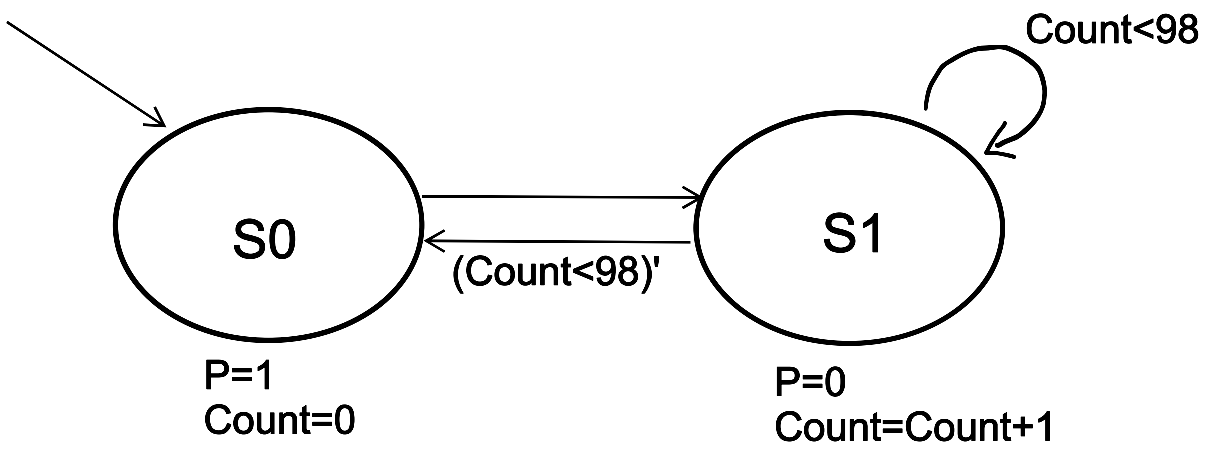
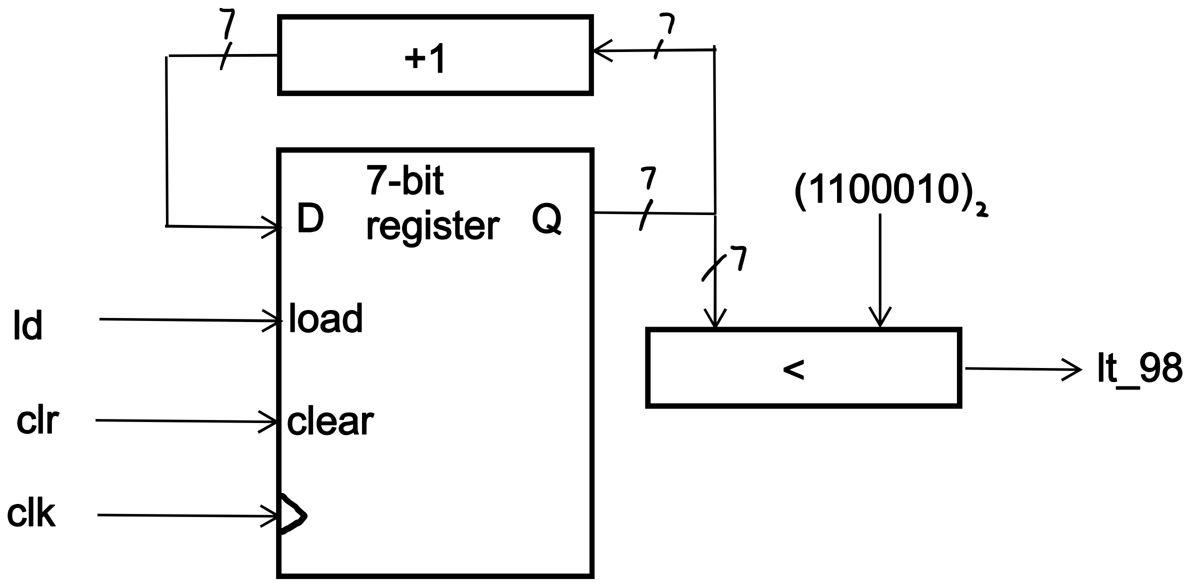
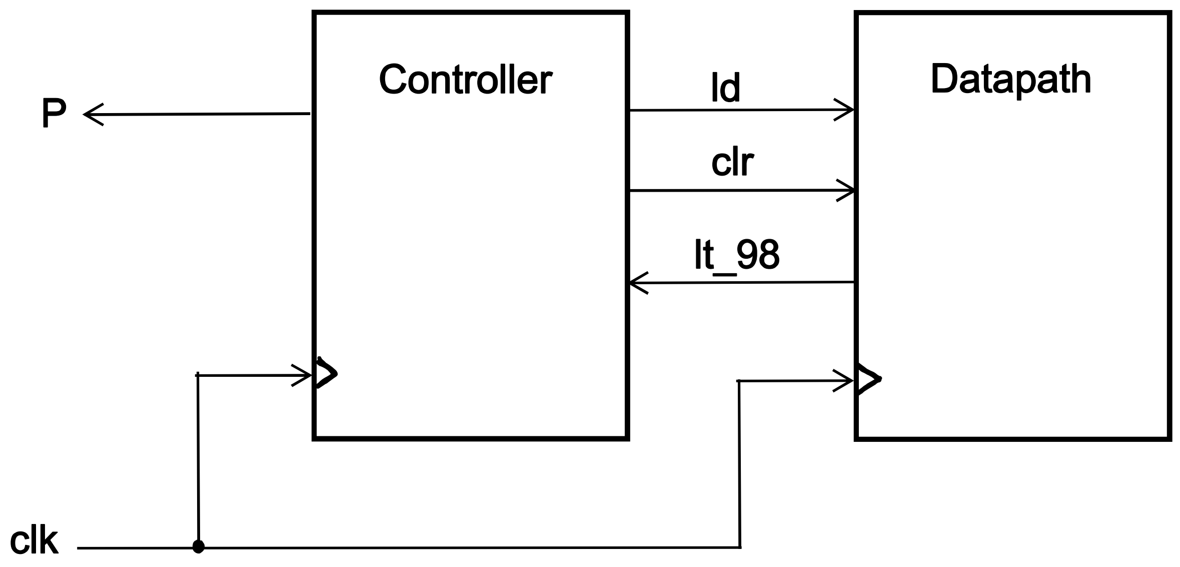
Input: N/A
Output: P
Local Register: count (7 bits)

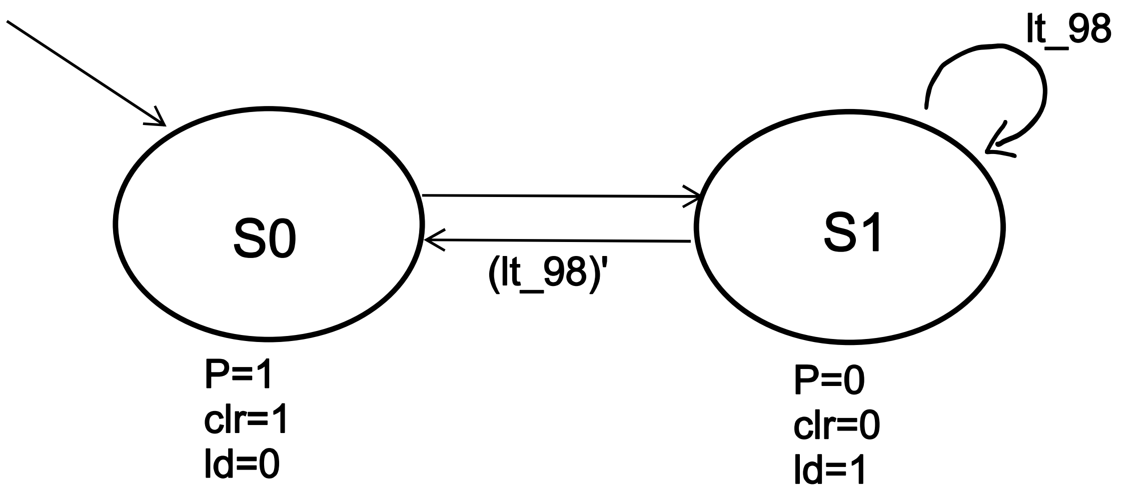


Datapath:





Input: lt_98
Output: P, clr, ld



The reason that you are severely deducted for the wrong judging condition for the clock cycle is as following:

1. You fail to achieve the desired function of this system.

2. We want you to keep in mind that "when a synchronous signal is set, the corresponding operation is not going to happen until the next clock cycle (the next rising edge) and everything is shifted by one clock cycle of time" though it makes the design process more complicated.

So in the FSM here, "ld = 1" is supposed to happen in the next clock cycle. That is, when you first come to S1, "count" is 0 because of the "clr = 1" in S0. It is not until the next clock cycle of S1 that "count" becomes 1 because "ld = 1" in the previous clock cycle. So when "count = 97", S1 has experienced 98 clock cycles and when "count = 98", S1 has experienced 99 clock cycles. Also when "count = 98", "lt_98" becomes 0. So, in the next rising edge (the 100th clock cycle), the FSM comes to S0 and output "P = 1".

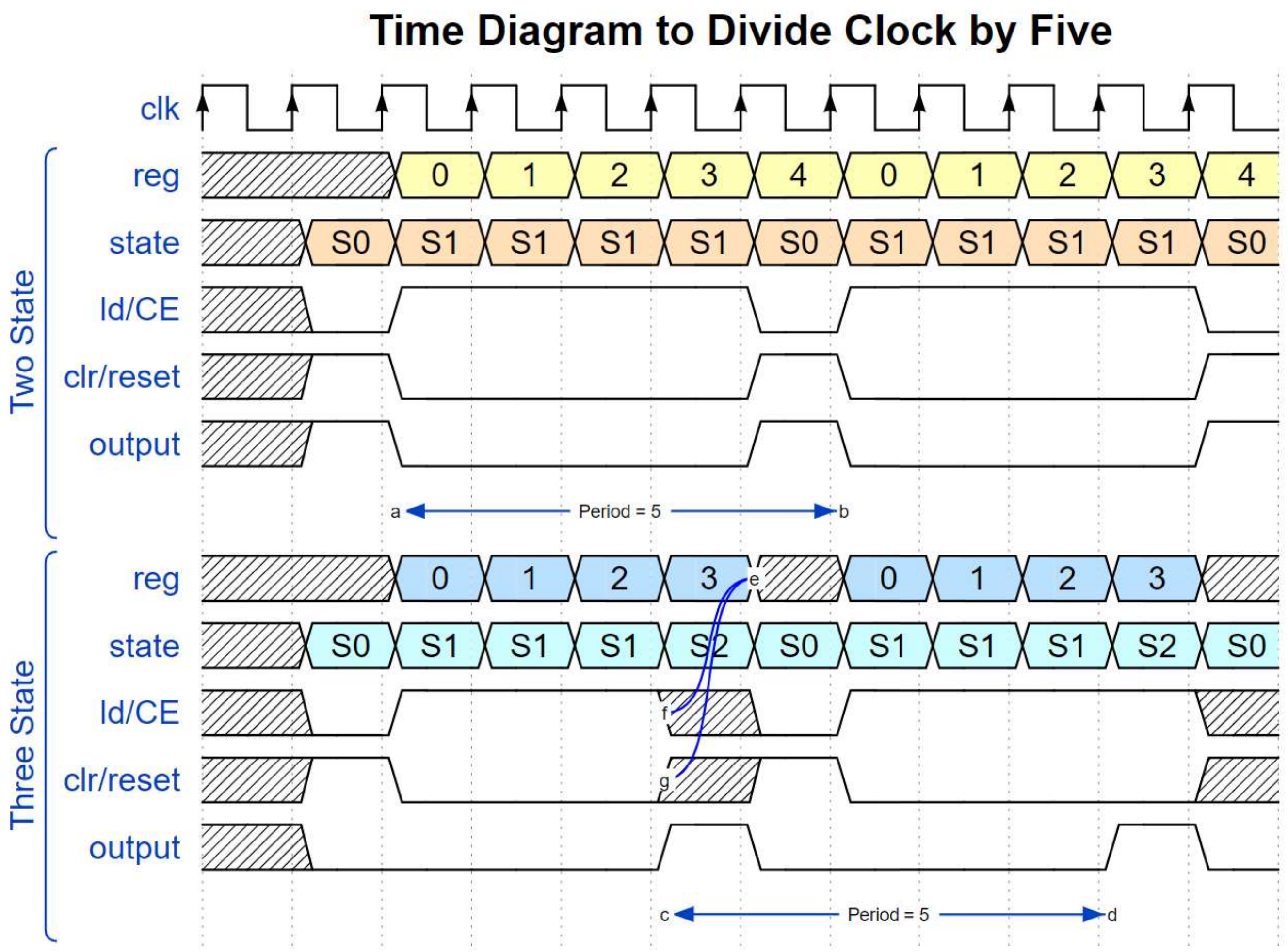
(We have discussed this in your lecture on July 16th. You can refer to the lecture recording of that day at about 01:12:00.)
*However, in the HLSM part, you are not deducted for this (<99 is also fine here) since the HLSM shows more a way of thinking. The deduction counted towards datapath part.

Here's a demo for what happens when you try to divide the clock by five.

If you have two states like this solution, the condition for transferring is "reg<3=5-2"

If you have three states like "clear(init)->count->output", the condition should be "reg<2=5-3"

At e, f, g, the values are X, the reason is that whether or not you clear the register at the state output doesn't influence the result.


Time Diagram to Divide Clock by Five
clk
Two State
reg
state
ld/CE
clr/reset
output
0 1 2 3 4 0 1 2 3 4
S0 S1 S1 S1 S1 S0 S1 S1 S1 S1 S0
a
Period = 5
b
Three State
reg
state
ld/CE
clr/reset
output
0 1 2 3 e 0 1 2 3
S0 S1 S1 S1 S2 S0 S1 S1 S1 S2 S0
f
g
c
Period = 5
d