Two Basic Memory Operations
The memory unit supports two fundamental operations: Read and Write. The read operation reads a previously stored data and the write operation stores a value in memory, see the figure below.

Memory Read and write Bus Cycles
The following steps have to be followed in a typical read cycle:
1. Place the address of the location to be read on the address bus.
2. Activate the memory read control signal on the control bus.
3. Wait for the memory to retrieve the data from the address memory location.
4. Read the data from the data bus.
5. Drop the memory read control signal to terminate the read cycle.

The following steps have to be followed in a typical read cycle:
1. Place the address of the location to be written on the address bus.
2. Place the data to be written on the data bus.
3. Activate the memory write control signal on the control bus.
4. Wait for the memory to store the data at the address location.
5. Drop the memory write control signal to terminate the write cycle.
Memory Read Bus Cycle

Memory Write Bus Cycle
Hardware Organization of the Memory Address Space

The 8088μP's memory address space as shown in figure below is implemented as single 1 Mbyte memory bank.

The 8086μP's 1 Mbyte memory address space as shown in figure below is implemented as two independent 512 Kbyte banks:

- Data bytes associated with an even address (00000, 00002, ....., FFFFE) reside in the low bank.
- Data bytes associated with an odd addresses (00001, 00003, ....., FFFFF) reside in the high bank.

Address bits A1 through A19 select the location that is to be accessed. They are applied to both banks in parallel. A0 (BLE) and bank high enable (BHE) are used as bank-select signals:

- A0 = 0 causes the low bank to be enabled.
- BHE = 0 causes the high bank to be enabled.

Each of the memory banks provides half of the 8086μP's 16-bit data bus. The lower bank transfers bytes of data over data lines D0 through D7, while data transfers for a high bank use D8 through D15.

The byte transfer operation and word transfer operation of the 8088μP is as illustrated in figure below. The byte transfer operation needs one bus cycle, while word transfer operation needs two bus cycles.
Figure below shows how a byte transfer operation is performed to address X, an even-addressed storage location. A₀ is set to logic 0 to enable the low bank of memory and BHE to logic 1 to disable the high bank. Data are transferred to or from the lower bank over data bus lines D₀ - D₇.

**Even address byte transfer by the 8086**

Figure below shows how a byte transfer operation is performed to an odd addressed storage location such as X + 1. A₀ is set to logic 1 and BHE to logic 0. This enables the high bank of memory and disables the low bank. Data are transferred over bus lines D₈ through D₁₅. D₈ represents the LSB.
Odd address byte transfer by the 8086

Figure below illustrates how an aligned word (at even address X) is accessed. Both the high and low banks are accessed at the same time. Both A₀ and BHE are set to 0. This 16-bit word is transferred over the complete data bus D₀ through D₁₅ in just one bus cycle.

Even address word transfer by the 8086

Figure below illustrates how a misaligned word (at address X + 1) is accessed. Two bus cycles are needed. During the first bus cycle, the byte of the word located at address X + 1 in the high bank is accessed over D₈ through D₁₅. Even though the data transfer uses data lines D₈ through D₁₅, to the processor it is the low byte of the addressed data word. In the second memory bus cycle, the even byte located at X + 2 in the low bank is accessed over bus lines D₀ through D₇.
First bus cycle

Second bus cycle

A_{19} - A_1
D_{19} - D_8
BHE (LOW)

D_7 - D_0
A_6 (HIGH)

A_{19} - A_1
D_{19} - D_8
BHE (HIGH)

D_7 - D_0
A_6 (LOW)
Input /Output Interface

Input / Output (I/O) devices provide the means by which the computer system can interact with the outside world. Computers use I/O devices (also called peripheral devices) for two major purposes:

1- To communicate with the outside world and,
2- Store data.

Devices that are used to communicate like, printer, keyboard, modem. Devices that are used to store data like disk drive. I/O devices are connected to the system bus through I/O controller (interface) – which acts as interface between the system bus and I/O devices. The 8086µP employs two different types of input/output (I/O): Isolated I/O and Memory-mapped I/O. These I/O methods differ in how I/O ports are mapped into the 8086's address spaces.

Isolated Input/Output:

The I/O devices are treated separate from memory. I/O ports are organized as bytes of data; the memory address space contains 1M consecutive byte addresses in the range 00000H, through FFFFFH; and the I/O address space contains 64K consecutive byte addresses in the range 0000H through FFFFH as shown in figure below.

The way in which the microprocessor deals with input/output devices is similar to the way in which it deals with memory. The only difference is that just the 16 least significant lines of the bus, AD₀ through AD₁₅, are in use (because I/O addresses are 16 bit long), and throughout the bus cycles, the M/IO control signal is set to 0.
Advantages of Isolated I/O
- Complete memory address space available for use by memory.
- I/O instructions maximizes the performance.

Disadvantage of Isolated I/O
- All inputs/outputs must take place between an I/O port and accumulator register.

Memory Mapped I/O
The address space dedicated to I/O devices is a part of the memory. Addresses E0000H - E0FFFH → 4096 memory locations are assigned to I/O ports. E0000H, E0001H, and E0002H correspond to byte-wide ports 0, 1, and 2. E0000H and E0001H correspond to word-wide port 0 at address E0000H.

Advantages of memory mapped I/O
- I/O transfers can take place between I/O port and any of the registers.

Disadvantage of memory mapped I/O
- Memory instructions perform slower.
- Part of the memory address space cannot be used to implement memory.