

《数字逻辑》

(第六讲)

厦门大学信息学院软件工程系 曾文华

2024年11月4日

课程内容

- 全书共9章：

第1章 基本知识

第2章 逻辑代数基础

第3章 集成门电路与触发器

第4章 组合逻辑电路

第5章 同步时序逻辑电路

第6章 异步时序逻辑电路

第7章 中规模通用集成电路及其应用

第8章 可编程逻辑器件

第9章 综合应用举例



第6章 异步时序逻辑电路

- 6.1 异步时序逻辑电路的特点与分类
- 6.2 脉冲异步时序逻辑电路
- 6.3 电平异步时序逻辑电路

- 时序逻辑电路：

- 同步时序逻辑电路（第5章）：该电路中的存储电路由带有时钟控制端的触发器组成，各触发器的时钟端均与统一的时钟脉冲信号（CP）相连接，各触发器状态的改变受到同一时钟信号的控制。
- 异步时序逻辑电路（第6章）：该电路中的存储电路由触发器或延时元件组成，电路中无统一的时钟信号同步，电路输入信号的变化将直接导致电路状态的变化。

- 异步时序逻辑电路：

- 脉冲异步时序逻辑电路：电路的输入信号为脉冲信号。
- 电平异步时序逻辑电路：电路的输入信号为电平信号。

6.1 异步时序逻辑电路的特点与分类

- 异步时序逻辑电路的**特点**:
 - ① 电路中**没有统一的时钟脉冲信号同步**，电路状态的改变是外部输入信号变化直接作用的结果。
 - ② 在状态转移过程中，各存储元件的状态变化不一定发生在同一时刻，不同状态的维持时间不一定相同，并且可能出现**非稳定状态**。
 - ③ 在研究异步时序逻辑电路时，**对输入信号**的形式有所区分，无论输入信号是脉冲信号还是电平信号，对其变化过程均**有一定约束**。
- 根据**输入信号形式**，异步时序逻辑电路**分为**:
 - ① **脉冲异步时序逻辑电路**：其存储电路由**触发器**组成（可以是钟控触发器，也可以是非钟控触发器），电路输入信号为脉冲信号。
 - ② **电平异步时序逻辑电路**：其存储电路由**延迟元件**组成（可以是专用的延迟元件，或者利用电路本身固有的延迟），通过延迟加反馈实现记忆功能，电路输入信号为电平信号。
- 根据**电路结构模型**，异步时序逻辑电路**分为**:
 - ① **Mealy型**异步时序逻辑电路。
 - ② **Moore型**异步时序逻辑电路。

6.2 脉冲异步时序逻辑电路

6.2.1 脉冲异步时序逻辑电路的结构模型

6.2.2 脉冲异步时序逻辑电路的分析

6.2.3 脉冲异步时序逻辑电路的设计

• 6.2.1 脉冲异步时序逻辑电路的结构模型

— 图6.1：脉冲异步时序逻辑电路的结构模型。

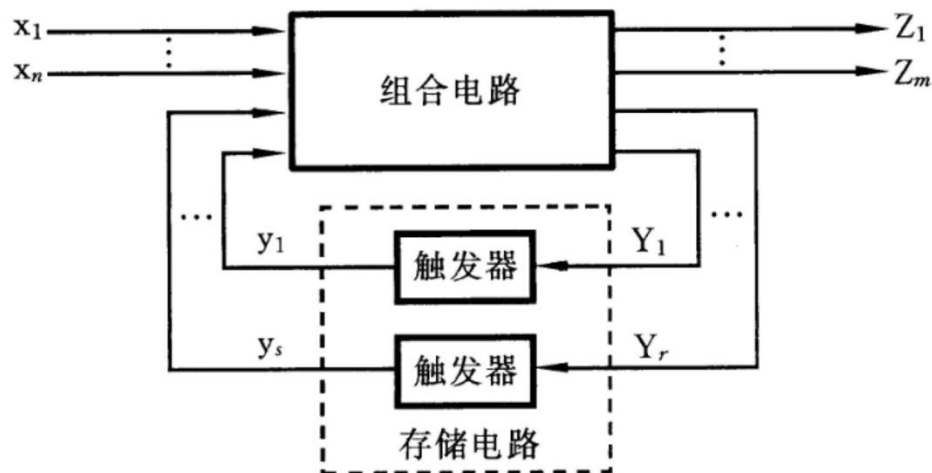


图 6.1 脉冲异步时序电路的结构模型

- 输入脉冲信号（即输入信号 $x_1 \sim x_n$ ）必须满足如下的**约束条件**：
 - ① 输入脉冲的宽度，必须保证触发器可靠翻转。
 - ② 输入脉冲的间隔，必须保证前一个脉冲引起的电路响应完全结束后，后一个脉冲才能到来。
 - ③ **不允许在两个或两个以上输入端同时出现脉冲**（这可能导致电路产生错误的状态转移）。

- **Mealy型**脉冲异步时序逻辑电路的**输出**通常是**脉冲信号**（因为输出是输入和状态的函数，而输入是脉冲信号，因此输出也会是脉冲信号）。

- **Moore型**脉冲异步时序逻辑电路的**输出**是**电平信号**（因为输出仅仅是状态函数，因此输出是电平信号）。

• 6.2.2 脉冲异步时序逻辑电路的分析

– 分析过程与同步时序逻辑电路（第5章）基本相同，**分析步骤**如下：

① 根据电路，写出**输出函数**和**激励函数**表达式。

② 根据激励函数以及触发器的功能表，列出**次态真值表**（或**次态方程组**）。

③ 根据次态真值表（或次态方程组）以及输出函数，作出**状态表**和**状态图**。

④ 拟定一典型输入序列，画出**时间图**，并用文字描述电路的逻辑功能。

– 脉冲异步时序逻辑电路分析与同步时序逻辑电路分析的**区别**：

① 当存储元件采用钟控触发器时，应将触发器的**时钟控制端**作为**激励函数**处理。

② 由于**不允许两个或两个以上输入端同时出现脉冲**，加之输入端无脉冲出现时电路状态不会发生变化。因此，分析时可以排除这些情况，从而使分析过程中使用的图、表简化。

— **例6.1**：分析图6.2所示脉冲异步时序逻辑电路，指出该电路的功能。

— 解：

• 该电路属于**Mealy型**脉冲异步时序逻辑电路。

• (1) 根据电路图，写出**输出函数和激励函数**表达式：

- $Z = x \cdot y_2 \cdot y_1$
- $J_2 = K_2 = 1 \quad C_2 = y_1$
- $J_1 = K_1 = 1 \quad C_1 = x$

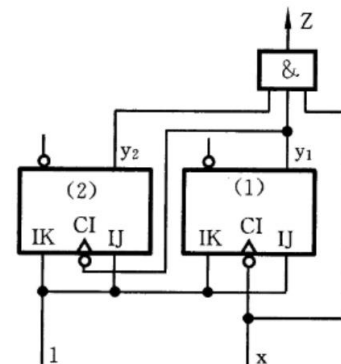


图 6.2 逻辑电路

• (2) 根据激励函数和J-K触发器的功能表，列出**次态真值表**（表6.1）：

- 因为电路中的J-K触发器为下降沿触发， $x=0$ 时输入端无脉冲出现，因此表6.1中不需要给出 $x=0$ 的情况（只有4种情况）。
- 表6.1中 \downarrow 表示触发器的时钟端出现**下降沿**。

表 3.23 钟控 J-K 触发器功能表

| J | K | Q^{n+1} | 功能说明 |
|---|---|-----------|------|
| 0 | 0 | Q | 不变 |
| 0 | 1 | 0 | 置 0 |
| 1 | 0 | 1 | 置 1 |
| 1 | 1 | \bar{Q} | 翻转 |

表 6.1 次态真值表

| 输 入 x | 现 态 $y_2 \quad y_1$ | | 激励函数 | | | | | | 次 态 $y_2^{n+1} \quad y_1^{n+1}$ | |
|----------|------------------------|-------|-------|-------|-------|--------------|--|--|------------------------------------|---|
| | J_2 | K_2 | C_2 | J_1 | K_1 | C_1 | | | | |
| 1 | 0 | 0 | | 1 | 1 | \downarrow | | | 0 | 1 |
| 1 | 0 | 1 | | 1 | 1 | \downarrow | | | 1 | 0 |
| 1 | 1 | 0 | | 1 | 1 | \downarrow | | | 1 | 1 |
| 1 | 1 | 1 | | 1 | 1 | \downarrow | | | 0 | 0 |

前3列直接给出

中间6列由激励函数得到

最后2列由触发器的功能表得到

- (3) 根据次态真值表和输出函数，作出状态表（表6.2），根据状态表画出状态图（图6.3）。

表 6.1 次态真值表

| 输 入 x | 现 态 | | 激励函数 | | | | | | | 次 态 | |
|----------|-------|-------|-------|-------|-------|-------|-------|-------|-------------|-------------|--|
| | y_2 | y_1 | J_2 | K_2 | C_2 | J_1 | K_1 | C_1 | y_2^{n+1} | y_1^{n+1} | |
| 1 | 0 | 0 | 1 | 1 | | 1 | 1 | 1 | 0 | 1 | |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | |
| 1 | 1 | 0 | 1 | 1 | | 1 | 1 | 1 | 1 | 1 | |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | |

表 6.2 状态表

| 现 态 | | 次态 $y_2^{n+1}y_1^{n+1}$ / 输出 Z | |
|-------|-------|--------------------------------|-----|
| y_2 | y_1 | x=1 | |
| 0 | 0 | 0 1 | / 0 |
| 0 | 1 | 1 0 | / 0 |
| 1 | 0 | 1 1 | / 0 |
| 1 | 1 | 0 0 | / 1 |

状态表是次态真值表的简化版

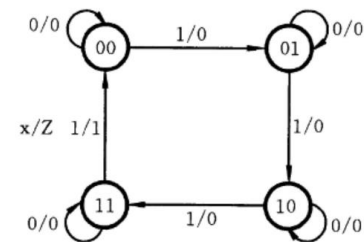


图 6.3 状态图

- (4) 画出在某种典型输入信号下的时间图（图6.4）。根据时间图可知，当收到第4个输入脉冲时，电路产生一个进位输出脉冲。因此，该电路是一个“模4加1计数器”。

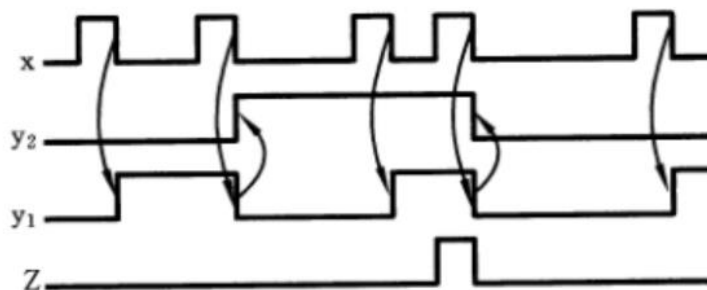


图 6.4 时间图

— **例6.2：**分析图6.5所示的脉冲异步时序逻辑电路。

— 解：

- 该电路属于**Moore型**脉冲异步时序逻辑电路。

- (1) 根据电路图，写出**输出函数和激励函数**表达式：

$$\begin{aligned} Z &= (y_2 + y_1) = y_2 \cdot y_1 \\ R_2 &= (x_3 + x_2 \cdot y_1) & S_2 &= x_1 \\ R_1 &= (x_1 + x_3 \cdot y_2 + x_2 \cdot y_1) & S_1 &= (x_2 \cdot y_2 \cdot y_1) \end{aligned}$$

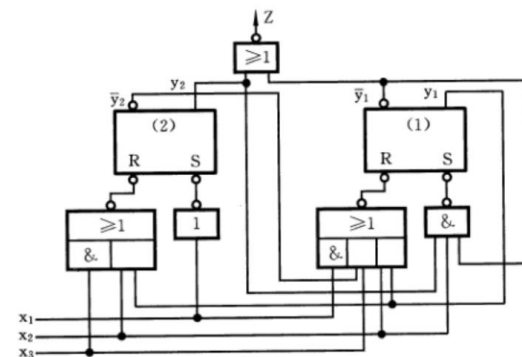


图 6.5 逻辑电路

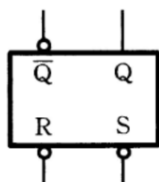
- (2) 根据激励函数和与非门构成的基本R-S触发器功能表（表3.13），列出**次态真值表**（表6.3）：
 - 表6.3中只需要列出输入分别为1的情况（ $x_1x_2x_3=100、010、001$ ），每种情况又有4种现态变化的情况（ $y_2y_1=00、01、10、11$ ），因此共12种情况。

表 6.3 次态真值表

| 输 入 | | | 现 态 | | 激励函数 | | | | 次 态 | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------------|-------------|
| x_1 | x_2 | x_3 | y_2 | y_1 | R_2 | S_2 | R_1 | S_1 | y_2^{n+1} | y_1^{n+1} |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |

表 3.13 与非门构成的基本R-S 触发器功能表

| R | S | Q^{n+1} | 功能说明 |
|---|---|-----------|------|
| 0 | 0 | d | 不定 |
| 0 | 1 | 0 | 置 0 |
| 1 | 0 | 1 | 置 1 |
| 1 | 1 | Q | 不变 |



- (3) 根据次态真值表和输出函数，作出状态表（表6.4），根据状态表画出状态图（图6.6）。

表 6.3 次态真值表

| 输 入 | | | 现 态 | | 激励函数 | | | | 次 态 | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------------|-------------|
| x_1 | x_2 | x_3 | y_2 | y_1 | R_2 | S_2 | R_1 | S_1 | y_2^{n+1} | y_1^{n+1} |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |

表 6.4 状态表

| 现 态 | | 次态 $y_2^{n+1} y_1^{n+1}$ | | | 输出 |
|-------|-------|--------------------------|-------|-------|----|
| y_2 | y_1 | x_1 | x_2 | x_3 | |
| 0 | 0 | 10 | 00 | 00 | 0 |
| 0 | 1 | 10 | 00 | 00 | 1 |
| 1 | 0 | 10 | 11 | 00 | 0 |
| 1 | 1 | 10 | 00 | 01 | 0 |

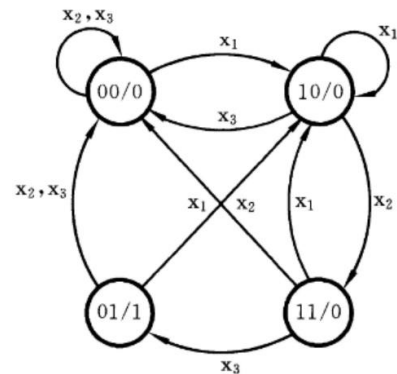


图 6.6 状态图

- (4) 画出在某种典型输入信号下的时间图（图6.7）。根据时间图可知，当输入端按照 x_1 、 x_2 、 x_3 的顺序依次出现脉冲时，将产生一个“1”输出信号，其它情况输出为“0”。因此，该电路是一个“ $x_1 - x_2 - x_3$ ”序列检测器。

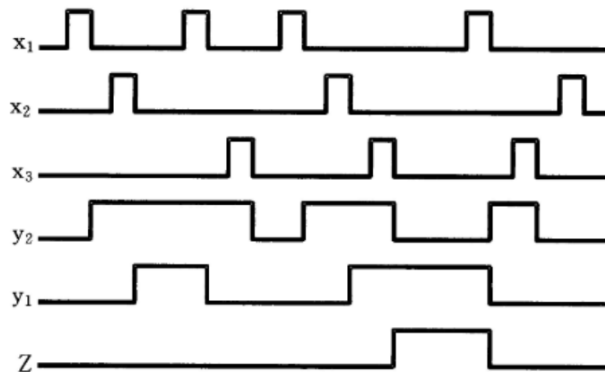


图 6.7 时间图

• 6.2.3 脉冲异步时序逻辑电路的设计

- 脉冲异步时序逻辑电路的设计过程与同步时序逻辑电路（第5章）基本相同，**设计步骤**如下：
 - ① 根据设计要求形成**原始状态图**和**原始状态表**。
 - ② 对原始状态表进行化简，得到**最简状态表**（最小状态表）。
 - ③ 对最简状态表进行状态编码，得到**二进制状态表**。
 - ④ 确定触发器的类型和数目。
 - ⑤ 根据二进制状态表和触发器的激励表，得到激励函数和输出函数**真值表**；由真值表，画**卡诺图**；由卡诺图，得到**激励函数和输出函数**的最简表达式。
 - ⑥ 根据激励函数和输出函数，画出**电路图**。
- 脉冲异步时序逻辑电路设计与同步时序逻辑电路设计**不同的地方**：
 - ① 由于不允许两个或两个以上输入端同时为1，所以在形成原始状态图和原始状态表时，若有多个输入信号，则**只需考虑多个输入信号中仅一个为1的情况**，从而使问题的描述得以简化。此外，在确定激励函数和输出函数时，可将两个或两个以上输入同时为1的情况，作为无关条件处理，这将有利于函数的简化。
 - ② 由于电路中没有统一的时钟脉冲，因此当存储电路采用钟控触发器时，**触发器的时钟端是作为激励函数处理的**，因此需要对触发器的激励表做适当的修改（见表6.5至表6.8）。

- 由表6.5至表6.8可知，在要求触发器状态保持不变时（0→0，或1→1），有两种不同的处理方法：
 - 方法一：CP=d，输入端根据需要取值。
 - 方法二：CP=0（触发器状态不变），输入端=d。
- 通常采用方法二（CP=0，输入端=d）。

表 6.5 D 触发器激励表

| Q | Q ⁿ⁺¹ | CP | D |
|---|------------------|----|---|
| 0 | 0 | d | 0 |
| | | 0 | d |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | d | 1 |
| | | 0 | d |

表 6.6 J-K 触发器激励表

| Q | Q ⁿ⁺¹ | CP | J | K |
|---|------------------|----|---|---|
| 0 | 0 | d | 0 | d |
| | | 0 | d | d |
| 0 | 1 | 1 | 1 | d |
| 1 | 0 | 1 | d | 1 |
| 1 | 1 | d | d | 0 |
| | | 0 | d | d |

表 6.7 T 触发器激励表

| Q | Q ⁿ⁺¹ | CP | T |
|---|------------------|----|---|
| 0 | 0 | d | 0 |
| | | 0 | d |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | d | 0 |
| | | 0 | d |

表 6.8 R-S 触发器激励表

| Q | Q ⁿ⁺¹ | CP | R | S |
|---|------------------|----|---|---|
| 0 | 0 | d | d | 0 |
| | | 0 | d | d |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | d | 0 | d |
| | | 0 | d | d |

- **例6.3：**用D触发器作为存储元件，设计一个“ $x_1 - x_2 - x_2$ ”序列检测器。该电路有两个输入（ x_1 、 x_2 ），一个输出Z。仅当 x_1 输入一个脉冲后， x_2 连续输入两个脉冲时，输出端Z由0变为1，该1信号将一直维持到输入端 x_1 或 x_2 再出现脉冲时，才由1变为0。其输入/输出时间图如图6.8所示。

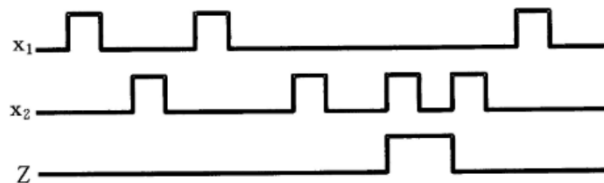


图 6.8 时间图

- 解：
- （假设）该电路属于Moore型脉冲异步时序逻辑电路。
 - （1）作出状态图（图6.9）和状态表（表6.9）。

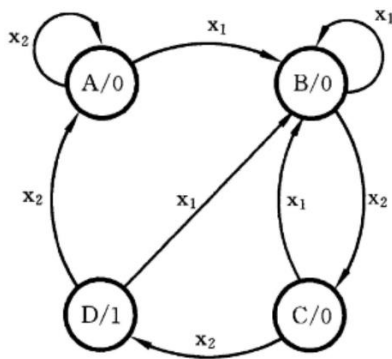


图 6.9 状态图

表 6.9 状态表

| 现 态 | 次 态 | | 输 出 Z |
|-----|-------|-------|----------|
| | x_1 | x_2 | |
| A | B | A | 0 |
| B | B | C | 0 |
| C | B | D | 0 |
| D | B | A | 1 |

- (3) 状态编码。根据相邻编码法的原则，可以得到表6.10的**编码方案**，从而得到**二进制状态表**（表6.11）。

| | 现 态 | | 次 态 $y_2^{n+1} y_1^{n+1}$ | | 输 出 |
|----------|-------|-------|---------------------------|---------|-----|
| | y_2 | y_1 | $x_1=1$ | $x_2=1$ | Z |
| A | 0 | 0 | 10 | 00 | 0 |
| C | 0 | 1 | 10 | 11 | 0 |
| B | 1 | 0 | 10 | 01 | 0 |
| D | 1 | 1 | 10 | 00 | 1 |

- (4) 确定输出函数和激励函数。采用2个D触发器作为存储单元，根据二进制状态表（表6.11）和D触发器的激励表（表6.5，状态不变时采用方法二： $CP=0$ ， $D=d$ ），得到激励函数和输出函数真值表（表6.12，输入只需要考虑两种情况：01、10）。

| 输 入 | | 现 态 | | 激励函数 | | | | 输出函数 |
|-------|-------|-------|-------|-------|-------|-------|-------|------|
| x_2 | x_1 | y_2 | y_1 | C_2 | D_2 | C_1 | D_1 | Z |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | d | 0 |
| | | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| | | 1 | 0 | 0 | d | 0 | d | 0 |
| | | 1 | 1 | 0 | d | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | d | 0 | d | 0 |
| | | 0 | 1 | 1 | 1 | 0 | d | 0 |
| | | 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |

- 由真值表（表6.12）画出卡诺图（图6.10，当输入 $x_2x_1=00$ 时，置 $C_2=d$ 、 $C_1=d$ 、 $D_2=d$ 、 $D_1=d$ 、 $Z=d$ ；当输入 $x_2x_1=11$ 时，置 $C_2=d$ 、 $C_1=d$ 、 $D_2=d$ 、 $D_1=d$ 、 $Z=d$ ）。

表 6.12 激励函数和输出函数真值表

| 输入 | | 现 态 | | 激励函数 | | | | 输出函数 |
|-------|-------|-------|-------|-------|-------|-------|-------|------|
| x_2 | x_1 | y_2 | y_1 | C_2 | D_2 | C_1 | D_1 | Z |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | d | 0 |
| | | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| | | 1 | 0 | 0 | d | 0 | d | 0 |
| | | 1 | 1 | 0 | d | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | d | 0 | d | 0 |
| | | 0 | 1 | 1 | 1 | 0 | d | 0 |
| | | 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |

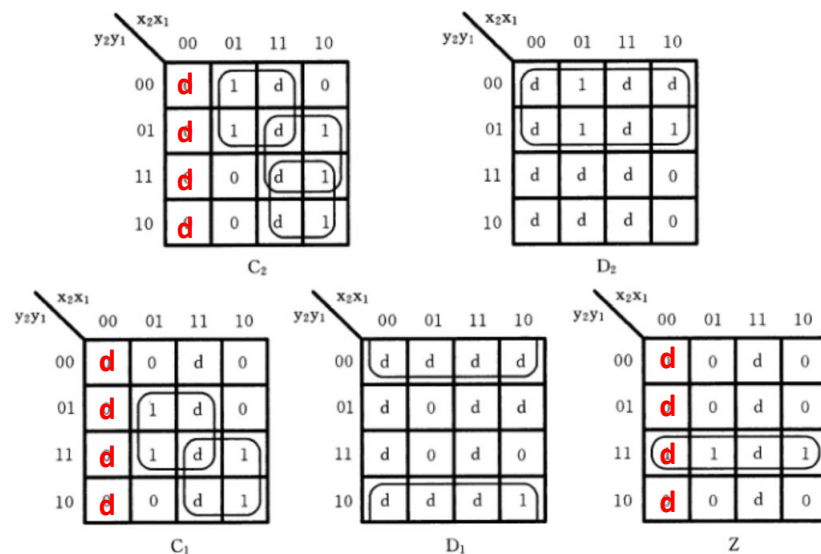


图 6.10 卡诺图

- 由卡诺图，得到激励函数和输出函数：

$$\begin{aligned}
 - C_2 &= x_1 \cdot y_2 + x_2 \cdot y_1 + x_2 \cdot y_2 & D_2 &= /y_2 \\
 - C_1 &= x_1 \cdot y_1 + x_2 \cdot y_2 & D_1 &= /y_1 \\
 - Z &= y_2 \cdot y_1
 \end{aligned}$$

- (5) 由激励函数和输出函数，画出电路图（图6.11）。

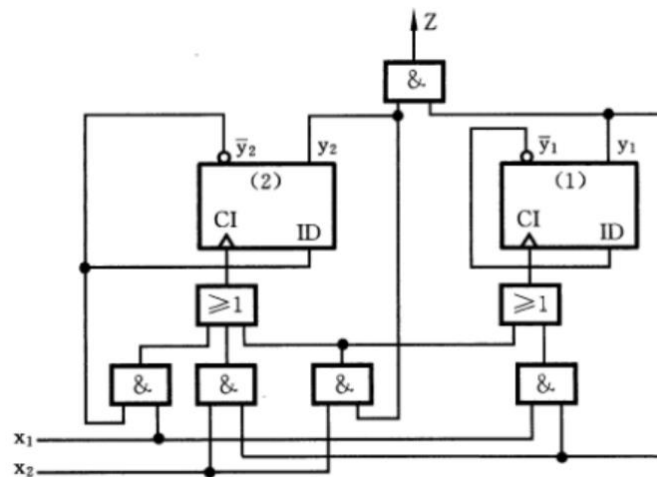


图 6.11 逻辑电路

- **例6.4**：用T触发器作为存储元件，设计一个**异步模8加1计数器**。该电路对输入端x出现的脉冲进行计数，当收到第8个脉冲时，输出端Z产生一个进位输出脉冲。

— 解：

- （假设）该电路属于**Mealy型**脉冲异步时序逻辑电路。
- （1）作出**状态图**（图6.12）和**二进制状态表**（表6.13）。

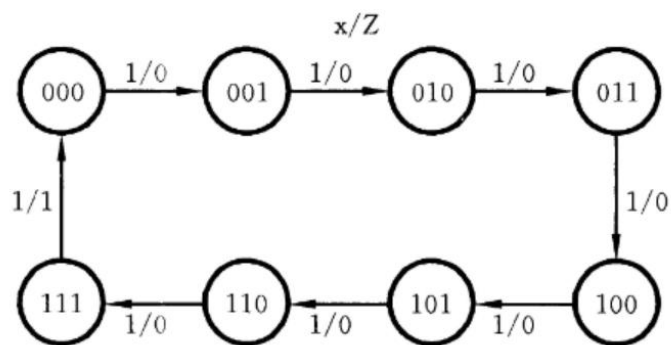


图 6.12 状态图

表 6.13 二进制状态表

| 现 态 $y_3 y_2 y_1$ | 次态 $y_3^{n+1} y_2^{n+1} y_1^{n+1}$ / 输出 Z | | | |
|----------------------|---|---|---|-----|
| | x=1 | | | |
| 0 0 0 | 0 | 0 | 1 | / 0 |
| 0 0 1 | 0 | 1 | 0 | / 0 |
| 0 1 0 | 0 | 1 | 1 | / 0 |
| 0 1 1 | 1 | 0 | 0 | / 0 |
| 1 0 0 | 1 | 0 | 1 | / 0 |
| 1 0 1 | 1 | 1 | 0 | / 0 |
| 1 1 0 | 1 | 1 | 1 | / 0 |
| 1 1 1 | 0 | 0 | 0 | / 1 |

• (2) 确定输出函数和激励函数。

— 采用3个T触发器（下降沿触发）作为存储单元，根据二进制状态表（表6.13）和T触发器的激励表（表6.7，状态不变时采用方法二：CP=0，T=d），得到状态转移关系及激励函数和输出函数真值表（表6.14，输入脉冲只需要考虑x=1的情况，激励函数中空白处表示CP=0，用下降沿符号↓表示CP=1）。

— 由表6.14可知，激励函数C₃可由y₂产生，激励函数C₂可由y₁产生，激励函数C₁可由x产生，并且T₃、T₂、T₁均可设为1。因此有：

$$\bullet \quad C_3=y_2 \quad C_2=y_1 \quad C_1=x \quad T_3=T_2=T_1=1 \quad Z=x \cdot y_3 \cdot y_2 \cdot y_1$$

表 6.13 二进制状态表

| 现 态 y ₃ y ₂ y ₁ | 次态 y ₃ ⁿ⁺¹ y ₂ ⁿ⁺¹ y ₁ ⁿ⁺¹ / 输出 Z |
|---|---|
| | x=1 |
| 0 0 0 | 0 0 1 / 0 |
| 0 0 1 | 0 1 0 / 0 |
| 0 1 0 | 0 1 1 / 0 |
| 0 1 1 | 1 0 0 / 0 |
| 1 0 0 | 1 0 1 / 0 |
| 1 0 1 | 1 1 0 / 0 |
| 1 1 0 | 1 1 1 / 0 |
| 1 1 1 | 0 0 0 / 1 |

表 6.7 T 触发器激励表

| Q | Q ⁿ⁺¹ | CP | T |
|---|------------------|----|---|
| 0 | 0 | d | 0 |
| | | 0 | d |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | d | 0 |
| | | 0 | d |

表 6.14 状态转移关系及激励函数、输出函数真值表

| 输入脉冲 | 现 态 y ₃ y ₂ y ₁ | 次 态 y ₃ ⁿ⁺¹ y ₂ ⁿ⁺¹ y ₁ ⁿ⁺¹ | 状态跳变 | | | 激励函数 | | | | | | 输出 |
|------|---|--|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----|
| x | | | y ₃ | y ₂ | y ₁ | C ₃ | T ₃ | C ₂ | T ₂ | C ₁ | T ₁ | Z |
| 1(↓) | 0 0 0 | 0 0 1 | | | ↑ | | d | | d | ↓ | 1 | 0 |
| 1(↓) | 0 0 1 | 0 1 0 | | ↑ | ↓ | | d | ↓ | 1 | ↓ | 1 | 0 |
| 1(↓) | 0 1 0 | 0 1 1 | | | ↑ | | d | | d | ↓ | 1 | 0 |
| 1(↓) | 0 1 1 | 1 0 0 | ↑ | ↓ | ↓ | ↓ | 1 | ↓ | 1 | ↓ | 1 | 0 |
| 1(↓) | 1 0 0 | 1 0 1 | | | ↑ | | d | | d | ↓ | 1 | 0 |
| 1(↓) | 1 0 1 | 1 1 0 | | ↑ | ↓ | | d | ↓ | 1 | ↓ | 1 | 0 |
| 1(↓) | 1 1 0 | 1 1 1 | | | ↑ | | d | | d | ↓ | 1 | 0 |
| 1(↓) | 1 1 1 | 0 0 0 | ↓ | ↓ | ↓ | ↓ | 1 | ↓ | 1 | ↓ | 1 | 1 |

- (3) 由激励函数和输出函数，画出**电路图**。

$$- C_3=y_2 \quad C_2=y_1 \quad C_1=x \quad T_3=T_2=T_1=1 \quad Z=x \cdot y_3 \cdot y_2 \cdot y_1$$

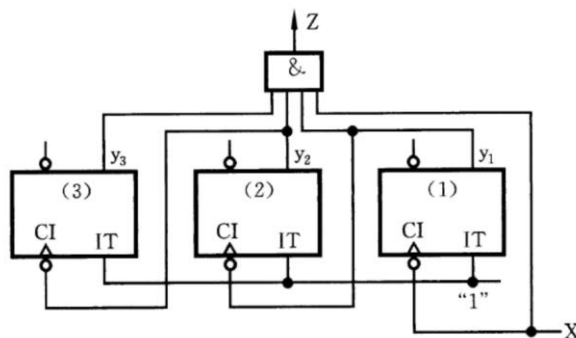


图 6.13 逻辑电路

表 6.14 状态转移关系及激励函数、输出函数真值表

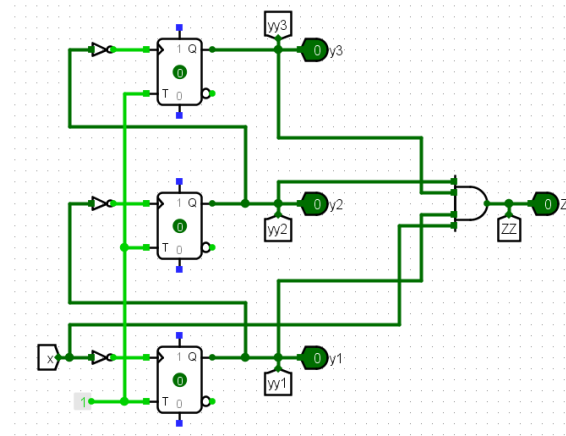
| 输入脉冲 | 现态 | 次态 | 状态跳变 | 激励函数 | | | | | | 输出 |
|------|---------------|---------------------------------|---------------|-------|-------|-------|-------|-------|-------|----|
| x | $y_3 y_2 y_1$ | $y_3^{n+1} y_2^{n+1} y_1^{n+1}$ | $y_3 y_2 y_1$ | C_3 | T_3 | C_2 | T_2 | C_1 | T_1 | Z |
| 1(↓) | 0 0 0 | 0 0 1 | | d | d | d | d | d | d | 0 |
| 1(↓) | 0 0 1 | 0 1 0 | ↑ ↓ | d | d | d | d | d | d | 0 |
| 1(↓) | 0 1 0 | 0 1 1 | ↑ | d | d | d | d | d | d | 0 |
| 1(↓) | 0 1 1 | 1 0 0 | ↑ ↓ ↓ | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1(↓) | 1 0 0 | 1 0 1 | ↑ | d | d | d | d | d | d | 0 |
| 1(↓) | 1 0 1 | 1 1 0 | ↑ ↓ | d | d | 1 | 1 | 1 | 1 | 0 |
| 1(↓) | 1 1 0 | 1 1 1 | ↑ | d | d | d | d | d | d | 0 |
| 1(↓) | 1 1 1 | 0 0 0 | ↓ ↓ ↓ | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- 假如采用**上升沿触发**的T触发器作为存储元件，则表6.14的真值表修改为下表，激励函数修改为（输出函数不变）：

$$- C_3=/y_2 \quad C_2=/y_1 \quad C_1=/x \quad T_3=T_2=T_1=1 \quad Z=x \cdot y_3 \cdot y_2 \cdot y_1$$

表 6.14 状态转移关系及激励函数、输出函数真值表

| 输入脉冲 | 现态 | 次态 | 状态跳变 | 激励函数 | | | | | | 输出 |
|------|---------------|---------------------------------|---------------|-------|-------|-------|-------|-------|-------|----|
| x | $y_3 y_2 y_1$ | $y_3^{n+1} y_2^{n+1} y_1^{n+1}$ | $y_3 y_2 y_1$ | C_3 | T_3 | C_2 | T_2 | C_1 | T_1 | Z |
| 1(↓) | 0 0 0 | 0 0 1 | | | d | | d | ↑ | 1 | 0 |
| 1(↓) | 0 0 1 | 0 1 0 | ↑ ↓ | | d | ↑ | 1 | ↑ | 1 | 0 |
| 1(↓) | 0 1 0 | 0 1 1 | ↑ | | d | | d | ↑ | 1 | 0 |
| 1(↓) | 0 1 1 | 1 0 0 | ↑ ↓ ↓ | ↑ | 1 | ↑ | 1 | ↑ | 1 | 0 |
| 1(↓) | 1 0 0 | 1 0 1 | ↑ | | d | | d | ↑ | 1 | 0 |
| 1(↓) | 1 0 1 | 1 1 0 | ↑ ↓ | | d | ↑ | 1 | ↑ | 1 | 0 |
| 1(↓) | 1 1 0 | 1 1 1 | ↑ | | d | | d | ↑ | 1 | 0 |
| 1(↓) | 1 1 1 | 0 0 0 | ↓ ↓ ↓ | ↑ | 1 | ↑ | 1 | ↑ | 1 | 1 |



6.3 电平异步时序逻辑电路

6.3.1 电平异步时序逻辑电路的结构模型与描述方法

6.3.2 电平异步时序逻辑电路的分析

6.3.3 电平异步时序逻辑电路的竞争

6.3.4 电平异步时序逻辑电路的设计

• 6.3.1 电平异步时序逻辑电路的结构模型与描述方法

– 1、电平异步时序逻辑电路的结构模型

- 图6.14：电平异步时序逻辑电路的结构模型。包括组合电路和存储电路，存储电路由反馈回路中的延迟元件构成，延迟元件一般不用专门插入延迟线，而是利用组合电路本身固有的分布延迟在反馈回路中的“集总”。

- 电路的方程组描述如下：
 - $Z_i = f_i(x_1, \dots, x_n, y_1, \dots, y_s)$
 - $Y_j = g_j(x_1, \dots, x_n, y_1, \dots, y_s)$
 - $y_j(t + \Delta t_j) = Y_j(t)$
 - Y_j 称为激励状态， y_j 称为二次状态。

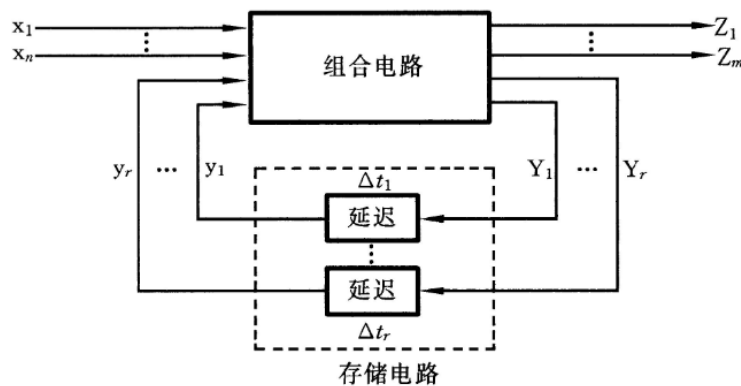


图 6.14 电平异步时序逻辑电路的结构模型

- 电平异步时序逻辑电路的**特点**:

- ① 电路输出和状态的改变是由输入电平信号的变化直接引起的, 电路可以及时地对输入信号的变化作出响应, **工作速度较高**。
- ② 电路的二次状态 y 和激励状态 Y 仅仅相差一个时间延迟, 即二次状态 y 是激励状态 Y 经过 Δt 延迟后的“重现”。当输入信号不变时, 二次状态 y 与激励状态 Y 相同 ($y=Y$), 此时电路处于**稳定状态**。
- ③ 输入信号的一次变化可能引起二次状态的**多次变化**。当电路处在稳定状态下、输入信号发生变化时, 若 Y 与 y 的值相同, 则电路处于**稳定状态**; 若 Y 与 y 的值不同, 则变化后的 Y 经过 Δt 延迟后形成新的 y 反馈到组合电路的输入端, 这个新的 y 又会引起 z 和 Y 的变化, 这是一个循环过程, 直到 $y=Y$ 。在变化过程终止前, 电路处于**不稳定状态**; 变化过程结束后, 电路进入一个**新的稳定状态**, 这是电平异步时序逻辑电路的重要特征。

– 2、输入信号的约束

- 为了保证电路可靠地工作, 对输入信号有**两条约束**:

- ① 不允许两个或两个以上输入信号同时发生变化。
- ② 输入信号变化引起的电路响应必须完全结束后, 才允许输入信号再次变化。

- 通常将满足上述条件的工作方式称为**基本工作方式**, 对应的电路称为**基本型电路**。

– 3、描述方式

状态表

状态图

- 在电平异步时序逻辑电路中，除了逻辑方程外，一般使用**流程表**和**总态图**描述一个电路的工作过程和逻辑功能。

- 流程表**是用来反映电路输出信号、激励状态和输入信号、二次状态之间关系的一种表格形式。

- 表6.15: Mealy型电路的流程表格式。

- 表6.16: Moore型电路的流程表格式。

表 5.1 Mealy 型电路状态表格式

| 现 态 | 次态/输出 | |
|-----|-------|-------------|
| | 输入 x | |
| | | |
| y | | y^{n+1}/Z |
| | | |

表 5.2 Moore 型电路状态表格式

| 现 态 | 次 态 | 输 出 |
|-----|-----------|-----|
| | 输入 x | |
| | | |
| y | y^{n+1} | Z |
| | | |

状态表

表 6.15 Mealy 型流程表格式

| 二次 状态 | 激励状态/输出 | |
|----------|---------|------|
| | | 输入 x |
| | | |
| y | | Y/Z |
| | | |

表 6.16 Moore 型流程表格式

| 二次 状态 | 激励状态 | | 输出 |
|----------|------|------|----|
| | | 输入 x | |
| | | | |
| y | | Y | Z |
| | | | |

- 图6.15(a)：用与非门构成的**基本R-S触发器**。
- 图6.15(b)：将图6.15(a)用电平异步时序逻辑电路表示。该电路属于**Moore型**，**Y**为激励状态，**y**为二次状态， Δt 为延迟元件。激励方程为：

$$w = \neg(R \cdot y) \quad Y = \neg(S \cdot w) = \neg S + w = \neg S + R \cdot y$$

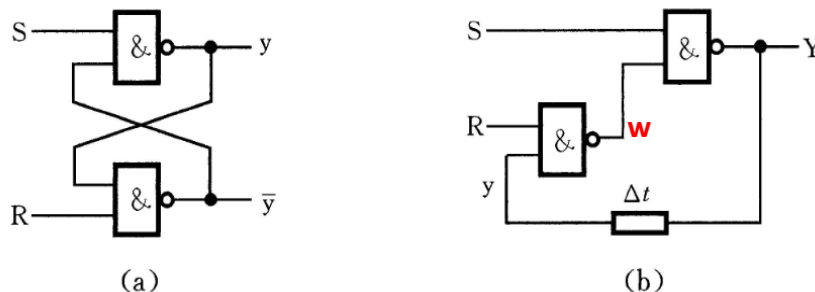


图 6.15 一个简单的电平异步时序逻辑电路

- 表6.17：R-S触发器的**流程表**。表中加圆圈的激励状态表示电路处于**稳态**（此时，激励状态=二次状态），否则表示电路处于**非稳态**；因为是**Moore型**，因此输出与二次状态**y**相同。

表 6.17 R-S 触发器流程表

| 二次状态 y | 激励状态 Y | | | | 输 出 |
|-------------|----------|---------|---------|---------|-----|
| | $RS=00$ | $RS=01$ | $RS=11$ | $RS=10$ | |
| 0 | d | ① | ① | 1 | 0 |
| 1 | d | 0 | ① | ① | 1 |

二次状态**y**和激励状态**Y**分别对应表3.13中的**Q**和 **Q^{n+1}**

表 3.13 与非门构成的基本
R-S 触发器功能表

| R | S | Q^{n+1} | 功能说明 |
|---|---|-----------|------|
| 0 | 0 | d | 不定 |
| 0 | 1 | 0 | 置 0 |
| 1 | 0 | 1 | 置 1 |
| 1 | 1 | Q | 不变 |

- **总态**是指电路输入 x 和二次状态 y 的组合，记为 (x,y) 。
 - **稳定总态**：在表6.17的流程表中用圆圈表示，共有4个。
 - **非稳定总态**：在图6.17的流程表中没有用圆圈表示，也有4个。
- **总态图**是反映稳定总态之间转移关系及相应输出的一种有向图。

- 表6.17：与表6.17对应的总态图。



表 6.17 R-S 触发器流程表

| 二次状态 y | 激励状态 Y | | | | 输 出 |
|-------------|----------|---------|---------|---------|-----|
| | $RS=00$ | $RS=01$ | $RS=11$ | $RS=10$ | |
| 0 | d | ① | ① | 1 | 0 |
| 1 | d | 0 | ① | ① | 1 |

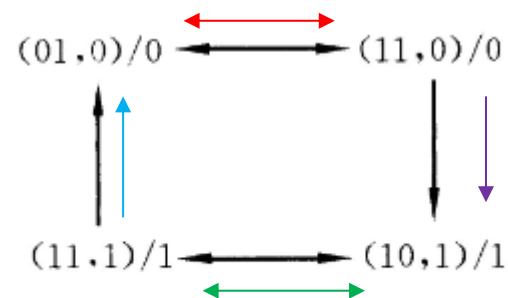


图 6.16 与表 6.17 对应的总态图

• 6.3.2 电平异步时序逻辑电路的分析

– 电平异步时序逻辑电路的**分析步骤**：

① 根据电路写出**输出函数**和**激励函数**表达式。

② 作出**流程表**。

③ 作出**总态图**或**时间图**。

④ 说明电路逻辑功能。

– **例6.5**：分析图6.17所示电平异步时序逻辑**电路**。

– 解：该电路属于**Moore型**（输出只与状态有关）。

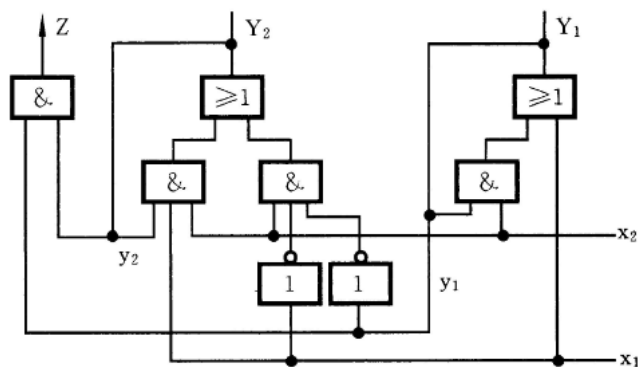


图 6.17 逻辑电路

— (1) 根据电路写出**输出函数和激励函数**：

- $Z = y_2 \cdot y_1$
- $Y_2 = x_2 \cdot x_1 \cdot y_2 + x_2 \cdot /x_1 \cdot /y_1$ $Y_1 = x_2 \cdot y_1 + x_1$

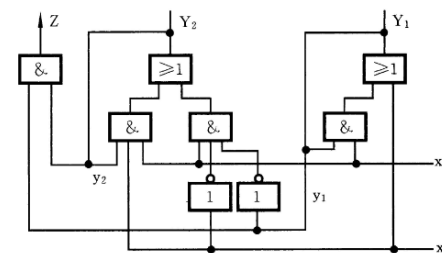


图 6.17 逻辑电路

— (2) 根据激励函数和输出函数作出**流程表**（表6.18）：

- 表6.18中画圆圈的为稳定总态，共6个。

— (3) 根据流程表作**总态图**（图6.18）：

- 图6.18中有6个稳定总态。

表 6.18 流程表

| 二次状态 $y_2 \quad y_1$ | | 激励状态 $Y_2 Y_1$ | | | | 输 出 Z |
|-------------------------|---|----------------|----------------|----------------|----------------|------------|
| | | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ | |
| 0 | 0 | ⊙00 | 01 | 01 | 10 | 0 |
| 0 | 1 | 00 | ⊙1 | ⊙1 | ⊙1 | 0 |
| 1 | 1 | 00 | 01 | ⊙1 | 01 | 1 |
| 1 | 0 | 00 | 01 | 1 | ⊙10 | 0 |

这1列直接给出

这4列根据激励函数得到

这1列根据输出函数得到

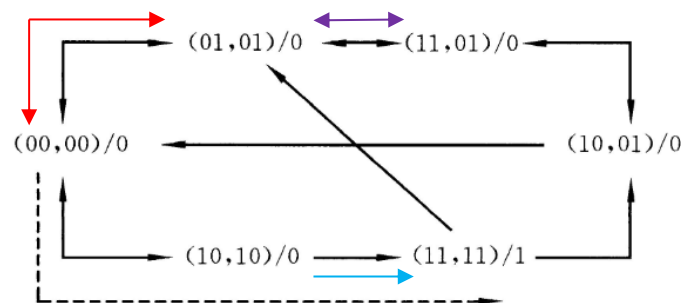


图 6.18 总态图

(00,00)/0: 第1个00表示输入 $x_2 x_1$, 第2个00表示二次状态 $y_2 y_1$, 第3个0表示输出 Z

- 假设电路的初始总态为 $(x_2x_1, y_2y_1)=(00,00)$ ，输入的变化序列为 $00 \rightarrow 10 \rightarrow 11 \rightarrow 01 \rightarrow 00 \rightarrow 01 \rightarrow 11 \rightarrow 10$ 。
- 根据流程表作出总态和输出响应序列：
 - 其中加*的表示是非稳定总态。

表 6.18 流程表

| 二次状态 $y_2 \ y_1$ | 激励状态 Y_2Y_1 | | | | 输出 Z |
|---------------------|---------------|-------------|-------------|-------------|-----------|
| | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ | |
| 0 0 | ①① | 01 | 01 | 0 | 0 |
| 0 1 | 00 | ①① | ①① | ①① | 0 |
| 1 1 | 00 | 01 | ①① | ①① | 1 |
| 1 0 | 00 | 01 | ①① | ①① | 0 |

| 时刻 t_i | t_0 | t_1 | t_2 | t_3 | t_4 | t_5 | t_6 | t_7 |
|-----------------------|-----------|-------------|-------------|-------------|-------------|-------------|-------------|-----------|
| 输入 x_2x_1 | 00 | 10 | 11 | 01 | 00 | 01 | 11 | 10 |
| 总态 (x_2x_1, y_2y_1) | $(00,00)$ | $(10,00)^*$ | $(11,10)^*$ | $(01,11)^*$ | $(00,01)^*$ | $(01,00)^*$ | $(11,01)^*$ | $(10,01)$ |
| 输出 Z | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

- 对应的时间图（图6.19）：

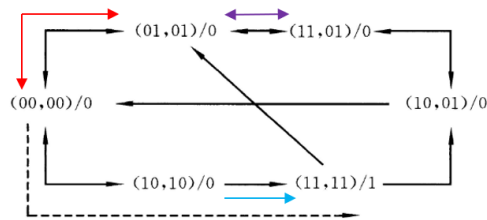


图 6.18 总态图

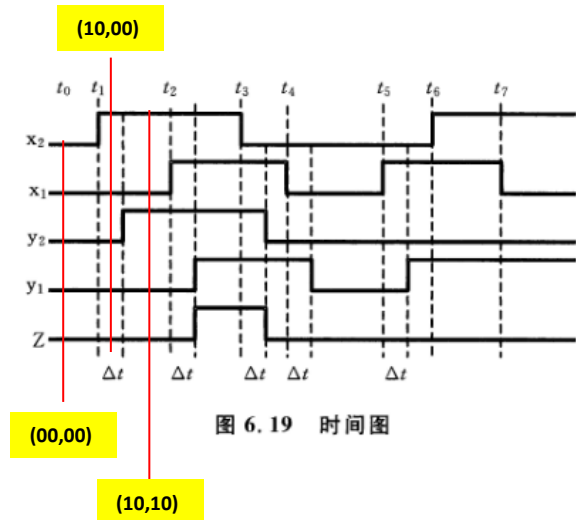


图 6.19 时间图

- （4）根据总态图和时间图可知，仅当电路收到输入序列“ $00 \rightarrow 10 \rightarrow 11$ ”时，才产生一个高电平输出信号，其他情况均输出低电平。因此该电路是一个“ $00 \rightarrow 10 \rightarrow 11$ ”序列检测器。

• 6.3.3 电平异步时序逻辑电路的竞争

- 前面对电平异步时序逻辑电路的分析是假设各反馈回路之间的延迟是相同的，即图6.14中的 $\Delta t_1 = \Delta t_2 = \dots = \Delta t_r$ 。
- 实际电路中，各反馈回路之间的延迟是不相同的，这样可能会导致竞争。
- 所谓竞争是指当输入信号变化引起电路中两个或两个以上状态变量发生变化时，由于各反馈回路延迟时间的不同，使状态变量的变化有先有后，从而导致不同状态响应过程的现象。
- 若竞争的结果最终能到达预定的稳态，则称为非临界竞争。若竞争的结果使电路到达不同的稳态，即状态转移不可预测，则称为临界竞争（与4.3小节“组合逻辑电路的险象”类似）。

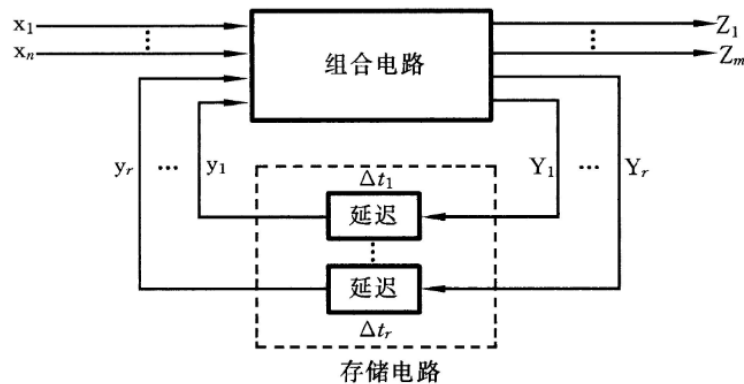


图 6.14 电平异步时序逻辑电路的结构模型

- 某电平异步时序逻辑电路的框图如图6.20所示，假设描述该电路的流程表如表6.19所示。
- 当电路处于稳定总态 $(x_2x_1, y_2y_1)=(00,00)$ ，输入 x_2x_1 由00->10时，电路会经过非稳定总态(10,00)（表6.19中标红色三角形的）到达稳定总态(10,11)。由于此次输入变化引起激励状态 Y_2Y_1 从00->11，当 $\Delta t_1 \neq \Delta t_2$ 时，将产生竞争。
- 同样，当电路处于稳定总态 $(x_2x_1, y_2y_1)=(10,11)$ ，输入 x_2x_1 由10->00时，由于激励状态 Y_2Y_1 从11->00，电路也将产生竞争。

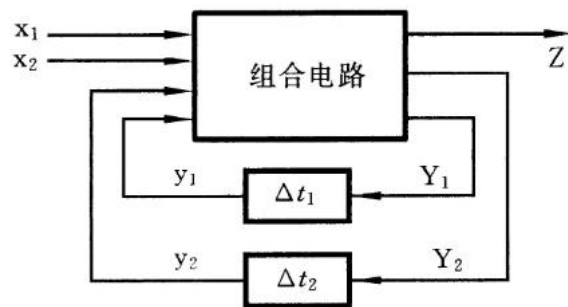


图 6.20 某电平异步时序电路框图

表 6.19 流程表

| 二次状态 $y_2 \quad y_1$ | | 激励状态 Y_2Y_1 /输出 Z | | | |
|-------------------------|---|-----------------------|-------------|-------------|---|
| | | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0 | 0 | ⊙0/0 | ⊙0/0 | 01/0 | 11/0 Δ (10,00) |
| 0 | 1 | 00/0 | ⊙1/0 | ⊙1/0 | ⊙1/0 |
| 1 | 1 | 00/0 | 00/0 | 10/0 | ⊙1/0 |
| 1 | 0 | 00/0 | 00/0 | ⊙0/1 | ⊙0/0 |

— 当电路处于稳定总态 $(x_2x_1, y_2y_1)=(00,00)$ ，输入 x_2x_1 由 $00 \rightarrow 10$ 时，其状态响应过程如下：

- ① 当 $\Delta t_1 = \Delta t_2$ 时，二次状态 y_2y_1 将同时响应激励状态 Y_2Y_1 的变化，即 y_2y_1 由 $00 \rightarrow 11$ ，总态变化过程为： $(00,00) \rightarrow (10,00) \rightarrow (10,11)$ ，即到达**预定的稳定总态 $(10,11)$** 。
- ② 当 $\Delta t_1 < \Delta t_2$ 时，二次状态 y_2 对激励状态 Y_2 的响应，优先于二次状态 y_1 对激励状态 Y_1 的响应，即 y_2y_1 由 $00 \rightarrow 10$ ，总态变化过程为： $(00,00) \rightarrow (10,00) \rightarrow (10,10)$ 。由于 $(10,10)$ 是稳定状态，故电路停留在该稳态，即到达一个**非预期的稳定总态 $(10,10)$** 。
- ③ 当 $\Delta t_1 > \Delta t_2$ 时，二次状态 y_2 对激励状态 Y_2 的响应，落后于二次状态 y_1 对激励状态 Y_1 的响应，即 y_2y_1 由 $00 \rightarrow 01$ ，总态变化过程为： $(00,00) \rightarrow (10,00) \rightarrow (10,01)$ 。由于 $(10,01)$ 是稳定状态，故电路停留在该稳态，即到达另一个**非预期的稳定总态 $(10,01)$** 。

— 因此，本次竞争为**临界竞争**（状态转移不可预测）。

表 6.19 流程表

| 二次状态 $y_2 \quad y_1$ | | 激励状态 Y_2Y_1 /输出 Z | | | |
|-------------------------|---|-----------------------|-------------|-------------|---------------|
| | | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0 | 0 | 00/0 | 00/0 | 01/0 | 11/0 (10,00) |
| 0 | 1 | 00/0 | 01/0 | 01/0 | 01/0 |
| 1 | 1 | 00/0 | 00/0 | 10/0 | 11/0 Δ |
| 1 | 0 | 00/0 | 00/0 | 10/1 | 10/0 |

$\Delta t_1 = \Delta t_2$

表 6.19 流程表

| 二次状态 $y_2 \quad y_1$ | | 激励状态 Y_2Y_1 /输出 Z | | | |
|-------------------------|---|-----------------------|-------------|-------------|---------------|
| | | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0 | 0 | 00/0 | 00/0 | 01/0 | 11/0 (10,00) |
| 0 | 1 | 00/0 | 01/0 | 01/0 | 01/0 |
| 1 | 1 | 00/0 | 00/0 | 10/0 | 11/0 |
| 1 | 0 | 00/0 | 00/0 | 10/1 | 10/0 Δ |

$\Delta t_1 < \Delta t_2$

表 6.19 流程表

| 二次状态 $y_2 \quad y_1$ | | 激励状态 Y_2Y_1 /输出 Z | | | |
|-------------------------|---|-----------------------|-------------|-------------|---------------|
| | | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0 | 0 | 00/0 | 00/0 | 01/0 | 11/0 (10,00) |
| 0 | 1 | 00/0 | 01/0 | 01/0 | 01/0 Δ |
| 1 | 1 | 00/0 | 00/0 | 10/0 | 11/0 |
| 1 | 0 | 00/0 | 00/0 | 10/1 | 10/0 |

$\Delta t_1 > \Delta t_2$

— 当电路处于稳定总态 $(x_2x_1, y_2y_1)=(10,11)$ ，输入 x_2x_1 由 $10 \rightarrow 00$ 时，其状态响应过程如下：

- ① 当 $\Delta t_1 = \Delta t_2$ 时，二次状态 y_2y_1 将同时响应激励状态 Y_2Y_1 的变化，即 y_2y_1 由 $11 \rightarrow 00$ ，总态变化过程为： $(10,11) \rightarrow (00,11) \rightarrow (00,00)$ ，即到达**预定的稳定总态 $(00,00)$** 。
- ② 当 $\Delta t_1 < \Delta t_2$ 时，二次状态 y_2 对激励状态 Y_2 的响应，优先于二次状态 y_1 对激励状态 Y_1 的响应，即 y_2y_1 由 $11 \rightarrow 01$ ，总态变化过程为： $(10,11) \rightarrow (00,11) \rightarrow (00,01) \rightarrow (00,00)$ ，即到达**预定的稳定总态 $(00,00)$** 。
- ③ 当 $\Delta t_1 > \Delta t_2$ 时，二次状态 y_2 对激励状态 Y_2 的响应，落后于二次状态 y_1 对激励状态 Y_1 的响应，即 y_2y_1 由 $11 \rightarrow 10$ ，总态变化过程为： $(10,11) \rightarrow (00,11) \rightarrow (00,10) \rightarrow (00,00)$ ，即到达**预定的稳定总态 $(00,00)$** 。

— 因此，本次竞争为**非临界竞争**（状态转移可预测）。

— 可以**采用流程表检查电路是否存在临界竞争**：若输入信号变化所到达的列只有一个稳态，则该竞争属于非临界竞争；若输入信号变化所到达的列有两个或两个以上稳态，则该竞争属于临界竞争。

表 6.19 流程表

| 二次状态 $y_2 \quad y_1$ | | 激励状态 Y_2Y_1 /输出 Z | | | |
|-------------------------|---|-----------------------|-------------|-------------|---------------|
| | | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0 | 0 | 00/0 Δ | 00/0 | 01/0 | 11/0 |
| 0 | 1 | 00/0 | 01/0 | 01/0 | 01/0 |
| 1 | 1 | 00/0 | 00/0 | 10/0 | 11/0 Δ |
| 1 | 0 | 00/0 | 00/0 | 10/1 | 10/0 |

(00,11)

$\Delta t_1 = \Delta t_2$

表 6.19 流程表

| 二次状态 $y_2 \quad y_1$ | | 激励状态 Y_2Y_1 /输出 Z | | | |
|-------------------------|---|-----------------------|-------------|-------------|---------------|
| | | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0 | 0 | 00/0 Δ | 00/0 | 01/0 | 11/0 |
| 0 | 1 | 00/0 | 01/0 | 01/0 | 01/0 |
| 1 | 1 | 00/0 | 00/0 | 10/0 | 11/0 Δ |
| 1 | 0 | 00/0 | 00/0 | 10/1 | 10/0 |

(00,01)

(00,11)

$\Delta t_1 < \Delta t_2$

表 6.19 流程表

| 二次状态 $y_2 \quad y_1$ | | 激励状态 Y_2Y_1 /输出 Z | | | |
|-------------------------|---|-----------------------|-------------|-------------|---------------|
| | | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0 | 0 | 00/0 Δ | 00/0 | 01/0 | 11/0 |
| 0 | 1 | 00/0 | 01/0 | 01/0 | 01/0 |
| 1 | 1 | 00/0 | 00/0 | 10/0 | 11/0 Δ |
| 1 | 0 | 00/0 | 00/0 | 10/1 | 10/0 |

(00,11)

(00,10)

$\Delta t_1 > \Delta t_2$

• 6.3.4 电平异步时序逻辑电路的设计

– 电平异步时序逻辑电路的设计步骤：

- ① 根据设计要求，建立原始流程表。
- ② 化简原始流程表，得到最简流程表。
- ③ 状态编码，得到二进制流程表。
- ④ 根据二进制流程表，求出激励状态和输出函数表达式。
- ⑤ 画出逻辑电路图。

– 1、建立原始流程表

- 通常根据题目要求，画出典型输入、输出的时间图（或作出原始总态图），然后根据时间图，建立原始流程表，具体过程如下：
- （1）画出典型输入/输出时间图，并设立相应状态，需要注意以下3点：
 - ① 符合题意，即正确体现设计要求。
 - ② 满足电平异步时序逻辑电路不允许两个或两个以上输入信号同时改变的约束条件。
 - ③ 尽可能反映输入信号在各种取值下允许发生的变化。
- （2）建立原始流程表，具体包括：
 - ① 画出原始流程表，并填入稳定状态和相应输出。
 - ② 填入非稳定状态并指定非稳定状态下的输出，完善流程表。
 - ③ 填入无关状态和无关输出。

- **例6.6:** 某电平异步时序逻辑电路有两个输入端 x_1 和 x_2 ，一个输出端 Z 。输出与输入之间的关系为：只要 $x_1x_2=00$ ，则 $Z=0$ ，在此之后当 $x_1x_2=01$ 或 10 时， $Z=1$ ；只要 $x_1x_2=11$ ，则 $Z=1$ ，在此之后当 $x_1x_2=01$ 或 10 时， $Z=0$ 。作出该电路的**原始流程表**。

• 解：

— (1) 画出典型输入/输出**时间图**（图6.21），并设立相应状态，共有**6个稳定状态**：

- 稳定状态①： $x_1x_2=00$ ， $Z=0$
- 稳定状态②： $x_1x_2=10$ ， $Z=1$
- 稳定状态③： $x_1x_2=01$ ， $Z=1$
- 稳定状态④： $x_1x_2=11$ ， $Z=1$
- 稳定状态⑤： $x_1x_2=10$ ， $Z=0$
- 稳定状态⑥： $x_1x_2=01$ ， $Z=0$

x_1x_2Z 共8种情况， $x_1x_2Z=001$ ， $x_1x_2Z=110$ ，这两种情况不会出现

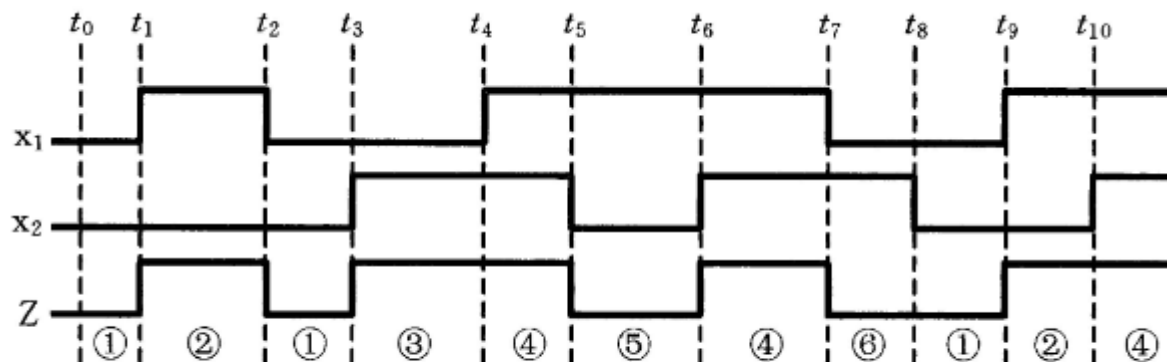


图 6.21 典型输入/输出时间图

– (2) 建立原始流程表。

- 首先，画出含有6个稳定状态的部分流程表(I)（表6.20），6个稳定状态如下：

- 稳定状态①: $x_1x_2=00$, $Z=0$
- 稳定状态②: $x_1x_2=10$, $Z=1$
- 稳定状态③: $x_1x_2=01$, $Z=1$
- 稳定状态④: $x_1x_2=11$, $Z=1$
- 稳定状态⑤: $x_1x_2=10$, $Z=0$
- 稳定状态⑥: $x_1x_2=01$, $Z=0$

表 6.20 部分流程表(I)

| 二次 状态 y | 激励状态 Y/输出 Z | | | |
|---------------|-------------|-------------|-------------|-------------|
| | $x_1x_2=00$ | $x_1x_2=01$ | $x_1x_2=11$ | $x_1x_2=10$ |
| 1 | ①/0 | | | |
| 2 | | | | ②/1 |
| 3 | | ③/1 | | |
| 4 | | | ④/1 | |
| 5 | | | | ⑤/0 |
| 6 | | ⑥/0 | | |

- 然后，填入非稳定状态，并指定非稳定状态下的输出，完善流程表，得到部分流程表(III)（表6.21）：

①-→②

输出=d是因为状态①的输出为0、状态②的输出为1，发生了变化，状态转换过程中，输出是不定的

①-→③

①-→⑤

①-→⑥

输出=0是因为状态①的输出是0、状态⑥的输出也是0，状态转换过程中，输出没有变化

③-→①

②-→①

④-→②

④-→③

⑤-→④

④-→⑤

④-→⑥

表 6.21 部分流程表(Ⅱ)

| 二次状态 y | 激励状态 Y/输出 Z | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_1 x_2 = 00$ | $x_1 x_2 = 01$ | $x_1 x_2 = 11$ | $x_1 x_2 = 10$ |
| 1 | ①/0 | 3/d | | 2/d |
| 2 | 1/d | | 4/1 | ②/1 |
| 3 | 1/d | ③/1 | 4/1 | |
| 4 | | 6/d | ④/1 | 5/d |
| 5 | 1/0 | | 4/d | ⑤/0 |
| 6 | 1/0 | ⑥/0 | 4/d | |

⑥-→④

- 最后，填入无关状态和无关输出（**d/d**），即可得到完整流程表（表6.22）。

表 6.22 完整流程表

| 二次状态 y | 激励状态 Y/输出 Z | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_1 x_2 = 00$ | $x_1 x_2 = 01$ | $x_1 x_2 = 11$ | $x_1 x_2 = 10$ |
| 1 | ①/0 | 3/d | → d/d | 2/d |
| 2 | 1/d | → d/d | 4/1 | ②/1 |
| 3 | 1/d | ③/1 | 4/1 | → d/d |
| 4 | → d/d | 6/d | ④/1 | 5/d |
| 5 | 1/0 | → d/d | 4/d | ⑤/0 |
| 6 | 1/0 | ⑥/0 | 4/d | → d/d |

– 2、化简原始流程表

- 原始流程表中的状态数目往往比较多，需要进行化简，得到最简流程表。
- 原始流程表的化简是建立在**状态相容**概念基础上的。因为原始流程表中的每一行代表一个稳定的状态，状态相容的概念被引申为**相容行**的概念。
- **相容行**：对于原始流程表中的某两行，如果每一列确定的输出相同，且确定的激励状态相同、交错、循环、相容或为各自本身，则这两行为相容行。
- 原始流程表的化简过程与不完全确定状态表（见第5章的5.3.3小节）的化简过程类似，其一般**步骤**如下：
 - ① 作隐含表，找出相容行。
 - ② 作合并图，求出最大相容行类。
 - ③ 从相容行类中选择一个最小闭覆盖。
 - ④ 作出最简流程表。

- 例6.7： 化简表6.22所示的原始流程表。

• 解:

- ① 根据完整流程表（表6.22）作**隐含表**（表6.22），从隐含表中找出**相容行对**：(1,2)、(1,3)、(2,3)、(2,6)、(3,5)、(4,5)、(4,6)、(5,6)。
- ② 根据相容行对作**合并图**（图6.23），从合并图中找出**最大相容行类**：(1,2,3)、(4,5,6)、(3,5)、(2,6)。
- ③ 从最大相容行类中选择**最小闭覆盖**：(1,2,3)、(4,5,6)。
- ④ 将最小闭覆盖(1,2,3)、(4,5,6)分别用A、B代表，即可得到**最简流程表**（表6.23）。

表 6.22 完整流程表

| 二次状态 y | 激励状态 Y/输出 Z | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_1 x_2 = 00$ | $x_1 x_2 = 01$ | $x_1 x_2 = 11$ | $x_1 x_2 = 10$ |
| 1 | ①/0 | 3/d | d/d | 2/d |
| 2 | 1/d | d/d | 4/1 | ②/1 |
| 3 | 1/d | ③/1 | 4/1 | d/d |
| 4 | d/d | 6/d | ④/1 | 5/d |
| 5 | 1/0 | d/d | 4/d | ⑤/0 |
| 6 | 1/0 | ⑥/0 | 4/d | d/d |

表 6.23 最简流程表

| 二次状态 y | 激励状态 Y/输出 Z | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_1 x_2 = 00$ | $x_1 x_2 = 01$ | $x_1 x_2 = 11$ | $x_1 x_2 = 10$ |
| A | Ⓐ/0 | Ⓐ/1 | B/1 | Ⓐ/1 |
| B | A/0 | Ⓑ/0 | Ⓑ/1 | Ⓑ/0 |

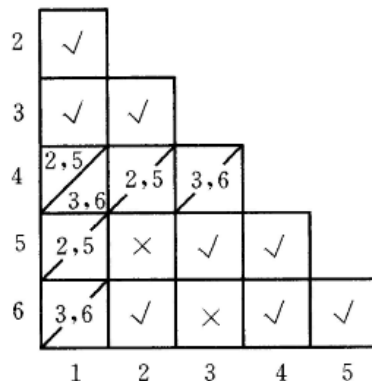


图 6.22 隐含表

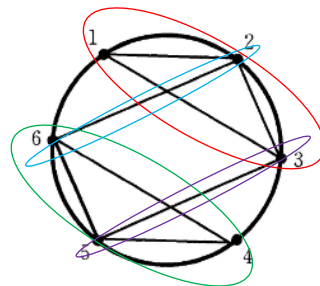


图 6.23 合并图

– 3、状态编码

- 在电平异步时序逻辑电路设计中，状态编码（用二进制代码表示最简流程表中的每个状态）时，考虑的主要问题是**如何避免反馈回路之间的临界竞争**（状态转移不可预测），保证电路可靠地实现预定功能。通常有**3种**状态编码方法：
 - （1）相邻状态，相邻分配
 - 如果能保证每次状态转移时，仅有一个状态变量变化，则不会竞争。可以通过“**相邻状态，相邻分配**”的方法消除竞争。
 - 所谓**相邻状态**是指稳态下输入取值作相邻变化时，需要直接发生转换的状态。所谓**相邻分配**是指分配给相邻状态的代码为相邻代码（仅一位不同）。
 - 通常采用**状态相邻图**的方法找出流程表中各状态的相邻关系。
 - 画**状态相邻图**的方法：先将流程表中的每一个状态用一个圆圈表示，然后从流程表中每一个稳态出发，找出输入取值作相邻变化时的下一个稳态，并用有向线段将其连接起来，表示这两个状态为相邻状态。

- **例6.8:** 对表6.24所示流程表进行**状态编码**，作出二进制流程表。
- 解：
 - 首先，根据“**相邻状态，相邻分配**”的原则，作相邻状态图（图6.24）。
 - 根据状态相邻图，可以得到状态分配方案（图6.25），A、B、C、D分别用00、01、10、11表示。
 - 根据状态分配方案，得到二进制流程表（表6.25）。

表 6.24 流程表

| 二次状态 y | 激励状态 Y | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A | Ⓐ | Ⓐ | B | C |
| B | A | Ⓑ | Ⓑ | Ⓑ |
| C | Ⓒ | A | D | Ⓒ |
| D | C | Ⓓ | Ⓓ | Ⓓ |

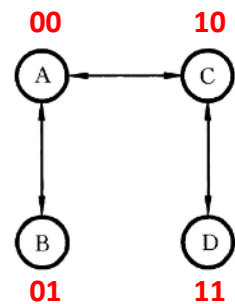


图 6.24 状态相邻图

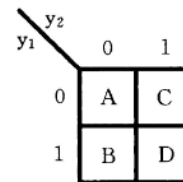


图 6.25 状态分配方案

表 6.25 二进制流程表

| 二次状态 $y_2 \ y_1$ | | 激励状态 $Y_2 Y_1$ | | | |
|---------------------|-----|----------------|----------------|----------------|----------------|
| | | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A | 0 0 | Ⓐ | Ⓐ | 01 | 10 |
| B | 0 1 | 00 | Ⓑ | Ⓑ | Ⓑ |
| D | 1 1 | 10 | Ⓖ | Ⓖ | Ⓖ |
| C | 1 0 | Ⓔ | 00 | 11 | Ⓕ |

• (2) 增加过渡状态，实现相邻分配

- 有一些流程表，状态之间的相邻关系出现由奇数个状态构成的闭环，因而无法直接实现状态的相邻分配。此时，可以通过增加过渡状态，实现相邻分配，得到一个无竞争的二进制流程表。

— **例6.9：**对表6.26所示流程表进行状态编码，得到二进制流程表。

— 解：

- 根据流程表，画出相邻状态图（图6.26）。因为状态之间的相邻关系出现了由奇数（3）个状态构成的闭环，此时需要在状态A与状态C之间增加一个过渡状态D。增加过渡状态后的流程表如表6.27所示，增加过渡状态后的状态相邻图如图6.27所示。用00、01、11、10分别表示A、B、C、D状态，则可得到二进制流程表（表6.28）。

表 6.26 流程表

| 二次状态 y | 激励状态 Y | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A | Ⓐ | B | C | Ⓐ |
| B | A | Ⓑ | Ⓑ | C |
| C | A | B | Ⓒ | Ⓒ |

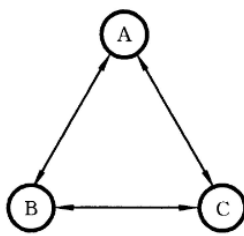


图 6.26 状态相邻图

表 6.28 二进制流程表

| 二次状态 $y_2 y_1$ | 激励状态 $Y_2 Y_1$ | | | |
|-------------------|----------------|----------------|----------------|----------------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A 0 0 | Ⓐ | 01 | 10 | Ⓐ |
| B 0 1 | 00 | Ⓑ | Ⓑ | 11 |
| C 1 1 | 10 | 01 | Ⓒ | Ⓒ |
| D 1 0 | 00 | dd | 11 | dd |

表 6.27 增加过渡状态后的流程表

| 二次状态 y | 激励状态 Y | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A | Ⓐ | B | D | Ⓐ |
| B | A | Ⓑ | Ⓑ | C |
| C | D | B | Ⓒ | Ⓒ |
| D | A | d | C | d |

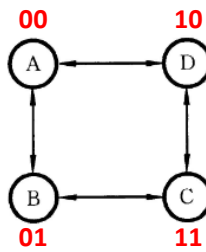


图 6.27 增加过渡状态后的状态相邻图

- (3) 允许非临界竞争，避免临界竞争

- **临界竞争**（若竞争的结果使电路到达不同的稳态，即状态转移不可预测，则称为临界竞争）会影响电路的正常工作，而**非临界竞争**（若竞争的结果最终能到达预定的稳态，则称为非临界竞争）并不影响电路的正常工作。因此，在状态分配时，只需**避免临界竞争**。

- 对于有些流程表，虽然无法用最少位数的代码实现无竞争的状态分配，但可以通过**将竞争限制在只有一个稳定的列**，即允许非临界竞争，从而实现无临界竞争的状态分配。

- **例6.10**：对表6.29所示流程表进行**状态编码**，得到二进制流程表。

- 解：

- 根据流程表作出状态相邻图（图6.28）。显然用2位二进制代码无法实现“相邻状态，相邻分配”。

表 6.29 流程表

| 二次状态 y | 激励状态 Y | | | | 输 出 Z |
|-----------|----------------|----------------|----------------|----------------|----------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ | |
| A | Ⓐ | C | D | Ⓐ | 0 |
| B | A | C | Ⓑ | Ⓑ | 0 |
| C | A | Ⓒ | Ⓒ | B | 0 |
| D | A | C | Ⓓ | A | 1 |

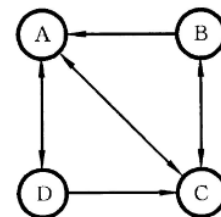


图 6.28 状态相邻图

- 可以通过增加过渡状态和增加代码位数实现相邻分配，但这样会导致**增加电路的复杂性**。
- 还可以采用“**允许非临界竞争，避免临界竞争**”的方法。
- 状态A和状态C之间的转移只发生在 $x_2x_1=00$ 和 01 这两列，而这两列各只有一个稳定状态，这就意味着，A和C发生转移时，即使产生竞争，也**属于非临界竞争**。即在状态分配时，**A和C可以不相邻**。
- 用00、01、11、10分别表示状态A、B、C、D，可以得到图6.30所示二进制流程表。

表 6.29 流程表

| 二次状态 y | 激励状态 Y | | | | 输 出 Z |
|-----------|-------------|-------------|-------------|-------------|----------|
| | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ | |
| A | Ⓐ | C | D | Ⓐ | 0 |
| B | A | C | Ⓑ | Ⓑ | 0 |
| C | A | Ⓒ | Ⓒ | B | 0 |
| D | A | C | Ⓓ | A | 1 |

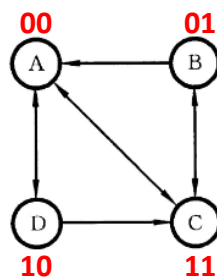


图 6.28 状态相邻图

表 6.30 二进制流程表

| 二次状态 $y_2 \ y_1$ | | 激励状态 Y_2Y_1 | | | | 输 出 Z |
|---------------------|-----|---------------|-------------|-------------|-------------|----------|
| | | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ | |
| A | 0 0 | Ⓐ | 11 | 10 | Ⓐ | 0 |
| B | 0 1 | 00 | 11 | Ⓑ | Ⓑ | 0 |
| C | 1 1 | 00 | Ⓒ | Ⓒ | 01 | 0 |
| D | 1 0 | 00 | 11 | Ⓓ | 00 | 1 |

– 4、确定激励状态和输出函数表达式

- 根据流程表可以作出激励状态、输出函数的卡诺图，根据卡诺图即可得到化简后的激励状态和输出函数的最简表达式。
- 例如由表6.30，画出激励状态和输出函数的卡诺图（图6.29）。根据卡诺图得到激励状态和输出函数的最简表达式：

$$Y_2 = \overline{x_2} \cdot x_1 + x_1 \cdot y_2 + x_1 \cdot y_1$$

$$Y_1 = \overline{x_2} \cdot x_1 + x_2 \cdot y_1$$

$$Z = y_2 \cdot y_1$$

表 6.30 二进制流程表

| 二次状态 $y_2 \ y_1$ | 激励状态 $Y_2 Y_1$ | | | | 输 出 Z |
|---------------------|----------------|----------------|----------------|----------------|------------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ | |
| 0 0 | 00 | 11 | 10 | 00 | 0 |
| 0 1 | 00 | 11 | 01 | 01 | 0 |
| 1 1 | 00 | 11 | 11 | 01 | 0 |
| 1 0 | 00 | 11 | 10 | 00 | 1 |

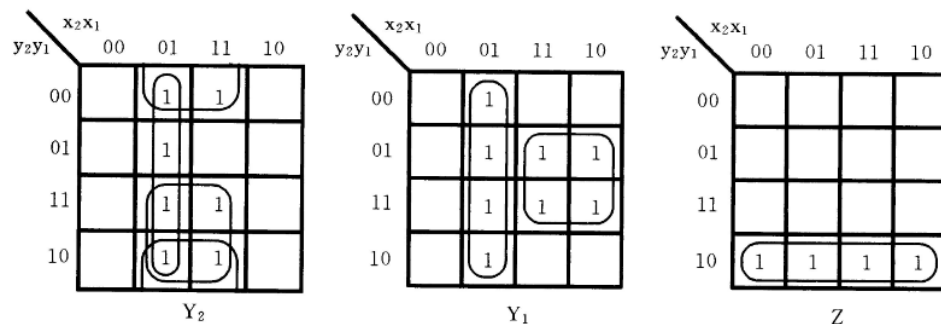


图 6.29 激励状态和输出函数卡诺图

- **例6.11**：用与非门设计一个**单脉冲发生器**，电路结构框图如图6.30所示。该电路有两个输入端 x_2 、 x_1 和一个输出端 Z 。 x_2 接时钟脉冲源， x_1 接手动控制按钮。当不按下按钮时（ $x_1=0$ ）， x_2 端的脉冲被封锁，输出 Z 为0，无脉冲输出；当按下按钮并释放（ x_1 由0→1，再由1→0）之后，输入端 x_2 出现的第一个完整脉冲被送至输出端 Z ，即用手启动一次，输出一个完整脉冲。电路规定每启动一次，必须在输出一个完整脉冲后才可再次启动。

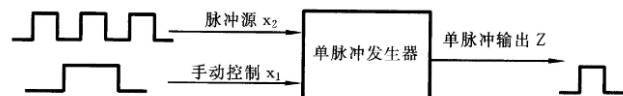


图 6.30 单脉冲发生器的结构框图

- 解：
 - （1）建立**原始流程表**。典型输入/输出**时间图**如图6.31所示，根据图6.31可知有7个状态（稳态），根据时间图可以建立**原始流程表**（表6.31）。因为电路规定每启动一次，必须在输出一个完整脉冲后才可再次启动；因此，处在稳态⑤时输入取值不允许从10→11，处在稳态⑥时输入取值不允许从00→01，处在稳态⑦时输入取值不允许从10→11。
 - 稳态①： $x_2x_1=00$ （启动信号和脉冲信号均未出现）， $Z=0$
 - 稳态②： $x_2x_1=10$ （有脉冲出现但没有启动信号）， $Z=0$
 - 稳态③： $x_2x_1=01$ （有启动信号但无脉冲信号）， $Z=0$
 - 稳态④： $x_2x_1=11$ （启动信号和脉冲信号同时出现）， $Z=0$
 - 稳态⑤： $x_2x_1=10$ （ x_2 端有启动信号结束后的不完整脉冲）， $Z=0$
 - 稳态⑥： $x_2x_1=00$ （启动信号已结束，但第一个完整脉冲尚未出现）， $Z=0$
 - 稳态⑦： $x_2x_1=10$ （出现了启动信号结束后的第一个完整脉冲，产生一个完整输出脉冲）， $Z=1$

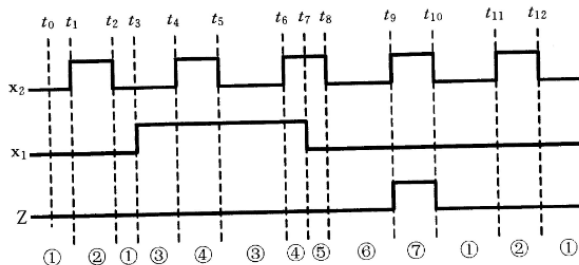


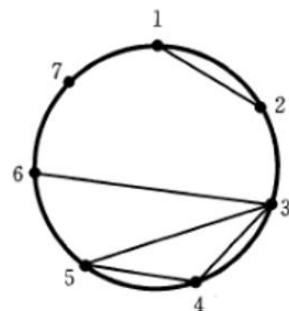
图 6.31 典型时间图

表 6.31 原始流程表

| 二次状态 y | 激励状态 Y/输出 Z | | | |
|-----------|-------------|-------------|-------------|-------------|
| | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 1 | ①/0 | 3/0 | d/d | 2/0 |
| 2 | 1/0 | d/d | 4/0 | ②/0 |
| 3 | 6/0 | ③/0 | 4/0 | d/d |
| 4 | d/d | 3/0 | ④/0 | 5/0 |
| 5 | 6/0 | d/d | d/d | ⑤/0 |
| 6 | ⑥/0 | d/d | d/d | 7/d |
| 7 | 1/d | d/d | d/d | ⑦/1 |

- (2) 化简原始流程表。
 - 首先作出隐含表（图6.32(a)），由隐含表得到相容行对：(1,2)、(3,4)、(3,5)、(3,6)、(4,5)。
 - 由相容行对，画状态合并图（图6.32(b)），得到最大相容行类：(1,2)、(3,4,5)、(3,6)、(7)。
 - 从最大相容行类中选择最小闭覆盖：(1,2)、(3,4,5)、(6)、(7)。
 - 分别用A、B、C、D分别代表(1,2)、(3,4,5)、(6)、(7)，即可得到最简流程表（表6.32）。

(a)



(b)

表 6.32 最简流程表

| 二次 状态 y | 激励状态 Y/输出 Z | | | |
|---------------|----------------|----------------|----------------|----------------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A | Ⓐ/0 | B/0 | B/0 | Ⓐ/0 |
| B | C/0 | Ⓑ/0 | Ⓑ/0 | Ⓑ/0 |
| C | Ⓒ/0 | d/d | d/d | D/d |
| D | A/d | d/d | d/d | Ⓓ/1 |

图 6.32 隐含表和状态合并图

- (3) **状态编码**。根据最简流程表作出**状态相邻图**（图6.33），分别用00、01、11、10表示A、B、C、D，则得到**二进制流程表**（表6.33）。
- (4) 确定**激励状态和输出函数**表达式。根据二进制流程表画出卡诺图（图6.34），根据卡诺图得到简化后的逻辑公式：
 - $Y_2 = x_2 \cdot y_2 + \overline{x_2} \cdot \overline{x_1} \cdot \overline{y_1} = \overline{((x_2 \cdot y_2) \cdot ((\overline{x_2} \cdot \overline{x_1} \cdot \overline{y_1})))}$
 - $Y_1 = x_1 + x_2 \cdot y_1 + \overline{y_2} \cdot \overline{y_1} = \overline{((\overline{x_1} \cdot ((\overline{x_2 \cdot y_1} \cdot \overline{y_2} \cdot \overline{y_1})))}$
 - $Z = x_2 \cdot y_2 = \overline{((\overline{x_2 \cdot y_2}))}$
- (5) 根据逻辑公式，画出**电路图**（图6.35）。

表 6.32 最简流程表

| 二次状态 y | 激励状态 Y/输出 Z | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A | Ⓐ/0 | B/0 | B/0 | Ⓐ/0 |
| B | C/0 | Ⓑ/0 | Ⓑ/0 | Ⓑ/0 |
| C | Ⓒ/0 | d/d | d/d | D/d |
| D | A/d | d/d | d/d | Ⓓ/1 |

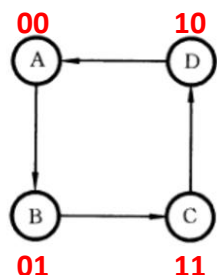


图 6.33 状态相邻图

表 6.33 二进制流程表

| 二次状态 $y_2 \ y_1$ | | | 激励状态 $Y_2 Y_1$ /输出 Z | | | |
|---------------------|---|---|----------------------|----------------|----------------|----------------|
| | | | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A | 0 | 0 | Ⓐ/0 | 01/0 | 01/0 | Ⓐ/0 |
| B | 0 | 1 | 11/0 | Ⓑ/0 | Ⓑ/0 | Ⓑ/0 |
| C | 1 | 1 | Ⓒ/0 | d/d | d/d | 10/d |
| D | 1 | 0 | 00/d | d/d | d/d | Ⓓ/1 |

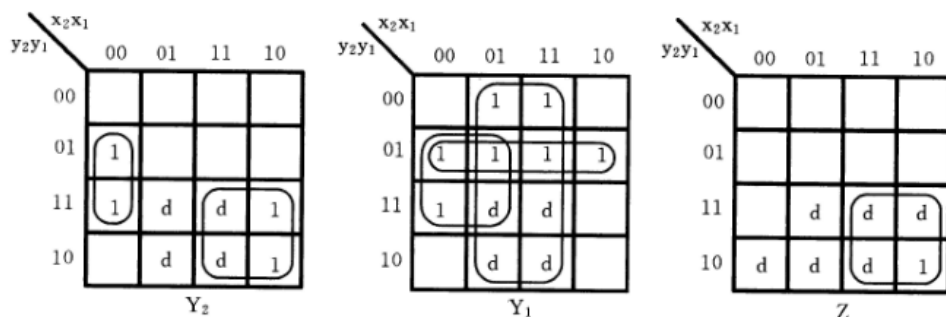


图 6.34 激励函数和输出函数卡诺图

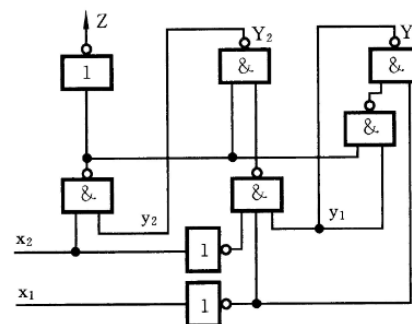


图 6.35 逻辑电路

本章小结

- 异步时序逻辑电路的**特点**:
 - ① 电路中没有统一的时钟脉冲信号同步，电路状态的改变是外部输入信号变化直接作用的结果。
 - ② 在状态转移过程中，各存储元件的状态变化不一定发生在同一时刻，不同状态的维持时间不一定相同，并且可能出现非稳定状态。
 - ③ 在研究异步时序逻辑电路时，对输入信号的形式有所区分，无论输入信号是脉冲信号还是电平信号，对其变化过程均有一定约束。
- 根据**输入信号形式**，异步时序逻辑电路**分为**:
 - ① **脉冲**异步时序逻辑电路：其存储电路由触发器组成（可以是钟控触发器，也可以是非钟控触发器），电路输入信号为脉冲信号。
 - ② **电平**异步时序逻辑电路：其存储电路由延迟元件组成（可以是专用的延迟元件，或者利用电路本身固有的延迟），通过延迟加反馈实现记忆功能，电路输入信号为电平信号。
- 脉冲异步时序逻辑电路的输入脉冲信号（即**输入信号x**）必须满足如下的**约束条件**:
 - ① 输入脉冲的宽度，必须保证触发器可靠翻转。
 - ② 输入脉冲的间隔，必须保证前一个脉冲引起的电路响应完全结束后，后一个脉冲才能到来。
 - ③ **不允许在两个或两个以上输入端同时出现脉冲**（这可能导致电路产生错误的状态转移）。
- **Mealy**型脉冲异步时序逻辑电路的输出通常是**脉冲信号**（因为输出是输入和状态的函数，而输入是脉冲信号，因此输出也会是脉冲信号）。
- **Moore**型脉冲异步时序逻辑电路的输出是**电平信号**（因为输出仅仅是状态的函数，因此输出是电平信号）。

- 脉冲异步时序逻辑电路的分析过程与同步时序逻辑电路（第5章）基本相同，分析步骤如下：
 - ① 根据电路，写出输出函数和激励函数表达式。
 - ② 根据激励函数以及触发器的功能表，列出电路次态真值表（或次态方程组）。
 - ③ 根据次态真值表（或次态方程组）以及输出函数，作出状态表和状态图。
 - ④ 拟定一典型输入序列，画出时间图，并用文字描述电路的逻辑功能。
- 脉冲异步时序逻辑电路分析与同步时序逻辑电路分析的区别：
 - ① 当存储元件采用钟控触发器时，应将触发器的时钟控制端作为激励函数处理。
 - ② 由于不允许两个或两个以上输入端同时出现脉冲，加之输入端无脉冲出现时电路状态不会发生变化。因此，分析时可以排除这些情况，从而使分析过程中使用的图、表简化。
- 脉冲异步时序逻辑电路的设计过程与同步时序逻辑电路（第5章）基本相同，设计步骤如下：
 - ① 根据要求形成原始状态图和原始状态表。
 - ② 对原始状态表进行化简，得到最简状态表（最小状态表）。
 - ③ 对最简状态表进行状态编码，得到二进制状态表。
 - ④ 确定触发器的类型和数目。
 - ⑤ 根据二进制状态表和触发器的激励表，确定激励函数和输出函数。
 - ⑥ 根据激励函数和输出函数，画出电路图。
- 脉冲异步时序逻辑电路设计与同步时序逻辑电路设计不同的地方：
 - ① 由于不允许两个或两个以上输入端同时为1，所以在形成原始状态图和原始状态表时，若有多个输入信号，则只需考虑多个输入信号中仅一个为1的情况，从而使问题的描述得以简化。此外，在确定激励函数和输出函数时，可将两个或两个以上输入同时为1的情况，作为无关条件处理，这将有利于函数的简化。
 - ② 由于电路中没有统一的时钟脉冲，因此当存储电路采用钟控触发器时，触发器的时钟端是作为激励函数处理的，因此需要对触发器的激励表做适当的修改（表6.5至6.8）。在要求触发器状态保持不变时（0→0，或1→1），通常采用CP=0、输入端=d的方式进行处理。

- 电平异步时序逻辑电路的结构模型：

- 包括组合电路和存储电路，存储电路由反馈回路中的延迟元件构成，延迟元件一般不用专门插入延迟线，而是利用组合电路本身固有的分布延迟在反馈回路中的“集总”。

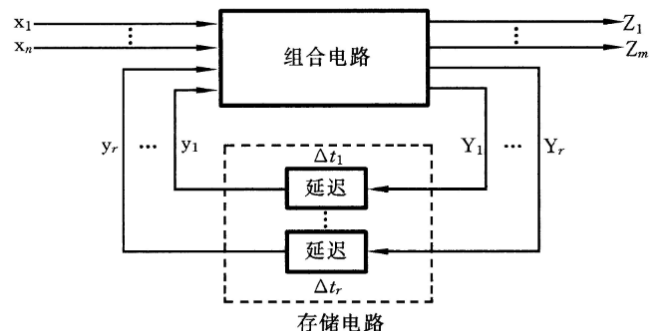


图 6.14 电平异步时序逻辑电路的结构模型

- 电平异步时序逻辑电路的分析步骤：

- ① 根据电路写出输出函数和激励函数表达式。
- ② 作出流程表。
- ③ 作出总态图或时间图。
- ④ 说明电路逻辑功能。

- 电平异步时序逻辑电路的竞争（类似组合逻辑电路中的险象）：

- 实际电路中，各反馈回路之间的延迟是不相同的，这样可能会导致竞争。
- 所谓竞争是指当输入信号变化引起电路中两个或两个以上状态变量发生变化时，由于各反馈回路延迟时间的不同，使状态变量的变化有先有后，而导致不同状态响应过程的现象。
- 若竞争的结果最终能到达预定的稳态，则称为非临界竞争（状态转移可预测）。若竞争的结果使电路到达不同的稳态，即状态转移不可预测，则称为临界竞争（状态转移不可预测）。

- 电平异步时序逻辑电路的**设计步骤**:

- ① 根据设计要求, 建立原始流程表。
- ② 化简原始流程表, 得到最简流程表。
- ③ 状态编码, 得到二进制流程表。
- ④ 根据二进制流程表, 求出激励状态和输出函数表达式。
- ⑤ 画出逻辑电路图。

- **建立原始流程表**。具体过程如下:

- (1) 画出典型输入/输出时间图, 并设立相应状态, 需要注意以下**3**点:
 - ① 符合题意, 即正确体现设计要求。
 - ② 满足电平异步时序逻辑电路不允许两个或两个以上输入信号同时改变的约束条件。
 - ③ 尽可能反映输入信号在各种取值下允许发生的变化。
- (2) 建立原始流程表, 具体包括:
 - ① 画出原始流程表, 并填入稳定状态和相应输出。
 - ② 填入非稳定状态并指定非稳定状态下的输出, 完善流程表。
 - ③ 填入无关状态和无关输出。

- **化简原始流程表。**原始流程表的化简过程与不完全确定状态表（见第5章的5.3.3小节）的化简过程类似，其一般步骤如下：
 - ① 作隐含表，找出相容行。
 - ② 作合并图，求出最大相容行类。
 - ③ 从相容行类中选择一个最小闭覆盖。
 - ④ 作出最简流程表。
- **状态编码。**为了避免反馈回路之间的临界竞争，保证电路可靠地实现预定功能。通常有3种状态编码方法：
 - ① 相邻状态，相邻分配。
 - ② 增加过渡状态，实现相邻分配。
 - ③ 允许非临界竞争，避免临界竞争。

习题 (P179–P181)

- 6.1
- 6.4
- 6.5
- 6.6
- 6.9
- 6.11
- 6.13
- 6.15

习题 (P179-P181)

6.1 分析图 6.36 所示脉冲异步时序逻辑电路。

- (1) 作出状态表和状态图；
- (2) 说明电路逻辑功能。

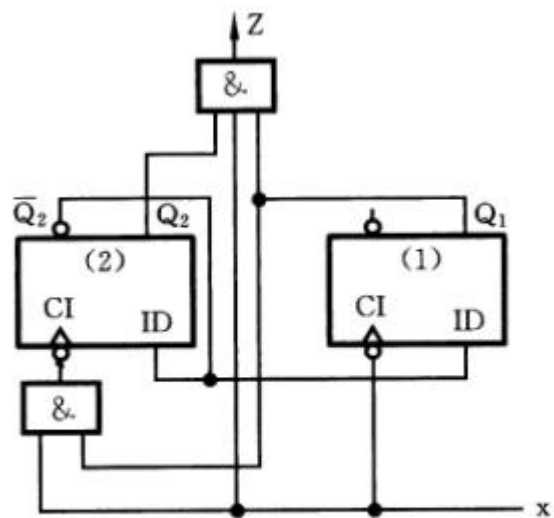


图 6.36 逻辑电路

习题 (P179-P181)

6.2 分析图 6.37 所示脉冲异步时序逻辑电路。

- (1) 作出状态表和时间图；
- (2) 说明电路逻辑功能。

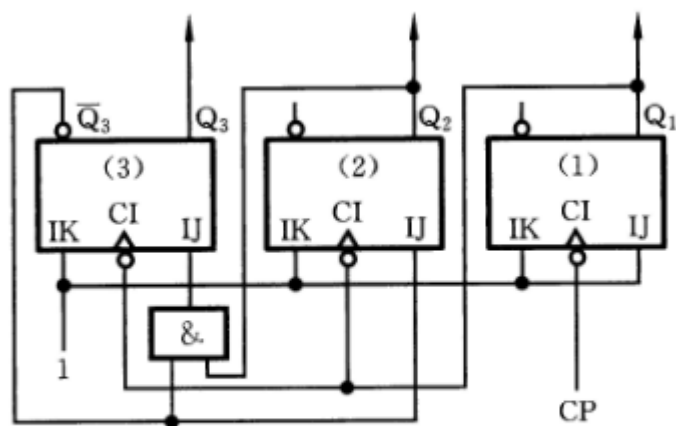


图 6.37 逻辑电路

习题 (P179-P181)

6.3 分析图 6.38 所示脉冲异步时序逻辑电路。

- (1) 作出状态表和状态图；
- (2) 说明电路逻辑功能。

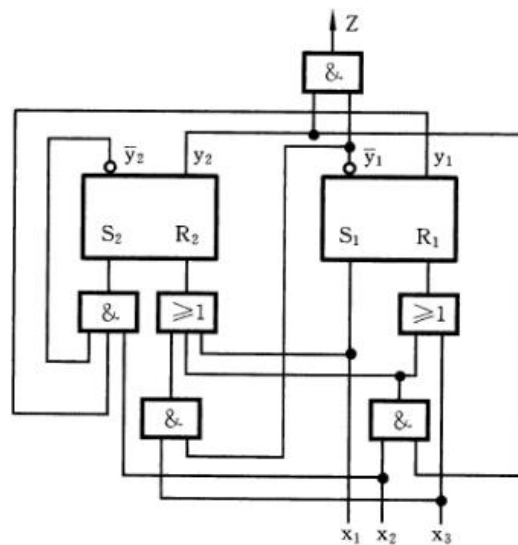


图 6.38 逻辑电路

习题 (P179-P181)

6.4 分析图 6.39 所示脉冲异步时序逻辑电路, 作出时间图并说明该电路逻辑功能。

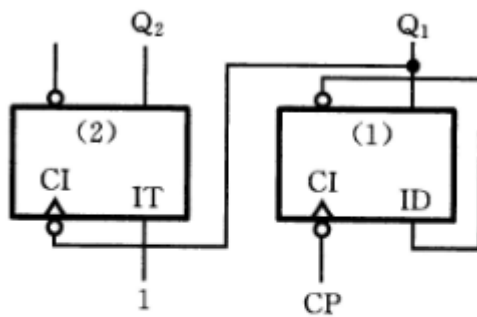


图 6.39 逻辑电路

习题 (P179-P181)

6.5 用 D 触发器作为存储元件,设计一个脉冲异步时序逻辑电路。该电路在输入端 x 的脉冲作用下,实现 3 位二进制减 1 计数的功能,当电路状态为“000”时,在输入脉冲作用下输出端 Z 产生一个借位脉冲,平时 Z 输出 0。

习题 (P179-P181)

6.6 用 T 触发器作为存储元件,设计一个脉冲异步时序逻辑电路,该电路有两个输入 x_1 和 x_2 ,一个输出 Z,当输入序列为“ $x_1-x_1-x_2$ ”时,在输出端 Z 产生一个脉冲,平时 Z 输出为 0。

习题 (P179–P181)

6.7 试用与非门构成的基本 R-S 触发器设计一个模 4 加 1 计数器。

习题 (P179-P181)

6.8 分析图 6.40 所示电平异步时序逻辑电路, 作出流程表。

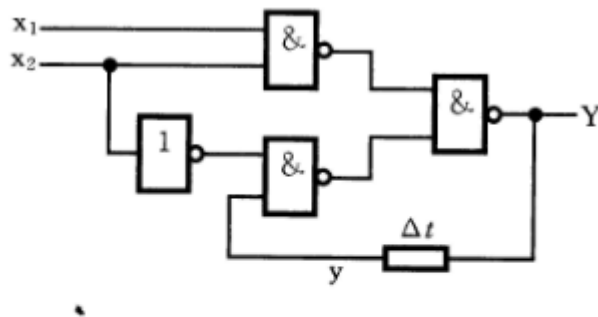


图 6.40 逻辑电路

图 6.41 逻辑电路

习题 (P179-P181)

6.10 某电平异步时序逻辑电路的流程表如表 6.34 所示。作出输入 x_2x_1 变化序列为 $00 \rightarrow 01 \rightarrow 11 \rightarrow 10 \rightarrow 11 \rightarrow 01 \rightarrow 00$ 时的总态 (x_2x_1, y_2y_1) 响应序列。

表 6.34 流程表

| 二次状态 $y_2 \quad y_1$ | 激励状态 Y_2Y_1 /输出 Z | | | |
|-------------------------|-----------------------|-------------|-------------|-------------|
| | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0 0 | 00/0 | 01/0 | 01/0 | 10/0 |
| 0 1 | 00/0 | 01/0 | 01/0 | 11/0 |
| 1 1 | 00/0 | 01/0 | 10/0 | 11/0 |
| 1 0 | 00/d | 00/1 | 10/1 | 10/1 |

习题 (P179-P181)

6.11 某电平异步时序逻辑电路有一个输入 x 和一个输出 Z , 每当输入 x 出现一次 $0 \rightarrow 1 \rightarrow 0$ 的跳变后, 当 x 为 1 时输出 Z 为 1, 典型输入/输出时间图如图 6.42 所示。试建立该电路的原始流程表。

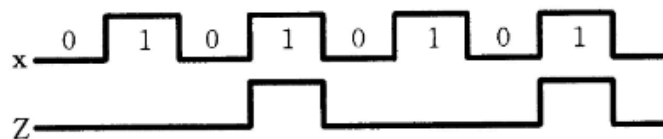


图 6.42 时间图

习题 (P179-P181)

6.12 简化表 6.35 所示的原始流程表。

表 6.35 原始流程表

| 二次状态 y | 激励状态/输出(Y/Z) | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| 1 | ①/0 | 5/d | d/d | 2/d |
| 2 | 1/d | d/d | 3/d | ②/0 |
| 3 | d/d | 5/d | ③/1 | 4/d |
| 4 | 1/d | d/d | 3/d | ④/1 |
| 5 | 1/d | ⑤/0 | 6/d | d/d |
| 6 | d/d | 5/d | ⑥/0 | 4/d |

习题 (P179-P181)

6.13 图 6.43 为某电平异步时序逻辑电路的结构框图。图中，

$$Y_2 = x_2 y_2 + \bar{x}_1 y_2 + x_2 \bar{x}_1 y_1$$

$$Y_1 = x_2 x_1 + \bar{x}_2 \bar{x}_1 y_2 + x_1 y_2 \bar{y}_1$$

$$Z = y_2 y_1$$

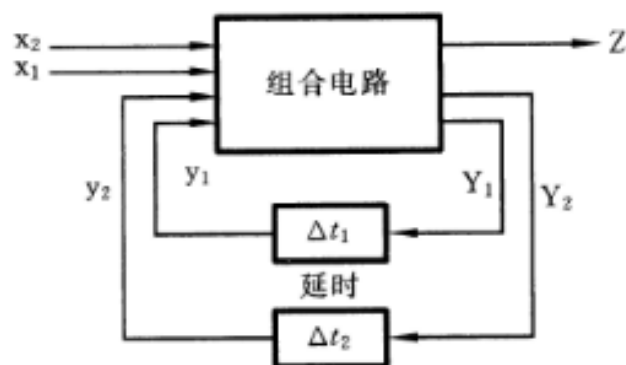


图 6.43 结构框图

习题 (P179-P181)

6.14 对表 6.36 所示的最简流程表进行无临界竞争的状态编码,并确定激励状态和输出函数表达式。

表 6.36 最简流程表

| 二次状态 y | 激励状态 Y/输出 Z | | | |
|-----------|----------------|----------------|----------------|----------------|
| | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A | Ⓐ/0 | Ⓐ/0 | Ⓐ/0 | C/d |
| B | Ⓑ/0 | A/0 | C/d | Ⓑ/0 |
| C | B/d | A/d | Ⓒ/1 | Ⓒ/1 |

习题 (P179–P181)

6.15 某电平异步时序逻辑电路有两个输入 x_1 、 x_2 和一个输出 Z 。当 $x_2=1$ 时, Z 总为 0; 当 $x_2=0$ 时, x_1 第一次从 $0 \rightarrow 1$ 的跳变使 Z 变为 1, 该 1 输出信号一直保持到 x_2 由 $0 \rightarrow 1$, 才使 Z 为 0。试用与非门实现该电路功能。

作业样例

- 1.1 冯·诺依曼结构计算机的基本思想是什么？按此思想设计的计算机硬件系统应由哪些部件组成？它们各有何作用？

• 答：

- 数学家冯·诺依曼提出了计算机制造的三个基本原则，即采用二进制逻辑、程序存储执行（存储程序和程序控制）以及计算机由五个部分组成（运算器、控制器、存储器、输入设备、输出设备），这套理论被称为冯·诺依曼体系结构。
- 计算机硬件系统应由运算器、控制器、存储器、输入设备、输出设备等组成。
- 运算器是一种用于信息加工处理的部件，它对数据进行算术运算和逻辑运算。运算器通常由算术逻辑单元（ALU, Arithmetic and Logic Unit）和一系列寄存器组成。通常将运算器一次运算能处理的二进制位数称为机器字长。现代计算机具有多个寄存器，称为寄存器组。
- 控制器是整个计算机的指挥中心，它可使计算机各部件协调工作。计算机中有两股信息在流动，一股是控制流信息，另一股是数据流信息。控制流信息的发源地是控制器，控制器产生控制流信息的依据来自3个方面：指令寄存器、状态寄存器和时序电路。
- 存储器的主要功能是存放程序和数据，目前计算机的主存储器都是半导体存储器。
- 输入设备就是将信息输入计算机的外部设备，它将人们熟悉的信息形式转换成计算机能接收并识别的信息形式。
- 输出设备就是将计算机运算结果转换成人们和其他设备能接收和识别的信息形式的设备，如字符、文字、图形、图像、声音等。

关于作业提交

- 作业必须**按时提交**（上传到学院的**FTP**服务器上），否则认为是迟交作业；如果期末仍然没有提交，则认为是未提交作业。
 - 作业完成情况成绩=第1次作业提交情况*第1次作业评分+第2次作业提交情况*第2次作业评分+.....+第N次作业提交情况*第N次作业评分。
 - 作业评分：**A**（好）、**B**（中）、**C**（差）三挡。
 - 作业提交情况：按时提交（**1.0**）、迟交（**0.5**）、未提交（**0.0**）。
- 请采用电子版的格式（**PPT文档**）上传到**FTP**服务器上，文件名取“学号+姓名+第X次作业.pptx”。
 - 例如：**11920222202406+刘济华+第6次作业.pptx**
- 下次上课时（**2024年11月18日**）会**随机抽取2位同学**到讲台上汇报作业。
- 第6次作业提交的截止日期为：**2024年11月17日晚上24点**。

Thanks