8086/88 Device Specifications

Both are packaged in DIP (Dual In-Line Packages)
- 8086: 16-bit microprocessor with a **16-bit** data bus
- 8088: 16-bit microprocessor with an **8-bit** data bus

Both are 5V parts (i.e. V_{DD} is 5V)
- 8086: Draws a maximum supply current of 360mA
- 8086: Draws a maximum supply current of 340mA
- 80C86/80C88: CMOS version draws 10mA with temp spec -40 to 225°F

Input/Output current levels:

<table>
<thead>
<tr>
<th>INPUT</th>
<th>Logic level</th>
<th>Voltage</th>
<th>Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic level</td>
<td>Voltage</td>
<td>Current</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.8V max</td>
<td>+/- 10uA max</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2.0V min</td>
<td>+/- 10uA max</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OUTPUT</th>
<th>Logic level</th>
<th>Voltage</th>
<th>Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic level</td>
<td>Voltage</td>
<td>Current</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.45V max</td>
<td>+2mA max</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2.4V min</td>
<td>- 400uA max</td>
<td></td>
</tr>
</tbody>
</table>

Yields a 350mV noise immunity for logic 0 (Output max can be as high as 450mV while input max can be no higher than 800mV). This limits the loading on the outputs.
8086/88 Pinout

8086 CPU

<table>
<thead>
<tr>
<th>Pin</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>GND</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>AD14</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD13</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD12</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD11</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD10</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD9</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD8</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD7</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD6</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD5</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD4</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD3</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD2</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD1</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>AD0</td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>NMI</td>
<td>Not Used</td>
<td></td>
</tr>
<tr>
<td>INTR</td>
<td>Interrupt</td>
<td></td>
</tr>
<tr>
<td>CLK</td>
<td>Clock</td>
<td></td>
</tr>
<tr>
<td>GND</td>
<td>Ground</td>
<td></td>
</tr>
</tbody>
</table>

MIN MODE (MAX MODE)

- VCC
- AD15
- A16/S3
- A17/S4
- A18/S5
- A19/S6
- BHE/S7
- MN/MX
- RD
- Hold
- HLDA
- WR
- M/IQ
- DT/R
- DEN
- ALE
- INTA
- TEST
- READY
- RESET

8086 CPU
8086/88 Pinout

- **AD15-AD0**
  - Multiplexed address (ALE=1)/data bus (ALE=0).

- **A19/S6-A16/S3** (multiplexed)
  - High order 4 bits of the 20-bit address OR status bits S6-S3.

- **M/IO**
  - Indicates if address is a Memory or IO address.

- **RD**
  - When 0, data bus is driven by memory or an I/O device.

- **WR**
  - Microprocessor is driving data bus to memory or an I/O device. When 0, data bus contains valid data.

- **ALE** (Address latch enable)
  - When 1, address data bus contains a memory or I/O address.

- **DT/R** (Data Transmit/Receive)
  - Data bus is transmitting/receiving data.

- **DEN** (Data bus Enable)
  - Activates external data bus buffers.
8086/88 Pinout

- $S_7$, $S_6$, $S_5$, $S_4$, $S_3$, $\overline{S_2}$, $\overline{S_1}$, $\overline{S_0}$
  - $S_7$: Logic 1, $S_6$: Logic 0.
  - $S_5$: Indicates condition of IF flag bits.
  - $S_4$-$S_3$: Indicate which segment is accessed during current bus cycle:

<table>
<thead>
<tr>
<th>S4</th>
<th>S3</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Extra segment</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Stack segment</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Code or no segment</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Data segment</td>
</tr>
</tbody>
</table>

- $\overline{S_2}$, $\overline{S_1}$, $\overline{S_0}$: Indicate function of current bus cycle (decoded by 8288).

<table>
<thead>
<tr>
<th>$\overline{S_2}$</th>
<th>$\overline{S_1}$</th>
<th>$\overline{S_0}$</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Interrupt Ack</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>I/O Read</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>I/O Write</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Halt</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$S_2$</th>
<th>$S_1$</th>
<th>$S_0$</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Opcode Fetch</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Memory Read</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>Memory Write</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Passive</td>
</tr>
</tbody>
</table>
### 8086/88 Pinout

- **INTR**
  
  When 1 and IF=1, microprocessor prepares to service interrupt. **INTA** becomes active after current instruction completes.

- **INTA**
  
  Interrupt Acknowledge generated by the microprocessor in response to INTR. Causes the interrupt vector to be put onto the data bus.

- **NMI**
  
  Non-maskable interrupt. Similar to INTR except IF flag bit is not consulted and interrupt is vector 2.

- **CLK**
  
  Clock input must have a duty cycle of 33% (high for 1/3 and low for 2/3s)

- **VCC/GND**
  
  Power supply (5V) and GND (0V)

- **MN/MX**
  
  Select minimum (5V) or maximum mode (0V) of operation.
8086/88 Pinout

- **BHE**
  Bus High Enable. Enables the most significant data bus bits (D_{15}-D_8) during a read or write operation.

- **READY**
  Used to insert wait states (controlled by memory and IO for reads/writes) into the microprocessor.

- **RESET**
  Microprocessor resets if this pin is held high for 4 clock periods.
  Instruction execution begins at FFFF0H and IF flag is cleared.

- **TEST**
  An input that is tested by the WAIT instruction.
  Commonly connected to the 8087 coprocessor.

- **HOLD**
  Requests a direct memory access (DMA). When 1, microprocessor stops and places address, data and control bus in high-impedance state.

- **HLDA** (Hold Acknowledge)
  Indicates that the microprocessor has entered the hold state.
8086/88 Pinout

- **RO/GT1 and RO/GT0**
  Request/grant pins request/grant direct memory accesses (DMA) during maximum mode operation.

- **LOCK**
  Lock output is used to lock peripherals off the system. Activated by using the LOCK: prefix on any instruction.

- **QS1 and QS0**
  The queue status bits show status of internal instruction queue. Provided for access by the numeric coprocessor (8087).
8284A Clock Generator

- Clock generation
- RESET synchronization
- READY synchronization
- Peripheral clock signal

Connection of the 8284 and the 8086.
8284A Clock Generator

8284A Chipset

- CSYNC
- PCLK
- AEN1
- RDY1
- READY
- RDY2
- AEN2
- CLK
- GND

- 1 → 18: VCC
- 2 → 17: X1
- 3 → 16: X2
- 4 → 15: ASYNC
- 5 → 14: EFI
- 6 → 13: F/C
- 7 → 12: OSC
- 8 → 11: RES
- 9 → 10: RESET

- RES → Schmitt trigger
- Schmitt trigger → DQ
- DQ → RESET
- X1 → OSC
- X2 → (EFI input to other 8284As)
- X2 → PCLK
- F/C → 2-to-1 mux
- 2-to-1 mux → div-by-3 counter
- div-by-3 counter +3 → PCLK
- div-by-2 counter +2 → CLK
- DQ → READY
- DQ → READY
- DQ → READY

- CSYNC → EFI
- EFI → X1
- X1 → OSC
- X2 → PCLK
- RDY1 → 2-to-1 mux
- AEN1 → 2-to-1 mux
- RDY2 → 2-to-1 mux
- AEN2 → 2-to-1 mux
- ASYNC → 2-to-1 mux
- 2-to-1 mux → div-by-3 counter
- div-by-3 counter +3 → PCLK
- div-by-2 counter +2 → CLK
- DQ → READY
- DQ → READY
- DQ → READY

UMBC
AN HONORS UNIVERSITY IN MARYLAND
Clock generation

Crystal is connected to X1 and X2.

XTAL OSC generates square wave signal at crystal's frequency which feeds:
- An inverting buffer (output OSC) which is used to drive the EFI input of other 8284As.
- 2-to-1 MUX
  - F/\bar{C} selects XTAL or EFI external input.

The MUX drives a divide-by-3 counter (15MHz to 5MHz).
This drives:
- The **READY** flipflop (READY synchronization).
- A second **divide-by-2 counter** (2.5MHz clk for peripheral components).
- The **RESET** flipflop.
- **CLK** which drives the 8086 CLK input.
**8284A Clock Generator**

**RESET**

Negative edge-triggered flipflop applies the RESET signal to the 8086 on the falling edge.

The 8086 samples the **RESET** pin on the rising edge.

Correct reset timing requires that the **RESET** input to the microprocessor becomes a logic 1 **NO LATER** than 4 clocks after power up and stay high for at least 50µs.
**BUS Buffering and Latching**

Computer systems have three buses
- Address
- Data
- Control

The Address and Data bus are *multiplexed (shared)* due to pin limitations on the 8086. The ALE pin is used to control a set of latches.

All signals MUST be buffered
- Buffered Latches for A₀-A₁₅.
- Control and A₁₆-A₁₉ + **BHE** are buffered separately.
- Data bus buffers must be bi-directional buffers.

In a 8086 system, the memory is designed with two banks
- High bank contains the higher order 8-bits and low bank the lower order 8-bits
- Data can be transferred as 8 bits from either bank or 16-bits from both
- **BHE** pin selects the high-order memory bank
**BUS Buffering and Latching**

- **8086 CPU**
  - VCC
  - AD15
  - AD14
  - A16/S3
  - AD13
  - A17/S4
  - AD12
  - A18/S5
  - AD11
  - A19/S6
  - AD10
  - BHE/S7
  - AD9
  - MN/MX
  - AD8
  - RD
  - AD7
  - Hold
  - AD6
  - HLDA
  - AD5
  - WR
  - AD4
  - M/IO
  - AD3
  - DT/R
  - AD2
  - DEN
  - AD1
  - ALE
  - AD0
  - INTA
  - NMI
  - TEST
  - INTR
  - READY
  - CLK
  - RESET
  - GND

- **Buffers**
  - Buffer
  - G Latches
  - D15
  - BB
  - GD
  - D8
  - BB
  - GD
  - D0
  - Control
  - AD0
  - AD1
  - AD2
  - AD3
  - AD4
  - AD5
  - AD6
  - A7
  - A8
  - A15
  - A16
  - BHE
  - A19

- **8086 Chipset**
  - 8086 CPU
  - BUS Buffering and Latching
  - Control
**BUS Timing**

**Writing**

- Dump address on address bus.
- Dump data on data bus.
- Issue a write (WR) and set M/IO to 1.

![Simplified 8086 Write Bus Cycle](image_url)

- **Address**
- **Valid Address**
- **Address/Data**
- **Data written to memory**
- **WR**

**One Bus Cycle**

- **T1**
- **T2**
- **T3**
- **T4**

**Simplified 8086 Write Bus Cycle**
**BUS Timing**

**Reading**

- Dump address on address bus.
- Issue a read (RD) and set M/IO to 1.
- Wait for memory access cycle.

![Simplified 8086 Read Bus Cycle](image.png)
**BUS Timing**

**Read Bus Timing:**

Bus Timing for a Read Operation
BUS Timing

During T₁:
- The address is placed on the Address/Data bus.
- Control signals M/IO, ALE and DT/R specify memory or I/O, latch the address onto the address bus and set the direction of data transfer on data bus.

During T₂:
- 8086 issues the RD or WR signal, DEN, and, for a write, the data.
- DEN enables the memory or I/O device to receive the data for writes and the 8086 to receive the data for reads.

During T₃:
- This cycle is provided to allow memory to access data.
- READY is sampled at the end of T₂.
  - If low, T₃ becomes a wait state.
  - Otherwise, the data bus is sampled at the end of T₃.

During T₄:
- All bus signals are deactivated, in preparation for next bus cycle.
- Data is sampled for reads, writes occur for writes.
**BUS Timing**

Each BUS CYCLE on the 8086 equals *four* system clocking periods (T states). The clock rate is *5MHz*, therefore one Bus Cycle is *800ns*. The transfer rate is *1.25MHz*.

Memory specifications (memory access time) must match constraints of system timing.

For example, bus timing for a read operation shows almost *600ns* are needed to read data.

However, memory must access faster due to setup times, e.g. Address setup and data setup. This subtracts off about *150ns*. Therefore, memory must access in at least *450ns* minus another *30-40ns* guard band for buffers and decoders.

*420ns* DRAM required for the 8086.
**BUS Timing**

**READY**

An input to the 8086 that causes wait states for slower memory and I/O components. A wait state ($T_W$) is an extra clock period inserted between $T_2$ and $T_3$ to lengthen the bus cycle.

For example, this extends a 460ns bus cycle (at 5MHz clock) to 660ns.

```
<table>
<thead>
<tr>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
</tr>
</thead>
<tbody>
<tr>
<td>200ns</td>
<td>800ns</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Text discusses role of 8284A and timing requirements for the 8086.
MIN and MAX Mode

Controlled through the MN/MX pin.

- Minimum mode is cheaper since all control signals for memory and I/O are generated by the microprocessor.
- Maximum mode is designed to be used when a coprocessor (8087) exists in the system.

Some of the control signals must be generated externally, due to redefinition of certain control pins on the 8086.

The following pins are lost when the 8086 operates in Maximum mode.

- ALE
- WR
- IO/M
- DT/R
- DEN
- INTA

This requires an external bus controller: 8288 Bus Controller.
Separate signals are used for I/O (IORC and IOWC) and memory (MRDC and MWTC).

Also provided are advanced memory (AIOWC) and I/O (AIOWC) write strobes plus INTA.
MAX Mode 8086 System

8086 CPU

AD0-AD15

INT

STB

8286 Transceiver

8288

8084A

RES

GND

VCC

8259A Interrupt Controller

8259A

8284A

IR 0-7

8286

Latches

T

OE

CLK

S0

S1

S2

MRDC

MWTC

DEN

DT/R

ALE

OIRC

IOWC

INTA

Address

Data

RAM

RD

WR