

数字逻辑设计

王鸿鹏

计算机科学与技术学院

wanghp@hit.edu.cn

利用触发器设计时序逻辑的方法

- (1) 根据需求——> 获得原始状态图、状态表
- (2) 最小化状态图、状态表
- (3) 状态编码（分配）——> 获得状态转移表
- (4) 状态转移表
触发器特征 } ——> 触发器激励表
- (5) 卡诺图化简——> { 激励（输入）函数表达式
输出函数表达式
- (6) 电路实现
- (7) 检查无关状态

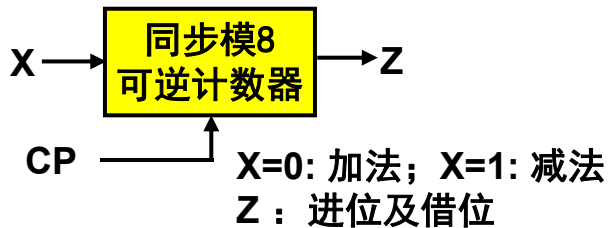
用触发器设计同步时序逻辑—实例

- 模8可逆计数器
- 时序锁
- 二进制串行加法器
- 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

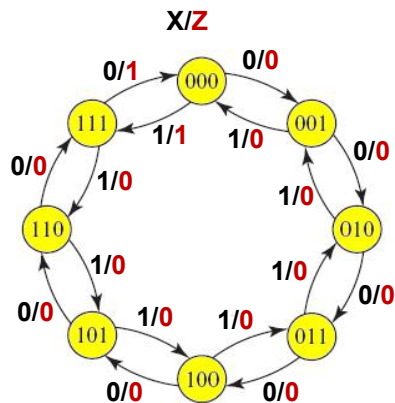
利用T触发器设计一个同步模8可逆计数器

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 | $\overline{Q_n}$ |

2. 状态转换真值表

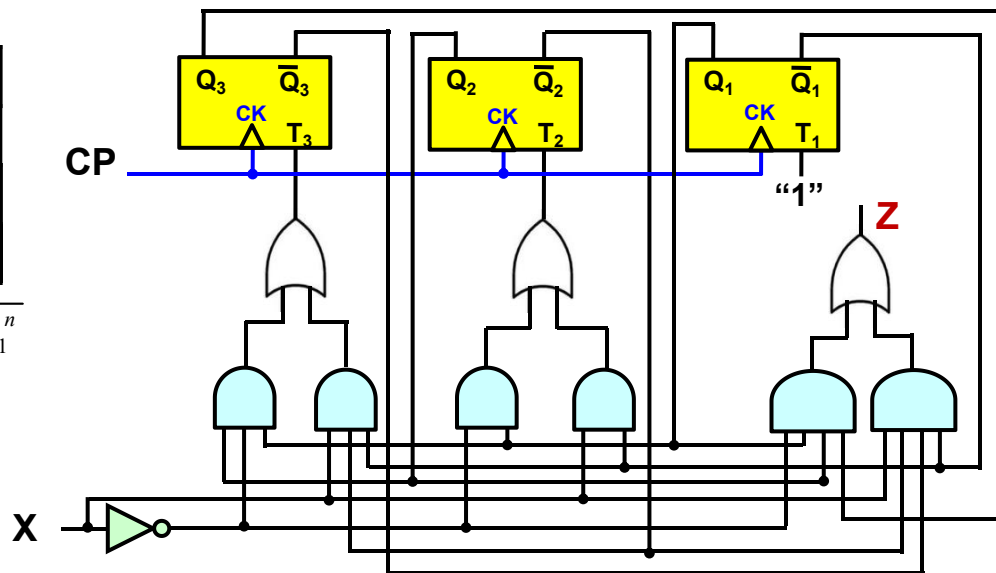
| 输入 现态 | | | | 次态 | | | 输入 输出 | | | |
|-------|---------|---------|---------|-------------|-------------|-------------|-------|-------|-------|---|
| X | Q_3^n | Q_2^n | Q_1^n | Q_3^{n+1} | Q_2^{n+1} | Q_1^{n+1} | T_3 | T_2 | T_1 | Z |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 |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

3. 卡诺图化简

$$T_3 = \overline{X} Q_2^n Q_l^n + X \overline{Q_2^n} \overline{Q_l^n}$$
$$T_2 = \overline{X} Q_1^n + X \overline{Q_1^n}$$

| | | | | | |
|----------|--|---------------|----|----|----|
| | | $Q_2^n Q_1^n$ | | | |
| XQ_3^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 |

4. 电路实现



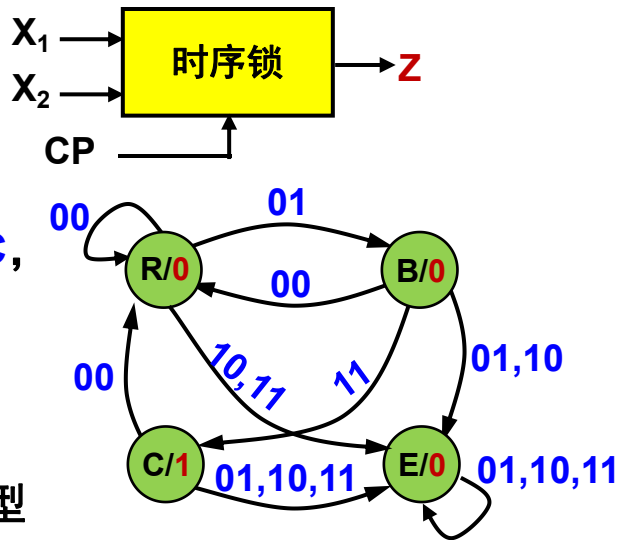
用触发器设计同步时序逻辑—实例

- 模8可逆计数器
- 时序锁
- 二进制串行加法器
- 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：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2

| 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 |

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 |

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 |

利用触发器设计同步时序逻辑_例2

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 |

$$Z = Q_2^nQ_1^n$$

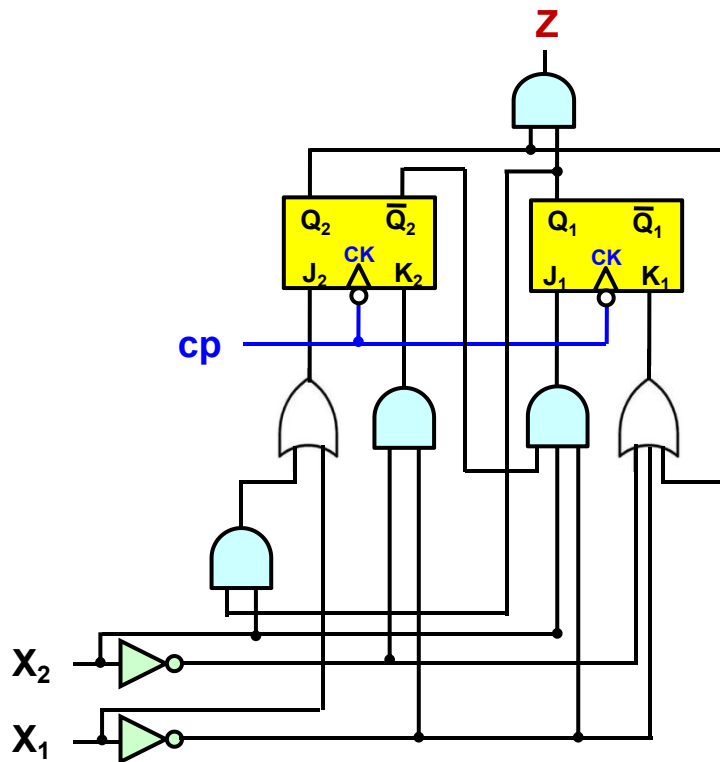
利用触发器设计同步时序逻辑_例2

6. 电路实现

$$\begin{cases} J_2 = X_2 Q_1^n + X_1 \\ K_2 = \overline{X_2} \overline{X_1} \\ J_1 = \overline{X_1} X_2 \overline{Q_2}^n \\ K_1 = Q_2^n + \overline{X_2} + \overline{X_1} \\ Z = Q_2^n Q_1^n \end{cases}$$

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

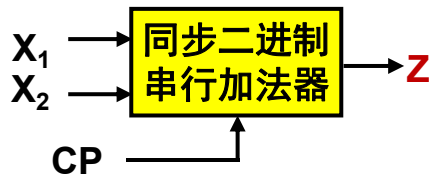


用触发器设计同步时序逻辑—实例

- 模8可逆计数器
- 时序锁
- 二进制串行加法器
- 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：利用JK触发器设计一个同步二进制串行加法器



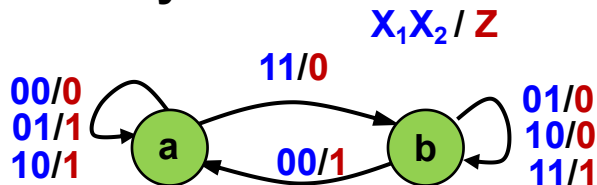
1. 原始状态图及状态表

① 设加法器内部状态

a—— 无进位

b—— 有进位

② Mealy 状态图



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 |

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3

5. 卡诺图化简

| | | | | |
|----------------------|----|----|----|----|
| $X_2 \backslash X_1$ | 00 | 01 | 11 | 10 |
| 0 | 0 | X | X | 0 |
| 1 | 0 | X | X | 1 |

$$\mathbf{J} = \mathbf{X}_1 \mathbf{X}_2$$

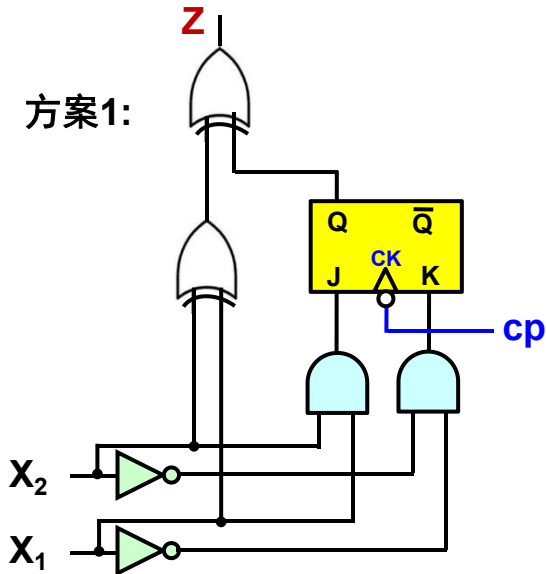
| | | | | | |
|-------|--|-----------|----|----|----|
| | | $X_2 Q^n$ | | | |
| X_1 | | 00 | 01 | 11 | 10 |
| 0 | | X | 1 | 0 | X |
| 1 | | X | 0 | 0 | X |

$$\mathbf{K} = \overline{\mathbf{X}}_1 \overline{\mathbf{X}}_2$$

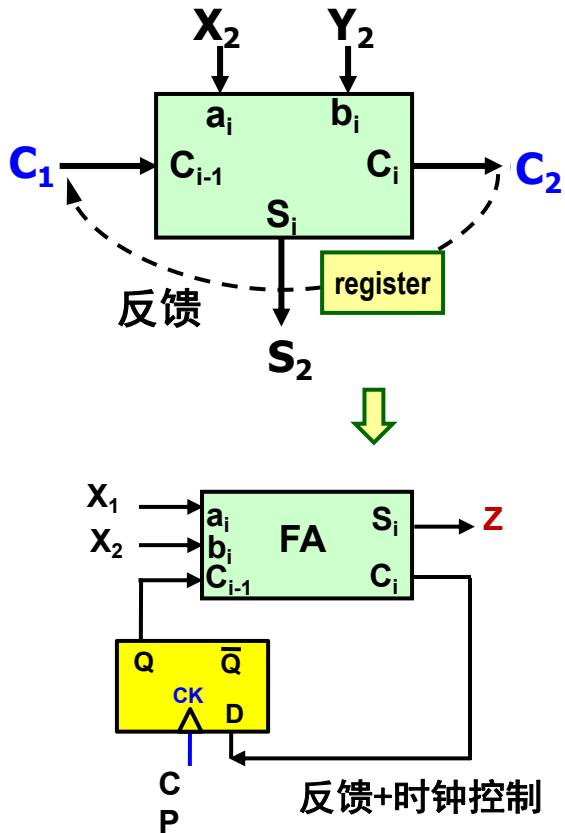
| $X_1 \backslash X_2 Q^n$ | 00 | 01 | 11 | 10 |
|--------------------------|----|----|----|----|
| 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 |

$$Z = X_1 \oplus X_2 \oplus \dots \oplus X_n$$

6. 电路实现



方案2: 如何用一位全加器实现?

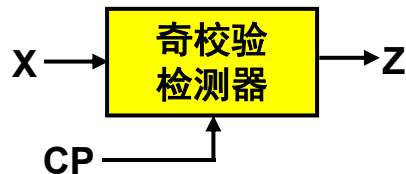


利用触发器设计时序逻辑——实例

- 模8可逆计数器
- 时序锁
- 二进制串行加法器
- 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：利用T触发器设计一个串行输入的奇校验检测器



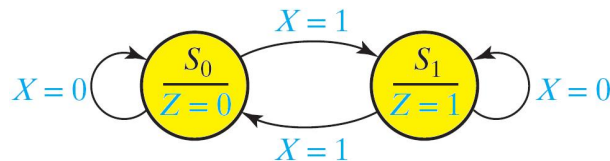
1. 原始状态图及状态表

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

$S_0: 0; S_1: 1$

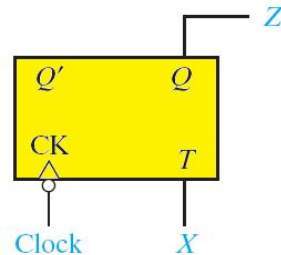
4. 状态转换真值表

| 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
- 时序锁
- 二进制串行加法器
- 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5

例：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5

- 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 |

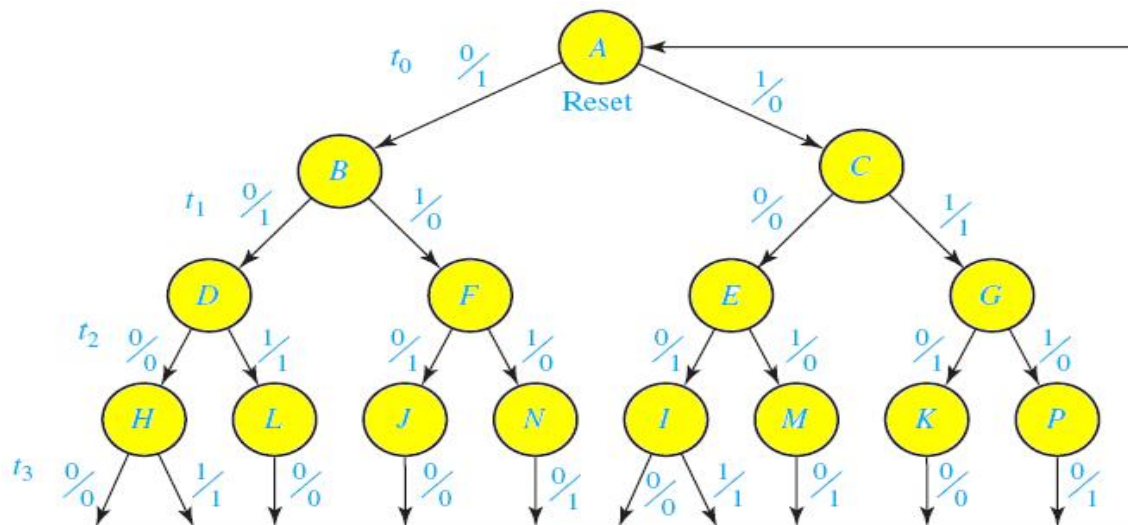
更复杂的同步时序设计_例5

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5

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5

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 | |
|-------------|-----|-------------------|---------|---------|---------|
| $Q_1Q_2Q_3$ | | $X = 0$ | $X = 1$ | $X = 0$ | $X = 1$ |
| 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 | xxx | 1 | x |
| – | 001 | xxx | xxx | x | x |



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



更复杂的同步时序设计_例5

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 |

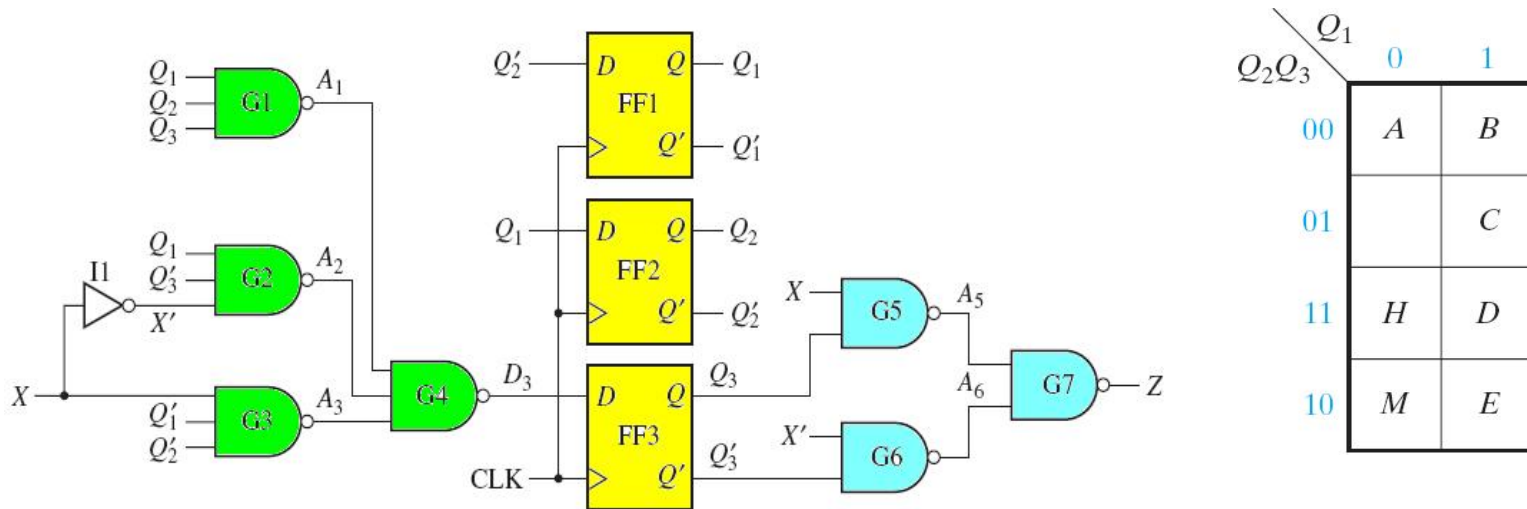
$$D_3 = Q_3^+ = Q_1 Q_2 Q_3 + X' Q_1 Q_3' + X Q_1' Q_2'$$

| $Q_2 Q_3$ | 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5

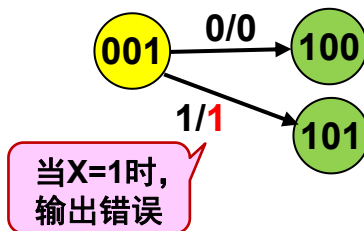
6. 电路实现



7. 无关项检查

将无关状态 $Q_1Q_2Q_3=001$ 代入次态方程和输出方程计算

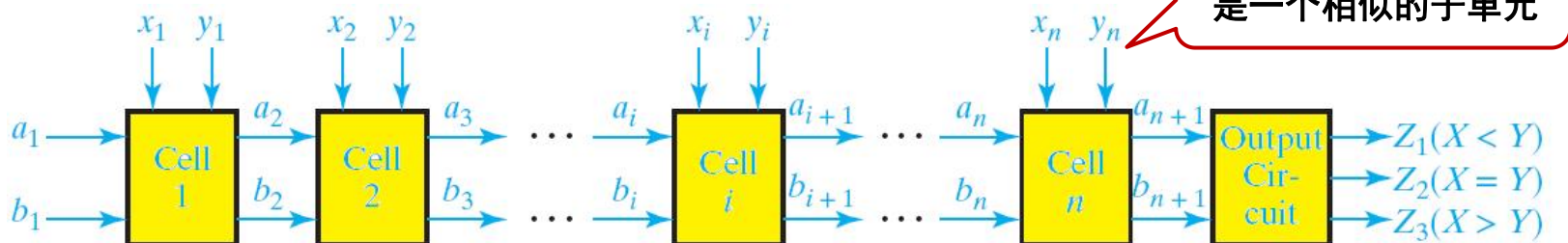
$$\begin{cases} D_1 = Q_1^+ = Q_2' \\ D_2 = Q_2^+ = Q_1 \\ D_3 = Q_3^+ = Q_1Q_2Q_3 + X'Q_1Q_3' + XQ_1'Q_2' \\ Z = X'Q_3' + XQ_3 \end{cases}$$



电路可以自启动

更复杂的同步时序设计——例6

例：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6

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6

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 |

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

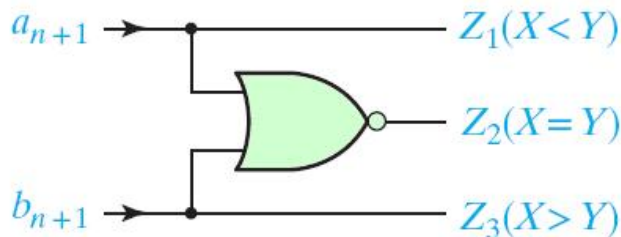
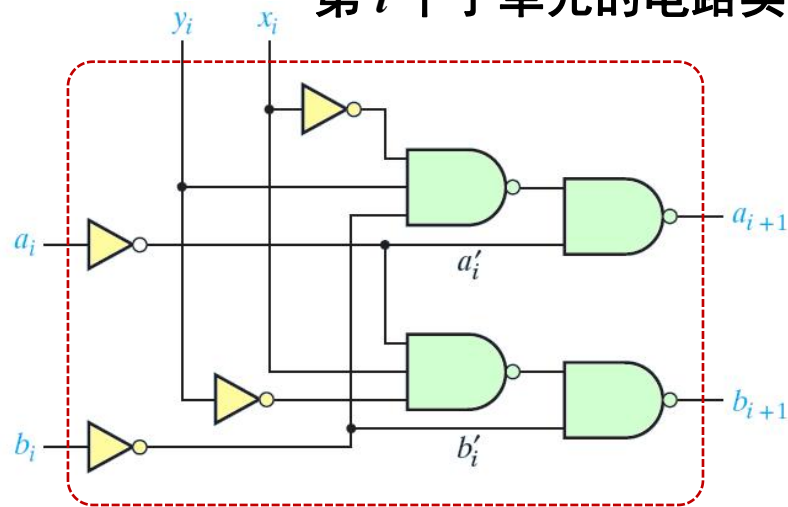
| a_{n+1} | | 0 | 1 |
|-----------|---|---|---|
| b_{n+1} | 0 | 1 | |
| | 1 | | X |

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

| 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

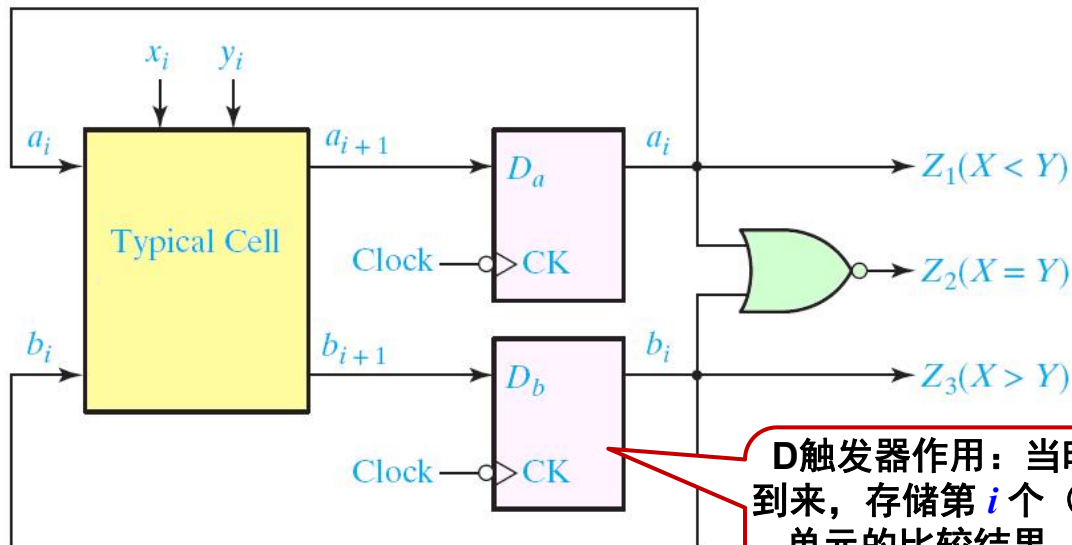


更复杂的同步时序设计_例6

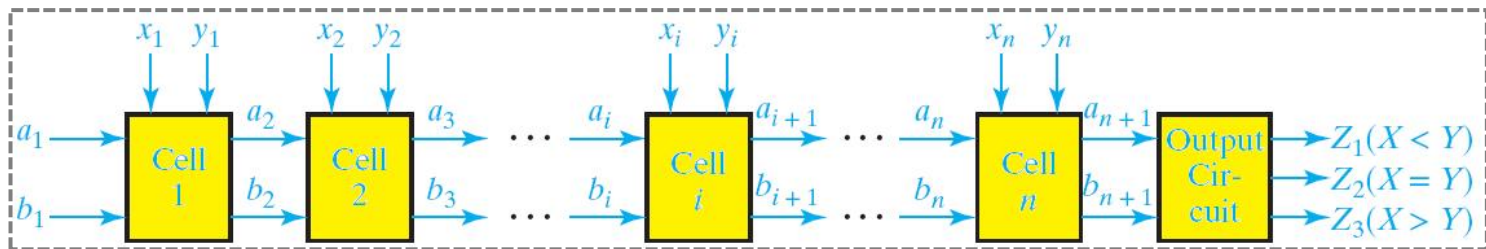
6. 电路实现

7. 无关项检查

(略)

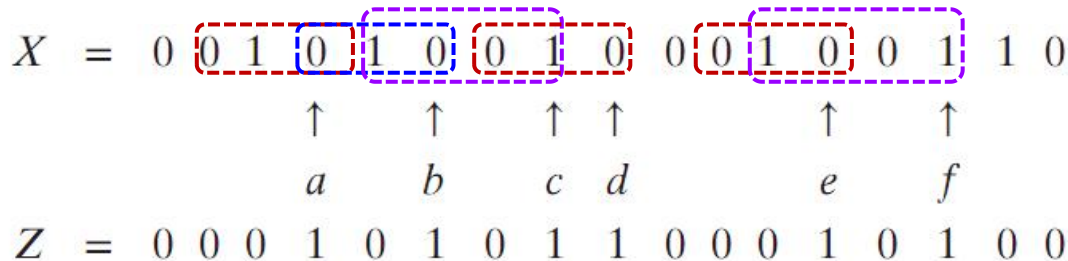


D触发器作用：当时钟信号到来，存储第 i 个（前一个）单元的比较结果，并产生第 $i+1$ 个单元的比较结果



更复杂的同步时序设计_例7

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

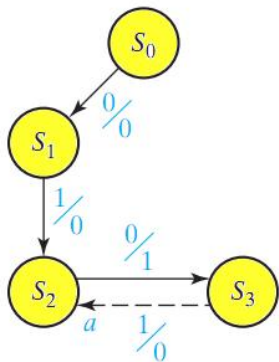
(1) 子序列010检测的状态设定

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7

(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

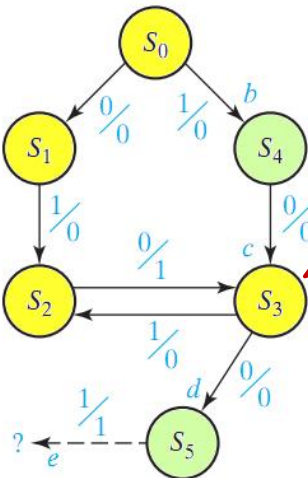
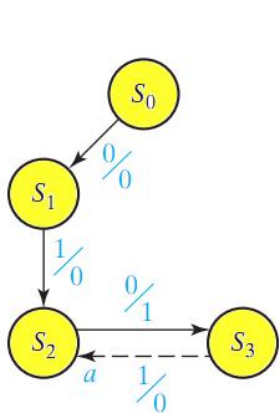
S₂——表示序列以“10”结束

S₃——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

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

更复杂的同步时序设计_例7

(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

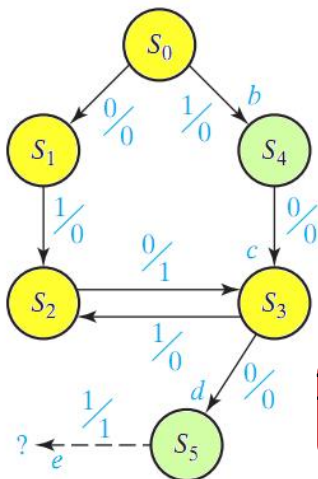
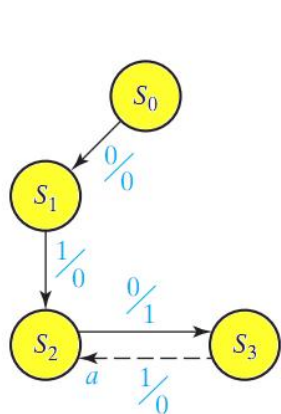
S_2 ——表示序列以“10”结束

S_3 ——表示序列以“010”结束，此时输出标志 $Z=1$ 。

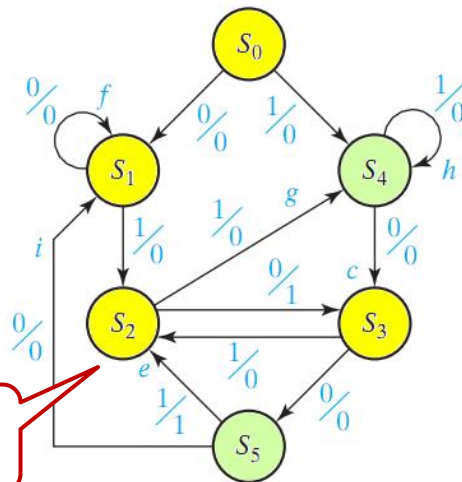
S_4 ——表示接收到1001序列的第一个“1”

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)

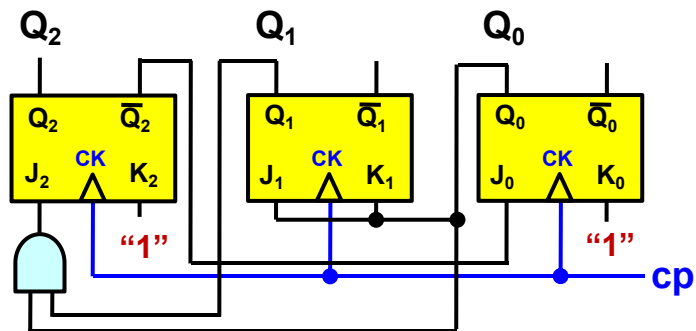
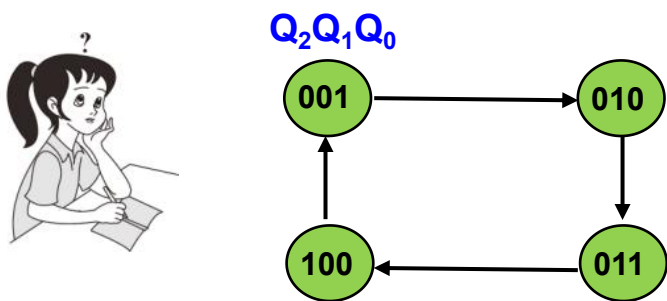
5. 卡诺图化简 (略)

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8

例: 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

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

$$J_1 = K_1 = Q_0^n$$

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

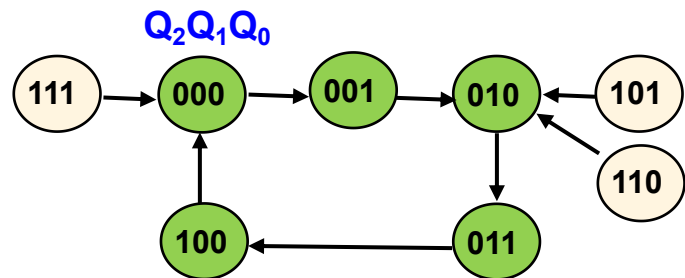
② 次态方程

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

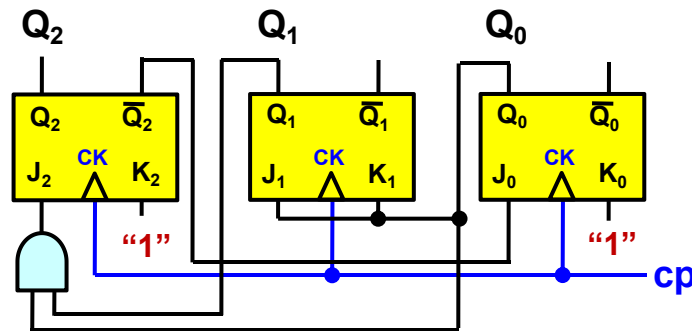
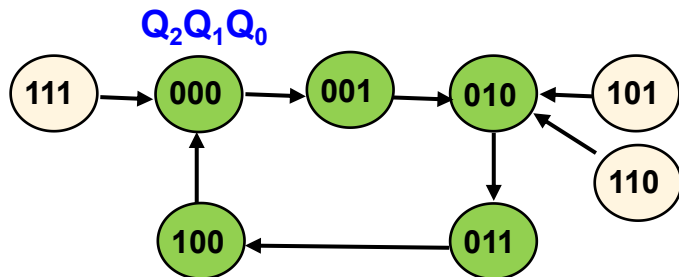
$$Q_1^{n+1} = Q_1^n \oplus Q_0^n$$

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

③ 正确的状态转换图



更复杂的同步时序设计_例8



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：

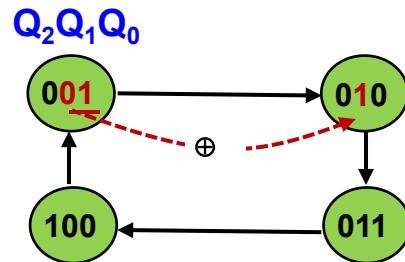
次态方程

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

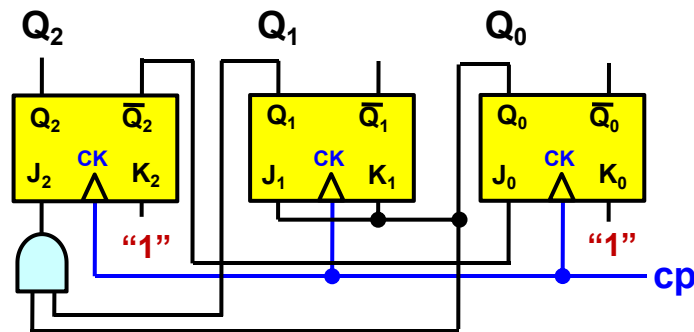
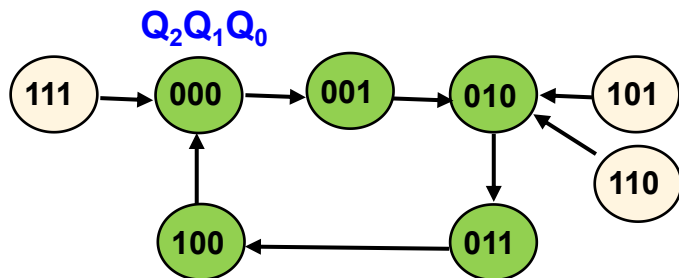
$$Q_1^{n+1} = Q_1^n \oplus Q_0^n$$

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

触发器FF1
没有问题

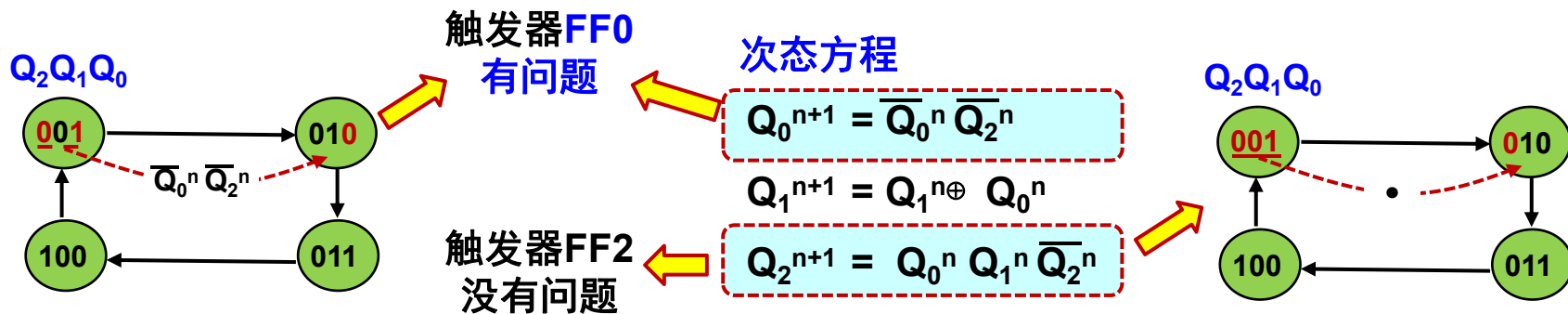


更复杂的同步时序设计_例8

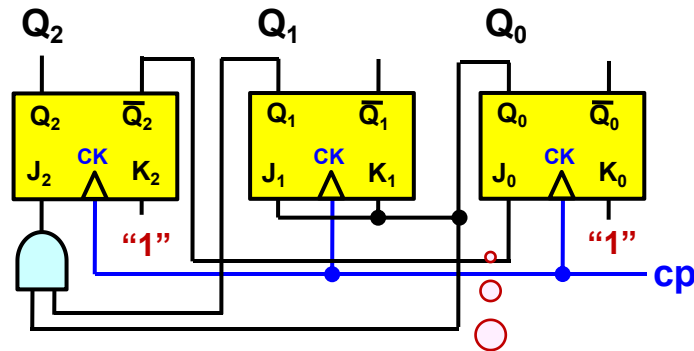
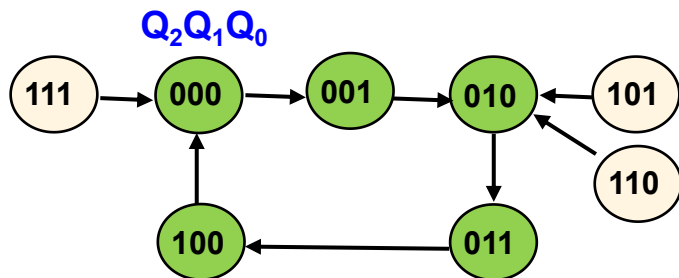


2. 故障分析

② 分析各触发器状态:



更复杂的同步时序设计_例8



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?

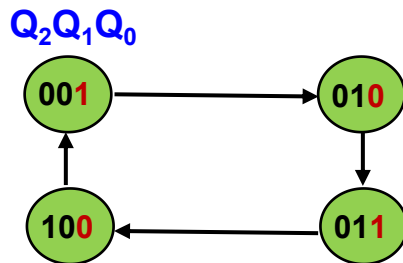
TTL电路管脚悬空
等效为高电平1

\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?

