Bus Cycle

Friday, August 29, 2025 9:00 AM

8086 Bus Cycle

The 8086 communicates with external memory and I/O devices using a **bus cycle**.
A bus cycle is the sequence of operations needed to transfer data or fetch instructions via the **system bus**.

- Each bus cycle consists of **four clock periods (T1, T2, T3, T4)**, called **T-states**.
- If the addressed device is too slow, the processor inserts **wait states (Tw)** after T3 until the device becomes ready.
- Control and status signals coordinate the transfer.

The bus cycle can be either a **Read Cycle** or a **Write Cycle**.

Signals

- ✓ **CLK**: System clock divides cycle into T1, T2, T3, optional Tw, T4.
- ✓ **ALE**: Latches address (only in T1).
- ✓ **A19–A16**: Address bits in T1 → become status lines (S6–S3) afterward.
- ✓ **AD15–AD0**: Multiplexed bus: address in T1, data during T2–T3–Tw, high-Z in T4.
- ✓ **RD’**: Asserted LOW only in read cycles (T2–T3–Tw).
- ✓ **WR’**: Asserted LOW only in write cycles (T2–T3–Tw).
- ✓ **DT/R’**: Sets bus transceiver direction: LOW = receive (read), HIGH = transmit (write).
- ✓ **DEN’**: Active-LOW; enables external transceivers only during actual data transfer.
- ✓ **READY**: Synchronizes slow devices; if LOW, 8086 inserts Tw wait state(s).

cpu → address latch

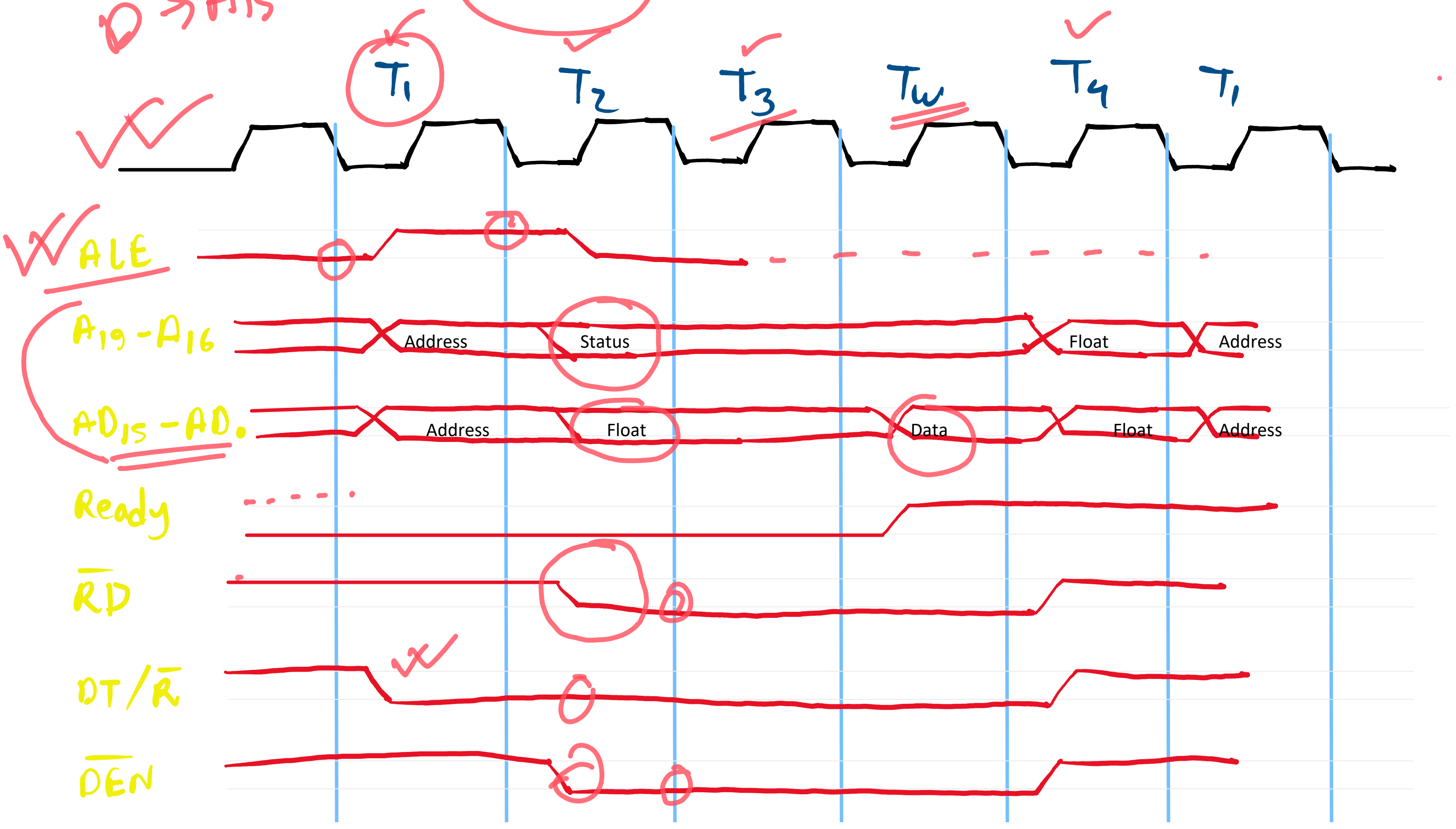
A/D

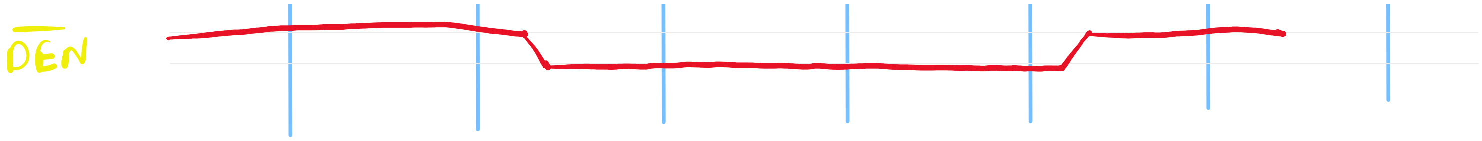
20 → A_{19}-A_0 Address → 20 bit

Data → 16 bit

D → A_{15}-A_0

8086 Read Cycle (fetching instruction/data)





The **read cycle** is used when the processor fetches instructions or reads data from memory/I/O.

T1 (Address Phase)

- **CLK**: defines the timing of T1.
- The 8086 places the **20-bit address** (segment + offset) on the multiplexed bus:
    - **A19–A16** = high-order address lines.
    - **AD15–AD0** = lower 16 address bits.
- **ALE (Address Latch Enable)** goes **HIGH** to latch the address externally.
- **RD'** and **WR'** are both HIGH (inactive).
- **DT/R'** = **LOW** → processor is preparing to **receive data**.
- **DEN'** = HIGH → transceivers disabled for now.

T2 (Command Phase)

- **ALE** goes LOW → address is removed from AD15–AD0.
- **A19–A16** lines switch from address to **status signals (S6–S3)**.
- **AD15–AD0** bus becomes **tri-stated**, preparing to receive data.
- **RD'** goes **LOW** → memory/I/O device is told to output data.
- **DEN'** goes **LOW** → external transceiver is enabled to pass data into CPU.
- **DT/R'** remains **LOW** (receive).

T3 (Data Phase)

- The addressed memory/I/O device places **data** on **AD15–AD0**.
- **RD'** remains LOW → strobe still active.
- **DEN'** remains LOW → transceiver still enabled.
- **READY** is checked at the end of T3:
    - If **READY = HIGH** → CPU will latch data at the end of T3.
    - If **READY = LOW** → CPU inserts wait state(s) Tw.

Tw (Optional Wait State)

- Bus signals remain same as in T3.
- Data is valid but CPU delays latching until **READY = HIGH**.

T4 (Completion Phase)

- The CPU latches the data from **AD15–AD0** into its internal register.
- **RD'** goes HIGH → read strobe disabled.
- **DEN'** goes HIGH → transceiver disabled.
- **DT/R'** returns to idle.
- Bus floats (high-Z) and becomes ready for the next cycle.

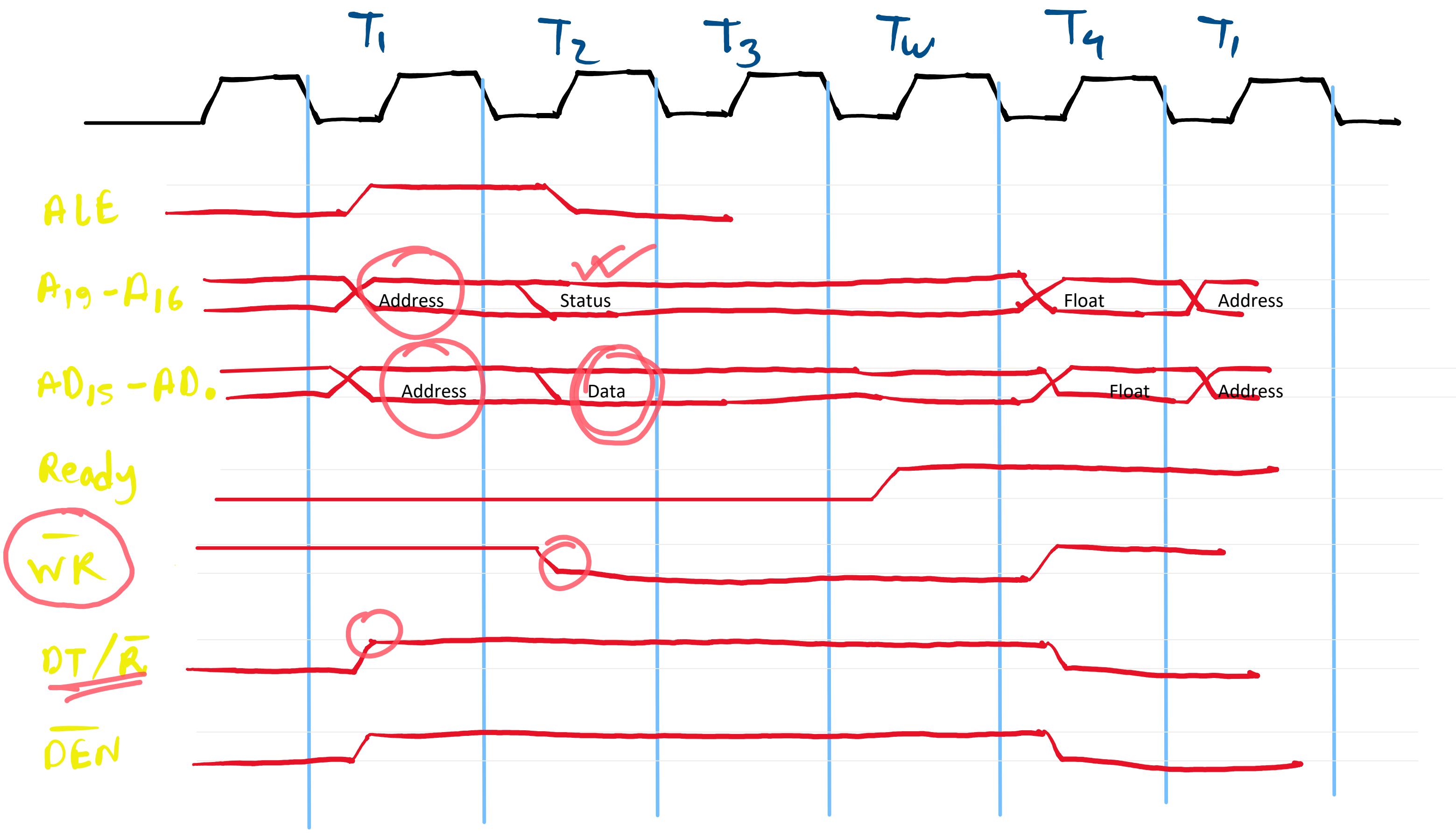
Example : Memory Read Cycle

Suppose the 8086 wants to **read a 16-bit word** from memory location 2000H.

1. **Address Phase (T1)**
    - CPU places 2000H on the address bus.
    - **ALE = HIGH** → external latch captures the address.
    - **A19–A16 = High-order address bits**, **AD15–AD0 = Lower address bits**.
    - Control lines: RD'=HIGH, WR'=HIGH (inactive).
2. **Command Phase (T2)**
    - **ALE = LOW** → address is latched, bus AD15–AD0 freed.
    - **RD' = LOW** → CPU requests memory read.
    - **DT/R' = LOW** → Bus set to *Receive mode*.
    - **DEN' = LOW** → Data transceivers enabled for input.
    - Memory decodes the address and gets ready to output data.
3. **Data Phase (T3 + Tw if needed)**
    - Memory places the contents of location 2000H on **AD15–AD0**.
    - CPU samples this data near end of T3.
    - If **READY=LOW**, wait states (Tw) are inserted until memory outputs valid data.
    - Example: If memory at 2000H = 3A5FH, CPU reads that word.
4. **Idle Phase (T4)**
    - **RD' = HIGH** → read request ends.
    - **DEN' = HIGH**, **DT/R' = inactive** → Bus released.
    - **AD15–AD0** float (High-Z).

Result: CPU successfully fetched the word 3A5FH from memory.

8086 Write Cycle (sending data)



The **write cycle** is used when the processor sends data to memory/I/O.

T1 (Address Phase)

- **CLK** defines T1.
- **A19–A16** = upper address bits, **AD15–AD0** = lower address bits.
- **ALE** = HIGH → address latched externally.
- **WR’** and **RD’** = HIGH (inactive).
- **DT/R'** = **HIGH** → CPU is in **transmit mode**.
- **DEN’** may begin to go LOW near the end of T1 to enable transceiver.

T2 (Command & Data Phase)

- **ALE** goes LOW.
- **A19–A16** = status signals (S6–S3).
- **AD15–AD0** = CPU drives valid **data** onto the bus.
- **WR’** goes **LOW** → write strobe active.
- **DEN’** goes **LOW** → transceiver enabled to output data.
- **DT/R'** = HIGH → data direction = output.

T3 (Data Hold Phase)

- **AD15–AD0** = data remains stable.
- **WR’** = LOW → device captures data.
- **DEN’** = LOW → bus enabled.
- **READY** is checked:
    - If **READY = HIGH** → CPU proceeds normally.

Example: Memory Write Cycle

Suppose the 8086 wants to **write a 16-bit word 55AAH** into memory location 3000H.

1. **Address Phase (T1)**
    - CPU places 3000H on address bus.
    - **ALE = HIGH** → address latched externally.
    - **A19–A16 = High-order address bits**, **AD15–AD0 = Lower address bits**.
    - Control lines: RD’=HIGH, WR’=HIGH (inactive).
2. **Command Phase (T2)**
    - **ALE = LOW** → address latched.
    - **WR’ = LOW** → CPU requests a memory write.
    - **DT/R’ = HIGH** → Bus set to *Transmit mode*.
    - **DEN’ = LOW** → Data transceivers enabled for output.
    - CPU starts driving data onto **AD15–AD0**.
3. **Data Phase (T3 + Tw if needed)**
    - CPU places the data word 55AAH on **AD15–AD0**.
    - Memory latches this data before end of T3.
    - If **READY=LOW**, CPU keeps WR’ LOW and holds data until READY=HIGH.

- **WR'** = LOW → device captures data.
- **DEN'** = LOW → bus enabled.
- **READY** is checked:
  - If **READY = HIGH** → CPU proceeds normally.
  - If **READY = LOW** → wait state(s) Tw are inserted.

Tw (Optional Wait State)

- Same as T3: CPU continues to drive valid data.
- **WR'** = LOW, **DEN'** = LOW, **DT/R'** = HIGH.

T4 (Completion Phase)

- Data transfer completed; device has latched the data.
- **WR'** goes HIGH → write strobe disabled.
- **DEN'** goes HIGH → transceiver disabled.
- **DT/R'** returns idle.
- **AD15–AD0** float (high-Z).
- Bus cycle ends.

- Memory latches this data before end of T3.
- If **READY=LOW**, CPU keeps WR' LOW and holds data until READY=HIGH.

4. Idle Phase (T4)

- **WR' = HIGH** → write request ends.
- **DEN' = HIGH**, **DT/R' = inactive** → bus released.
- **AD15–AD0** float.

Result: The word 55AAH is successfully written to memory location 3000H.