时序逻辑基础

时序逻辑电路与组合逻辑电路不同。在时序逻辑电路中，状态可以被记忆，电路整体受一个时钟信号 `clk` 的控制，此外往往还有一个异步的重置信号 `rst`。

锁存器和触发器是时序逻辑电路中的两类基本元件，在它们的基础上，寄存器和计数器等元件得以产生。

锁存器

锁存器没有时钟信号。它被用来「锁」住一个值，受一个或多个控制信号的控制。

RS 锁存器

RS 锁存器有两个输入端——R 对应 Reset，用来置 0；S 对应 Set，用来置 1。

对于使用或非门实现的 RS 锁存器（高有效），假设 Q_{n+1} 是其输出的「次态」，也就是下一轮输出电平的状态；Q_n 是现态, 即现在输出的状态，那么 Q_n 和 Q_{n+1} 与 R 和 S 之间的关系如下表：

置0端 R	置1端 S	次态 Q_{n+1}
0	0	Q_n
0	1	1
1	0	0
1	1	—

对于与非门实现的 RS 锁存器（低有效），只需要翻转上表左边两栏中的 0 和 1 就可以了。RS 锁存器在 R 和 S 都无效的时候不会改变输出，在 R 和 S 单独有效的时候置 0 或者置 1，不允许两者都有效。它的状态方程是：

$$Q_{n+1} = S + R'Q_n$$

门控 D 锁存器

门控 D 锁存器有两个输入端——「门」端 G 和输入端 D。门控 D 锁存器的功能表如下：

使能端 G	输入端 D	现态 Q_n	次态 Q_{n+1}
0	X	0	0
0	X	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

如其名所言，门控 D 锁存器在「门」打开（也就是 G 有效）时，输出端将直接拷贝输入端 D；否则，会保持原先的值。因此，它的状态方程是：

$$Q_{n+1} = GD + G'Q_n$$

触发器

触发器与锁存器最大的不同是，触发器有一个时钟信号接入，并在这个时钟信号的控制之下动作。一般触发器的触发方式是边沿触发的，这意味着它只关心时钟信号的变化而不是时钟信号的高低。

D 触发器

D 触发器只有一个输入端，是应用最广的一种触发器。它的输出次态与现态无关；换言之，它的输出仅与输入信号有关。在触发的那一瞬间，它的输出端会直接拷贝输入端，并且保持这个值直到下一次被触发。

对于上升沿触发的 D 触发器，它的状态表如下：

时钟端 CK	输入端 D	现态 Q_n	次态 Q_{n+1}
↑	0	0	0
↑	0	1	0
↑	1	0	1
↑	1	1	1

它的状态方程是 $Q_{n+1} = D$。式子中并没有 Q_n。

如果把 D 触发器的反相输出端与输入端 D 相连，就可以得到一个二分频电路：输出端会产生一个频率是时钟频率一半的方波时钟。

一个小提示：在时序电路中，如果在时钟边沿到来的那一瞬间输入信号也有改变，我们一般认为这个时候采样到的是原值（改变前的值），尽管实际电路中这种情况可能导致不确定的结果。

RS 触发器

RS 触发器有两个输入端 R (Reset) 和 S (Set)，就像 RS 锁存器一样，不同的是它的状态只有在时钟边沿时才会更新。上升沿触发的 RS 触发器的状态表如下：

时钟端 CK	输入端 R	输入端 S	现态 Q_n	次态 Q_{n+1}
↑	0	0	0	0
↑	0	0	1	1
↑	0	1	0	1
↑	0	1	1	1
↑	1	0	0	0
↑	1	0	1	0
↑	1	1	0	—
↑	1	1	1	—

同样地，它不允许 R 和 S 同时为高电平。它的状态方程是 $Q_{n+1} = S + R'Q_n$。

JK 触发器

JK 触发器 ~~里面住着可爱的 JK~~ 得名于它的发明者 Jack Kil，有两个输入端 J 和 K。它可以理解为 RS 触发器的升级版，取消了 RS 触发器中 R 和 S 不能同时为高的限制。对于这种情况，它会翻转现态作为次态。

下降沿触发的 JK 触发器的状态表如下：

时钟端 CK	输入端 J	输入端 K	现态 Q_n	次态 Q_{n+1}
↓	0	0	0	0
↓	0	0	1	1
↓	0	1	0	0
↓	0	1	1	0
↓	1	0	0	1
↓	1	0	1	1
↓	1	1	0	1
↓	1	1	1	0

它的状态方程是 $Q_{n+1} = JQ'_n + K'Q_n$。可以发现它比 RS 触发器多与了一项 Q'_n。事实上，如果忽略 RS 不能为 1 这个条件，那么 R 和 K 端对应，S 和 J 端对应。

T 触发器

T 触发器只有一个输入 T。与 D 触发器完全相反，它的输出完全由现态决定，T 则是控制是否翻转现态。下降沿触发的 T 触发器的状态表如下：

时钟端 CK	输入端 T	现态 Q_n	次态 Q_{n+1}
↓	0	0	0
↓	0	1	1
↓	1	0	1
↓	1	1	0

事实上 T 触发器相当于把 JK 触发器的 J 和 K 接在了一起。因此它的状态方程是 $Q_{n+1} = T \oplus Q_n$。

如果把 T 触发器的 T 端恒置为 1，就得到了 T' 触发器。T' 触发器将在触发的瞬间无条件翻转现态作为次态。

对于上文中的所有触发器，都可以额外增加两个输入端：异步清零端和异步置 1 端。异步清零端有效时，触发器无条件输出 0；异步置 1 端有效时，触发器无条件输出 1。

触发器类型的转换

触发器类型的转换，即用某一种触发器 A 和一些外围逻辑门电路，实现触发器 B 的功能。下面介绍方法。

假设我们需要使用 JK 触发器实现 D 触发器的功能。先写出两个触发器的状态方程：

$$\begin{cases} JK: Q_{n+1} = JQ_n' + K'Q_n \\ D: Q_{n+1} = D \end{cases}$$

我们的目标是得到 J 和 K 端关于 D 的函数（这样才能把 D 转换之后接上去）。因此，我们令上两式右侧相等，得到

$$JQ_n' + K'Q_n = D$$

改写成

$$JQ_n' + K'Q_n = DQ_n + DQ_n'$$

于是

$$\begin{cases} J = D \\ K = D' \end{cases}$$

另一种更好的方式是用卡诺图来转换。本质上，假设 $A_1, A_2, \cdots, A_m$ 是触发器 A 的输入端，$B_1, B_2, \cdots, B_n$ 是触发器 B 的输入端，我们的目标是用 A 来实现 B。那么我们需要找的就是 B_i 和 A_i 与 Q_n 之间的关系。例如，如果要用 RS 触发器实现 JK 触发器，先找出次态现态转换的四种情况以及它们对应的 RS 和 JK 触发器的输入：

$Q_n \longrightarrow Q_{n+1}$	R	S	J	K
$0 \longrightarrow 0$	×	0	0	×
$0 \longrightarrow 1$	0	1	1	×
$1 \longrightarrow 0$	1	0	×	1
$1 \longrightarrow 1$	0	×	×	0

然后画出 R 和 S 关于现态以及 J 和 K 的卡诺图

Q_n \ JK	00	01	11	10
0	0	0	1	1
1	×	0	0	×

S 的

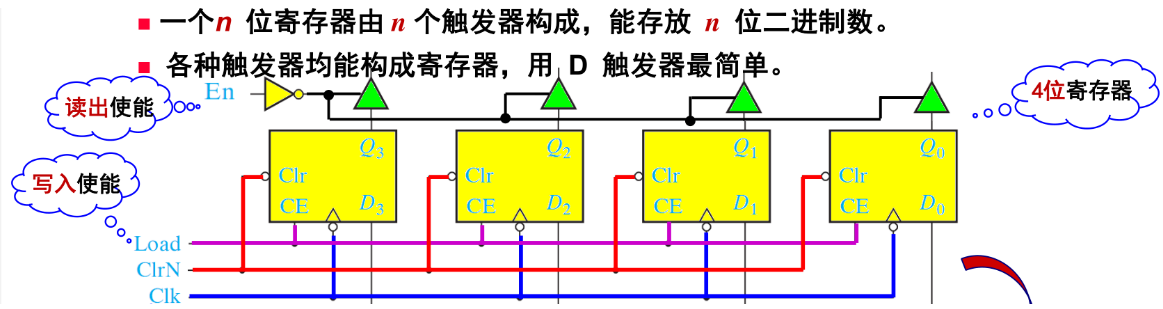
Q_n \ JK	00	01	11	10
0	×	×	0	0
1	0	1	1	0

R 的

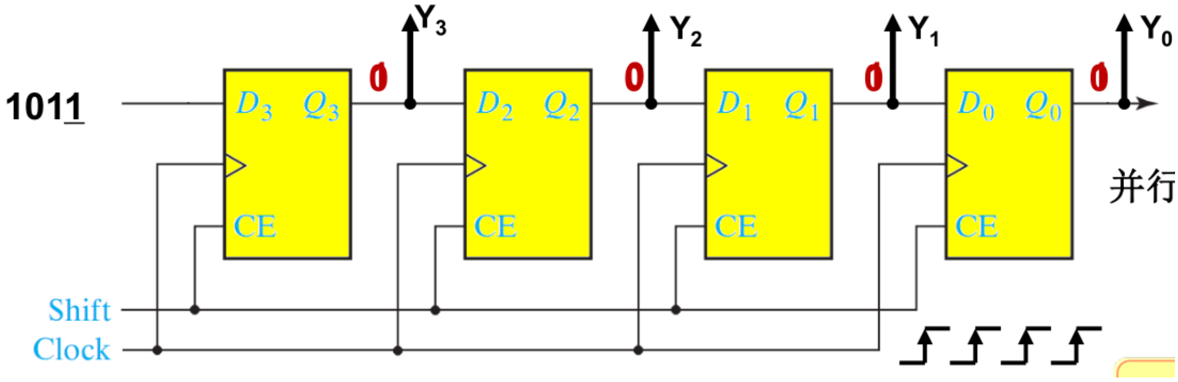
根据卡诺图写出式子就行了。

寄存器

寄存器顾名思义，是用来「寄存」数据的。上文提到的各类触发器中，D 触发器的输出仅与输入有关，故十分用来寄存单个比特。把许多 D 触发器级联起来，就形成了寄存器：



如果把这些 D 触发器的接法改成首尾相接（后一个触发器的 D 端接在前一个触发器的 Q 端），那么就形成了移位寄存器。下图展示的是一种串入并出的右移寄存器：



把整个设计反向，就得到左移寄存器（数据从右边进，不断向左移动）。此外，还有移动方向可变得双向移位寄存器。

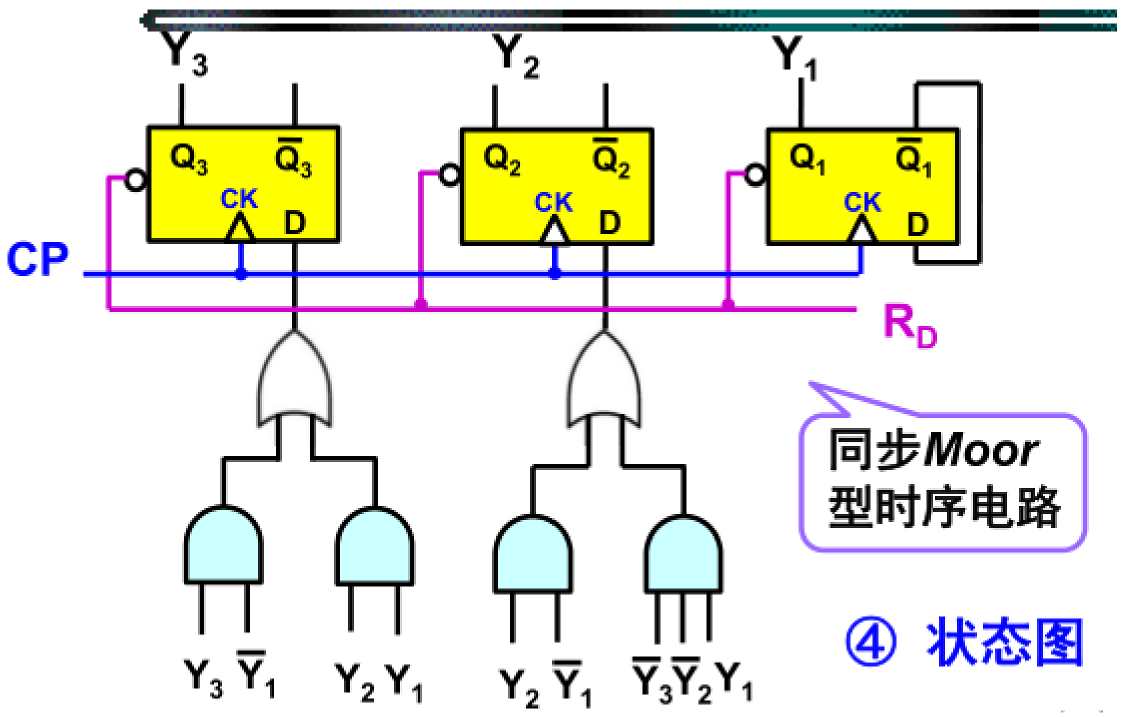
计数器

计数器是一种能够在有限个输出状态之间循环转换的元件。将前文提到的右移寄存器最右边的输出反着接到输入端，就可以得到一个循环往复的计数器。计数器的模指的是计数器能循环的状态的个数。由 n 个触发器构成的 n 位二进制计数器一共有 2^n 个状态，但一般模小于 2^n，也就是说其中只有一些状态能参与循环。

对计数器电路的分析包含下面 5 个部分：

- 输入方程：指的是计数器中所有触发器的**输入**端的接法。
- 输出方程：指的是计数器的所有**输出端和触发器**之间的接法。对于 Moore 型计数器，它的输出就是计数器的输出，因此不需要考虑这个问题。
- 状态转移方程：指的是计数器中各个**触发器的状态转移方程**（即次态输出和现态输出、输入之间的关系）。
- 状态转换表：指的是计数器的**现态**和**次态**之间的转换，以及对应的触发条件。
- 状态图：指的是计数器各状态之间的循环关系。

例如下面的计数器电路



输入方程：

$$\begin{cases} D_3 = Y_3Y_1' + Y_2Y_1 \\ D_2 = Y_2Y_1' + Y_3'Y_2'Y_1 \\ D_1 = Y_1' \end{cases}$$

状态转移方程：

$$\begin{cases} Y_3^{n+1} = D_3 \\ Y_2^{n+1} = D_2 \\ Y_1^{n+1} = D_1 \end{cases}$$

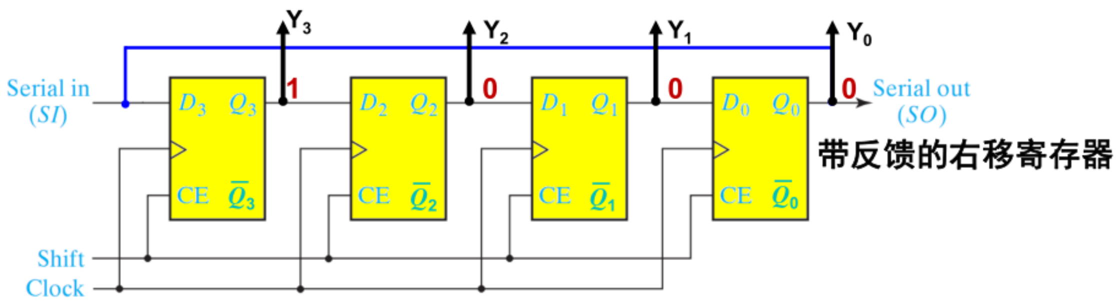
由上面的两组方程可以画出下面的状态转换表：

现态			次态		
Y_3^n	Y_2^n	Y_1^n	Y_3^{n+1}	Y_2^{n+1}	Y_1^{n+1}
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	0	0	0
1	1	0	1	1	1
1	1	1	1	0	0

由这个表容易得到状态转换的关系。（**模 6 加法计数器**）

环形计数器

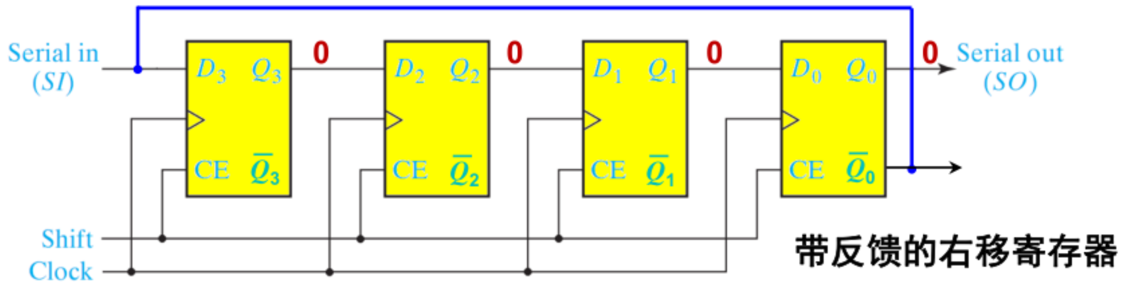
将右移寄存器的最右端输出接到输入，得到一个环形计数器。



n 个触发器制成的环形计数器一共有 n 个可循环的状态，不能自启动，需要预置。

扭环形计数器

将右移寄存器的最右端反相输出接到输入，得到一个扭环形计数器。



n 个触发器制成的扭环形计数器一共有 2n 个可循环的状态，不能自启动，需要预置。

所谓「扭」环正是因为它是用最右端输出的反相信号返回去接到输入。