8086 is operating.

b. Deduce the values of the Ao and BHE pins during 1 instruction cycle of the above given mov operation.

c. Illustrate the timing diagram for the given mov operation showcasing only the CLK, M/IO', and RD'/WR' pins [use either RD' or WR' depending on the operation].

# Interrupts

1. Suppose you have to design a new Priority Interrupt Controller (PIC) named “X”. It should be able to handle 10 different interrupt requests at a certain time and should be able to address 16,384 types of interrupts. But keep in mind that the process of handling interrupt requests and priority handling would be the same as 8259A PIC.

a. Explain in brief how an 8259A PIC co-operates with the 8086 microprocessor to handle interrupts from multiple devices with just one interrupt request pin.
b. Deduce with reasoning how many data bus lines of the PIC “X” would be connected to the 8086 data bus?
c. In cascading mode, deduce with reasoning the maximum number of I/O devices PIC “X” can handle. Also find the number of cascading pins for PIC “X”.

2. Assume the table is a portion of the current memory address space of an Intel 8086. The microprocessor currently has the following values in its registers: SS = 2000h, SP = 1124h, CS = 3000h, IP = 1450h. Now a signal arrives at the INTR pin of this 8086.

| Addr | 00117h | 00116h | 00115h | 00114h | 00276h | 00277h | 00248h | 00279h |
|---|---|---|---|---|---|---|---|---|
| Data | 45h | 86h | 22h | 14h | 12h | 34h | 56h | 78h |

a. If the 8086 decides to service the interrupt, then do the bits of the flag register change? Explain with reasoning.
b. “Hardware interrupts can also be represented as Software interrupts” - Explain your views with regards to this statement.
c. If the signal is of Interrupt type 69, then deduce the values of CS and IP as the 8086 starts the service routine.
d. Deduce the values of CS, IP, and SS after the interrupt service routine has been completed.

3.

| Address | 0006A | 0006B | 0006C | 0006D | 0006Eh | 0006F |
|---|---|---|---|---|---|---|
| Data | 00h | 08h | 00h | 03h | 00h | 05h |

a. Use the given table to calculate the starting address of the ISR for interrupt type 27.
b. Suppose the above-mentioned ISR is a program consisting of nothing but 25 MOV operations, an IRET instruction at the end, and nothing else. The structure of each MOV operation in the ISR is “**MOV AL, [XXXX h]**”, where XXXX is an Offset. **Calculate** the physical address where we can find the IRET instruction

4. Assume you have several faulty 8259 PICs where the IR6 and IR7 pins do not work, and CAS2 is always giving a low (0) signal.

a. In this scenario explain what will be the maximum number of interrupts that can be serviced through these faulty PICs?
b. For a particular project, your microprocessor needs to provide service to 16 interrupts coming from I/O devices. If you use the same faulty PICs mentioned above then explain how many slave PICs will be required? Also using the proper diagram, illustrate how the master and slave PICs will be connected with each other.

5.

| Address | 000B6 | 000B7 | 000B8 | 000B9 | 000BA |
|---|---|---|---|---|---|
| Data | 20h | 30h | 40h | 50h | 60h |

a. Suppose, the interrupt pointer for Interrupt type “X” is 31434 h. Additionally, the corresponding CS values of the ISR for this interrupt are stored starting from the memory location 000B6 h. Now, deduce what value will be stored in the memory location 000B5.
b. Using the result obtained from “A”, deduce the value of “X”.
c. “All software interrupts are maskable”- Do you agree with this? Explain your answer
d. When servicing an interrupt explain why does the 8086 clear IF and TF?

6.

a. A student designed a new PIC called ‘PIC 2.0’ which takes 20 interrupt requests via its IRR0 – IRR19 pins. The size of its Interrupt Mask register (IMR), Interrupts Service Register (ISR), and Interrupt Request Register (IRR) is 20 bits. It also has 4 Cascading (CAS) pins.
    i. Assume the value of ISR0 - ISR4 is 10000, the value of IMR0 - IMR4 is 10101, and the value of IRR0 - IRR4 is 01110. Now, explain the order in which the interrupts IR0-IR4 will be serviced. Assume IR0 has the highest priority and IR19 has the least priority.
b. The student decided to cascade 20 PIC 2.0’s together in a master-slave configuration. However, due to a design flaw in the PIC 2.0, she noticed that the master PIC 2.0 could not communicate with all the connected 20 slave PIC 2.0s. Now assuming all IRR pins are Unmasked.
    i. Calculate the maximum number of interrupts that a master PIC 2.0 can handle in cascade mode. Show necessary calculations.
    ii. Explain how you would correct the flaw in the design so that the master can communicate with all the 20 slave PIC 2.0s.
    iii. Explain 2 possible methods of stopping the servicing of any interrupts between the master and the 8086 without disconnecting the wires or disabling the INTR / INT / INTA’ pins.

7. Use the given table to calculate the starting address of the ISR for interrupt type 29.

| Address | 00074 | 00075 | 00076 | 00077 | 00078 | 00079 |
|---|---|---|---|---|---|---|
| Data | 08h | 00h | 03h | 00h | 00h | 05h |

a. Assuming that the above ISR has only n “MOV AL, [XXXX h]” (XXXX are 4 hex bits offset). If the IRET instruction is found in the address 0003D, find the value of n?
b. Assume you have several faulty 8259 PICs where the IR5 and IR6 pins do not work, and CAS2 is always giving a low (0) signal.
    i. In this scenario explain what will be the maximum number of interrupts that can be serviced through these faulty PICs?
    ii. Draw the block diagram for such a system connecting the master and slave PICs

8. Assume a hypothetical scenario where CS and IP values for the starting address of an Interrupt Service Routine (ISR) are 1230h and 2000h, respectively. Here each memory location can store a maximum of 1 byte data. Now the ISR consists of 15 lines of code and assume 3 bits are required to store each line of code. Now, deduce mathematically the address where the IRET instruction will be found.

9. Assume the CS value required to locate an ISR is CBCCh and is stored at memory locations 000AEh and 000AFh respectively. Additionally assume the Interrupt vector (IV) of that ISR is CCEF2h. Hence, deduce the Interrupt Type that caused the aforementioned ISR and the value that will be stored at 000ADh.

10. Assume at T= 0 ns, an interrupt request is made at IRR = 4 of a PIC working in fixed priority mode. The duration required to serve the interrupt is 20 ns. However, at T=10 ns another interrupt request is made at IRR = 2 which requires 25 ns to be served. Finally, at T = 15 ns, the last interrupt request arrives at IRR = 7 which takes 5 ns to get served.

a. Explain how the PIC performs its task in this case. Your answer must refer to the changes in ISR and IRR at each stage. Assume the interrupt flag can never be reset to 0.
b. Deduce the time at which PIC starts serving the IRR 7 interrupt request.
c. Illustrate the Internal Block Diagram of an 8259A PIC.

# Basic Input Output

1. Explain what is variable addressing? Which register is used for variable addressing?

2. Explain what is Programmed and Interrupt Driven I/O? Why do we need DMA?

3. Explain how data is transferred between the I/O and the memory of an Intel 8086 using the 8237A DMA controller.

4. (a) To adopt the utility of 15 channels, estimate the minimum number of secondary 8237 DMA controllers needed.

(b) Construct the DMA cascading diagram based on your answer in (a). Your diagram should show how the secondary controllers are connected with the primary controller using the appropriate pins.

5. (a) To adopt the utility of 13 channels, estimate the minimum number of secondary 8237 DMA controllers needed.

(b) Construct the DMA cascading block diagram based on your answer in (a). Your diagram should show how the secondary controllers are connected with the primary controller using the appropriate pins.

6. Describe the values of the given pins of an 8237 DMA Controller during a DMA write cycle:

i. $\overline{IOR}$ ii. $\overline{IOW}$ iii. $\overline{MEMW}$ iv. $\overline{MEMR}$

7. Describe the values of the given pins of an 8237 DMA Controller during a DMA write cycle:

i. $\overline{IOR}$ ii. $\overline{IOW}$ iii. $\overline{MEMW}$ iv. $\overline{MEMR}$

8. Describe the tasks of the given pins of the 8237 DMA Controller:

a. AEN
b. DACK3-DACK0
c. HRQ
d. DREQ
e. HLDA
f. $\overline{IOR}$
g. $\overline{IOW}$
h. $\overline{MEMW}$
i. $\overline{MEMR}$
j. DB0-DB7

9. Intel. decided to create a new version of the 8237A DMA controller called “DMA 6” which supports 6 channels instead of the traditional 4 channels and has 20 address pins. Additionally, the size of the port addresses are now changed to 20 bits. A special register in 8086 named “QX” keeps the value of the 20 bits port address when using Variable Addressing. But for fixed addressing assume the size of the P8 byte remains the same.

(a) Deduce the maximum number of I/O devices the DMA 6 can handle.

(b) Calculate the total size of the memory that the DMA 6 is capable of addressing

(c) Deduce the new range of addressing space for fixed addressing and variable addressing if Isolated I/O is used for I/O address mapping

(d) Assume DMA 6 is in cascading mode. Illustrate using a diagram how 2 secondary DMA 6's can be connected with one primary DMA 6. Also, explain the total process of how data transfer takes place between an Intel 8086 and the cascaded DMA 6's.

10. Which mode does the 8086 microprocessor operate in when a DMA controller is used in the system and connected to the 8086 using HOLD and HLDA pins? (Minimum mode/Maximum mode)

11. For data transfer the following scenarios, among programmed I/O, Interrupt based I/O and DMA based I/O, which one would you prefer and why?

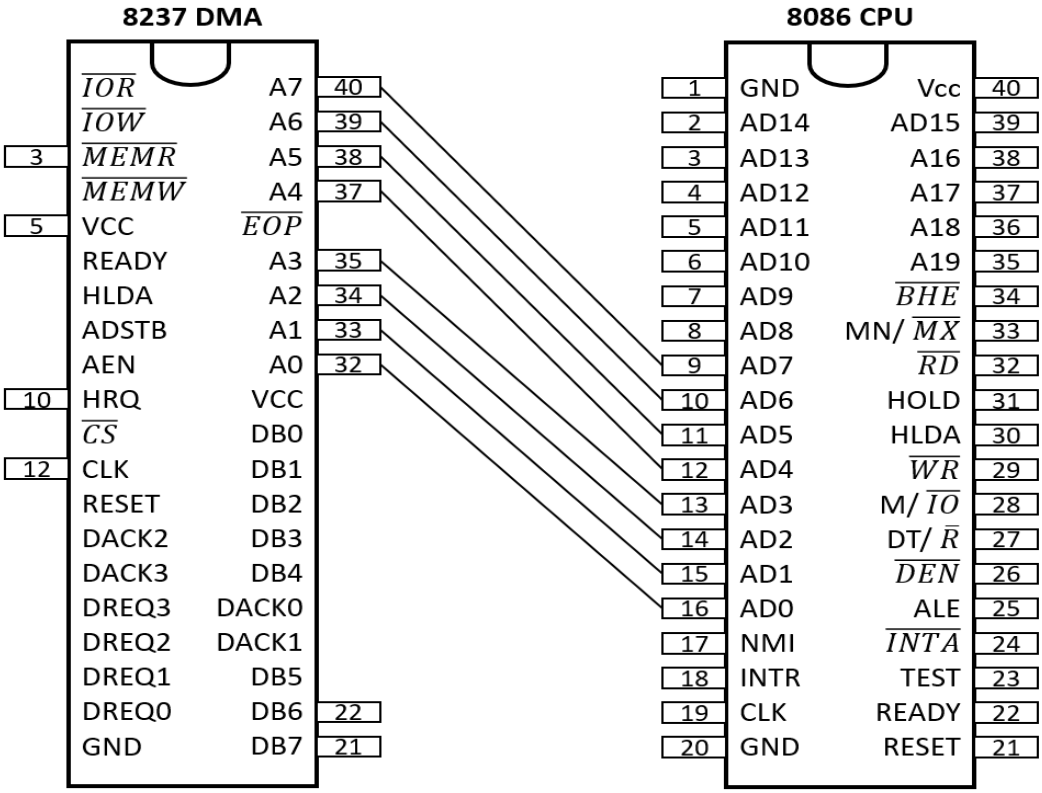
a. Matrix keypad connected to a door locking mechanism
b. Power outage detection in computers
c. Transferring data from Hard disk to RAM.

12. How many I/O ports does the 8086 processor support in the isolated I/O method?

13. Compare the advantages and disadvantages of Memory mapped I/O vs Isolated I/O.

14. A particular I/O device utilizes the MOV operation to take input from a temperature sensor. Is the device memory-mapped or isolated?

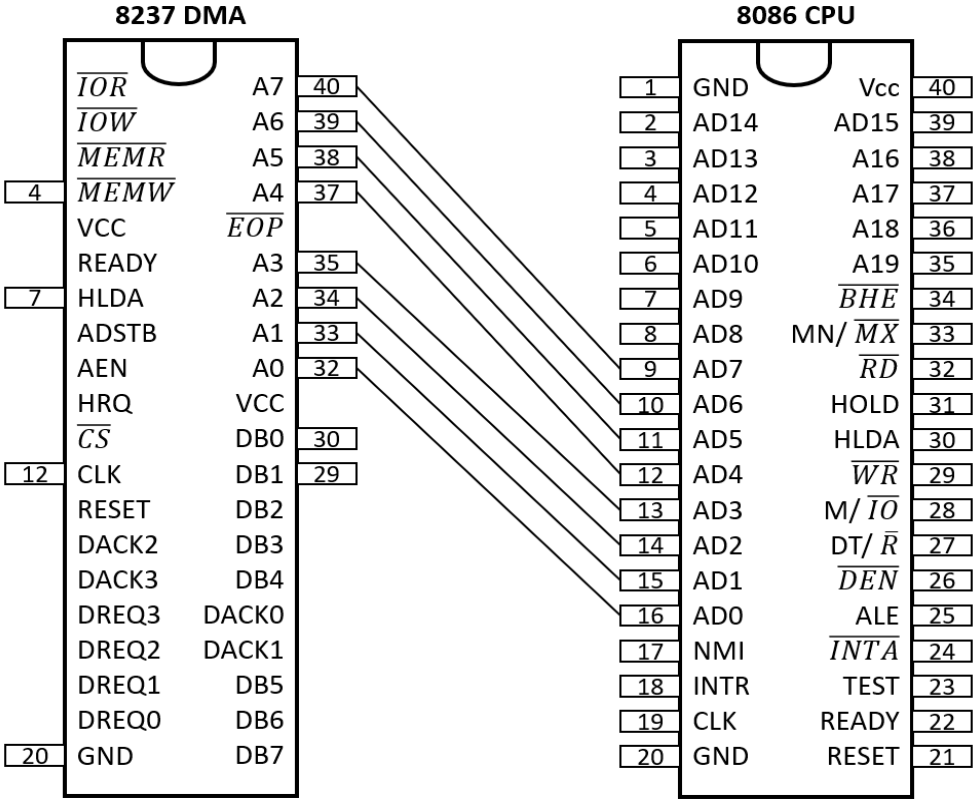
15.



a. Based on your knowledge of the operation of the DMA chip, connect the six free pins of the 8237 IC to the appropriate pins of the 8086 IC. The address bus of the two ICs has been connected as a demonstration. Note that DB(0-7) pins of 8237 carry the higher byte of the 16-bit memory address. Assume both chips run on the same clock. Also if two pins have the same source, connect them with each other.

b. “An 8086-based system can support input-output devices in the memory mapped mode and isolated mode simultaneously.” Do you agree? Explain with reasoning.

16.

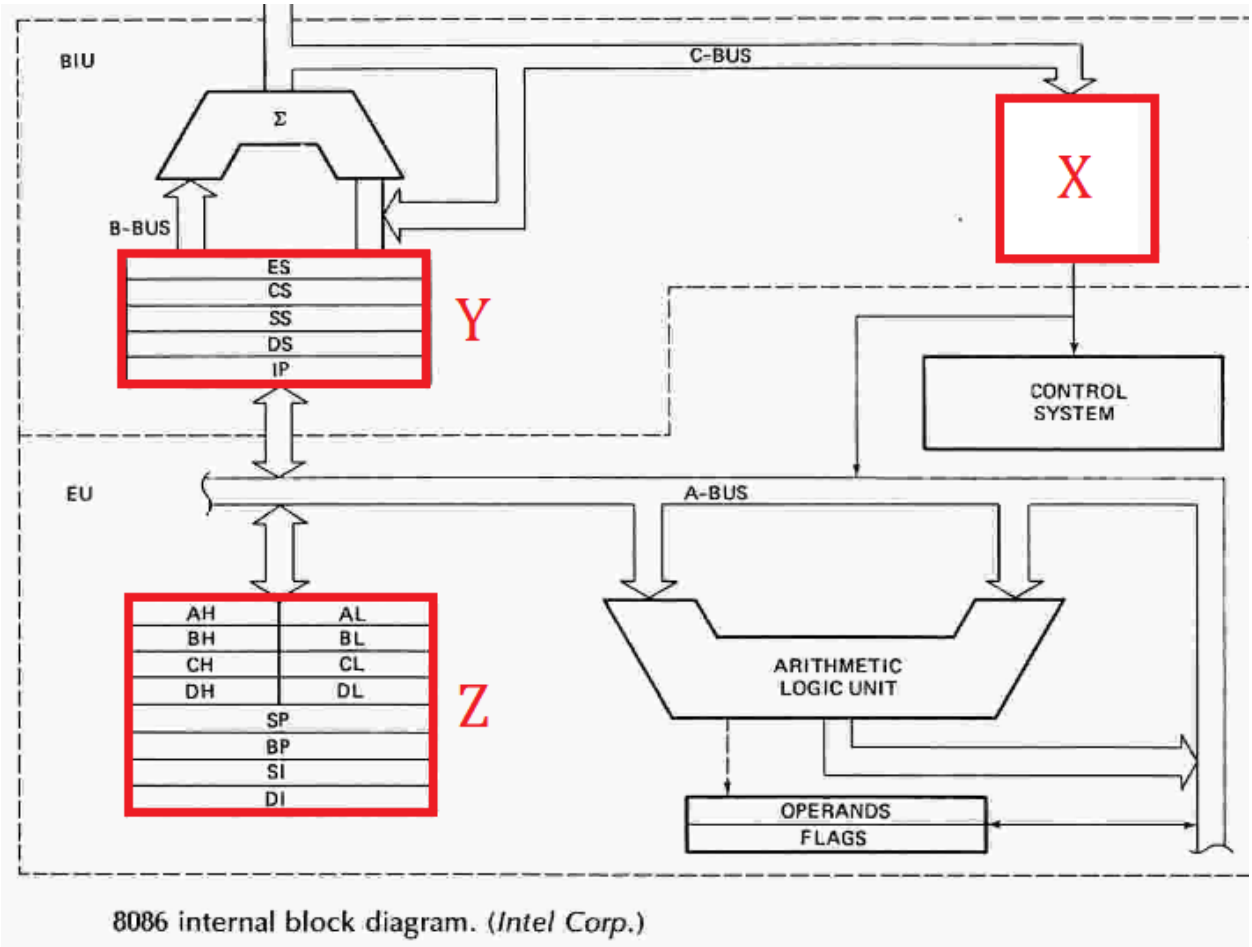



a. Based on your knowledge of the operation of the DMA chip, connect the six free pins of the 8237 IC to the appropriate pins of the 8086 IC. The address bus of the two ICs has been connected as a demonstration. Note that DB(0-7) pins of 8237 carry the higher byte of the 16-bit memory address. Assume both chips run on the same clock. Also if two pins have the same source, connect them with each other.

b. “By using memory mapped I/O, we can access a greater number of input/output ports compared to the isolated I/O method.” Do you agree? Explain with reasoning.

# Quiz Questions

## Q1 (Spring 2023 Section 5)




8086 internal block diagram. (*Intel Corp.*)

1. **Analyze** the Figure shown above.
   (a) **Identify** 'X', 'Y', and 'Z'. **Determine** the necessity of 'X'.
   (b) Briefly **explain** the differences between 'Y' and 'Z'.

**2.** A 2kB RAM is placed in the memory space with a starting address of 1B800h. What will be the ending address?

**3**. The address **0010:5000 h** contains an instruction. Calculate the value of IP for a program to execute that instruction if **CS is 0000 h**.

# Q1 (Spring 2023 Section 7 Set-A)

A microprocessor has a 26 Bit Address Bus.

| | |
|---|---|
| 1. What is the memory capacity in megabytes? | |
| 2. What is the physical address (in hexadecimal) of the first memory location? | |
| 3. What is the physical address (in hexadecimal) of the final memory location? | |

For each of the following registers, write down which functional unit it belongs to

| | |
|---|---|
| 4. SS | |
| 5. IP | |
| 6. SI | |

The register values are given for the following instruction >>> MOV AL, [SI]
DS = **1000h**, CS = **3000h**, SS = **8A40h**, SI = **2000h**, DI = **030Fh,** IP = **1234h**, SP = **0010h**

| | |
|---|---|
| 7. What is the physical address of the source data? | |
| 8. What is the logical address of the current instruction? | |
| 9. Consider the command >>> PUSH AL. What is the physical address of the location where the value of AL is stored? | |

A certain data is stored at the location **FFFAh:FFCAh** of the RAM of the 8086 processor (overlapping segmentation)

| | |
|---|---|
| 10. What is the physical address of the mentioned data? | |

Considering overlapping segmentation, fill up the following table:

| | Segment | Offset | Physical Address |
|---|---|---|---|
| 11. | **6900h** | **0420h** | |
| 12. | **0100h** | | **FFFFh** |
| 13. | | **0109h** | **1F309h** |
| 14. | | **ABC1h** | **C2105h** |

Show the result **in binary** and write down the flag values after the execution of the last instruction:

>>> MOV AX, FF13H
>>> MOV BX, 0135H
>>> SUB AX, BX

| | |
|---|---|
| AX | |
| BX | |
| 15. AX (After SUB) | |

| | |
|---|---|
| 16. PF | |
| 17. SF | |
| 18. OF | |
| 19. CF | |
| 20. ZF | |

# Q1 (Spring 2023 Section 7 Set-B)

A microprocessor has a 28 Bit Address Bus.

| | |
|---|---|
| 1. What is the memory capacity in megabytes? | |
| 2. What is the physical address (in hexadecimal) of the first memory location? | |
| 3. What is the physical address (in hexadecimal) of the final memory location? | |

For each of the following registers, write down which functional unit it belongs to

| | |
|---|---|
| 4. DI | |
| 5. ES | |
| 6. BX | |

The register values are given for the following instruction >>> MOV AL, [DI]
DS = **2000h**, CS = **4000h**, SS = **8A40h**, SI = **2000h**, DI = **030Fh,** IP = **F234h**, SP = **0020h**

| | |
|---|---|
| 7. What is the physical address of the source data? | |
| 8. What is the logical address of the current instruction? | |
| 9. Consider the command >>> PUSH AL. What is the physical address of the location where the value of AL is stored? | |

A certain data is stored at the location **FFEBh:FF1Dh** of the RAM of the 8086 processor (overlapping segmentation)

| | |
|---|---|
| 10. What is the physical address of the mentioned data? | |

Considering overlapping segmentation, fill up the following table:

| | Segment | Offset | Physical Address |
|---|---|---|---|
| 11. | **4200h** | **0069h** | |
| 12. | **0200h** | | **FFFFh** |
| 13. | | **0209h** | **2F309h** |
| 14. | | **AE31h** | **A2101h** |

Show the result **in binary** and write down the flag values after the execution of the last instruction:

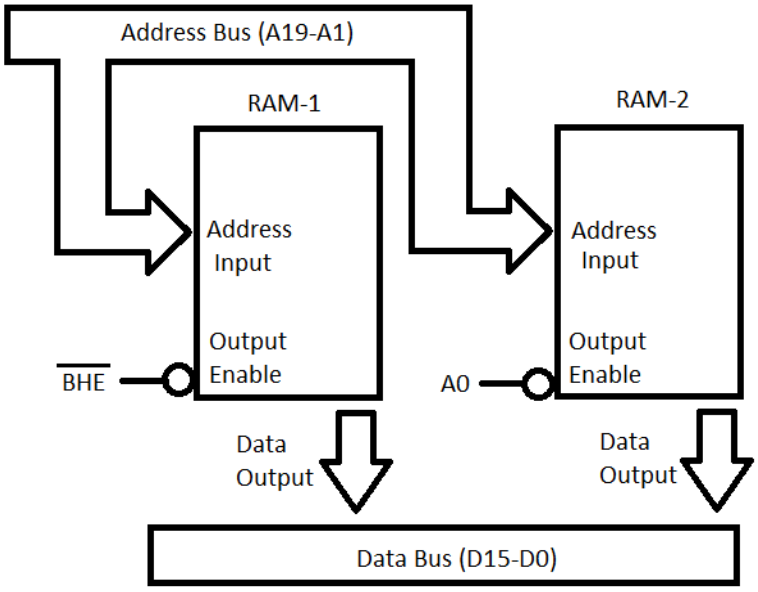
>>> MOV AX, FF13H
>>> MOV BX, 0135H
>>> SUB BX, AX

| | |
|---|---|
| BX | |
| AX | |
| 15. BX (After SUB) | |

| | |
|---|---|
| 16. PF | |
| 17. SF | |
| 18. OF | |
| 19. CF | |
| 20. ZF | |

# Q2 (Spring 2023 Section 7 Set-A)

Two individual RAM modules are connected to the 8086 processor in the following way to enable word size data transfer in a single bus-cycle:




| | | |
|---|---|---|
| 1 | For the given connection, identify whether RAM-1 is even bank or odd bank | |
| 2 | What is the memory size (in kilobytes) of the RAM-1 memory module? | |
| 3 | Which bits of the data bus are the data output pins of the RAM-1 module connected to? | |
| 4 | Which bits of the data bus are the data output pins of the RAM-2 module connected to? | |

Consider the following two scenarios:

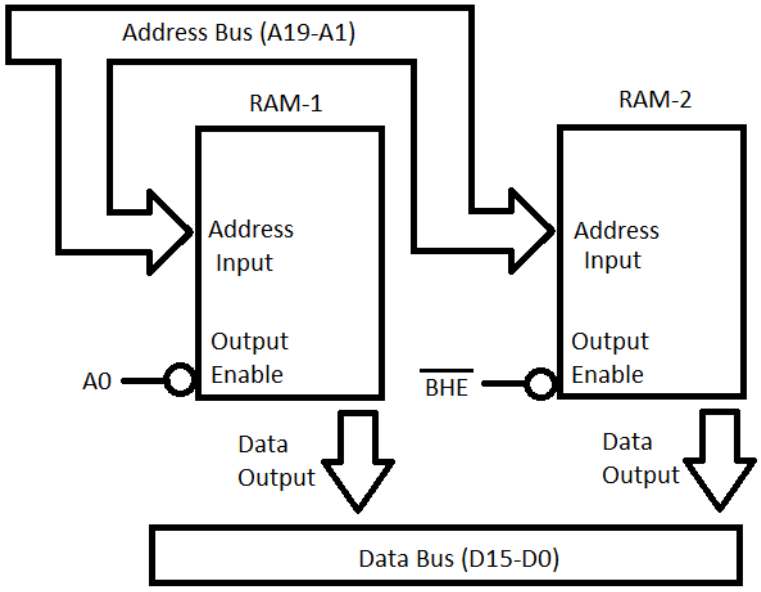
| **SCENARIO 1: DS = 0000H** | **SCENARIO 2: DS = 0000H** |
|---|---|
| (a) MOV BX, *last_4_digits_of_your_ID*<br>(b) MOV AL, [BX]<br>(c) MOV AH, [BX+1] | (a) MOV BX, *last_4_digits_of_your_ID*<br>(b) MOV AX, [BX] |

| | | |
|---|---|---|
| 5 | For scenario 1, instruction (b), which memory bank does the data come from? | |
| 6 | For scenario 1, instruction (b), what are the values of A0 and $\overline{BHE}$ ? | |
| 7 | For scenario 1, instruction (c), which memory bank does the data come from? | |
| 8 | For scenario 1, instruction (c), what are the values of A0 and $\overline{BHE}$ ? | |

| | | |
|---|---|---|
| 9 | After the execution of the given instructions, does the AX register contain the same data in both scenarios? (Yes/No) | |
| 10 | Do scenario 1 and scenario 2 require the same number of bus cycles for the given instructions? (Yes/No) | |

# Q2 (Spring 2023 Section 7 Set-B)

Two individual RAM modules are connected to the 8086 processor in the following way to enable word size data transfer in a single bus-cycle:




| | | |
|---|---|---|
| 1 | For the given connection, identify whether RAM-1 is even bank or odd bank | |
| 2 | What is the memory size (in kilobytes) of the RAM-1 memory module? | |
| 3 | Which bits of the data bus are the data output pins of the RAM-1 module connected to? | |
| 4 | Which bits of the data bus are the data output pins of the RAM-2 module connected to? | |

Consider the following two scenarios:

| **SCENARIO 1: DS = 0000H** | **SCENARIO 2: DS = 0000H** |
|---|---|
| (a) MOV BX, *last_4_digits_of_your_ID*<br>(b) MOV AL, [BX]<br>(c) MOV AH, [BX+1] | (a) MOV BX, *last_4_digits_of_your_ID*<br>(b) MOV AX, [BX] |

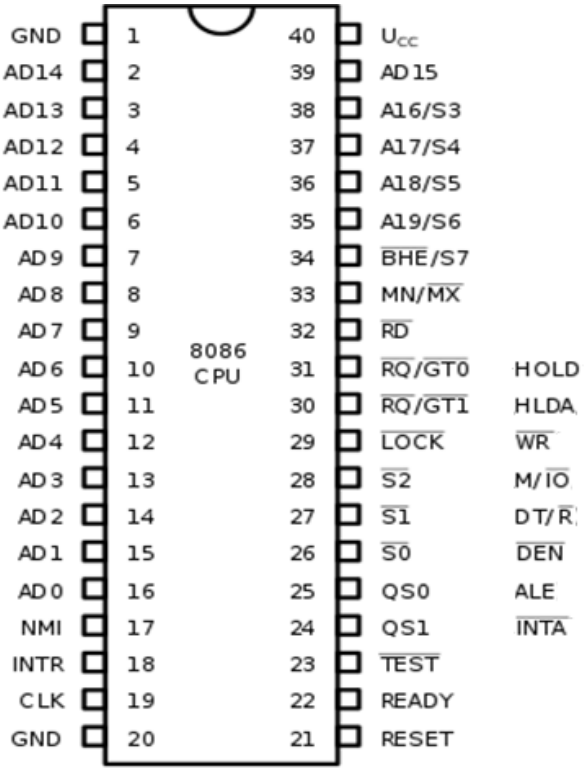
| | | |
|---|---|---|
| 5 | For scenario 1, instruction (b), which memory bank does the data come from? | |
| 6 | For scenario 1, instruction (b), what are the values of A0 and $\overline{BHE}$ ? | |
| 7 | For scenario 1, instruction (c), which memory bank does the data come from? | |
| 8 | For scenario 1, instruction (c), what are the values of A0 and $\overline{BHE}$ ? | |

| | | |
|---|---|---|
| 9 | After the execution of the given instructions, does the AX register contain the same data in both scenarios? (Yes/No) | |
| 10 | Do scenario 1 and scenario 2 require the same number of bus cycles for the given instructions? (Yes/No) | |

# Q3 (Spring 2023 Section 7)

Write in brief the functionality of each pin mentioned below:

| | |
|---|---|
| Vcc | |
| Gnd | |
| Clk | |
| AD0-AD15 | |
| A16-A19 | |
| $\overline{BHE}$ | |
| NMI | |
| INTR | |
| $\overline{INTA}$ | |
| MN/$\overline{MX}$ | |
| $\overline{RD}$ | |
| HOLD | |
| HLDA | |
| $\overline{LOCK}$ | |
| $\overline{WR}$ | |
| M/$\overline{IO}$ | |
| ALE | |
| $\overline{TEST}$ | |
| READY | |
| RESET | |


GND 1
AD14 2
AD13 3
AD12 4
AD11 5
AD10 6
AD9 7
AD8 8
AD7 9
AD6 10
AD5 11
AD4 12
AD3 13
AD2 14
AD1 15
AD0 16
NMI 17
INTR 18
CLK 19
GND 20
8086
CPU
40 $U_{CC}$
39 AD15
38 A16/S3
37 A17/S4
36 A18/S5
35 A19/S6
34 $\overline{BHE}$/S7
33 MN/$\overline{MX}$
32 $\overline{RD}$
31 $\overline{RQ}$/$\overline{GT0}$ HOLD
30 $\overline{RQ}$/$\overline{GT1}$ HLDA
29 $\overline{LOCK}$ $\overline{WR}$
28 $\overline{S2}$ M/$\overline{IO}$
27 $\overline{S1}$ DT/$\overline{R}$
26 $\overline{S0}$ $\overline{DEN}$
25 QS0 ALE
24 QS1 $\overline{INTA}$
23 $\overline{TEST}$
22 READY
21 RESET