The 8088 and 8086 Microprocessors and Their Memory and Input/output Interfaces

8.1 The 8088 and 8086 Microprocessors
8.2 Minimum-Mode and Maximum-Mode System
8.3 Minimum-Mode Interface
8.4 Maximum-Mode Interface
8.5 Electrical Characteristics
8.6 System Clock
8.7 Bus Cycle and Time States
8.8 Hardware Organization of the Memory Address Space
The 8088 and 806 Microprocessors and Their Memory and Input/output Interfaces

8.9 Memory Bus Status Codes
8.10 Memory Control Signals
8.11 Read and Write Bus Cycles
8.12 Memory Interface Circuits
8.13 Programmable Logic Arrays
8.14 Types of Input/Output
8.15 An Isolated Input/Output Interface
8.16 Input/Output Data Transfer
8.17 Input/Output Instructions
8.18 Input/Output Bus Cycles

8.1 The 8088 and 806 Microprocessors

- The 806, announced in 1978, was the first 16-bit microprocessor introduced by Intel Corporation.
- 806 and 8088 are internally 16-bit MPU. However, externally the 806 has a 16-bit data bus and the 8088 has an 8-bit data bus.
8.1 The 8088 and 8086 Microprocessors

- 8086 and 8088 both have the ability to address up to 1 Mbyte of memory and 64K of input/output port.
- The 8088 and 8086 are both manufactured using *high-performance metal-oxide semiconductor (HMOS) technology*.
- The 8088 and 8086 are housed in a 40-pin dual in-line package and many pins have multiple functions.

![Intel 8086-1 Microprocessor](source)

**Intel D8086-1 Microprocessor**
- Processor Speed: 10.0 MHz
- Bus Speed: 10.0 MHz
- FPU: no

8.1 The 8088 and 8086 Microprocessors

- **CMOS, Complementary Metal-Oxide-Semiconductor**, is a major class of integrated circuits used in chips such as microprocessors, microcontrollers, static RAM, digital logic circuits, and analog circuits such as image sensors.
- Two important characteristics of CMOS devices are high noise immunity and low static power supply drain. Significant power is only drawn when its transistors are switching between on and off states; consequently, CMOS devices do not produce as much heat as other forms of logic such as TTL. CMOS also allows a high density of logic functions on a chip.
8.1 The 8088 and 8086 Microprocessors

Pin layout of the 8086 and 8088 microprocessor

8.2 Minimum-Mode and Maximum-Mode System

- The 8086 and 8088 microprocessors can be configured to work in either of two modes:
  - The minimum mode: $\text{MN/MX} = 1$
  - The maximum mode: $\text{MN/MX} = 0$
- The mode selection feature lets the 8088 or 8086 better meet the needs of a wide variety of system requirements.
- Minimum mode 8088/8086 systems are typically smaller and contain a single processor.
- Depending on the mode of operation selected, the assignment for a number of the pins on the microprocessor package are changed.
8.2 Minimum-Mode and Maximum-Mode System

<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>A0–A7</td>
<td>Address/data bus</td>
<td>3-state</td>
</tr>
<tr>
<td>A15–A8</td>
<td>Address bus</td>
<td>Output, 3-state</td>
</tr>
<tr>
<td>A10/A9</td>
<td>Address/status</td>
<td>Output, 3-state</td>
</tr>
<tr>
<td>MN/IX</td>
<td>Minimum/maximum Mode control</td>
<td>Input</td>
</tr>
<tr>
<td>RD</td>
<td>Read control</td>
<td>Output, 3-state</td>
</tr>
<tr>
<td>TEST</td>
<td>Wait on net control</td>
<td>Input</td>
</tr>
<tr>
<td>READY</td>
<td>Wait state control</td>
<td>Input</td>
</tr>
<tr>
<td>RESET</td>
<td>System reset</td>
<td>Input</td>
</tr>
<tr>
<td>NMI</td>
<td>Nonmaskable interrupt request</td>
<td>Input</td>
</tr>
<tr>
<td>INTR</td>
<td>Interrupt request</td>
<td>Input</td>
</tr>
<tr>
<td>CLK</td>
<td>System clock</td>
<td>Input</td>
</tr>
<tr>
<td>VCC</td>
<td>+5 V</td>
<td>Input</td>
</tr>
<tr>
<td>GND</td>
<td>Ground</td>
<td>Input</td>
</tr>
</tbody>
</table>

Signals common to both minimum and maximum mode

8.2 Minimum-Mode and Maximum-Mode System

<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOLD</td>
<td>Hold request</td>
<td>Input</td>
</tr>
<tr>
<td>HLDA</td>
<td>Hold acknowledge</td>
<td>Output</td>
</tr>
<tr>
<td>WR</td>
<td>Write control</td>
<td>Output, 3-state</td>
</tr>
<tr>
<td>IO/M</td>
<td>IO/memory control</td>
<td>Output, 3-state</td>
</tr>
<tr>
<td>DT/R</td>
<td>Data transmit/receive</td>
<td>Output, 3-state</td>
</tr>
<tr>
<td>D/EN</td>
<td>Data enable</td>
<td>Output, 3-state</td>
</tr>
<tr>
<td>SSO</td>
<td>Status line</td>
<td>Output, 3-state</td>
</tr>
<tr>
<td>ALE</td>
<td>Address latch enable</td>
<td>Output</td>
</tr>
<tr>
<td>INTA</td>
<td>Interrupt acknowledge</td>
<td>Output</td>
</tr>
</tbody>
</table>

Unique minimum-mode signals
8.2 Minimum-Mode and Maximum-Mode System

<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>RD/GT1, 0</td>
<td>Request/grant bus access control</td>
<td>Bidirectional</td>
</tr>
<tr>
<td>LOCK</td>
<td>Bus priority lock control</td>
<td></td>
</tr>
<tr>
<td>S2 - S0</td>
<td>Bus cycle status</td>
<td></td>
</tr>
<tr>
<td>QS1, QS0</td>
<td>Instruction queue status</td>
<td></td>
</tr>
</tbody>
</table>

Unique maximum-mode signals

8.2 Minimum-Mode and Maximum-Mode System

EXAMPLE

Which pins provide different signal functions in the minimum-mode 8088 and minimum-mode 8086?

Solution:
(a) Pins 2 through 8 on the 8088 are address lines A\textsubscript{14} through A\textsubscript{8}, but on the 8086 they are address/data lines AD\textsubscript{14} through AD\textsubscript{8}.
(b) Pin 28 on the 8088 is IO/\overline{M} output and on the 8086 it is the M/\overline{I\overline{O}} output.
(c) Pin 34 of the 8088 is the SSO output, and on the 8086 this pin supplies the BHE/S\textsubscript{7}.
8.3 Minimum-Mode Interface

Block diagram of the minimum-mode 8088 MPU

611 37100 微處理機原理與應用 Lecture 08-13

8.3 Minimum-Mode Interface

Block diagram of the minimum-mode 8086 MPU

611 37100 微處理機原理與應用 Lecture 08-14
8.3 Minimum-Mode Interface

- The minimum-mode signals can be divided into the following basic groups:
  - Address/Data bus
  - Status signals
  - Control signals
  - Interrupt signals
  - DMA interface signals

8.3 Minimum-Mode Interface

- **Address/Data bus**
  - The address bus is used to carry address information to the memory and I/O ports.
  - The address bus is 20-bit long and consists of signal lines A₀ through A₁₉.
  - A 20-bit address gives the 8088 a 1 Mbyte memory address space.
  - Only address line A₀ through A₁₅ are used when addressing I/O. This gives an I/O address space of 64 Kbytes.
  - The 8088 has 8 multiplexed address/data bus lines (A₀~A₇) while 8086 has 16 multiplexed address/data bus lines (A₀~A₁₅).
### 8.3 Minimum-Mode Interface

#### Status signals

- The four most significant address, $A_{19}$ through $A_{16}$ are multiplexed with *status signal* $S_6$ through $S_3$.
- Bits $S_4$ and $S_3$ together form a 2-bit binary code that identifies which of the internal segment registers was used to generate the physical address. $S_5$ is the logic level of the internal interrupt flag. $S_6$ is always at the 0 logic level.

<table>
<thead>
<tr>
<th>$S_4$</th>
<th>$S_3$</th>
<th>Address Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Alternate (relative to the ES segment)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Stack (relative to the SS segment)</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Code/None (relative to the CS segment or a default of zero)</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Data (relative to the DS segment)</td>
</tr>
</tbody>
</table>

#### Control signals

- The *control signals* are provided to support the memory and I/O interfaces of the 8088 and 8086.
  - ALE – Address Latch Enable
  - IO/M – IO/Memory (8088)
  - M/IO – Memory/IO (8086)
  - DT/R – Data Transmit/Receive (8088/8086)
  - SS0 – System Status Output (8088)
  - BHE – Bank High Enable (8086)
  - RD – Read (8088/8086)
  - WR – Write (8088/8086)
  - DEN – Data Enable (8088/8086)
  - READY – Ready (8088/8086)
8.3 Minimum-Mode Interface

Interrupt signals

- The **interrupt signals** can be used by an external device to signal that it needs to be serviced.
  - **INTR** – Interrupt Request
  - **INTA** – Interrupt Acknowledge
  - **TEST** – Test (can be use to synchronize MPU)
  - **NMI** – Nonmaskable Interrupt
  - **RESET** – Reset (hardware reset of the MPU)

8.3 Minimum-Mode Interface

DMA interface signals

- When an external device wants to take control of the system bus, it signals this fact to the MPU by switching HOLD to the 1 logic level.
- When in the hold state, signal lines AD0 through AD7, A8 through A15, A16/S3 through A19/S6, SSO, IO/M, DT/R, RD, WR, DEN, and INTR are all put into high-Z state.
- The 8088 signals external devices that the signal lines are in the high-Z state by switching its HLDA output to the 1 logic level.
8.4 Maximum-Mode Interface

- The maximum-mode configuration is mainly used for implementing a multiprocessor/coprocessor system environment.
- **Global resources** and **local resources**
- In the maximum-mode, facilities are provided for implementing allocation of global resources and passing bus control to other microprocessors sharing the system bus.

---

8.4 Maximum-Mode Interface

- **8288 bus controller**

![8088 maximum-mode block diagram](image)
8.4 *Maximum-Mode Interface*

- 8288 bus controller

In the maximum-mode, 8088/8086 outputs a status code on three signal lines, \( S_0, S_1, S_2 \), prior to the initialization of each bus cycle.

The 3-bit bus status code identifies which type of bus cycle is to follow and are input to the external bus controller device, 8288.

The 8288 produces one or two command signals for each bus cycle.
8.4 Maximum-Mode Interface

- 8288 bus controller

<table>
<thead>
<tr>
<th>Status Inputs</th>
<th>CPU Cycle</th>
<th>8288 Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>S₂</td>
<td>S₁</td>
<td>S₀</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Bus status code

Block diagram and pin layout of 8288
8.4 Maximum-Mode Interface

- Lock signal
  - The lock signal (LOCK) is meant to be output (logic 0) whenever the processor wants to lock out the other processor from using the bus.

- Local bus control signals
  - The request/grant signals (RQ/GT₀, RQ/GT₁) provide a prioritized bus access mechanism for accessing the local bus.

---

8.4 Maximum-Mode Interface

- Queue status signals
  - The 2-bit queue status code QS₀ and QS₁ tells the external circuitry what type of information was removed from the queue during the previous clock cycle.

<table>
<thead>
<tr>
<th>QS₁</th>
<th>QS₀</th>
<th>Queue Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 (low)</td>
<td>0</td>
<td>No Operation. During the last clock cycle, nothing was taken from the queue.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>First byte. The byte taken from the queue was the first byte of the instruction.</td>
</tr>
<tr>
<td>1 (high)</td>
<td>0</td>
<td>Queue Empty. The queue has been reinitialized as a result of the execution of a transfer of instruction.</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Subsequent Byte. The byte taken from the queue was a subsequent byte of the instruction.</td>
</tr>
</tbody>
</table>

Queue status code
8.4 Maximum-Mode Interface

**EXAMPLE**

If the bus status code $S_2S_1S_0$ equals 101, what type of bus activity is taking place? Which command output is produced by the 8288?

Solution:

Looking at the bus status table, we see that bus status code 101 identifies a read memory bus cycle and causes the MRDC output of the bus controller to switch to logic 0.

8.5 Electrical Characteristics

- Power is applied between pin 40 ($V_{cc}$) and pins 1 (GND) and 20 (GND).
- The nominal value of $V_{cc}$ is specified as $+5\text{V}$ dc with a tolerance of $\pm 10\%$.
- Both 8088 and 8086 draw a maximum of 340mA from the supply.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Test condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{IH}$</td>
<td>Input high voltage</td>
<td>$+2.0 \text{V}$</td>
<td>$V_{cc} + 0.5 \text{V}$</td>
<td>$I_{IH}=2.0 \text{mA}$</td>
</tr>
<tr>
<td>$V_{IL}$</td>
<td>Output high voltage</td>
<td>$+0.45 \text{V}$</td>
<td>$V_{cc} + 0.45 \text{V}$</td>
<td>$I_{OL}=400 \mu\text{A}$</td>
</tr>
</tbody>
</table>

I/O voltage levels
8.6 System Clock

- The time base for synchronization of the internal and external operations of the microprocessor in a microcomputer system is provided by the clock (CLK) input signal.
- The standard 8088 operates at 5 MHz and the 8088-2 operates at 8 MHz.
- The 8086 is manufactured in three speeds: 5-MHz 8086, 8-MHz 8086-2, and the 10-MHz 8086-1.
- The CLK is externally generated by the 8284 clock generator and driver IC.

8.6 System Clock

- Block diagram of the 8284 clock generator
8.6 System Clock

- Block diagram of the 8284 clock generator

Connecting the 8284 to the 8088

- The fundamental crystal frequency is divided by 3 within the 8284 to give either a 5- or 8-MHz clock signal.
8.6 System Clock

- CLK waveform
  - The signal is specified at Metal Oxide Semiconductor (MOS)-compatible voltage level.
  - The period of the 5-MHz 8088 can range from 200 ns to 500 ns, and the maximum rise and fall times of its edges equal 10 ns.

![CLK waveform diagram]

- PCLK and OSC signals
  - The peripheral clock (PCLK) and oscillator clock (OSC) signals are provided to drive peripheral ICs.
  - The clock output at PCLK is half the frequency of CLK. The OSC output is at the crystal frequency which is three times of CLK.

![PCLK and OSC signals diagram]
8.6 System Clock

EXAMPLE

If the CLK input of an 8086 MPU is to be driven by a 9-MHz signal, what speed version of the 8086 must be used and what frequency crystal must be attached to the 8284.

Solution:

The 8086-1 is the version of the 8086 that can be run at 9-MHz. To create the 9-MHz clock, a 27-MHz crystal must be used on the 8284.

8.7 Bus Cycle and Time States

- A bus cycle defines the basic operation that a microprocessor performs to communicate with external devices.
- Examples of bus cycles are the memory read, memory write, input/output read, and input/output write.
- The bus cycle of the 8088 and 8086 microprocessors consists of at least four clock periods.
- If no bus cycles are required, the microprocessor performs what are known as idle states.
- When READY is held at the 0 level, wait states are inserted between states T3 and T4 of the bus cycle.
8.7 Bus Cycle and Time States

EXAMPLE

What is the duration of the bus cycle in the 8088-based microcomputer if the clock is 8 MHz and the two wait states are inserted.

Solution:

The duration of the bus cycle in an 8 MHz system is given by

$$ t_{cyc} = 500 \text{ ns} + N \times 125 \text{ ns} $$

In this expression the N stands for the number of wait states. For a bus cycle with two wait states, we get

$$ t_{cyc} = 500 \text{ ns} + 2 \times 125 \text{ ns} = 500 \text{ ns} + 250 \text{ ns} = 750 \text{ ns} $$
8.8 Hardware Organization of the Memory Address Space

1M bytes

A19 - A0
D7 - D0

1Mx8 memory bank of the 8088

1M BYTES

FFFF
FFFF

2
1
0

High and low memory banks of the 8086

512K bytes

A19 - A1
D15 - D8
BHE
D7 - D0
A0

512K BYTES

FFFF
FFFFD

5
3
1

512K BYTES

FFFFE
FFFFC

4
2
0
8.8 Hardware Organization of the Memory Address Space

Transfer X

\[
\begin{array}{c}
\text{X+1} \\
(X) \\
0
\end{array}
\]

\[A_{19} - A_0, \quad D_7 - D_0\]

Byte transfer by the 8088

\[X+1, \quad (X), \quad 0\]

\[A_{19} - A_0, \quad D_7 - D_0\]

Word transfer by the 8088
8.8 Hardware Organization of the Memory Address Space

Even address byte transfer by the 8086:

\[ Y+1 \]
\[ X+1 \]
\[ A_{19} - A_i \]
\[ D_{15} - D_8 \]
\[ BHE (HIGH) \]

Odd address byte transfer by the 8086:

\[ Y \]
\[ X \]
\[ A_6 (LOW) \]
\[ D_7 - D_0 \]
\[ BHE (LOW) \]
8.8 Hardware Organization of the Memory Address Space

Even address word transfer by the 8086

Odd-address word transfer by the 8086
8.8 Hardware Organization of the Memory Address Space

EXAMPLE

Is the word at memory address 0123116 of an 8086-based microcomputer aligned or misaligned? How many cycle are required to read it from memory?

Solution:

The first byte of the word is the second byte at the aligned-word address 0123016. Therefore, the word is misaligned and required two bus cycles to be read from memory.

8.9 Address Bus Status Codes

Whenever a memory bus cycle is in progress, an address bus status code $S_4S_3$ is output by the processor.

$S_4S_3$ identifies which one of the four segment register is used to generate the physical address in the current bus cycle:

- $S_4S_3=00$ identifies the extra segment register (ES)
- $S_4S_3=01$ identifies the stack segment register (SS)
- $S_4S_3=10$ identifies the code segment register (CS)
- $S_4S_3=11$ identifies the data segment register (DS)

The memory address reach of the microprocessor can thus be expanded to 4 Mbytes.
### 8.10 Memory Control Signals

#### Minimum-mode memory control signals

- **ALE** – Address Latch Enable – used to latch the address in external memory.
- **IO/M** – Input-Output/Memory – signal external circuitry whether a memory of I/O bus cycle is in progress.
- **DT/R** – Data Transmit/Receive – signal external circuitry whether the 8088 is transmitting or receiving data over the bus.
- **RD** – Read – identifies that a read bus cycle is in progress.
- **WR** – Write – identifies that a write bus cycle is in progress.
- **DEN** – Data Enable – used to enable the data bus.
- **SSO** – Status Line – identifies whether a code or data access is in progress.
8.10 Memory Control Signals

- The control signals for the 8086's minimum-mode memory interface differs in three ways:
  - IO/M signal is replaced by M/IO signal.
  - The signal SSO is removed from the interface.
  - BHE (bank high enable) is added to the interface and is used to select input for the high bank of memory in the 8086's memory subsystem.

8.10 Memory Control Signals

- Maximum-mode memory control signals
8.10 Memory Control Signals

- Maximum-mode memory control signals
  - MRDC – Memory Read Command
  - MWTC – Memory Write Command
  - AMWC – Advanced Memory Write Command

<table>
<thead>
<tr>
<th>Status Inputs</th>
<th>CPU Cycle</th>
<th>8288 Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>S_3 S_2 S_1 S_0</td>
<td>Interrupt Acknowledge</td>
<td>INTA</td>
</tr>
<tr>
<td>0 0 0 0</td>
<td>Read I/O Port</td>
<td>IORC</td>
</tr>
<tr>
<td>0 0 1 0</td>
<td>Write I/O Port</td>
<td>IOWC, AOIOWC</td>
</tr>
<tr>
<td>0 1 1 1</td>
<td>Halt</td>
<td>None</td>
</tr>
<tr>
<td>1 0 0 0</td>
<td>Instruction Fetch</td>
<td>MRDC</td>
</tr>
<tr>
<td>1 0 1 0</td>
<td>Read Memory</td>
<td>MRDC</td>
</tr>
<tr>
<td>1 1 0 0</td>
<td>Write Memory</td>
<td>MWTC, AMWC</td>
</tr>
<tr>
<td>1 1 1 1</td>
<td>Passive</td>
<td>None</td>
</tr>
</tbody>
</table>

8.11 Read and Write Bus Cycle

- Read cycle

Minimum-mode memory read bus cycle of the 8088
8.11 Read and Write Bus Cycle

- Read cycle

Minimum-mode memory read bus cycle of the 8086

Maximum-mode memory read bus cycle of the 8086
8.11 Read and Write Bus Cycle

- **Write cycle**

Minimum-mode memory write bus cycle of the 8088

Maximum-mode memory write bus cycle of the 8086
8.12 Memory Interface Circuit

- Address bus latches and buffers
- Bank write and bank read control logic
- Data bus transceivers/buffers
- Address decoders

Memory interface block diagram
8.12 Memory Interface Circuit

Address bus latches and buffers

Operation of the 74F373

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>OC</td>
<td>Enable C</td>
</tr>
<tr>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>H</td>
<td>X</td>
</tr>
</tbody>
</table>

Block diagram of a D-type latch

Circuit diagram of the 74F373
8.12 Memory Interface Circuit

A review of flip-flop/latch logic

Cross-NOR S-R flip-flop

Cross-NAND S-R flip-flop

RESET SET
8.12 Memory Interface Circuit

A review of flip-flop/latch logic

The D latch is used to capture, or ‘latch’ the logic level which is present on the data line when the clock input is high.

\[
\begin{array}{c}
S & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\
R & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
Q_{t-1} & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
Q_t & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
\end{array}
\]
8.12 Memory Interface Circuit

A review of flip-flop/latch logic

Positive edge-triggered JK flip-flop

D-type latch
8.12 Memory Interface Circuit

- Address bus latches and buffers

Address latch circuit

Bank write and bank read control logic

Bank write control logic  Bank read control logic
8.12 Memory Interface Circuit

- Data bus transceivers

Block diagram and circuit diagram of the 74F245 octal bus transceiver

Data bus transceiver circuit
8.12 Memory Interface Circuit

- Address decoder

Address bus configuration with address decoding

```
Address decoder

Microprocessor address bus
ADn-AD0, A9-A0, BHE

Address latches

System address bus
An-A0, BHE

Address decoder

CE, CE

Chip enable

G

L

H

Y3

Y2

Y1

Y0

OUTPUTS

INPUTS

2-line to 4-line decoder
```

Block diagram and operation of the 74F139 decoder

```
<table>
<thead>
<tr>
<th>ENABLE</th>
<th>SELECT</th>
<th>Y0</th>
<th>Y1</th>
<th>Y2</th>
<th>Y3</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>X</td>
<td>X</td>
<td>H</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>L</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>L</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
</tr>
<tr>
<td>L</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>L</td>
</tr>
</tbody>
</table>
```
8.12 Memory Interface Circuit

- Address decoder

Circuit diagram of the 74F139 decoder

Address decoder circuit using 74F139
8.12 Memory Interface Circuit

- Address decoder

Block diagram and operation of the 74F138 decoder

Circuit diagram of the 74F138 decoder
8.12 Memory Interface Circuit

- Address decoder

Address decoder circuit using 74F138

8.13 Programmable Logic Arrays

- Programmable logic array, PLA, are general-purpose logic devices that have the ability to perform a wide variety of specialized logic functions.

- A PLA contains a general-purpose AND-OR-NOT array of logic gate circuits.

- The process used to connect or disconnect inputs of the AND gate array is known as programming, which leads to the name programmable logic array.
8.13 Programmable Logic Arrays

- Major types of programmable logic architecture
  - Simple Programmable Logic Devices (SPLDs)
    - PAL, GAL, PLA, EPLD
  - Complex Programmable Logic Devices (CPLDs)
    - EPLD, PEEL, EEPLD, MAX
  - Field Programmable Gate Arrays (FPGAs)
    - LCA, pASIC, FLEX, APEX, ACT, ORCA, Virtex, pASIC
  - Field Programmable InterConnect (FPICs)

- PLAs, GALs, and EPLDs
  - Early PLA devices were all manufactured with the bipolar semiconductor process.
  - Bipolar devices are programmed with an interconnect pattern by burning out fuse links within the device.
  - PLAs made with bipolar technology are characterized by slower operating speeds and higher power consumption.
  - Two kinds of newer PLA, manufactured with the CMOS process, are in wide use today: the GAL and EPLD.
8.13 Programmable Logic Arrays

- Block diagram of a PLA
  - The logic levels applied at inputs I₀ through I₁₅ and the programming of the AND array determine what logic levels are produced at outputs F₀ through F₁₅.
  - The capacity of a PLA is measured by three properties: the number of inputs, the number of outputs, and the number of product terms (P-terms).

- Architecture of a PLA
8.13 Programmable Logic Arrays

- Architecture of a PLA

![ PLA Architecture Diagram ]

(a) Typical PLA architecture. (b) PLA with output latch

8.13 Programmable Logic Arrays

- Standard PAL™ device
  - A PAL, programmable array logic, is a PLA in which the OR array is fixed; only the AND array is programmable.
  - The 16L8 is a widely used PAL IC. It is housed in a 20-pin package. It has 10 dedicated input, 2 dedicated outputs, and 6 programmable I/O lines.
  - The 16L8 is manufactured with bipolar technology. It operates from a +5V±10% dc power supply and draw a maximum of 180mA.
  - The 20L8 has 20 inputs, 8 outputs and 64 P-terms.
  - The 20R8 is the register output version of 20L8.
8.13 Programmable Logic Arrays

- Standard PAL™ device

16L8 circuit diagram and pin layout

8.13 Programmable Logic Arrays

- Standard PAL™ device

20L8 circuit diagram and pin layout
8.13 Programmable Logic Arrays

- Standard PAL™ device

16R8 circuit diagram and pin layout

20R8 circuit diagram and pin layout
### 8.13 Programmable Logic Arrays

- Expanding PLA capacity

![Diagram](image)

Expanding output word length

Expanding input word length

### 8.14 Types of Input/Output

- Isolated input/output

  - When using isolated I/O in a microcomputer system, the I/O device are treated separate from memory.

  - The memory address space contains 1 M consecutive byte address in the range $0000_{16}$ through $FFFF_{16}$, and that the I/O address space contains 64K consecutive byte addresses in the range $0000_{16}$ through $FFFF_{16}$.

  - All input and output data transfers must take place between the AL or AX register and I/O port.
8.14 Types of Input/Output

- **Isolated input/output**

- **Memory-mapped input/output**
  - In the case of memory-mapped I/O, MPU looks at the I/O port as though it is a storage location in memory.
  - Some of the memory address space is dedicated to I/O ports.
  - Instructions that affect data in memory are used instead of the special I/O instructions.
  - The memory instructions tend to execute slower than those specifically designed for isolated I/O.
8.14 Types of Input/Output

- Memory-mapped input/output

Memory-mapped I/O ports
8.15 Isolated Input/Output Interface

- I/O devices:
  - Keyboard
  - Printer
  - Mouse
  - 82C55A, etc.

- Functions of interface circuit:
  - Select the I/O port
  - Latch output data
  - Sample input data
  - Synchronize data transfer
  - Translate between TTL voltage levels and those required to operate the I/O devices.

Minimum-mode interface

Minimum-mode 8088 system I/O interface
8.15 Isolated Input/Output Interface

- Minimum-mode interface

Minimum-mode 8086 system I/O interface

- Maximum-mode interface

Maximum-mode 8088 system I/O interface
8.15 Isolated Input/Output Interface

- Maximum-mode interface

Maximum-mode 8086 system I/O interface

Status Inputs | CPU Cycle | 8288 Command
---|---|---
S<sub>2</sub> S<sub>1</sub> S<sub>0</sub> | Interrupt Acknowledge | INTA
0 0 1 | Read I/O Port | RDRC
0 1 0 | Write I/O Port | IOWC, AIOWC
0 1 1 | Halt | None
1 0 0 | Instruction Fetch | MRDC
1 1 0 | Write Memory | MWTC, AMWC
1 1 1 | Passive | None

I/O bus cycle status codes
8.16 Input/Output Data Transfers

- Input/output data transfers in the 8088 and 8086 microcomputers can be either byte-wide or word-wide.
- I/O addresses are 16 bits in length and are output by the 8088 to the I/O interface over bus lines AD_0 through AD_7 and A_8 through A_15.
- In 8088, the word transfers is performed as two consecutive byte-wide data transfer and takes two bus cycle.
- In 8086, the word transfers can takes either one or two bus cycle.
- Word-wide I/O ports should be aligned at even-address boundaries.

8.17 Input/Output Instructions

<table>
<thead>
<tr>
<th>Mnemonic</th>
<th>Meaning</th>
<th>Format</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN</td>
<td>Input direct</td>
<td>IN Acc, Port</td>
<td>(Acc) ←(Port)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Acc = AL or AX</td>
</tr>
<tr>
<td></td>
<td>Input indirect (variable)</td>
<td>IN Acc, DX</td>
<td>(Acc) ←(DX)</td>
</tr>
<tr>
<td>OUT</td>
<td>Output direct</td>
<td>OUT Port, Acc</td>
<td>(Port) ←(Acc)</td>
</tr>
<tr>
<td></td>
<td>Output indirect (variable)</td>
<td>OUT DX, Acc</td>
<td>(DX) ←(Acc)</td>
</tr>
</tbody>
</table>
8.17 Input/Output Instructions

EXAMPLE

Write a sequence of instructions that will output the data FF₁₆ to a byte-wide output port at address AB₁₆ of the I/O address space.

Solution:

First, the AL register is loaded with FF₁₆ as an immediate operand in the instruction

```
MOV  AL, 0FFH
```

Now the data in AL can be output to the byte-wide output port with the instruction

```
OUT  0ABH, AL
```

---

8.17 Input/Output Instructions

EXAMPLE

Write a sequence of instructions that will output FF₁₆ to a byte-wide output port at address AB₁₆ of the I/O address space.

Solution:

First, the AL register is loaded with FF₁₆ as an immediate operand in the instruction

```
MOV  AL, 0FFH
```

Now the data in AL can be output to the byte-wide output port with the instruction

```
OUT  0ABH, AL
```

---

8.17 Input/Output Instructions

EXAMPLE

Write a series of instructions that will output FF₁₆ to an output port located at address B000₁₆ of the I/O address space.

Solution:

The DX register must first be loaded with the address of the output port. This is done with the instruction

```
MOV  DX, 0B000H
```

Next, the data that are to be output must be loaded into AL with the instruction

```
MOV  AL, 0FFH
```

Finally, the data are output with the instruction

```
OUT  DX, AL
```
8.17 Input/Output Instructions

**EXAMPLE**

Data are to be read in from two byte-wide input ports at addresses AA₁₆ and A9₁₆ and then output as a word-wide output port at address B000₁₆. Write a sequence of instructions to perform this input/output operation.

**Solution:**

First read in the byte at address AA₁₆ into AL and move it into AH.

\[
\text{IN AL, 0AAH}
\]

\[
\text{MOV AH, AL}
\]

Now the other byte can be read into AL by the instruction

\[
\text{IN AL, 09AH}
\]

And to write out the word of data

\[
\text{MOV DX, 0B000H}
\]

\[
\text{OUT DX, AX}
\]

8.18 Input/Output Bus Cycle

- Input bus cycle of the 8088

![Input bus cycle of the 8088](image-url)
8.18 Input/Output Bus Cycle

■ Output bus cycle of the 8088

8.18 Input/Output Bus Cycle

■ Input bus cycle of the 8086
8.18 Input/Output Bus Cycle

Output bus cycle of the 8086