

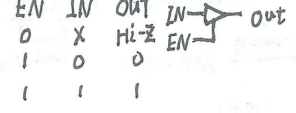
2.6 two-level Optimization: Map Manipulation
 多级电路可降低 gate input cost (相较两级电路 SOP 或 POS)
 步骤: (1) 对函数进行转化 (2) 计算 cost

- (1) 因式分解 (factoring) (4) 代入 (substitution)
- (2) 函数分解 (decomposition) (5) 消降 (elimination)
- (3) 提取 (extraction)

文字成本: 对布尔表达式中的文字进行计数
 门输入成本: (G) ① 全部文字数 ② 除单文字项外的全部项数

(4N) + ④ 不同取值值的单文字数 \bar{A} 与 \bar{B} 为不同取值
 \bar{A} 与 \bar{A} 为相同取值

2.7 other gate type



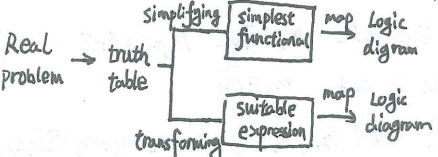
Chap 3 Combinational Logic design

3.1 Combinational circuits

combinational Logic Circuits: 只与当前输入有关, 与过去输入无关
 Sequential Logic Circuits

- 分析法: 1. 写出各门的逻辑表达式 3. 简化表达式
- 2. 逐层代入, 写出整体式子 4. 画出新电路图

- 设计步骤: 1. 对需求逻辑抽象 (如分配器译码) 3. 化简, 并用化简结果画圈 (k-map)
- 2. 写出真值表/初始表达式 (minitem) 4. mapping, 并仿真测试



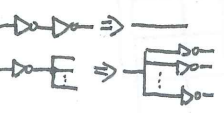
3.2 Hierarchical design

反复分解一个大函数直至 block 都是够简单. 足够简单的 block 叫 primitive block.
 这些 block 的集合则叫 hierarchy. 分解成可重复利用的块

3.3 Technology Mapping

NAND Mapping Algorithm:

- 1. 将 AND 与 OR 替换成 NAND
- 2. 执行下列两种操作直到 电路输入与非门输出



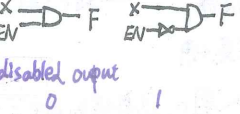
3.4 functions and functional blocks

SSI, MSI, LSI, VLSI

3.5 Rudimentary logic function

① $F = (F_3, F_2, F_1, F_0)$ is a bus.

② enabling function



3.5 Commonly Logic Function Design

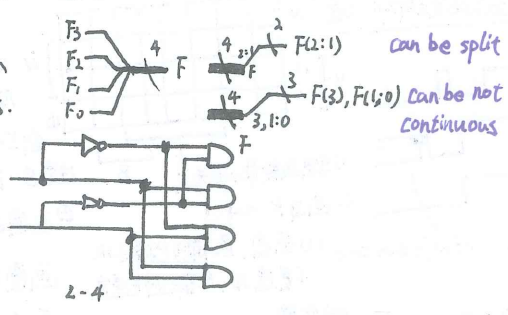
① decoder (Translate an coding into another coding)

n-to-m decoder ($m \leq 2^n$)

构造 3-8 译码器可用 1 个 1-2 译码器与 1 个 2-4 译码器
 而 2-4 译码器可用 2 个 1-2 译码器构造

② Encoder

Instruction encoder 输入: 10 bits, 每位对应 0-9
 Priority Encoder 输出: 4 bits 对应 BCD 码
 decimal-to-B-CD

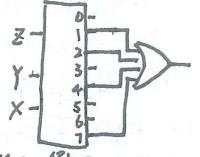
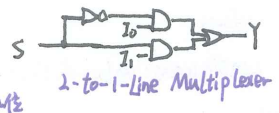


⑤ Multiplexers

Logic circuits that perform selecting 从 input 中选择一路并作为 output.

n 位控制 input 1 个 output Y
 2^n 信息 input

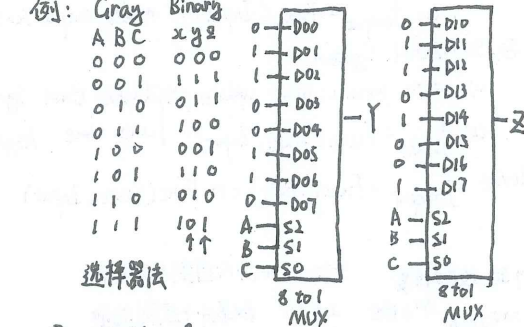
Multiplexer 可进行宽度扩展
 原先每个信息 input 都宽为 1, 现宽可为 m 位



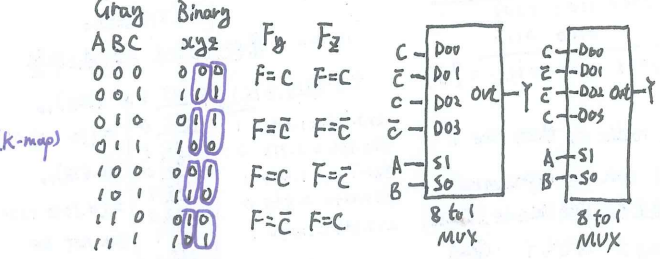
3.6 Implementing Methods of Combinational Functions

① Decoder and OR Gates

找出 output 对应的最小项组合. 将 Decoder 对应的输出作为 or 门输入 例: $S(x, y, z) = \sum m(1, 2, 4, 7)$
 先将 1, 2, 4, 7 对应的值 OR



② m-wide 2^n -to-1-line multiplexer + inverter



3.7 Programming Technologies

Inputs → Fixed AND array (decoder) → Programmable OR array → Output

(a) Programmable read-only memory (PROM)

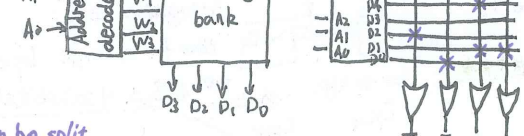
Inputs → Programmable Connections → Programmable AND array → Fixed OR array → Output

(b) Programmable array Logic (PAL) device

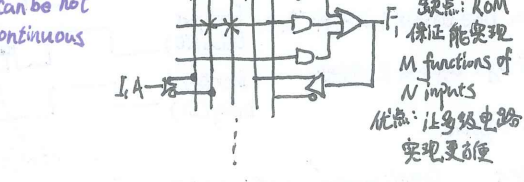
Inputs → Programmable Connections → Programmable AND array → Programmable OR array → Output

(c) Programmable Logic array (PLA) device

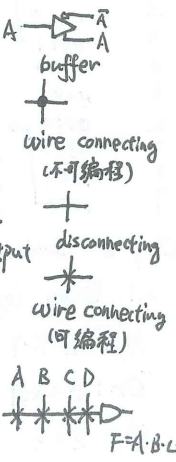
ROM 就是一个 multiple-output 真值表. 将 input 视为数据地址



③ PAL



Logical symbol



$F = A \cdot B \cdot C \cdot D$