UNIT 1 - Chapter 2

MICROPROCESSOR ARCHITECTURE AND MICROCOMPUTER SYSTEMS

MICROPROCESSOR ARCHITECTURE AND ITS OPERATIONS

- Microprocessor is a programmable logic device, designed with registers, flip-flops and timing elements.
- The microprocessor has a set of instructions for data manipulation and communication with peripherals.
- This process of data manipulation and communication is determined by the logic design of the microprocessor called Architecture.
- The various functions performed by microprocessor can be classified in three general catagories:
  - Microprocessor initiated operations
  - Internal operations
  - Peripheral (externally initiated) operations

Microprocessor-Initiated Operations and 8085 Bus organization

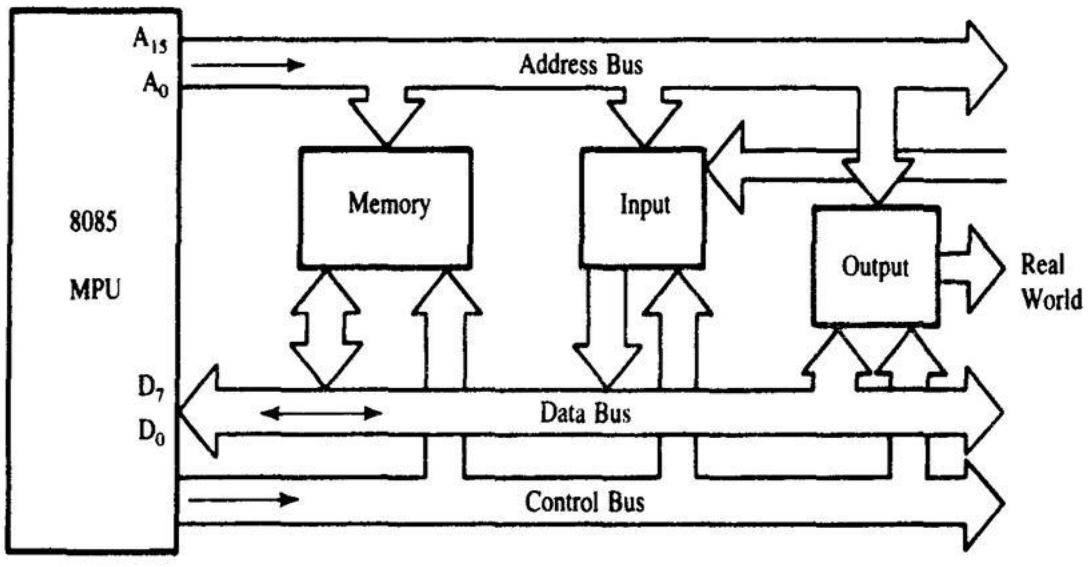
The MPU performs primarily 4 operations

- Memory Read: Reads data or instructions from memory.
- Memory Write: Writes data or instructions into memory.
- I/O Read: Accepts data from input device.
- I/O Write: Sends data to output device.

For all these operations MPU needs to communicate with I/O and memory. Hence has to perform following steps.

- Step 1: Identify the peripheral or the memory location (with its address)
- Step 2: Transfer binary information (data and instructions)
- Step 3: Provide timing and synchronization signals.

System Bus



- System bus is the communication path between the microprocessor and peripherals.
- It is a group of wires to carry bits.

Address Bus

- 16-bit unidirectional bus. i.e., 16 wires, each wire carrying one bit.
- Address carried by these bus lines tells memory to open the designated memory location. Data can be read or written on that memory location depending on control signal.
- A 16-bit binary number allows 2^{16} different memory locations to be addressed (64K bytes memory). i.e., from 0000 0000 0000 0000 to 1111 1111 1111 1111. (0000 H to FFFF H).
- To communicate with memory the microprocessor sends an address on the address bus. Example- for 0000 0000 0000 0011 sent to memory, location 0003H is selected.

Data Bus

- 8-bit bidirectional bus. i.e., 8 wires, each wire carrying one bit.
- Data bus is used to transmit data, i.e., information, results of arithmetic and logical operation etc. between memory and microprocessor
- Data bus also carries instructions from memory to microprocessor.
- Size of the data bus limits the number of possible instructions to 256. i.e., from 00H to FFH.

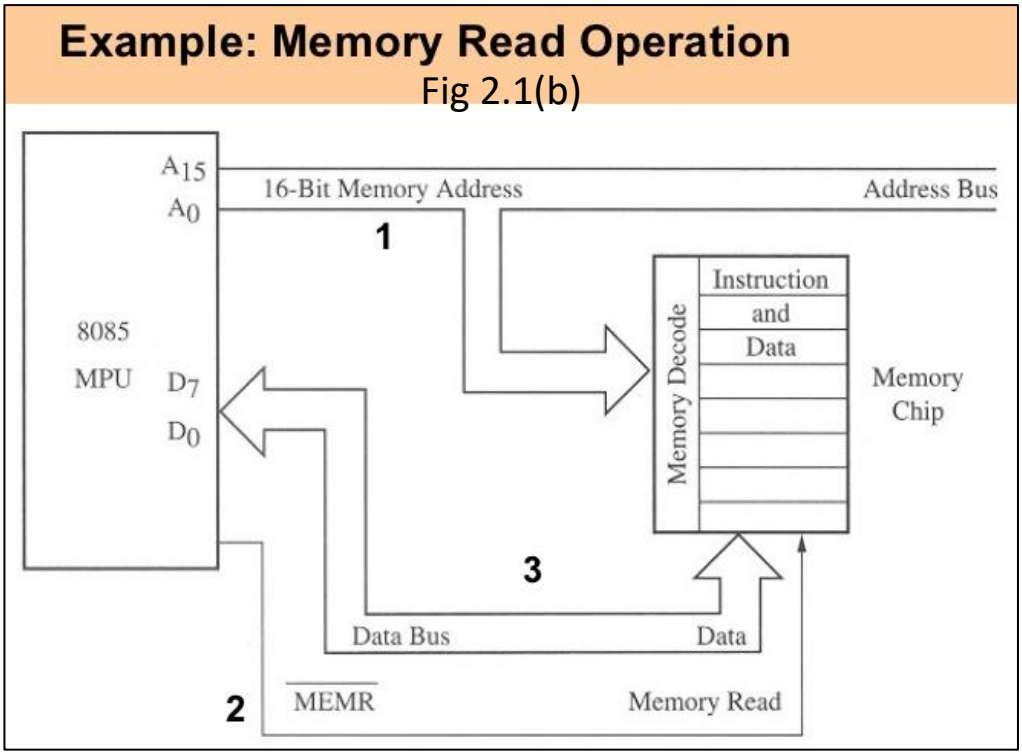
Control Bus

- Comprised of various lines that carry synchronization signals.
- The MPU uses such lines for providing control and timing signals.
- These are individual lines that provide a pulse to indicate an MPU operation.
- Normally control signals are of following types-
  - Memory Read
  - Memory Write
  - I/O Read
  - I/O Write
- When Memory Read or I/O Read is active, data is input to the processor.
- When memory Write or I/O Write is active, data is output from the processor.

How does this bus system work?

Example: Memory Read Operation (refer fig 2.1 b)

- The MPU places 16-bit address on the address bus. The address is decoded by external logic circuit.
- MPU sends a pulse called Memory Read ($\overline{\text{MEMR}}$) as the control signal.
- This pulse activates the memory chip and the contents (8-bit data) of the memory location is placed on data bus.
- This data is brought to microprocessor through data bus.



Internal Data Operations and 8085 Registers

Internal architecture of 8085 microprocessor determines how and what operations can be performed with the data.

The operations are:

- Store 8-bit data
- Perform arithmetic and logical operations
- Test for conditions
- Sequence the execution of instructions
- Store data temporarily during execution in the defined R/W memory locations called stack

To perform these operations, the microprocessor requires registers, ALU, control unit and internal buses.

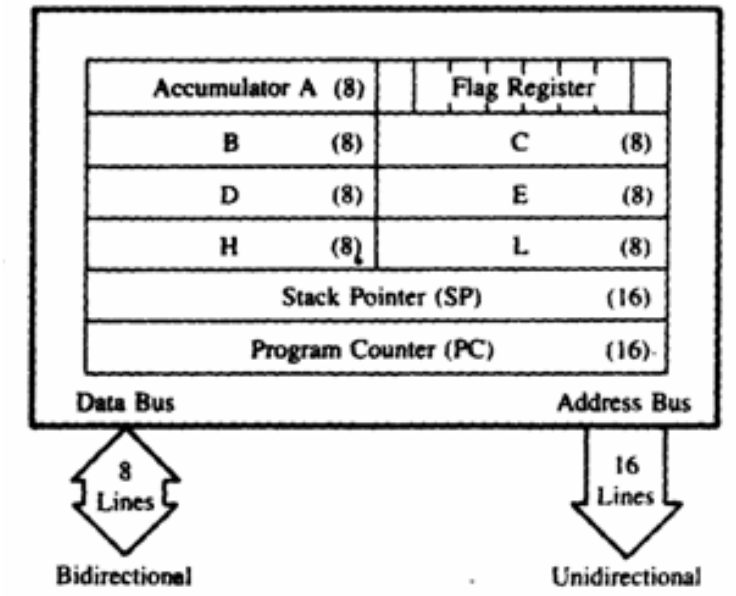


Fig 2.2
8085 Programmable registers

Fig 2.2 shows programming model of 8085 displaying internal registers.

The functions of these registers in following sample program are explained. The Hex codes of these instructions are stored in memory locations from 2000H to 2005H.

Sample Program:

Address	Hex Code	Instruction
2000	06	MVI B, 86H
2001	86	
2002	3E	MVI A, F2H
2003	F2	
2004	80	ADD B
2005	76	HLT

When user enters memory address 2000H and executes the program, the processor places the address 2000H in Program Counter.

- Program Counter is a 16-bit register. When user enters memory address 2000H and executes the program, the processor places the address 2000H in Program Counter. Program counter keeps incrementing by 1 for every code executed. It sequences the execution of instructions.
- In first instruction MVI B, 86H – register B is loaded with 8-bit data 86H.
- In second instruction MVI A, F2H – Accumulator is loaded with 8-bit data F2H.
- In third instruction ADD B – data in register B is added to Accumulator and answer is saved in accumulator. Also the content of Flag register are modified.
- The fifth instruction HLT – deals with concept of stack. Stack pointer is 16-bit register used as memory pointer to identify the stack. Stack is a part of user memory used by µp for temporary storage of data during execution.

Peripheral or Externally Initiated Operations

External devices can initiate the following operations for which individual pins are assigned:

- Reset: when Reset pin is activated, all internal operations are suspended and Program Counter is cleared ([PC] = 0000H)
- Interrupt: The µp can be interrupted from normal execution of instructions and asked to execute some other set of instructions called ISR (Interrupt Service Routine). The µp resumes its operation after completing ISR
- Ready: This signal is used to synchronize slower peripherals with the microprocessor.
- Hold: When Hold pin is activated, the µp releases the control of the buses and allows external peripheral to use them. Example: DMA (Direct Memory Access)

MEMORY

- There are different types of memories which can be classified into two groups: Main (prime) memory and Storage (secondary) memory.
- Prime / Main memory is again of two types – RAM (Random Access Memory)/RWM (Read Write Memory) and ROM (Read Only Memory)
- Magnetic tapes or disks can be used as Secondary/Storage memory.
- RWM is made up of registers and each register has a group of flip-flops that stores bits of information. These flip flops are called memory cells. The number of bits stored in register is called memory word. User can use RWM to write programs and store data.
- While ROM stores information permanently in the form of diodes.

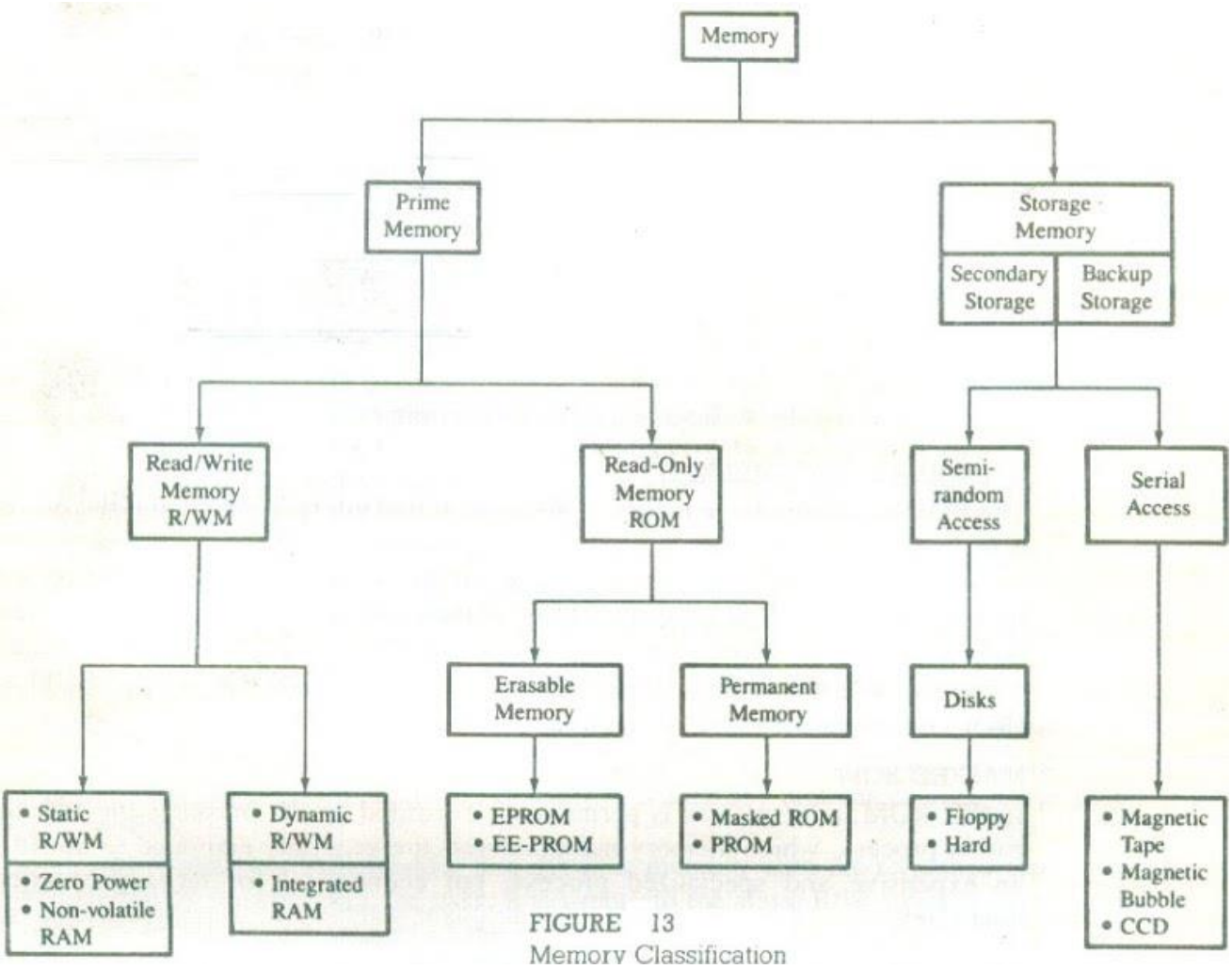


FIGURE 13
Memory Classification

RAM or RWM

- RWM (Read/Write Memory) is popularly known as RAM (Random Access Memory). It is used to store the information which is alterable, such as writing programs or data.
- This memory is volatile. That means when the power is turned off, all the contents are destroyed.
- There are two types of RAM – SRAM and DRAM

SRAM (Static Random Access Memory)

- Made up of flip-flops
- Stores the bit as a voltage
- Memory chip has low density

- High speed and expensive
- Used as cache memory
- High speed cache memory improves the performance of processor

DRAM (Dynamic Random Access Memory)

- Made up of MOSFETs and capacitors. Metal-Oxide-Semiconductor Field-Effect Transistor)
- Stores the bit as a charge
- Memory chip has high density
- Comparatively slower and cheaper
- Low power Consumption
- As information is stored in the form of charge it remains only for $1/4^{th}$ of a second (charge leaks). Hence it is needed to be read and written again (refresh) after every few milliseconds.
- Hence requires extra circuitry (DRAM Controller) to refresh the memory.

ROM

- ROM (Read Only Memory) is non-volatile memory. It retains the stored information even when the power is turned off.
- Information can only be read.
- Once the bit pattern is stored it is permanent or semi-permanent.
- Permanent group includes – Masked ROM and PROM
- Semi-permanent group includes – EPROM, EEPROM and Flash memory

Masked ROM

- Bit pattern is permanently recorded by masking and metalizing process.
- Memory manufacturers do this process.
- It is expensive and specialized process but economical for large production quantities.

PROM (Programmable Read Only Memory)

- Has nichrome wires arranged in a matrix. These wires can be viewed as diodes or fuses.
- This memory can be programmed by the user with special PROM programmer.
- Prom programmer selectively burns the fuses according to the bit pattern to be stored. The information stored will be permanent.
- These are also known as WORM (Write Once Read Many) or OTP (One time Programmable) memory

EPROM (Erasable Programmable Read Only Memory)

- EPROM is programmed in the same manner of PROM
- However EPROM can be erased and programmed number of times
- To erase EPROM we expose the device to strong source of ultra-violet light.
- Selective erasing is not possible in EPROM.
- EPROM chip has to be removed from the system in order to erase it.

EEPROM (Electrically Erasable Programmable Read Only Memory)

- EEPROM are electrically erasable and programmable ROM

- They are similar to EPROM, but erase operation can be done electrically
- Any byte within an EEPROM can be erased and rewritten. The data can be erased byte by byte.
- Once written, the new data will remain in the device forever or at least until it is electrically erased.
- Write cycles are also significantly longer than writes to a RAM, so you wouldn't want to use an EEPROM for your main system memory.

Flash ROM

- Flash memory devices are high density, low cost, nonvolatile, fast (to read, but not to write)
- Flash memory devices are electrically programmable
- Erased at the rate of one sector at a time and not byte by byte
- Typical sector size ranges from 256 bytes – 16 kilo-bytes (kb)

Differentiate between SRAM DRAM

SRAM (Static Random Access Memory)	**DRAM (Dynamic Random Access Memory)**
Made up of Flip-Flops	Made up of MOSFETs and Capacitors
Stores bits in the form of voltage. Low density	Stores bits in the form of charge. High density
Retains its contents as long as electrical power is applied to the chip.	Retains its contents only for $1/4^{th}$ of a second.
Offer extremely fast access time	Offer slow access time compared to SRAM
An embedded system includes a small block of SRAM (a few hundred kilobytes)	An embedded system includes a much larger block of DRAM (in megabytes)
SRAM does not require additional circuitry for its use	DRAM does not require additional circuitry called DRAM controller for its use
SRAM does not need to be refreshed periodically	DRAM needs to be refreshed in order to keep the memory alive
SRAM devices are much more expensive to produce	DRAM devices are comparatively cheaper & affordable to produce

Differentiate between EPROM EEPROM

EPROM (Erasable Programmable Read Only Memory)	**EEPROM (Electrically Erasable Programmable Read Only Memory)**
Erased by exposing to Ultra Violet light	Erased by applying voltage of 20V to 25V
Time required for erasing is 15 to 20 min	Time required for erasing is 10 ns

Selective erasing is not possible	Selective erasing is possible
Entire data will be erased	Data can be erased bit by bit
Chip needs to be removed from the system in order to erase it	Chip need not be removed from system in order to erase it

I/O DEVICES

I/O with 8-bit Addresses (Peripheral mapped I/O)

- MPU uses 8 address lines to identify the input or an output device. This is known as peripheral mapped I/O also known as I/O mapped I/O
- The 8 lines can have 256 addresses. Thus MPU can identify 256 input and output devices with addresses ranging from 00H to FFH.
- The input and output devices are differentiated by control signals. The MPU uses I/O Read control signal for input devices while I/O Write control signal for output devices.
- The entire range of I/O address from 00H to FFH is called I/O map.
- To connect input or output device to MPU, connections must be made through **Tristate interfacing devices** so they will be enabled only when MPU chooses to communicate with them.
- In case of memory we don't have to be concerned with these problems because of internal address decoding, Read/Write buffers.
- Steps in communicating with an I/O device:
    - MPU places 8-bit address on address bus, which is decoded by external decode logic.
    - MPU sends control signal (I/O Read or I/O Write) and enables the I/O device.
    - Data are transferred using data bus.

I/O with 16-bit Addresses (Memory mapped I/O)

- MPU uses 16 address lines to identify I/O device. The I/O is connected as if it is a memory register. This is known as memory mapped I/O.
- The 16 lines can have 2^{16} addresses. Thus MPU can identify 64Kbyte of memory with addresses ranging from 0000H to FFFFH.
- Steps in communicating with an I/O device:
    - MPU places 16-bit address on address bus, which is decoded by address decoder.
    - MPU sends control signal (Memory Read or Memory Write) and enables the I/O device.
    - Data are transferred using data bus.

Tristate Logic Devices

Buffers

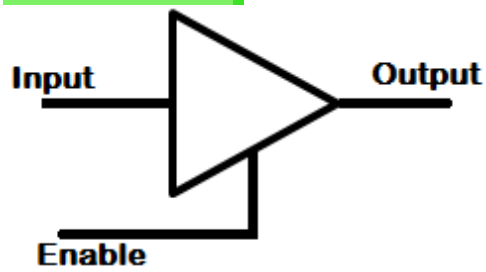
- Buffer is a logic circuit that enhances the current or power.
- This is similar to non-inverting amplifier with unit gain.

- The buffer is used to increase the driving capability of a logic circuit.
- The logic level of output is the same as that of input.

Tri-state Buffer

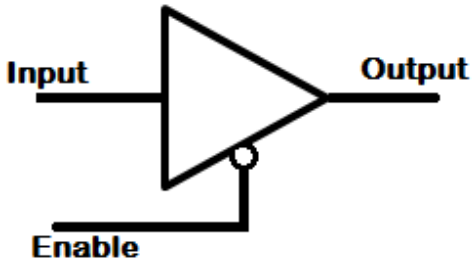
- It has one input line, one output line & one Enable line
- Depending on Enable line it is further classified into two types
  1) Active High Buffer 2) Active Low Buffer

Active High Buffer



When Enable is High--- Buffer is active. Therefore logic level of output is same as that of the input
When Enable is Low--- Buffer goes into high impedance state (Disabled)

Active Low Buffer



When Enable is Low- Buffer is active. Therefore logic level of output is the same as that of input
When Enable is High- Buffer goes into high impedance state (Disabled)

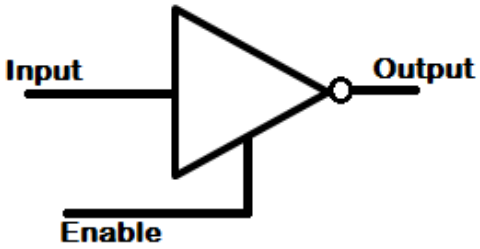
Inverters

- Inverter is a logic circuit that inverts the input signal.
- This is exactly opposite to buffer.
- The output is complement of input.

Tri-state Inverter

- It has one input line, one output line & one Enable line
- Depending on Enable line it is further classified into two types
  1) Active High inverter 2) Active Low inverter

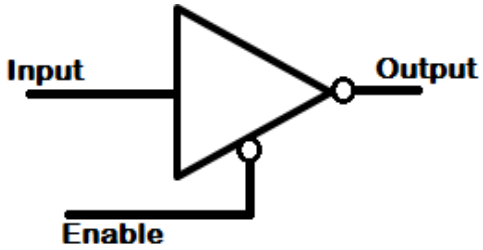
Active High Inverter



When Enable is High--- Inverter is active. Therefore logic level of output is complement of the input

When Enable is Low--- Inverter goes into high impedance state (Disabled)

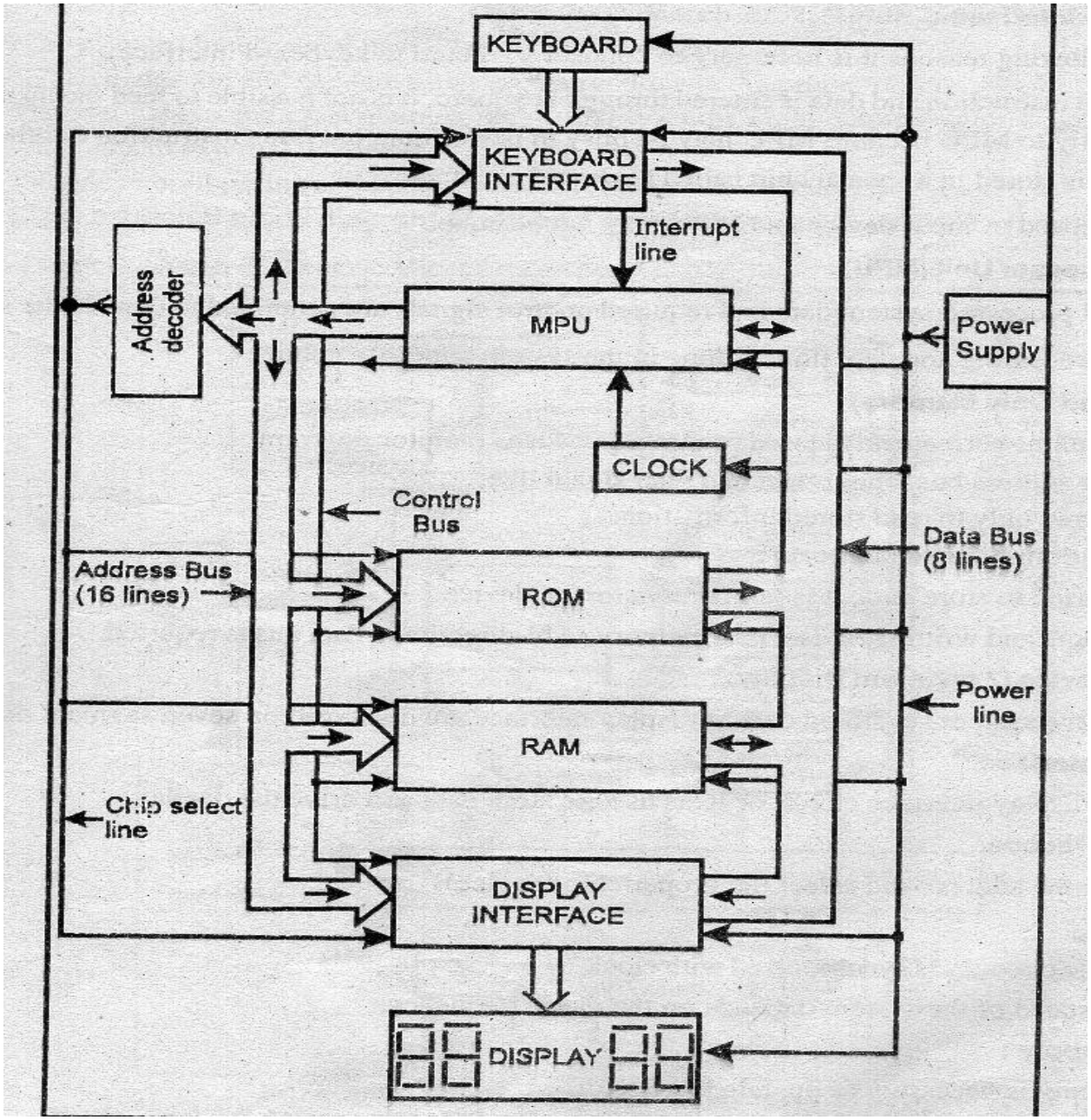
Active Low Buffer



When Enable is Low--- Inverter is active. Therefore logic level of output is complement of the input

When Enable is High--- Inverter goes into high impedance state (Disabled)

8085 based Single Board Microcomputer


KEYBOARD
KEYBOARD INTERFACE
Interrupt line
Address decoder
MPU
Power Supply
CLOCK
Control Bus
Data Bus (8 lines)
Address Bus (16 lines)
ROM
Power line
RAM
Chip select line
DISPLAY INTERFACE
DISPLAY