

微机原理及接口技术

Hardware Principles and Interfacing of Modern Computer

Lecture 8: Hardware
Specification & Principles

陈启军，张 伟

Email: zhang_wi@mail.tongji.edu.cn

Dept. Of Control Science and Engineering, TongJi University



Reference

- **The Intel Microprocessors: 8086/8088...:
Architecture, Programming, and Interfacing,
[美]Barry B. Brey (巴里 B.布雷) 著, 机械工业出版社,
2005, ISBN 7-111-16052-5**
- **Intel CPU Reference Manual & Reference Design,
<http://www.intel.com>**

Attention

● 本章标题是“8086/8088 Hardware Specifications”，但是，它讲的并不仅仅是8086/8088的硬件规范，其中穿插了很多硬件设计的原则和思路，阅读时应注意总结和体会

- 本章尽管未涉及计算机工作的基本原理，但其内容却是后续章节的基础，值得认真阅读

Content

● CPU(8086/8088)的引脚定义和引脚功能(Pin Functions)

- 要特别体会CPU芯片是如何与总线连接的，特别是控制总线中的R/W和Interrupt处理部分

● 时钟生成器(Clock Generator) 8284A

- 脉冲(pulse)和时钟(clock)

● 数字信号的缓冲与锁定(Buffering & Latching)

● 总线时序(Bus Timing)

- 时序图(Timing diagram)
- 读写是如何实现的

● 状态转换机制

- 就绪和等待状态(Ready & Wait State)

● 最小系统原则

Pin-Outs and Pin Functions

- **Key Idea:** 要使用一个集成电路芯片，必须阅读其参考手册，充分了解该器件的特性
 - 对CPU芯片的使用而言，我们最起码要了解其每个引脚的功能定义

Pin-Outs and Pin Functions

● 原理1：数字电路设计的原则

– 电源供给的要求

- 允许的电压波动：8086/8088 $5V + 10\% \Rightarrow$ 必要的稳压措施
- 足够的电源容量：360 mA maximum for 8086

– 引脚输入电流要求(current requirement for input pin)

– 引脚输出驱动能力的约束(current drive capability for output pin)

- 限制了输出引脚上不能够连接过多的器件 fan-out < 10
- $\Rightarrow$ 限制了总线上不能挂接过多的设备
- $\Rightarrow$ 引入高阻态以改善负载状况

Pin-Outs and Pin Functions

● 原理1：数字电路设计的原则(续)

— 逻辑0-1的规定

- 输入要求：逻辑0： $< 0.8V$ 逻辑1： $> 2.0V$
- 输出要求：逻辑0： $< 0.45V$ 逻辑1： $> 2.4V$

注意，逻辑0和1之间存在较大范围未定义区间，实际中应避免出现这种情况。

两个逻辑0和两个逻辑1也不一致，中间也存在一个区间(noise immunity)，它约束了数字信号在导线上的最大传输距离，如果过长就需要加缓冲器或者锁存器。

— 高频问题

— 低功耗设计问题

- 后两个问题目前很突出

Pin-Outs and Pin Functions

- 问题：总线(Bus)在计算机中起着连接各个不同部件(例如CPU, Memory和硬盘)的作用，那么这种连接究竟是如何实现的？
 - 以CPU和总线的连接为例

Pin-Outs and Pin Functions

原理2：CPU与总线的连接原则——Pin-Pin直连

- 地址总线 $\Leftrightarrow$ 引脚AD7-AD0, AD15-AD8, A19-A16
- 数据总线 $\Leftrightarrow$ 引脚AD7-AD0, AD15-AD8
 - 引脚复用技术(multiplexing), 通过时序进行功能区分
- 控制总线 $\Leftrightarrow$ 引脚RD, WR, IO/M, INTR, NMI
- 其它功能引脚: RESET, CLK, Vcc, GND

Pin-Outs and Pin Functions

🌐 思考：为什么要引入引脚复用技术？它有什么好处和缺点？在芯片工作时如何判断复用之后的引脚的功能？

Clock Generator 8284A

Key Idea: 时钟，数字电路的动力之源

- 计算机作为典型的数字电路系统，其每一个数字部件的工作都需要时钟的推动。
- 不仅如此，通过调控时钟脉冲的先后顺序，也可以实现各个部件的顺序调控（例如同步工作还是异步异作，异步工作时各个器件之间又如何交互）

Clock Generator 8284A

● 概念：时钟，脉冲发生器/振荡器，时钟发生器

● 时钟生成器8284A

- 可以提供多种时钟信号
- 引脚规范：X1和X2, EFI, CLK, PCLK, RESET
- 时钟生成器的内部实现：
 - 晶体振荡器产生最原始的时钟信号，然后通过计数器对该时钟信号进行分频以得到各种类型的时钟脉冲序列，送往计算机内各个部件使用
- Reset信号的生成(p. 317, v6)

Clock Generator 8284A

- 思考：时钟生成器是如何与计算机内其它各个部件(例如)相连的？各个部件是如何获得自己所需的时钟脉冲的？如果时钟信号异常，整个计算机会表现出怎样的行为？
- 思考：计算机的Reset信号是如何产生并且生效的？它与其它的信号相比有什么特点？

Bus Buffering & Latching

- 概念：缓冲器(Buffer)和锁存器(Latch)
- 问题：为什么在数字电路中要引入缓冲器和锁存器？
 - 缓冲器：解决匹配问题
 - Logic 0: 32 mA sink current (p. 319, v6)
 - Logic 1: 5.2 mA source current
 - 锁存器：解决引脚复用之后的解复用(de-multiplexing)问题

Bus Buffering & Latching

锁存器的使用

- 使用锁存器和ALE Pin完成解复用的工作过程(p.317, v6)
- 锁存器与CPU和Bus的连接: Fig. 9-5,9-6 (p.318-319,v6)

缓冲器的使用

- 缓冲器与CPU和Bus的连接: Fig.9-7, 9-8
- 缓冲器的副作用: 引入额外的延迟(timing delay)
- 锁存器的额外好处: 代替缓冲器, 提高驱动能力

Bus Timing

● **Key Idea:** 在正确的时间，正确的地点做正确的事情 => 做事的时机很重要

- 当不同的部件集合在一起完成同一任务时，必须相互协调，这主要是通过控制时钟脉冲信号实现的

=> 时钟的电平

- 时钟脉冲给出和收回的时机也很重要

=> 时钟的边沿(上升沿和下降沿)

Bus Timing

● 时序图(Timing Diagram)的阅读

- 把握电平变化的先后顺序和边沿跳变的作用(voltage & edge)

● 基本的读写时序(Read/Write Timing)

Fig. 9-9, 9-10 (p.321-323, v6)

- T1: used by the processor to provide the memory or I/O with the address.
- T2: provides access time to the memory and also is where the READY input is sampled.
- T3: the data are sampled or sent to the memory or I/O
- T4: to deactivate the control signals

State Transition

● **Key Idea:** 一个系统的行为可以通过其状态转换图进行严格的描述

- 相比时序图，状态图是一个更接近用户的、可以描述系统行为的工具，例如PCI总线的行为可以通过一个PCI状态机(图)描述

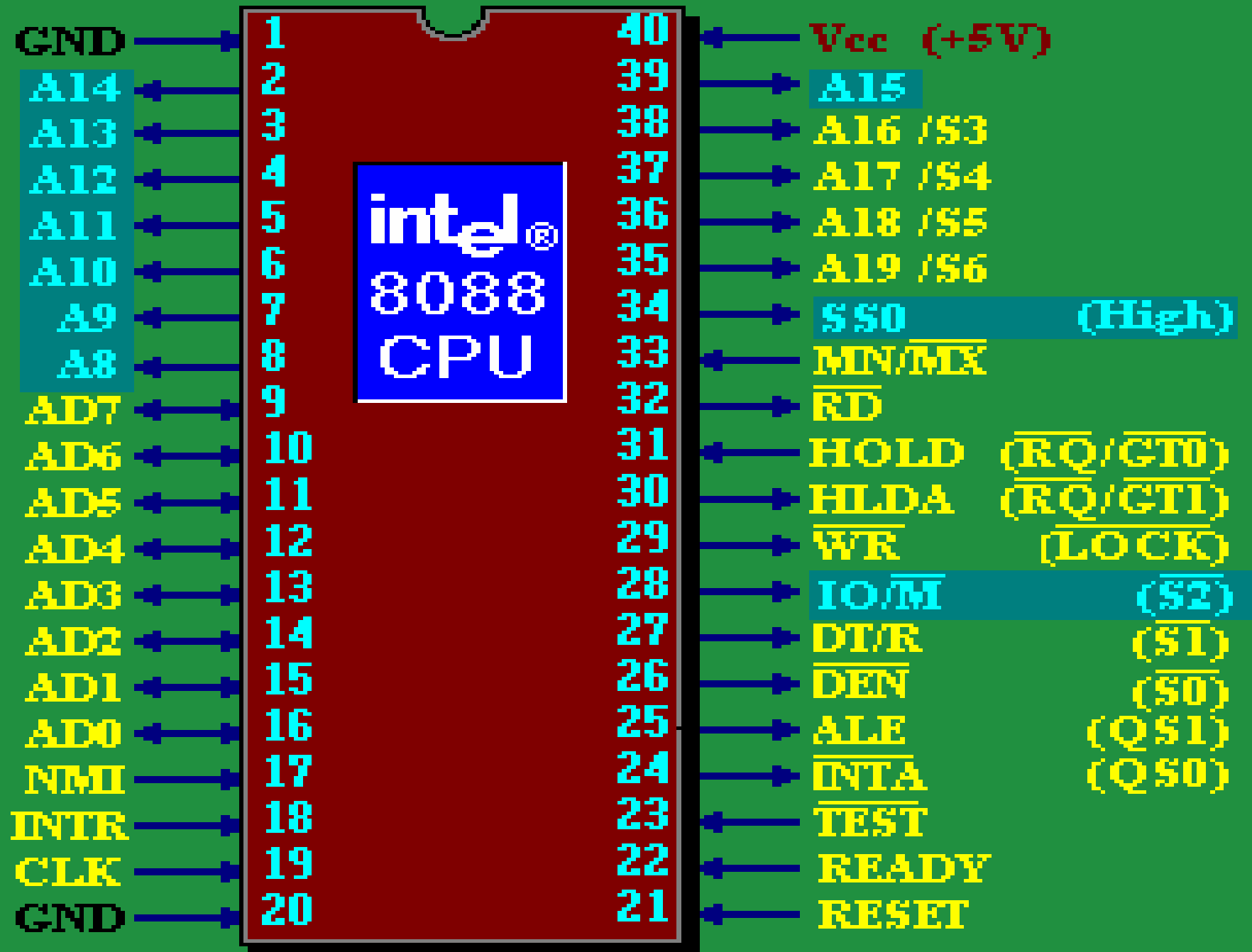
[历史状态] + [当前输入] = [未来状态] + [相应处理]

课本9-5节讲述的只是上述思想一个简单的示例

Minimum System

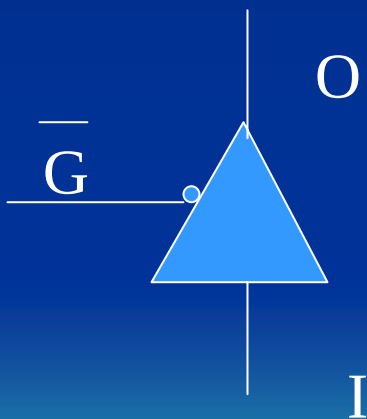
Key Idea: 从最简单的做起

- Minimum Mode和Maximum Mode
 - 依据控制信号的产生方式不同



8088的地址线

- AD0~AD7: 地址/数据分时复用, 双向, 三态。
- A8~A15: 单纯地址线, 单向, 三态。
- A16/S3~A19/S6: 地址/状态分时复用, 单向, 三态。



$\overline{G}$	I	O
0	1	1
	0	0
1	X	高阻

系统复位操作

8088CPU通过RESET引脚上的触发信号来引起8088系统复位和启动。当信号变成高电平时，8088CPU结束现行操作，各内部寄存器复位成初值，如表：

复位时各内部寄存器的值	
标志寄存器	清零
指令指针IP	0000H
CS寄存器	FFFFH
DS寄存器	0000H
SS寄存器	0000H
ES寄存器	0000H
指令队列	变空
其他寄存器	0000H

8086的控制线

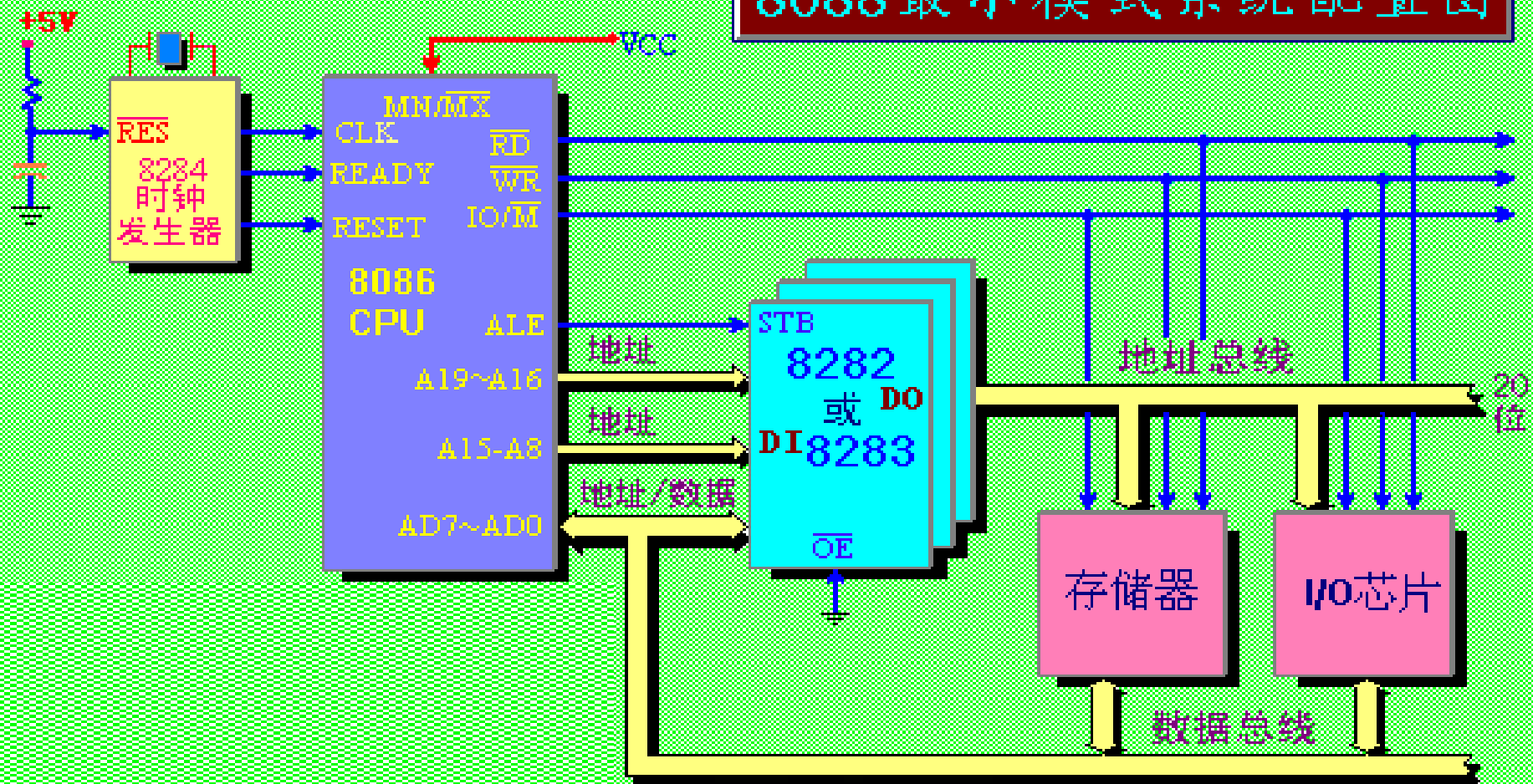
- RESET
- READY
- - RD
- M/-IO
- INTR
- NMI
- -TEST
- MN/- MX

8086系统配置

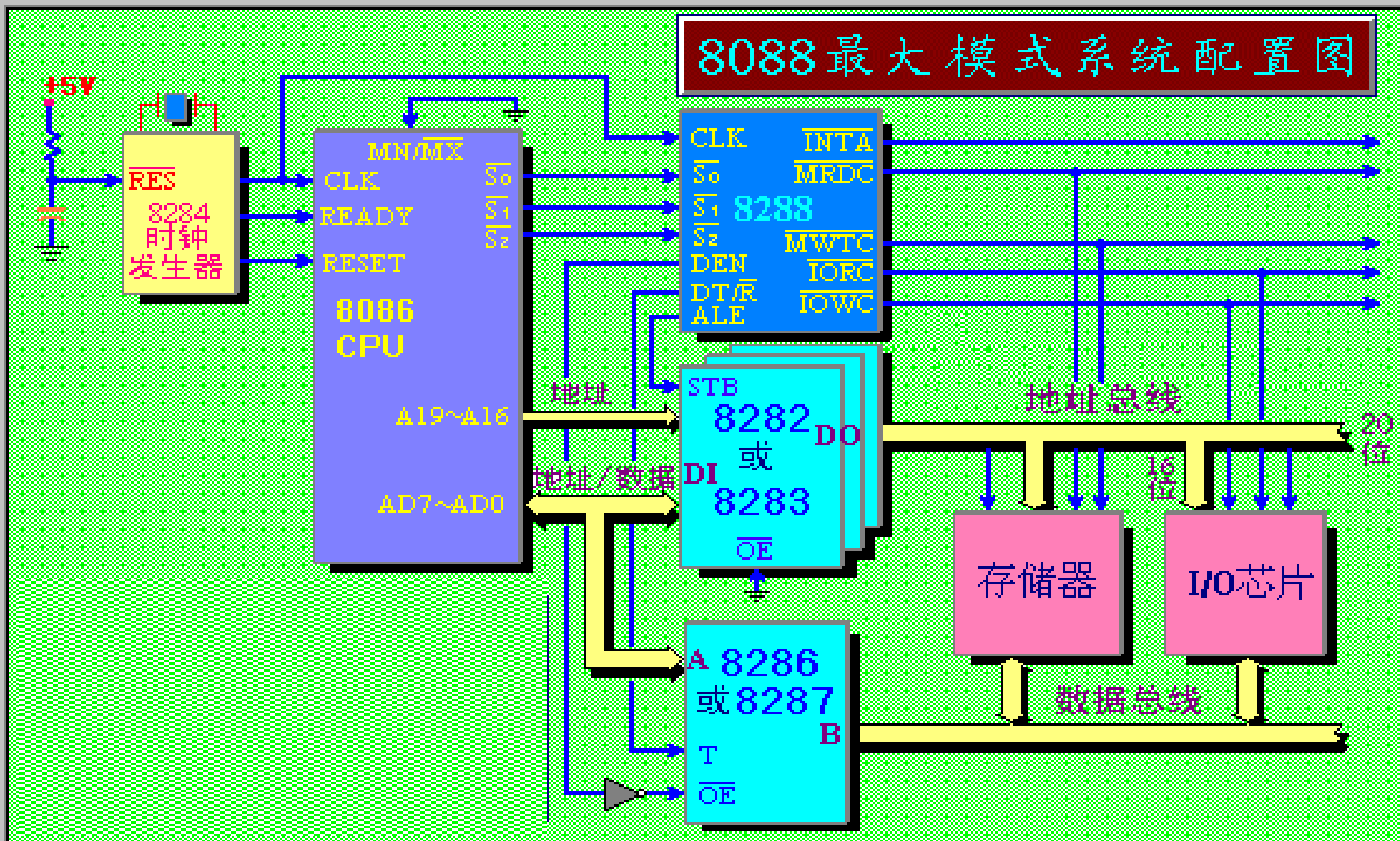
- 8086系统有最小模式和最大模式两种系统配置方式
- ❖ 当 $\overline{MN}/MX$ 接低电平时，构成最大模式。这时：
 - ❖ 系统中允许多个CPU共同工作。
 - ❖ 控制信号有的直接由CPU产生或接收，有的则需总线控制器经译码后产生。
- ❖ 当 $\overline{MN}/MX$ 接高电平时，构成最小模式。这时：
 - ❖ 系统中只有一个CPU。
 - ❖ 所有的总线控制信号均由CPU直接产生或接收。

8088系统配置

8088最小模式系统配置图



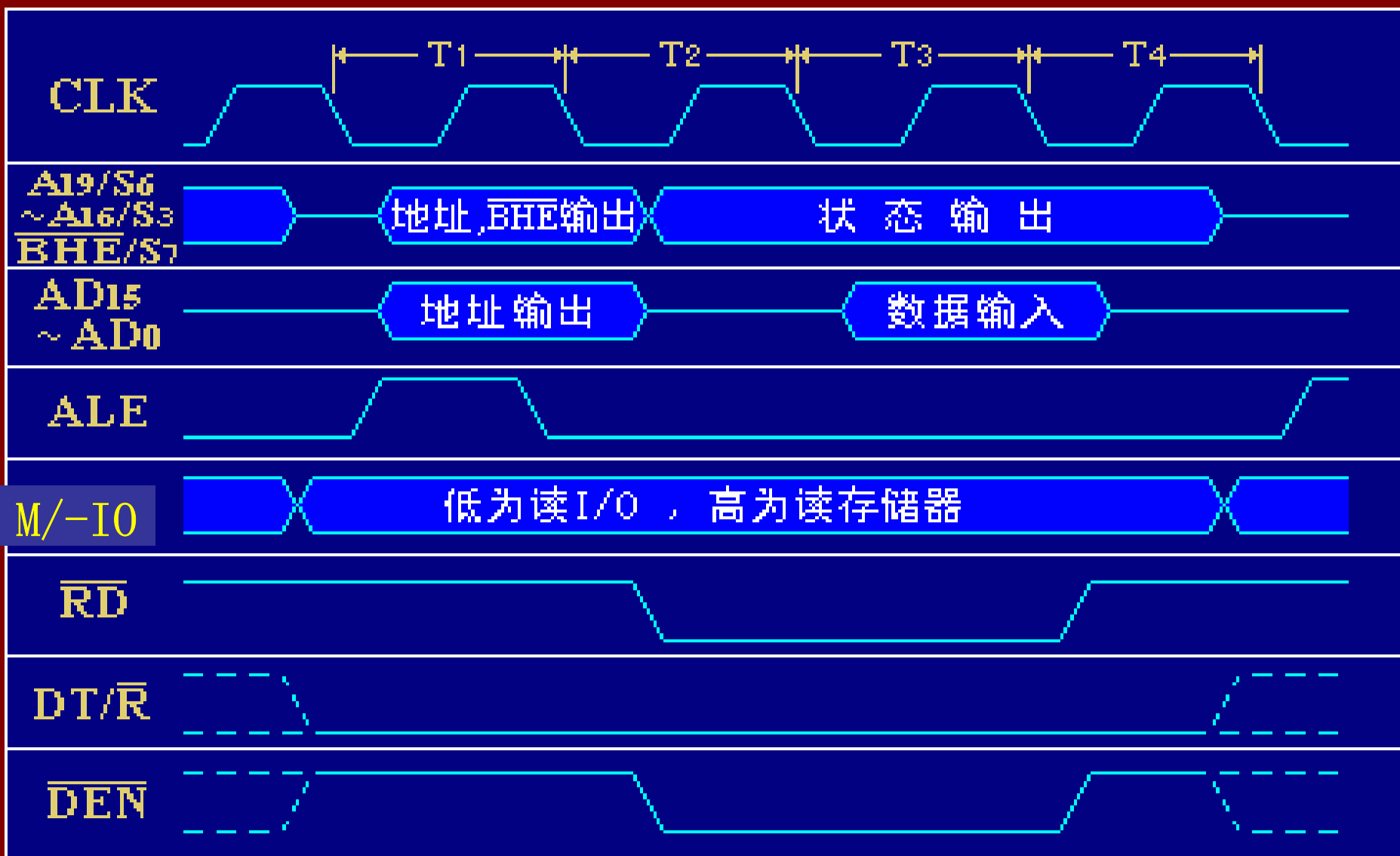
8088系统配置



时序中三个周期的定义

- **指令周期**：执行一条指令所需要的时间。不同指令的指令周期的长短是不同的，一个指令周期由几个总线周期组成。
- **总线周期**：BIU完成一次访问存储器或I/O端口操作所需要的时间，称作一个总线周期。一个总线周期由几个T状态组成。
- **时钟周期**：CPU时钟频率的倒数，也称T状态。在8088中，每个总线周期至少包含4个时钟周期（ $T_1 \sim T_4$ ）。一般情况下， T_1 状态传送地址， $T_2 \sim T_4$ 状态传送数据。

读总线周期



写总线周期

