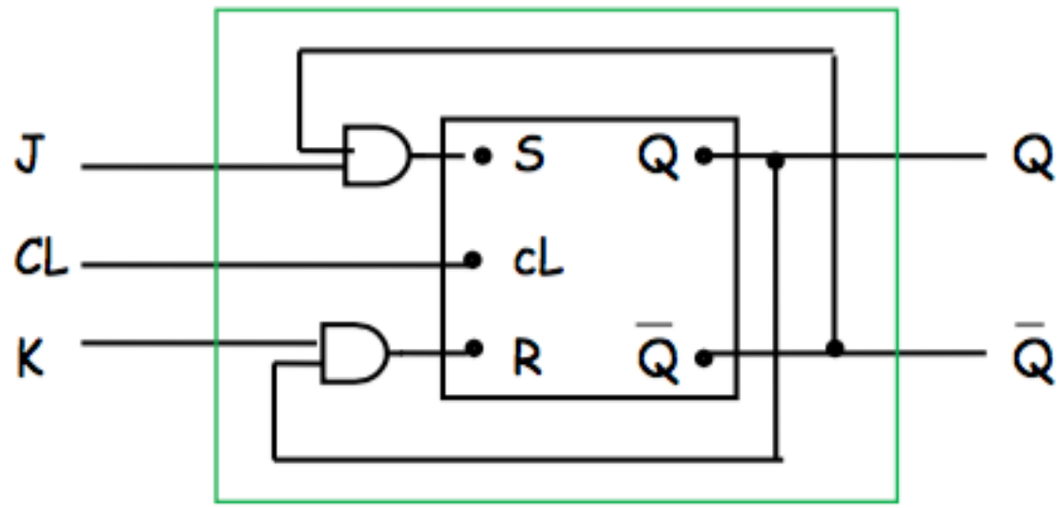
CL=CSCI 160

CLASS 21

HW 21 - assigned

J-K Flip-Flop



Describe its performance. For the truth table consider as inputs J and K only.

Exclude the case J = K = 1 from this truth table, and treat it separately.

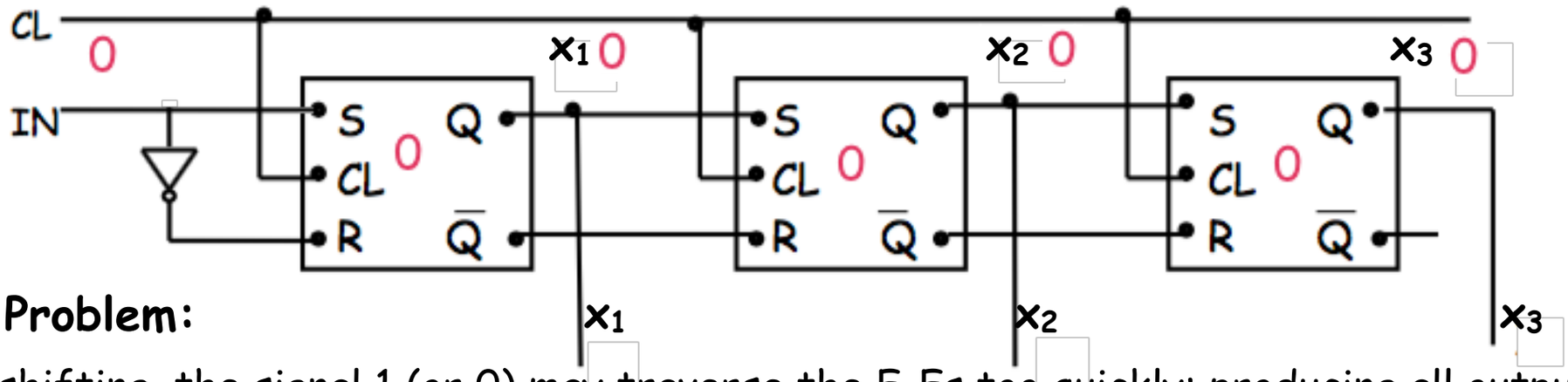
Shifting

Consider the diagram:

We initialize $x_1 = x_2 = x_3 = 0$. For **IN** we input the string of signals **IN** = 0, 1, 0, 0, 0, in sequence.

CL:

It will allow us to see the shifting of the 1 as it progresses through x_1, x_2 , x_3, in time.



Racing Problem:

Instead of shifting, the signal 1 (or 0) may traverse the F-Fs too quickly: producing all outputs =1 (0) at same time.

We will solve this problem later, and for now suppose it is solved, that is:

All S-R-CL Flop-Flops are such that **the time needed for the signal to go through the Flip-Flop is greater than the time needed for CL to run the rising edge** from 0V(=0) to 2V(=1) [when F-F enabled].

IN = 0 1 0 0 0

Time counts	CL	IN	X_1	X_2	X_3
1	1	0	0	0	0
	0	0	0	0	0
2	1	1	0	0	0
	0	1	1	0	0
3	1	0	1	0	0
	0	0	0	1	0
4	1	0	0	1	0
	0	0	0	0	1
5	1	0	0	0	1
	0	0	0	0	0

These F-Fs are described the next slide,

and have the racing problem solved:

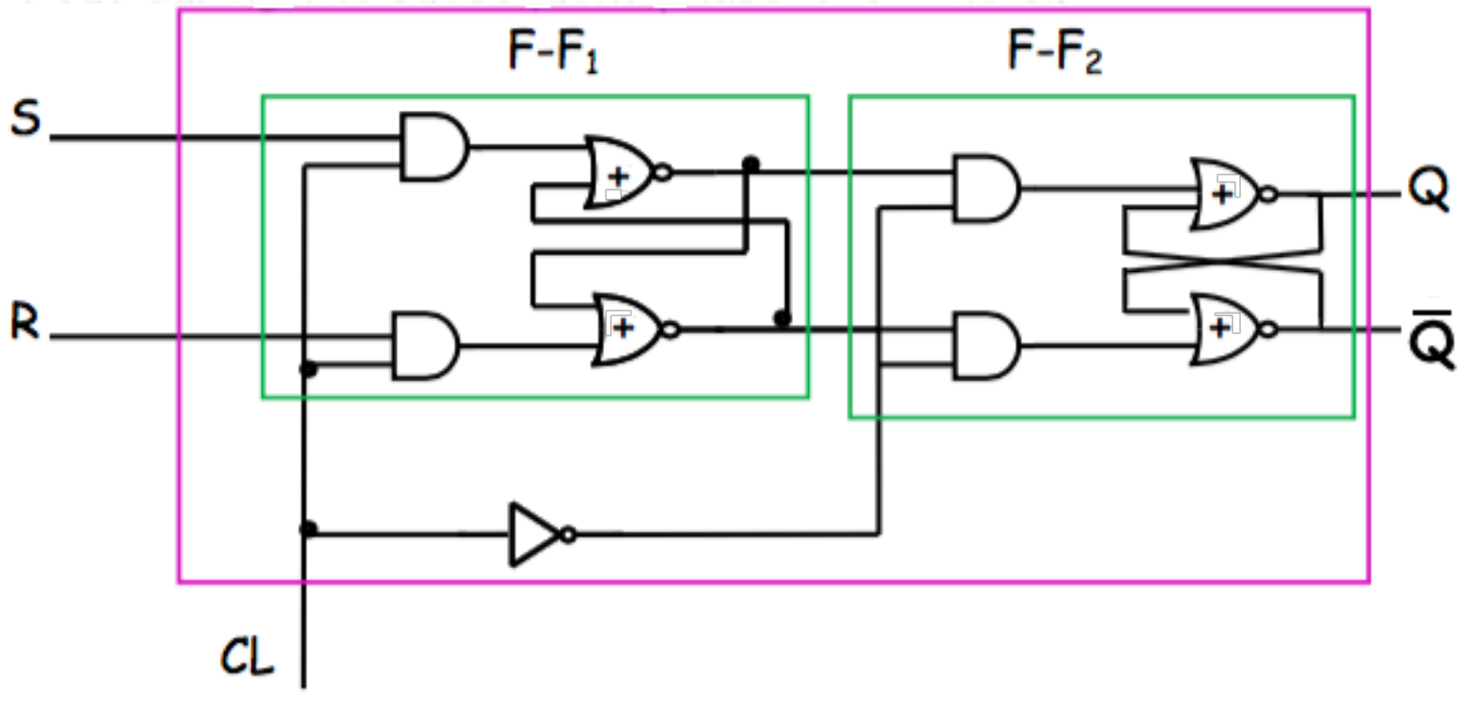
The signal goes

inside the F-F on phase 1

and out of F-F on phase 0 of CL.

No change in outputs during phase 1 of CL.

Racing Problem Solution is the following F-F:



During the phase '1' of CL, the signal goes into the F-F_1 but gets stuck in-between F-F_1 and F-F_2, as F-F_2 is not enabled. During the subsequent phase '0' of CL, F-F_2 gets enabled, and the signal gets finally through F-F_2 and outputs the big F-F [ ] .

The big F-F [ ] solves the 'Racing Problem'