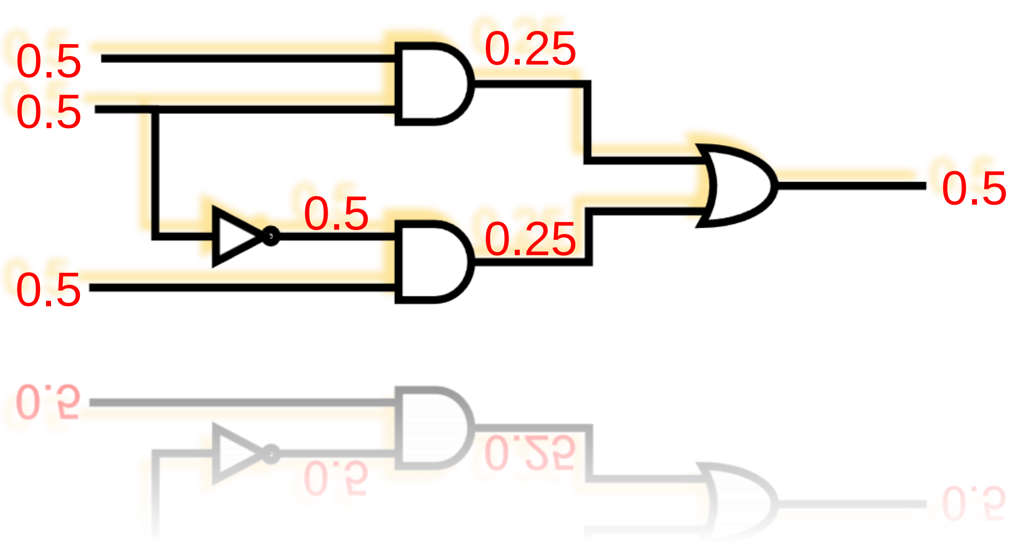
Testability Measure

- Introduction
- SCOAP
  - Combinational
  - Sequential
- COP
- High-level testability measures

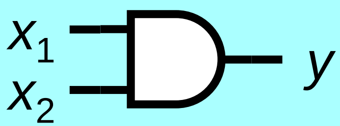
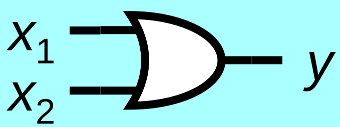
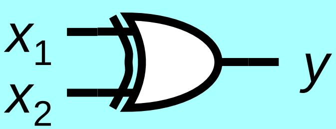
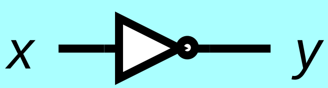
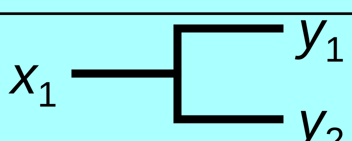


Sequential SCOAP Measures

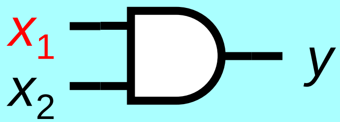
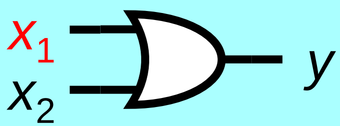
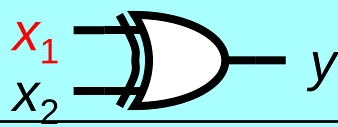
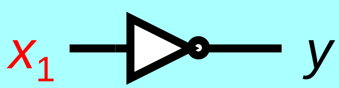
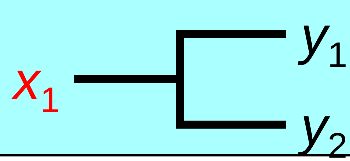
- ***Sequential controllability: $SC^0(N)$, $SC^1(N)$***
  - **Minimum number of FF assignments (number of clock cycles) required to control 0 or 1 on node *N***
  - **smaller number means easier to control**
- ***Sequential observability: $SO(N)$***
  - **Minimum number of FF assignments required to propagate logical value on node *N* to a primary output**
- **NOTE: assume no scan**
  - **Can only control PI, observe PO**
  - **Can NOT control FF, can NOT observe FF**

Sequential SCOAP Measures
# of Clock Cycles Needed

$SC^0(N)$ and $SC^1(N)$

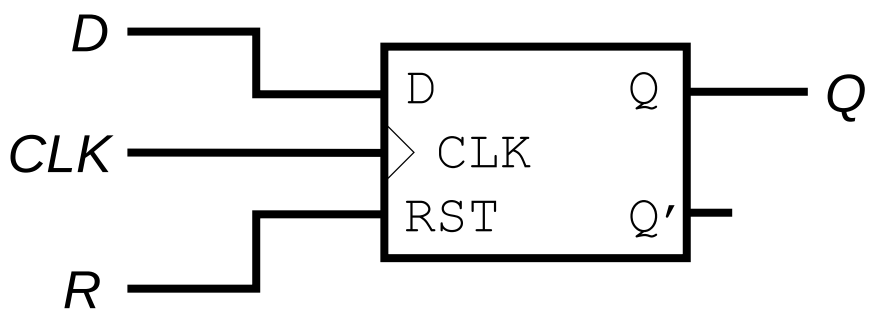
	$SC^0(y)$	$SC^1(y)$
Primary inputs	**0 (not 1)**	**0**
	$\min[SC^0(x_1),SC^0(x_2)]$ $\cancel{+1}$	$SC^1(x_1) + SC^1(x_2)$
	$SC^0(x_1) + SC^0(x_2)$	$\min[SC^1(x_1),SC^1(x_2)]$
	$\min[SC^0(x_1) + SC^0(x_2),$ $SC^1(x_1) + SC^1(x_2)]$	$\min[SC^0(x_1) + SC^1(x_2),$ $SC^1(x_1) + SC^0(x_2)]$
	$SC^1(x)$	$SC^0(x)$
	$SC^0(y_1) = SC^0(y_2) = SC^0(x_1)$	$SC^1(y_1) = SC^1(y_2) = SC^1(x_1)$

SO(N)

	$SO(x_1)$
Primary outputs	**0**
	$SO(y) + SC^1(x_2)\ \cancel{+1}$
	$SO(y) + SC^0(x_2)$
	$SO(y) + \min[SC^0(x_2), SC^1(x_2)]$
	$SO(y)$
	$\min[SO(y_1), SO(y_2)]$

Flip-Flop (Controllability)

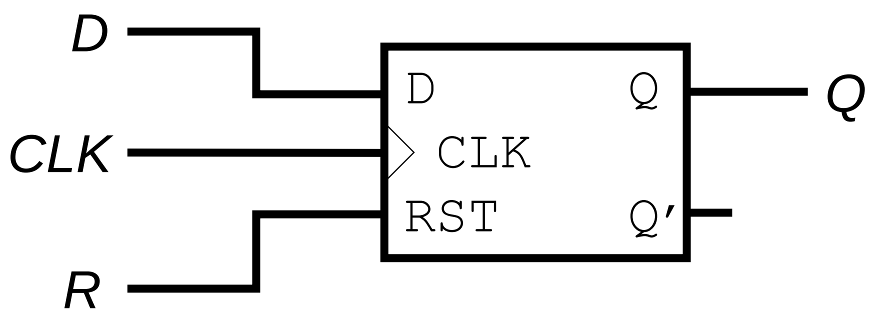
- **Positive edge triggered, asynchronous reset**



$$CC^1(Q) = CC^1(D) + CC^1(CLK) + CC^0(CLK) + CC^0(R)$$
$$SC^1(Q) = SC^1(D) + SC^1(CLK) + SC^0(CLK) + SC^0(R) + 1$$

$$CC^0(Q) = \min[CC^1(R),\ CC^0(D) + CC^1(CLK) + CC^0(CLK) + CC^0(R)]$$
$$SC^0(Q) = \min[SC^1(R),\ SC^0(D) + SC^1(CLK) + SC^0(CLK) + SC^0(R)] + 1$$

Flip-Flop (Observability)



$CO(D) = CO(Q) + CC^1(CLK) + CC^0(CLK) + CC^0(R)$

$SO(D) = SO(Q) + SC^1(CLK) + SC^0(CLK) + SC^0(R) + 1$

Seq. SCOAP Computation Alg.

- Computation of SC, SO is similar to CC, CO
  - but require *iterations* for controllability to converge

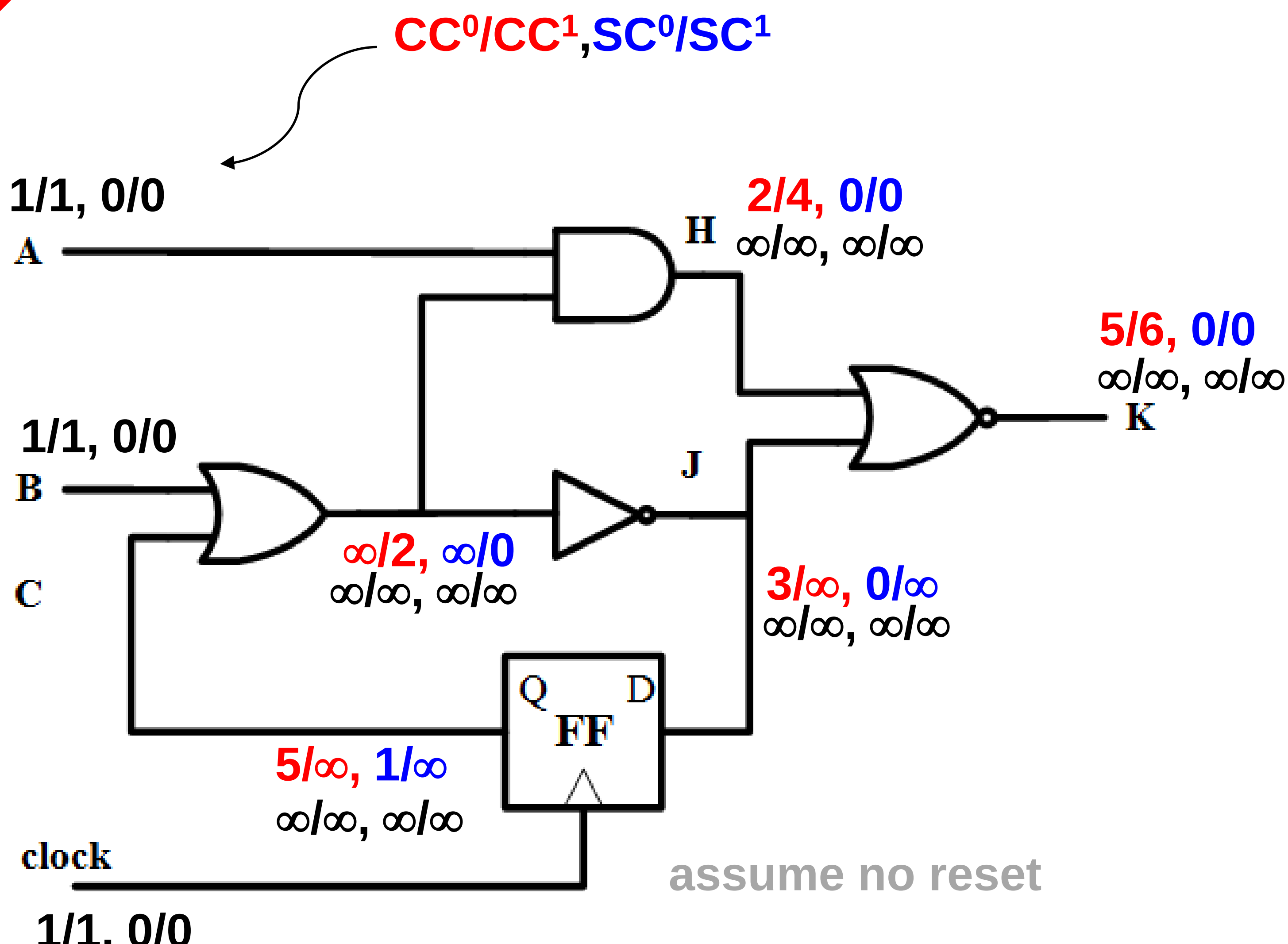
Controllability:

1. For all PI's, set $CC^0 = CC^1 = 1$ and $SC^0 = SC^1 = 0$
2. For all other nodes, set $CC^0 = CC^1 = \infty$ and $SC^0 = SC^1 = \infty$
3. Propagate controllability from PI's to PO's → Iterate until numbers stabilize.

Observability:

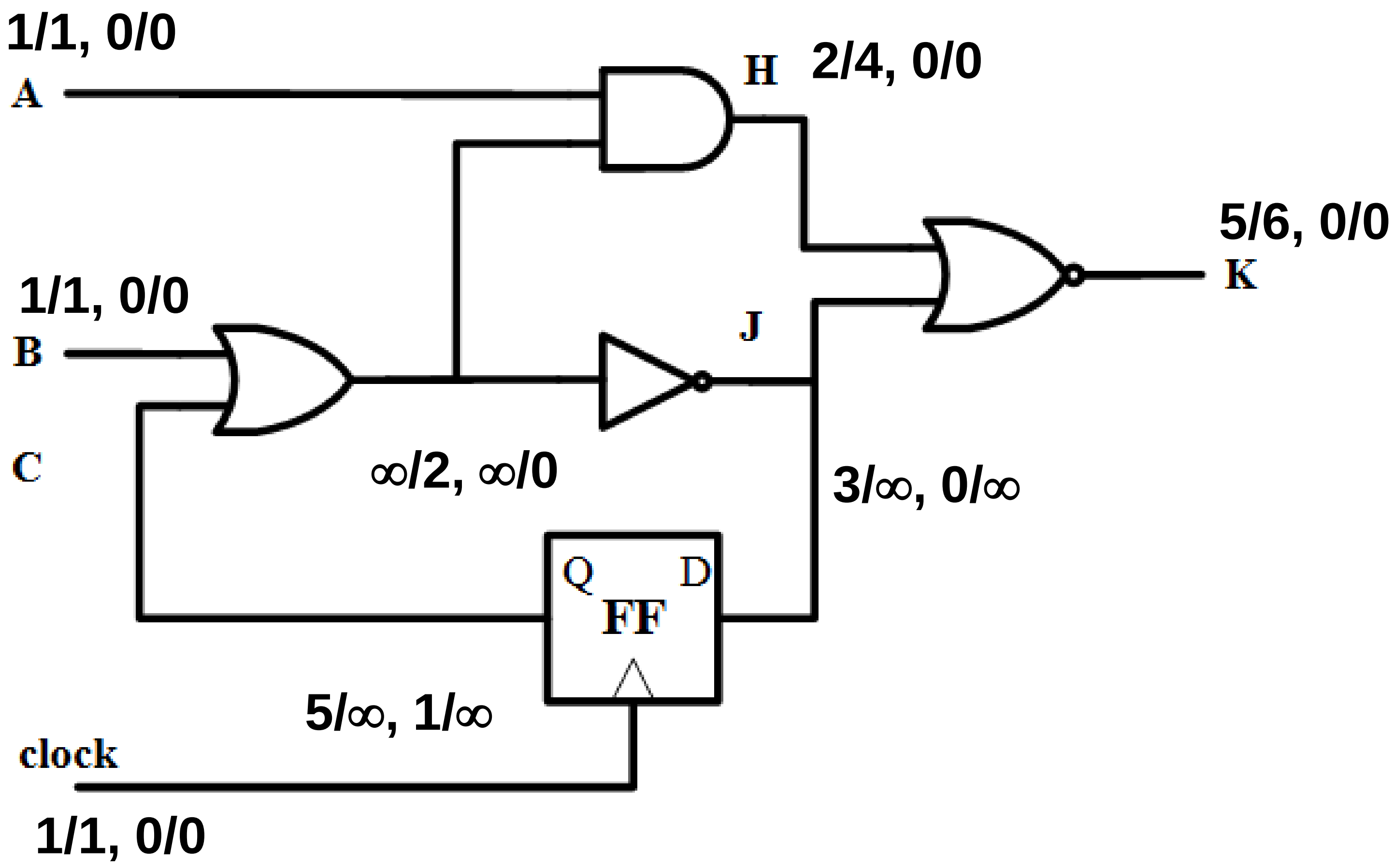
1. For all PO's, set CO = SO = 0
2. For all other nodes, set CO = SO = ∞
3. Propagate observability from PO's to PI's ←
   (note: no iteration needed for CO/SO)

Controllability Computation - 1

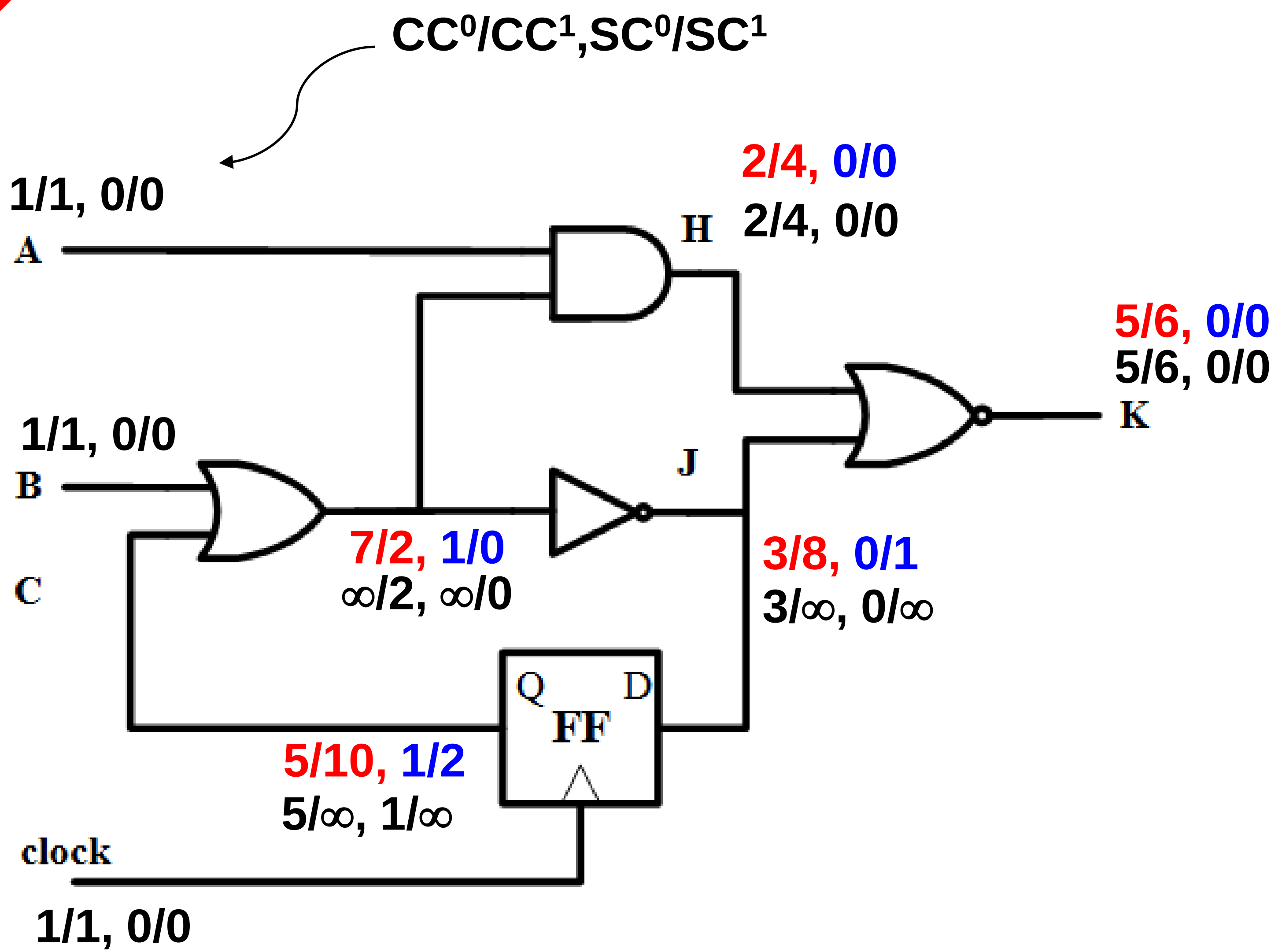


Quiz

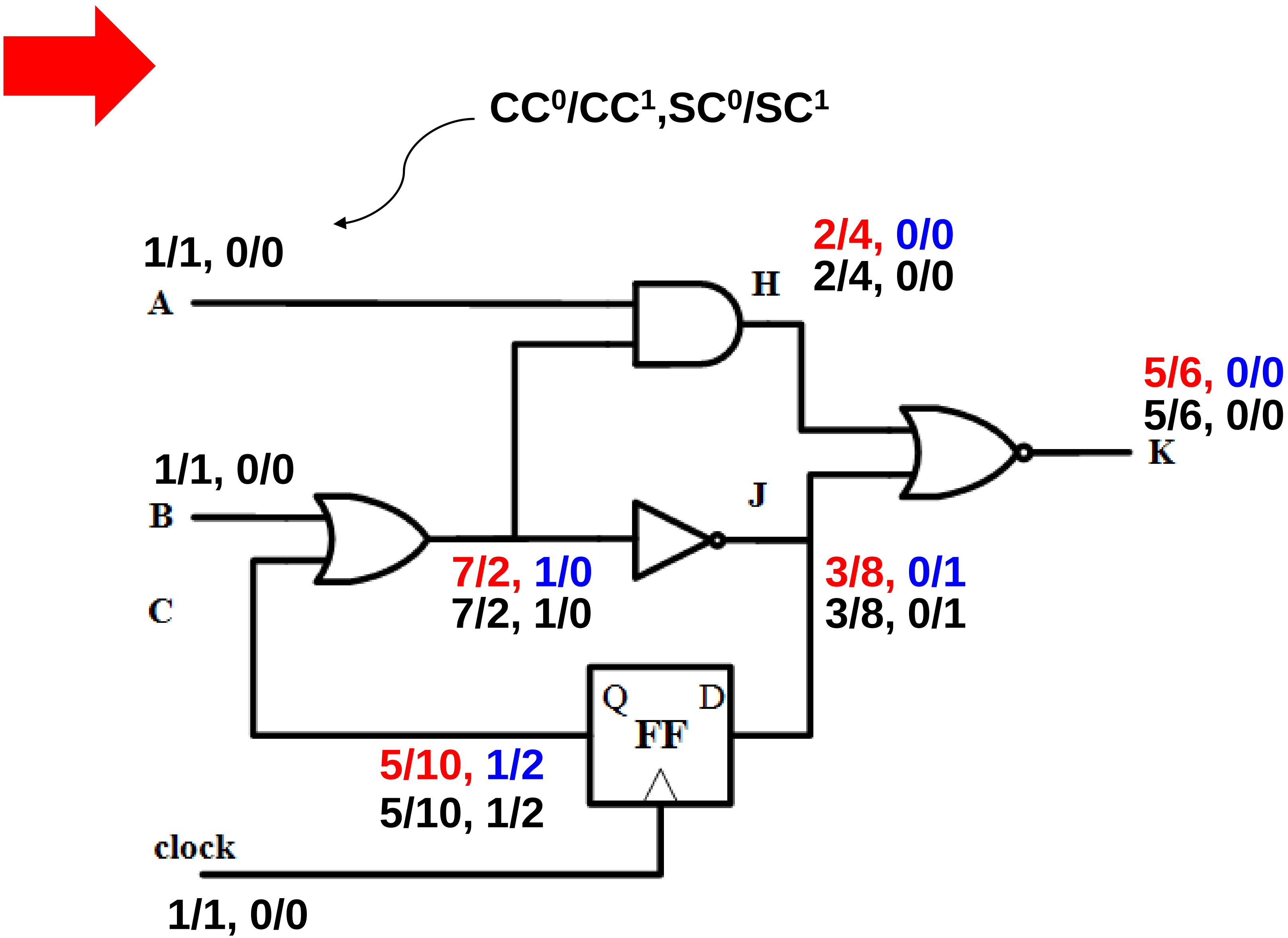
Q: Given numbers from the 1st iteration,
please continue to calculate CC^0/ CC^1, SC^0/SC^1 in 2nd iteration.



Controllability Computation - 2

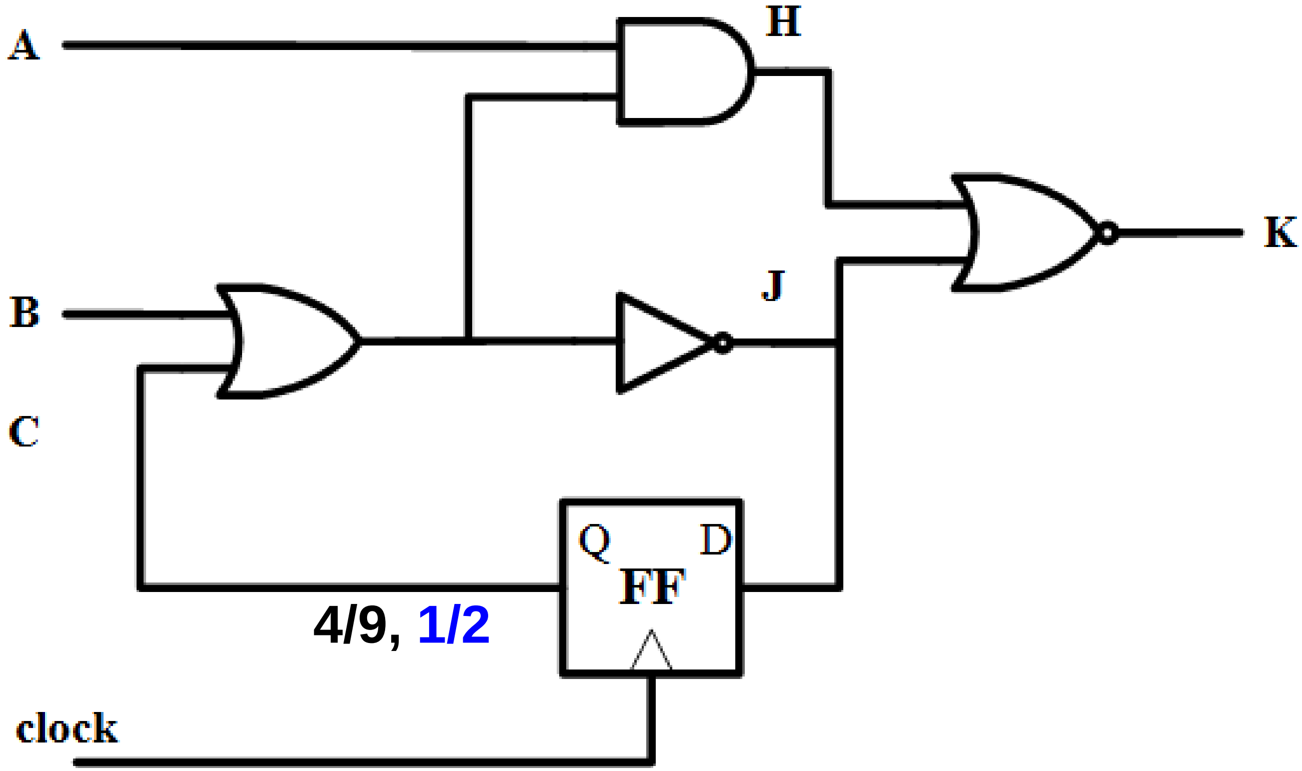


Controllability Computation - 3



Quiz

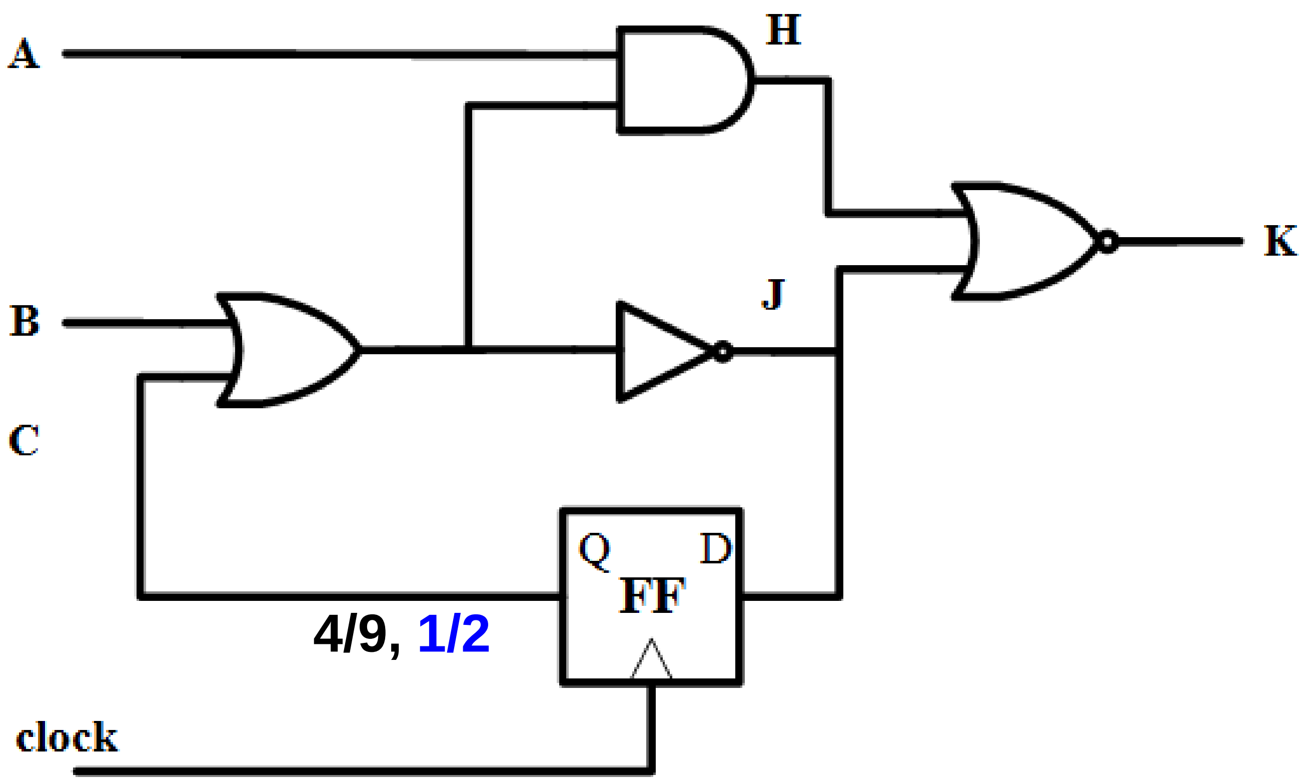
Q1: Generate a sequence of test patterns to control *C* to 0?
Q2: Generate a sequence of test patterns to control *C* to 1?
(assume no scan. can only assign PI)



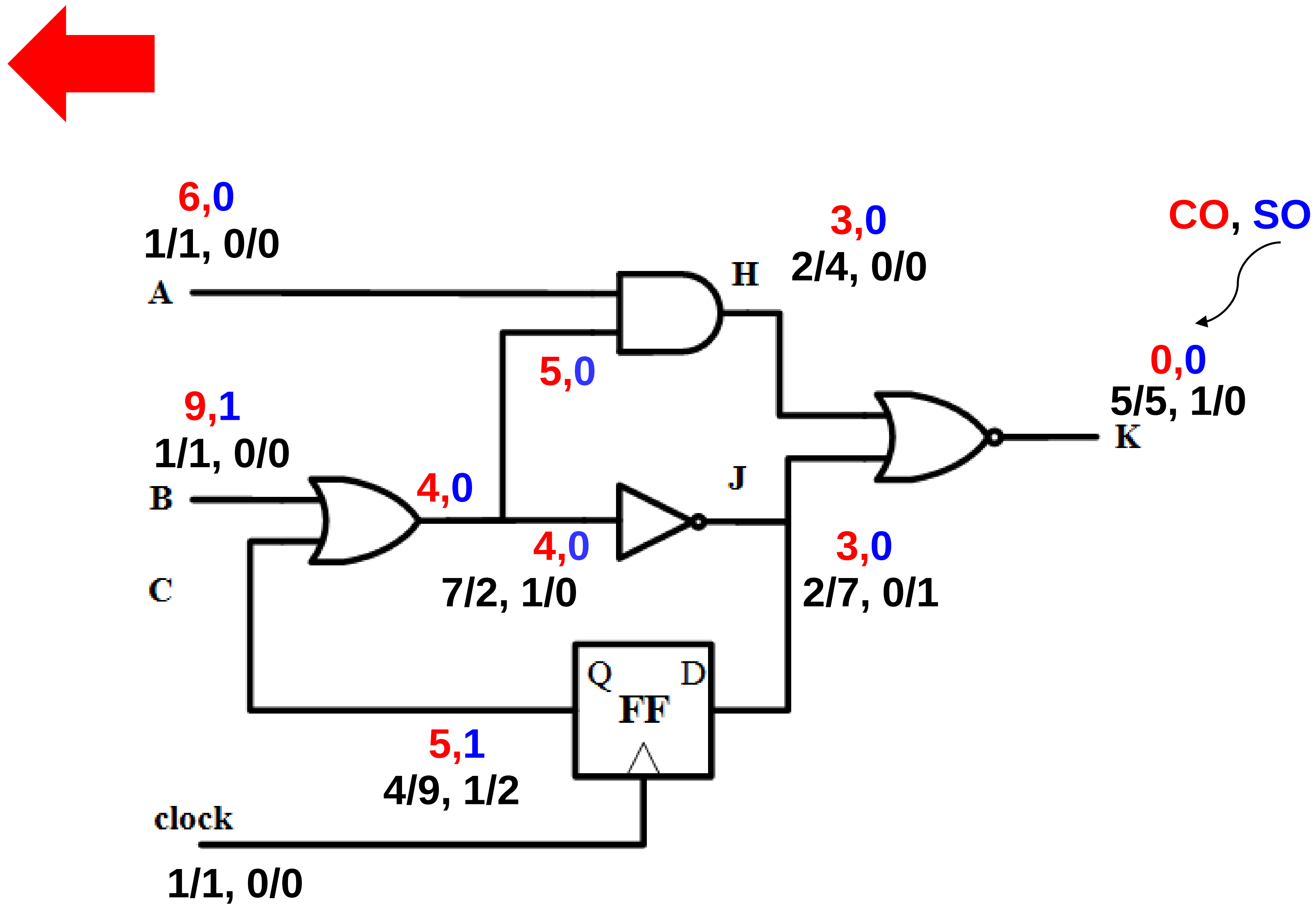
4/9, 1/2

What Does SC^1=2 Mean?

- **Control *C* to zero is easier. Assign *B*=1 and pulse one clock**
- **Control *C* to one is more difficult. Assign *B*=1 and *B*=0. Two clocks**

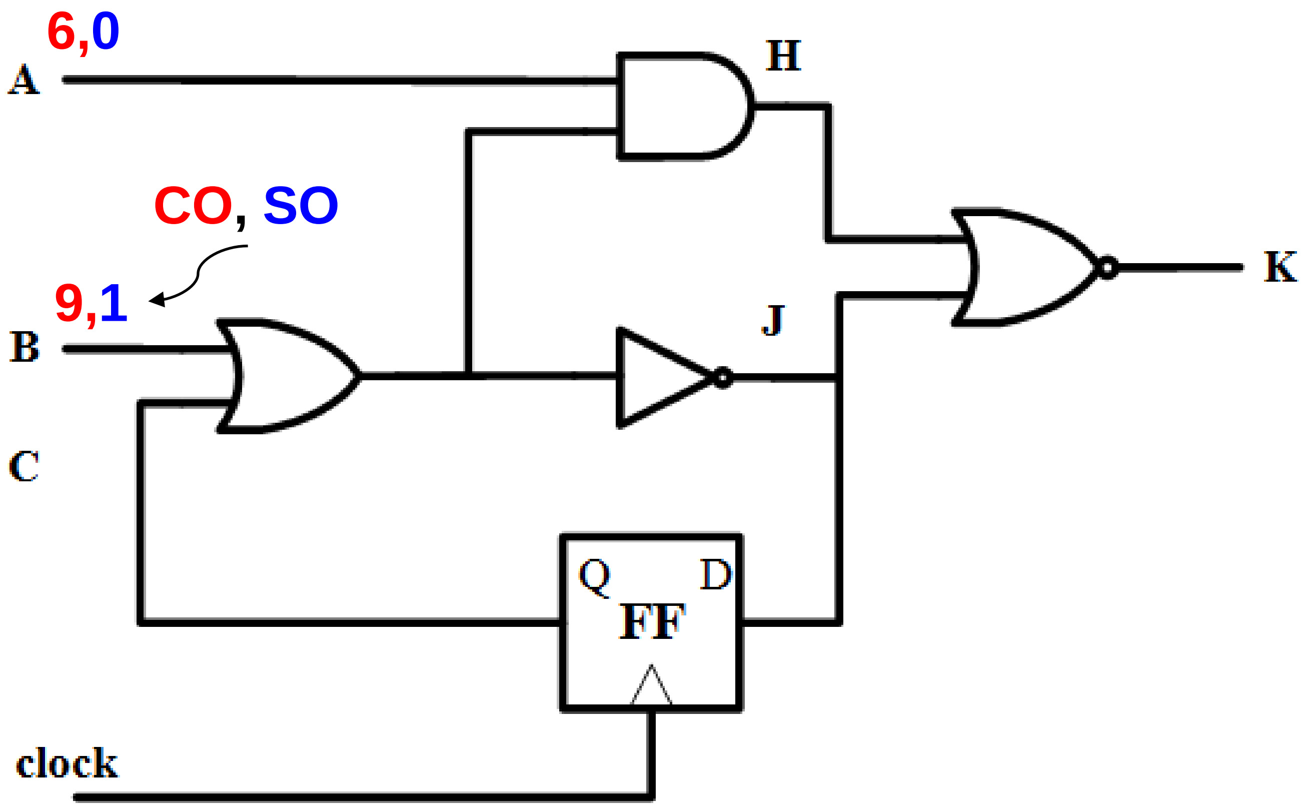


Obervability Computation



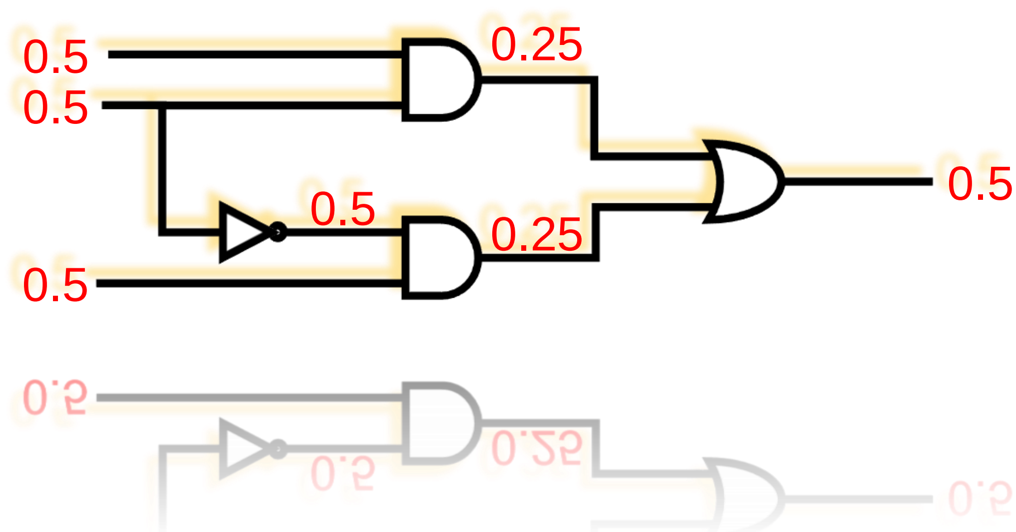
Quiz

Q1: Generate a sequence of test patterns to observe *A*?
Q2: Generate a sequence of test patterns to observe *B*?
(assume no scan. can only assign PI)



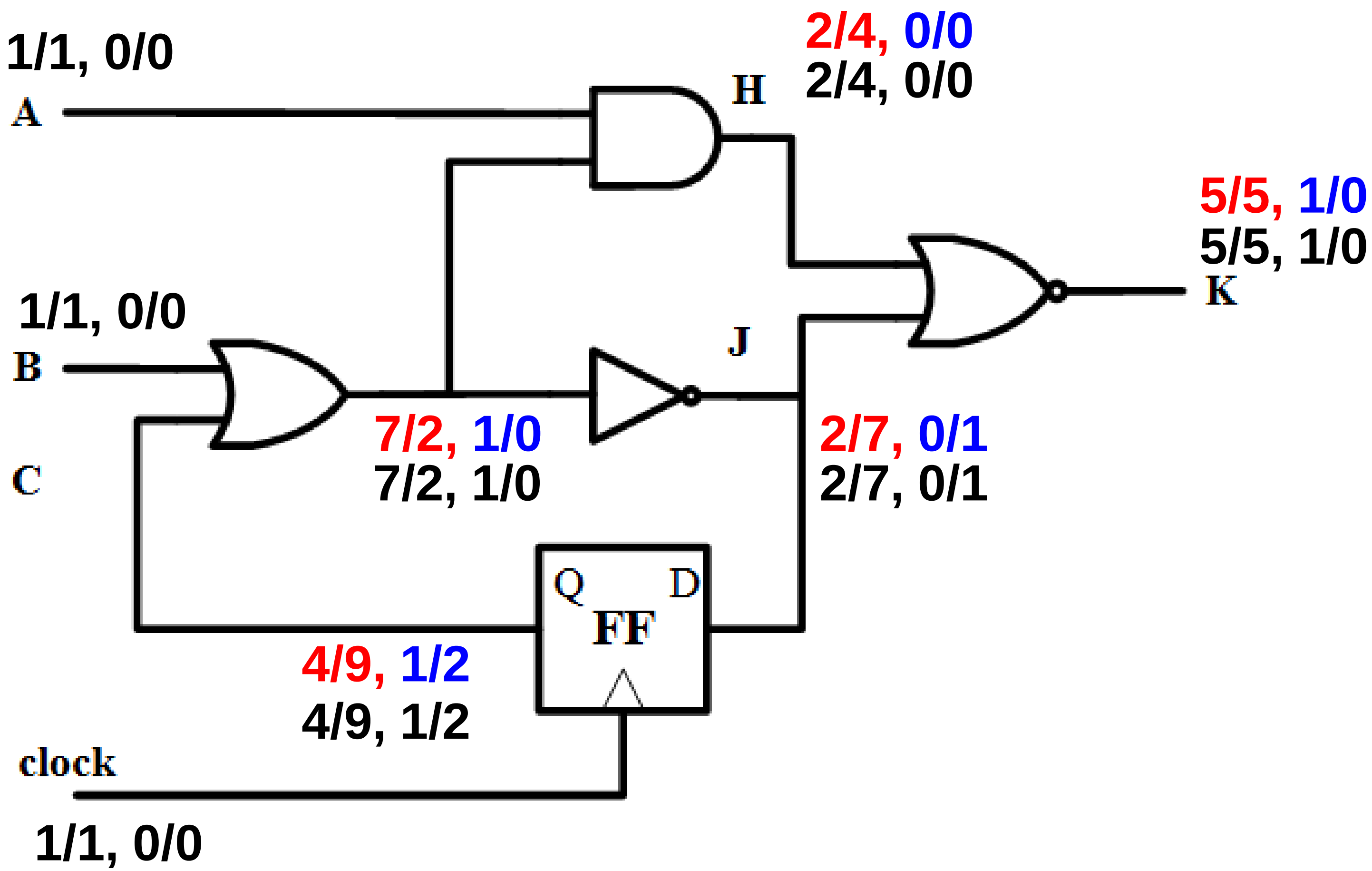
Summary

- Sequential SCOAP
  - SC^0 SC^1 = 0 and 1 controllability; SO = observability
  - Measure FF assignments (or clock cycles) needed
  - Smaller means easier
  - No scan is allowed
- Requires iterations to compute SC



FFT

- **When does the algorithm fail to converge?**



Computing Sequential SCOAP

- Computation of $SC^0(N)$, $SC^1(N)$, and $SO(N)$ is similar to
  - $CC^0(N)$, $CC^1(N)$, and $CO(N)$.

- Differences are
  1. Increments sequential SCOAP by 1 only when signals propagate from FF inputs to Q, or backwards
  2. May require ***iterations*** for controllability to converge