

§3. 组合逻辑电路

逻辑电路 $\left\{ \begin{array}{l} \text{组合逻辑电路} \\ \text{时序逻辑电路} \end{array} \right.$

1. 组合逻辑电路的分析

(1) 组合逻辑电路的特点

结构特征

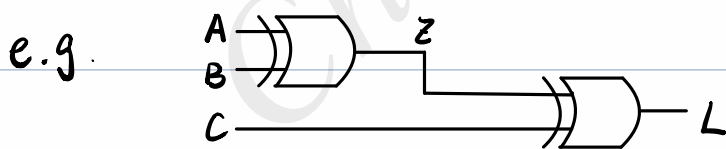
- ① 输入、输出间无反馈延迟通路
- ② 电路中不含记忆单元。

工作特征

电路的输出状态只取决于同一时刻的输入状态

(2) 分析方法

- ① 由逻辑图写出各输出端的逻辑表达式。
- ② 化简和变换
- ③ 列出真值表
- ④ 分析确定电路功能 (文字表述)



① 逻辑表达式

$$L = Z \oplus C = A \oplus B \oplus C$$

② 写真值表

③ 逻辑功能

奇数个1, $L = 1$

偶数个1, $L = 0$

“奇校验电路”

A	B	C	$Z = A \oplus B$	$L = (A \oplus B \oplus C)$
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	0
1	0	0	1	1
1	0	1	1	0
1	1	0	0	0
1	1	1	0	1

2 组合逻辑电路的设计

1) 设计过程

- ① 逻辑抽象
- ② 列真值表
- ③ 写逻辑表达式
- ④ 简化
- ⑤ 画逻辑图

2) 优化实现 用指定芯片实现逻辑电路

单输出电路

相同输入端的与非门比与门或者或门所用晶体管少

多输出电路

考虑“共享乘积项”

多级逻辑电路

- ① 提取公因子
- ② 函数分解

3 组合逻辑电路中的竞争冒险

1) 产生原因

信号通过逻辑门有延迟时间

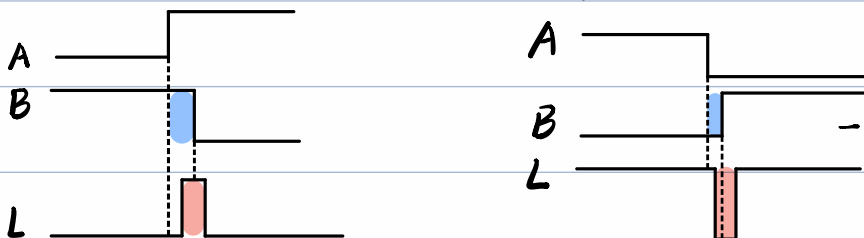


不考虑门的延迟时间, 且 $B = \bar{A}$

$$L = AB = 0$$

$$L = A + B = 1$$

考虑门的延迟时间, 且用非门实现 $B = \bar{A}$



竞争: 当一个逻辑门的两个输入端的信号同时向相反方向变化, 而变化的时间有差异的现象.

冒险: 两个输入端的信号取值的变化方向相反时, 由竞争而可能产生输出干扰脉冲的现象.

(2) 消除方法

① 发现并消除互补项

$$L = (A+B)(\bar{A}+C)$$

当 $B=C=0$ 时, $L = A\bar{A}$ 可能出現竞争冒险

变换 L 为 $L = AC + \bar{A}B + BC$

② 增加乘积项避免互补项相加

$$L = AC + BC$$

当 $A=B=1$ 时, $L = C + \bar{C}$ 可能出現竞争冒险

L	BC	00	01	11	10
A					
0					1
1			1	1	1

$$L = AC + BC + AB$$

③ 输出端并联电容

4 编码器

编码: 赋予二进制码含义的过程

编码器 { 普通编码器: 任何时刻只允许输入一个有效的编码信号
优先编码器: 允许多个输入信号同时有效, 只对优先级高的输入进行编码.

4线 — 2线普通编码器

输入				输出	
I_0	I_1	I_2	I_3	Y_1	Y_0

1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

4线 — 2线优先编码器

输入				输出	
I_0	I_1	I_2	I_3	Y_1	Y_0
1	0	0	0	0	0
x	1	0	0	0	1
x	x	1	0	1	0
x	x	x	1	1	1

典型编码器电路

8线 — 3线优先编码器 CD4532

输入信号高电平有效，输出为二进制代码。

I_0	E_0	EI	GS	E_0	状态
I_1	GS	0	0	0	不工作
I_2	CD4532	1	0	1	工作, 但无输入
I_3	Y_0	1	1	0	工作, 且有输入
I_4	Y_1	1	1	1	不可能出现
I_5	Y_2				
I_6					
I_7					
EI					

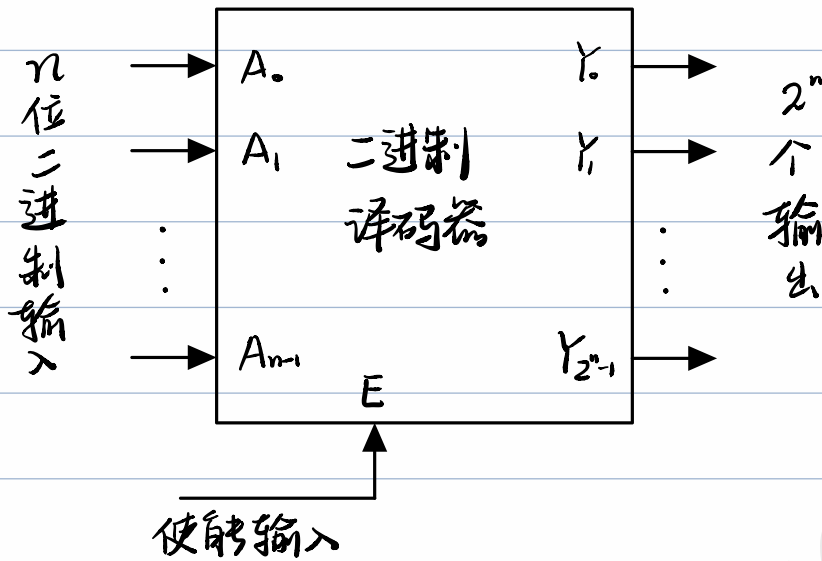
输入									输出				
EI	I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	Y_2	Y_1	Y_0	GS	E_0
0	x	x	x	x	x	x	x	x	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	1
1	1	x	x	x	x	x	x	x	1	1	1	1	0
1	0	1	x	x	x	x	x	x	1	1	0	1	0
1	0	0	1	x	x	x	x	x	1	0	1	1	0
1	0	0	0	1	x	x	x	x	1	0	0	1	0
1	0	0	0	0	1	x	x	x	0	1	1	1	0
1	0	0	0	0	0	1	x	x	0	1	0	1	0
1	0	0	0	0	0	0	1	x	0	0	1	1	0
1	0	0	0	0	0	0	0	1	0	0	0	1	0

5 译码器

二进制代码 $\xrightarrow{\text{译码}}$ 高低电平信号 / 其他代码

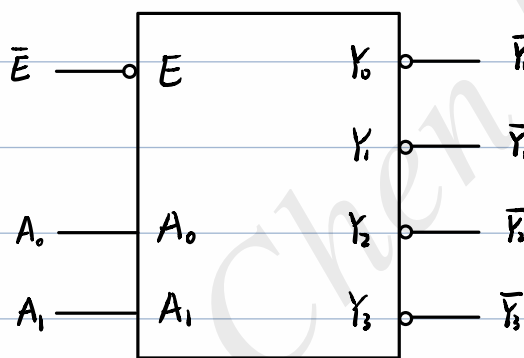
1) 二进制译码器

“多输入” “多输出”



n 个输入
 2^n 个输出
 1 个使能输入端

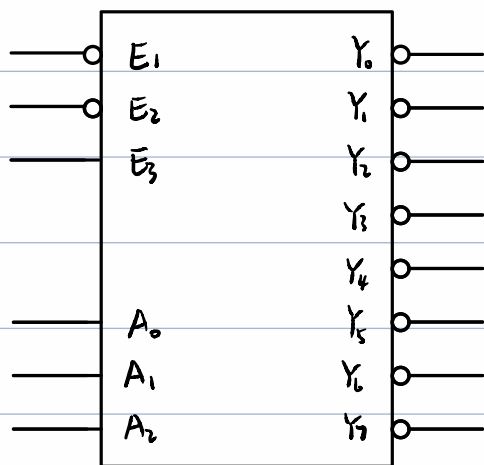
2线-4线译码器



$$\begin{aligned} \overline{Y_0} &= \overline{E \cdot \overline{A_1} \cdot \overline{A_0}} = \overline{E \cdot m_0} \\ \overline{Y_1} &= \overline{E \cdot \overline{A_1} \cdot A_0} = \overline{E \cdot m_1} \\ \overline{Y_2} &= \overline{E \cdot A_1 \cdot \overline{A_0}} = \overline{E \cdot m_2} \\ \overline{Y_3} &= \overline{E \cdot A_1 \cdot A_0} = \overline{E \cdot m_3} \end{aligned}$$

输入			输出			
\overline{E}	A_1	A_0	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

3线 — 8线译码器



$$\overline{Y_0} = \overline{E \overline{A_2} \overline{A_1} \overline{A_0}}$$

$$\overline{Y_1} = \overline{E \overline{A_2} \overline{A_1} A_0}$$

$$\overline{Y_2} = \overline{E \overline{A_2} A_1 \overline{A_0}}$$

$$\overline{Y_3} = \overline{E \overline{A_2} A_1 A_0}$$

...

输入

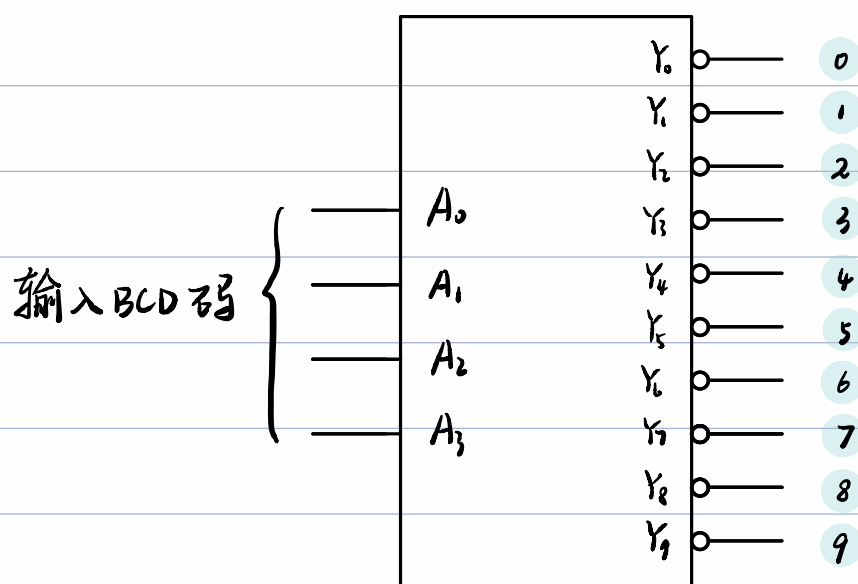
输出

E_3	$\overline{E_2}$	$\overline{E_1}$	A_2	A_1	A_0	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$	$\overline{Y_4}$	$\overline{Y_5}$	$\overline{Y_6}$	$\overline{Y_7}$
X	1	X	X	X	X	1	1	1	1	1	1	1	1
X	X	1	X	X	X	1	1	1	1	1	1	1	1
0	X	X	X	X	X	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	0	0	1	1	0	1	1	1	1	1	1
1	0	0	0	1	0	1	1	0	1	1	1	1	1
1	0	0	0	1	1	1	1	1	0	1	1	1	1
1	0	0	1	0	0	1	1	1	1	0	1	1	1
1	0	0	1	0	1	1	1	1	1	1	0	1	1
1	0	0	1	1	0	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	1	1	1	1	0

(2) 二—十进制译码器

功能：将 8421 BCD 码译成 10 个状态输出

⇒ 4 个输入端，10 个输出端，输出低电平有效
 当输出超过 8421 码的范围时 (1010 ~ 1111)
 输出均为高电平，此时没有有效译码输出。



BCD 输入

输出

A_3	A_2	A_1	A_0	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$	$\overline{Y_4}$	$\overline{Y_5}$	$\overline{Y_6}$	$\overline{Y_7}$	$\overline{Y_8}$	$\overline{Y_9}$
0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1	1	1	1
0	0	1	1	1	1	1	0	1	1	1	1	1	1
0	1	0	0	1	1	1	1	0	1	1	1	1	1
0	1	0	1	1	1	1	1	1	0	1	1	1	1
0	1	1	0	1	1	1	1	1	1	0	1	1	1
1	0	1	1	1	1	1	1	1	1	1	0	1	1
1	0	0	0	1	1	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	1	1	1	1	0

e.g. 若二-十进制译码器输出为 111101111

可以表示为16进制 (3EF)_H

(3) 七段显示译码器

数字显示电路

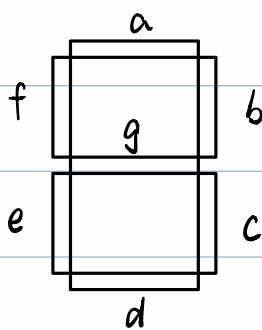
译码驱动器

数码显示器

七段数字显示器

液晶显示器

七段数码管



发光二极管的连接

- 共阴极显示器
- 共阳极显示器

显示译码器 / 七段译码器

功能：将输入的4位BCD码翻译成“七段码”
送给数码管。

- 分类
- 输出高电平有效：用来驱动共阴极显示器
 - 输出低电平有效：用来驱动共阳极显示器

e.g. 74HC4511 输出高电平有效

(4) 数字显示电路

显示电路

- 静态显示电路：一个译码器与一个显示器相连接
- 动态扫描显示电路：一个译码器与多个显示器相

↓ (视觉暂留) 连接, 再增加显示控制电路.

注意 ① f_c 太高 \Rightarrow 看不清显示的数码

f_c 太低 \Rightarrow 看到4个数字轮流显示

② 通常 $25\text{Hz} < f_c < 100\text{Hz}$

6 数据分配器与数据选择器

(1) 数据分配器 (相当于单刀多掷开关)

数据分配 将公共数据线上的数据根据需要送到不同的通道上去。

实现 ① 3线-8线译码器 74H138

改变 A_0, A_1, A_2 的值

将数据分配到不同

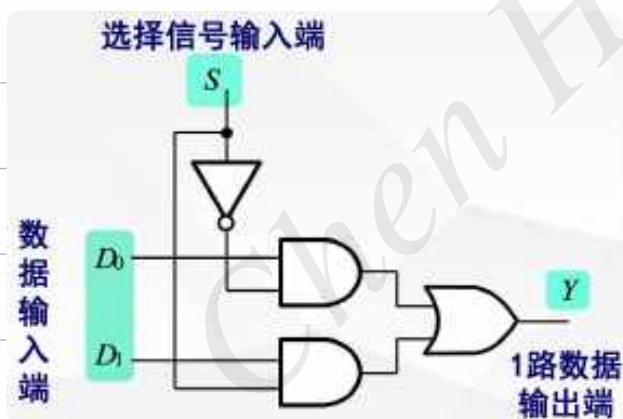
输出端。

② 三态门

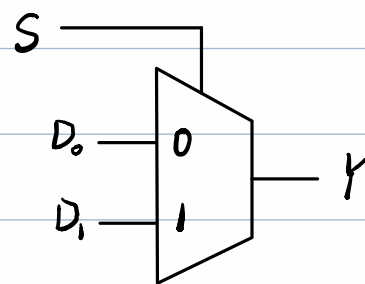
(2) 数据选择器

从多个数据源中选择一路送到公共线路上

① 2选1数据选择器



逻辑符号



逻辑表达式 $Y = \bar{S}D_0 + SD_1$

② 4选1数据选择器

$$Y = D_0 \bar{S}_1 \bar{S}_0 + D_1 \bar{S}_1 S_0 + D_2 S_1 \bar{S}_0 + D_3 S_1 S_0$$

③ 8选1数据选择器

2个4选1 + 1个2选1

数据选择器的应用

△ 变量数 = 选通端数,

用数据选择器实现函数的步骤:

- ① 将函数变换为最小项表达式
- ② 让数据选择器处于使能状态, 并将函数的输入变量与地址信号 S_2, S_1, S_0 相连接
- ③ 将数据输入 $D_0 \sim D_7$ 接相应逻辑电平, 即在逻辑表达式中的 m_i , 则相应 $D_i = 1$, 其余为 0.

7 数值比较器与加法运算电路

1) 数值比较器

一位数值比较器

真值表

A	B	$F_{A>B}$	$F_{A<B}$	$F_{A=B}$
0	0	0	0	1
0	1	0	1	0
1	0	1	0	0
1	1	0	0	1

逻辑表达式

$$F_{A>B} = A\bar{B}$$

$$F_{A<B} = \bar{A}B$$

$$F_{A=B} = \bar{A}\bar{B} + AB = \overline{A\bar{B} + \bar{A}B}$$

二位数值比较器

A_1 B_1	A_0 B_0	$F_{A>B}$	$F_{A<B}$	$F_{A=B}$
$A_1 > B_1$	x	1	0	0
$A_1 < B_1$	x	0	1	0
$A_1 = B_1$	$A_0 > B_0$	1	0	0
$A_1 = B_1$	$A_0 < B_0$	0	1	0
$A_1 = B_1$	$A_0 = B_0$	0	0	1

$$F_{A>B} = A_1 \bar{B}_1 + (A_1 B_1 + \bar{A}_1 \bar{B}_1) A_0 \bar{B}_0$$

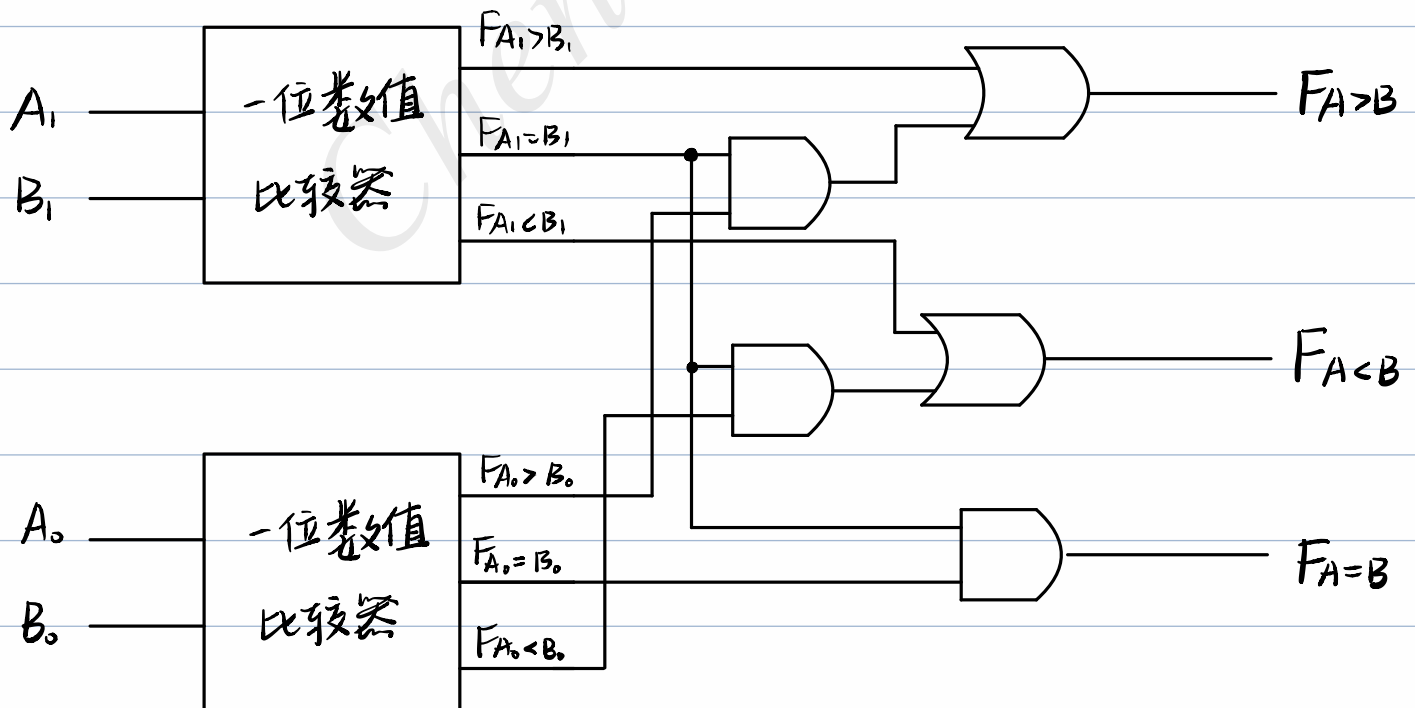
$$= F_{A_1>B_1} + F_{A_1=B_1} \cdot F_{A_0>B_0}$$

$$F_{A<B} = \bar{A}_1 B_1 + (A_1 B_1 + \bar{A}_1 \bar{B}_1) \bar{A}_0 B_0$$

$$= F_{A_1<B_1} + F_{A_1=B_1} \cdot F_{A_0<B_0}$$

$$F_{A=B} = (A_1 B_1 + \bar{A}_1 \bar{B}_1) (A_0 B_0 + \bar{A}_0 \bar{B}_0)$$

$$= F_{A_1=B_1} \cdot F_{A_0=B_0}$$



(2) 典型数值比较器

四位数值比较器 74HC85

数值比较器的位数扩展

{ 串联
{ 并联 (1块)

(3) 加法器

{ 半加器
{ 全加器

两个1位二进制数相加时, 不考虑低位的进位.

考虑...

半加器

A	B	进位 C	和 S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$$S = \bar{A}B + A\bar{B} = A \oplus B$$

$$C = AB$$

全加器

A	B	<u>C_i</u>	S	C _o
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$S = \bar{A}\bar{B}C_1 + \bar{A}B\bar{C}_1 + A\bar{B}\bar{C}_1 + ABC_1$$

$$= A \oplus B \oplus C$$

$$C_0 = AB + \bar{A}\bar{B}C_1 + \bar{A}BC_1$$

$$= AB + (A \oplus B)C_1$$

(4) 多位数加法器

串行进位加法器：逻辑电路简单

超前进位加法器：运算速度快

$$\begin{array}{r} \text{e.g.} \quad A_3 \quad A_2 \quad A_1 \quad A_0 \\ + \quad B_3 \quad B_2 \quad B_1 \quad B_0 \\ \hline C_3 \quad S_3 \quad S_2 \quad S_1 \quad S_0 \end{array}$$

$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

$$C_i = A_i B_i + (A_i \oplus B_i) C_{i-1}$$

$$\text{设 } G_i = A_i B_i \quad P_i = A_i \oplus B_i$$

$$S_i = P_i \oplus C_{i-1}$$

$$C_i = G_i + P_i C_{i-1}$$

当 $A_i = B_i = 1$ 时, $G_i = 1, P_i = 0 \Rightarrow C_i = 1$

C_i 产生变量

当 $A_i = 0, B_i = 1$ 或 $A_i = 1, B_i = 0$ 时, $P_i = 1, G_i = 0$

$\Rightarrow C_i = C_{i-1}$ P_i 传输变量

8 组合可编程逻辑器件

PLD: Programmable Logic Device

(1) 性能特点

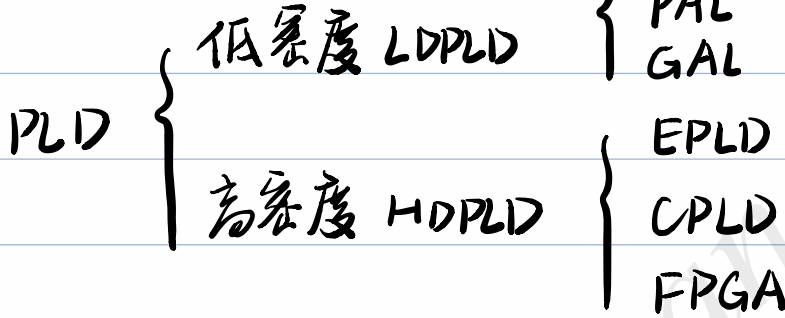
① 逻辑功能强

② 集成度高

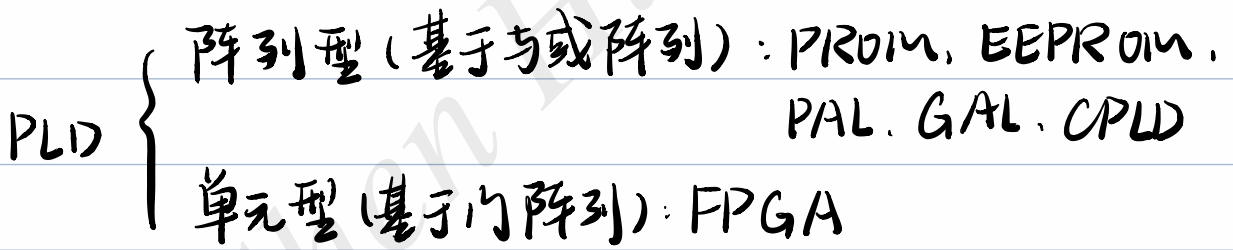
- ③ 设计方法灵活
- ④ 具有完善先进的开发工具
- ⑤ 系统处理速度快
- ⑥ 系统具有加密功能
- ⑦ 使用方便

(v) 分类

① 按密度分



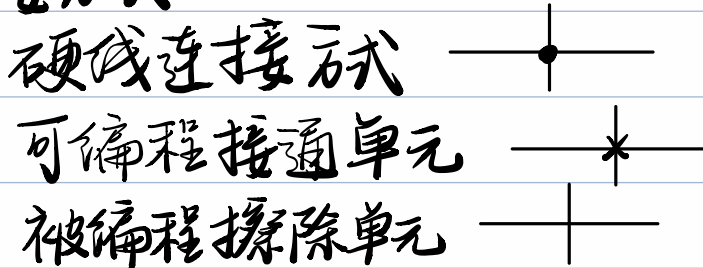
② 按结构特点分



(3) PLD 的逻辑符号表示方法

门阵列的每个交叉点都是一个“单元”

连接方式



(4) 组合逻辑电路的 PLD 实现