

Unit 13

—Programmable Logic Devices

张彦航

School of Computer Science
Zhangyanhang@hit.edu.cn

ROM及其应用

- PLD的背景知识
- ROM及其应用

ROM及其应用

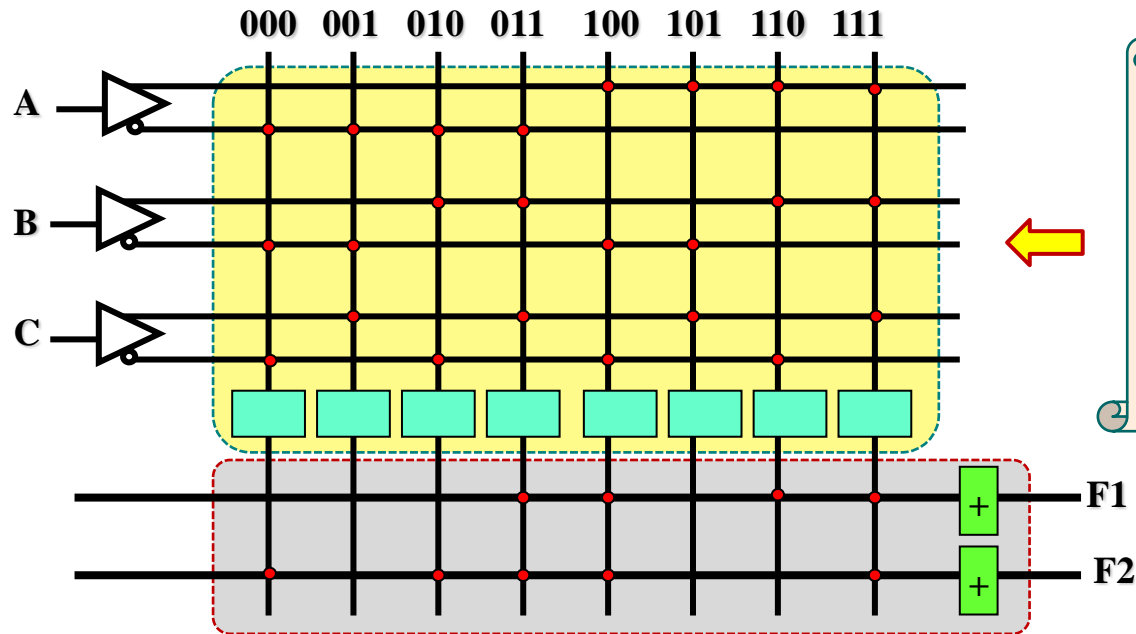
Applications——1. 设计组合逻辑电路

$$F1 = A'BC + AB'C' + ABC' + ABC, \quad F2 = A'B'C' + A'BC' + A'BC + AB'C' + ABC$$

$$F1 = \Sigma(3, 4, 6, 7), \quad F2 = \Sigma(0, 2, 3, 4, 7)$$

利用ROM设计

组合逻辑函数的通用方法



1. 将函数表达式描述成最小项之和 Σm_i 的形式
2. 画出全译码的与阵
3. 若 Σ 中包含 m_i , 则在或阵中, 在输出线与第 m_i 条字线的交点处画一个 **交点** 处画一个 **·**。

ROM及其应用

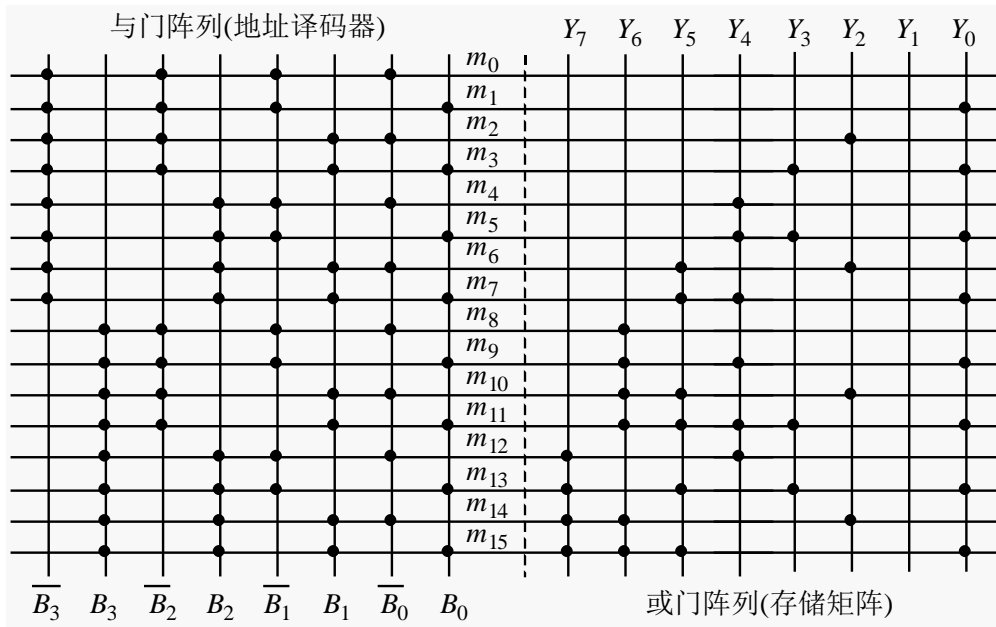
Applications——2. 设计函数运算表

例：设计实现函数运算 $y = x^2$ ， x 是4位二进制数

$$x = B_3 B_2 B_1 B_0 : 0 \sim 15$$

$$y = x^2 : 0 \sim 15^2$$

$$y : Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0$$



实现函数运算 $y = x^2$ 的真值表

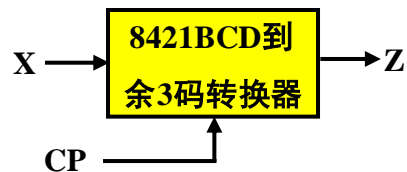
输 入				输 出								注
B_3	B_2	B_1	B_0	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0	十进制数
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	1	1
0	0	1	0	0	0	0	0	0	1	0	0	4
0	0	1	1	0	0	0	0	1	0	0	1	9
0	1	0	0	0	0	0	1	0	0	0	0	16
0	1	0	1	0	0	0	1	1	0	0	1	25
0	1	1	0	0	0	1	0	0	1	0	1	36
0	1	1	1	0	0	1	1	0	0	0	1	49
1	0	0	0	0	1	0	0	0	0	0	0	64
1	0	0	1	0	1	0	1	0	0	0	1	81
1	0	1	0	0	1	1	0	0	1	0	0	100
1	0	1	1	0	1	1	1	1	0	0	1	121
1	1	0	0	1	0	0	1	0	0	0	0	144
1	1	0	1	1	0	1	0	1	0	0	1	169
1	1	1	0	1	1	0	0	0	1	0	0	196
1	1	1	1	1	1	1	0	0	0	0	1	225

ROM及其应用

Applications——3. 设计一个同步时序的码制转换器，将串行输入的8421BCD码转换为串行输出的余3码。

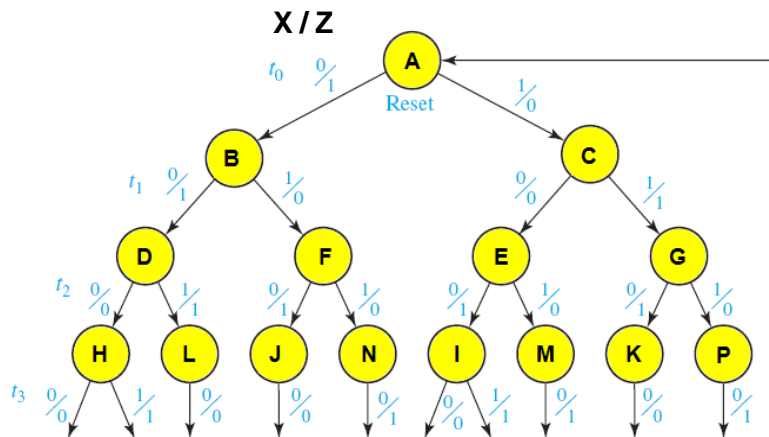
t_0 时刻: 看 $X \rightarrow Z(0 \rightarrow 1)$
 t_1 时刻: 看 $t_1 t_0$ 时刻的 $X \rightarrow Z$
 t_2 时刻: 看 $t_2 t_1 t_0$ 时刻的 $X \rightarrow Z$
 t_3 时刻: 看 $t_3 t_2 t_1 t_0$ 时刻的 $X \rightarrow Z$

- 转换器的输入和输出都是最低位优先



X Input (BCD)				Z Output (excess-3)			
t_3	t_2	t_1	t_0	t_3	t_2	t_1	t_0
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

① State graph and state table



ROM及其应用

② Reduction of State Table

Time	Present State	Next State		Present Output (Z)	
		X = 0	1	X = 0	1
t_0	A	B	C	1	0
t_1	B	D	E	1	0
	C	E	E	0	1
t_2	D	H	H	0	1
	E	H	M	1	0
t_3	H	A	A	0	1
	M	A	-	1	-

④ Transition table

(b) Transition table

$Q_1 Q_2 Q_3$	$Q_1^+ Q_2^+ Q_3^+$		Z	
	X = 0	X = 1	X = 0	X = 1
A 0 0 0	001	010	1	0
B 0 0 1	011	100	1	0
C 0 1 0	100	100	0	1
D 0 1 1	101	101	0	1
E 1 0 0	101	110	1	0
H 1 0 1	000	000	0	1
M 1 1 0	000	-	1	-

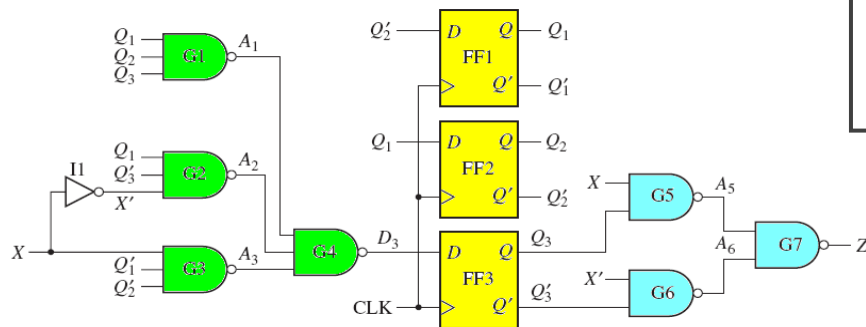
(c) Truth table

X	Q_1	Q_2	Q_3	Z	D_1	D_2	D_3
0	0	0	0	1	0	0	1
0	0	0	1	1	0	1	1
0	0	1	0	0	1	0	0
0	0	1	1	0	1	0	1
0	1	0	0	1	1	0	1
0	1	0	1	0	0	0	0
0	1	1	0	1	0	0	0
0	1	1	1	x	x	x	x
1	0	0	0	0	0	1	0
1	0	0	1	0	1	0	0
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	1	0	0	0	1	1	0
1	1	0	1	1	0	0	0
1	1	1	0	x	x	x	x
1	1	1	1	x	x	x	x

③ State Assignment

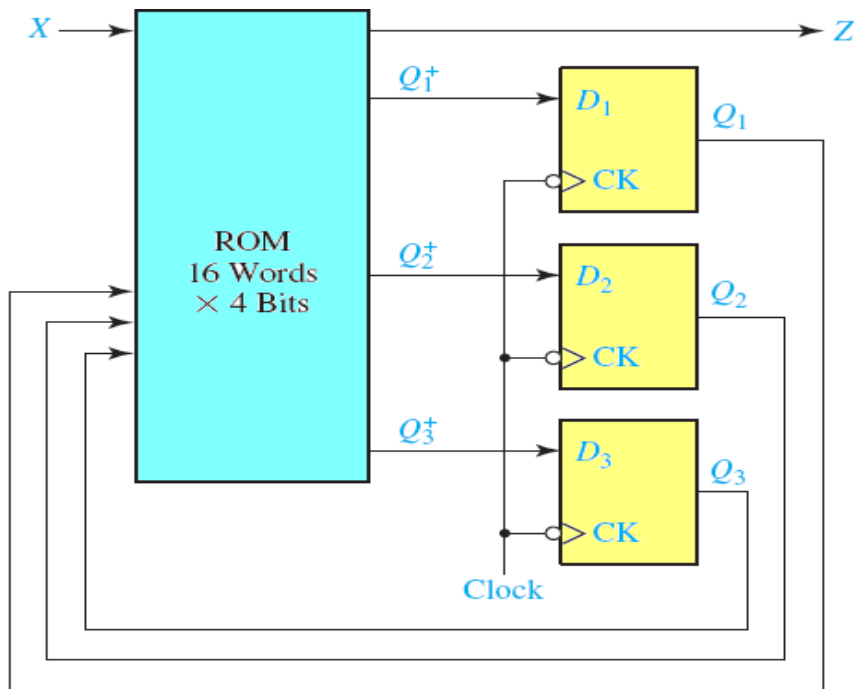
$Q_2 Q_3$ \ Q_1	0	1
00	A	B
01		C
11	H	D
10	M	E

方案1: 逻辑门+D触发器



ROM及其应用

方案2: ROM+D触发器



每一组输入对应ROM的一个存储单元的地址

每一组输出对应ROM的一个存储单元中的存放内容

(c) Truth table

X	Q_1	Q_2	Q_3	Z	D_1	D_2	D_3
0	0	0	0	1	0	0	1
0	0	0	1	1	0	1	1
0	0	1	0	0	1	0	0
0	0	1	1	0	1	0	1
0	1	0	0	1	1	0	1
0	1	0	1	0	0	0	0
0	1	1	0	1	0	0	0
0	1	1	1	x	x	x	x
1	0	0	0	0	0	1	0
1	0	0	1	0	1	0	0
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	1	0	0	0	1	1	0
1	1	0	1	1	0	0	0
1	1	1	0	x	x	x	x
1	1	1	1	x	x	x	x