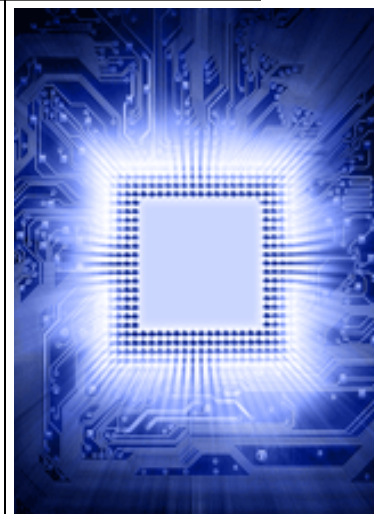


第六章 组合逻辑设计实践

南京大学人工智能学院
2018-2019 春季





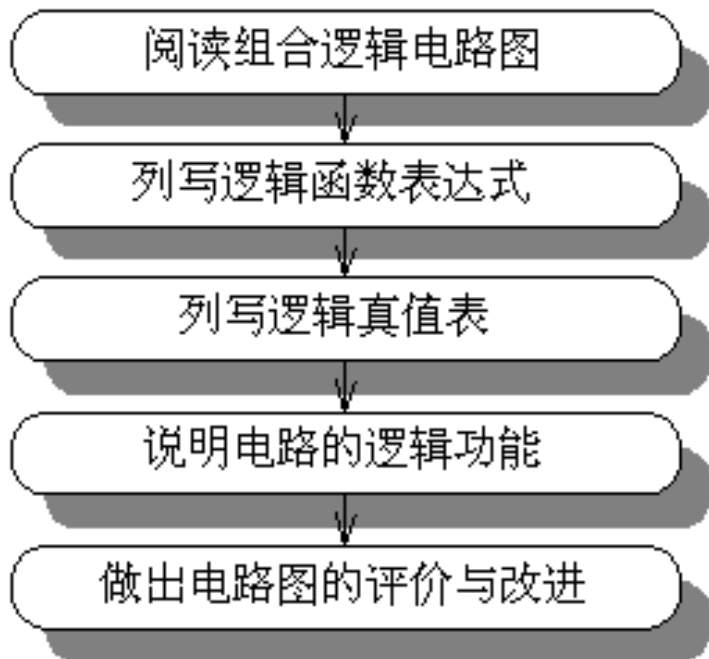
本章学习导引



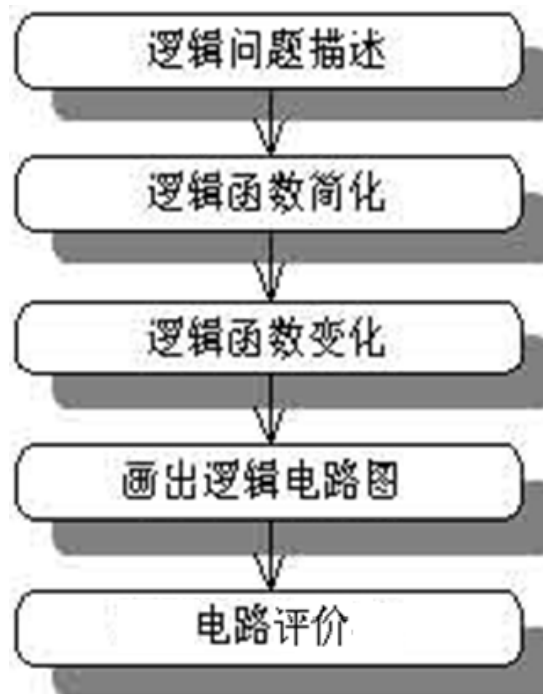
● 组合逻辑设计：

- 根据应用需求设计出逻辑电路图的过程。
- 确定的输入产生确定的输出。

分析过程

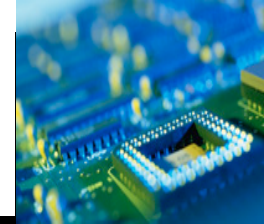


设计过程





补充例



- 设计半加器，对两个一位二进制数相加，得到“和”及“进位”（半加不考虑低位的进位）

1。列出真值表

A	B	S_H	C_H
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



补充例



- 2. 布尔函数表达式:

$$\bar{S}_H = AB + \bar{A}\bar{B} \quad C_H = AB$$

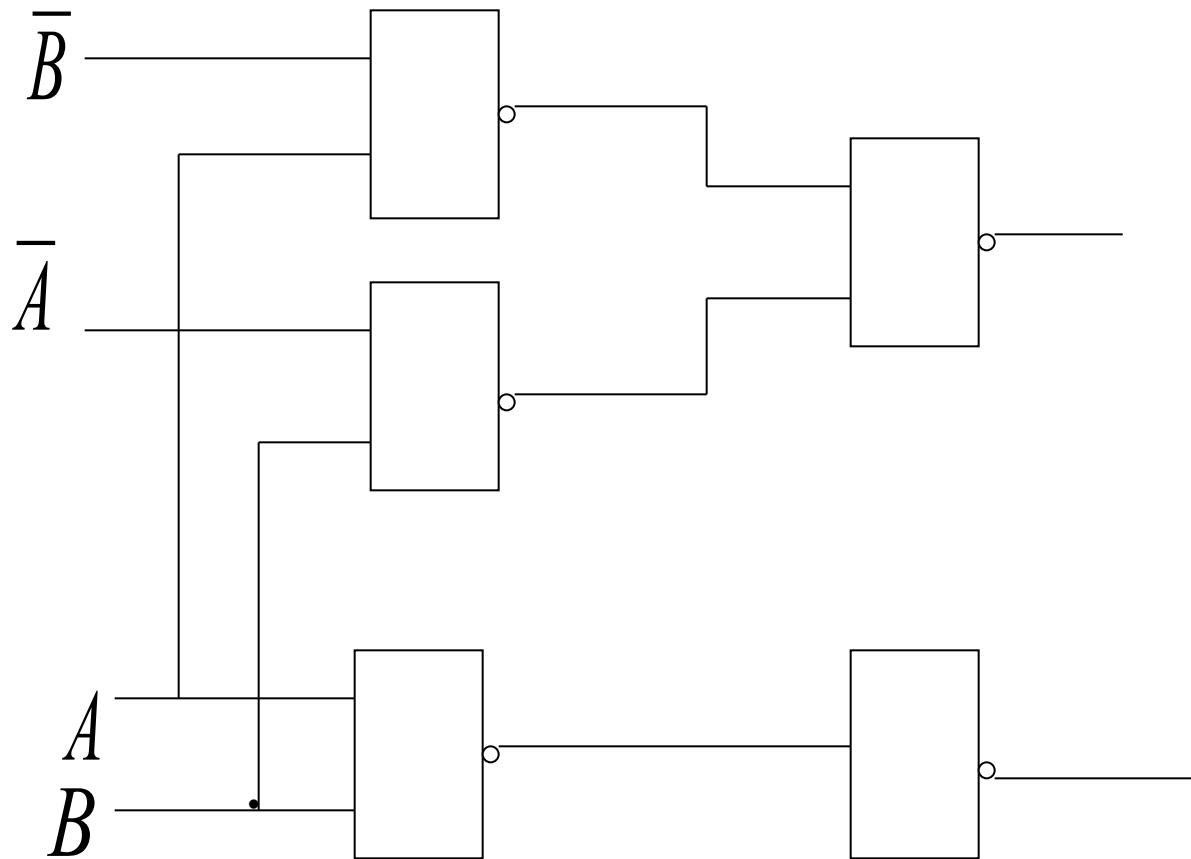
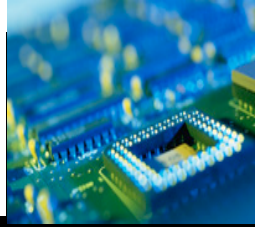
- 3. 画卡诺图

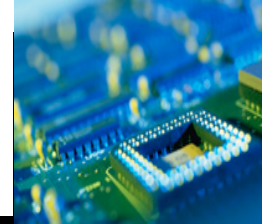
		<i>B</i>	
		0	1
<i>A</i>	0	0	1
	1	1	0

		<i>B</i>	
		0	1
<i>A</i>	0	0	0
	1	0	1

- 4. 画逻辑电路（用与非门实现）

$$S_H = \overline{\overline{AB + \bar{A}\bar{B}}} = \overline{\overline{AB}\overline{\bar{A}\bar{B}}} \quad C_H = \overline{\overline{AB}}$$



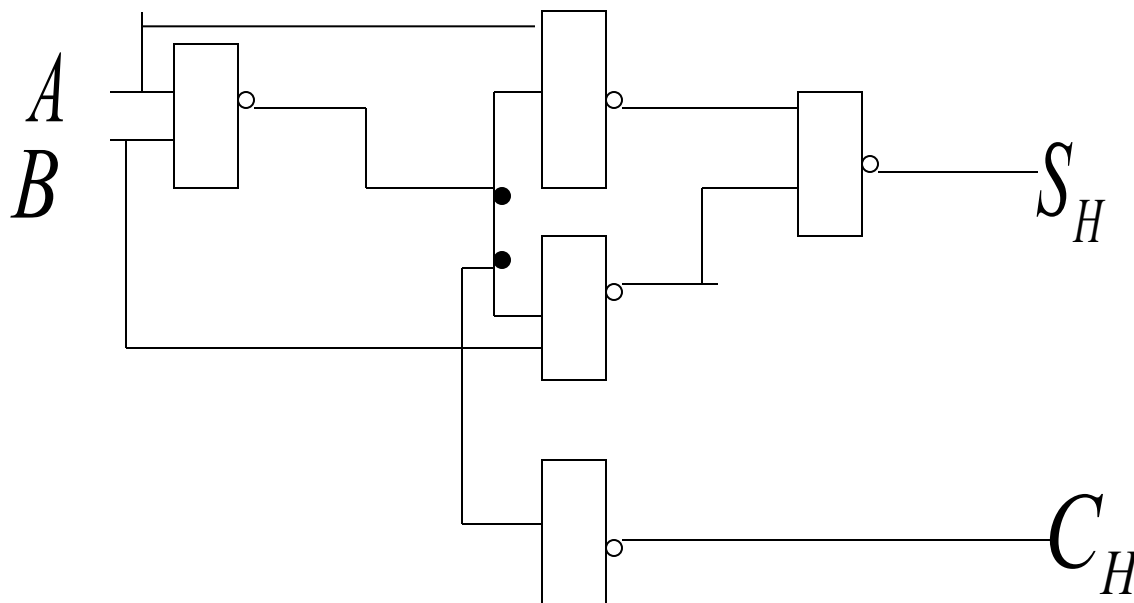


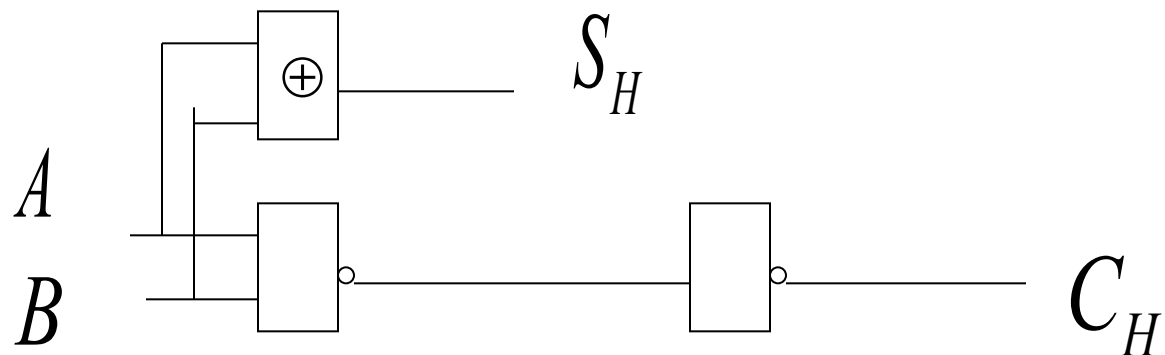
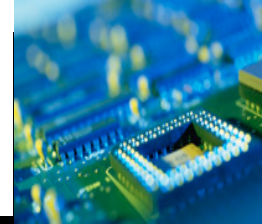
- 有时要求输入变量只能用原变量形式，可作相应的变换：对上例：

$$S_H = \overline{\overline{A}B\overline{A}\overline{B}} = \overline{\overline{A}\overline{B}\overline{A}B} = \overline{A(\overline{A} + \overline{B})B(\overline{A} + \overline{B})} =$$

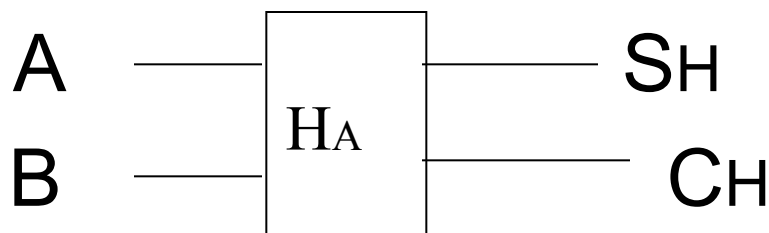
$$\overline{\overline{A}AB\overline{B}AB}$$

$$C_H = \overline{\overline{AB}}$$



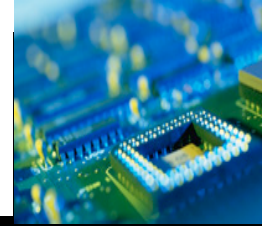


对一些常用的基本逻辑电路可封装为一个基本部件，在其它较复杂的电路中用（对上例）





本章学习导引



- 大型复杂应用问题的设计要考虑：
 - 规范化；
 - 结构化；
 - 构件化。



主要内容



- 设计文档编制标准
- 电路的定时
- 组合**PLD**的内部结构
- 译码器
- 编码器
- 三态器件
- 多路复用器
- ...



1、设计文档编制标准



- 设计文档是项目设计正确、可维护的重要基础。
- 设计文档可以是综合性的单文档描述，也可以是分列的多文档描述形态。
- 项目设计文档一般包括：
 - 规格说明书；
 - 方框图；
 - 原理图；
 - 定时图；
 - 结构化逻辑器件描述；
 - 电路描述。



1、设计文档编制标准



- 规格说明书：
 - 准确描述电路或系统的输入、输出及功能。
 - 主要是外部特性的描述。
- 方框图：
 - 系统主要功能模块及其基本互连的非正式图示描述。
- 原理图：
 - 系统的电气元件、元件间互连和构建系统所需的全部细节的正式说明，包括**IC**类型、参考标志符和引脚编号。（和常说的逻辑图不同）
 - 能够生成生产用的材料清单。



1、设计文档编制标准



- 定时图：
 - 说明各种逻辑信号的值随时间的变化情况。
 - 关键信号之间的因果延迟。
- 结构化逻辑器件描述；
 - 描述PLD、FPGA或者ASIC的内部功能设计。
 - 通常用HDL描述；也可以用逻辑表达式、状态表、状态图等形式。
- 电路描述
 - 电路功能设计的描述性文本，和其它文档一起解释电路的工作原理。应该包括设计的假设、限制等。



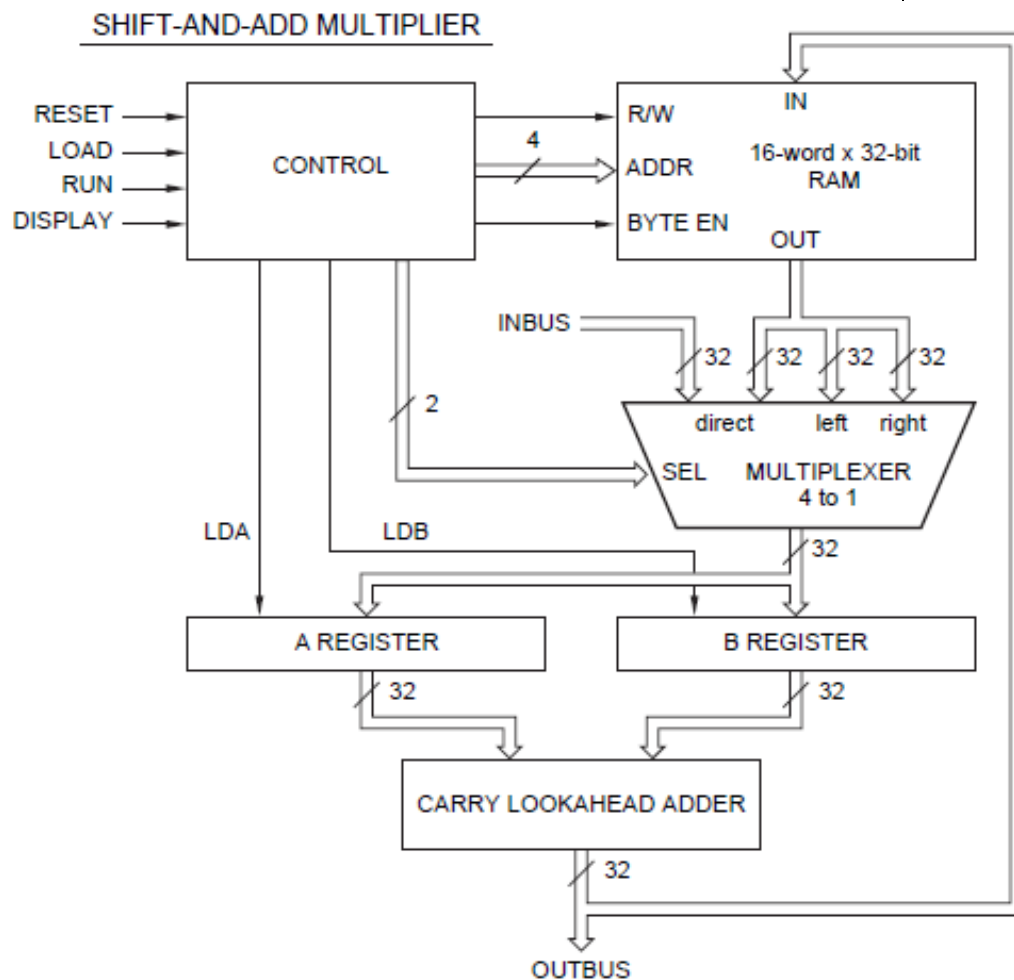
组合逻辑设计实践



1.1 方框图



- 展示系统的输入、输出，功能模块、内部数据通路和重要的控制信号。





1.1 方框图



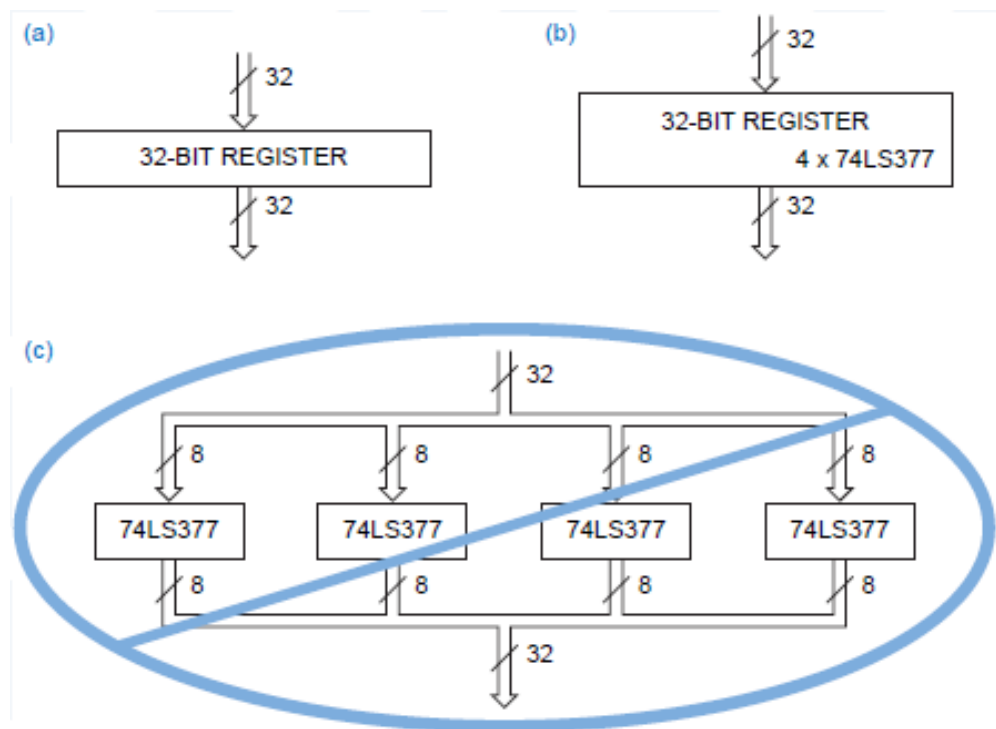
- 要点：
 - 能展示最重要的系统元素以及工作原理。
 - 每个方框图不要超过1页。
 - 大型系统可以另附页描述子系统。
 - 重要的控制信号和总线要命名。
 - 明确表明控制流和数据流。
 - 不要包含细节。
 - (generally, NO gate symbols, NO pins).



1.1 方框图



- 可以不出现的内容：
 - 实现模块细节
 - 有效电平





1.2 原理图



- 门的符号：
 - 基本逻辑门的形状

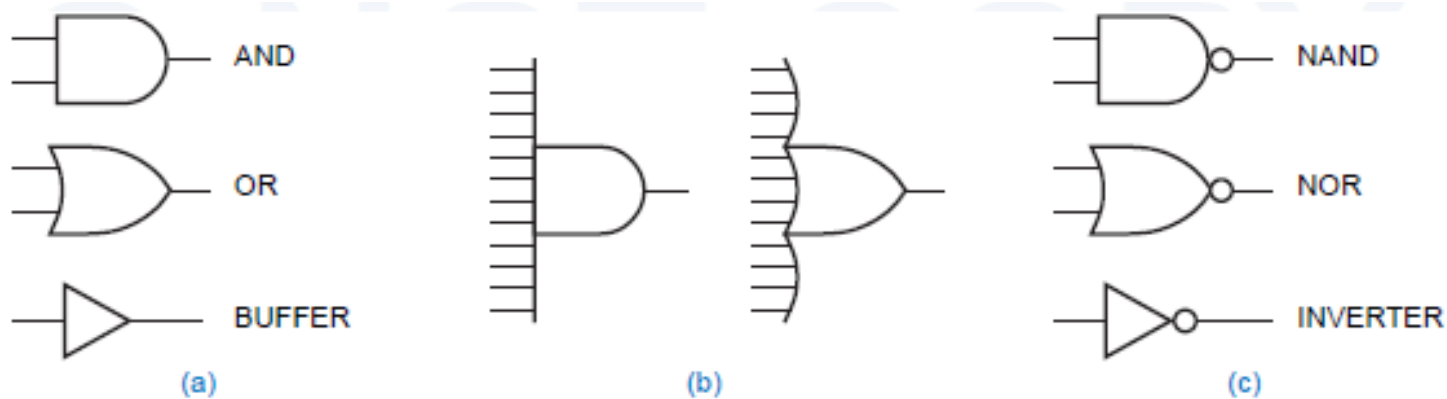


Figure 5-3 Shapes for basic logic gates: (a) AND, OR, and buffers; (b) expansion of inputs; (c) inversion bubbles.



1.2 原理图



- 门的符号：
 - 广义德·摩根定理下的等效符号

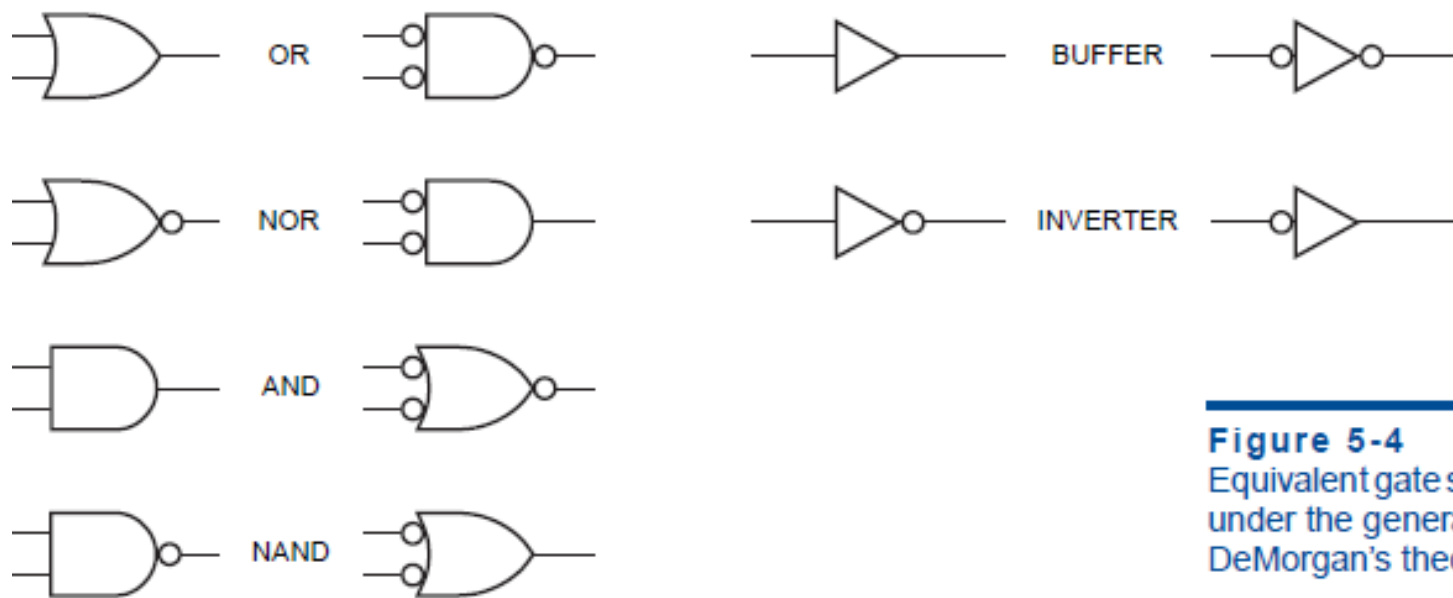


Figure 5-4
Equivalent gate symbols
under the generalized
DeMorgan's theorem.



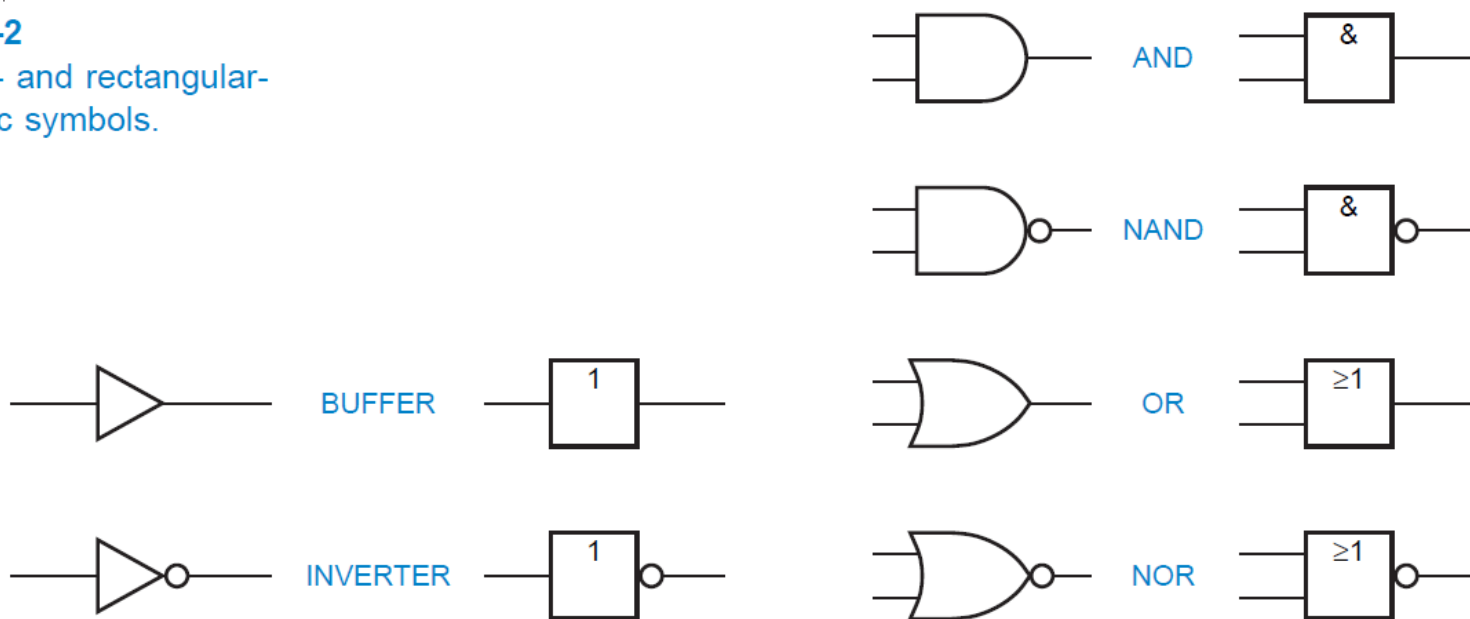
1.2 原理图



- 门的符号：
 - IEEE的两种逻辑符号表示方式：
 - ANSI/IEEE Std 91-1984
 - IEC 617

Figure A-2

Distinctive- and rectangular-shape logic symbols.





1.2 原理图



- 信号名和有效电平：
 - 在逻辑电路中，每个输入和输出都应有一个可描述的字符-数字标识，即信号名。
 - 实际应用中，会定义一些富有含义的命名形式。
 - 动作：GO, PAUSE;
 - 检测条件：READY, ERROR等。
 - 有效电平
 - 高电平有效：信号在高电平时完成命名的动作或表示命名的条件。
 - 低电平有效。
 - 信号有效、无效或者取消。



1.2 原理图



- 信号名和有效电平：
 - 有效电平命名习惯。

<i>Active Low</i>	<i>Active High</i>
READY-	READY+
ERROR.L	ERROR.H
ADDR15(L)	ADDR15(H)
RESET*	RESET
ENABLE~	ENABLE
~GO	GO
/RECEIVE	RECEIVE
TRANSMIT_L	TRANSMIT



1.2 原理图



- 引脚的有效电平：
 - 器件的引脚也和门电路的输入输出一样有高低有效电平之分。
 - 用反相圈表示低电平有效引脚，无反相圈表示高电平有效引脚。

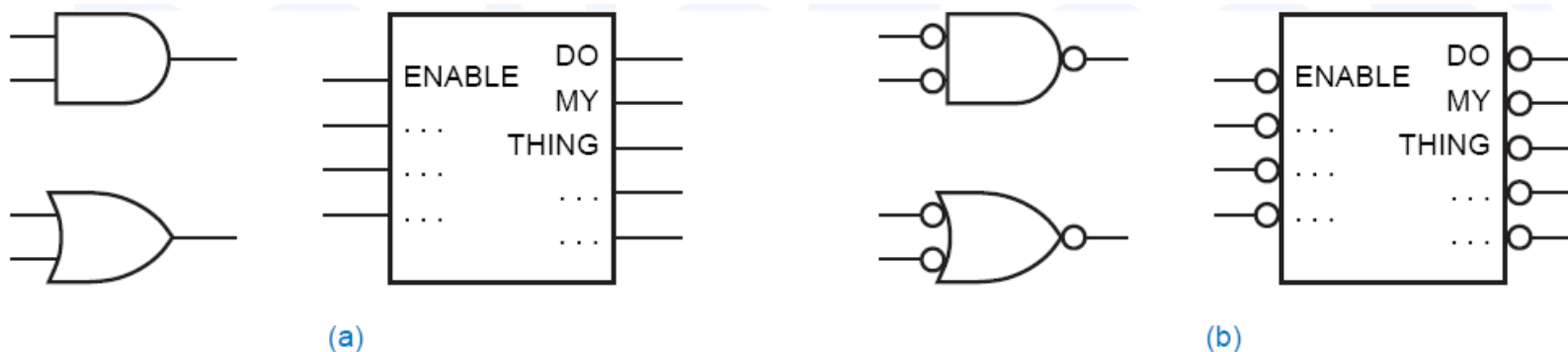


Figure 5-5 Logic symbols: (a) AND, OR, and a larger-scale logic element; (b) the same elements with active-low inputs and outputs.



1.2 原理图



- 引脚的有效电平：
 - 与功能、或功能、反相器和非反相器的不同方式



(a)



(b)



(c)



(d)



(a)



(b)



(c)



(d)



(a)



(b)



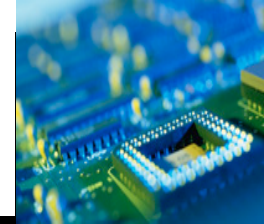
(c)



(d)



1.2 原理图



- 圈到圈逻辑设计：
 - 目的：使用包含有效电平标识的逻辑符号或信号命名方式使得逻辑电路更容易理解。

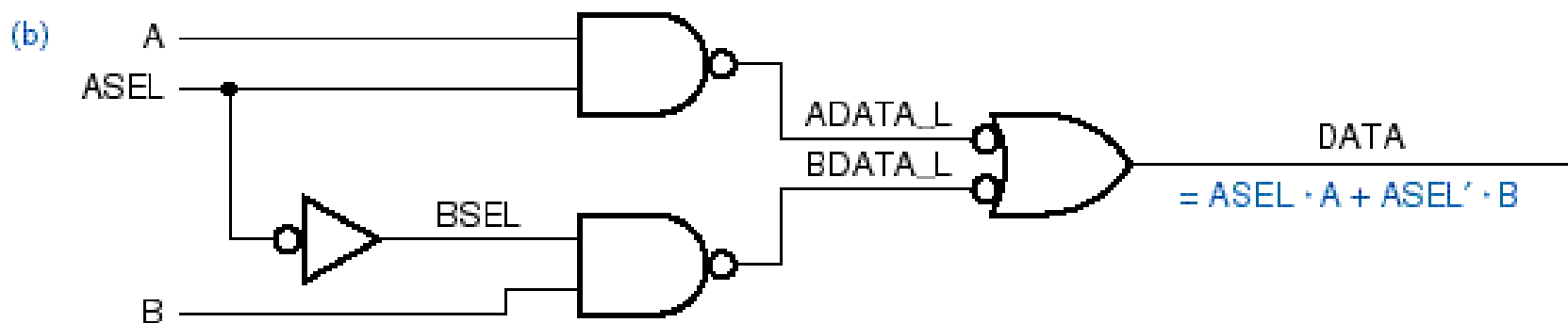
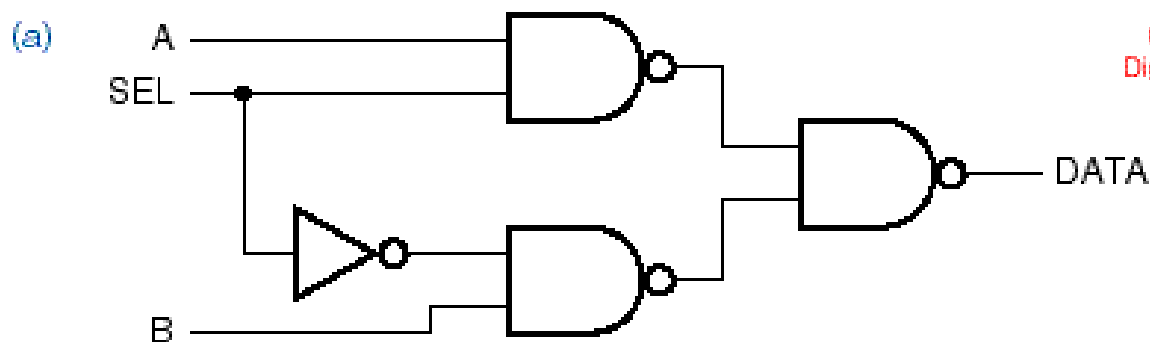




1.2 原理图



- 圈到圈逻辑设计：
 - 例子1: (a)语义含糊; (b)语义清楚



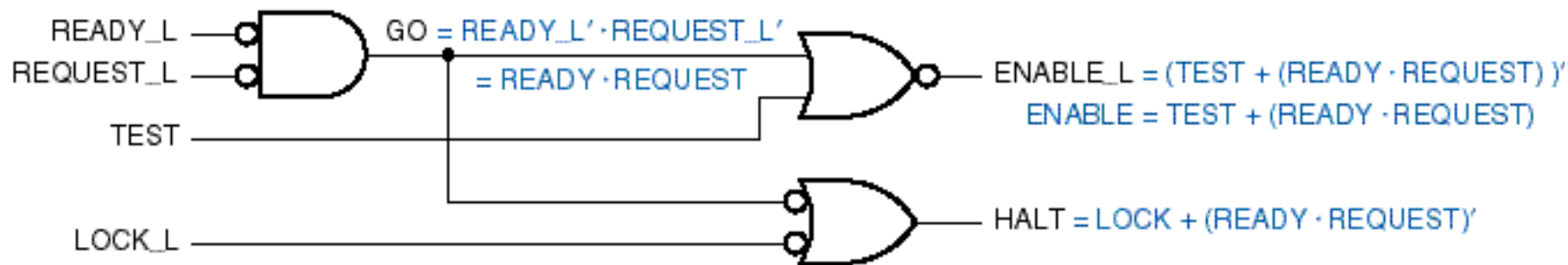


1.2 原理图



- 圈到圈逻辑设计:

- 例子2:



Copyright © 2000 by Prentice Hall, Inc.
Digital Design Principles and Practices, 3/e



1.2 原理图



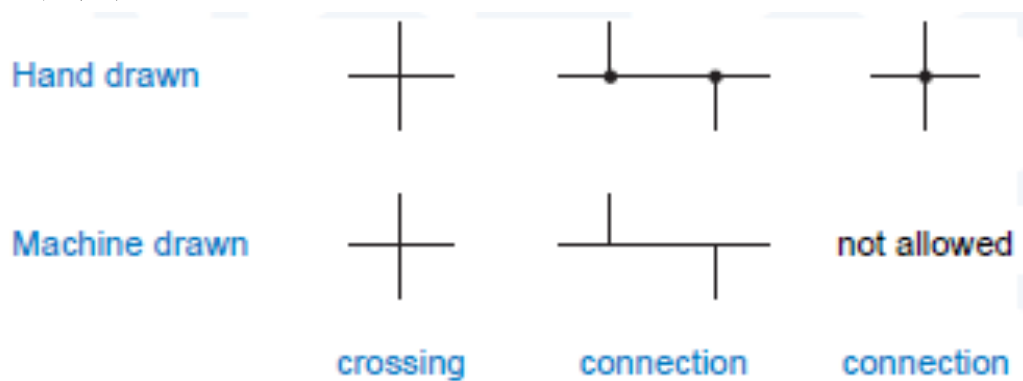
- 圈到圈逻辑设计原则：
 - 器件信号名应该与器件输出引脚有相同的有效电平。
 - 有反相圈则低电平有效，反之，高电平有效。
 - 输入引脚的有效电平应该与连接输入引脚的信号有效电平相同。
 - 应该尽量避免矛盾的连接。



1.2 原理图



- 绘制布局图：
 - 通常以输入在左边，输出在右边的正常方位画逻辑符号。
 - 门符号、逻辑组件符号；
 - 甚至是一页完整的原理图。中间出现输入、输出，尽量扩展到页边缘。
- 交叉线画法



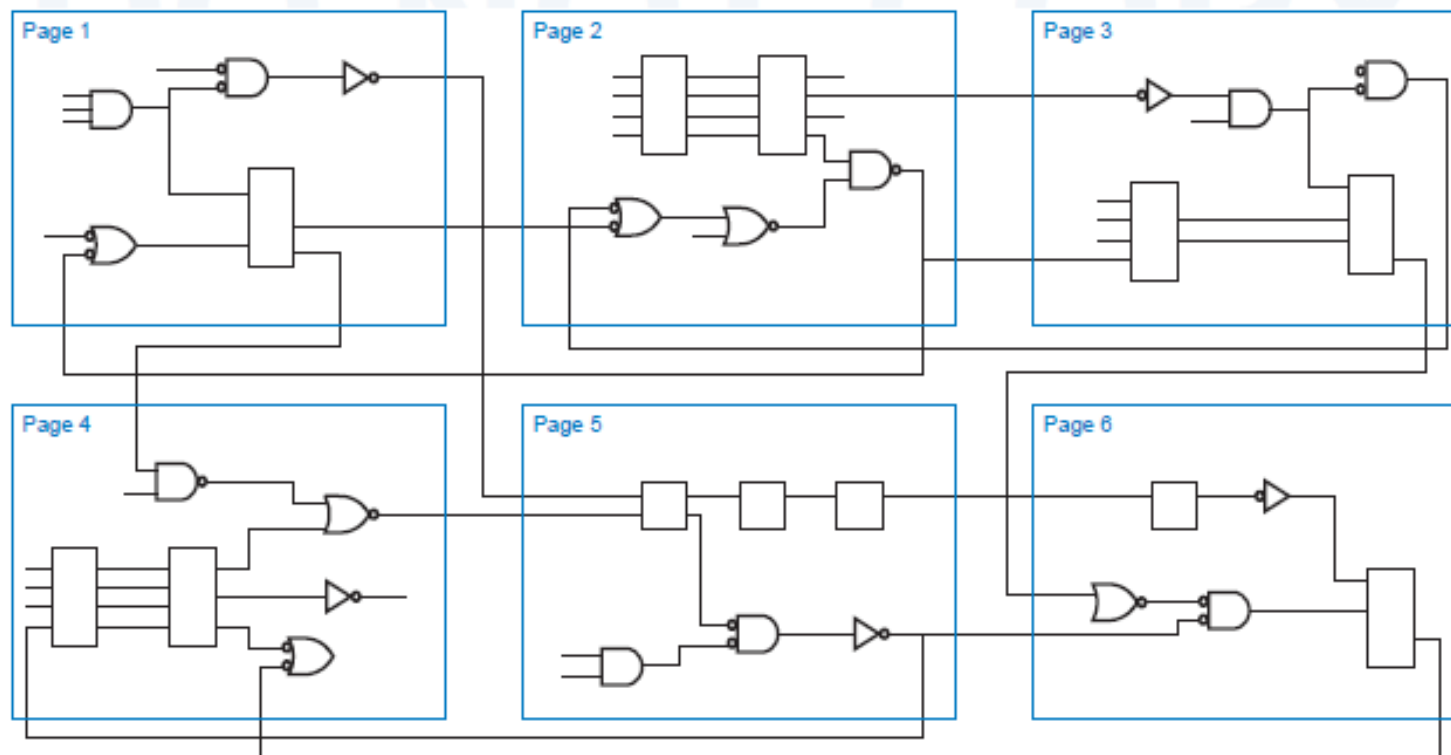


1.2 原理图



- 绘制布局图：多页原理图
 - “平面”结构画法

Figure 5-14 Flat schematic structure.



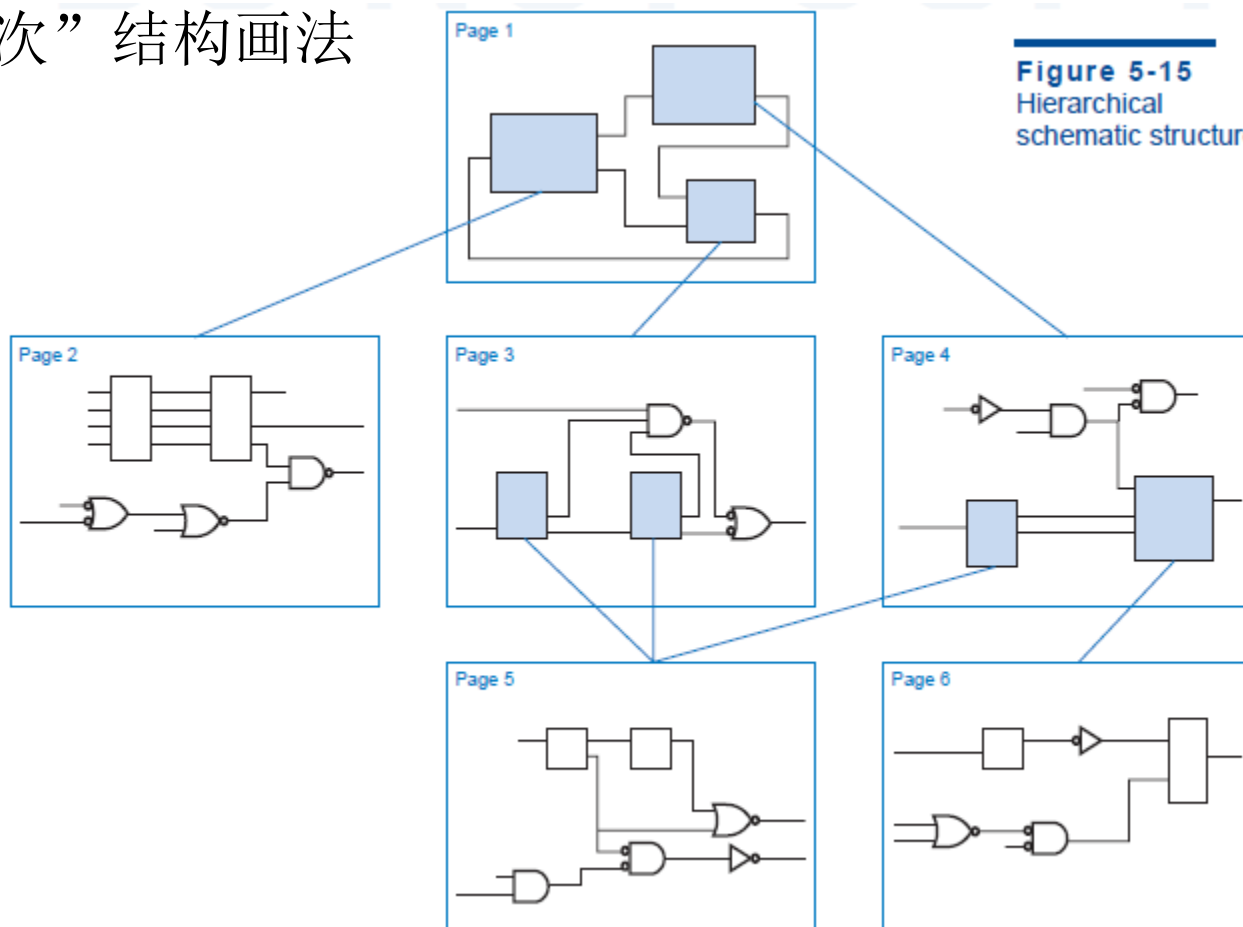


1.2 原理图



- 绘制布局图：多页原理图

- “层次”结构画法





1.2 原理图



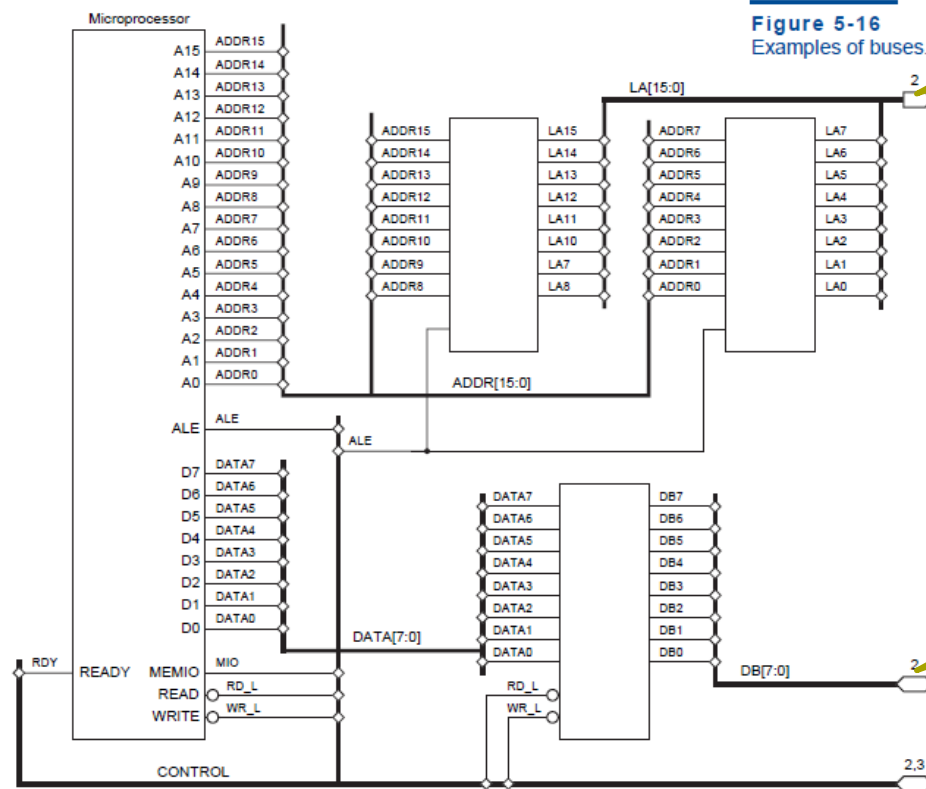
- 绘制布局图：总线

- 总线是两根或多根信号线的集合

- 地址总线ADDR[15:0]，数据总线DATA[31:0]，控制总线。

- 画法：

- 描述性名称
- 粗线
- 特殊的连接点



单向连接
到次页

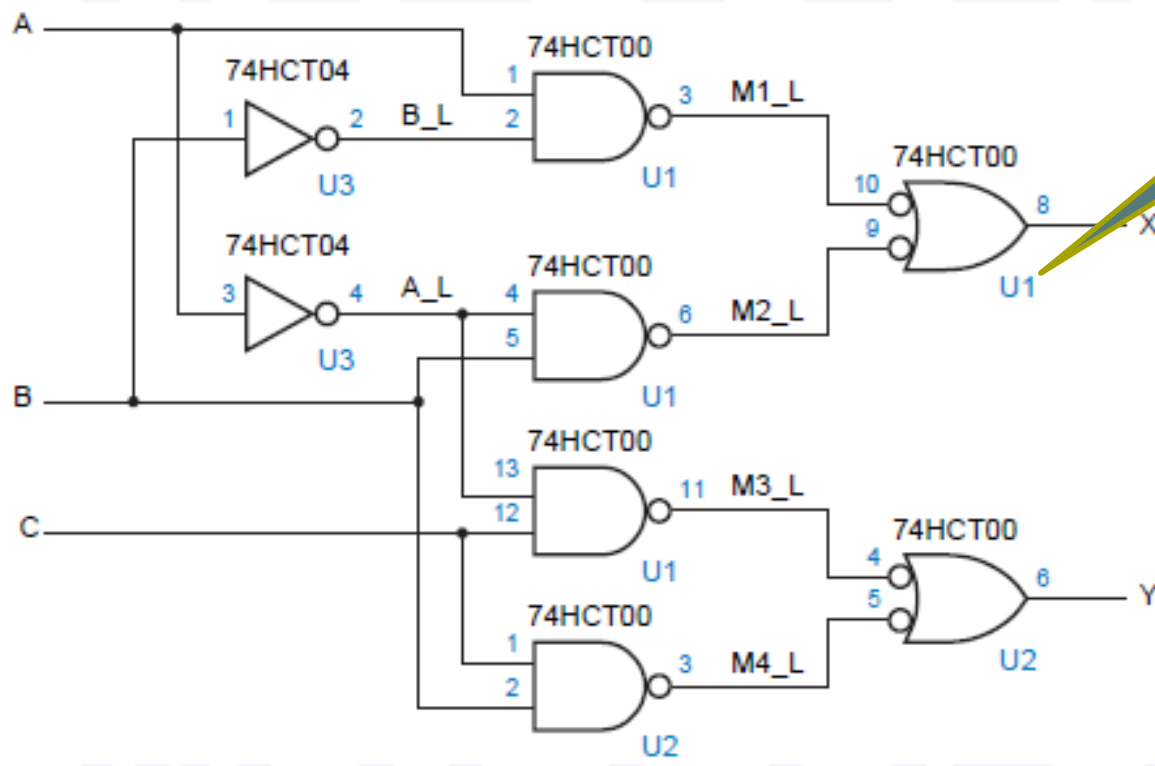
双向连接
到次页



1.2 原理图



- 绘制布局图：图示信息
 - 完整的原理图应该标示出IC类型、参考标识符以及引脚编号



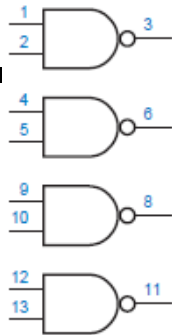
注意这些符号



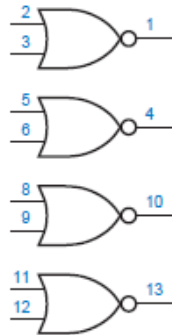
1.2 原理图

- 绘制布局图
 - 通用器件参考标识符和引脚

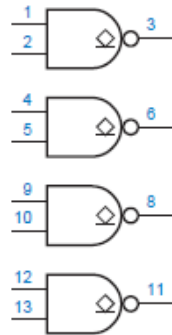
74x00



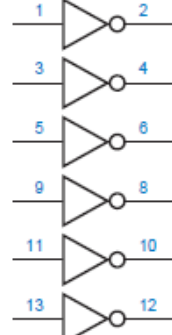
74x02



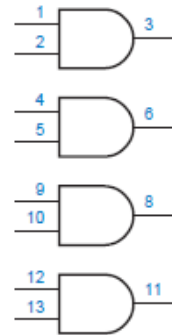
74x03



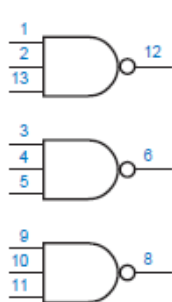
74x04



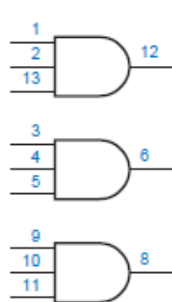
74x08



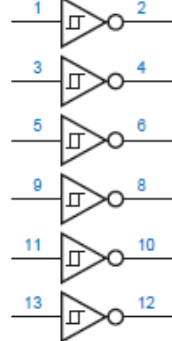
74x10



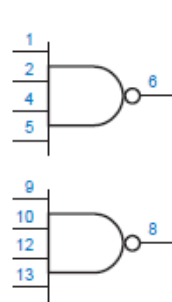
74x11



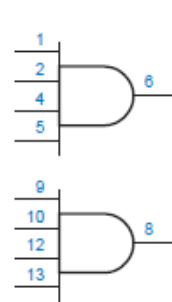
74x14



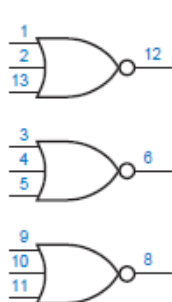
74x20



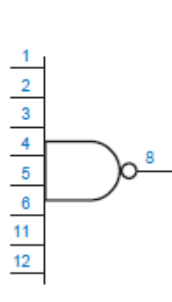
74x21



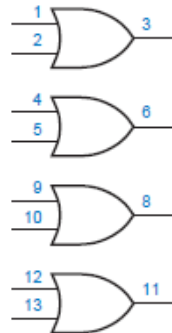
74x27



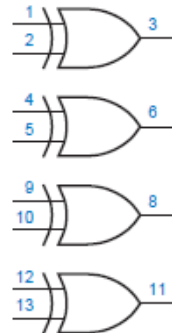
74x30



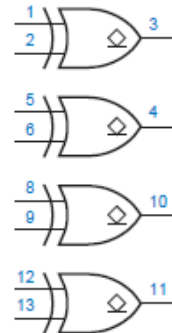
74x32



74x86



74x266

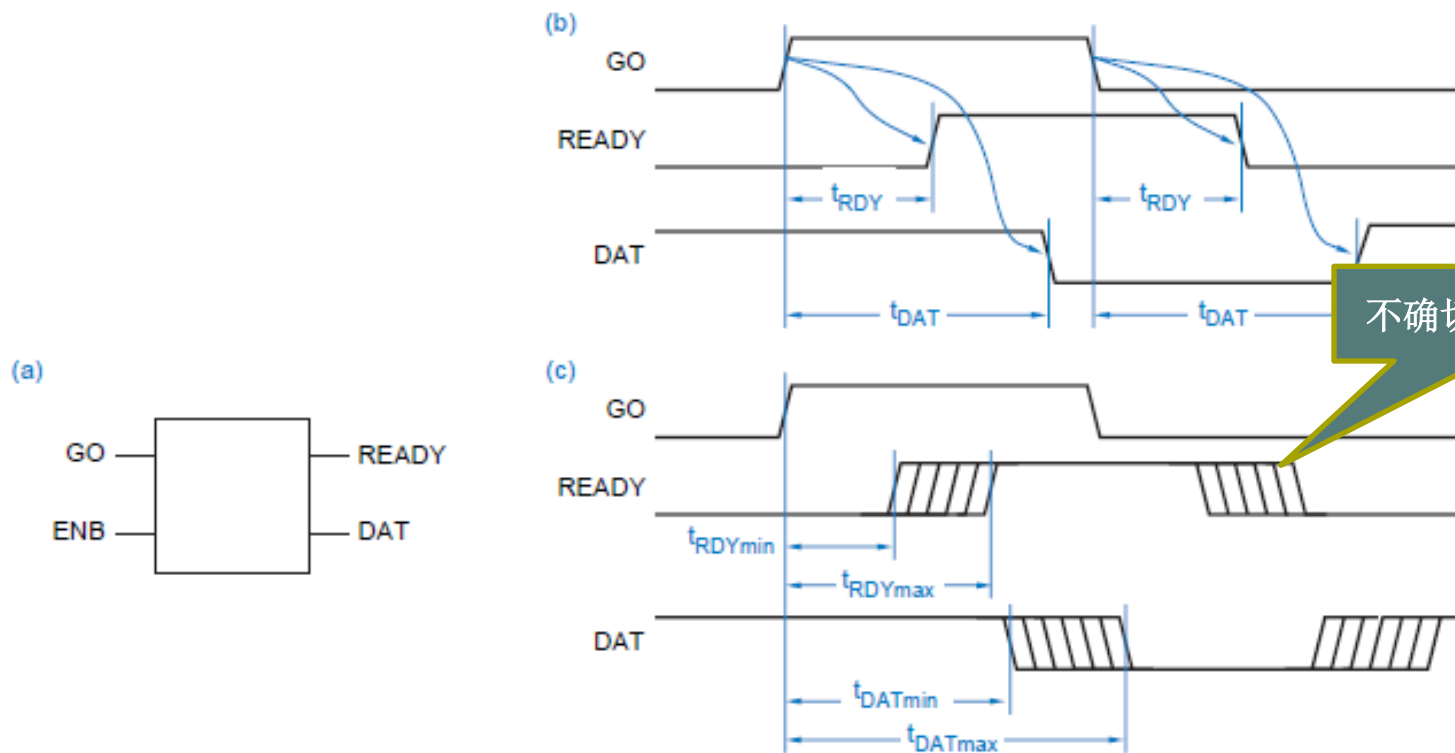




1.3 定时图



- 表明信号作为时间函数的逻辑行为。
 - 可用来解释系统内信号间的时序关系；
 - 也可用来定义加到系统上的外部信号的定时要求





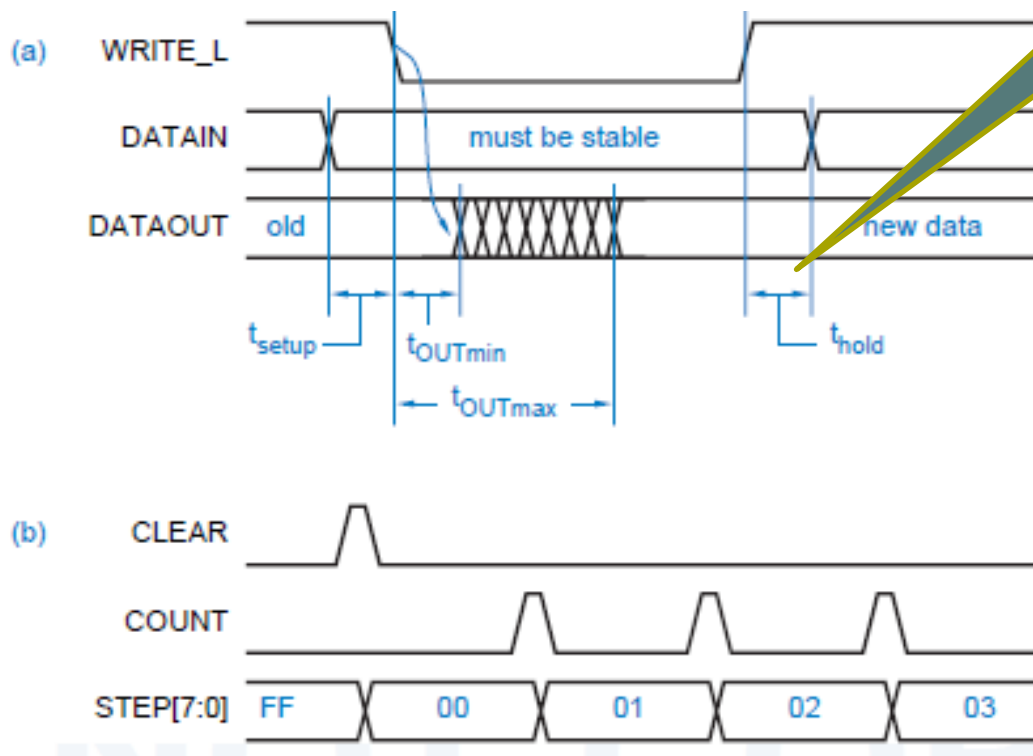
1.3 定时图



- 表明信号作为时间函数的逻辑行为。

- 确切和不确切的转换 图a;
- 顺序传输 图b。

这几个时间都有特殊含义





1.3 定时图



● 传输延迟

- 通路输入端的变化引起通路输出端变化所需的时间。
 - 低到高: T_{pLH} ; 高到低: T_{pHL} 。
- 较大电路中, 组合IC的逻辑设计者利用各个器件的规格说明分析整个电路的定时。
 - 整个电路的延迟等于各个通路延迟之和。

● 定时规格说明

- 最大延迟
- 典型延迟
- 最小延迟

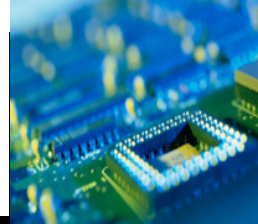
Table 5-2 Propagation delay in nanoseconds of selected 5-V CMOS and TTL SSI parts.

Part number	74HCT		74AHCT				74LS			
	Typical		Typical		Maximum		Typical		Maximum	
	t_{pLH}	t_{pHL}	t_{pLH}	t_{pHL}	t_{pLH}	t_{pHL}	t_{pLH}	t_{pHL}	t_{pLH}	t_{pHL}
'00, '10	11	35	5.5	5.5	9.0	9.0	9	10	15	15
'02	9	29	3.4	4.5	8.5	8.5	10	10	15	15
'04	11	35	5.5	5.5	8.5	8.5	9	10	15	15
'08, '11	11	35	5.5	5.5	9.0	9.0	8	10	15	20

74xx00



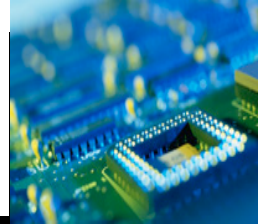
2、电路的定时分析



- 一个设计的电路定时需要详细分析各部分器件的逻辑行为才能最终确定。
 - 反向门时， T_{pLH} 和 T_{pHL} 趋于中和。
 - 非反向门时， T_{pLH} 和 T_{pHL} 趋于加宽。
 - 最坏延迟需要分析电路中各种通路的延迟。
 - 大型电路一定需要自动工具来辅助分析。
- 定时分析工具
 - 早期阶段，使用定时分析工具即可估计延迟。
 - 综合阶段，实际的通路和理想的有差距，需要调整以适应选定器件的要求。
 - 定时终结Timing Closure：往往几个月。



3、组合型PLD



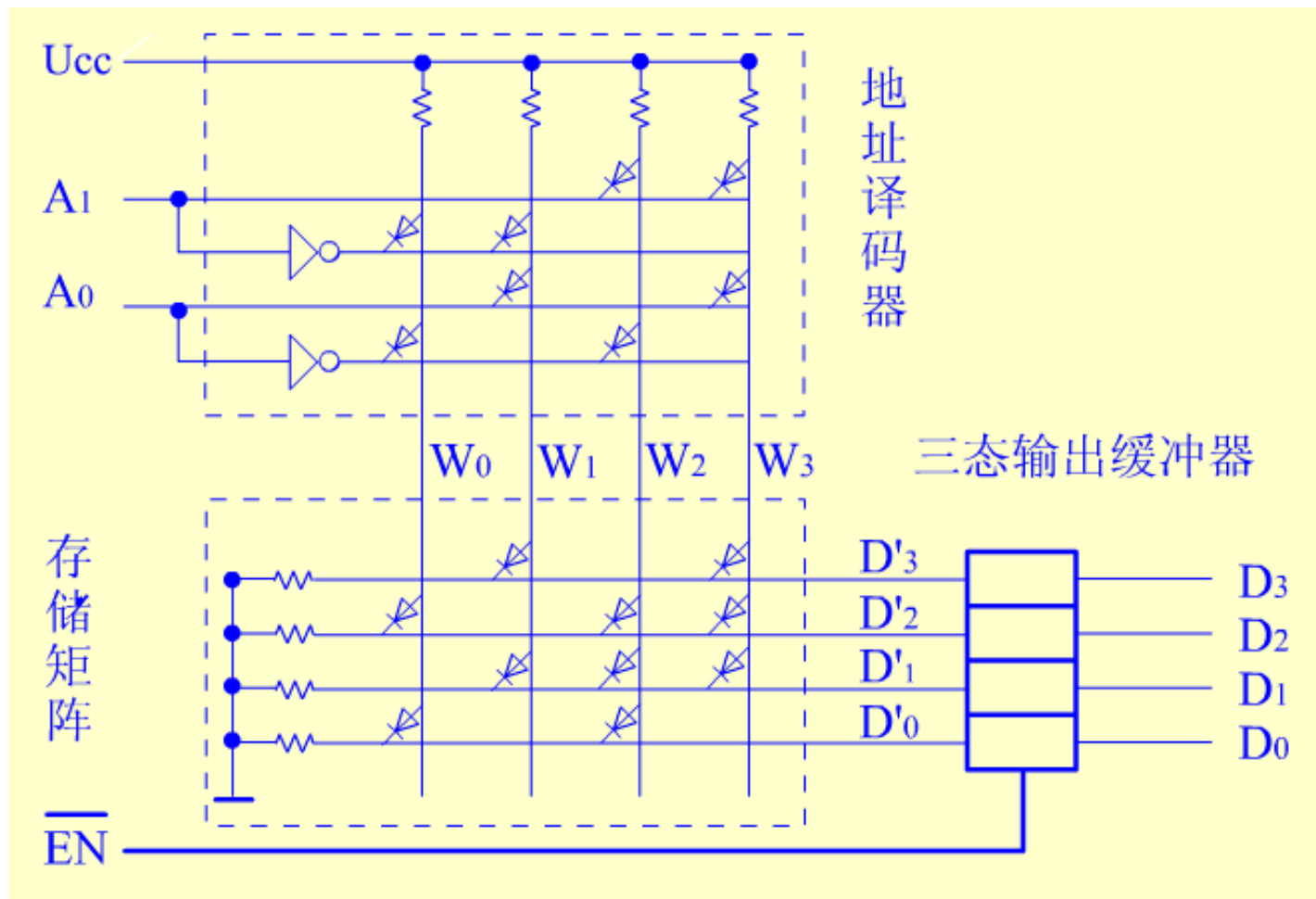
- 定义：
 - 组合型：是指电路具有组合电路的特性。
 - PLD: Programmable Logic Device, 可编程逻辑器件
 - PLA: Programmable Logic Array;
 - PAL: Programmable Array Logic;
 - GAL: General Array Logic;
 - CPLD: Complex Programmable Logic Device
- 认识：
 - 可编程逻辑器件是数字电子设计的一次革命。
 - 改变了传统的电子器件设计的流程，为数字系统的发展奠定了坚实的基础。



补：Programmable? W & H



- ROM的工作原理





补：Programmable? W & H



- ROM的编程

ROM 真值表

A_1	A_0	D_3	D_2	D_1	D_0
0	0	0	1	0	1
0	1	1	0	1	0
1	0	0	1	1	1
1	1	1	1	1	0

- 如果把ROM看作组合逻辑电路，则地址码A1A0是输入变量，数据码D3---D0是输出变量。



补：Programmable? W & H



- ROM的编程

- 由上图可得输出函数表达式：

- $D3 = A1'A0 + A1A0$

- $D2 = A1'A0' + A1A0' + A1A0$

- $D1 = A1'A0 + A1A0' + A1A0$

- $D0 = A1'A0' + A1A0'$

- 逻辑函数是与或表达式，每一条字线对应输入变量的一个最小项。

现在的问题是如何在数据区写上合适的值，
以实现相应的逻辑功能？



补：Programmable? W & H



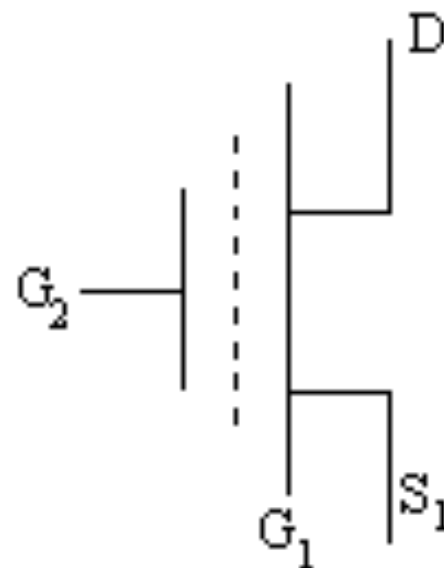
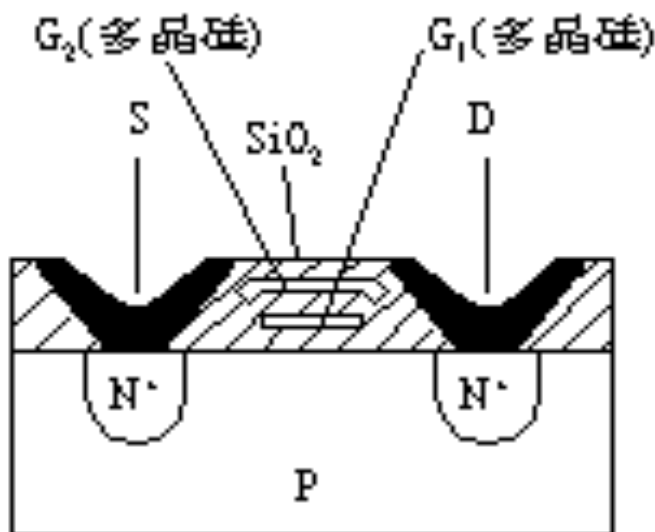
- 存储元记忆原理
 - EPROM采用浮栅雪崩注入型MOS管，其结构与NMOS管相似。
 - 但它有两个栅极：**G1**栅称为浮栅（无引出线），**G2**栅为控制栅。
 - 漏极**D**加上几十伏脉冲电压时，使沟道中电场增强，产生很多高能电子。
 - 此时若在**G2**栅加正电压，可使沟道中的电子穿过氧化层注入到**G1**栅，使**G1**栅变负。由于**G1**栅被绝缘的氧化层包围，泄露电流很小，一旦电子注入到**G1**栅就能长期保存。



补：Programmable? W & H



- 存储0:
 - G1栅有电子积累（变负）时，该MOS管开启电压很高，G2栅加高电平后MOS管仍截止，相当于存0。
- 存储1:
 - G1栅无电子积累时，该MOS管开启电压很低，G2栅加高电平后MOS管导通，相当于存1。



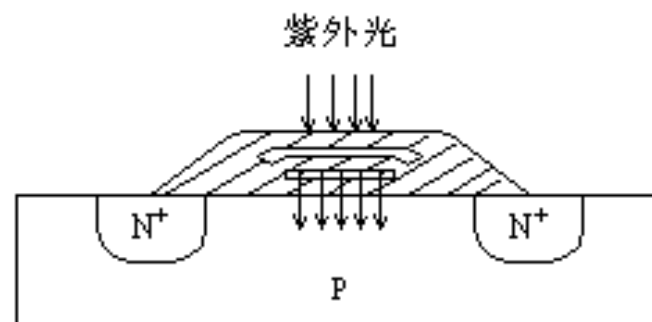
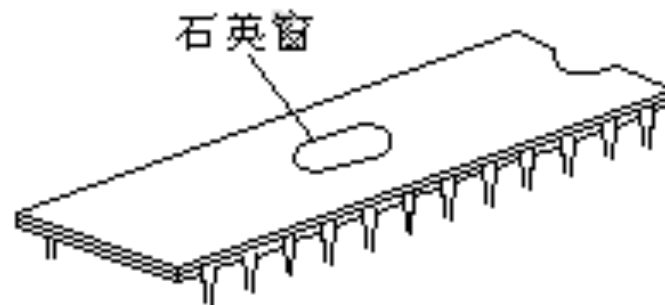


补：Programmable? W & H



● 擦除方法

- 这种器件的上方有一个石英窗口，见图所示。当用光子能量较高的紫外线照射G1浮栅时，G1中的电子获得足够的能量，从而穿过氧化层回到衬底，见图所示。
- 这样，浮栅上的电子消失，达到擦去存储信息的目的，相当于存储器又存了全1。





补：Programmable? W & H



- 其它器件
 - 电擦可编程只读存储器EEPROM，也写成E²PROM。
 - 电改写可编程只读存储器EAROM。
- 应用：
 - 由于ROM的地址译码器输出是全部输入变量的最小项，每一位数据的输出是这些最小项之和，因此任何形式的组合逻辑函数均能通过向ROM写入数据来实现。



补：Programmable? W & H



- 【例】用ROM实现4位二进制码到格雷码的转换
 - 列出二进制码到格雷码的转换真值表

二进制码	格雷码
0000	0000
0001	0001
0010	0011
0011	0010
0100	0110
0101	0111
0110	0101
0111	0100
1000	1100
1001	1101
1010	1111
1011	1110
1100	1010
1101	1011
1110	1001
1111	1000



补：Programmable? W & H



- 【例】用ROM实现4位二进制码到格雷码的转换

- 由真值表写出最小项表达式

$$G3 = \sum(8, 9, 10, 11, 12, 13, 14, 15)$$

$$G2 = \sum(4, 5, 6, 7, 8, 9, 10, 11)$$

$$G1 = \sum(2, 3, 4, 5, 10, 11, 12, 13)$$

$$G0 = \sum(1, 2, 5, 6, 9, 10, 13, 14)$$

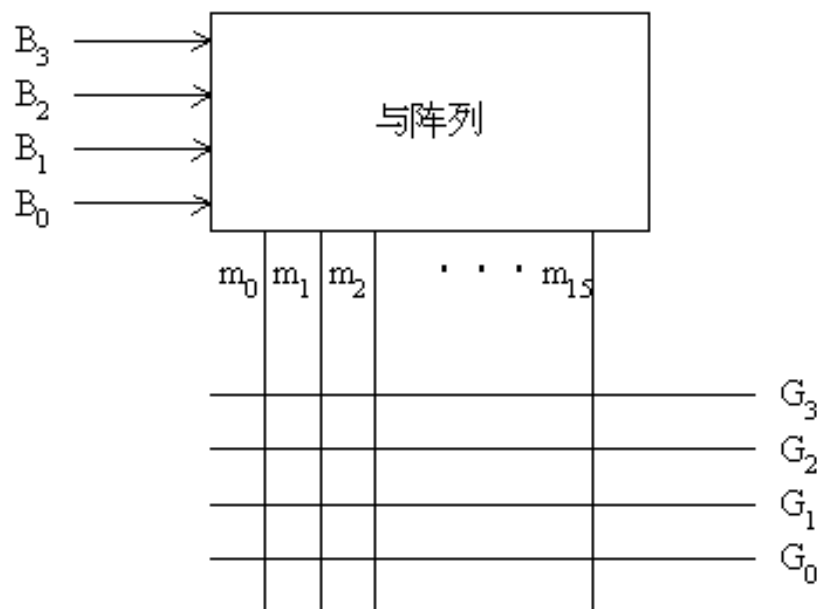
其中输入量为二进制码B3---B0组成的最小项，
输出量G3---G0为格雷码数据输出。



补：Programmable? W & H

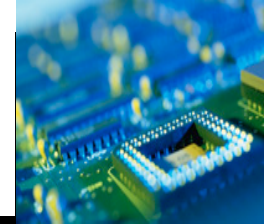


- 【例】用ROM实现4位二进制码到格雷码的转换
 - 根据最小项表达式，画出ROM阵列图
 - 其中与阵列不编程，或阵列编程。所以只画出或阵列编程图。

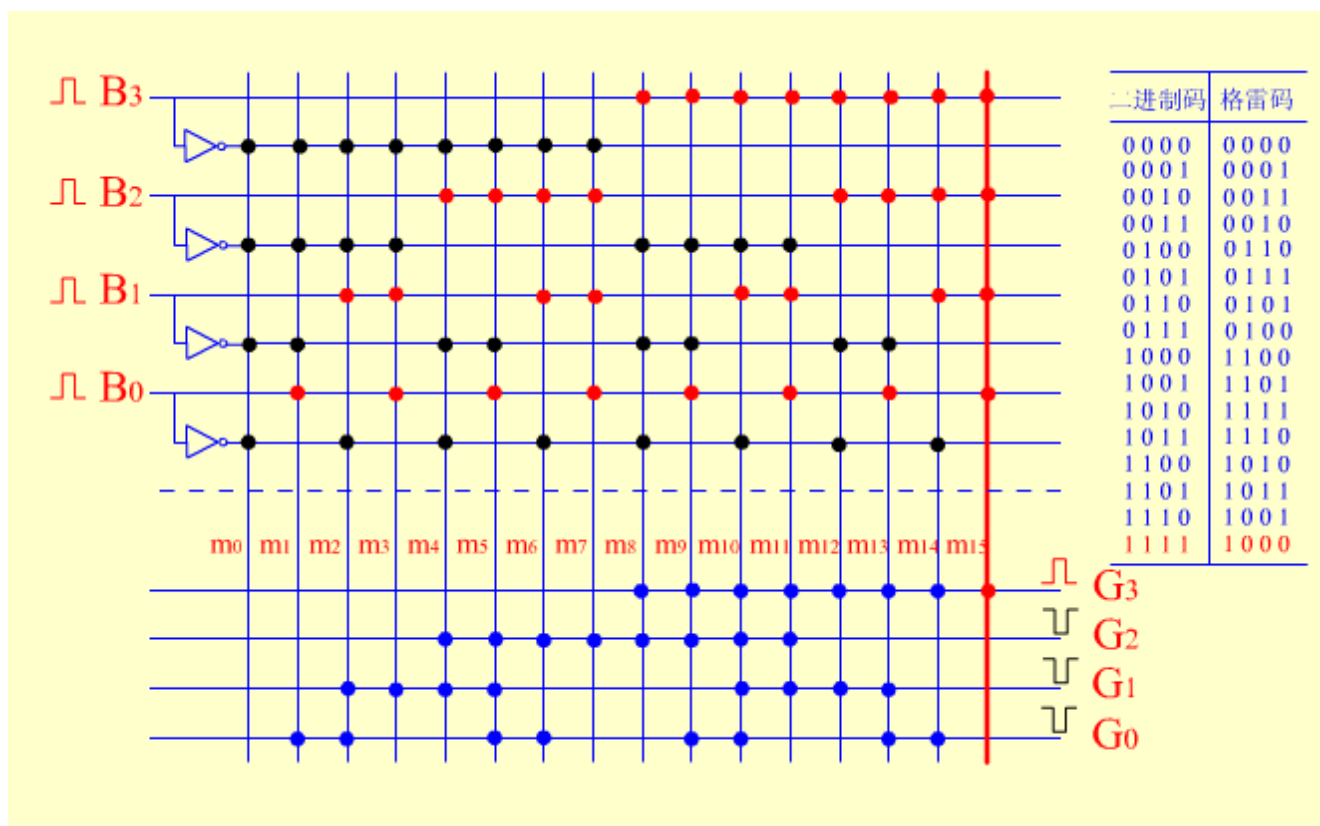




补：Programmable? W & H



- 【例】用ROM实现4位二进制码到格雷码的转换



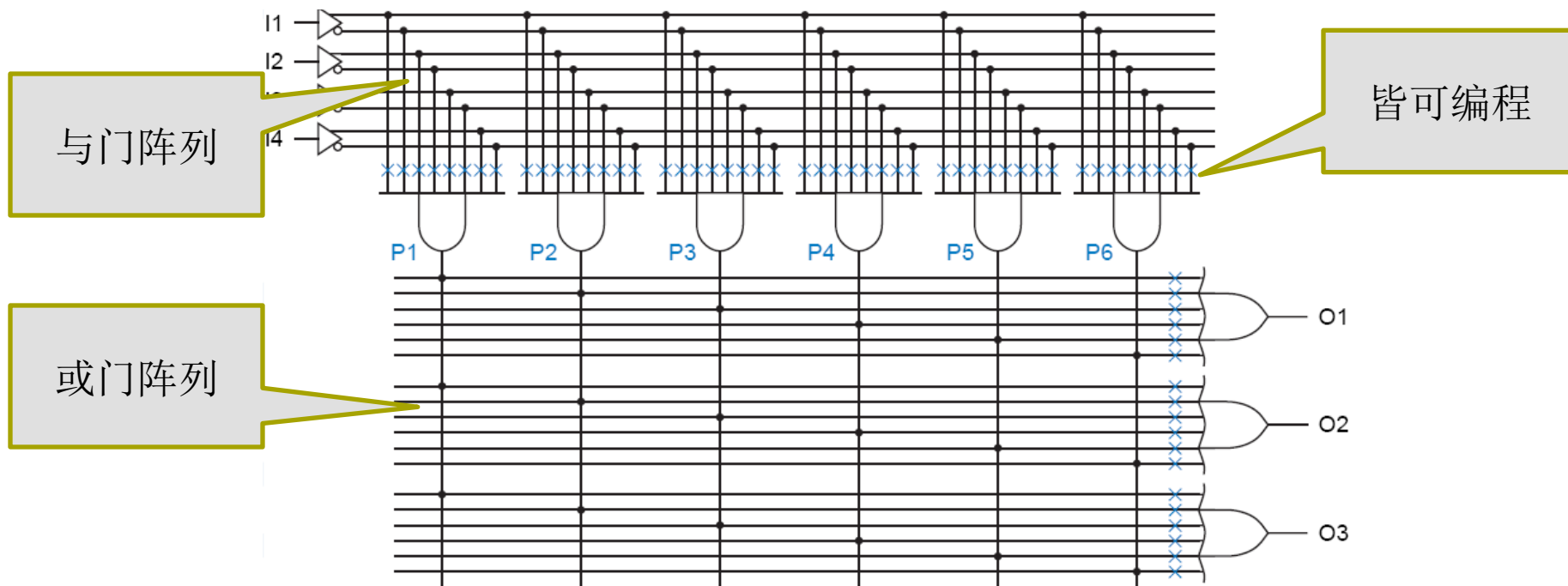


3、组合型PLD



- PLA 可编程逻辑阵列

- PLA是组合的，二级“与-或”器件，对其编程可以实现任何“积之和”逻辑表达式。
- 仅受器件尺寸限制：输入、输出、乘积项的数目。





3、组合型PLD



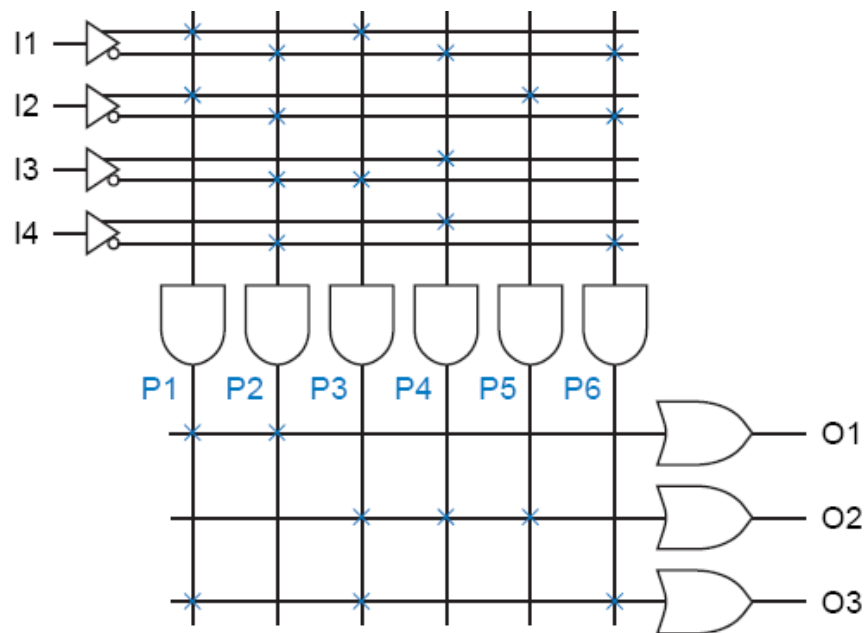
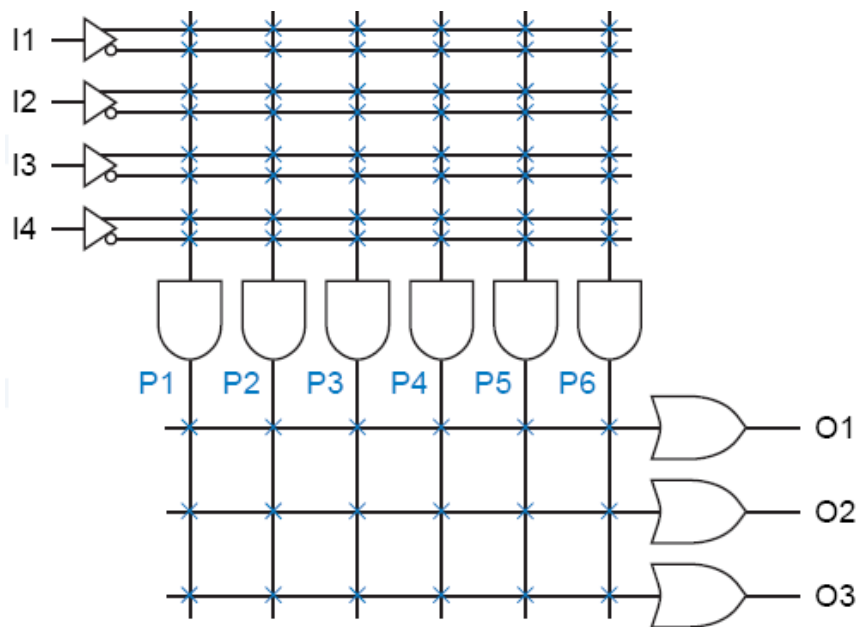
- PLA 可编程逻辑阵列

- 紧凑的表示方式:

$$O1 = I1 \cdot I2 + I1' \cdot I2' \cdot I3' \cdot I4'$$

$$O2 = I1 \cdot I3' + I1' \cdot I3 \cdot I4 + I2$$

$$O3 = I1 \cdot I2 + I1 \cdot I3' + I1' \cdot I2' \cdot I4'$$



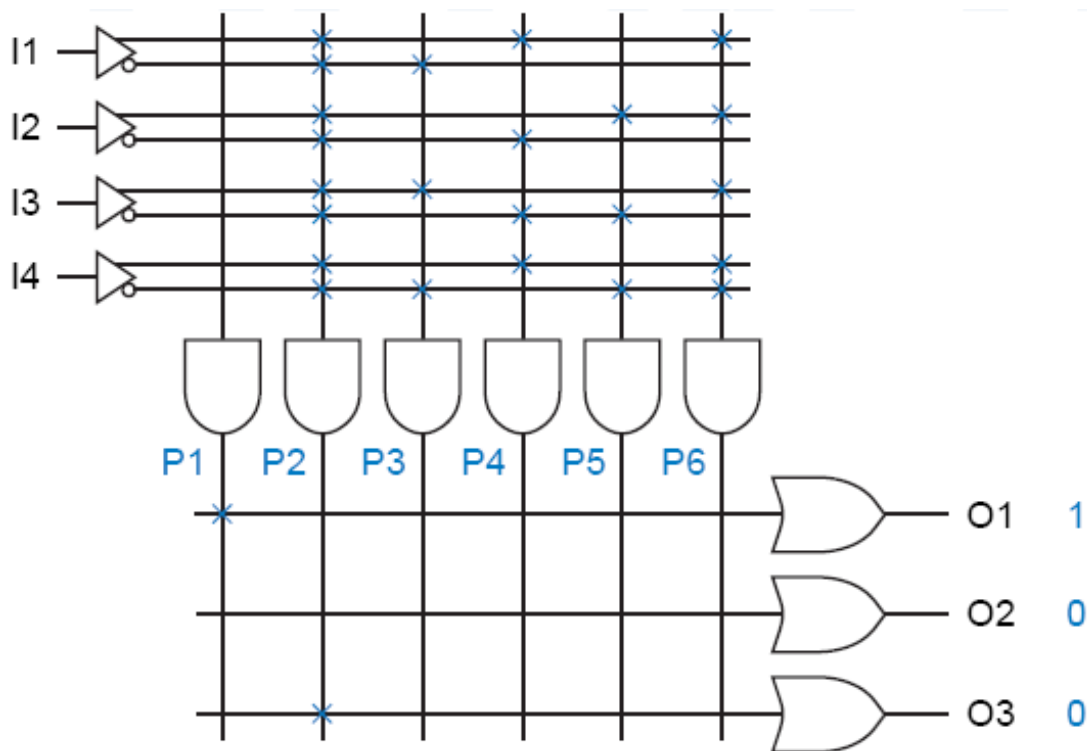


3、组合型PLD



- PLA 可编程逻辑阵列

- 实现常数输出：

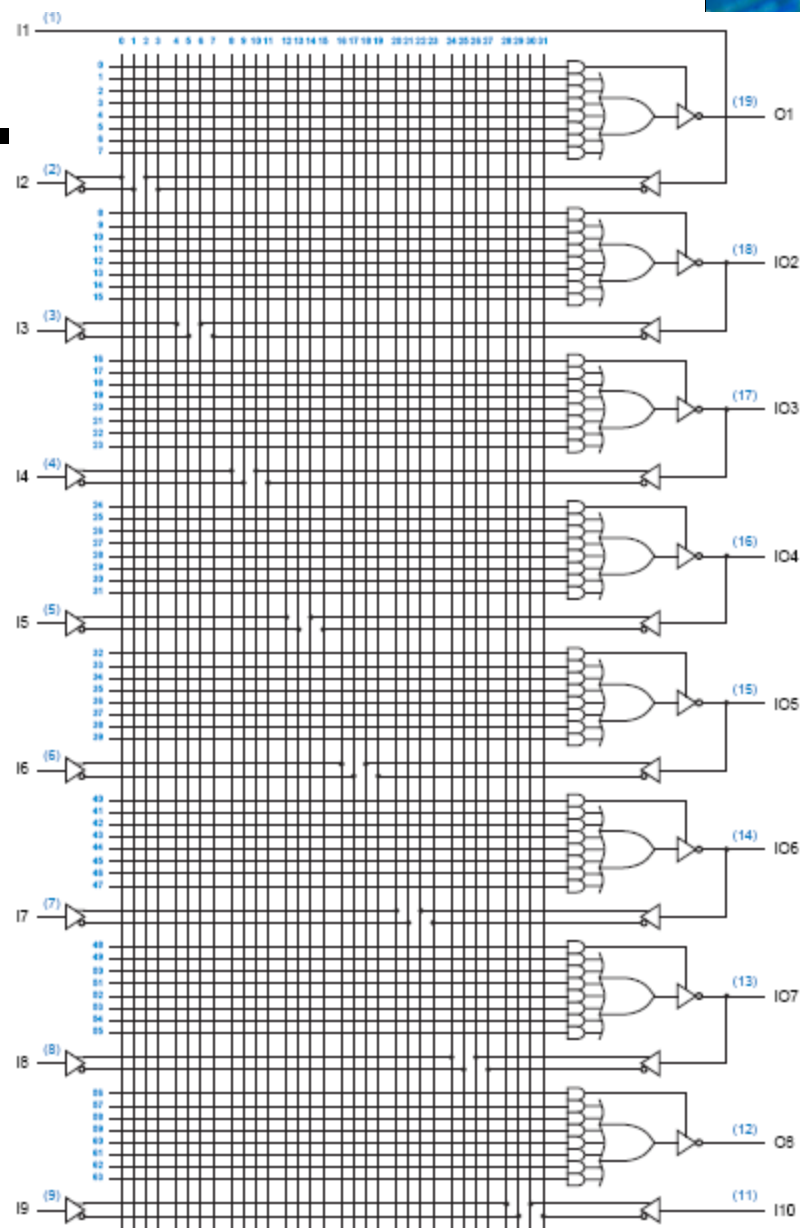
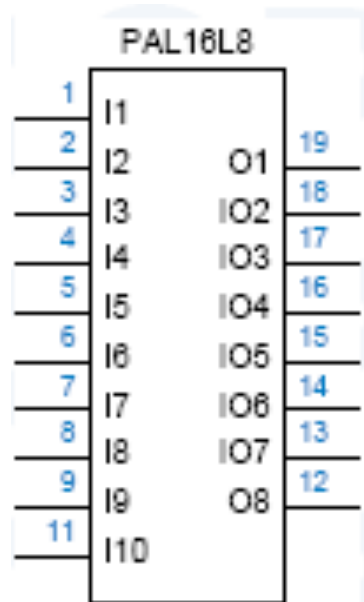




3、组合型PLD

- PAL 可编程阵列逻辑
 - 或门阵列固定;
 - 与门阵列可编程;
 - 双向输入/输出引脚。

PAL16L8





3、组合型PLD



- PAL16L8的特点：
- 有固定的或阵列，7个与门相连；
- 每个输出都有单独的三态门使能信号，使能信号由一个乘积项控制：
- 或门输出有反相器；
- 输出引脚中有6个可以用作输入。
 - 可以设为专用输入；
 - 可以设为专用输出，输出信号可以反馈进入电路，扩展电路处理能力。
 - 可编程输出控制。

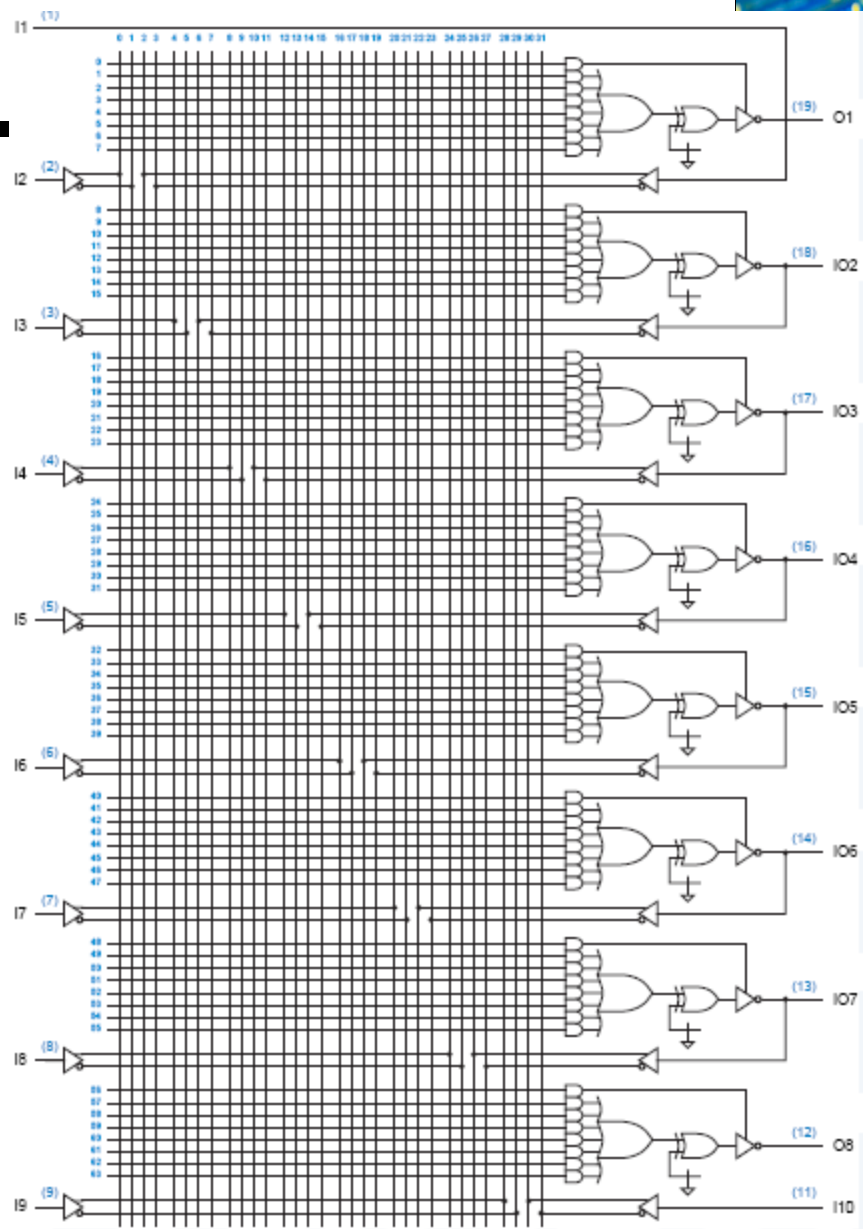
PAL20L8功能类似。PAL是AMD公司的注册商标。



3、组合型PLD

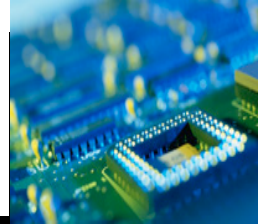
- GAL通用阵列逻辑:
- GAL16V8是一个时序逻辑器件，但可以编程设置为组合逻辑器件。此时，其结构和PAL16L8类似。
- 多了一个输出极性控制的异或门。

GAL是Lattice Semiconductor的注册商标。





3、组合型PLD



- CPLD复杂可编程逻辑器件：
- 是将多个PLD集成到单个芯片上，并提供可编程的互连和输入/输出结构。
- 现在一般把所有超过某一集成度（如1000门以上）的PLD器件都称为CPLD。
- CPLD由可编程逻辑的功能块围绕一个可编程互连矩阵构成。
- 由固定长度的金属线实现逻辑单元之间的互连，并增加了I/O控制模块的数量和功能。
- 可以把CPLD的基本结构看成由可编程逻辑阵列（LAB）、可编程I/O控制模块和可编程内部连线（PIA）等三部分组成。

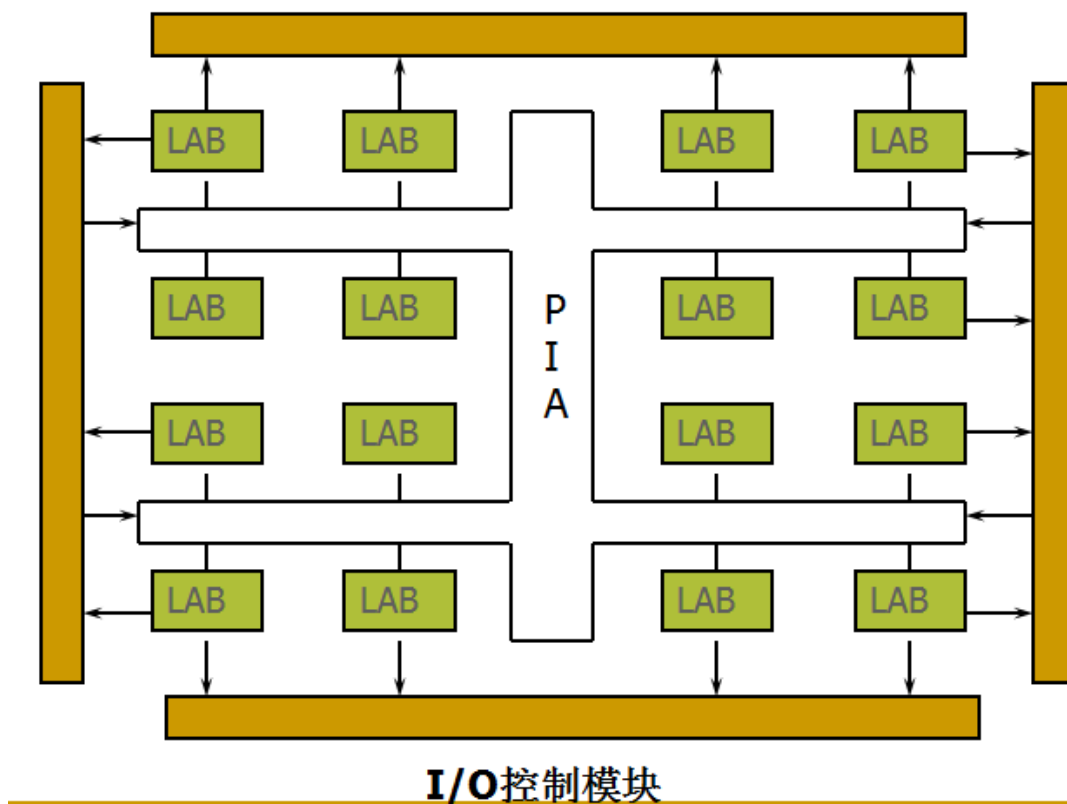


3、组合型PLD



- CPLD复杂可编程逻辑器件：

MAX7123的结构





3、组合型PLD



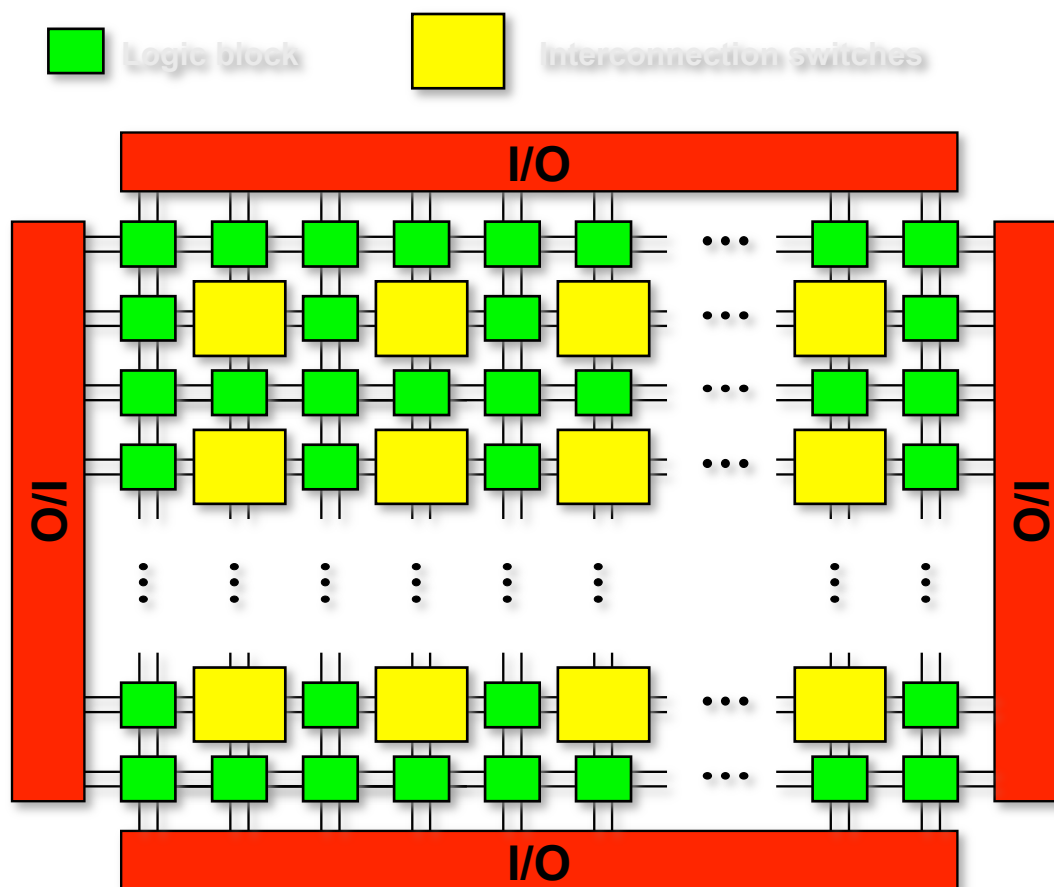
- CPLD复杂可编程逻辑器件：
 - 复杂的逻辑功能，可以使用综合工具在CPLD上实现。
 - 综合工具（又称整合器fitter）要将逻辑功能划分成一个个PLD构件，然后再找到含有足够的PLD、乘积项、内部链接和外部IO引脚的最小可能器件。
 - 设计者可以制定约束以保证综合的效果。
- 器件的编程与测试
 - 在编程模式下，通过编程引脚将设计内容送到芯片内部。
 - 在系统编程：运行状态编程。
 - 编程信息保护。



3、PLD



- FPGA: Field Programmable Gate Array:



3、PLD

• DE2-70

