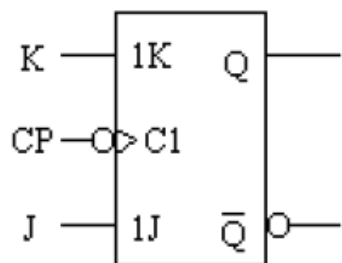


实验九 计数器的设计

原理：

1、

符号：



表达式：

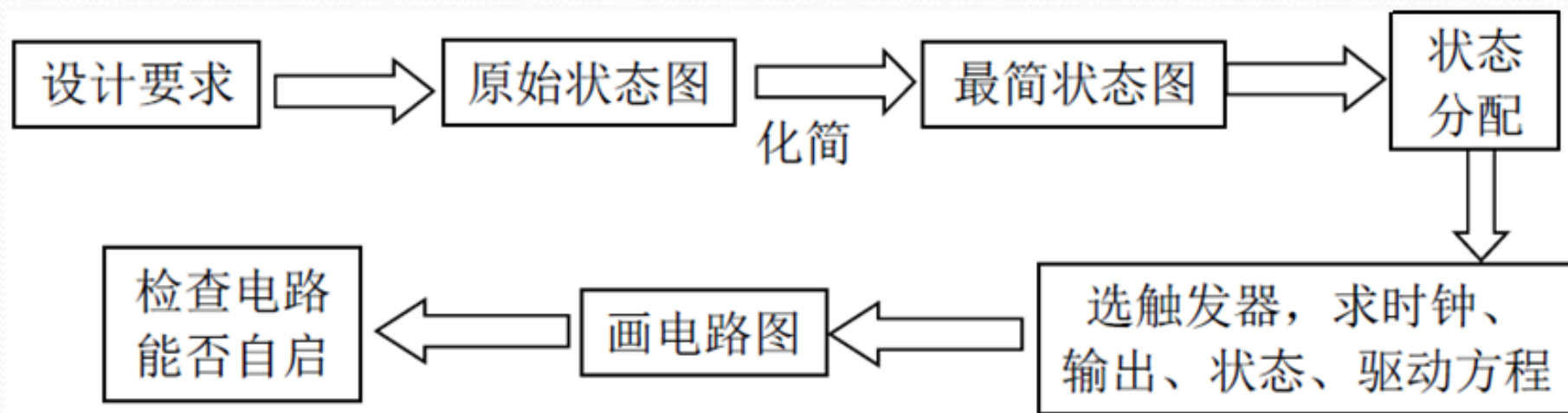
$$Q^{n+1} = J\overline{Q}^n + \overline{K}Q^n$$

JK触发器功能表：

CP	J	K	Q ⁿ	Q ⁿ⁺¹	功能
↓	0	0	0	0	保持
↓	0	0	1	1	保持
↓	0	1	0	0	清零
↓	0	1	1	0	清零
↓	1	0	0	1	置位
↓	1	0	1	1	置位
↓	1	1	0	1	翻转
↓	1	1	1	0	翻转

注意：JK触发器要求接高电平的，不能悬空，否则会导致输出错误，例如清零端。

2、时序逻辑电路的设计流程



3、同步计数器和异步计数器

同步计数器的触发信号是同一个信号。具体来说，每一级的触发器接的都是同一个CLK信号。

异步计数器的每一级的触发器的CLK信号是不同的，触发器状态变化不是同步的。

4、异步触发器：存在触发器逐级延迟问题。同步计数器：各级触发器输出相差小，译码时能避免出现尖峰，但是电路实现较复杂。

内容1

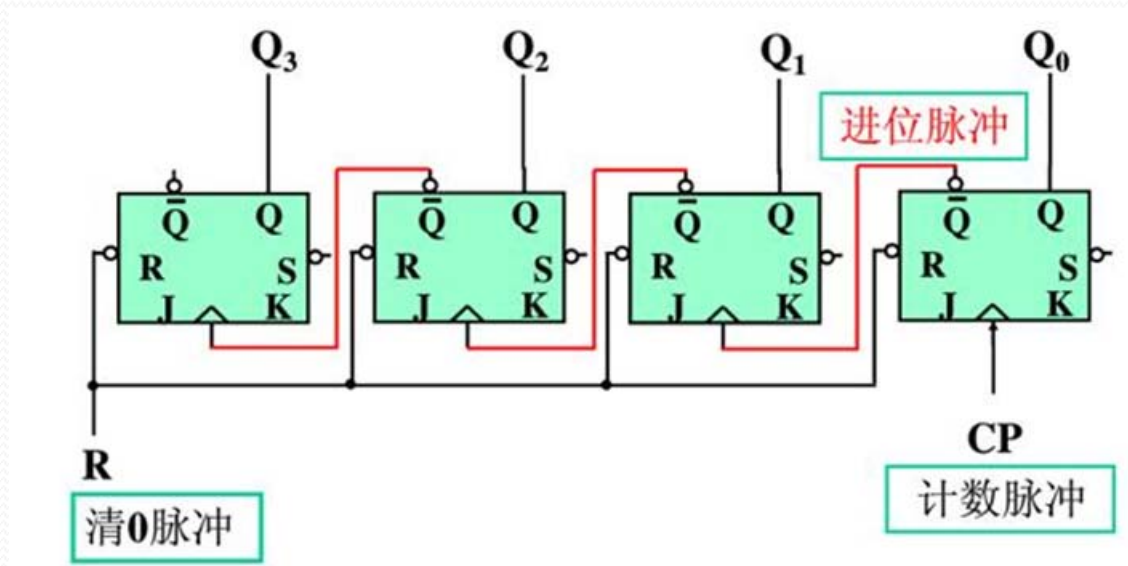
1、使用JK触发器设计一个16进制异步加法计数器，并用逻辑分析仪观察并记录CP和每一位的输出波形。

实验内容1设计提示：

(1) 综合考虑16进制计数器的每一位的状态变化特点和JK触发器的功能表。

(2) 举例：16进制异步减法计数器

异步：每级触发器的CP是不同的信号



16进制减法计数器

Q_3	Q_2	Q_1	Q_0
0	0	0	0
1	1	1	1
1	1	1	0
1	1	0	1
1	1	0	0
1	0	1	1
1	0	1	0
1	0	0	1
1	0	0	0
0	1	1	1
0	1	1	0
0	1	0	1
0	1	0	0
0	0	1	1
0	0	1	0
0	0	0	1

内容2

1、使用JK触发器设计一个16进制同步加法计数器，并用逻辑分析仪观察并记录CP和每一位的输出波形。

实验内容2设计提示：

(1) 按照时序电路的设计步骤得到JK触发器的驱动方程，画出逻辑图，连接电路实现。

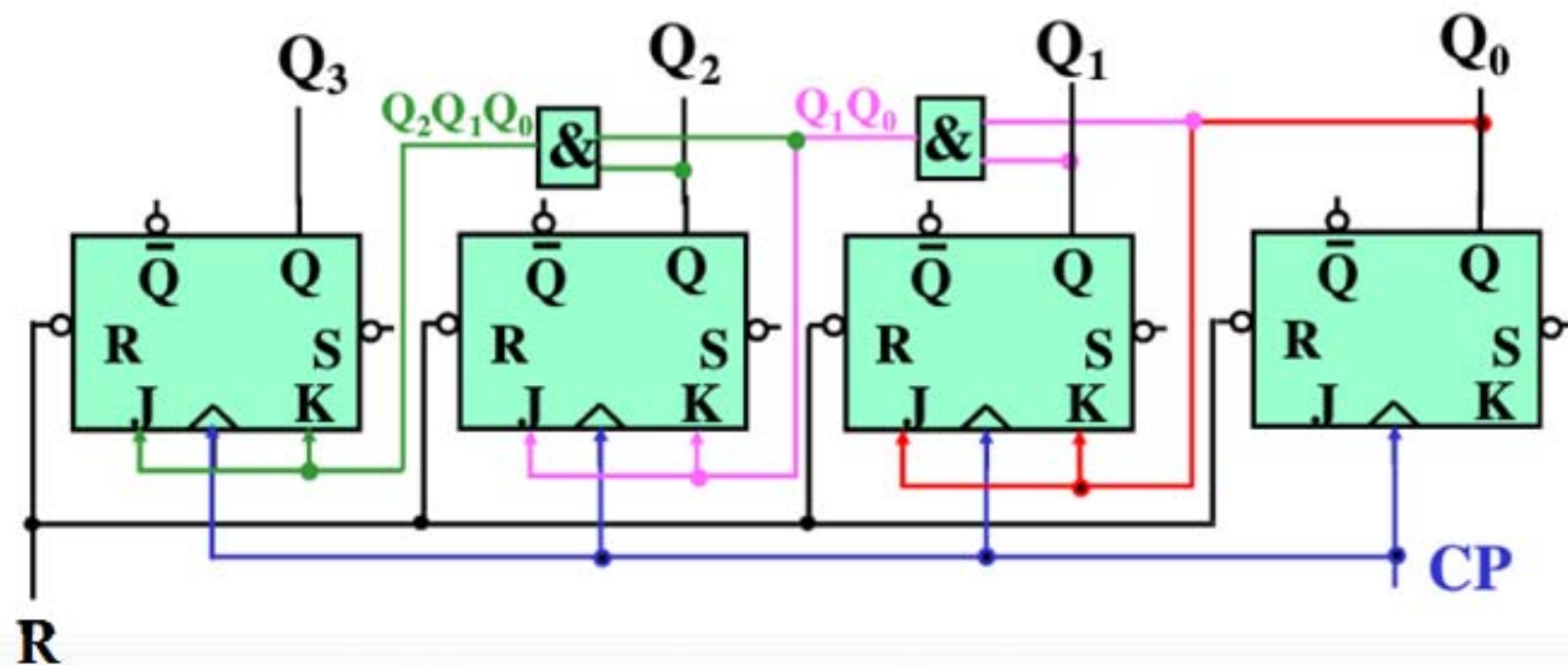
(2) 时序电路的设计：
画状态迁移图->化简次态卡诺图->得出 Q_3, Q_2, Q_1, Q_0 状态方程->结合JK触发器的特征表达式得出每一级JK触发器的驱动方程

16进制加法计数器

Q_3	Q_2	Q_1	Q_0
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

(2) 参考：16进制同步加法计数器

同步：每级触发器CP都接同一个连续脉冲



内容3

- 1、使用JK触发器和门电路设计实现一个二进制四位计数器模仿74LS194功能（详见实验七 表二）。要求在实验箱上设计实现左移或右移功能；在proteus软件上实现置零，保持，左移，右移，并行送数功能。

实验内容3设计提示：

(1) 对比JK触发器和74LS194功能表

表(二) 74LS194 功能表

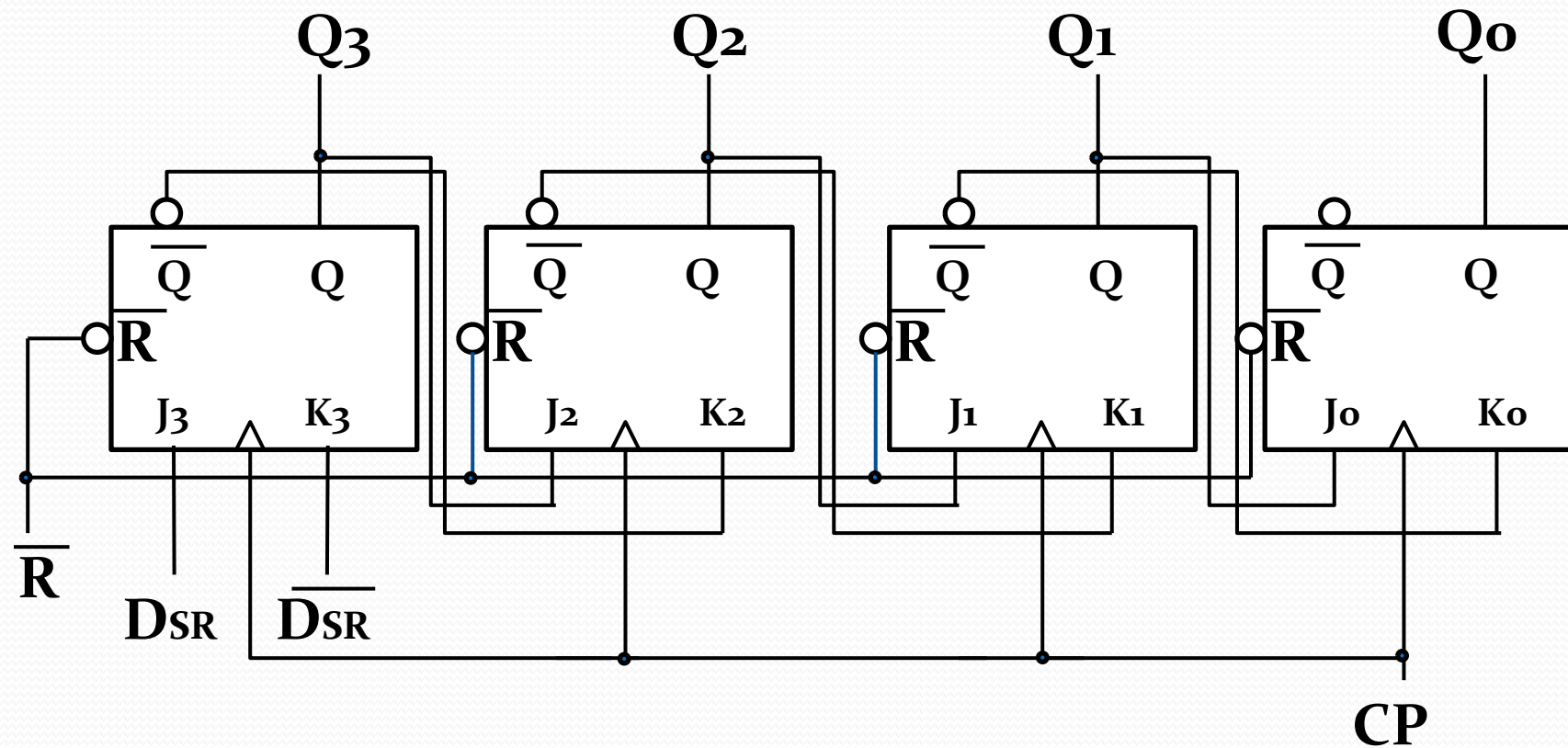
\overline{Cr}	S_1	S_0	工作状态
0	X	X	置零
1	0	0	保持
1	0	1	右移
1	1	0	左移
1	1	1	并行送数

74LS73清零
端实现

JK触发器功能表：

CP	J	K	Q^n	Q^{n+1}	功能
↓	0	0	0	0	保持
↓	0	0	1	1	保持
↓	0	1	0	0	清零
↓	0	1	1	0	清零
↓	1	0	0	1	置位
↓	1	0	1	1	置位
↓	1	1	0	1	翻转
↓	1	1	1	0	翻转

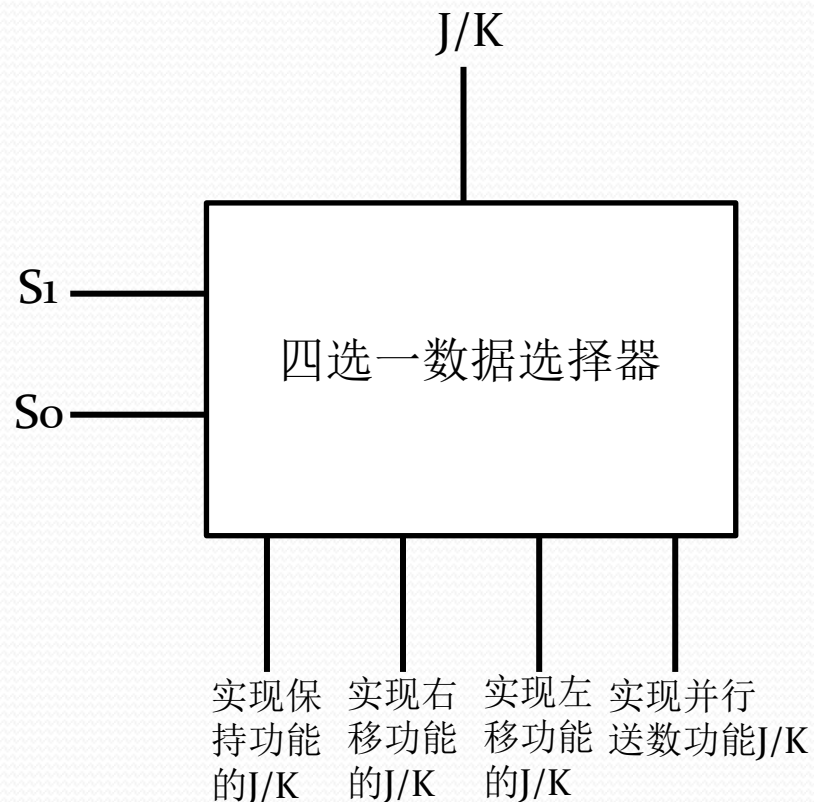
(2) 以右移为例:



第一级触发器（最左位） $J_3 = D_{SR}$ $K_3 = \overline{D_{SR}}$ ，第一级触发器的输出 Q_3 ，按 $J_2 = Q_3, K_2 = \overline{Q_3}$ 下一级触发器，第三、四级触发器也按照第二级触发器接法。

当 D_{SR} 是高电平时，第一级触发器置位.在下一个时钟下降沿到来后，第二级触发器置位...;当 D_{SR} 是低电平时，第一级触发器清零.在下一个时钟下降沿到来后，第二级触发器清零...;从而实现 D_{SR} 的右移。

(3) 在proteus软件上实现74LS194置零，保持，左移，右移，并行送数功能，每一级JK触发器的J和K接入信号可采用4选1数据选择器（例如74LS153 双四选一数据选择器）进行切换，从而达到功能的切换。



S_2 , S_1 是四选一数据选择器的输出控制端，也是实现74LS194的功能切换控制端