

数字逻辑设计

张春慨

School of Computer Science

ckzhang@hit.edu.cn

用触发器设计同步时序逻辑—实例

- 序列发生器
- 模8可逆计数器
- 自动售卖机
- 时序锁
- 二进制串行加法器
- 串行输入的8421BCD码检测器
- 奇偶校验器
- 更复杂的同步时序逻辑设计

序列信号发生器的设计

序列信号发生器：能循环产生一组特定的串行数字序列信号的电路。

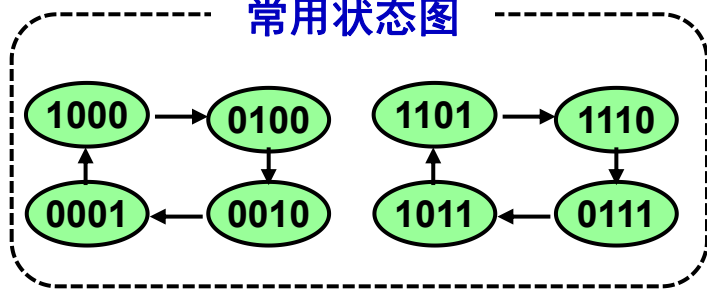
序列的长度：序列信号的位数。如：序列为00011，则序列长度为5。

序列状态之间很有规律

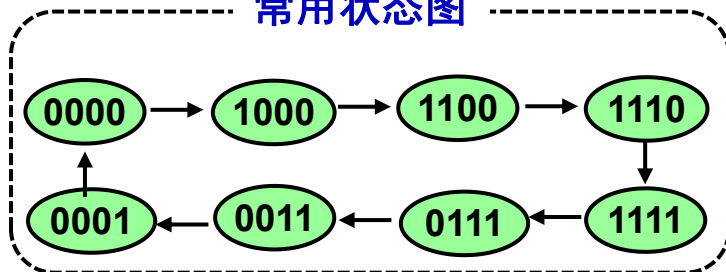
特殊类型

- 使用环形计数器设计
- 使用扭环计数器设计

常用状态图



常用状态图



序列信号发生器的设计-1

任意类型

- 使用**D触发器**设计
- 使用**计数器** + **数据选择器**设计;
- 用**移位寄存器** + **反馈电路**设计(逻辑门 or 译码器 or 数据选择器)
- 用**计数器** + **PROM**设计

大体思路:

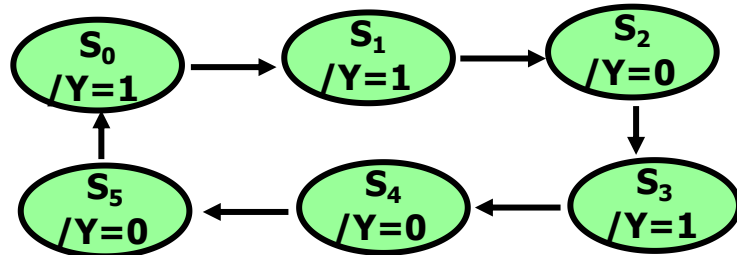
1. 实现序列信号一个周期之内的波形
2. 将此波形循环再现

例1: 用D触发器设计一个 **110100** 序列信号发生器

方法1: 利用D触发器

- 序列信号长度为 L, 则取 L 个不同的状态
- 每个状态下时序电路的输出就是序列信号中的一位。

1. 画状态转换图



时序电路的不同状态对应输出序列中的各位。

序列信号发生器的设计-1

2. 状态编码

S_0 — 000, S_3 — 011
 S_1 — 001, S_4 — 100
 S_2 — 010, S_5 — 101

3. 状态转换真值

$Q_2Q_1Q_0$	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}	Y
0 0 0	0	0	1	1
0 0 1	0	1	0	1
0 1 0	0	1	1	0
0 1 1	1	0	0	1
1 0 0	1	0	1	0
1 0 1	0	0	0	0

4. 卡诺图化简

Q_1Q_0	00	01	11	10
Q_2 0	1	0	0	1
Q_2 1	1	0	X	X

$$D_0 = Q_0'$$

Q_1Q_0	00	01	11	10
Q_2 0	0	0	1	0
Q_2 1	1	0	X	X

$$D_2 = Q_2Q_0' + Q_1Q_0$$

Q_1Q_0	00	01	11	10
Q_2 0	0	1	0	1
Q_2 1	0	0	0	X

$$D_1 = Q_2'Q_1'Q_0 + Q_1Q_0'$$

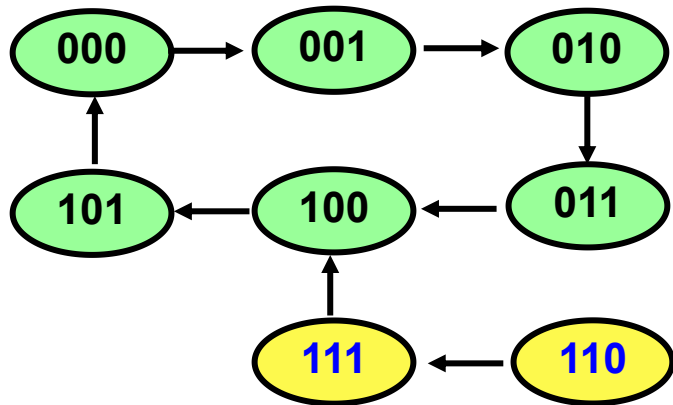
Q_1Q_0	00	01	11	10
Q_2 0	1	1	1	0
Q_2 1	0	0	X	X

$$Y = Q_2'Q_1' + Q_1Q_0$$

序列信号发生器的设计-1

5. 电路实现(略)

6. 检查无关项



时序电路可以
自启动。

方法1：利用D触发器

- 序列信号长度为 L ，则取 L 个不同的状态
- 每个状态下时序电路的输出就是序列信号中的一位。

序列信号发生器的设计-1

例1：设计一个 **110100** 序列信号发生器

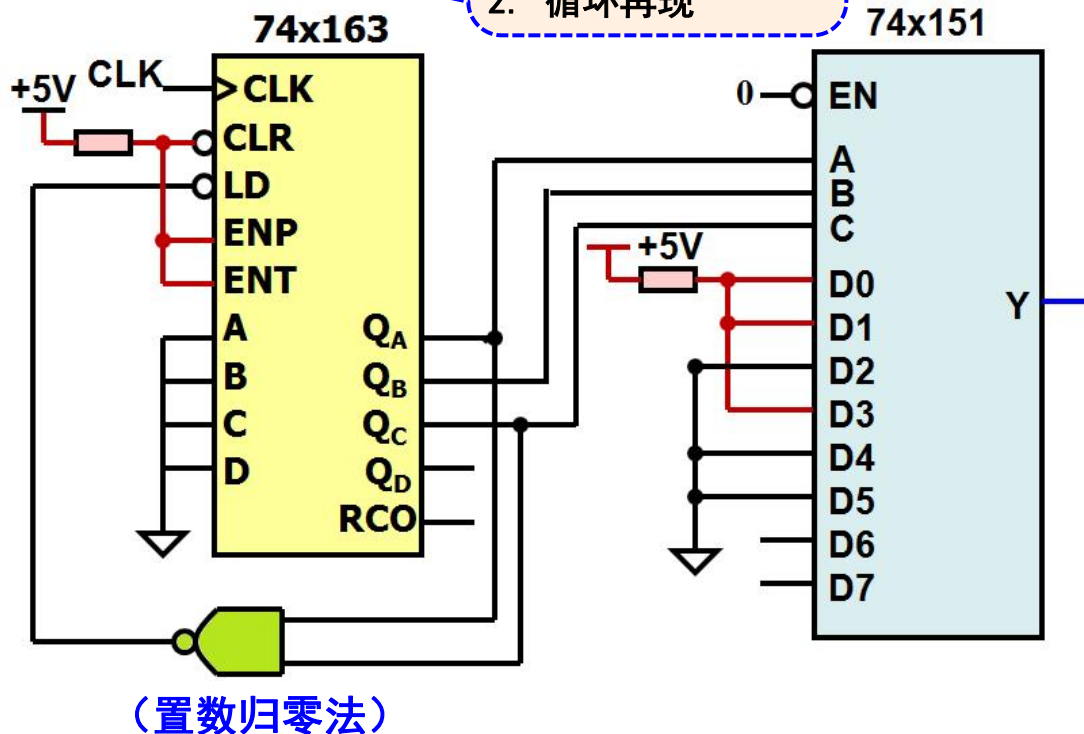
方法2： 利用计数器+数据选择器

大体思路：

1. 实现序列信号一个周期之内的波形
2. 循环再现

计数器+数据选择器
设计序列信号发生器的方法

- ❑ 数据选择器74151的输入 D_0 - D_5 接成110100。
- ❑ 74163接成模6加法计数器
- ❑ 计数器输出连接到数据选择器的选择控制端CBA，经过循环选择产生所需序列。

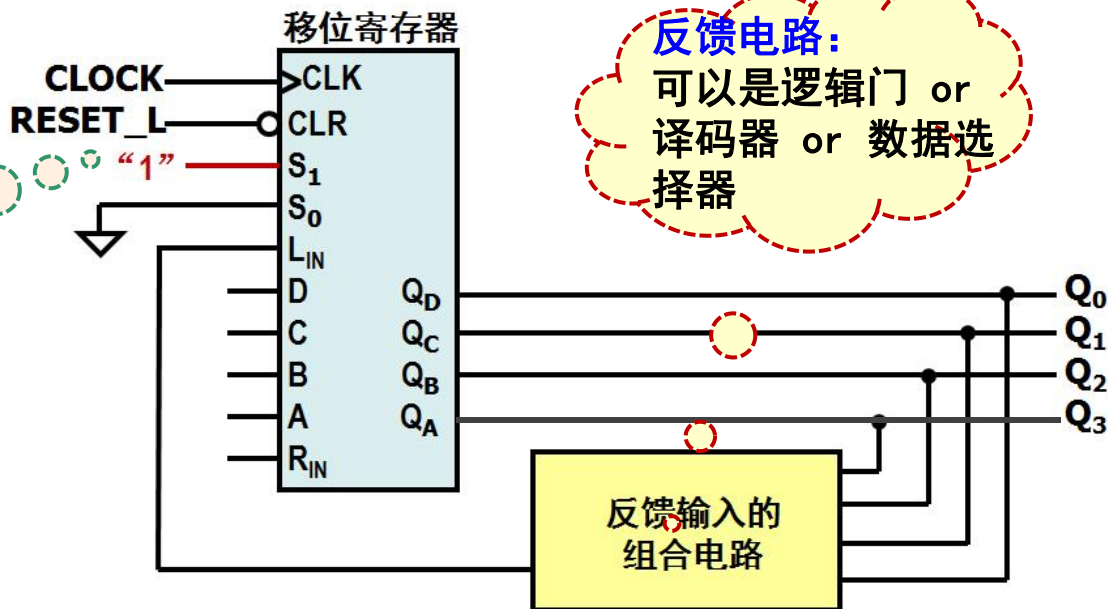


序列信号发生器的设计-2

例2：设计一个 **00010111** 序列信号发生器

方法3： 移位寄存器+反馈电路设计（逻辑门 or 译码器 or 数据选择器）

原理： 寄存器设置为左移模式，利用反馈电路为左移串行输入端提供输入



反馈电路：

可以是逻辑门 or
译码器 or 数据选择器

反馈输入的
组电路

序列信号发生器的设计-2

例2：设计一个 **00010111**序列信号发生器

方法3：移位寄存器+反馈电路设计（逻辑门 or 译码器 or 数据选择器）

具体方法

□ 确定移位寄存器的位数。

序列信号长度为L，则移位寄存器的位数 n 应满足：

$$2^n \geq L$$

试探法： n 为满足条件的最小值，将序列数据循环左移，画状态图。检查状态图中所有L个状态是否两两不同，是，则 n 值可用；否则取 $n+1$ ，重复上述操作。

□ 画状态转换表，确定左移时最低位输入的卡诺图，求出表达式。如果有无关项，检察电路的自启动能力

□ 实现最低位反馈输入（逻辑门 or 译码器 or 数据选择器）

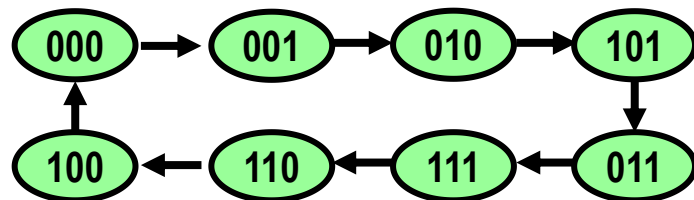
□ 取移位寄存器的某位输出即为所要求的序列信号。

1. 确定移位寄存器位数

序列长度 $L=8$ ，则 $n=3$

2. 状态转换图

00010111 11000



用**74194**的低3位 $Q_B Q_C Q_D$ 输出

序列信号发生器的设计-2

例2：设计一个 00010111序列信号发生器

方法3: 移位寄存器+反馈电路设计

3. 状态转换真值表

Q_B	Q_C	Q_D	L_{IN}
0	0	0	1
0	0	1	0
0	1	0	1
1	0	1	1
0	1	1	1
1	1	1	0
1	1	0	0
1	0	0	0

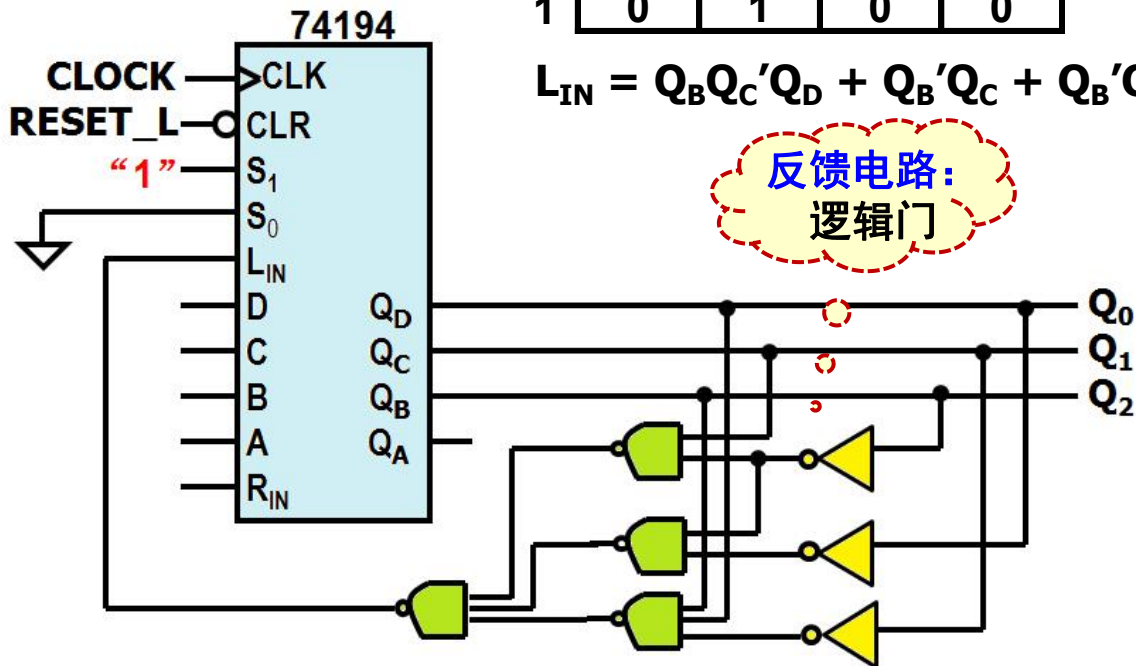
**Q_B输出即为
序列信号**

4.卡诺图化简

		Q_1Q_0			
		00	01	11	10
Q_2	0	1	0	1	1
	1	0	1	0	0

$$\mathbf{L}_{IN} = \mathbf{Q}_B \mathbf{Q}_C' \mathbf{Q}_D + \mathbf{Q}_B' \mathbf{Q}_C + \mathbf{Q}_B' \mathbf{Q}_D'$$

反馈电路： 逻辑门



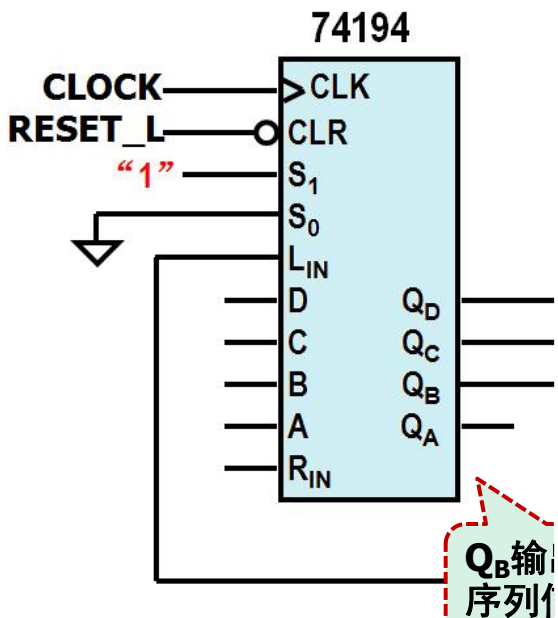
序列信号发生器的设计-2

例2: 设计一个 **00010111**序列信号发生器 **1~4. 同上**

方法3: 移位寄存器+反馈电路设计

$$\begin{aligned} L_{IN} &= Q_B Q_C' Q_D + Q_B' Q_C + Q_B' Q_D' \\ &= \sum m(0, 2, 3, 5) \end{aligned}$$

转换为最小项表达式



$Q_B Q_C Q_D$ 分别接74151的选择控制端CBA, 则:

$$\begin{aligned} D_0 &= D_2 = D_3 = D_5 = 1, \\ D_1 &= D_4 = D_6 = D_7 = 0 \end{aligned}$$

反馈电路:
数据选择器

序列信号发生器的设计-2

例2: 设计一个 00010111序列信号发生器 1~4. 同上

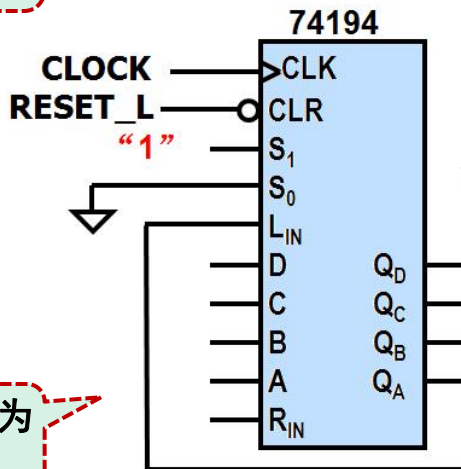
方法3: 移位寄存器+反馈电路设计

$$L_{IN} = Q_B Q_C' Q_D + Q_B' Q_C + Q_B' Q_D'$$
$$= \Sigma m(0, 2, 3, 5)$$

转换为最小项表达式

$Q_B Q_C Q_D$ 分别接74138的地址输入端CBA, 则:

Y_0, Y_2, Y_3, Y_5 分别被译中时, 反馈回1, 否则反馈回0



Q_B 输出即为序列信号

反馈电路:
译码器

序列信号发生器设计方法总结

方法总结

特殊类型

- 使用环形计数器设计
- 使用扭环计数器设计

任意类型

- 使用D触发器设计
- 使用计数器 + 数据选择器设计；
- 用移位寄存器+反馈电路设计（逻辑门 or 译码器 or 数据选择器）
- 用计数器 + PROM设计

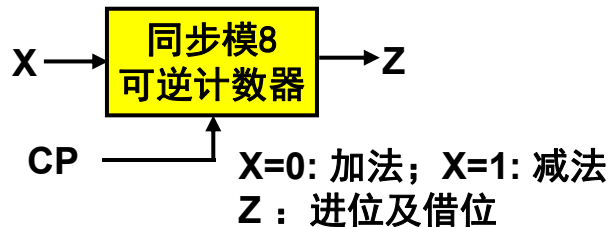
用触发器设计同步时序逻辑—实例

- 序列发生器
- 模8可逆计数器
- 自动售卖机
- 时序锁
- 二进制串行加法器
- 串行输入的8421BCD码检测器
- 奇偶校验器
- 更复杂的同步时序逻辑设计

利用触发器设计同步时序逻辑_例1

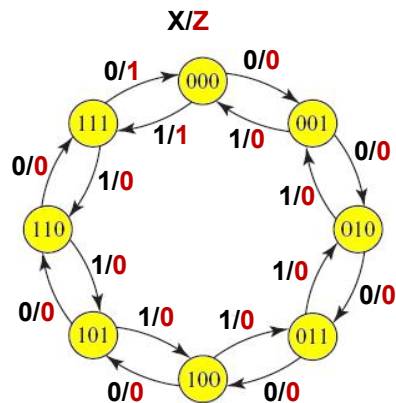
例1：利用T触发器设计一个同步模8可逆计数器

确定 T_3 ：看 $Q_3^n \rightarrow Q_3^{n+1}$
确定 T_2 ：看 $Q_2^n \rightarrow Q_2^{n+1}$
确定 T_1 ：看 $Q_1^n \rightarrow Q_1^{n+1}$



1. 原始状态图及状态表

需要3个T触发器



T触发器驱动表

输入端T	次态 Q_{n+1}
0	Q_n
1	\bar{Q}_n

2. 状态转换真值表

输入 现态				次态			输入 输出			
X	Q_3^n	Q_2^n	Q_1^n	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	T_3	T_2	T_1	Z
0	0	0	0	0	0	1	0	0	1	0
0	0	0	1	0	1	0	0	1	1	0
0	0	1	0	0	1	1	0	0	1	0
0	0	1	1	1	0	0	1	1	1	0
0	1	0	0	1	0	1	0	0	1	0
0	1	0	1	1	1	0	0	1	1	0
0	1	1	0	1	1	1	0	0	1	0
0	1	1	1	0	0	0	1	1	1	1
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	0	0	0	0	0	1	0
1	0	1	0	0	0	1	0	1	1	0
1	0	1	1	0	1	0	0	0	1	0
1	1	0	0	0	1	1	1	1	1	0
1	1	0	1	1	0	0	0	0	1	0
1	1	1	0	1	0	1	0	1	1	0
1	1	1	1	1	1	0	0	0	1	0

利用触发器设计同步时序逻辑_例1

3. 卡诺图化简

$XQ_3^n \backslash Q_2^n Q_1^n$	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	1	0	0	0
10	1	0	0	0

$$T_3 = \overline{X} \overline{Q_2^n} Q_1^n + X \overline{Q_2^n} \overline{Q_1^n}$$

$XQ_3^n \backslash Q_2^n Q_1^n$	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	1	0	0	1
10	1	0	0	1

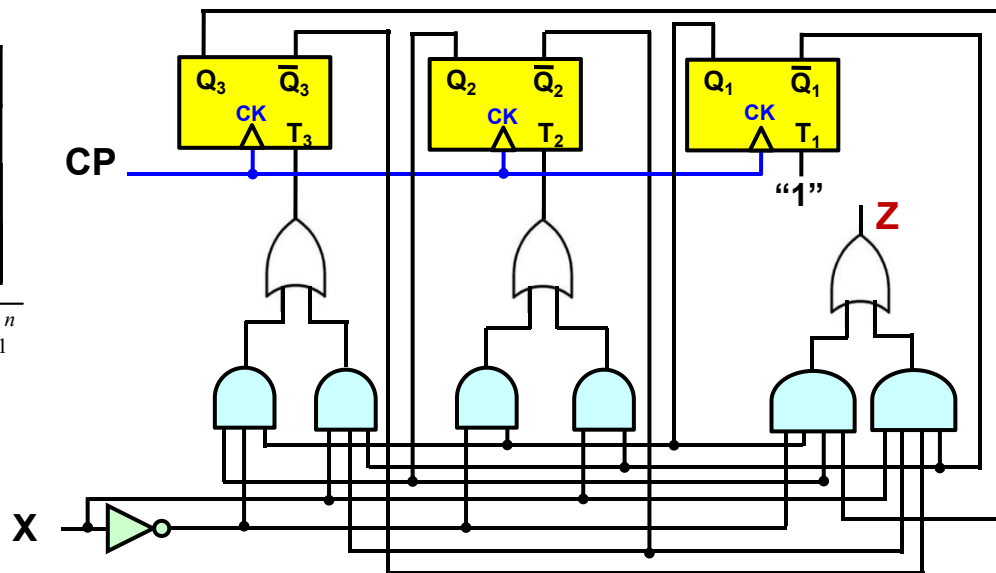
$$T_2 = \overline{X} Q_1^n + X \overline{Q_1^n}$$

$XQ_3^n \backslash Q_2^n Q_1^n$	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	0	0	0	0
10	1	0	0	0

$$T_1 = 1$$

$$Z = X \overline{Q_3^n} \overline{Q_2^n} \overline{Q_1^n} + \overline{X} Q_3^n Q_2^n Q_1^n$$

4. 电路实现



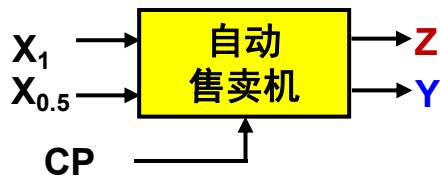
用触发器设计同步时序逻辑—实例

- 模8可逆计数器
- 自动售卖机
- 时序锁
- 二进制串行加法器
- 串行输入的8421BCD码检测器
- 奇偶校验器
- 更复杂的同步时序逻辑设计

利用触发器设计同步时序逻辑_例2

例2：利用D触发器设计一个自动售卖机

- 只接收硬币： 0.5 ¥ , 1 ¥
- 每次投币只接收一枚硬币
- 机器收到1.5 ¥ , 给出一瓶饮料
- 机器收到2.0 ¥ , 给出一瓶饮料, 找回0.5 ¥



$X_1 X_{0.5} = 00$: 0 ¥

$X_1 X_{0.5} = 01$: 0.5 ¥

$X_1 X_{0.5} = 10$: 1 ¥

Y=1/ 0: 给/不给 饮料

Z=1/ 0 : 找零/不找零

1. 原始状态图及状态表

① 状态设定

S_0 —初始状态, 无投币

S_1 —机器收到0.5 ¥

S_2 —机器收到1.0 ¥ (2个 0.5 ¥, or 1个1.0 ¥)

if (机器又收到1个0.5 ¥)

then Y=1, 且 Z=0, 回到 S_0

Else If (机器又收到1个1 ¥)

then Y=1, 且 Z=1, 回到 S_0

Solution 1:
Mealy circuit

利用触发器设计同步时序逻辑_例2

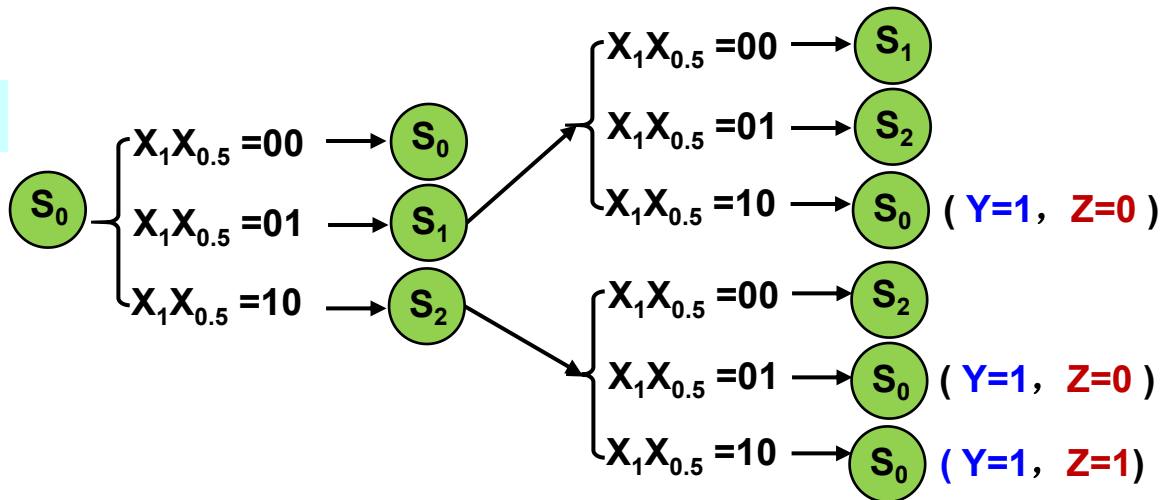
② 状态转换分析

Solution 1: Mealy circuit

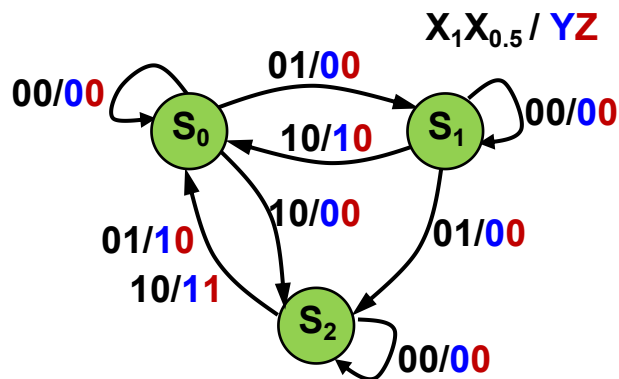
S_0 —无投币

S_1 —0.5 ¥

S_2 —机器收到1.0 ¥



③ Mealy 状态图



④ 状态表

现态 S^n	S^{n+1} / Z			
	$X_1X_{0.5}=00$	$X_1X_{0.5}=01$	$X_1X_{0.5}=10$	$X_1X_{0.5}=11$
S_0	$S_0 / 00$	$S_1 / 00$	$S_2 / 00$	X / XX
S_1	$S_1 / 00$	$S_2 / 00$	$S_0 / 10$	X / XX
S_2	$S_2 / 00$	$S_0 / 10$	$S_0 / 11$	X / XX

利用触发器设计同步时序逻辑_例2

④ 状态表

现态 S^n	S^{n+1} / Z			
	$X_1 X_{0.5}=00$	$X_1 X_{0.5}=01$	$X_1 X_{0.5}=10$	$X_1 X_{0.5}=11$
S_0	$S_0 / 00$	$S_1 / 00$	$S_2 / 00$	X / XX
S_1	$S_1 / 00$	$S_2 / 00$	$S_0 / 10$	X / XX
S_2	$S_2 / 00$	$S_0 / 10$	$S_0 / 11$	X / XX

2. 状态化简

3. 状态分配

S_0 — 00

S_1 — 01

S_2 — 10

需要2个D触发器

	0	1
0	S_0	S_1
1	S_2	

4. 状态转换真值

输入		现态		次态		输入		输出	
X_1	$X_{0.5}$	Q_2^n	Q_1^n	Q_2^{n+1}	Q_1^{n+1}	D_2	D_1	Y	Z
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	1	0	0
0	0	1	0	1	0	1	0	0	0
0	0	1	1	X	X	X	X	X	X
0	1	0	0	0	1	0	1	0	0
0	1	0	1	1	0	1	0	0	0
0	1	1	0	0	0	0	0	1	0
0	1	1	1	X	X	X	X	X	X
1	0	0	0	1	0	1	0	0	0
1	0	0	1	0	0	0	0	1	0
1	0	1	0	0	0	0	0	1	1
1	0	1	1	X	X	X	X	X	X
1	1	0	0	X	X	X	X	X	X
1	1	0	1	X	X	X	X	X	X
1	1	1	0	X	X	X	X	X	X
1	1	1	1	X	X	X	X	X	X

利用触发器设计同步时序逻辑_例2

5. 卡诺图化简

$X_1X_{0.5}$		$Q_2^nQ_1^n$			
		00	01	11	10
00	0	0	X	1	
01	0	1	X	0	
11	X	X	X	X	
10	1	0	X	0	

$$D_2 = \bar{X}_1\bar{X}_{0.5}Q_2^n + Q_1^nX_{0.5} + X_1\bar{Q}_1^n\bar{Q}_2^n$$

$X_1X_{0.5}$		$Q_2^nQ_1^n$			
		00	01	11	10
00	0	1	X	0	
01	1	0	X	0	
11	X	X	X	X	
10	0	0	X	0	

$$D_1 = \bar{X}_1\bar{X}_{0.5}Q_1^n + X_{0.5}\bar{Q}_1^n\bar{Q}_2^n$$

$X_1X_{0.5}$		$Q_2^nQ_1^n$			
		00	01	11	10
00	0	0	X	0	
01	0	0	X	1	
11	X	X	X	X	
10	0	1	X	1	

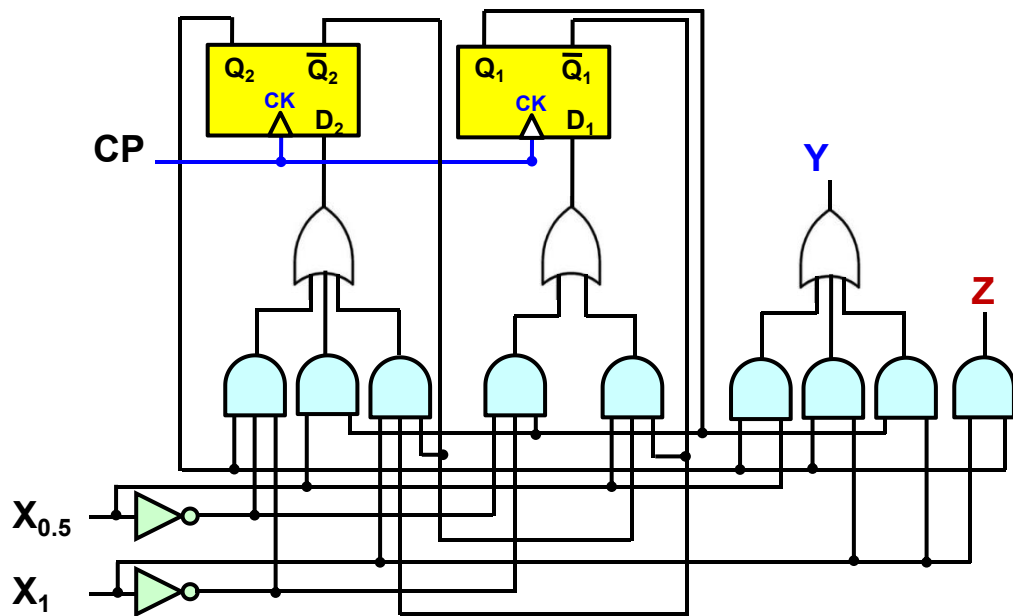
$$Y = Q_2^nX_{0.5} + Q_2^nX_1 + X_1Q_1^n$$

$X_1X_{0.5}$		$Q_2^nQ_1^n$			
		00	01	11	10
00	0	0	X	0	
01	0	0	X	0	
11	X	X	X	X	
10	0	0	X	1	

$$Z = X_1Q_2^n$$

利用触发器设计同步时序逻辑_例2

6. 电路实现



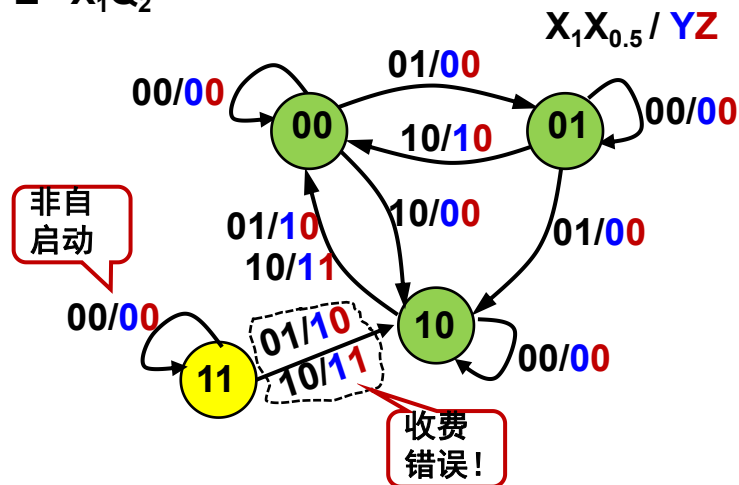
! 电路需要预置

7. 检查无关项

无关状态: $Q_2^n Q_1^n = 11$

$X_1 X_{0.5}$ 分别为 00, 01, 10 时, 带入计算

$$\begin{cases} Q_2^{n+1} = D_2 = \bar{X}_1 \bar{X}_{0.5} Q_1^n + Q_1 X_{0.5} + X_1 \bar{Q}_1^n \bar{Q}_2^n \\ Q_1^{n+1} = D_1 = \bar{X}_1 \bar{X}_{0.5} Q_2^n + X_{0.5} \bar{Q}_1^n \bar{Q}_2^n \\ Y = Q_2^n X_{0.5} + Q_2^n X_1 + X_1 Q_1^n \\ Z = X_1 Q_2^n \end{cases}$$



利用触发器设计同步时序逻辑_例2

1. 原始状态图及状态表

① 状态设定 (标记收到的钱数)

S_0 —初始状态, 机器收到0 ¥

S_1 —机器收到0.5 ¥

S_2 —机器收到1.0 ¥

S_3 —机器收到1.5 ¥

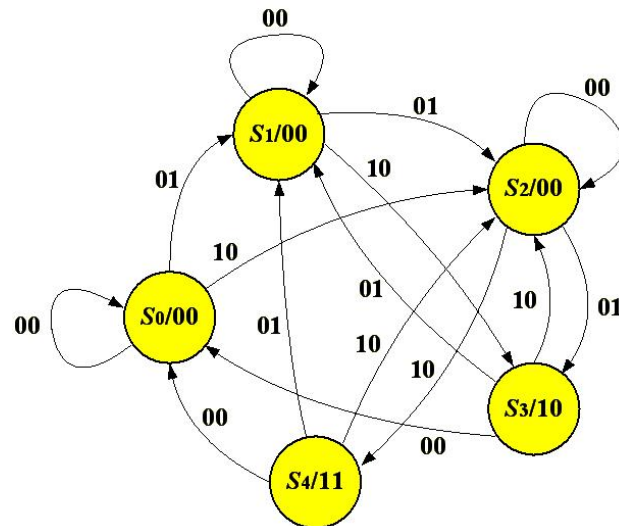
S_4 —机器收到2.0 ¥

Solution 2:
Moor circuit

③ Moor 状态表

现态 S_n	次态 S_{n+1}			输出 YZ
	$X_1X_2 = 00$	$X_1X_2 = 01$	$X_1X_2 = 10$	
S_0	S_0	S_1	S_2	00
S_1	S_1	S_2	S_3	00
S_2	S_2	S_3	S_4	00
S_3	S_0	S_1	S_2	10
S_4	S_0	S_1	S_2	11

② Moor 状态图



2. 状态化简

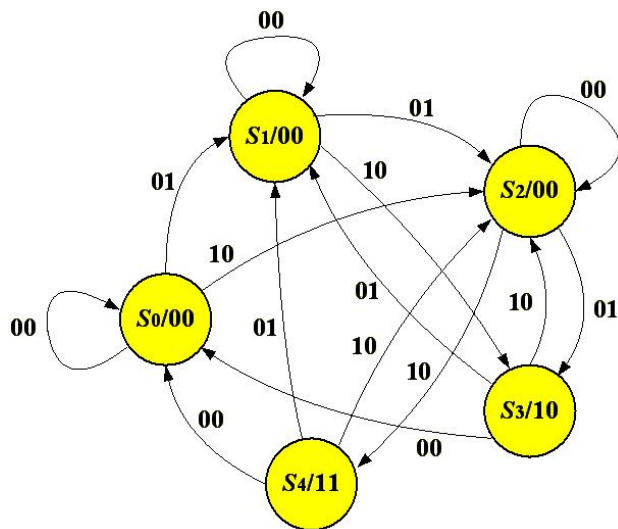
3. 状态分配

需要3个D触发器

$Q_2^n Q_1^n$		00	01	11	10
Q_3^n	0	S_0	S_3		S_1
	1	S_4			S_2

S_0	000
S_1	010
S_2	110
S_3	001
S_4	100

4. 状态转换真值表



S₀ — 000
S₁ — 010
S₂ — 110
S₃ — 001
S₄ — 100

[illegible]

利用触发器设计同步时序逻辑_例2

5. 卡诺图化简

$X_1=0$

$X_{0.5}Q_3^n$ \ $Q_2^nQ_1^n$	00	01	11	10
00	0	0	X	0
01	0	X	X	1
11	0	X	X	0
10	0	0	X	1

$X_1=0$

$X_{0.5}Q_3^n$ \ $Q_2^nQ_1^n$	00	01	11	10
00	0	0	X	1
01	0	X	X	1
11	1	X	X	0
10	1	1	X	1

$X_1=0$

$X_{0.5}Q_3^n$ \ $Q_2^nQ_1^n$	00	01	11	10
00	0	0	X	0
01	0	X	X	0
11	0	X	X	1
10	0	0	X	0

$X_1=1$

$X_{0.5}Q_3^n$ \ $Q_2^nQ_1^n$	00	01	11	10
00	1	1	X	0
01	1	X	X	1
11	X	X	X	X
10	X	X	X	X

$X_1=1$

$X_{0.5}Q_3^n$ \ $Q_2^nQ_1^n$	00	01	11	10
00	1	1	X	0
01	1	X	X	0
11	X	X	X	X
10	X	X	X	X

$X_1=1$

$X_{0.5}Q_3^n$ \ $Q_2^nQ_1^n$	00	01	11	10
00	0	0	X	1
01	0	X	X	0
11	X	X	X	X
10	X	X	X	X

$$D_3 = \bar{X}_{0.5}Q_3^nQ_2^n + \bar{Q}_3^nX_{0.5}Q_2^n + X_1\bar{Q}_2^n$$

$$D_2 = \bar{X}_{0.5}Q_3^n + \bar{Q}_2^nX_{0.5} + X_1\bar{Q}_2^n + \bar{X}_1\bar{X}_{0.5}Q_2^n$$

$$D_1 = X_{0.5}Q_3^nQ_2^n + \bar{Q}_3^nX_1Q_2^n$$

利用触发器设计同步时序逻辑_例2

$X_1=0$

$Q_2^n Q_1^n$	00	01	11	10
$X_{0.5} Q_3^n$	0	1	X	0
00	0	1	X	0
01	1	X	X	0
11	1	X	X	0
10	0	1	X	1

$X_1=1$

$Q_2^n Q_1^n$	00	01	11	10
$X_{0.5} Q_3^n$	0	1	X	0
00	0	1	X	0
01	1	X	X	0
11	X	X	X	X
10	X	X	X	X

$$Y = \overline{Q_2^n} Q_3^n + Q_1^n$$

$X_1=0$

$Q_2^n Q_1^n$	00	01	11	10
$X_{0.5} Q_3^n$	0	0	X	0
00	0	0	X	0
01	1	X	X	0
11	1	X	X	0
10	0	0	X	0

$X_1=1$

$Q_2^n Q_1^n$	00	01	11	10
$X_{0.5} Q_3^n$	0	0	X	0
00	0	0	X	0
01	1	X	X	0
11	X	X	X	X
10	X	X	X	X

$$Z = \overline{Q_2^n} Q_3^n$$

$$\begin{cases} D_3 = \overline{X_{0.5}} Q_3^n Q_2^n + \overline{Q_3^n} X_{0.5} Q_2^n + X_1 \overline{Q_2^n} \\ D_2 = \overline{X_{0.5}} Q_3^n + \overline{Q_2^n} X_{0.5} + X_1 \overline{Q_2^n} + \overline{X_1} \overline{X_{0.5}} Q_2^n \\ D_1 = X_{0.5} Q_3^n Q_2^n + \overline{Q_3^n} X_1 Q_2^n \\ Y = \overline{Q_2^n} Q_3^n + Q_1^n \\ Z = \overline{Q_2^n} Q_3^n \end{cases}$$

6. 电路实现(略)

7. 检查无关项(略)

Moor型电路与Mealy型电路比较

- Moor型电路中的状态总数相对要多一些，需要使用较多的触发器资源。
- Moor型电路的输出只与状态有关，输出没有毛刺。

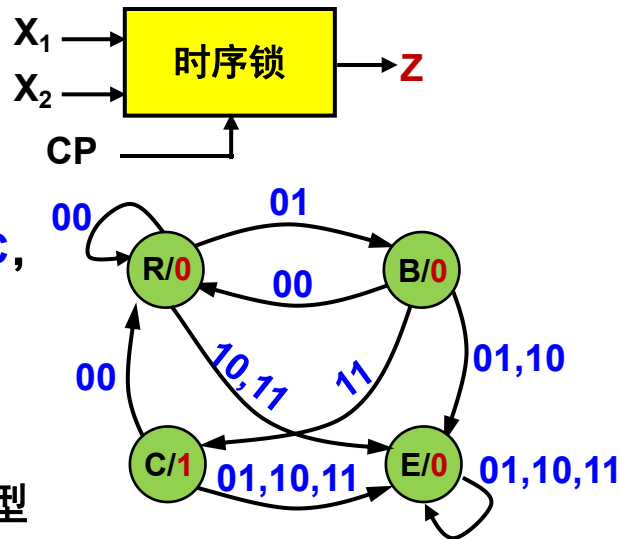
用触发器设计同步时序逻辑—实例

- 模8可逆计数器
- 自动售卖机
- 时序锁
- 二进制串行加法器
- 串行输入的8421BCD码检测器
- 奇偶校验器
- 更复杂的同步时序逻辑设计

利用触发器设计同步时序逻辑_例3

例3：利用JK触发器设计一个时序锁

- 输入: X_1X_2 , 输出: Z
- 该锁内部有四个状态 R 、 B 、 C 、 E
- 依次输入00、01、11, 时序锁从状态 $R \rightarrow B \rightarrow C$, 并开锁 ($Z=1$)
- 不是上述序列, 进入状态 E (error)
- 任何时候只要输入00, 都将返回状态 R



摩尔型

1. 原始状态图及状态表

① 状态设定

R —初始状态, 输入00

B —输入00后, 再输入01

C —输入00、01后, 再输入11, 且 $Z=1$

E —错误状态

现态 S_n	次态 S_{n+1}				输出 Z
	$X_1X_2 = 00$	$X_1X_2 = 01$	$X_1X_2 = 11$	$X_1X_2 = 10$	
R	R	B	E	E	0
B	R	E	C	E	0
C	R	E	E	E	1
E	R	E	E	E	0

利用触发器设计同步时序逻辑_例3

现态 S_n	次态 S_{n+1}				输出 Z
	$X_1X_2 = 00$	$X_1X_2 = 01$	$X_1X_2 = 11$	$X_1X_2 = 10$	
R	R	B	E	E	0
B	R	E	C	E	0
C	R	E	E	E	1
E	R	E	E	E	0

2. 状态化简

3. 状态分配

需要2个JK触发器

R: 00, B: 01

E: 10, C: 11

	0	1
0	R	B
1	E	C

4. 状态转换真值表

$J_2 K_2$: 看 $Q_2^n \rightarrow Q_2^{n+1}$

$J_1 K_1$: 看 $Q_1^n \rightarrow Q_1^{n+1}$

输入		现态		次态		输入				输出
X_1	X_2	Q_2^n	Q_1^n	Q_2^{n+1}	Q_1^{n+1}	J_2	K_2	J_1	K_1	Z
0	0	0	0	0	0	0	X	0	X	0
0	0	0	1	0	0	0	X	X	1	0
0	0	1	0	0	0	X	1	0	X	0
0	0	1	1	0	0	X	1	X	1	1
0	1	0	0	0	1	0	X	1	X	0
0	1	0	1	1	0	1	X	X	1	0
0	1	1	0	1	0	X	0	0	X	0
0	1	1	1	1	0	X	0	X	1	1
1	0	0	0	1	0	1	X	0	X	0
1	0	0	1	1	0	1	X	X	1	0
1	0	1	0	1	0	X	0	0	X	0
1	0	1	1	1	0	X	0	X	1	1
1	1	0	0	1	0	1	X	0	X	0
1	1	0	1	1	1	1	X	X	0	0
1	1	1	0	1	0	X	0	0	X	0
1	1	1	1	1	0	X	0	X	1	1

利用触发器设计同步时序逻辑_例3

5. 卡诺图化简

X_1X_2	$Q_2^nQ_1^n$			
	00	01	11	10
00	0	0	X	X
01	0	1	X	X
11	1	1	X	X
10	1	1	X	X

$$J_2 = X_2Q_1^n + X_1$$

X_1X_2	$Q_2^nQ_1^n$			
	00	01	11	10
00	X	X	1	1
01	X	X	0	0
11	X	X	0	0
10	X	X	0	0

$$K_2 = \bar{X}_2\bar{X}_1$$

X_1X_2	$Q_2^nQ_1^n$			
	00	01	11	10
00	0	X	X	0
01	1	X	X	0
11	0	X	X	0
10	0	X	X	0

$$J_1 = \bar{X}_1X_2\bar{Q}_2^n$$

X_1X_2	$Q_2^nQ_1^n$			
	00	01	11	10
00	X	1	1	X
01	X	1	1	X
11	X	0	1	X
10	X	1	1	X

$$K_1 = Q_2^n + \bar{X}_2 + \bar{X}_1$$

X_1Q_2	$Q_2^nQ_1^n$			
	00	01	11	10
00	0	0	1	0
01	0	0	1	0
11	0	0	1	0
10	0	0	1	0

$$Z = Q_2^nQ_1^n$$

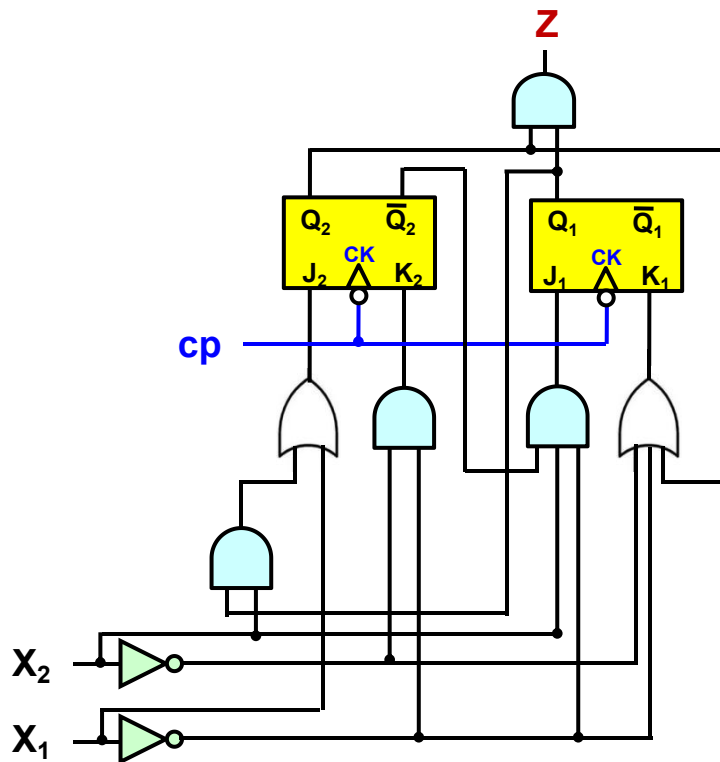
利用触发器设计同步时序逻辑_例3

6. 电路实现

$$\left\{ \begin{array}{l} J_2 = X_2 Q_1^n + X_1 \\ K_2 = \overline{X_2} \overline{X_1} \\ J_1 = \overline{X_1} X_2 \overline{Q_2}^n \\ K_1 = Q_2^n + \overline{X_2} + \overline{X_1} \\ Z = Q_2^n Q_1^n \end{array} \right.$$

密码锁

- 一维开锁：密码正确
- 二维开锁：有限时间+密码正确
- 三维开锁：
有限时间+有限按键次数+密码正确

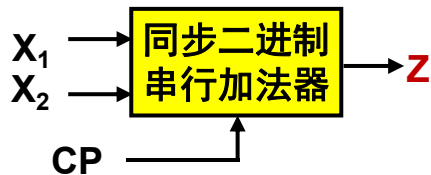


用触发器设计同步时序逻辑—实例

- 模8可逆计数器
- 自动售卖机
- 时序锁
- 二进制串行加法器
- 串行输入的8421BCD码检测器
- 奇偶校验器
- 更复杂的同步时序逻辑设计

利用触发器设计同步时序逻辑_例4

例4：利用JK触发器设计一个同步二进制串行加法器



③ Mealy 状态表

现态 Q^n	Q^{n+1} / Z			
	$X_1X_2=00$	$X_1X_2=01$	$X_1X_2=10$	$X_1X_2=11$
a	a / 0	a / 1	a / 1	b / 0
b	a / 1	b / 0	b / 0	b / 1

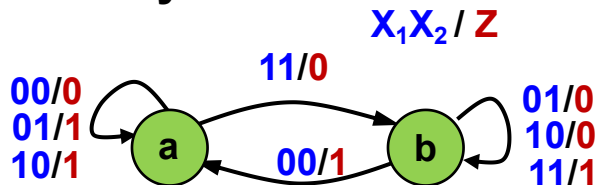
1. 原始状态图及状态表

① 设加法器内部状态

a—— 无进位

b—— 有进位

② Mealy 状态图



2. 状态化简 3. 状态分配 $a=0, b=1$

4. 状态转换真值表

输入 现态 次态				输入 输出		
X_1	X_2	Q^n	Q^{n+1}	J	K	Z
0	0	0	0	0	X	0
0	0	1	0	X	1	1
0	1	0	0	0	X	1
0	1	1	1	X	0	0
1	0	0	0	0	X	1
1	0	1	1	X	0	0
1	1	0	1	1	X	0
1	1	1	1	X	0	1

利用触发器设计同步时序逻辑_例4

5. 卡诺图化简

X_1	$X_2 Q^n$			
	00	01	11	
0	0	X	X	0
1	0	X	X	1

$$J = X_1 X_2$$

X_1	$X_2 Q^n$			
	00	01	11	
0	X	1	0	X
1	X	0	0	X

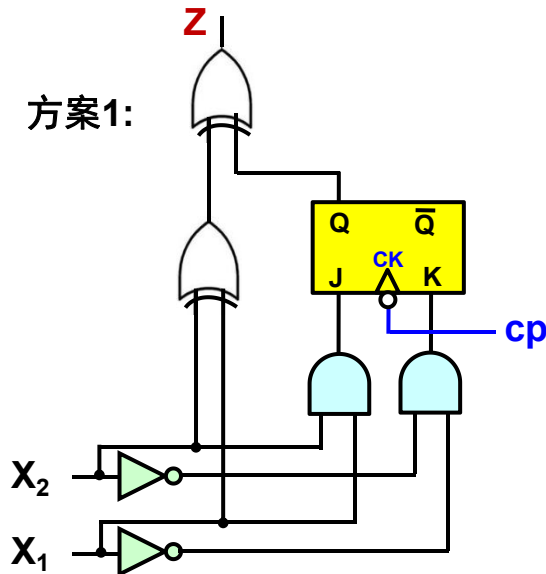
$$K = \bar{X}_1 \bar{X}_2$$

X_1	$X_2 Q^n$			
	00	01	11	
0	0	1	0	1
1	1	0	1	0

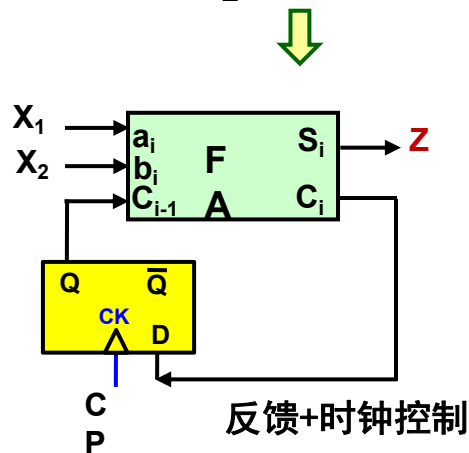
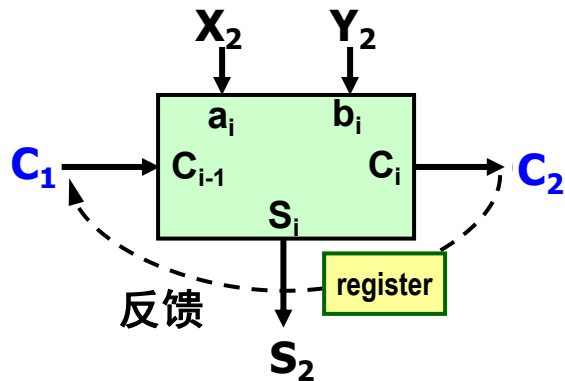
$$Z = X_1 \oplus X_2 \oplus Q^n$$

6. 电路实现

方案1:



方案2: 如何用一位全加器实现?



用触发器设计同步时序逻辑—实例

- 模8可逆计数器
- 自动售卖机
- 时序锁
- 二进制串行加法器
- 串行输入的8421BCD码检测器
- 奇偶校验器
- 更复杂的同步时序逻辑设计

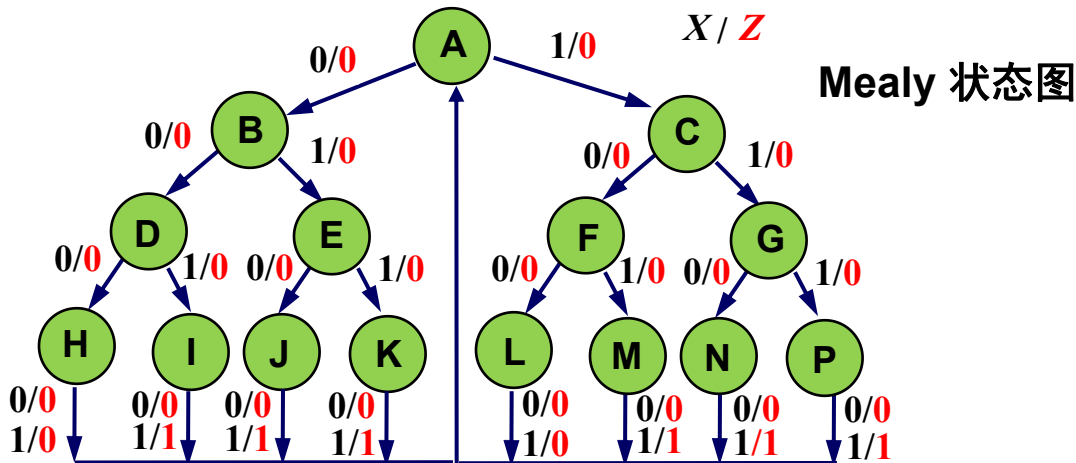
利用触发器设计同步时序逻辑_例5

例5：用D触发器设计一个串行输入的8421BCD码误码检测器

要求：

- 8421BCD码**低位在前**、**高位在后**串行地加到检测器的输入端。
- 电路每接收一组代码，即在收到第4位代码时判断。若是错误代码，则输出为1，否则输出为0，电路又回到初始状态并开始接收下一组代码。

1. 原始状态图及状态表



利用触发器设计同步时序逻辑_例5

2. 状态化简

现态 Q^n	Q^{n+1}/Z	
	$X=0$	$X=1$
A	B / 0	C / 0
B	D / 0	E / 0
C	F / 0	G / 0
D	H / 0	I / 0
E	J / 0	K / 0
F	L / 0	M / 0
G	N / 0	P / 0
H	A / 0	A / 0
I	A / 0	A / 1
J	A / 0	A / 1
K	A / 0	A / 1
L	A / 0	A / 0
M	A / 0	A / 1
N	A / 0	A / 1
P	A / 0	A / 1



现态 Q^n	Q^{n+1}/Z	
	$X=0$	$X=1$
A	B / 0	C / 0
B	D / 0	E / 0
C	F / 0	G / 0
D	H / 0	I / 0
E	I / 0	I / 0
F	H / 0	I / 0
G	I / 0	I / 0
H	A / 0	A / 0
I	A / 0	A / 1




现态 Q^n	Q^{n+1}/Z	
	$X=0$	$X=1$
A	B / 0	C / 0
B	D / 0	E / 0
C	D / 0	E / 0
D	H / 0	I / 0
E	I / 0	I / 0
H	A / 0	A / 0
I	A / 0	A / 1

利用触发器设计同步时序逻辑_例5

2. 状态化简

现态 Q^n	Q^{n+1} / Z	
	$X=0$	$X=1$
A	B / 0	C / 0
B	D / 0	E / 0
C	D / 0	E / 0
D	H / 0	I / 0
E	I / 0	I / 0
H	A / 0	A / 0
I	A / 0	A / 1



现态 Q^n	Q^{n+1} / Z	
	$X=0$	$X=1$
A	B / 0	B / 0
B	D / 0	E / 0
D	H / 0	I / 0
E	I / 0	I / 0
H	A / 0	A / 0
I	A / 0	A / 1

3. 状态分配

规则1: 次态相同, 现态编码应相邻


HI, DE 应相邻

规则2: 同一现态对应的次态应给予相邻编码

DE, HI 应相邻

规则3: 输出相同, 现态编码应相邻

ABDEH应相邻



		$Q_2^n Q_1^n$			
Q_3^n		00	01	11	
		A / 0	B	D	I
	1			E	H

A: 000; B: 001
D: 011; I: 010
E: 111; H: 110

确定 D_3 : 看 Q_3^{n+1}
 确定 D_2 : 看 Q_2^{n+1}
 确定 D_1 : 看 Q_1^{n+1}

4. 状态转换真值表

Q_3^n	$Q_2^n Q_1^n$			
	00	01	11	10
0	A	B	D	I
1			E	H



现态 Q^n	Q^{n+1} / Z	
	$X=0$	$X=1$
A	B / 0	B / 0
B	D / 0	E / 0
D	H / 0	I / 0
E	I / 0	I / 0
H	A / 0	A / 0
I	A / 0	A / 1

输入及现态				次态			输入 输出			
X	Q_3^n	Q_2^n	Q_1^n	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	D_3	D_2	D_1	Z
0	0	0	0	0	0	1	0	0	1	0
0	0	0	1	0	1	1	0	1	1	0
0	0	1	0	0	0	0	0	0	0	0
0	0	1	1	1	1	0	1	1	0	0
0	1	0	0	X	X	X	X	X	X	X
0	1	0	1	X	X	X	X	X	X	X
0	1	1	0	0	0	0	0	0	0	0
0	1	1	1	0	1	0	0	1	0	0
1	0	0	0	0	0	1	0	0	1	0
1	0	0	1	1	1	1	1	1	1	0
1	0	1	0	0	0	0	0	0	0	1
1	0	1	1	0	1	0	0	1	0	0
1	1	0	0	X	X	X	X	X	X	X
1	1	0	1	X	X	X	X	X	X	X
1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	0	1	0	0	1	0	0

利用触发器设计同步时序逻辑_例5

5. 卡诺图化简

XQ_3^n	$Q_2^n Q_1^n$			
	00	01	11	10
00	0	0	1	0
01	X	X	0	0
11	X	X	0	0
10	0	1	0	0

XQ_3^n	$Q_2^n Q_1^n$			
	00	01	11	10
00	0	1	1	0
01	X	X	1	0
11	x	x	1	0
10	0	1	1	0

$$D_3 = \overline{Q_3^n} Q_2^n Q_1^n \overline{X} + X \overline{Q_2^n} Q_1^n$$

$$D_2 = Q_1^n$$

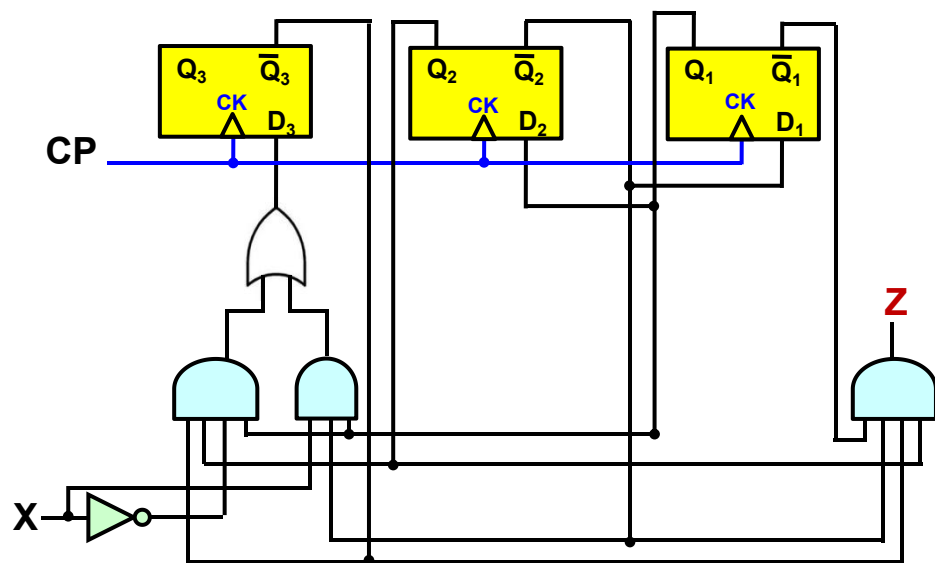
XQ_3^n	$Q_2^n Q_1^n$			
	00	01	11	10
00	1	1	0	0
01	x	x	0	0
11	x	x	0	0
10	1	1	0	0

XQ_3^n	$Q_2^n Q_1^n$			
	00	01	11	10
00	0	0	0	0
01	x	x	0	0
11	x	x	0	0
10	0	0	0	1

$$D_1 = \overline{Q_2^n}$$

$$Z = X \overline{Q_3^n} Q_2^n \overline{Q_1^n}$$

6. 电路实现

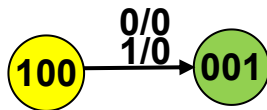
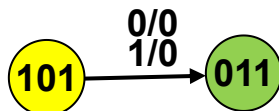


利用触发器设计同步时序逻辑_例5

7. 无关项检查

$Q_2^n Q_1^n$		00	01	11	10
Q_3^n	0	A	B	D	I
	1			E	H

将无关状态 $Q_3^n Q_2^n Q_1^n = 100$ 和 101 分别代入次态方程和输出方程计算



电路可以自启动



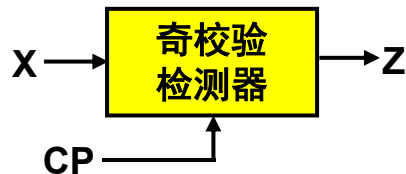
$$\left\{ \begin{array}{l} Q_i^{n+1} = D_i \\ D_3 = \overline{Q_3^n} \overline{Q_2^n} Q_1^n \overline{X} + X \overline{Q_2^n} Q_1^n \\ D_2 = Q_1^n \\ D_1 = \overline{Q_2^n} \\ Z = X \overline{Q_3^n} \overline{Q_2^n} \overline{Q_1^n} \end{array} \right.$$

利用触发器设计时序逻辑——实例

- 模8可逆计数器
- 自动售卖机
- 时序锁
- 二进制串行加法器
- 串行输入的8421BCD码检测器
- 奇偶校验器
- 更复杂的同步时序逻辑设计

利用触发器设计同步时序逻辑_例6

例6：利用T触发器设计一个串行输入的奇校验检测器



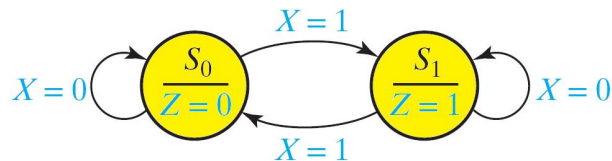
1. 原始状态图及状态表

① 状态设定

S_0 ——表示收到偶数个“1”，初始为0个“1”

S_1 ——表示收到奇数个“1”

② Moor 状态图



③ 状态表

现态 Q^n	次态 Q^{n+1}		输出 Z
	$X=0$	$X=1$	
S_0	S_0	S_1	0
S_1	S_1	S_0	1

2. 状态化简

3. 状态分配

$S_0: 0; S_1: 1$

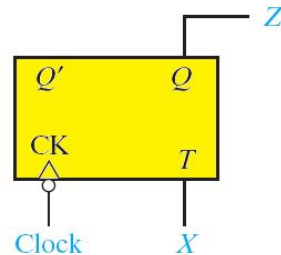
4. 状态转换真值表

输入	现态	次态	输入	输出
X	Q^n	Q^{n+1}	T	Z
0	0	0	0	0
0	1	1	0	1
1	0	1	1	0
1	1	0	1	1

5. 卡诺图化简

$T=X; Z=Q^n$

6. 电路实现



用触发器设计同步时序逻辑—实例

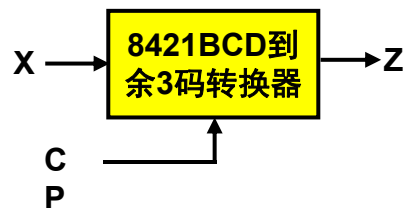
- 模8可逆计数器
- 自动售卖机
- 时序锁
- 二进制串行加法器
- 串行输入的8421BCD码检测器
- 奇偶校验器
- 更复杂的同步时序逻辑设计

更复杂的同步时序设计_例7

例7：利用D触发器设计一个同步时序的码制转换器，将串行输入的8421BCD码转换为余3码。

- 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				1
			1				0
			0				1
			1				0
			0				1
			1				0
			0				1
			1				0
			0				1
			1				0



□ t_0 时刻:

输入为0, 输出为1

输入为1, 输出为0

□ $t_1 \sim t_3$ 时刻:

单纯看没有规律,
要联合前一时刻的
输入一同来看

更复杂的同步时序设计_例7

- t_0 时刻: 输入为0, 输出为1; 输入为1, 输出为0
- $t_1 \sim t_3$ 时刻: 单纯看没有规律, 要联合前一时刻的输入一同来看

$t_1 t_0$ 时刻 输入	$t_1 t_0$ 时刻 输出
00	11
01	00
10	01
11	10

$t_2 t_1 t_0$ 时刻 输入	$t_2 t_1 t_0$ 时刻 输出
000	011
001	100
010	101
011	110
100	111
101	000
110	001
111	010

$t_3 t_2 t_1 t_0$ 时刻 输入	$t_3 t_2 t_1 t_0$ 时刻 输出
0000	0011
0001	0100
0010	0101
0011	0110
0100	0111
0101	1000
0110	1001
0111	1010
1000	1011
1001	1100

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

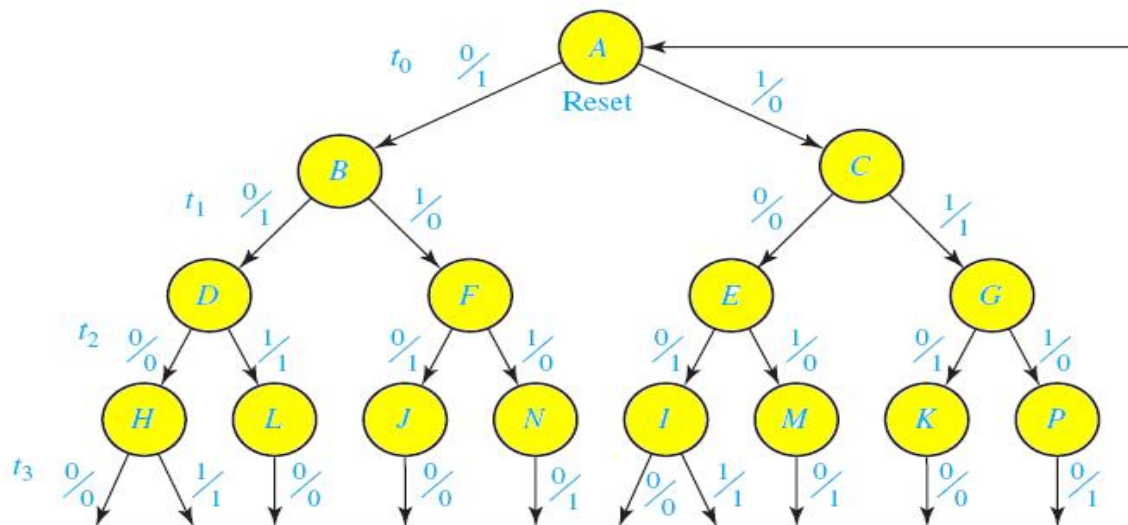
更复杂的同步时序设计_例7

1. 原始状态图及状态表

- t_0 时刻: 输入为0, 输出为1; 输入为1, 输出为0
- $t_1 \sim t_3$ 时刻: 单纯看没有规律, 要联合前一时刻的输入一同来看

$t_1 t_0$ 时刻 输入	$t_1 t_0$ 时刻 输出
00	11
01	00
10	01
11	10

$t_2 t_1 t_0$ 时刻 输入	$t_2 t_1 t_0$ 时刻 输出
000	011
001	100
010	101
011	110
100	111
101	000
110	001
111	010



$t_3 t_2 t_1 t_0$ 时刻 输入	$t_3 t_2 t_1 t_0$ 时刻 输出
0000	0011
0001	0100
0010	0101
0011	0110
0100	0111
0101	1000
0110	1001
0111	1010
1000	1011
1001	1100

更复杂的同步时序设计_例7

2. 状态化简

Time	Input Sequence Received (Least Significant Bit First)	Present State	Next State		Present Output (Z)	
			X = 0	1	X = 0	1
t_0	reset	A	B	C	1	0
t_1	0	B	D	F	1	0
	1	C	E	G	0	1
t_2	00	D	H	L	0	1
	01	E	I	M	1	0
	10	F	J	N	1	0
	11	G	K	P	1	0
t_3	000	H	A	A	0	1
	001	I	A	A	0	1
	010	J	A	—	0	—
	011	K	A	—	0	—
	100	L	A	—	0	—
	101	M	A	—	1	—
	110	N	A	—	1	—
	111	P	A	—	1	—



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	—

更复杂的同步时序设计_例7

3. 状态分配

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. 状态转换真值表

		$Q_1^+ Q_2^+ Q_3^+$		Z	
		$Q_1 Q_2 Q_3$	$X = 0$	$X = 1$	$X = 0$
A	000	100	101	1	0
B	100	111	110	1	0
C	101	110	110	0	1
D	111	011	011	0	1
E	110	011	010	1	0
H	011	000	000	0	1
M	010	000	x x x	1	x
—	001	x x x	x x x	x	x



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



更复杂的同步时序设计_例7

4. 状态转换真值表

		$Q_1^+ Q_2^+ Q_3^+$		Z	
		$X=0$	$X=1$	$X=0$	$X=1$
A	000	100	101	1	0
B	100	111	110	1	0
C	101	110	110	0	1
D	111	011	011	0	1
E	110	011	010	1	0
H	011	000	000	0	1
M	010	000	x x x	1	x
-	001	x x x	x x x	x	x

5. 卡诺图化简

$Q_2 Q_3$	XQ_1			
	00	01	11	10
00	1	1	1	1
01	X	1	1	X
11	0	0	0	0
10	0	0	0	X

$$D_1 = Q_1^+ = Q_2'$$

$Q_2 Q_3$	XQ_1			
	00	01	11	10
00	0	1	1	0
01	X	1	1	X
11	0	1	1	0
10	0	1	1	X

$$D_2 = Q_2^+ = Q_1$$

$Q_2 Q_3$	XQ_1			
	00	01	11	10
00	0	1	0	1
01	X	0	0	X
11	0	1	1	0
10	0	1	0	X

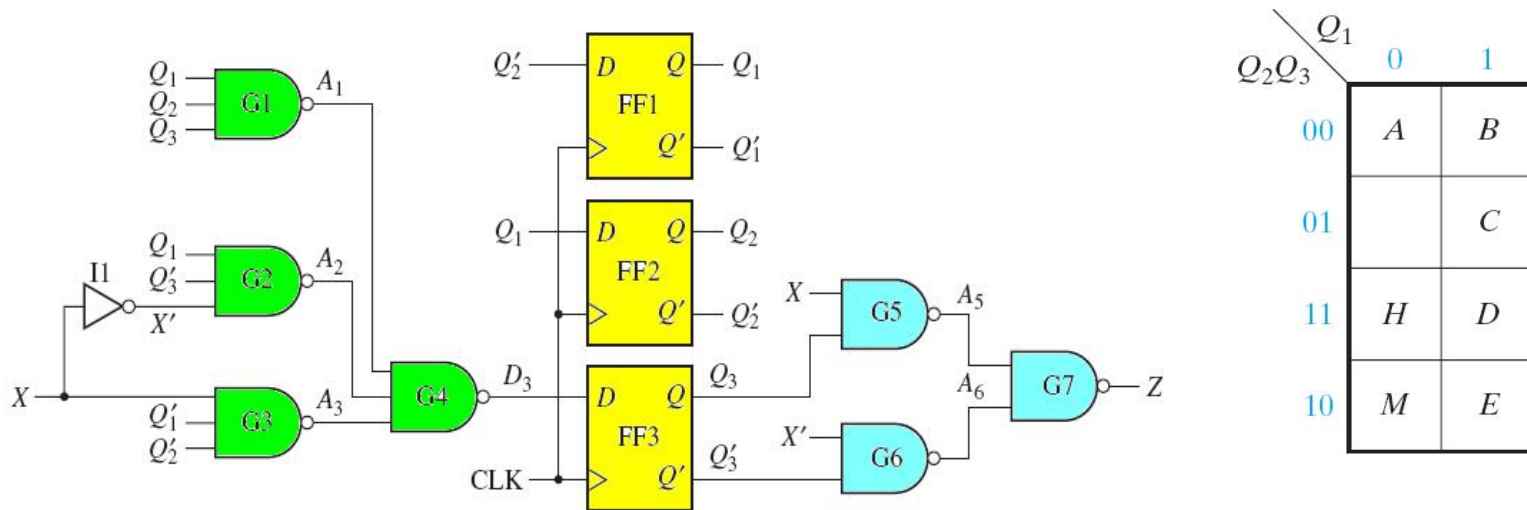
$$D_3 = Q_3^+ = Q_1 Q_2 Q_3 + X' Q_1 Q_3' + X Q_1' Q_2'$$

$Q_2 Q_3$	XQ_1			
	00	01	11	10
00	1	1	0	0
01	X	0	1	X
11	0	0	1	1
10	1	1	0	X

$$Z = X' Q_3' + X Q_3$$

更复杂的同步时序设计_例7

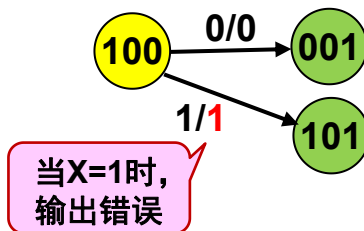
6. 电路实现



7. 无关项检查

将无关状态 $Q_3 Q_2 Q_1 = 100$ 代入次态方程和输出方程计算

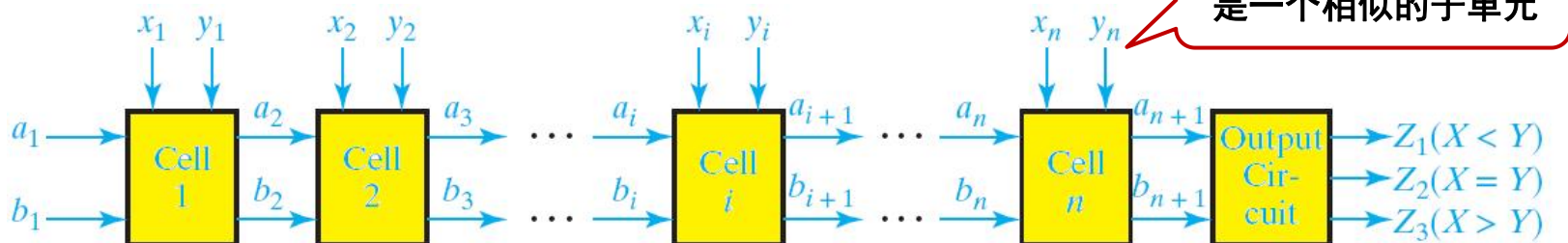
$$\begin{cases} D_1 = Q_1^+ = Q_2' \\ D_2 = Q_2^+ = Q_1 \\ D_3 = Q_3^+ = Q_1 Q_2 Q_3 + X' Q_1 Q_3' + X Q_1' Q_2' \\ Z = X' Q_3' + X Q_3 \end{cases}$$



电路可以自启动

更复杂的同步时序设计_例8

例8：迭代电路设计——利用D触发器设计一个比较器，能对两个 n 位二进制数进行比较。



1. 原始状态图及状态表

对于第 i 个单元，设状态——

S_0 : $X = Y$ 时

S_1 : $X > Y$ 时

S_2 : $X < Y$ 时

Z_2 、 Z_3 、 Z_1 分别取值为1

- 由 n 个比较子单元 (cell) 构成
- 从高位到低位，逐位对应比较，并将前一位比较的结果传送给下一位
- 第 i 个单元的比较结果： $X = Y$, $X > Y$, or $X < Y$.

更复杂的同步时序设计_例8

1. 原始状态图及状态表

	S_i	S_{i+1}				$Z_1 Z_2 Z_3$
		$x_i y_i = 00$	01	11	10	
$X = Y$	S_0	S_0	S_2	S_0	S_1	0 1 0
$X > Y$	S_1	S_1	S_1	S_1	S_1	0 0 1
$X < Y$	S_2	S_2	S_2	S_2	S_2	1 0 0

在第*i*个（前一个）单元有比较结果的前提下，根据输入取值，可以确定第*i+1*个单元的比较结果

对于第*i*个单元，设状态——
 S_0 : $X = Y$ 时
 S_1 : $X > Y$ 时
 S_2 : $X < Y$ 时
 Z_1 、 Z_2 、 Z_3 分别取值为1

2. 状态化简

3. 状态分配

S_0 : 00

S_1 : 01

S_2 : 10

需要两个触发器，
用 a,b 来表示

4. 状态转换真值表

$a_i b_i$	$a_{i+1} b_{i+1}$				$Z_1 Z_2 Z_3$
	$x_i y_i = 00$	01	11	10	
0 0	00	10	00	01	0 1 0
0 1	01	01	01	01	0 0 1
1 0	10	10	10	10	1 0 0

更复杂的同步时序设计_例8

5. 卡诺图化简

$x_i y_i$		00	01	11	10
$a_i b_i$	00	0	1	0	0
	01	0	0	0	0
	11	X	X	X	X
	10	1	1	1	1

$$a_{i+1} = a_i + x_i' y_i b_i'$$

$x_i y_i$		00	01	11	10
$a_i b_i$	00	0	0	0	1
	01	1	1	1	1
	11	X	X	X	X
	10	0	0	0	0

$$b_{i+1} = b_i + x_i y_i' a_i'$$

a_{n+1}		0	1
b_{n+1}	0		1
	1		X

$$Z_1 = a_{n+1}$$

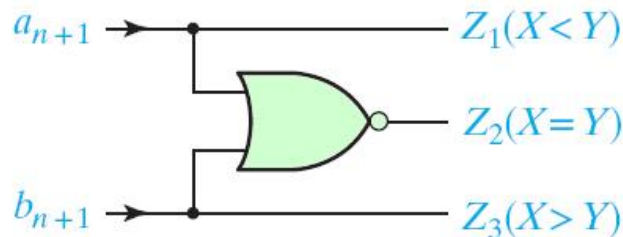
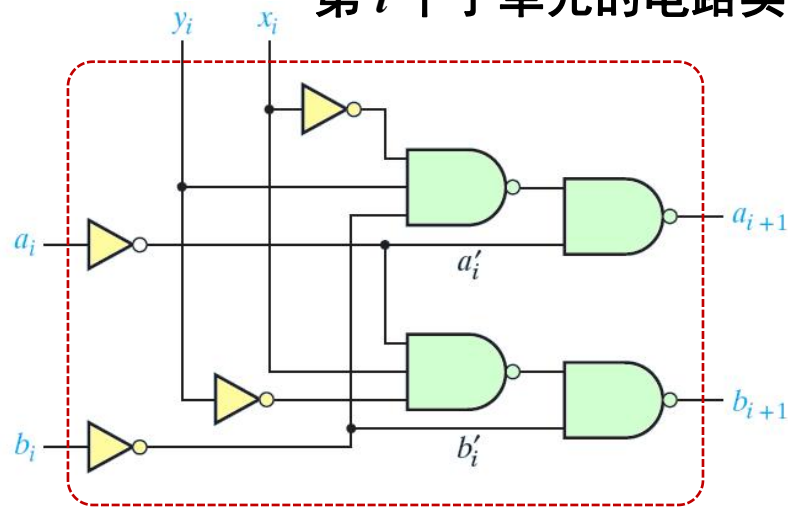
a_{n+1}		0	1
b_{n+1}	0	1	
	1		X

$$Z_2 = a_{n+1}' b_{n+1}'$$

a_{n+1}		0	1
b_{n+1}	0		
	1	1	X

$$Z_3 = b_{n+1}$$

第 i 个子单元的电路实现

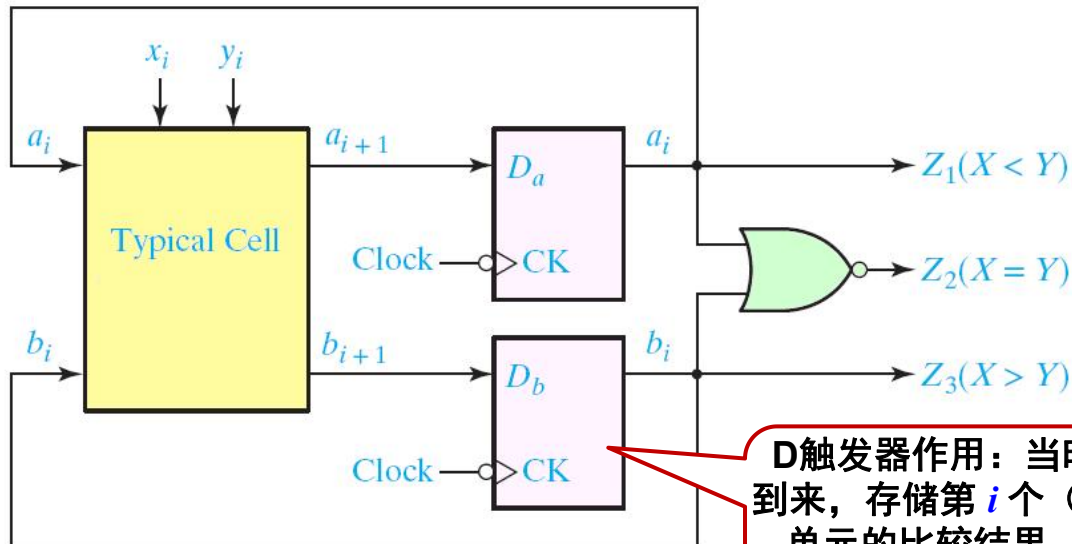


更复杂的同步时序设计_例8

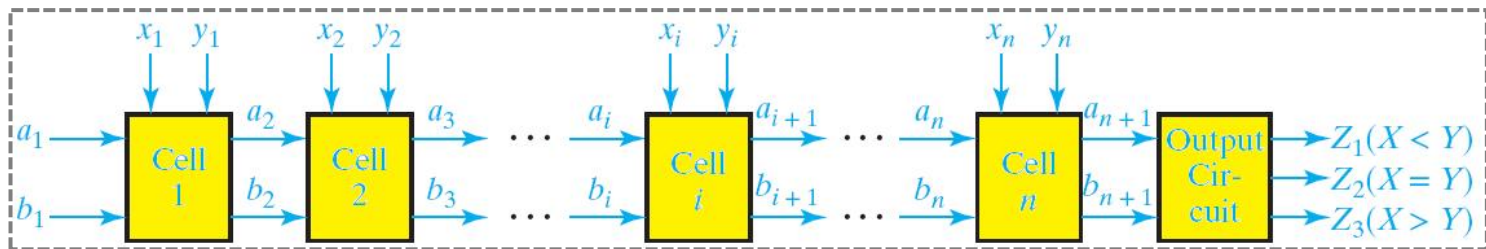
6. 电路实现

7. 无关项检查

(略)

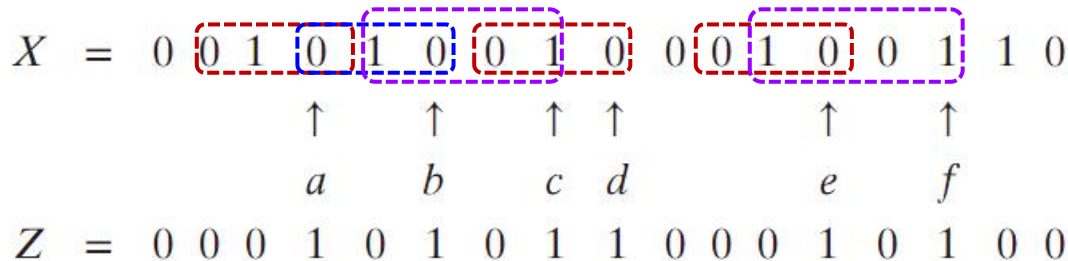


D触发器作用：当时钟信号到来，存储第 i 个（前一个）单元的比较结果，并产生第 $i+1$ 个单元的比较结果



更复杂的同步时序设计_例9

例9: 利用D触发器设计一个同步时序电路, 当输入序列以010或1001结尾时 (允许重叠检测), 输出Z为1, 否则Z=0.



1. Mealy型原始状态图构建

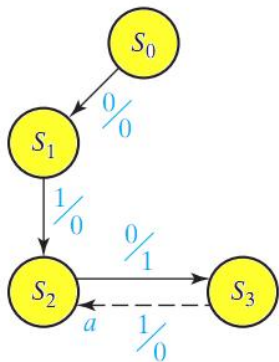
(1) 子序列010检测的状态设定

S_0 ——初始复位状态, 表示没有任何输入

S_1 ——表示序列以“0”结束

S_2 ——表示序列以“10”结束

S_3 ——表示序列以“010”结束, 此时输出标志 $Z=1$ 。



(1) 010检测的局部状态图

更复杂的同步时序设计_例9

(2) 子序列1001检测的状态设定

S₀——初始复位状态，表示没有任何输入

S₁——表示序列以“0”结束

S₂——表示序列以“10”结束

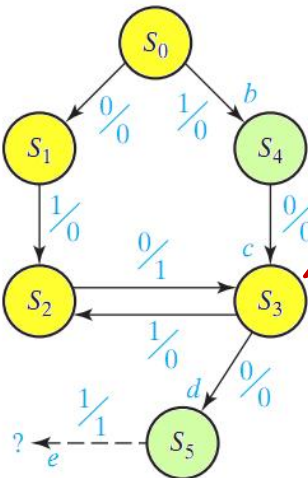
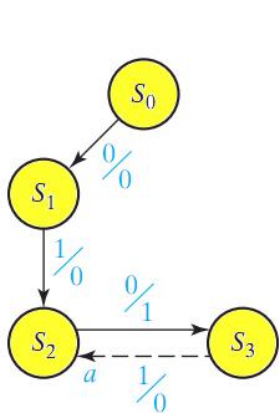
S_3 ——表示序列以“010”结束，此时输出标志 $Z=1$ 。

S₄——表示接收到1001序列的第一个“1”

S₅——表示序列以“100”结束。

重叠检测：010中的10可以被1001检测重

用



(2) 1001检测的局部状态图

重叠检测：010中的10可以被1001检测重

用

$$\begin{array}{cccccccccccccccccccc} X & = & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & \underline{1} & \underline{0} & 0 & 1 & 1 & 0 \\ & & & & & \uparrow & & \uparrow & & \uparrow & \uparrow & & & & \uparrow & & \uparrow & & \\ & & & & & a & & b & & c & d & & & & e & & f & & \\ Z & = & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{array}$$
$$\begin{array}{cccccccccccccccccccc}
X & = & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & \underline{1} & 0 & 0 & 1 & 1 & 0 \\
& & & & & \uparrow & & \uparrow & & \uparrow & \uparrow & & & \uparrow & & & \uparrow & & \\
& & & & & a & & b & & c & d & & & e & & & f & & \\
Z & = & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0
\end{array}$$

更复杂的同步时序设计_例9

(2) 子序列1001检测的状态设定

S_0 ——初始复位状态，表示没有任何输入

S_1 ——表示序列以“0”结束

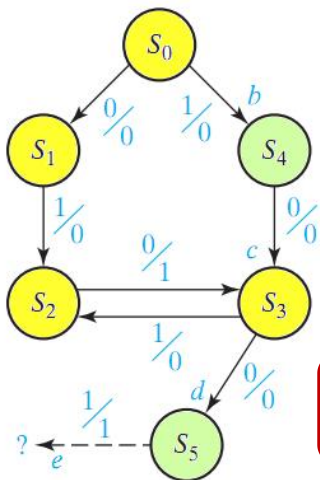
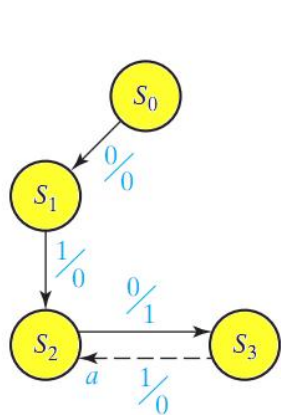
S_2 ——表示序列以“10”结束

S_3 ——表示序列以“010”结束，此时输出标志 $Z=1$ 。

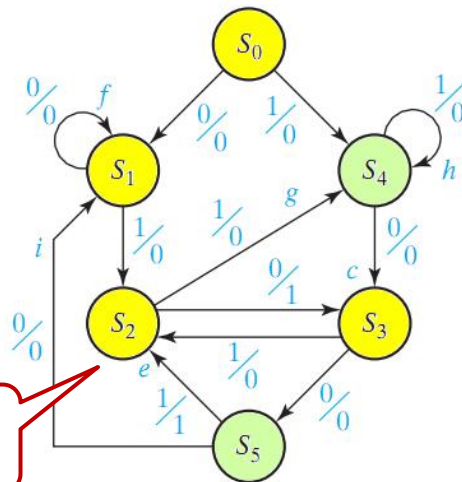
S_4 ——表示接收到1001序列的第一个“1”

S_5 ——表示序列以“100”结束。

重叠检测：010中的
10可以被1001检测重
用



重叠检测：1001中的
01可以被010检测重
用



2. 状态化简 (略)

3. 状态分配 (略)

4. 状态转换真值表 (略)

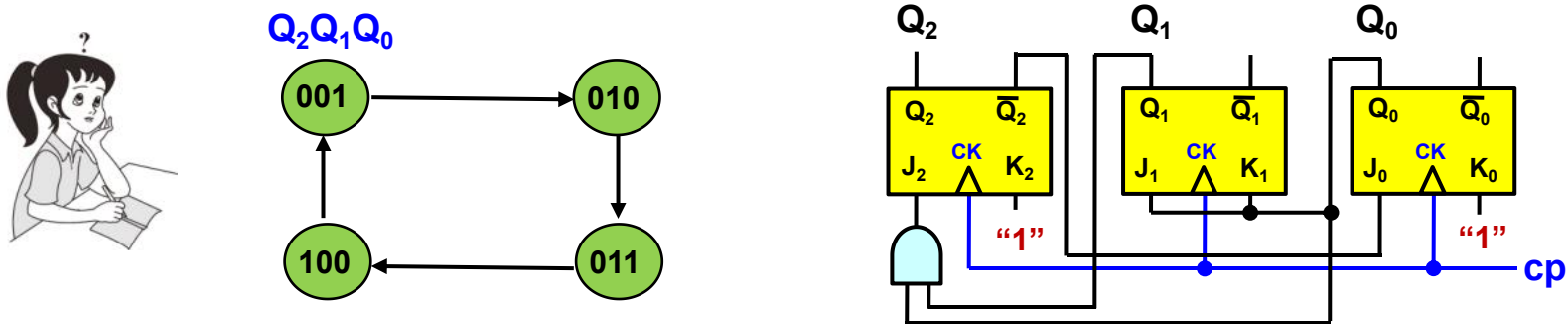
5. 卡诺图化简 (略)

6. 电路实现 (略)

(3) 010及1001检测的完整状态图

更复杂的同步时序设计_例10

例10: 某同步时序电路如下所示, 按图接线后, 试验得到如下的循环状态。经检查: 触发器工作正常, 试分析故障所在。



1. 获得正确状态图

① 输入方程

$$J_0 = \overline{Q_2}^n, K_0 = 1$$

$$J_1 = K_1 = Q_0^n$$

$$J_2 = Q_0^n Q_1^n, K_2 = 1$$

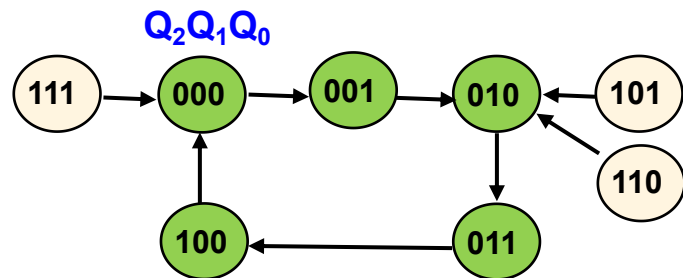
② 次态方程

$$Q_0^{n+1} = \overline{Q_0}^n \overline{Q_2}^n$$

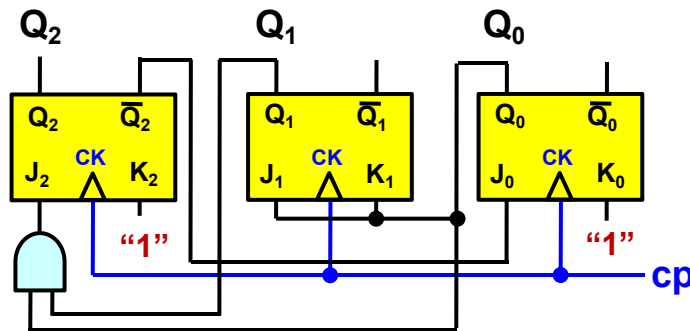
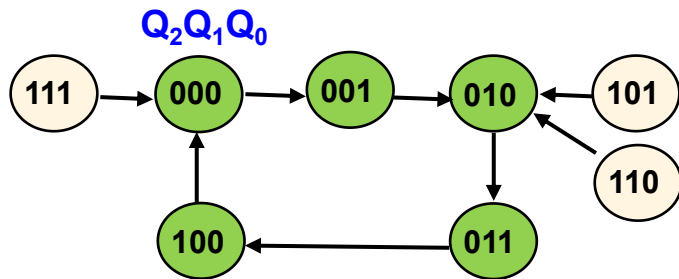
$$Q_1^{n+1} = Q_1^n \oplus Q_0^n$$

$$Q_2^{n+1} = Q_0^n Q_1^n \overline{Q_2}^n$$

③ 正确的状态转换图



更复杂的同步时序设计_例10



④ 电路功能：模5加法计数器，可自启动

2. 故障分析

① 触发器工作正常：说明——电源和地线接触良好、时钟信号CP正常送入
故障只可能在进位链或驱动回路中

② 分析各触发器状态：

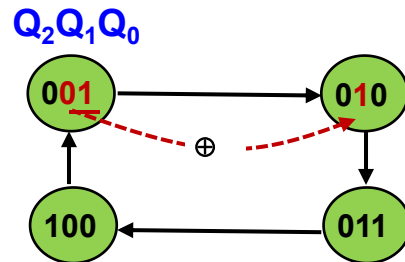
次态方程

$$Q_0^{n+1} = \overline{Q_0^n} \overline{Q_2^n}$$

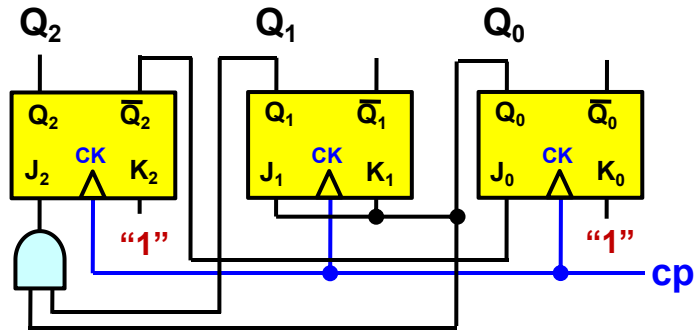
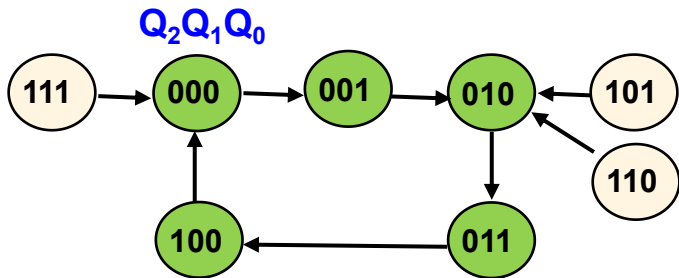
$$Q_1^{n+1} = Q_1^n \oplus Q_0^n$$

$$Q_2^{n+1} = Q_0^n Q_1^n \overline{Q_2^n}$$

触发器FF1
没有问题

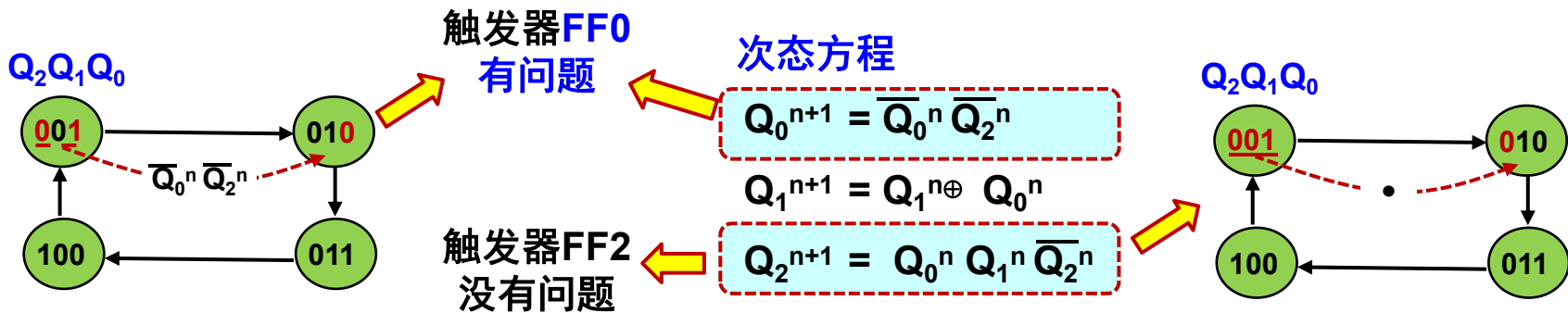


更复杂的同步时序设计_例10

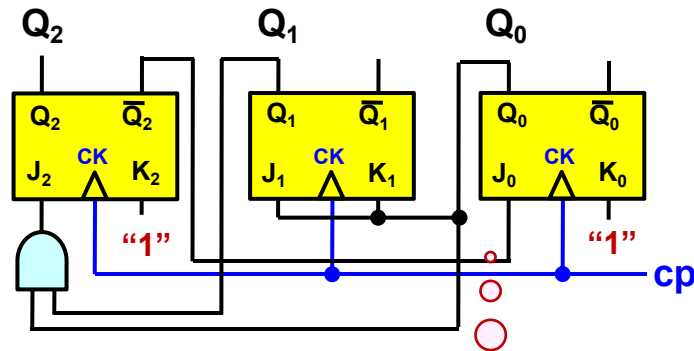
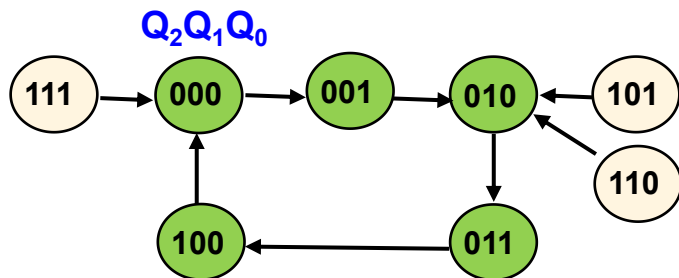


2. 故障分析

② 分析各触发器状态:



更复杂的同步时序设计_例10



2. 故障分析

③ 针对触发器0分析:



K_0 接触不良?

J_0 接触不良?

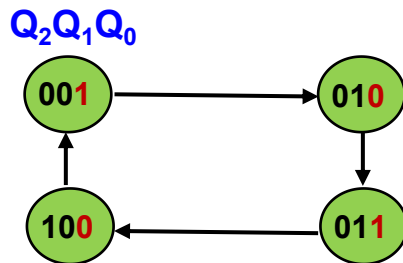
TTL电路管脚悬空
等效为高电平1

\bar{Q}_2 没有接入, J_0 悬空
等效为高电平1

K_0 没问题

触发器变成T',
符合故障现象

结论: \bar{Q}_2 没有
接入, J_0 悬空



讨论:某同步时序电路如下所示，在电路状态转换图保持不变的前提下，把电路中的JK触发器换成D触发器，应该怎样设计？如果将电路改成模8计数器，最简单的实现方法是什么？

