Chapter 2

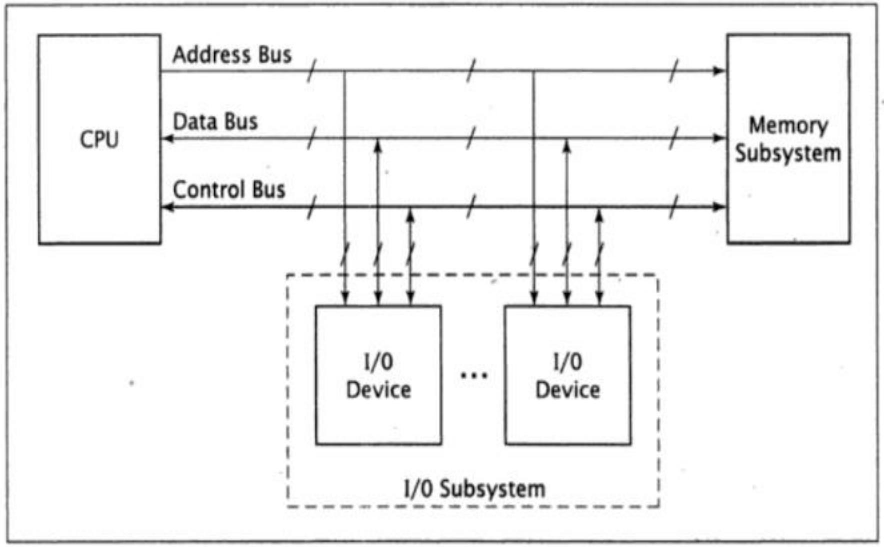
Computer Organization

Er. Shiva Ram Dam
Assistant Professor
Pokhara University

Contents:

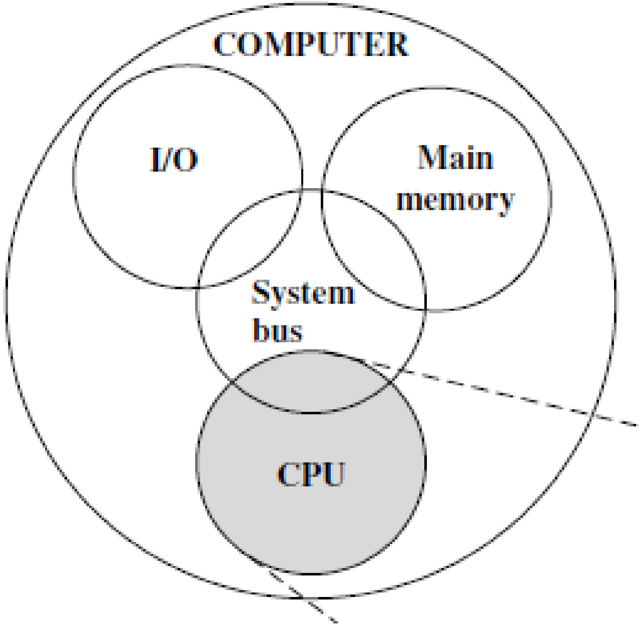
1. Basic Computer Organization

- This organization has three main components:
  - CPU,
  - memory subsystem, and
  - IO subsystem



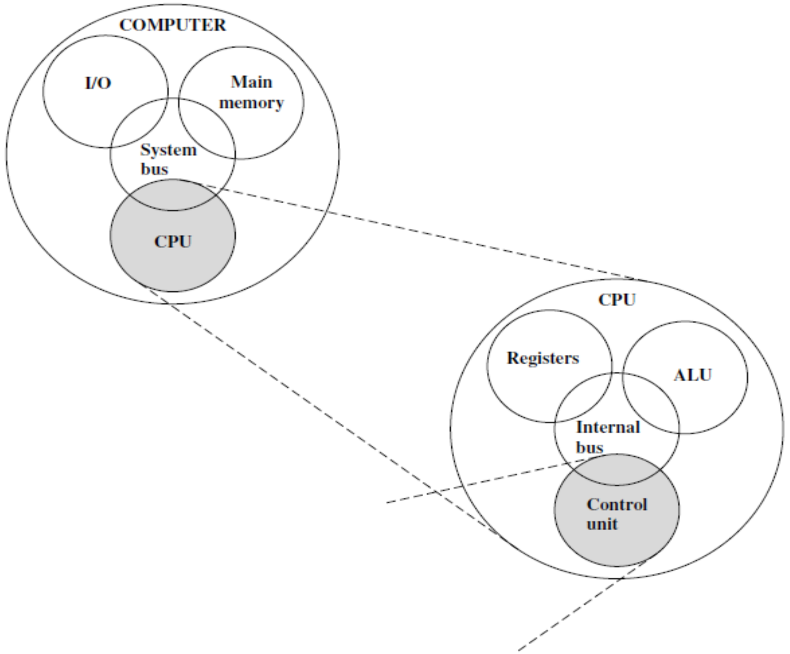
Structural view of Computer System

CPU	Controls the operation of the computer and performs its data processing functions; also referred as processor.
Main Memory	Stores data
I/O	Moves data between the memory and its external environment
System Interconnection	Some mechanism that provides for communication among CPU, main memory and I/O. it is done through system bus.



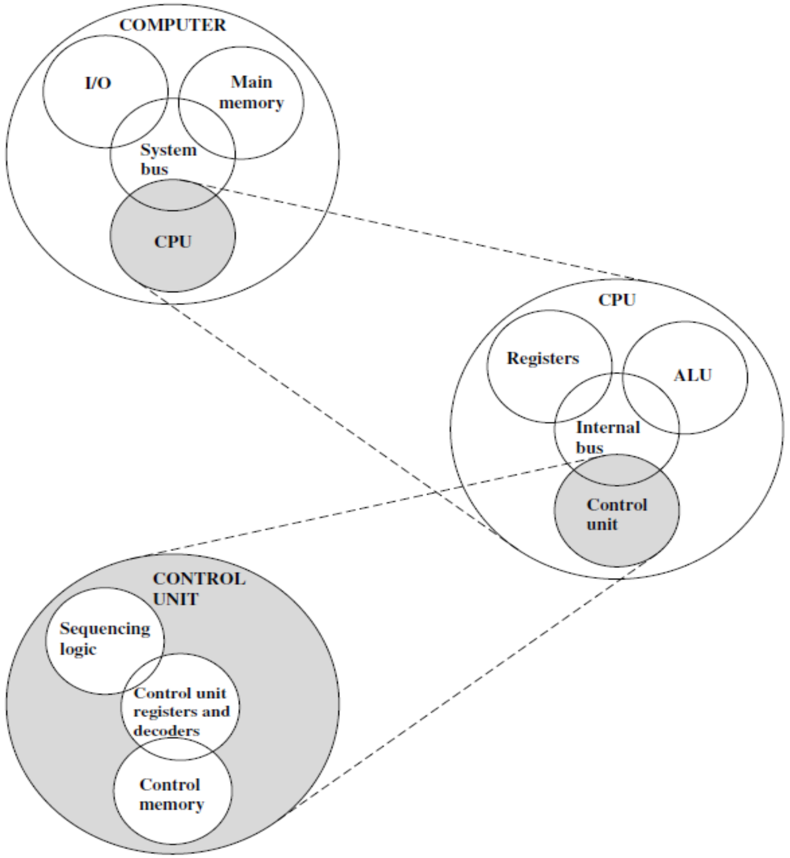
Structural view of Computer System (Contd.)

Control Unit:	Controls the operation of the CPU and hence the computer.
ALU:	Performs the computer's data processing function.
Registers:	Provides storage internal to the CPU.
CPU interconnection:	Some mechanisms that provide for communication among the CU, ALU and registers.



Structural view of Computer System (Contd.)

Control Memory:	The registers which store data related to control operation.
Sequential Logic:	Digital circuit whose output depends on current input and state of circuit.
CU register & decoder:	Performs storage and interpretation of the instruction.



2. System Buses

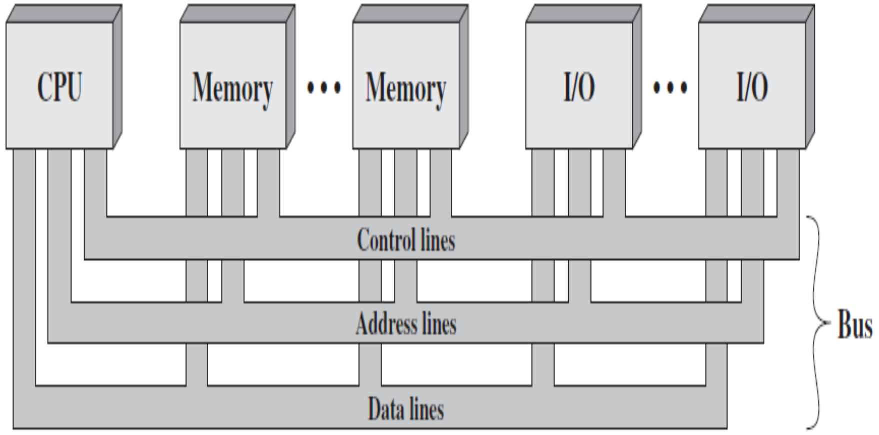
- Physically, a bus is a set of wires.
- The components of the computer are connected to the buses.
- To send information form one component to another, the source component outputs data onto the bus.
- The destination component then inputs this data from the bus.
- As the complexity of a computer system increases, it becomes more efficient at using buses rather than direct connections between every pair of devices.
- Buses use less space on a circuit board and require less power than a large number of direct connections.
- They also require fewer pins on the chip or chips that comprise the CPU.

2. System Buses (Contd.)

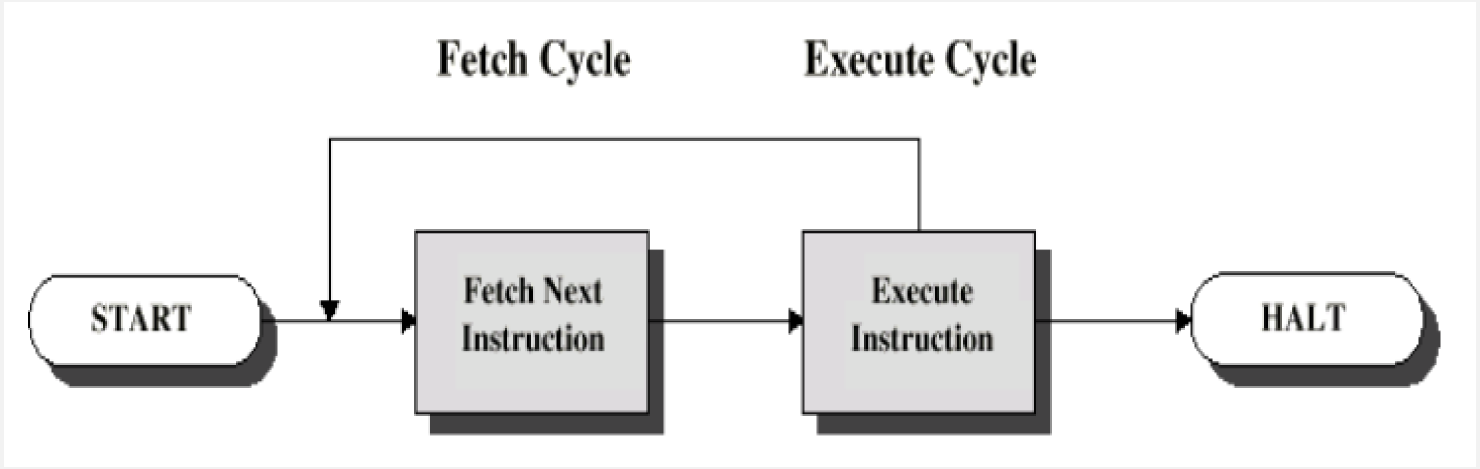
- It is a shared transmission medium. Multiple devices connect to the bus, and a signal transmitted by one device is available for reception by all other devices attached to the bus.
- If two devices transmit during the same time period, their signals will overlap and become garbled.
- Thus, only one device, at a time, can successfully transmit.
- Typically, a bus consists of multiple communication pathways or lines.
- Each line is capable of transmitting signals representing binary 1 or 0.
- An 8-bit unit of data can be transmitted over eight bus lines.
- A bus that connects major computer components (processor, memory, IO) is called a system bus.

2. System Buses (Contd.)

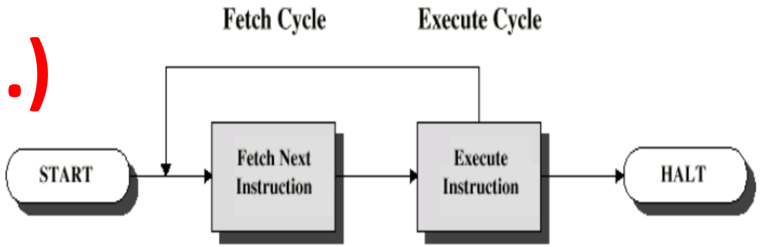
Data Lines	The data lines provide a path for moving data among system modules. These lines, collectively, are called the data bus.
Address Lines	The address lines are used to designate the source or destination of the data on the data bus.
Control Lines	The control lines are used to control the access to and the use of the data and address lines. Control signals transmit both command and timing information among system modules. Timing signals indicate the validity of data and address information. Command signals specify operation to be performed.



3. Instruction Cycle



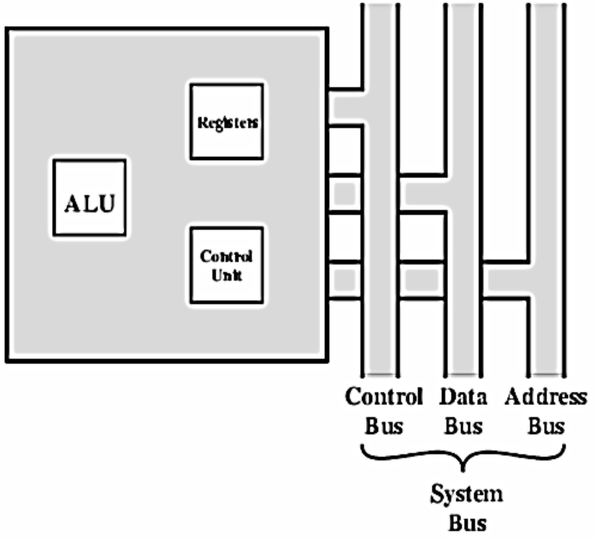
3. Instruction Cycle (Contd.)



- The instruction cycle is the procedure a microprocessor goes through to process an instruction.
- First the microprocessor **fetches** or reads, the instruction from memory.
- Then it **decodes** the instruction, determining which instruction it has fetched.
- Finally, it performs the operations necessary to **execute** the instruction.
- Each of these functions- fetch, decode and execute- consists of a sequence of one or more operations.

4. CPU organization

- CPU consists of three components:
  - ALU,
  - CU and
  - Registers.
- ALU performs arithmetic and logical operation,
- CU provides control signals, and
- registers store binary information and provide very fast information to ALU and out of CPU.
- These components are connected with each other and with external environment through system bus as shown below:



2.4 Memory Sub-system Organization and Interfacing

- Memory is the group of circuits used to store data.
- Two types of Memory:
  1. RAM
     a) SRAM:
        - More like registers, no need of refresh

     b) DRAM:
        - Leaky capacitor, needs refresh
  2. ROM
     a) PROM
     b) EPROM
     c) EEPROM
     d) Flash Memory

RAM Vs ROM

RAM

- Volatile in nature.
- Used for reading and writing data in any order as required.
- Data is stored and read many times to and from this type of memory.
- Memory cells can be accessed for information transfer from any desired random location.

ROM

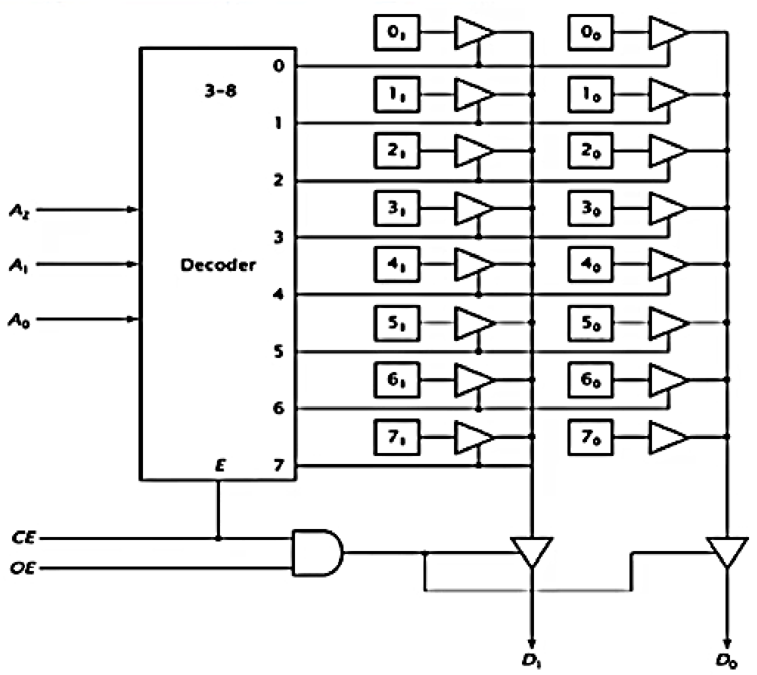
- Read only memory (ROM) contains a permanent pattern of data that cannot be changed.
- A ROM is non-volatile, i.e., no power source is required to maintain the bit values in memory.
- While it is possible to read a ROM, it is not possible to write new data into it.
- The data or program is permanently presented in main memory and never be loaded from a secondary storage device with the advantage of ROM.

Internal Chip Organization

- Linear organization
- Two-dimensional organization

Linear organization

- Consider an 8 x 2 ROM chip: (Chip is in 2^n x m representation where n is address line and m is data line)
  - Address line = 3
  - Data lines = 2
  - 16 bits of internal storage
  - If CE = 0, decoder is disabled and no location is selected. The Tri-state buffer for that location's cell are enabled, allowing data to pass to the output buffers.
  - If CE = 1and OE= 1, tri-state buffer for that location cell is enabled and data is output from the chip; otherwise the outputs are tri-stated.

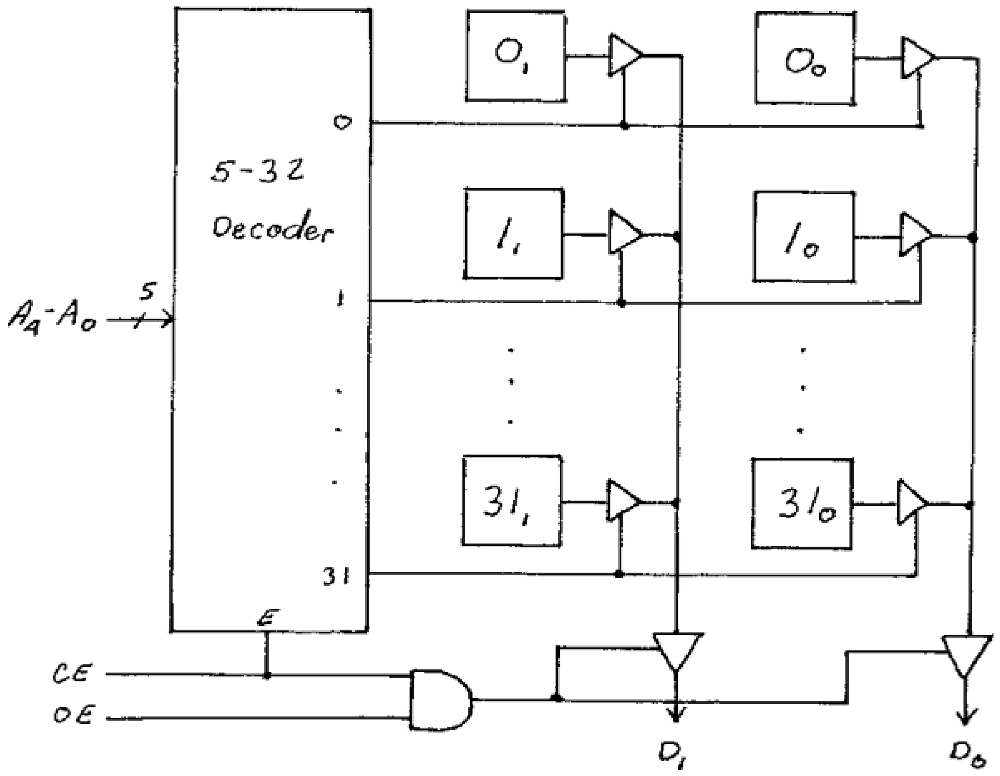


Classwork:

- **show the internal linear configuration of a 32x2 memory chip.**

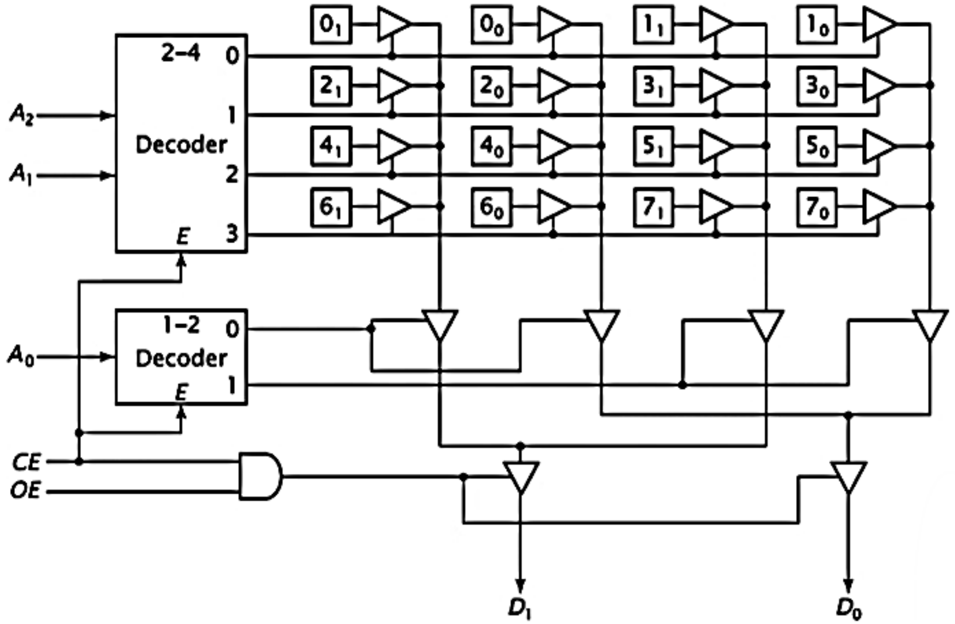
Classwork:

- show the internal linear configuration of a 32x2 memory chip.



Two-dimensional organization

- As the number of locations increases, the size of address decoder needed, becomes extremely large.
- To remedy this problem, the memory chip can be designed using multiple dimension of decoding.
- This configuration has four rows with four bits per row; each holds two data values.
- Here,
- A0→ selects one of the 2 locations in a row
- A1, A2→ selects one of 4 rows.

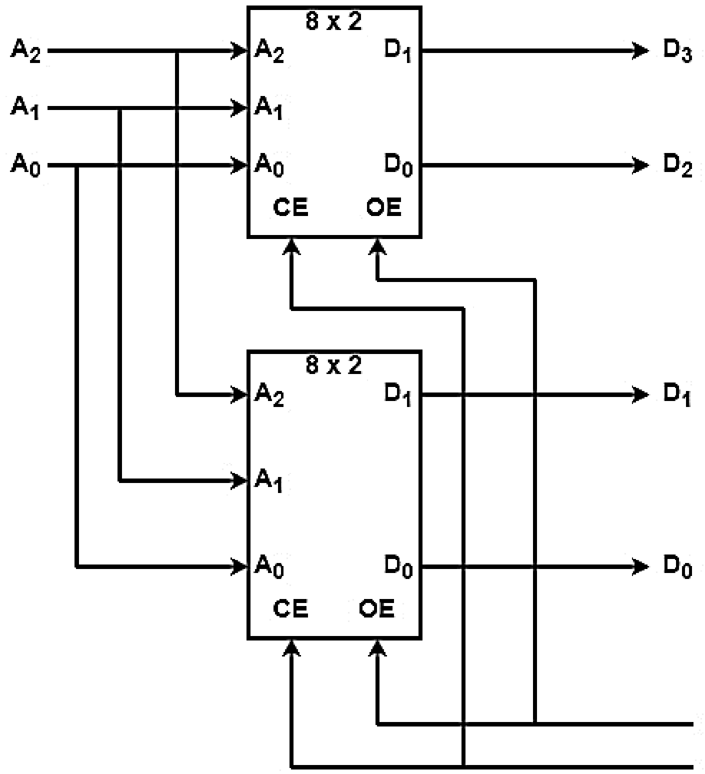


2.5 Memory Sub system Configuration

- There is technique for joining memory chips to form a memory subsystem.
- Two or more chips can be combined to generate a memory with more bits per location.
- This is done by linking the corresponding address and control signals of the chips and linking their data pins to various bits of the data bus.

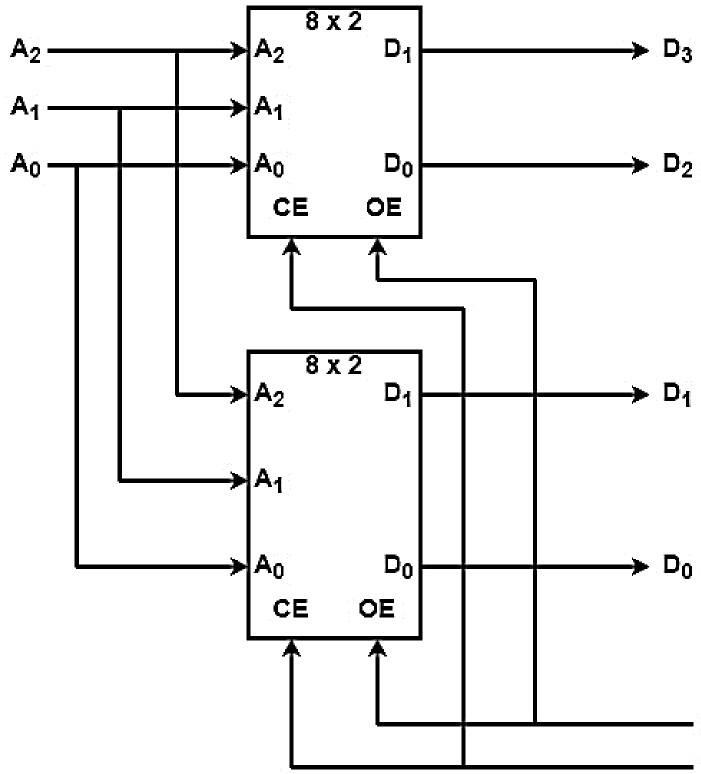
Construction of 8 x 4 ROM from two 8 x 2 ROMs:

- **Two 8 x 2 chips** can be **combined** to generate an **8 x 4 memory** as displayed in the figure.
- **Both chips get the equal three address inputs** from the bus, and the same chip enables and output enables signals.



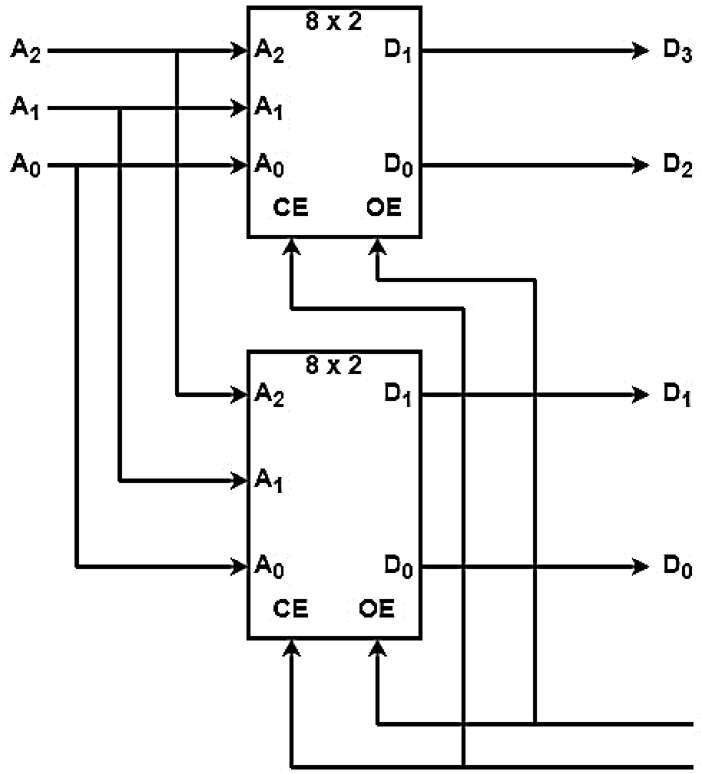
Construction of 8 x 4 ROM from two 8 x 2 ROMs:

- The data chips of the first chips are linked to bits 3 and 2 of the data bus(i.e. D_3 and D_2)
- and those of the other chips are linked to bits 1 and 0 (i.e. D_1 and D_0)
- When the CPU reads data, it locates the address on the address bus.
- Both chips read in address bits A_2, A_1, and A_0 and implement their internal decoding..



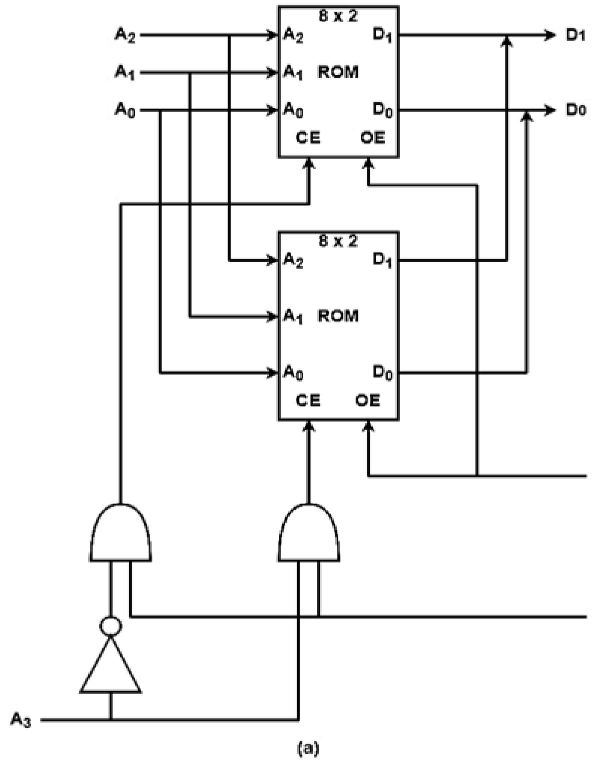
Construction of 8 x 4 ROM from two 8 x 2 ROMs:

- If the **CE** (Chip Enable) and **OE** (Output Enable) signals are **activated**, the **chips output their data onto the four bits** of the data bus.
- Because the address and enable signals are equal for both chips, **either both chips and neither chip is active** at any given time.
- The device never has only one of the two, active.
- For this reason, they facilitate as only a single 8 x 4 chip at least until the CPU is concerned.

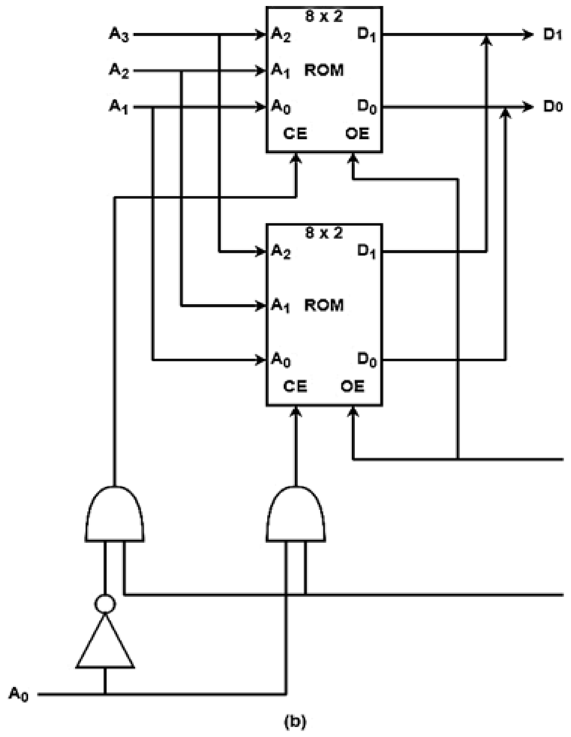


Construction of 16 x 2 ROM from two 8 x2 ROM chip:

- Instead of generating wider words, chips can be combined to make more words.
- The similar two 8 x 2 chips could alternatively be configured as a 16 x 2 memory subsystem.



a) With high-order interleaving

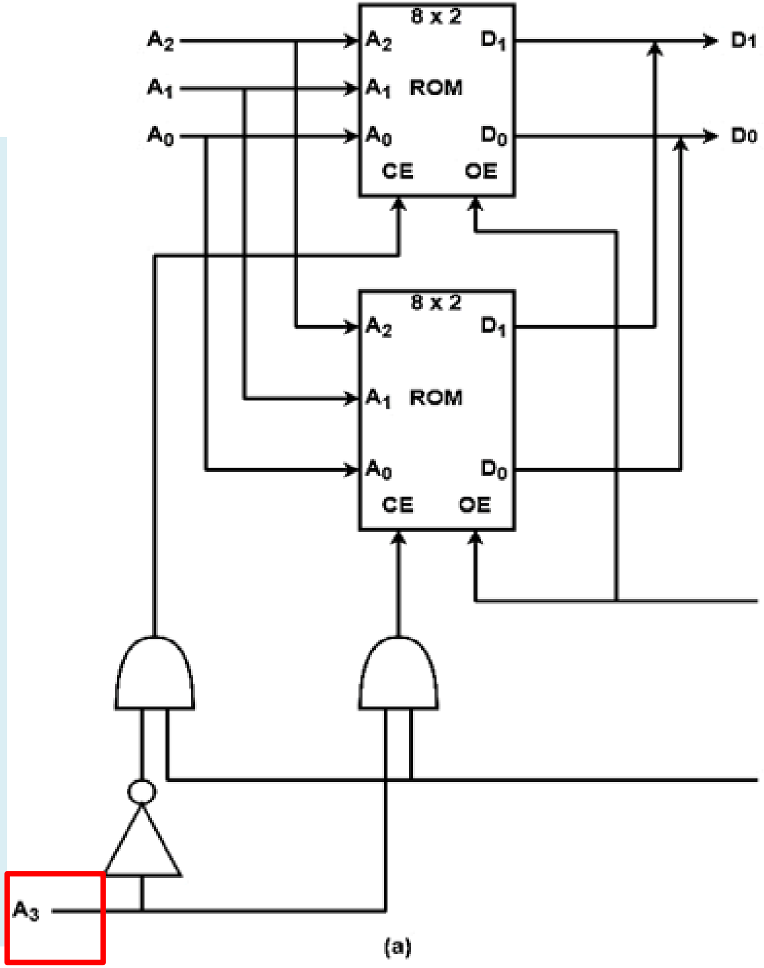


b) with low-order interleaving

Figure: A 16 x2 memory subsystem constructed from two 8 x 2 TOM chip

Using high-order interleaving

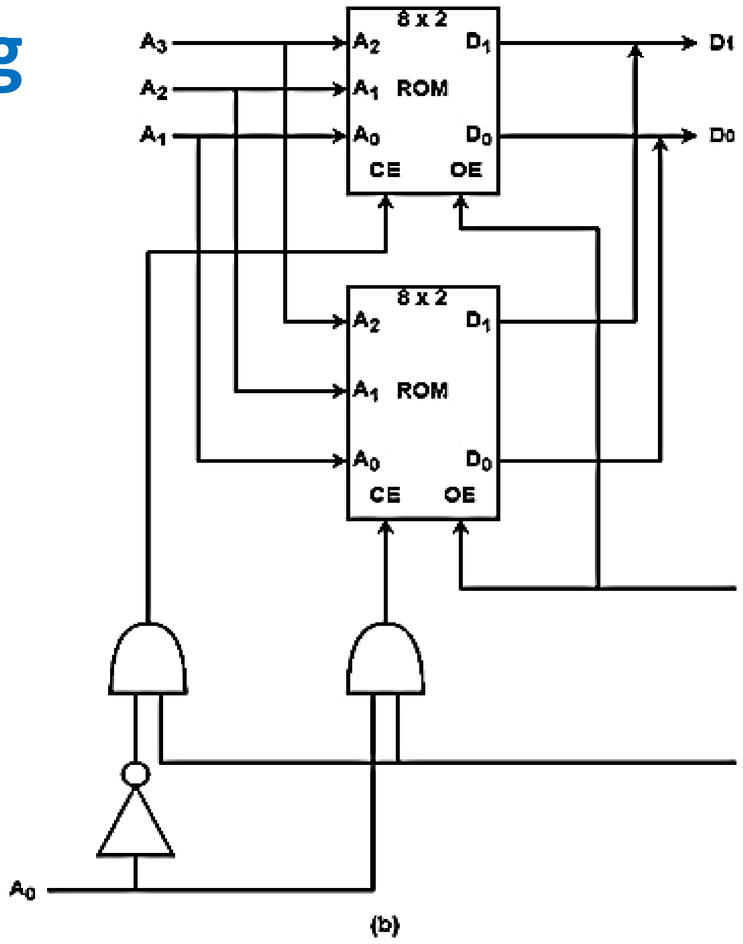
- The configuration in figure (a) uses **high-order interleaving.** All memory locations inside a chip are contiguous within system memory.
- The **upper chip** is configured as memory **locations 0 to 7** (0000 to 0111) and
- the **lower chip** as **locations 8 to 15** (1000 to 1111).
- The upper chip **always** has **A_3=0** and the lower chip has **A_3=1.**
- This difference can choose one of the two chips.
- When **A_3=0,** the **upper chip is enabled** and the **lower chip is disabled.**
- The output enables can be linked, because only the chip that is enabled with output data.
- Both chips correspond to similar data bits, both are linked to D_1 and D_0 of the data bus



a) *With high-order interleaving*

Using low-order interleaving

- Consider the configuration displayed in figure (b) which uses low-order interleaving.
- The **upper chip is enabled** when **A_0=0,** or by addresses XXX0, in this case, **0, 2, 4, 6, 8, 10,12, and 14.**
- The **lower chip is enabled** when **A_0=1,** which is correct for addresses **1, 3, 5, 7, 9, 11, 13, and 15.**



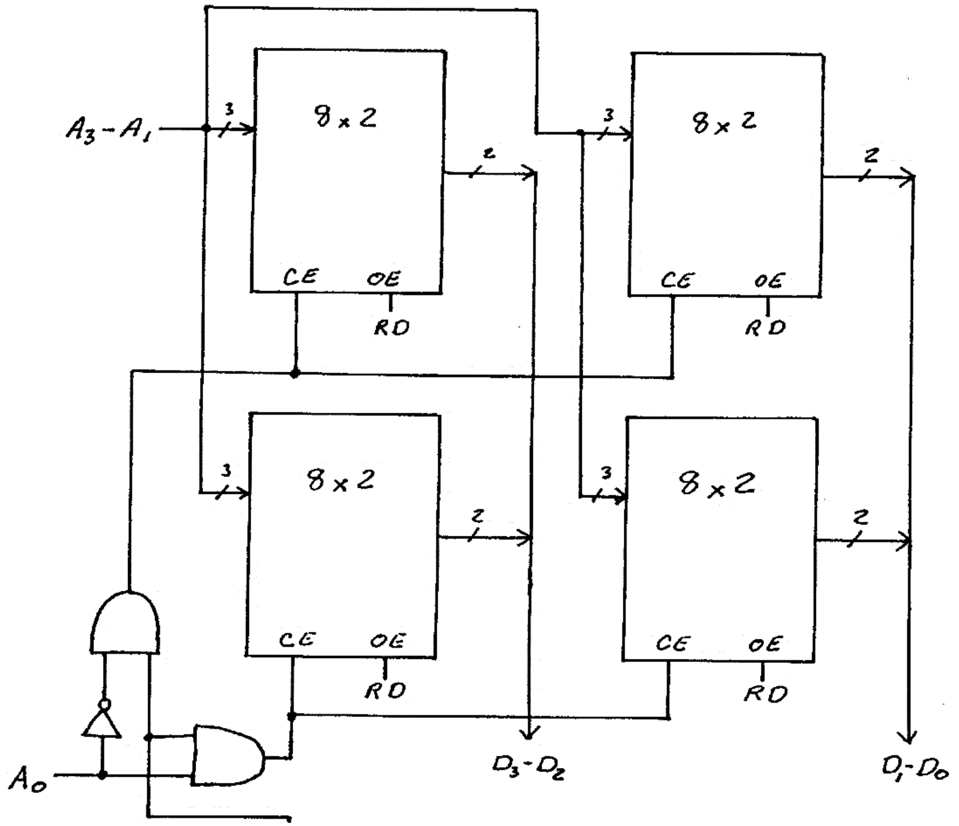
b) with low-order interleaving

Classwork:

1. Design a 16x4 memory subsystem with low order interleaving using 8x2 memory chips for a computer system with an 8-bit address bus.

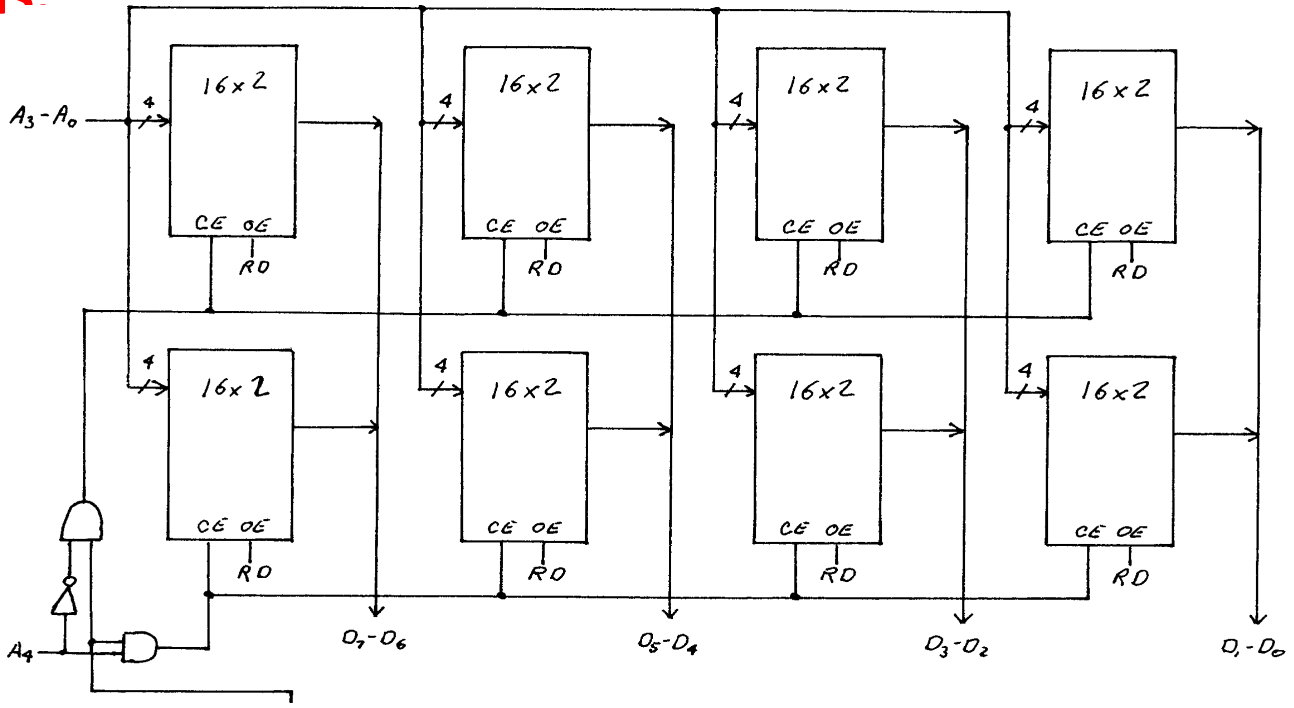
Classwork:

Design a 16x4 memory subsystem with low order interleaving using 8x2 memory chips for a computer system with an 8-bit address bus.

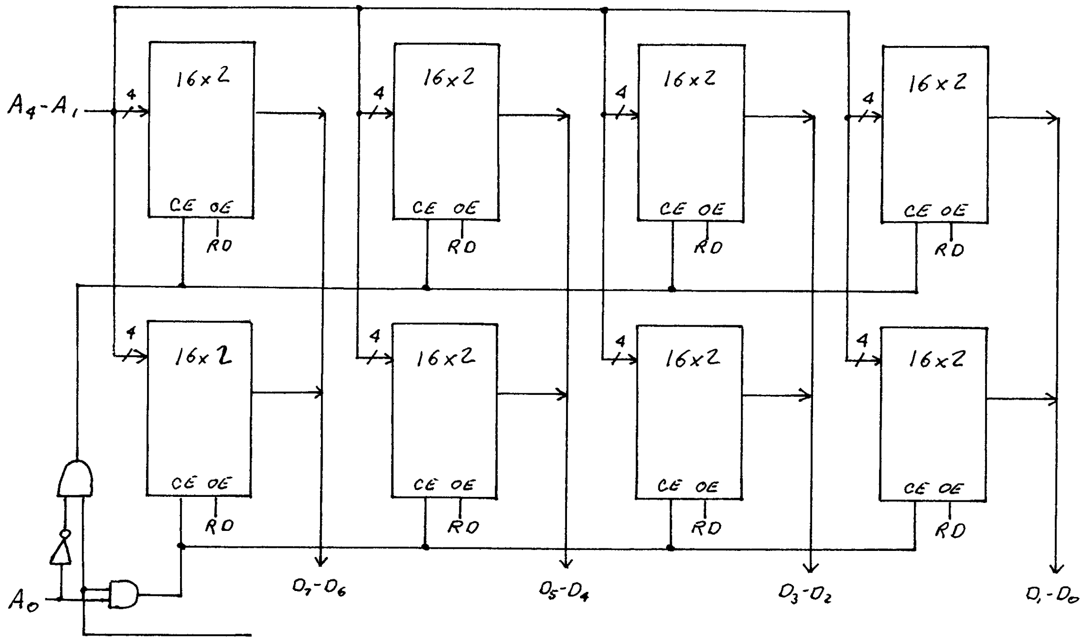


2. Design a 32x 8 memory subsystem with high-order interleaving using 16x2 memory chips for a computer system with an 8-bit address bus.

Design a 32x 8 memory subsystem with high-order interleaving using 16x2 memory chips for a computer system with an 8-bit address bus

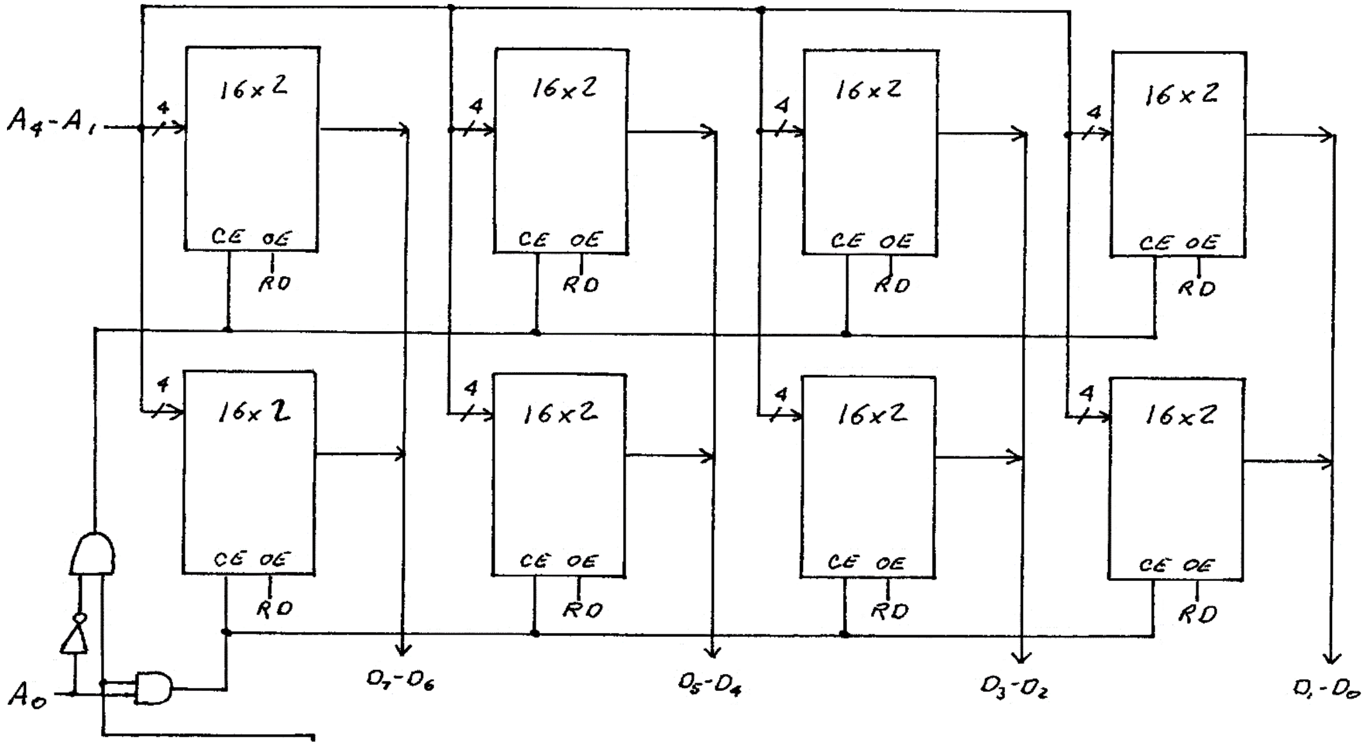


3. Design a 32x8 memory subsystem with low order interleaving using 16x2 memory chips for a computer system with an 8-bit address bus.



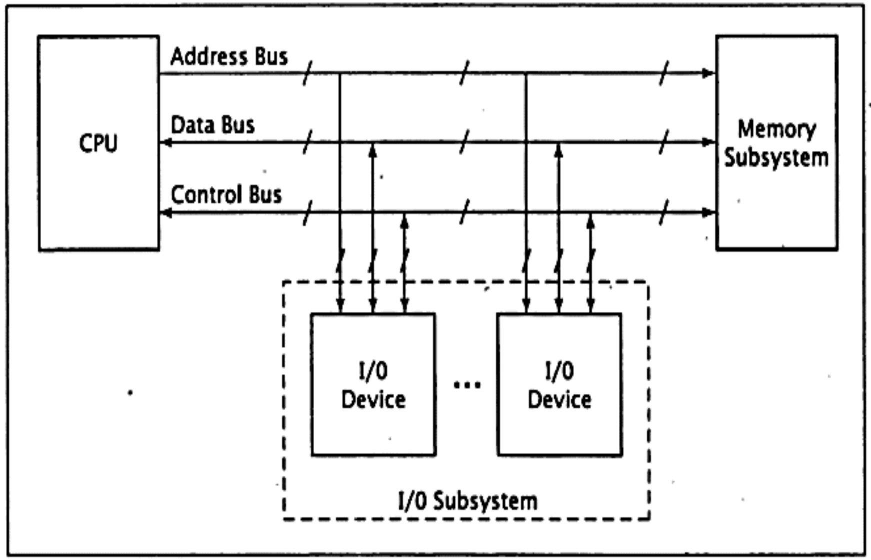
A_4-A_1
4
16x2
CE OE
RD
A_0
D_7-D_6
D_5-D_4
D_3-D_2
D_1-D_0

- **2. Design a 32x8 memory subsystem with low order interleaving using 16x2 memory chips for a computer system with an 8-bit address bus.**



6. I/O subsystem Organization and Interfacing

- The I/O subsystem is treated as an independent unit in the computer.
- The CPU initiates I/O command generally as Read, Write, Scan, etc.
- As shown in figure:
  - The **I/O device** is **connected to** the computer system **address, data and control buses**.
  - **Each I/O device includes I/O circuitry** that interacts with the buses.



Input device

- The data from the input device goes to the tri-state buffer.
- **When** the values on the **address bus and control bus are correct,** the **buffers are enabled and data passes onto the data bus.**
- The CPU, then can read these data.
- The key to this design is the enable logic.
- For the input device, the read signal (RD) should be asserted.
- Figure (b) shows the enable logic for an input device at address 11110000 with 8-bit address and control signals RD and IO/$\overline{M}$ (IO/Memory).

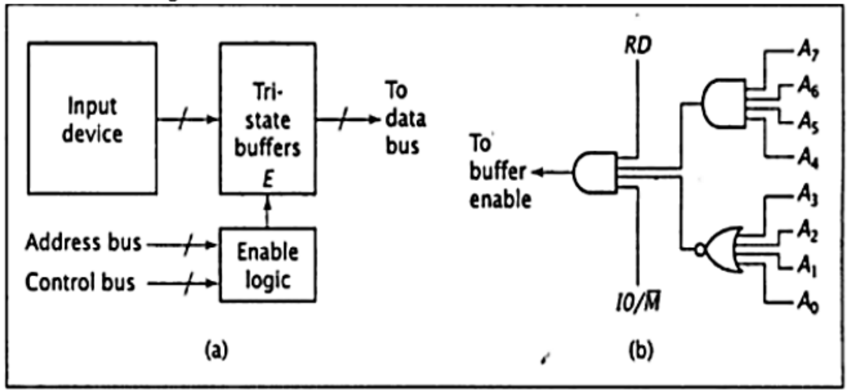


Figure: An Input device: a) with its interface, and b) the enable logic for the tri-state buffers

Output Device:

- For the interface circuit for an output device, the tri-state buffer is replaced by a register.
- **Only the device with the correct address will read it in.**
- The Load logic (LD) plays the role of the enable logic as in the input device.
- **When** this **logic receives the correct address and control signals**, it **asserts the LD signal** of the register, causing it to **read data from the system's data bus**.
- **The output device can then read the data from the register** at its leisure while the CPU performs other tasks.
- The logic to generate the load signal for an output device at address 1111 0000 is shown in figure (b).

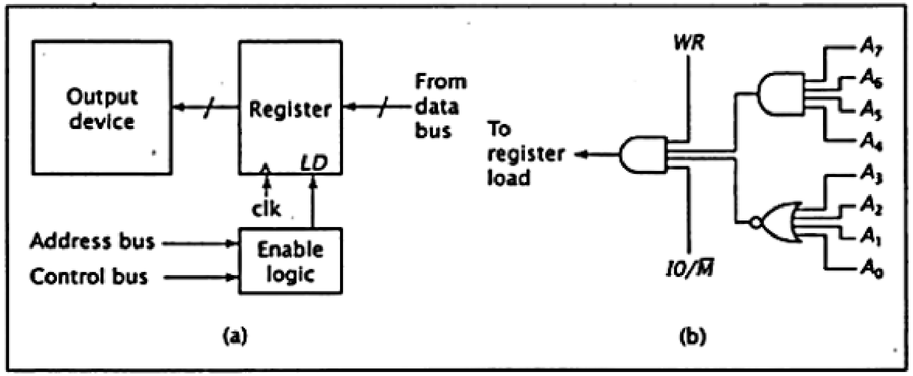


Figure: An Output device: a) with its interface and ,b) the load logic for the register

Bidirectional IO device

- Some devices are used for both input and output. A personal computer's hard disk drive falls into this category.
- Such a device requires a combined interface that is essentially two interfaces, one for input and the other for output.
- Some logic elements, such as the gates that check the address and the address bus, can be used to generate both the buffer enable and register load signals.
- Figure below shows a combined I/O interface for address 1111 0000.

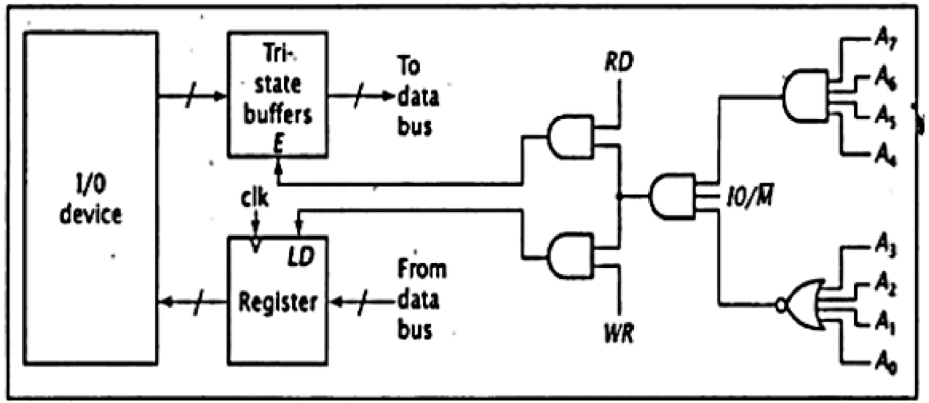
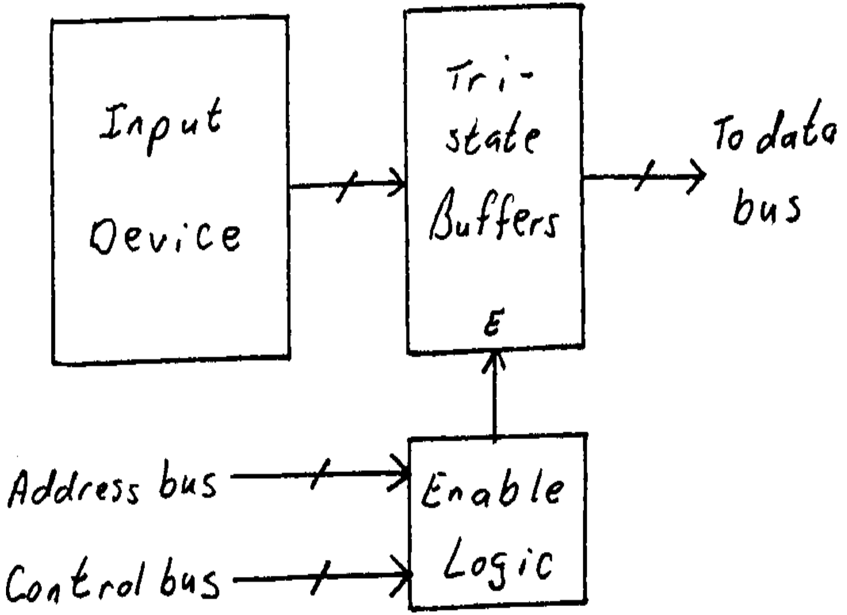
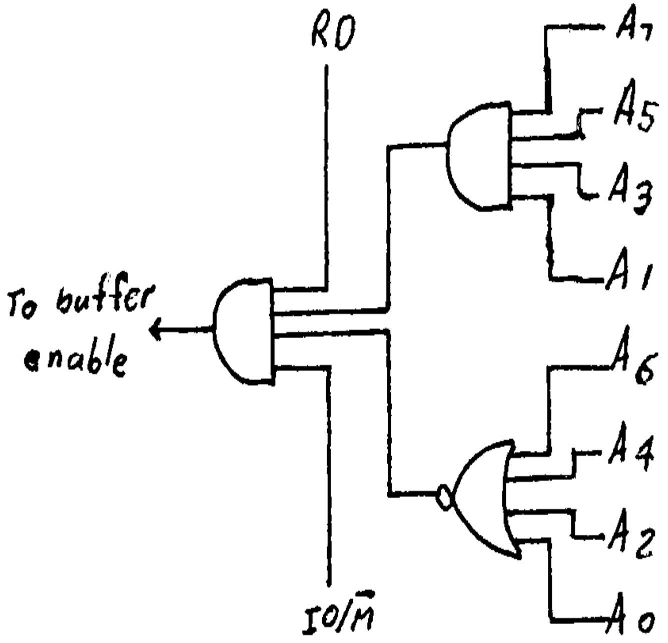


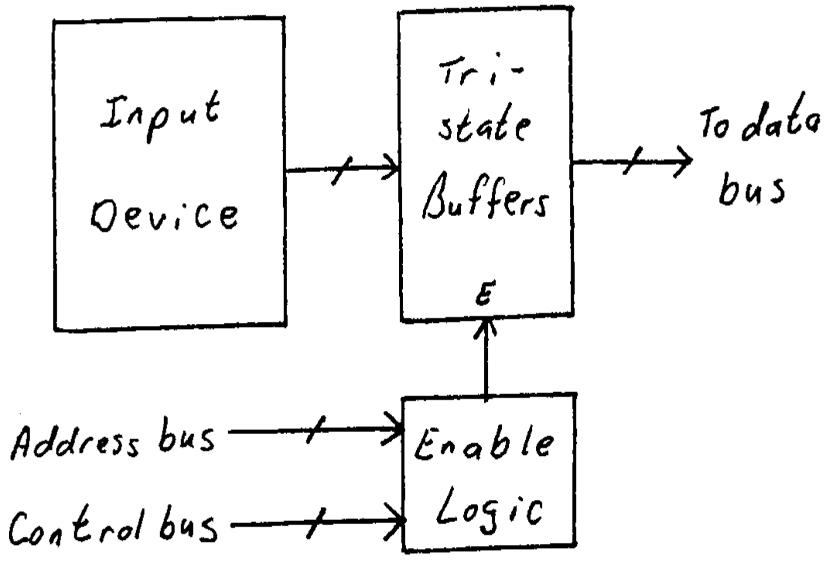
Figure: An bidirectioinal Input/Output device with its interface and enable/load logic

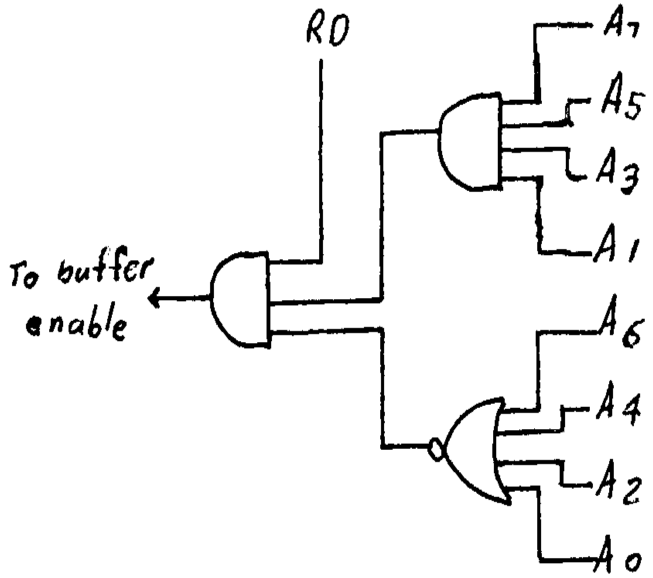
1. Design an interface for an input device which has binary address 1010 1010. Its computer system uses isolated I/0.



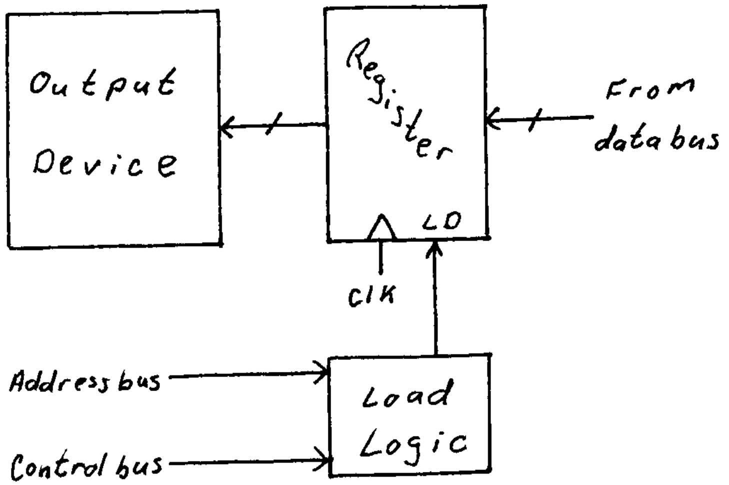


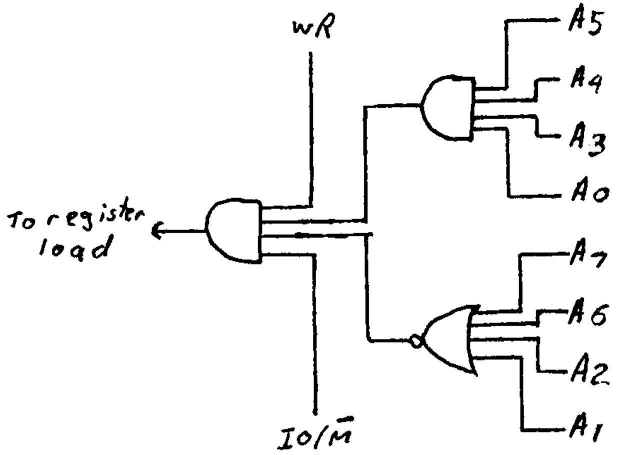
2. Design an interface for an input device which has binary address 1010 1010. Its computer system uses memory mapped I/O.



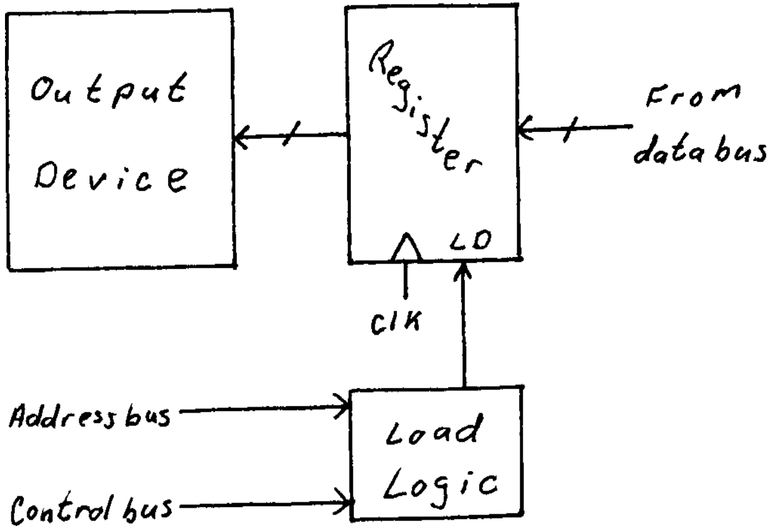


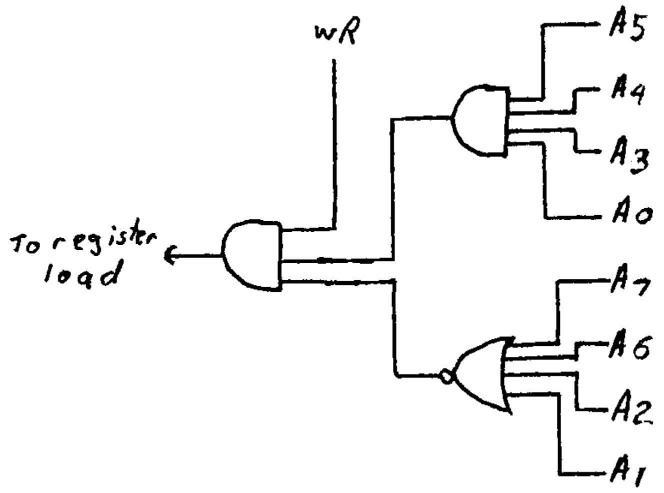
3. Design an interface for an output device which has binary address 0011 1001. Its computer system uses isolate I/O.



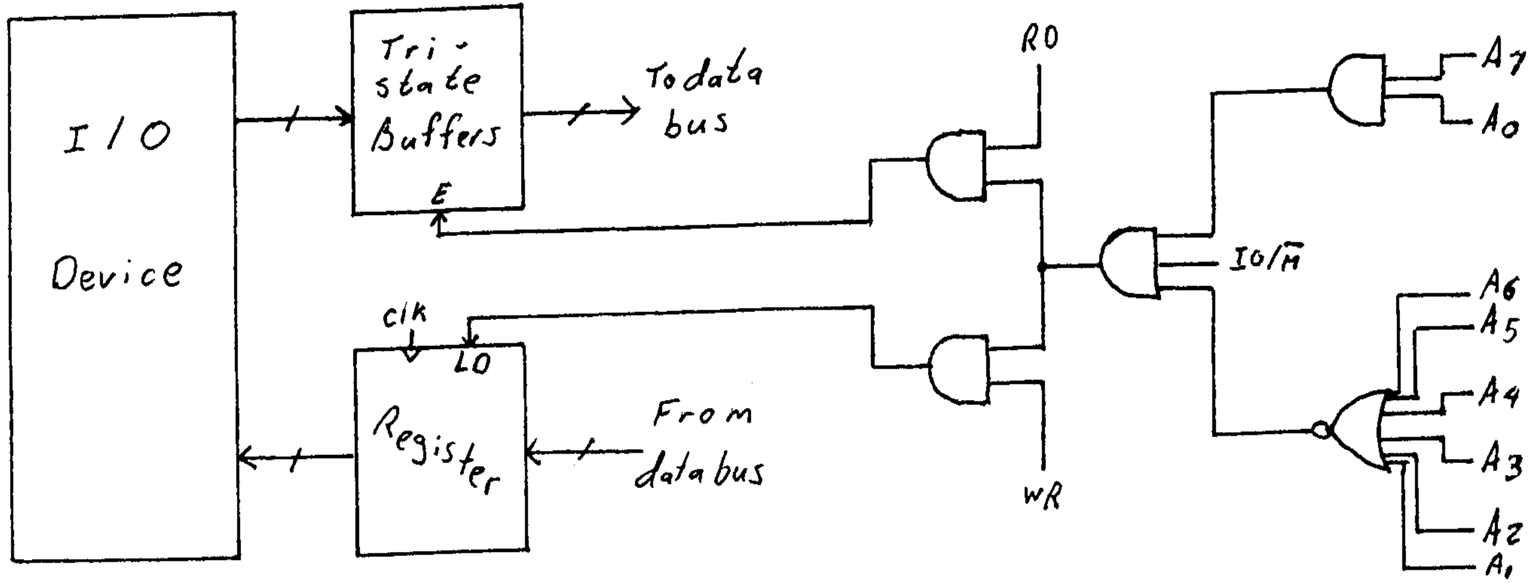


4. Design an interface for an output device which has binary address 0011 1001. Its computer system uses memory mapped I/O.



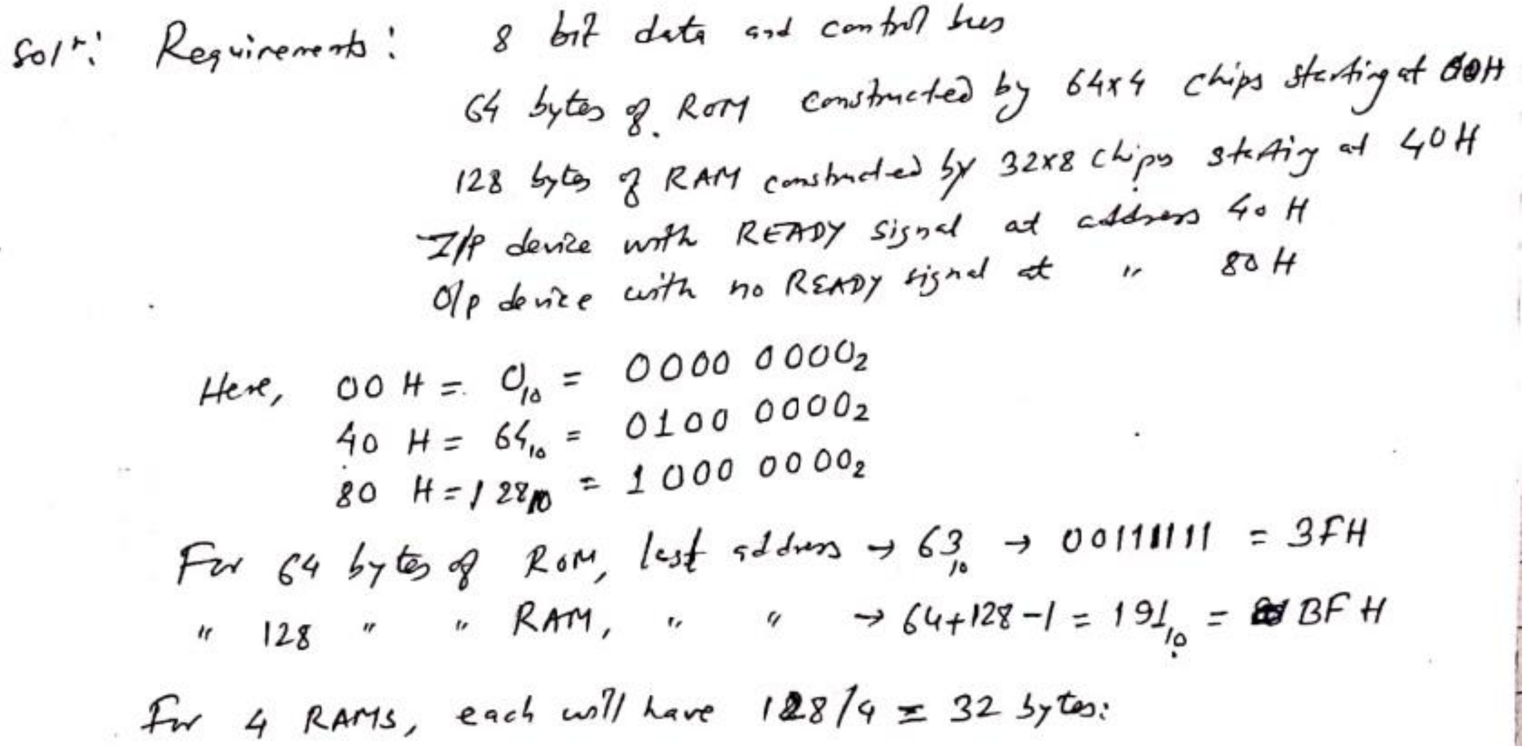


5. Design an interface for a bidirectional Input/Output device that has binary address 1000 0001. Its computer system uses Isolated I/O.



A computer system with 8-bit data bus and 8-bit control bus uses isolated I/O. It has 64 bytes of ROM starting at 00H constructed using 64 x 4 chips; 128 bytes of RAM, starting at address 40H constructed using 32 x 8 chips; an input device with READY signal at address 40H; and an output with no READY signal at address 80H. Show the design for this system. Include all enable and load logic.

A computer system with 8-bit data bus and 8-bit control bus uses isolated I/O. It has 64 bytes of ROM starting at 00H constructed using 64 x 4 chips; 128 bytes of RAM, starting at address 40H constructed using 32 x 8 chips; an input device with READY signal at address 40H; and an output with no READY signal at address 80H. Show the design for this system. Include all enable and load logic.



Sol^n: Requirements: 8 bit data and control bus

64 bytes of ROM constructed by 64x4 chips starting at 00H

128 bytes of RAM constructed by 32x8 chips starting at 40H

I/P device with READY signal at address 40 H

O/p device with no READY signal at " 80 H

Here, $00\,H = 0_{10} = 0000\ 0000_2$

$40\,H = 64_{10} = 0100\ 0000_2$

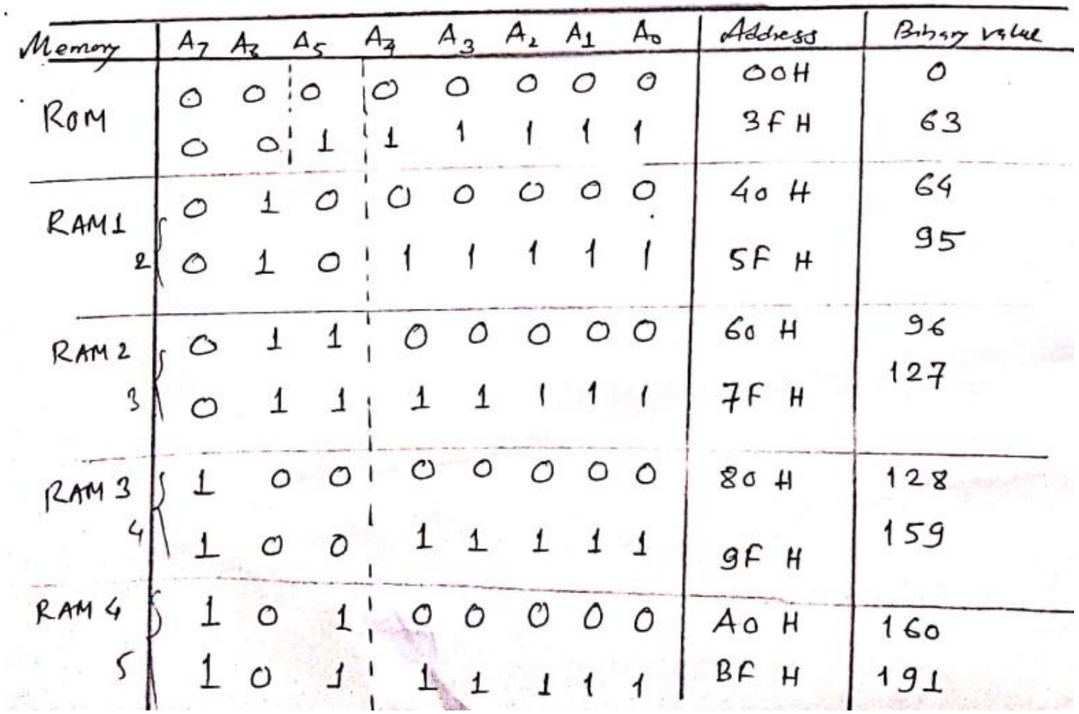
$80\,H = 128_{10} = 1000\ 0000_2$

For 64 bytes of ROM, last address $\rightarrow 63_{10} \rightarrow 00111111 = 3FH$

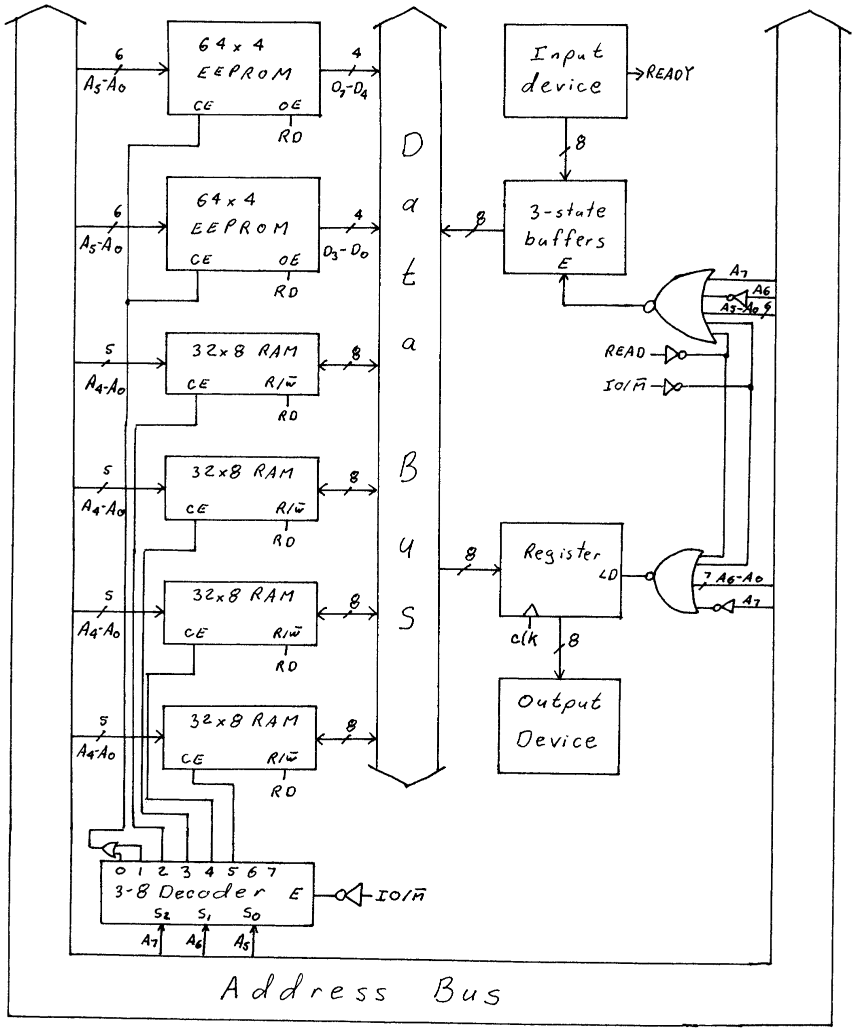
" 128 " " RAM, " " $\rightarrow 64+128-1 = 191_{10} = BF\,H$

For 4 RAMS, each will have $128/4 = 32$ bytes:

A computer system with 8-bit data bus and 8-bit control bus uses isolated I/O. It has 64 bytes of ROM starting at 00H constructed using 64 x 4 chips; 128 bytes of RAM, starting at address 40H constructed using 32 x 8 chips; an input device with READY signal at address 40H; and an output with no READY signal at address 80H. Show the design for this system. Include all enable and load logic.



Memory	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	Address	Binary value
ROM	0	0	0	0	0	0	0	0	00H	0
	0	0	1	1	1	1	1	1	3F H	63
RAM 1	0	1	0	0	0	0	0	0	40 H	64
2	0	1	0	1	1	1	1	1	5F H	95
RAM 2	0	1	1	0	0	0	0	0	60 H	96
3	0	1	1	1	1	1	1	1	7F H	127
RAM 3	1	0	0	0	0	0	0	0	80 H	128
4	1	0	0	1	1	1	1	1	9F H	159
RAM 4	1	0	1	0	0	0	0	0	A0 H	160
5	1	0	1	1	1	1	1	1	BF H	191



64 x 4 EEPROM
A_5-A_0
6
4
D_7-D_4
CE
OE
RD
64 x 4 EEPROM
D_3-D_0
32x8 RAM
A_4-A_0
5
8
R/$\bar{W}$
Data Bus
Input device
READY
3-state buffers
E
A_7
A_6
A_5-A_0
READ
IO/$\bar{M}$
Register
LD
clk
A_6-A_0
7
Output Device
0 1 2 3 4 5 6 7
3-8 Decoder
S_2 S_1 S_0
A_5
Address Bus

End of chapter