Pin Diagram of 8086 Microprocessor

Introduction

The **Intel 8086** is a 16-bit microprocessor built using **HMOS technology**, containing approximately **29,000 transistors**. It is housed in a **40-pin Dual In-line Package (DIP)**. Its pins are multiplexed to support efficient use of limited pin count, and the processor can operate in either:

- **Minimum Mode** (single processor system).
- **Maximum Mode** (multiprocessor or coprocessor system).

Key Facts

- Technology: **HMOS**.
- Transistor Count: **29,000**.
- Package: **40-pin DIP**.
- Total Pins: **40**.
- Address Bus: **20-bit** (A0–A19).
- Data Bus: **16-bit** (D0–D15).
- Control Signals: Read/Write, Interrupt, Clock, DMA, etc.
- Modes of Operation: **Minimum** and **Maximum**.

Pin Configuration and Modes

The 8086 has two different pin diagrams depending on the operating mode:

- **Minimum Mode (Uniprocessor System):** Pin 33 (`MN/MX#`) is held **high**. In this mode, the CPU itself issues all the control signals.
- **Maximum Mode (Multiprocessor System):** Pin 33 (`MN/MX#`) is held **low**. In this mode, an external bus controller IC (**8288**) generates the control signals.

The main difference between the pin diagrams lies in **pin 24 to pin 31**, which are configured differently for Minimum and Maximum mode.

Pin Diagram

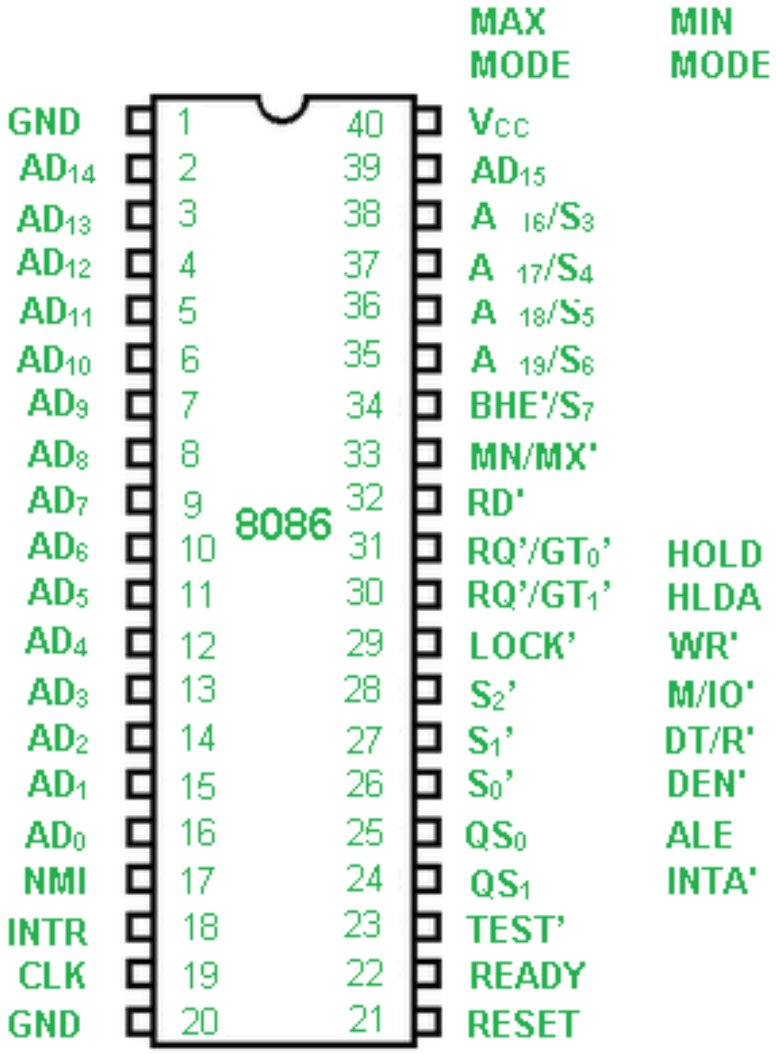

Figure 1: *

Figure: Pin Diagram of Intel 8086 Microprocessor (40 pins)

Pin Description

The 40 pins of 8086 are divided into the following functional groups.

1. Address/Data Bus (Multiplexed)

AD0–AD15 (Pins 2–16, 39–40)

- Multiplexed as the **lower 16-bit address bus (A0–A15)** and the **16-bit data bus (D0–D15)**.
- **T1 state:** Carries address A0–A15.
- **T2–T4 states:** Carries data D0–D15.
- Used to address memory (2^{20} = 1 MB) and I/O (2^{16} = 64 kB).

A16–A19 / S3–S6 (Pins 35–38)

- Multiplexed as **upper 4-bit address bus (A16–A19)** and **status lines (S3–S6)**.
- **S3, S4:** Indicate segment register in use.
- **S5:** Reflects interrupt flag (IF).
- **S6:** Always 0 (reserved).

$\overline{\text{BHE}}$/S7 (Pin 34)

- Functions as **Bus High Enable (BHE)** signal during **T1**, remains **low**.
- Acts as a **chip select** signal for the higher byte on **AD15–AD8**.
- Functions as **status signal S7** during **T2–T4**, remains **high**.
- Together with **A0**, it determines references to memory (byte or word access).

Memory Bank Selection using BHE and A0:

BHE	A0	Access Type
0	0	Both banks active, **16-bit word transfer** on AD15–AD0
0	1	Only **high bank** active, upper byte from/to odd address on AD15–AD8
1	0	Only **low bank** active, lower byte from/to even address on AD7–AD0
1	1	No bank active

2. Control and Status Signals

ALE (Pin 25)

- **Address Latch Enable (ALE)** output signal.
- Active in **Minimum Mode (MIN Mode)**.
- Generates a **positive-going pulse** to indicate that a **valid address** is present on the **AD0–AD15 lines (Address/Data lines)**.
- Used to latch the lower 16-bit address into external latches when multiplexed address/data lines are used.
- Ensures proper separation of address and data during memory or I/O operations.

RD (Pin 32)

- **Read (RD)** output signal.
- Indicates a **CPU read operation** from memory or an input device.
- Active **low**: logic 0 indicates the CPU is reading data.
- Enables memory or I/O device to place valid data on the bus.
- Works together with **M/$\bar{\text{IO}}$** and **DEN** signals to ensure correct data timing.
- Active in both **Minimum Mode** and **Maximum Mode**.

WR (Pin 29)

- **Write (WR)** output signal.
- Active in **Minimum Mode (MIN Mode)**.
- Indicates a **CPU write operation** to memory or an I/O device.
- Active **low**: logic 0 indicates the CPU is writing data.
- Data is placed on the bus by the CPU and transferred to the addressed memory or output device.
- Works in coordination with **M/$\bar{\text{IO}}$** and **DEN** signals to ensure proper timing.

M/$\bar{IO}$ (Pin 28)

- **Memory / I/O (M/$\bar{IO}$)** status output signal.
- Active only in **Minimum Mode (MIN Mode)**.
- Distinguishes whether the current bus cycle is accessing **memory** or an **I/O device**.
- **High (1)** = Memory access cycle.
- **Low (0)** = I/O access cycle.
- Provides control information to external hardware (like memory chips or I/O devices) so they can respond correctly.

DT/$\bar{R}$ (Pin 27)

- **Data Transmit / Receive (DT/$\bar{R}$)** output signal.
- Active in **Minimum Mode (MIN Mode)**.
- Controls the **direction of data flow** through the bus transceivers.
- **High (1)** = CPU transmits data to the bus (sending data out).
- **Low (0)** = CPU receives data from the bus (accepting data in).
- Ensures proper data transfer between the microprocessor and memory or I/O devices.

DEN (Pin 26)

- **Data Enable (DEN)** output signal.
- Active in **Minimum Mode (MIN Mode)**.
- Indicates the **availability of valid data** on the multiplexed address/data lines (AD0–AD15).
- Active from **middle of T2** until **middle of T4** during a bus cycle.
- Enables external transceivers to drive or latch data at the correct time.
- Ensures data is transferred only when valid, avoiding bus contention.

3. Power, Clock, and Reset

VCC (Pin 40)

- Provides **+5V DC** supply to the microprocessor.
- Powers all internal circuits of the 8086 CPU.

GND (Pins 1, 20)

- **Ground reference** for the microprocessor.
- Provides a common reference for all voltages and signals.

CLK (Pin 19)

- **External clock input** for the microprocessor.
- Typical frequency range: **5–10 MHz**.
- Synchronizes all internal operations and timings of the 8086.

RESET (Pin 21)

- **RESET signal** initializes the microprocessor.
- Must be applied **after 50 μs of power-on** and remain active for at least **4 clock cycles**.
- When RESET is active:
  - Processor immediately terminates its present activity.
  - All three buses are **tristated**.
  - **ALE** and **HLDA** signals go low.
  - All internal registers are cleared to **0000H**.
  - **CS = F000H**, **IP = FFF0H**.
- When RESET returns to low, execution begins from the physical address **FFFF0H**.
- Typically, an **EPROM** is mapped at addresses **FFFF0H – FFFFFH** to store the start-up program.

READY (Pin 22)

- Used to **insert wait states** into the microprocessor's bus cycle.
- When **HIGH** $\rightarrow$ 8086 proceeds with the current bus cycle.
- When **LOW** $\rightarrow$ 8086 enters a **wait state** until READY becomes high.
- Synchronizes the CPU with slower memory or I/O devices.

4. Interrupts and DMA

INTR (Pin 18)

- **Interrupt Request (Maskable)** input signal.
- Triggered by external hardware devices requesting CPU attention.
- The request is recognized only if the **Interrupt Flag (IF) = 1**.
- On acknowledgment, the 8086 issues an **Interrupt Acknowledge (INTA)** signal.
- The CPU then reads the interrupt type number from the external device and executes the corresponding **Interrupt Service Routine (ISR)**.
- **Priority:** Lowest priority among hardware interrupts (below NMI).
- **Can be disabled** by clearing the IF flag (using `CLI` instruction).
- Supports vectored interrupts (through external hardware like 8259 PIC).

NMI (Pin 17)

- **Non-Maskable Interrupt (NMI)** input signal.
- **High-priority** hardware interrupt that **cannot be disabled** by software (independent of the Interrupt Flag IF).
- Triggered on the **rising edge** of the NMI input.
- Typically used for critical events such as:
  - Power failure or emergency shutdown signals.
  - Memory parity errors.
  - Hardware malfunctions.
- When activated, CPU immediately suspends current execution and jumps to a predefined interrupt vector.
- **ISR Location:** Vector stored at `00008H` (CS:IP fetched from memory at addresses `00008H{0000BH`).
- Cannot be masked or ignored → guarantees CPU response to urgent events.

INTA (Pin 24)

- **Interrupt Acknowledge (INTA)** output signal.
- Active in **Minimum Mode (MIN Mode)**.
- Microprocessor receives interrupt request through **INTR** or **NMI**.
- CPU acknowledges the interrupt request by issuing the **INTA** signal.
- Used by external hardware to provide the interrupt type number to the CPU.

Feature	**Maskable Interrupt (INTR)**	**Non-Maskable Interrupt (NMI)**
Can be disabled?	Yes, can be masked/disabled using the **Interrupt Flag (IF)**.	No, always recognized by CPU regardless of IF.
Priority	Lower priority.	Higher priority (urgent).
8086 Example Pin	INTR (Pin 18).	NMI (Pin 17).
Use cases	General-purpose interrupts such as I/O devices, keyboard, timers, etc.	Critical events like power failure, memory parity error, hardware malfunction.
ISR Location	Vectored externally during **INTA** cycle (type number supplied by device/PIC).	Fixed at interrupt vector `00008H` (CS:IP fetched from 00008H–0000BH).

TEST (Pin 23)

- The **TEST** input pin is used in conjunction with the **WAIT** instruction for **external synchronization**.
- When the CPU executes a `WAIT` instruction, it continuously monitors the logic level of the TEST pin.
- **Operation:**
  - If TEST = 1 $\Rightarrow$ CPU continues normal execution (no wait).
  - If TEST = 0 $\Rightarrow$ CPU enters a **wait state** until TEST returns to 1.
- **Connection with 8087 NDP (Numeric Data Processor):**
  - The BUSY output pin of the 8087 is connected to the TEST input pin of the 8086.
  - When the 8087 is busy, it pulls the TEST signal **high**, causing the 8086 to **WAIT**.
  - Once the 8087 completes execution, the BUSY signal goes **low**, which sets TEST low.
  - The 8086 then resumes execution of its program.
- This mechanism ensures proper coordination between the 8086 CPU and its coprocessor (8087).

HOLD (Pin 31) / HLDA (Pin 30)

- **HOLD (Hold Request, Pin 31):**
  - Active-high input signal used by **Direct Memory Access (DMA)** controllers or other bus masters to request control of the system buses.
  - When asserted, the 8086 completes the current bus cycle and then releases the **Address, Data, and Control buses**.
  - During this time, **ALE, DEN, and DT/$\bar{R}$** go low, and the buses are **tristated**.
- **HLDA (Hold Acknowledge, Pin 30):**
  - Output signal from 8086 acknowledging that the buses have been released.
  - Activated one clock cycle after **HOLD** is recognized.
  - When **HOLD** goes low, **HLDA** also goes low, and the CPU regains bus control.
- **Use Case:** Ensures DMA controllers or coprocessors (like 8087) can access memory or I/O directly without CPU intervention.
- Active in **Minimum Mode (MIN Mode)** only.

5. Mode Control

MN/$\bar{M}$X (Pin 33)

- **High = Minimum Mode:** 8086 works as the only processor, directly generating all control signals.
- **Low = Maximum Mode:** 8086 works in multiprocessor or coprocessor systems, and an external bus controller (8288) generates control signals.

Minimum Mode

Minimum Mode Characteristics

- Designed for a **single-processor environment**.
- The CPU internally generates all necessary control signals: RD, WR, INTA, ALE, DT/$\bar{R}$, DEN.
- Provides simpler and cheaper system design.
- Ideal for small systems with no coprocessor or DMA controller.

Maximum Mode

Maximum Mode Characteristics

- Designed for **multiprocessor or coprocessor systems** (e.g., 8086 + 8087 NDP).
- Requires an external **8288 bus controller** to generate control signals.
- Provides advanced features such as:
  - Bus arbitration (multiple bus masters).
  - LOCK prefix for atomic operations.
  - Special signals for coprocessor coordination.
- Supports higher performance and multitasking environments.

Special Maximum Mode Pins

RQ/GT0, RQ/GT1 (Pins 30, 31)

- Request/Grant pins used for **bus arbitration** in a multiprocessor system.
- Function as **bidirectional signals**:
  - A processor can use RQ/GT to request the bus.
  - The CPU responds with a grant after completing the current cycle.
- RQ/GT0 has higher priority than RQ/GT1.

LOCK (Pin 29)

- **LOCK** output signal.
- Active in **Maximum Mode (MAX Mode)** only.
- Activated by the **LOCK prefix instruction** to ensure exclusive use of shared memory in multiprocessor systems.
- Remains active until the completion of the **next instruction**.
- When **LOCK = low**:
  - All interrupts are masked.
  - **HOLD** requests from external devices are not granted.
- Ensures atomic read-modify-write operations on memory.

QS1 / QS0 (Pins 24, 25)

- **Queue Status Signals (QS1 / QS0)** output signals.
- Active in **Maximum Mode (MAX Mode)** only.
- Provide the status of the **instruction queue**.
- **Status Table:**

QS1	QS0	Status
0	0	No operation
0	1	First byte of opcode from the queue
1	0	Queue is being emptied
1	1	Subsequent byte from the queue

- Used by external hardware (like coprocessors or prefetch logic) to track instruction fetch and execution.

S2 / S1 / S0 (Pins 26–28)

- **Status Signals (S2 / S1 / S0)** output signals.
- Active in **Maximum Mode (MAX Mode)** only.
- Indicate the **type of operation** being carried out by the processor.
- Active during **T4 of the previous cycle** and **T1 and T2 of the current cycle**.
- Passive during **T3** of the current bus cycle.
- **Status Table:**

S2	S1	S0	CPU Cycle
0	0	0	Interrupt acknowledge
0	0	1	Read I/O Port
0	1	0	Write I/O Port
0	1	1	HALT
1	0	0	Code access
1	0	1	Read memory
1	1	0	Write memory
1	1	1	Passive

- Used by external hardware to identify the current CPU operation for proper bus control.