Friday, August 29, 2025 1:13 PM

8086 Address and Data Bus Concepts

1. Multiplexed Address/Data Bus

- The **8086** has a **16-bit multiplexed address/data bus (AD15–AD0)**.
  - During **T1 cycle**: these lines carry the **lower 16 bits of the address (A15–A0)**.
  - During **T2–T4 cycles**: the same lines are used to **transfer data (D15–D0)**.
- This multiplexing reduces pin count but requires **demultiplexing** using external latches.

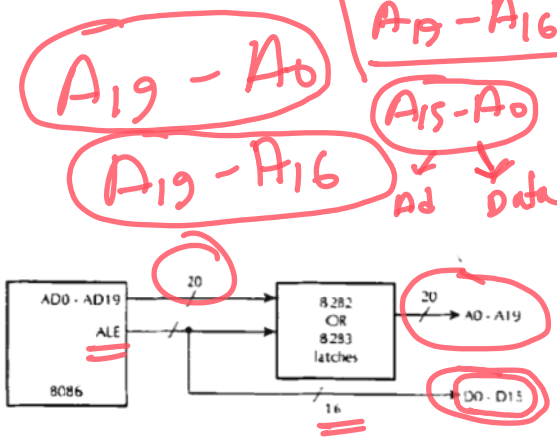


FIGURE 3.10 Separate address and data buses

2. Role of ALE (Address Latch Enable)

- **ALE** is a control signal generated by the 8086 during **T1 cycle**.
- ALE is connected to a latch device (e.g., **8282/8283**).
- When ALE = HIGH → the latch captures the address present on AD15–AD0.
- When ALE goes LOW → the address is held stable for the rest of the bus cycle, while AD15–AD0 are released for data.
- This ensures that the address remains valid for memory/IO chip selection during the full bus cycle.

3. Higher Address Lines (A19–A16)

- The 8086 provides **20-bit physical addressing** (1 MB address space).
- The higher 4 address lines **A19–A16** are not multiplexed and remain valid throughout the cycle.

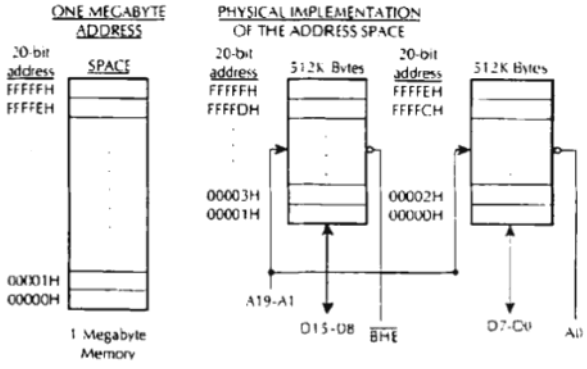


FIGURE 3.11 8086 memory

4. Memory Address Space and Bank Concept

- Total addressable memory = **1 MB (00000H – FFFFFH)**.
- Divided into **two 512 KB banks**:
  - **Lower Bank (even addresses)** → connected to **D7–D0**.
  - **Upper Bank (odd addresses)** → connected to **D15–D8**.
- The bank selection is controlled by **A0** and **BHE'** **(Bus High Enable, active LOW)**:

BHE	A0	Byte transferred
0	0	Both bytes
0	1	Upper byte to/from odd address
1	0	Lower byte to/from even address
1	1	None

BHE'	A0	Data Transfer
1	0	Lower bank (even address, D7–D0)
0	1	Upper bank (odd address, D15–D8)
0	0	Both banks (word at even address)
1	1	No transfer

Examples of Byte and Word Access

1. Byte Access – Even Address

Suppose the instruction is:

MOV DH, [BX]

and physical address = **20004H (even)**.

- A0 = 0 → selects **even bank**.
- BHE' = 1 → disables upper bank.
- Data comes from **D7–D0 lines** only.
- Result → 8-bit data loaded into DH.

2. Byte Access – Odd Address

Suppose address = **20005H (odd)**.

- A0 = 1 → disables even bank.
- BHE' = 0 → selects **odd bank**.
- Data comes from **D15–D8 lines** only.
- Result → 8-bit data loaded into register.

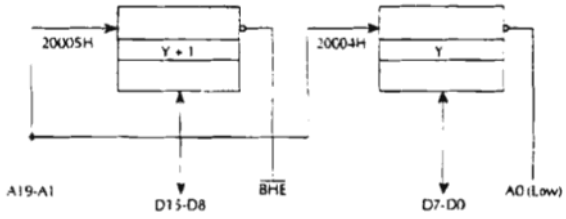


FIGURE 3.12 Even-addressed word transfer.

Word Access – Even Address

Suppose instruction:

MOV [BX], CX

and physical address = **20004H (even)**.

- A0 = 0, BHE' = 0 → both banks enabled.
- **CL (low byte)** written to even bank (20004H → D7–D0).
- **CH (high byte)** written to odd bank (20005H → D15–D8).
- Entire word transferred in **one bus cycle**.

Word Access – Odd Address

Suppose address = **20005H (odd)**.

- Requires **two bus cycles**:
  1. First cycle: A0 = 1, BHE' = 0 → odd bank selected. Data → [20005H] from D15–D8.
  2. Second cycle: A0 = 0, BHE' = 1 → even bank selected. Data → [20006H] from D7–D0.
- Word transfer completed in **two cycles** because of misalignment.

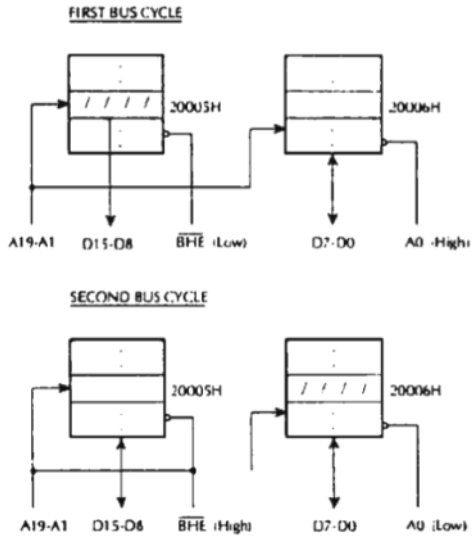
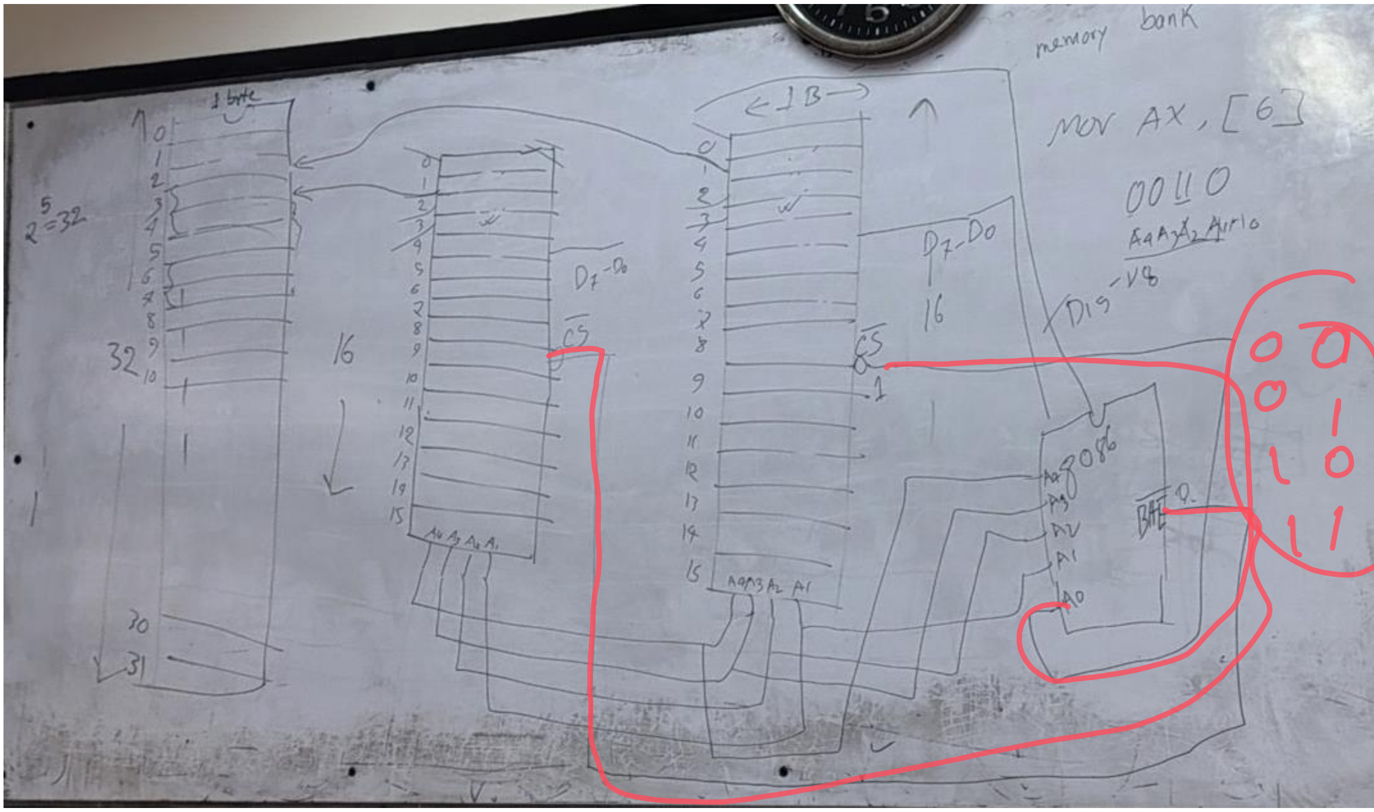


FIGURE 3.13 Odd-addressed word transfer.



1 byte
$2^5 = 32$
32
16
D7-D0
CS
<-1B->
memory bank
MOV AX, [6]
00110
A4A3A2A1A0
D15-D8
8086
A4
A3
A2
A1
A0
BHE