Chapter 02
Computer Organization

Er. Hari K.C.

Department of Software Engineering

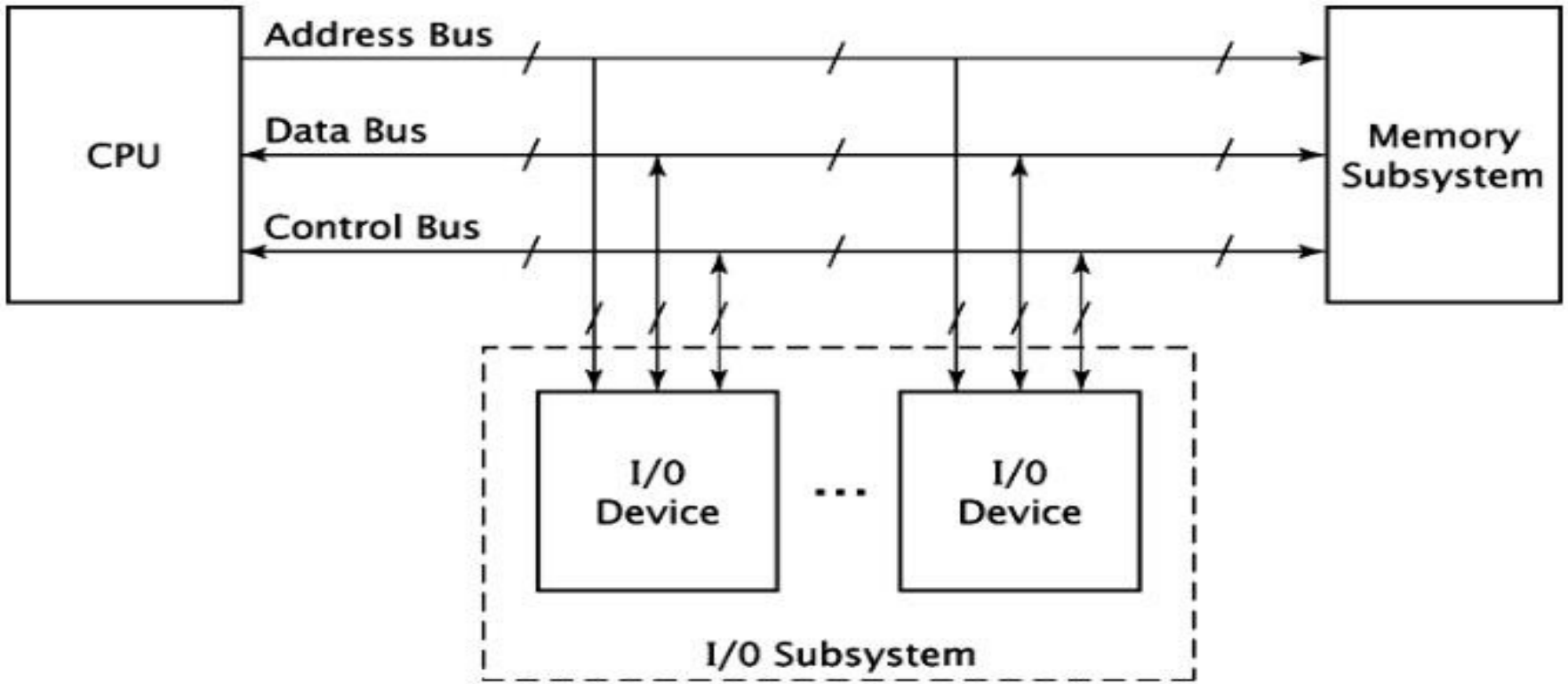
Gandaki College of Engineering and Science

Computer Organization

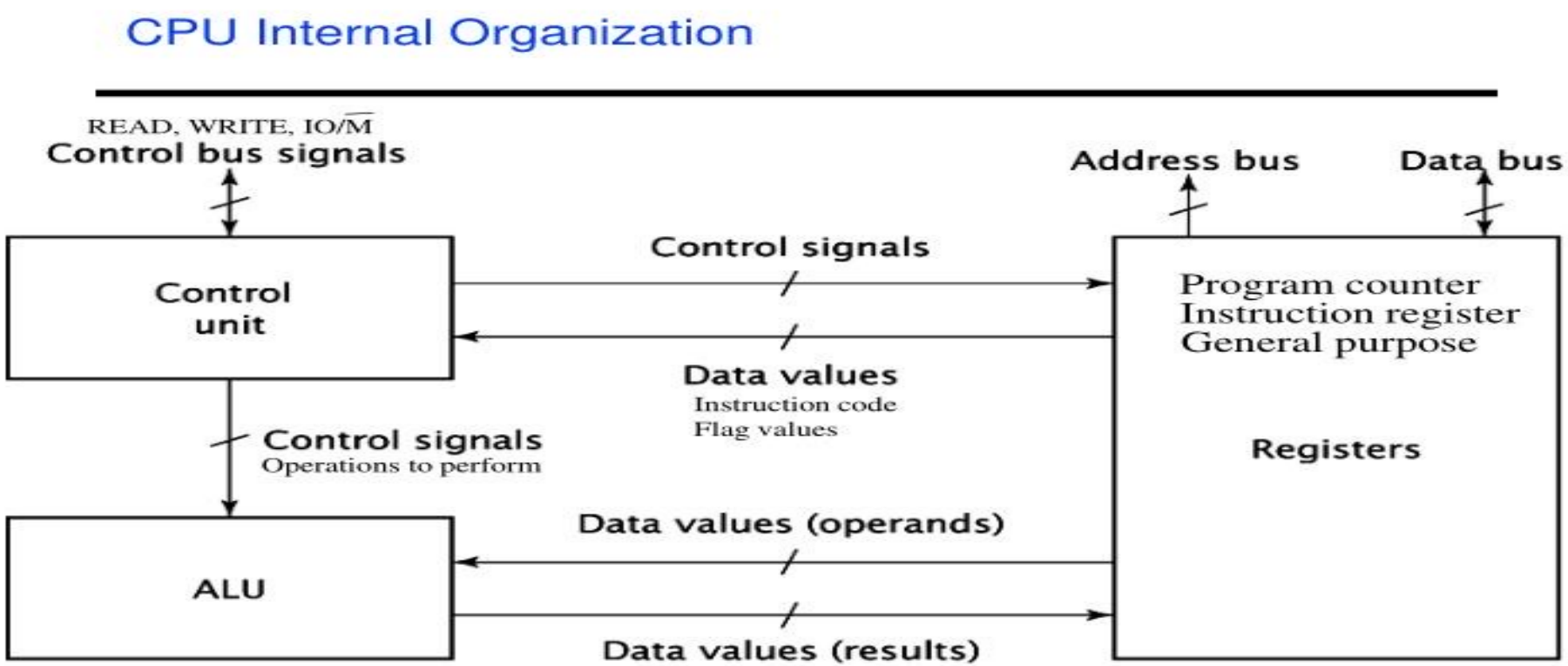
- It refers to the operational units and their interconnection that realize the architectural specification.
- It includes hardware details, control signals, interface between computer and peripherals, memory technology
- It is an architectural design issue whether a computer have multiply instructions
- It is an organizational design issue whether the multiply instruction is implemented by special multiply unit or by repeated use of adder.

Computer Architecture	Computer Organization
It refer to the attributes that are visible to the programmer.	Computer Organization is concerned with the structure and behaviour of a computer system as seen by the user.
It acts as the interface between hardware and software.	It deals with the components of a connection in a system.
Computer Architecture helps us to understand the functionalities of a system.	Computer Organization tells us how exactly all the units in the system are arranged and interconnected.
A programmer can view architecture in terms of instructions, addressing modes and registers.	Whereas Organization expresses the realization of architecture.
While designing a computer system architecture is considered first.	An organization is done on the basis of architecture.
Computer Architecture deals with high-level design issues.	Computer Organization deals with low-level design issues.
Architecture involves Logic (Instruction sets, Addressing modes, Data types, Cache optimization)	Organization involves Physical Components (Circuit design, Adders, Signals, Peripherals)

Generic Computer Organization



CPU organization



Central Processing Unit (CPU)

- The computer system is nothing without the Central processing Unit so, it is also known as the brain or heat of computer.
- The CPU is an electronic hardware device which can perform different types of operations such as arithmetic and logical operation.
- The CPU contains three parts: the arithmetic logic unit , control unit and registers.
- The control unit (CU) controls all the activities or operations which are performed inside the computer system. It receives instructions or information directly from the main memory of the computer.
- The arithmetic and logical unit is the combinational digital electronic circuit that can perform arithmetic operations on integer binary numbers.It presents the arithmetic and logical operation. The outputs of ALU will change asynchronously in response to the input. The basic arithmetic and bitwise logic functions are supported by ALU.

- **Registers:** They are processor memory. They are made of flipflops. They store data temporarily during execution. Types : special purpose and general purpose.

Input Devices

- The user provides the set of instruction or information to the computer system with the help of input devices such as the keyboard, mouse, scanner, etc.
- The data representation to the computer system is in the form of binary language after that the processor processes the converted data. The input unit implements the data which is instructed by the user to the system.

Output Devices

- The output devices produce or generate the desired result according to our input, such as a printer, monitor, etc. These devices convert the data into a human-readable form from binary code.
- The computer system is linked or connected to the outside world with the help of output devices. The primary examples of output devices are a printer, projector, etc. These devices have various features which are given below:
- These devices receive or accept the data in the binary form.
- The output devices convert the binary code into the human-readable form.
- These devices produce the converted result and show to the user.

Main Memory

- The Random Access Memory is the main memory of the computer system, which is known as RAM. The main memory can store the operating system software, application software, and other information. The Ram is one of the fastest memory, and it allows the data to be readable and writeable.

Bus:

Bus is the group of communication lines to transfer signals and data among computer components.

They are conducting wires.

There are varieties of bus such as local bus, system bus, input-output bus.

System bus are data bus, address bus and control bus.

Data bus carry data from processor to memory or vice versa

Address bus carry address from processor to memory.

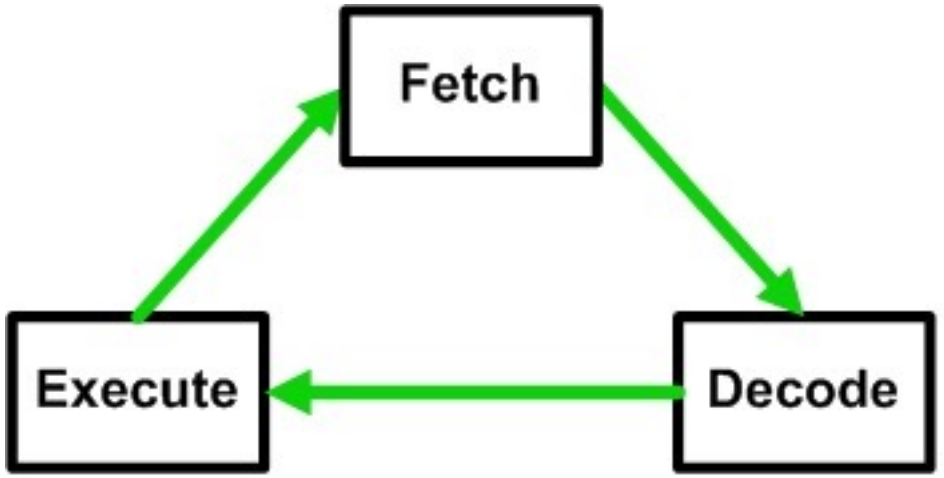
Control bus carry control and timing signals between computer components

Instruction cycle

- The time required to complete the execution of instruction.
- It consist of fetch cycle, decode cycle and execute cycle.
- Fetch cycle is used to read the instruction from the memory.
- Decode cycle is used to decode and identify the function of instruction.
- Execute cycle is used to run the instruction

such that performing the required operation,

storing the result in registers or in memory.



Instruction cycle state diagram

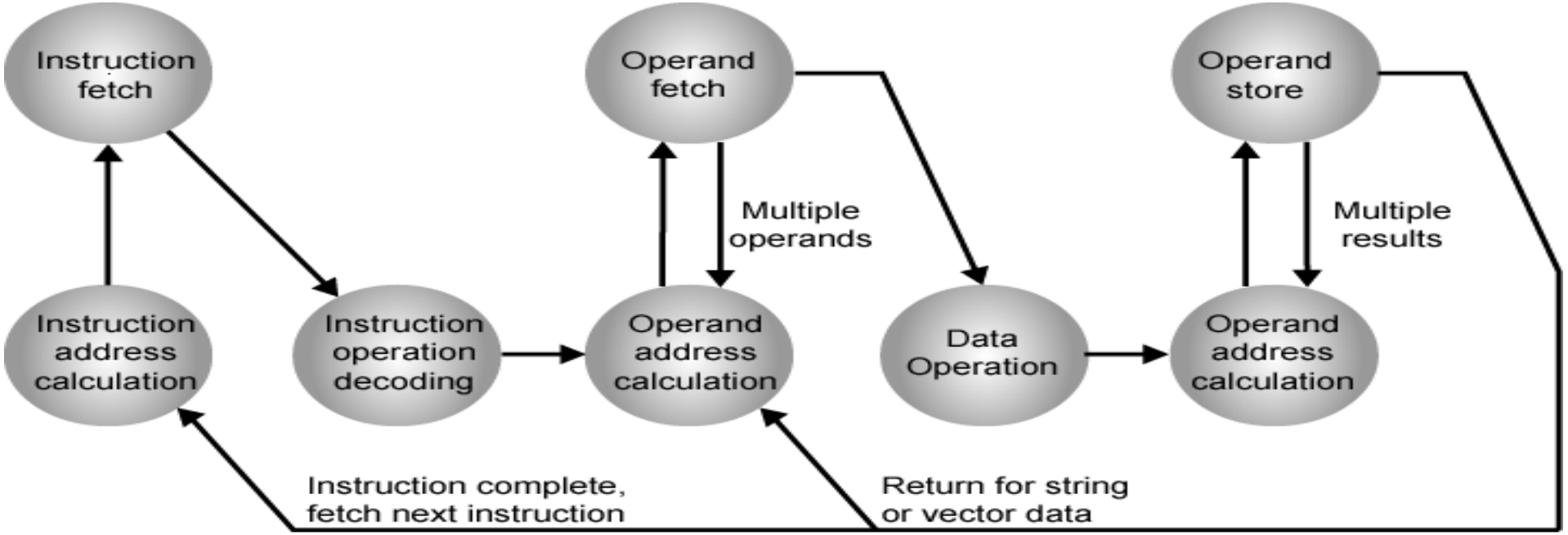
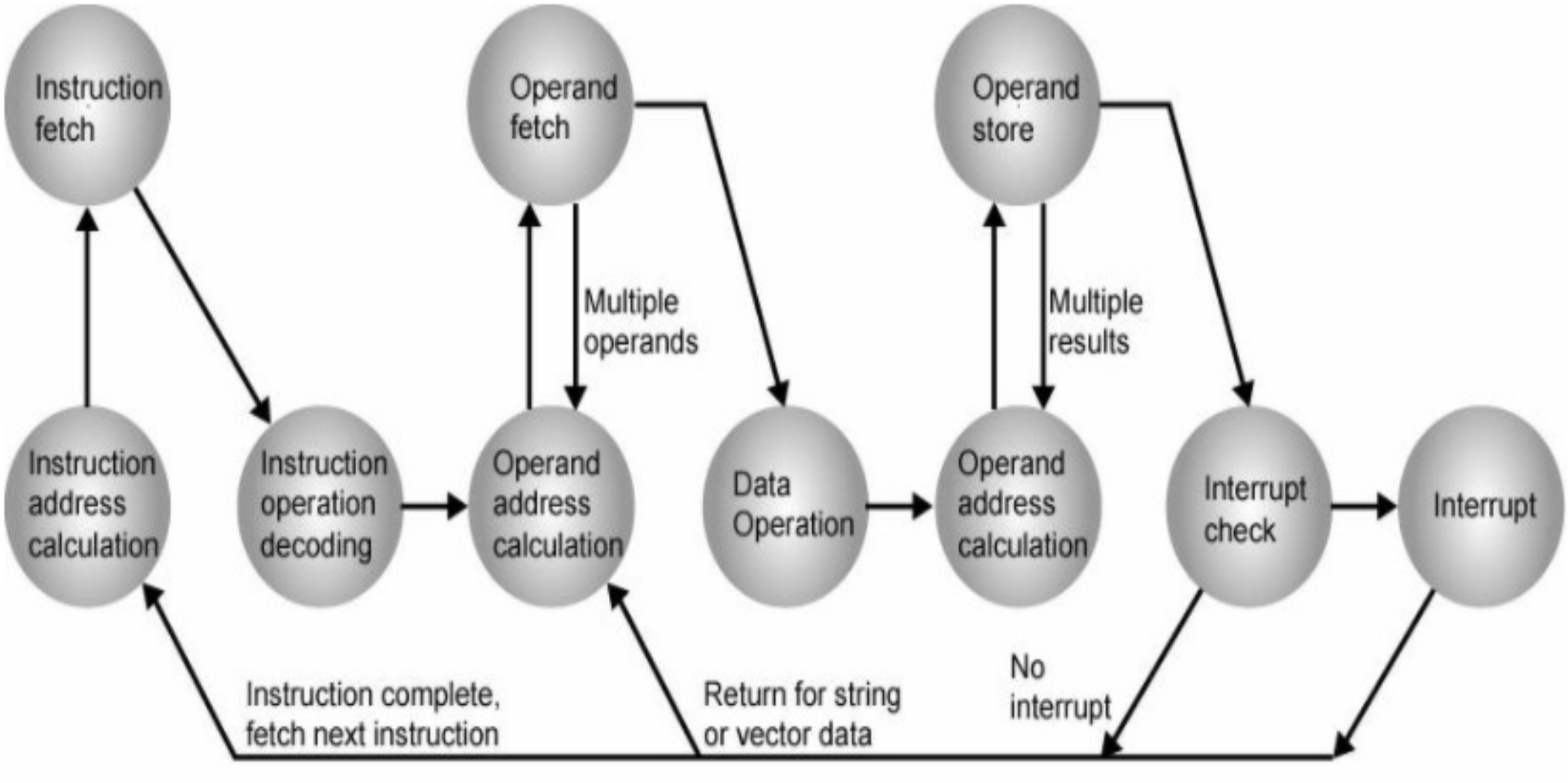


Figure : Instruction Cycle State Diagram


Instruction fetch
Operand fetch
Operand store
Multiple operands
Multiple results
Instruction address calculation
Instruction operation decoding
Operand address calculation
Data Operation
Operand address calculation
Interrupt check
Interrupt
Instruction complete, fetch next instruction
Return for string or vector data
No interrupt

- **Instruction address calculation (iac):**

Determine the address of the next instruction to be executed. Usually, this involves adding a fixed number to the address of the previous instruction. For example, if each instruction is 16 bits long and memory is organized into 16-bit words, then add 1 to the previous address. If, instead, memory is organized as individually addressable 8-bit bytes, then add 2 to the previous address.

- **Instruction fetch (if):**

Read instruction from its memory location into the processor.

- **Instruction operation decoding (iod):**

Analyze instruction to determine type of operation to he performed and operand(s) to be used.

- **Operand address calculation (oac):**

If the operation involves reference to an operand in memory or available via I/O. then determine the address of the operand.

- **Operand fetch (of):**

Fetch the operand from memory or read it in from I/O,

- **Data operation (do):**

Perform the operation indicated in the instruction.

- **Operand store (os):**

Write the result into memory or out to I/O

Memory subsystem

Types of Memory

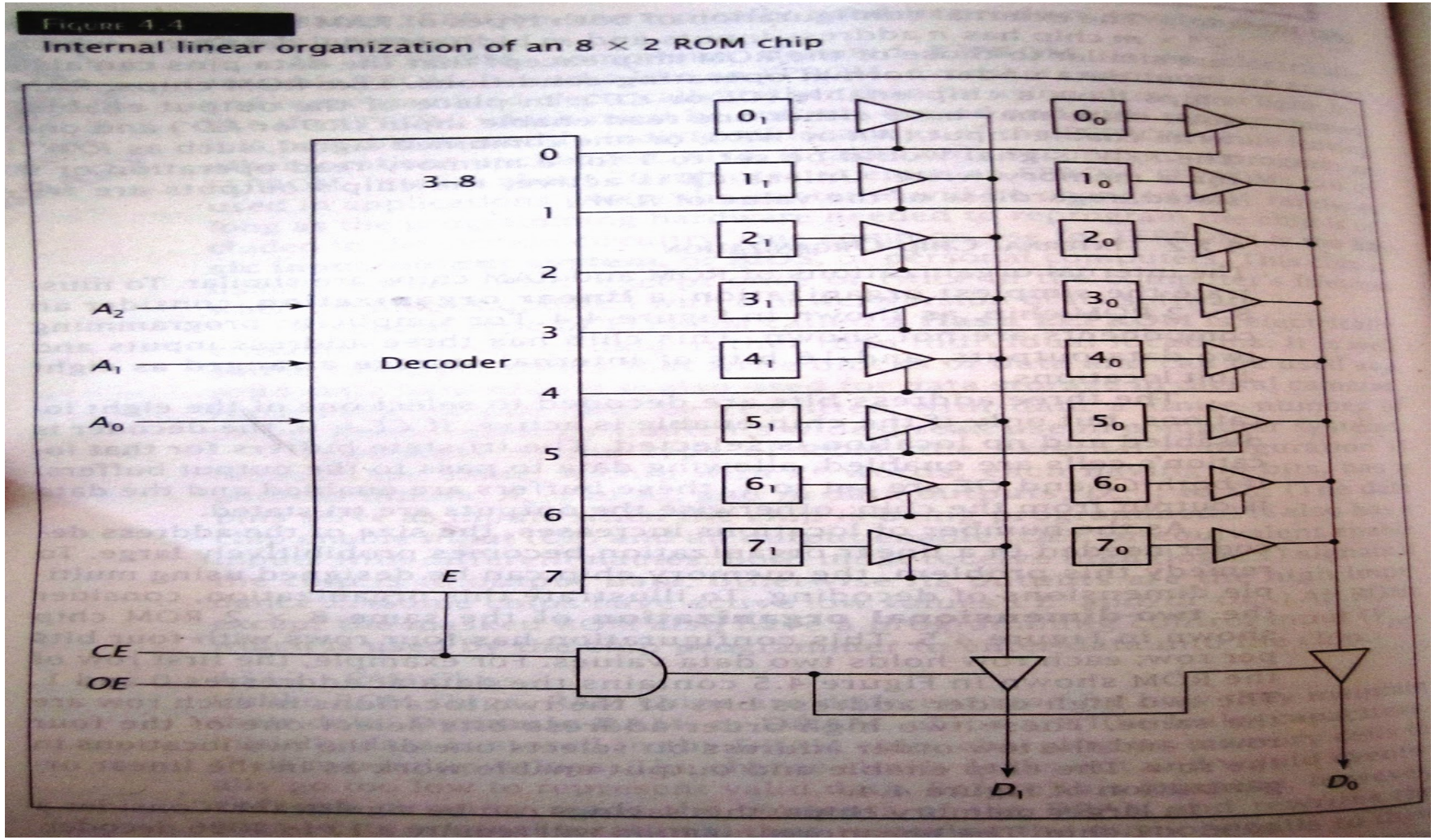
- Read Only Memory (ROM)
  - Masked ROM: programmed with data as chip is fabricated
  - Programmable Read Only Memory (PROM): can be programmed by user, but only once
  - Erasable PROM (EPROM): content can be erased and reprogrammed
  - Electrically Erasable PROM (EEPROM): can modify individual locations on the EEPROM
- Chip with 2^n words, each having m bits, has n address inputs, A_{n-1} to A_0, and m data outputs, D_{m-1} to D_0
- D is used as input to program chip
- Has chip enable (CE), output enable (OE), and program control input (V_{pp})
- CE must be active for something to happen

Types of Memory

- Random Access Memory (RAM)
  - Dynamic RAM (DRAM): like leaky capacitors, if not refreshed will eventually loose data. Used for primary memory.
  - Static RAM (SRAM): Does not have to be refreshed. Faster than DRAM but more expensive. Used for cache memory.
- Each 2^n X m chip has n address inputs and m bidirectional data pins
- Chips have chip enable (CE or CE')
- Chips may have either read enable input (RD or RD') and write enable (WR or WR') or one combined signal, such as R/W' . R/W' would be set to 1 for read and 0 for write.
- CE must be active for read or write to happen

Memory subsystem

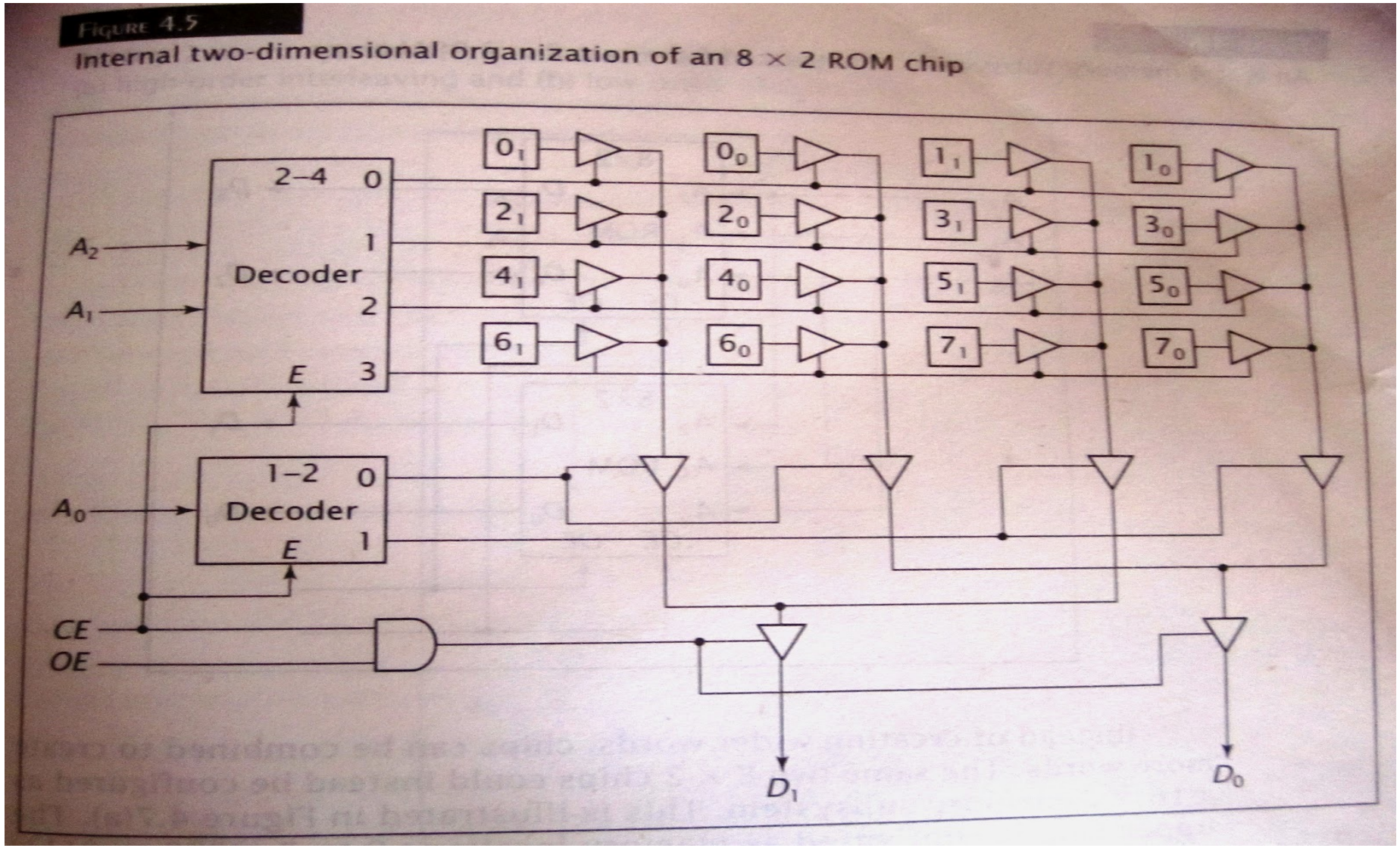
1) Linear organization of ROM:



- 8* 2 ROM
- 3 address line inputs(A2, A1,A0) and two data line outputs(D1,D0) as shown in figure.
- 16 bits of internal storage arranged as eight 2 bit location

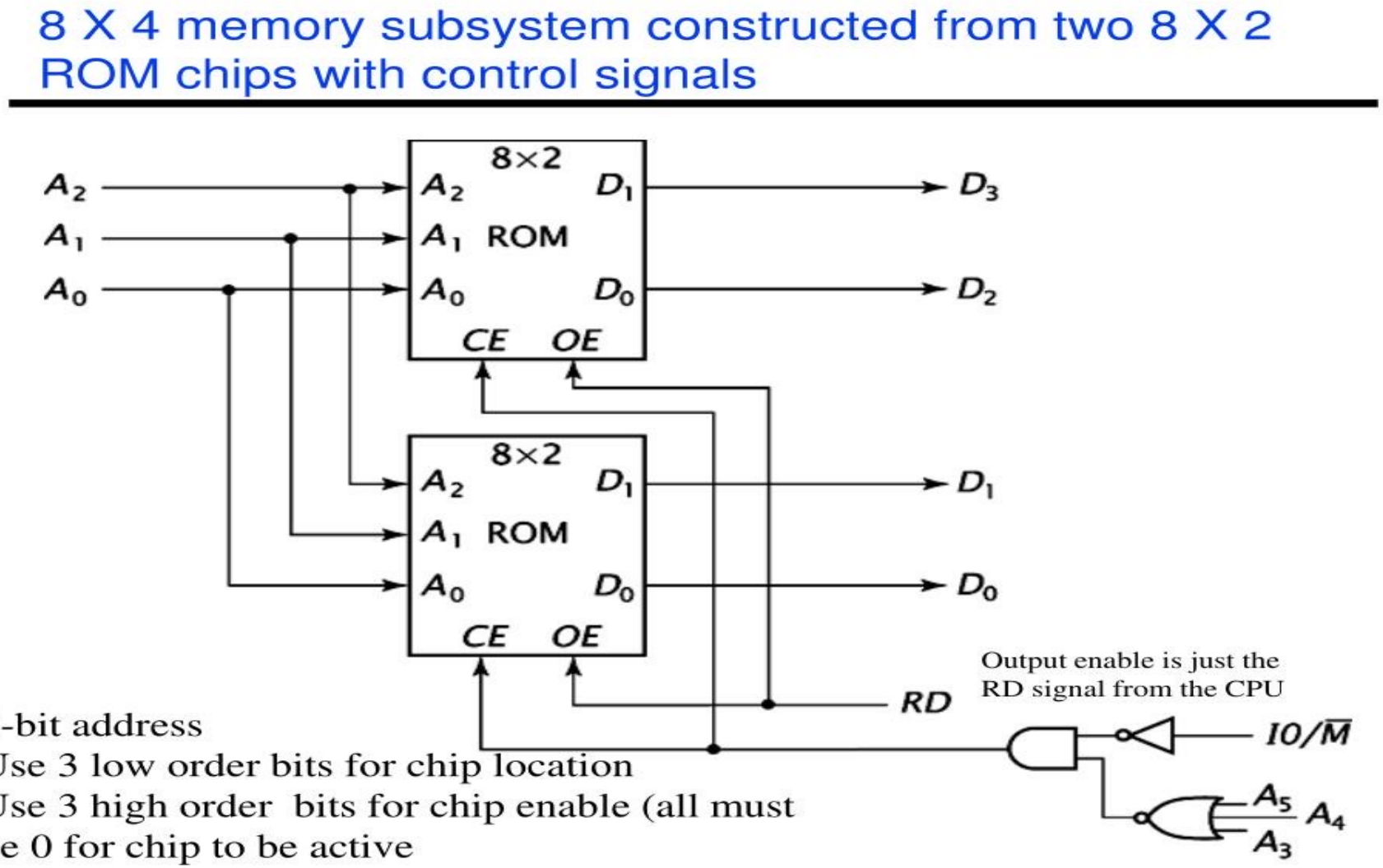
- The three address bits are decoded to select one of the eight location but CE(chip enable) must be active. CE =1.
- if CE =0 , decoder is disabled and no location is selected.
- The tristate buffers for that location cell are enabled allowing data to pass to output buffers
- if CE =1 and OE = 1, the buffers are enabled and data is output to data lines otherwise output are not available.

Internal two dimensional organization of 8*2 ROM



Q) How can a 8*4 ROM can be constructed from two 8*2 ROM chips.

Solution:



The two 8*2 ROM chips can be combined to create an 8*4 ROM.

Both chips receive the same three address inputs form the bus as well as same chip enable signal and output enable signal.

The data pins of first chip is connected to bits 3 and 2 of data bus and those of other chip are connected to bits 1 and 0.

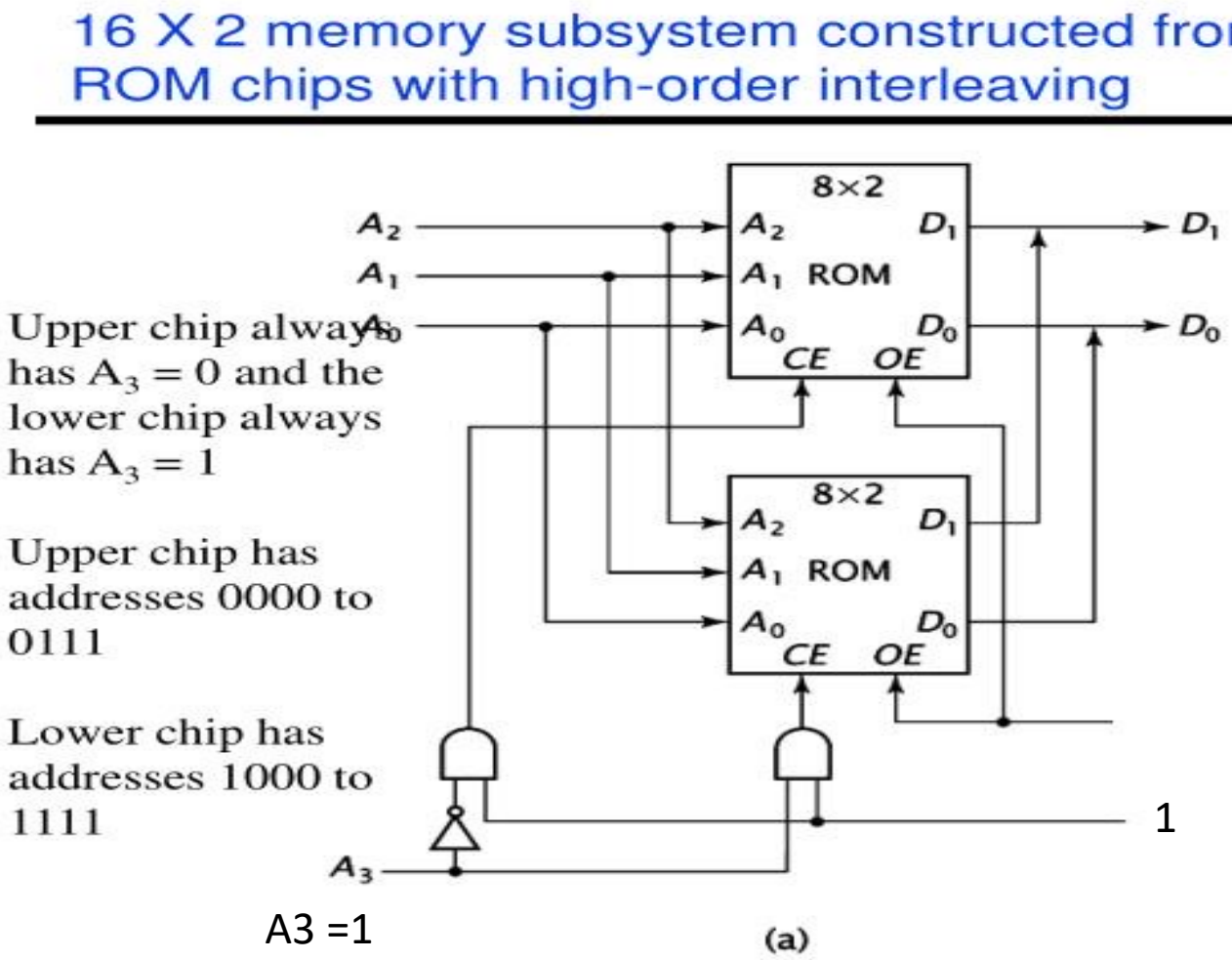
When CPU reads data , it places address on address bus. Both chips read in address bits A2, A1 and A0 and perform their internal decoding

if CE and OE are activated , the chips output their data on 4 bits of data bus.

since address and enable signals are same for both chips , either both chips or neither chips is active at any time.

so they act just as an single 8*4 chip.

16* 2 memory constructed from two 8*2 ROM with high order interleaving



A3	A2	A1	A0
0	0	0	0
.			
.			
0	1	1	1

- The upper chip is configured as memory location 0 to 7 (0000 to 0111) and lower chip as location 8 to 15 (1000 to 1111)
- The upper chip has always A3 =0 and lower chip has A3 = 1.
- When A3 = 0, the upper chip is enabled and lower chip is disabled.
- When A3 =1, the lower chip is enabled and upper chip is disabled.
- Both chips corresponds to same data bits , both are connected to D1 and D0 of data bus.

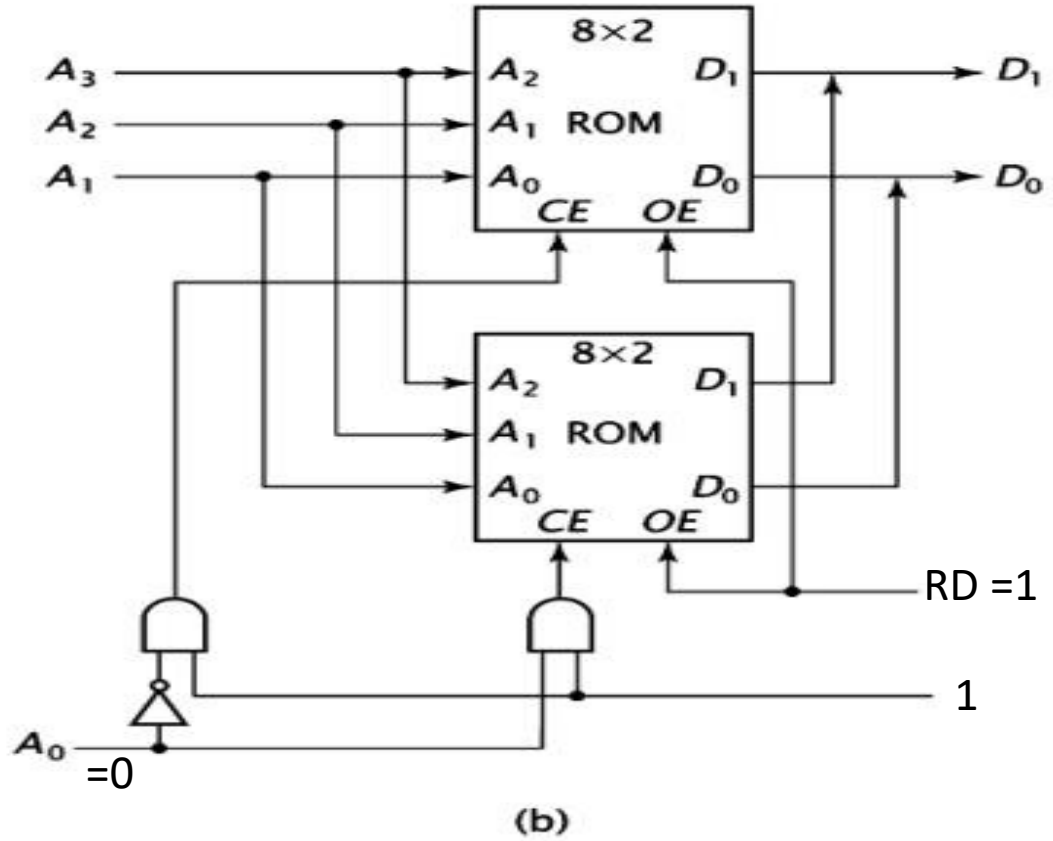
16* 2 memory constructed from two 8*2 ROM with low order interleaving

16 X 2 memory subsystem constructed from two 8 X 2 ROM chips with low-order interleaving

Upper chip enabled for $A_0 = 0$, or addresses 0, 2, 4, 6, 8, 10, 12, 14

Lower chip enabled for $A_0 = 1$, or addresses 1, 3, 5, 7, 9, 11, 13, 15

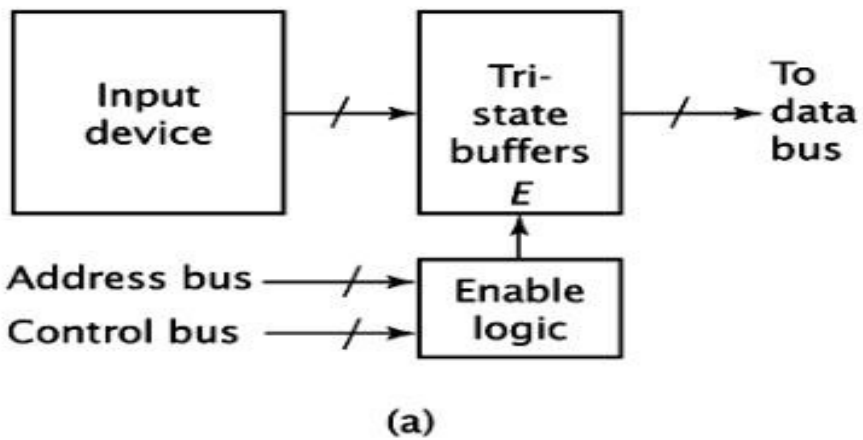
Low order interleaving offers speed advantages for pipelined memory access



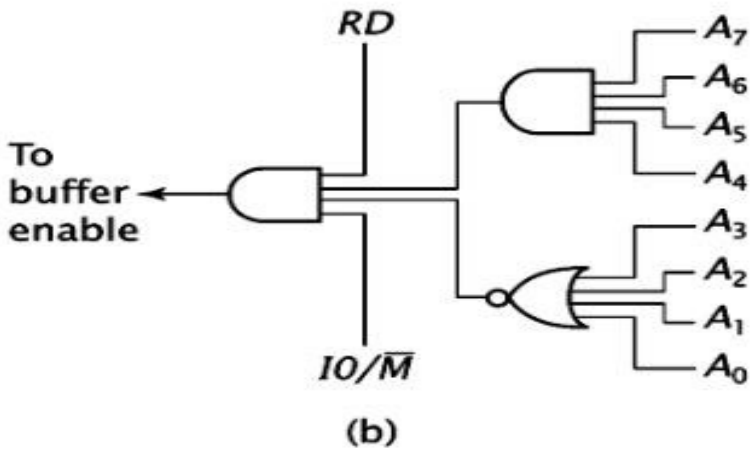
Input/output subsystem

- Input device

An Input Device



Interface

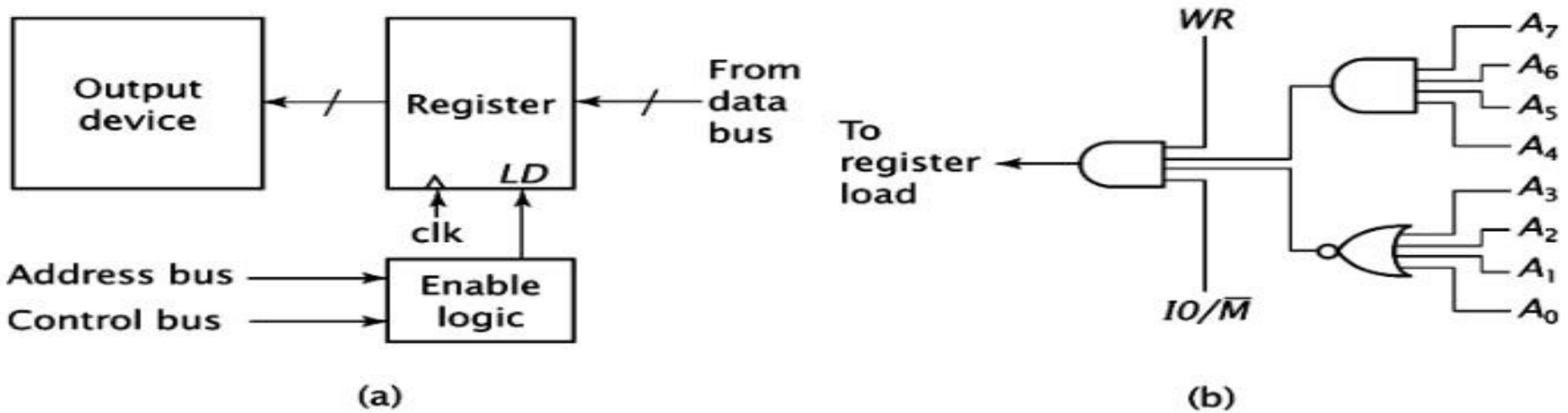


Enable logic for tri-state buffers
I/O device address is 11110000

- The data from input device goes to tristate buffers.
- When the values on address bus and control bus are correct, the buffers are enabled and data passes to data bus.
- Then CPU read the data.
- Each I/O device has unique address . The enable logic must not enable the buffers unless it receives the correct address from address bus.
- It must also get correct control signals from control bus.
- The tristate buffers are used in input device interfacing to make sure that no more than one devices write data to the data bus at any time.

Output subsystem

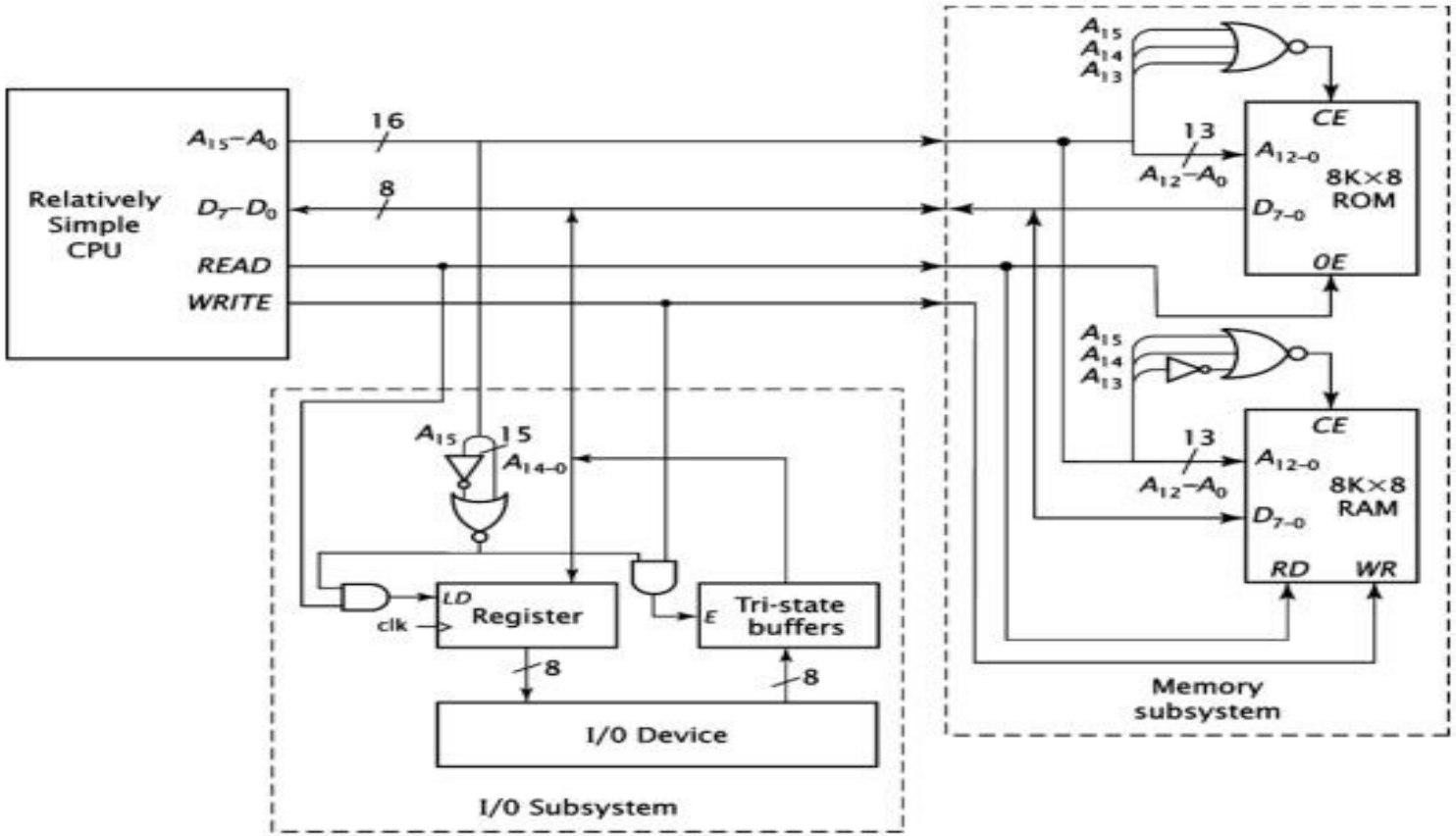
An Output Device



Tri-state buffers are not need for output because the data is put on the data bus and only the device at the address buss address will read the data from the data bus

A relatively simple computer

A relatively simple computer, final design



End of chapter 02