ECE 2300
Digital Logic & Computer Organization

Spring 2025

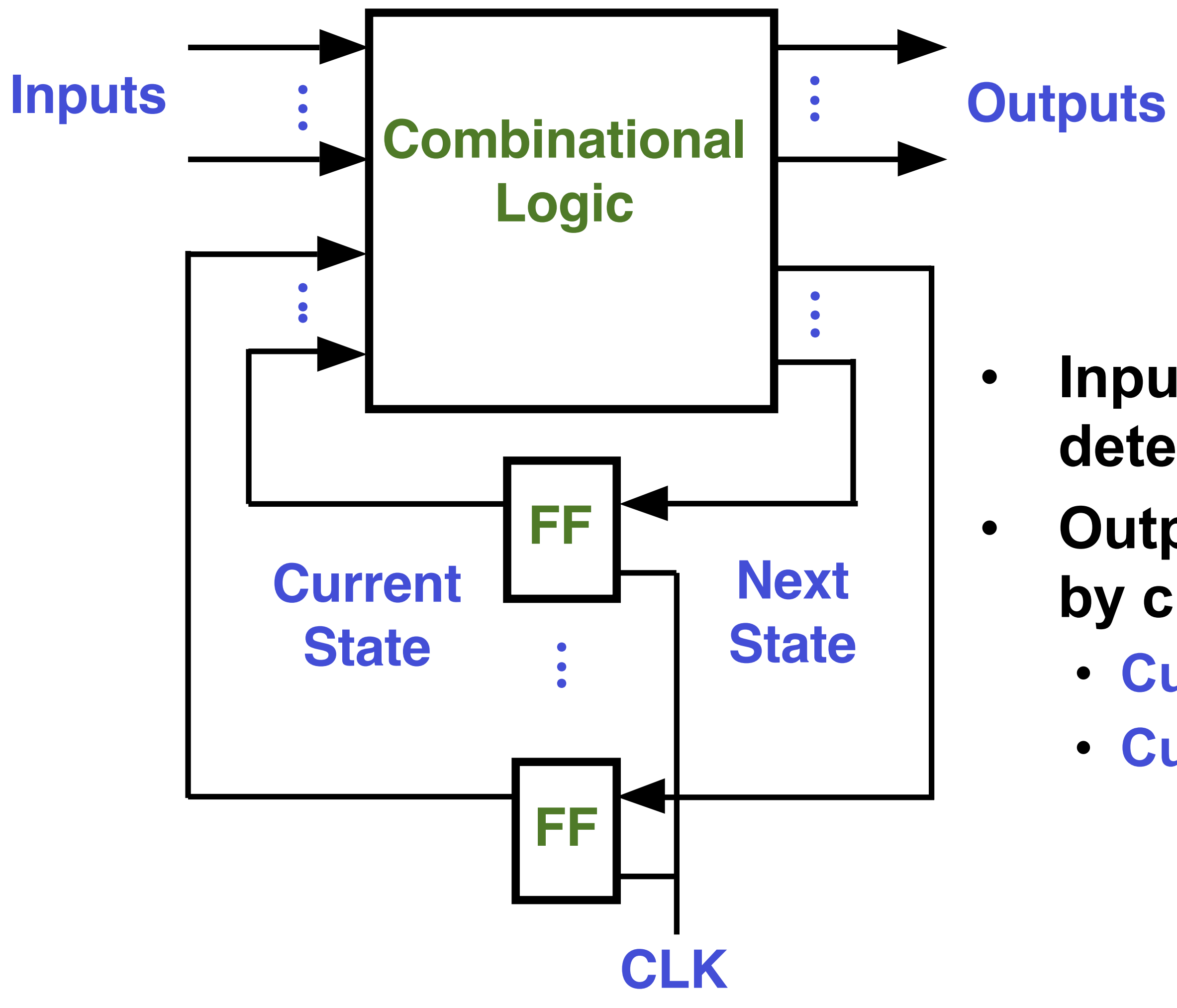
More FSMs
Timing

Cornell University

Announcements

- HW4 will be released today
- Lab 2a due tomorrow

Review: FSM General Circuit Form



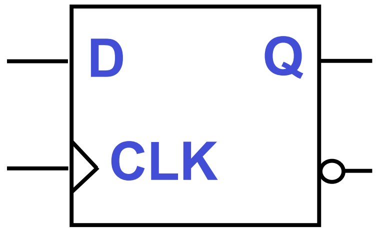
- Inputs and current state determine state transitions
- Output changes determined by changes in
  - Current state (Moore), or
  - Current state + inputs (Mealy)

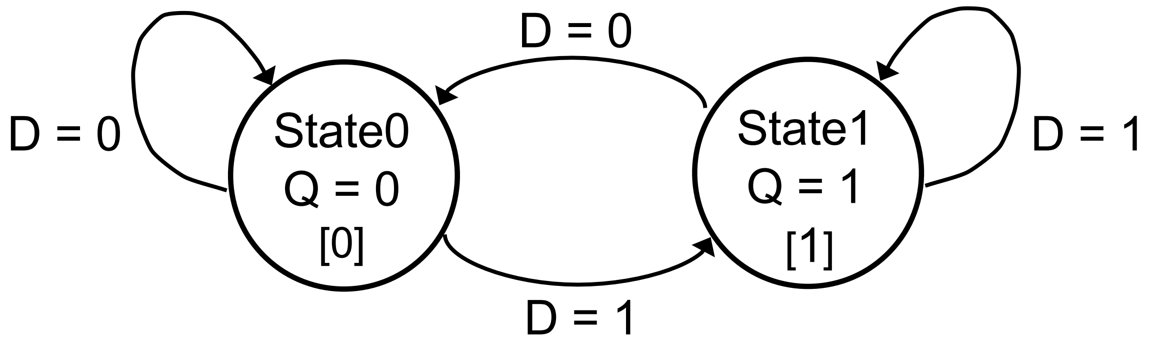
FSMs: True or False?

- The next state in Moore FSM only depends on its current state
- The output logic in Mealy FSM is sequential
- D flip-flop is a Moore machine

Recap: Moore State Diagram for DFF

- Input: D
- Output: Q





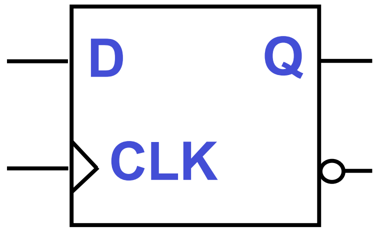
S	*S**		*Q*
	D = 0	D = 1	
0	0	1	0
1	0	1	1

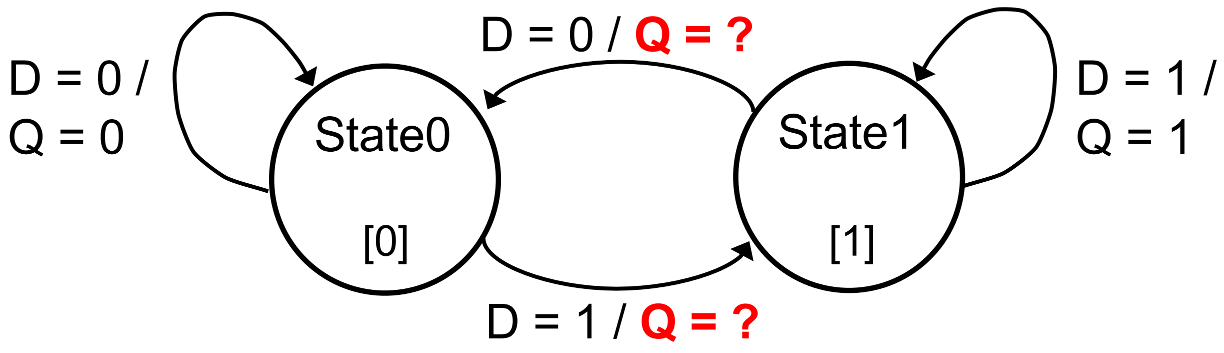
S* = D

Q = S

Mealy State Diagram for DFF?

- Input: D
- Output: Q

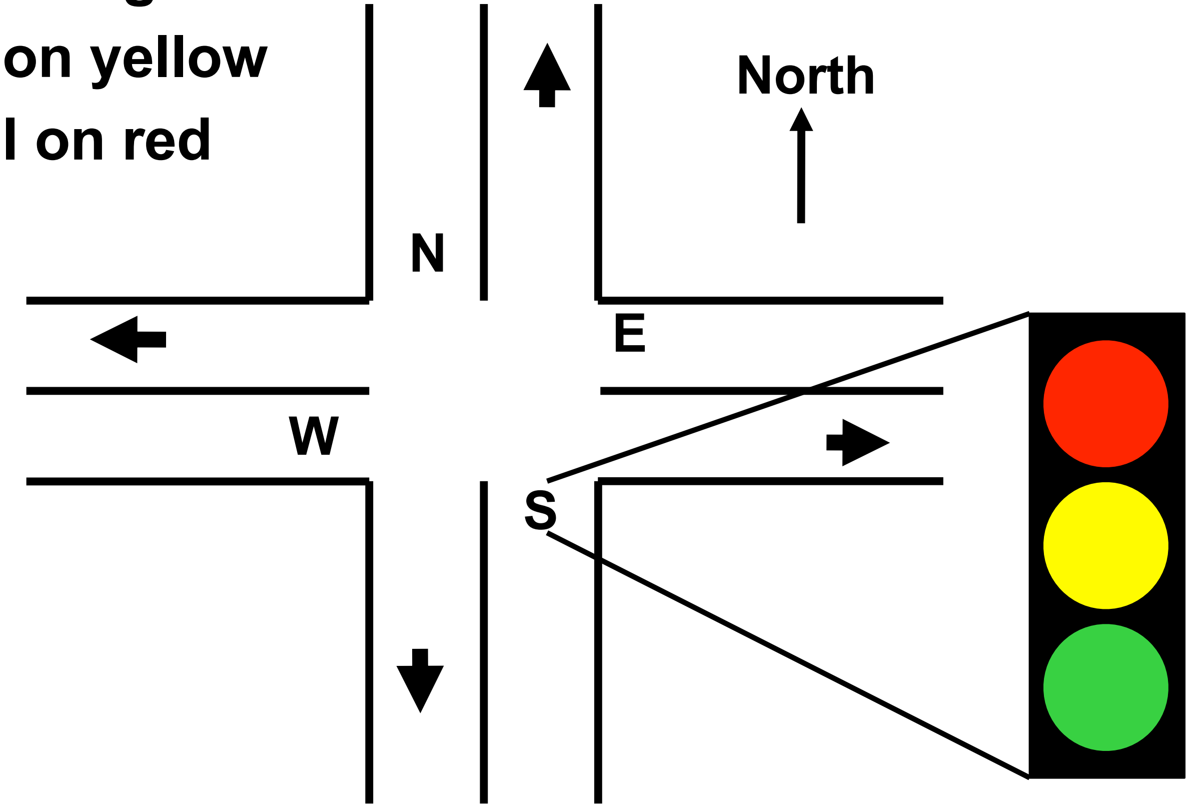




S	S*, Q	
	D = 0	D = 1
0	0, 0	1, ?
1	0, ?	1, 1

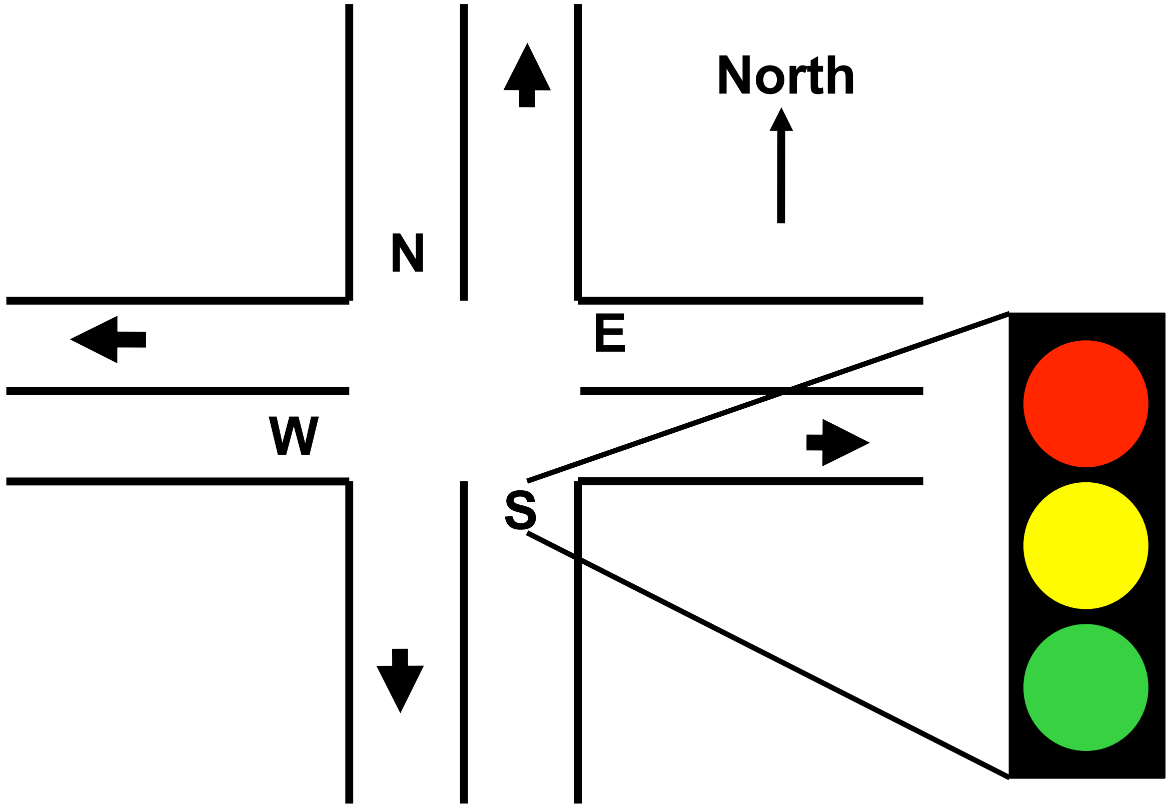
Traffic Light Controller

- 4-way intersection with traffic lights
  - East-West (E & W), North-South (N & S)
- Opposing lanes sequence together
  - 20 seconds dwell on green
  - 5 seconds dwell on yellow
  - 25 seconds dwell on red



Four Scenarios

- E & W Green / N & S Red for 20 seconds
- E & W Yellow / N & S Red for 5 seconds
- E & W Red / N & S Green for 20 seconds
- E & W Red / N & S Yellow for 5 seconds

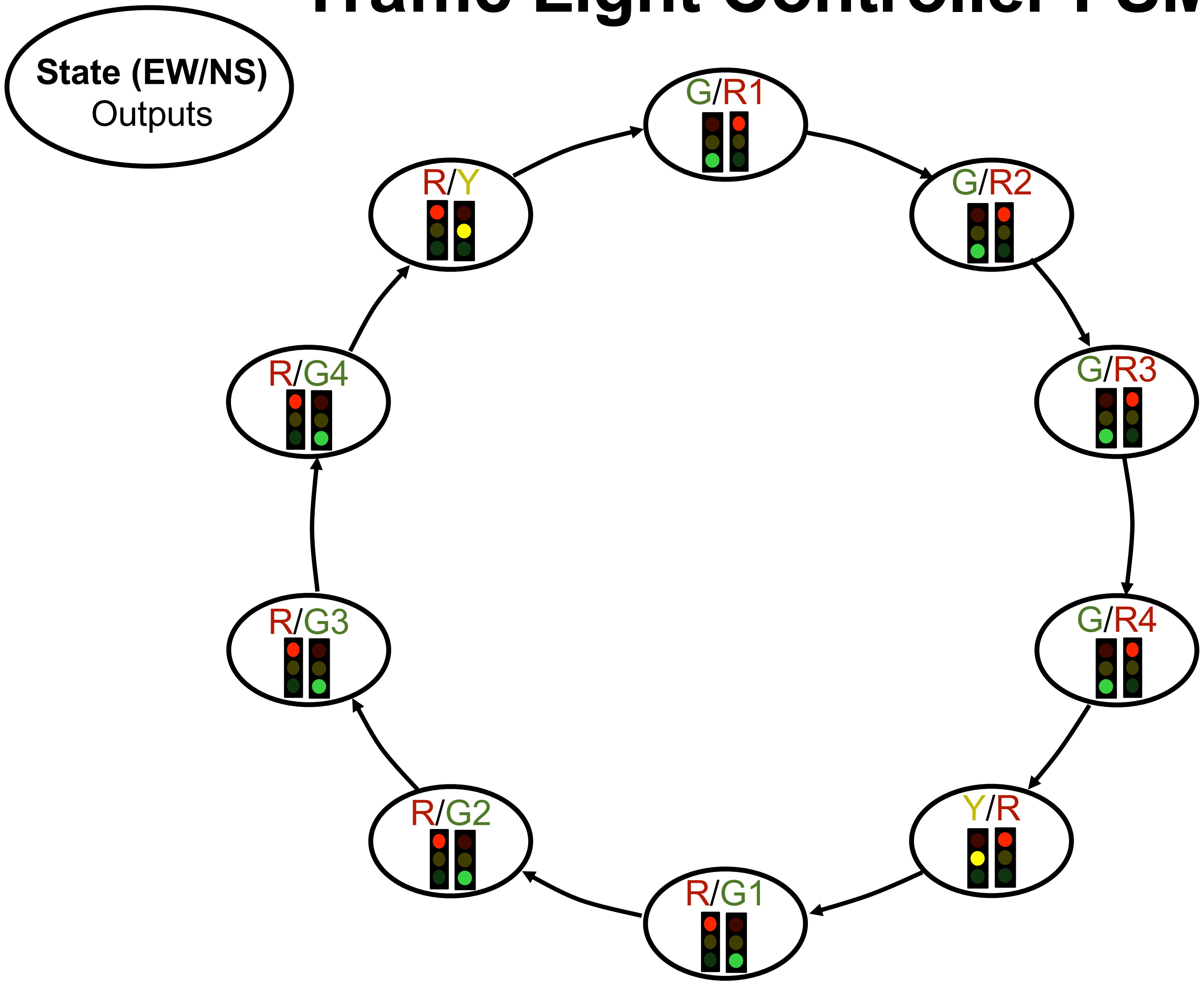


Traffic Light Controller States

- 10 states
  - E & W Green / N & S Red1 for 5 seconds
  - E & W Green / N & S Red2 for 5 seconds
  - E & W Green / N & S Red3 for 5 seconds
  - E & W Green / N & S Red4 for 5 seconds
  - E & W Yellow / N & S Red for 5 seconds
  - E & W Red / N & S Green1 for 5 seconds
  - E & W Red / N & S Green2 for 5 seconds
  - E & W Red / N & S Green3 for 5 seconds
  - E & W Red / N & S Green4 for 5 seconds
  - E & W Red / N & S Yellow for 5 seconds

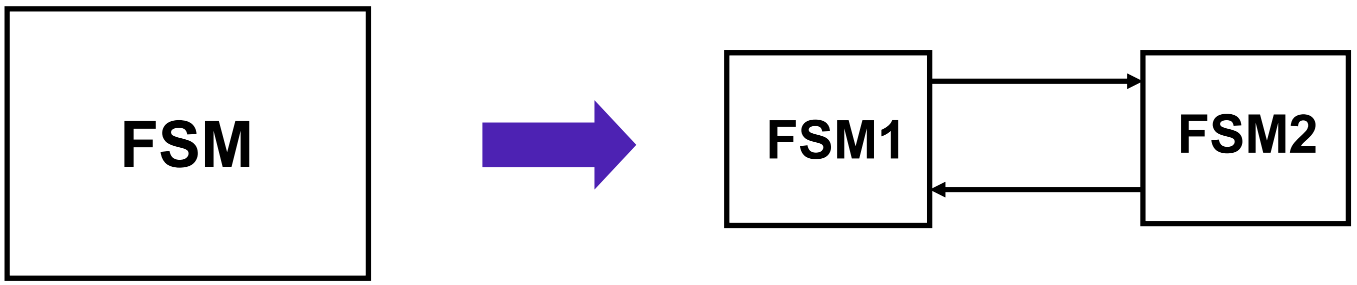
Clock period is 5 seconds

Traffic Light Controller FSM



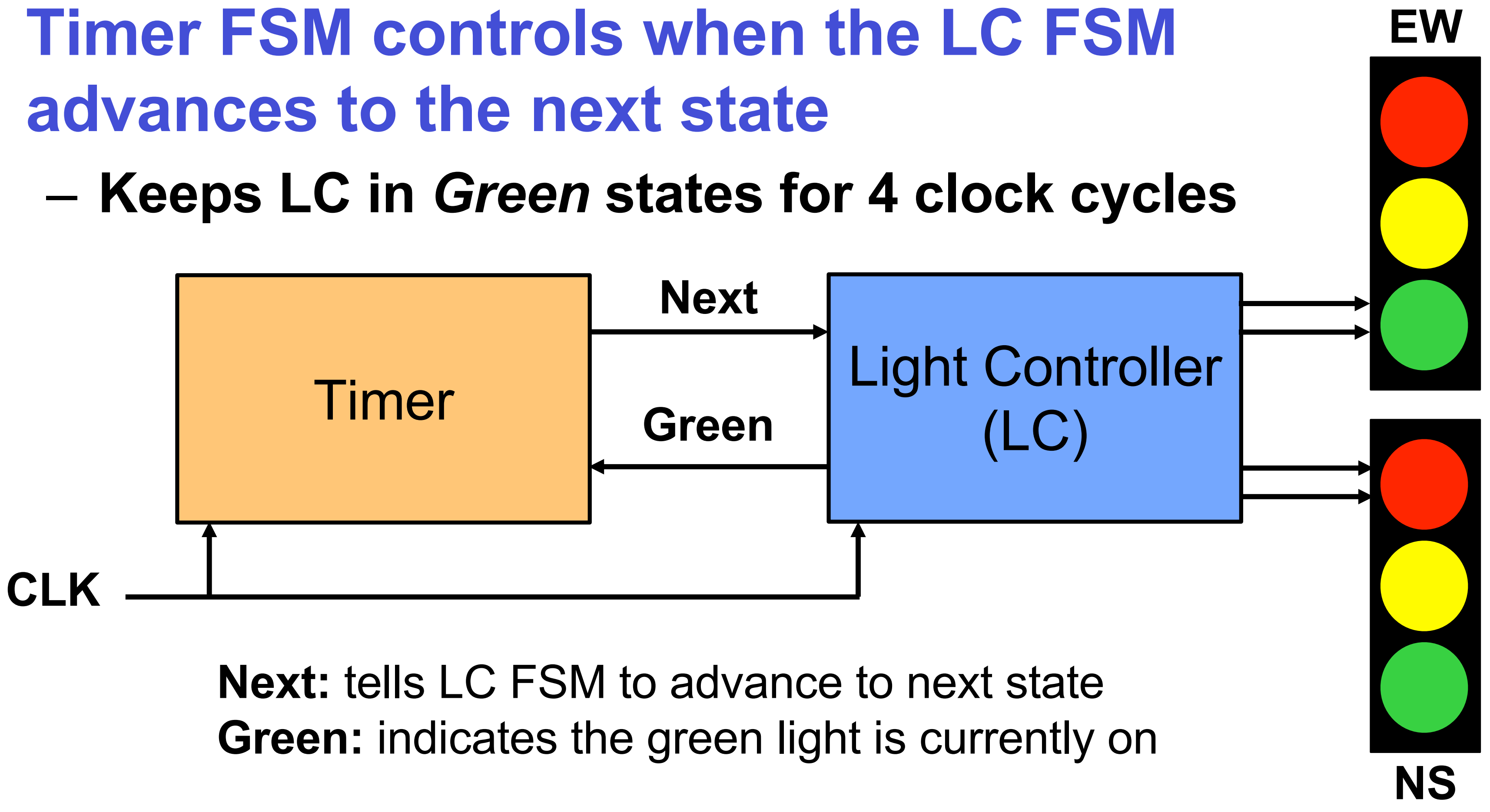
Factoring FSMs

- Break FSM into multiple communicating FSMs
- Simplifies large FSMs
- May result in fewer states



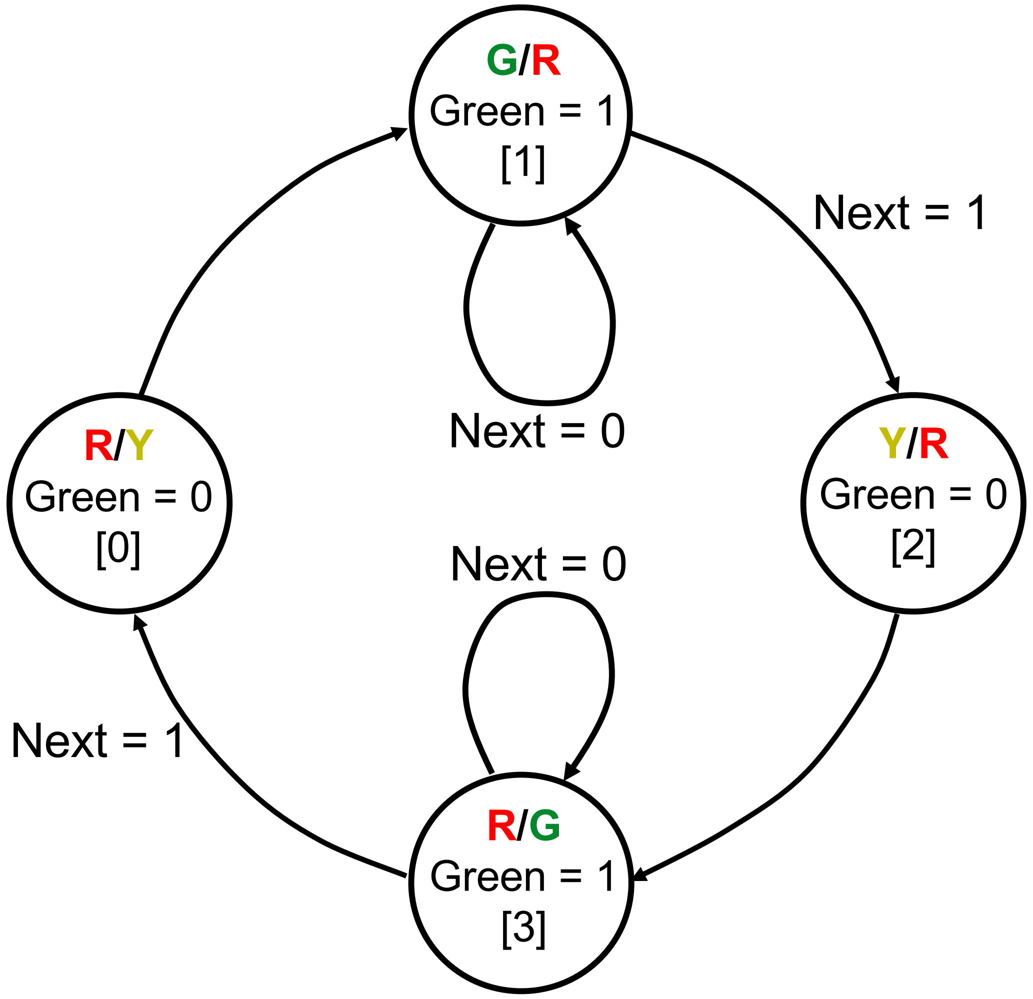
Traffic Light Controller Using 2 FSMs

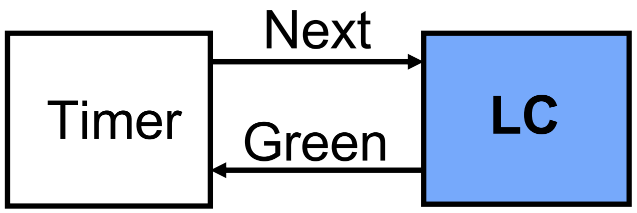
- Light Controller (LC) FSM has 4 states
  - G/R, Y/R, R/G, R/Y
- Timer FSM controls when the LC FSM advances to the next state
  - Keeps LC in *Green* states for 4 clock cycles



Next: tells LC FSM to advance to next state
Green: indicates the green light is currently on

Light Controller (LC) FSM



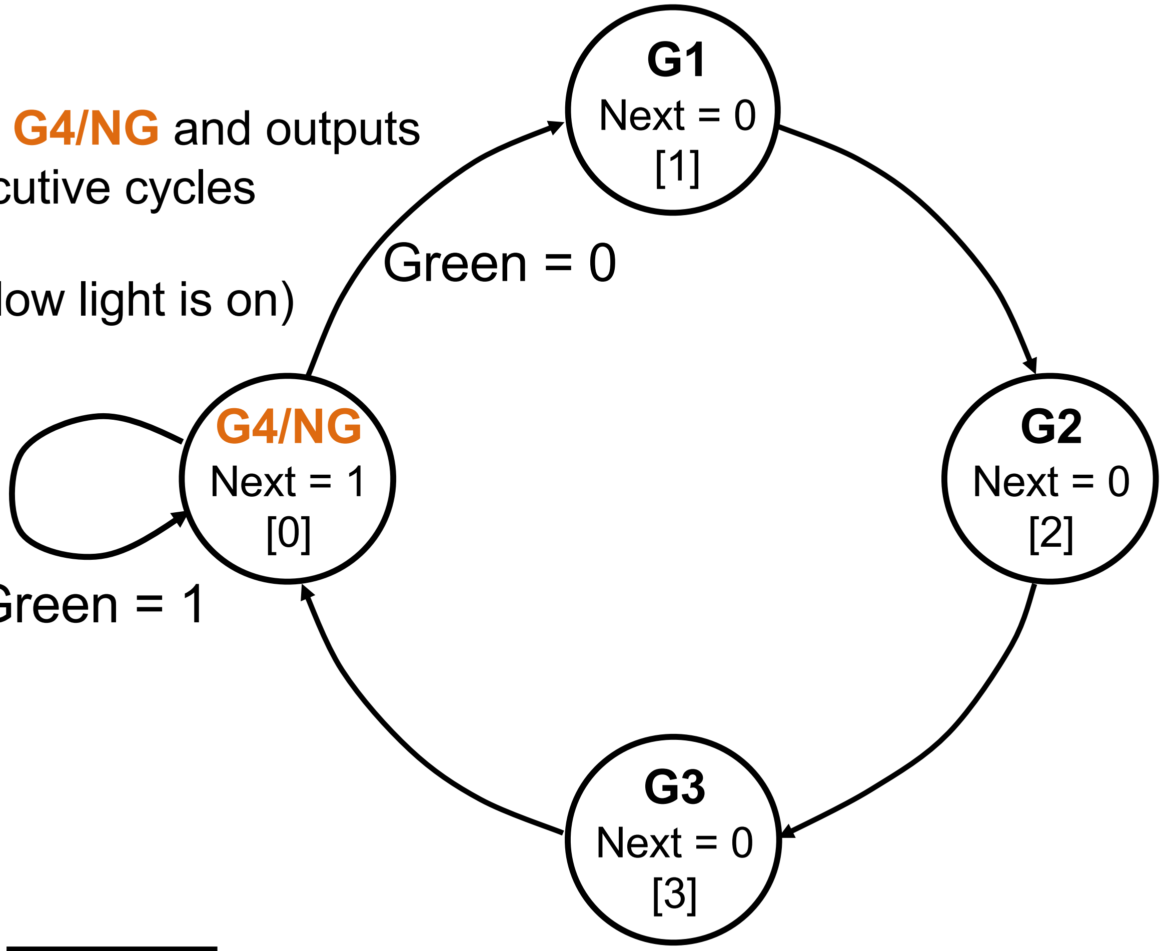


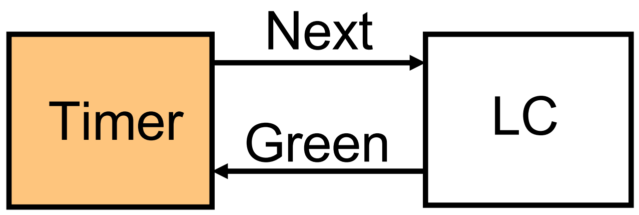
Timer FSM

Timer stays in state **G4/NG** and outputs Next=1 for 2 consecutive cycles

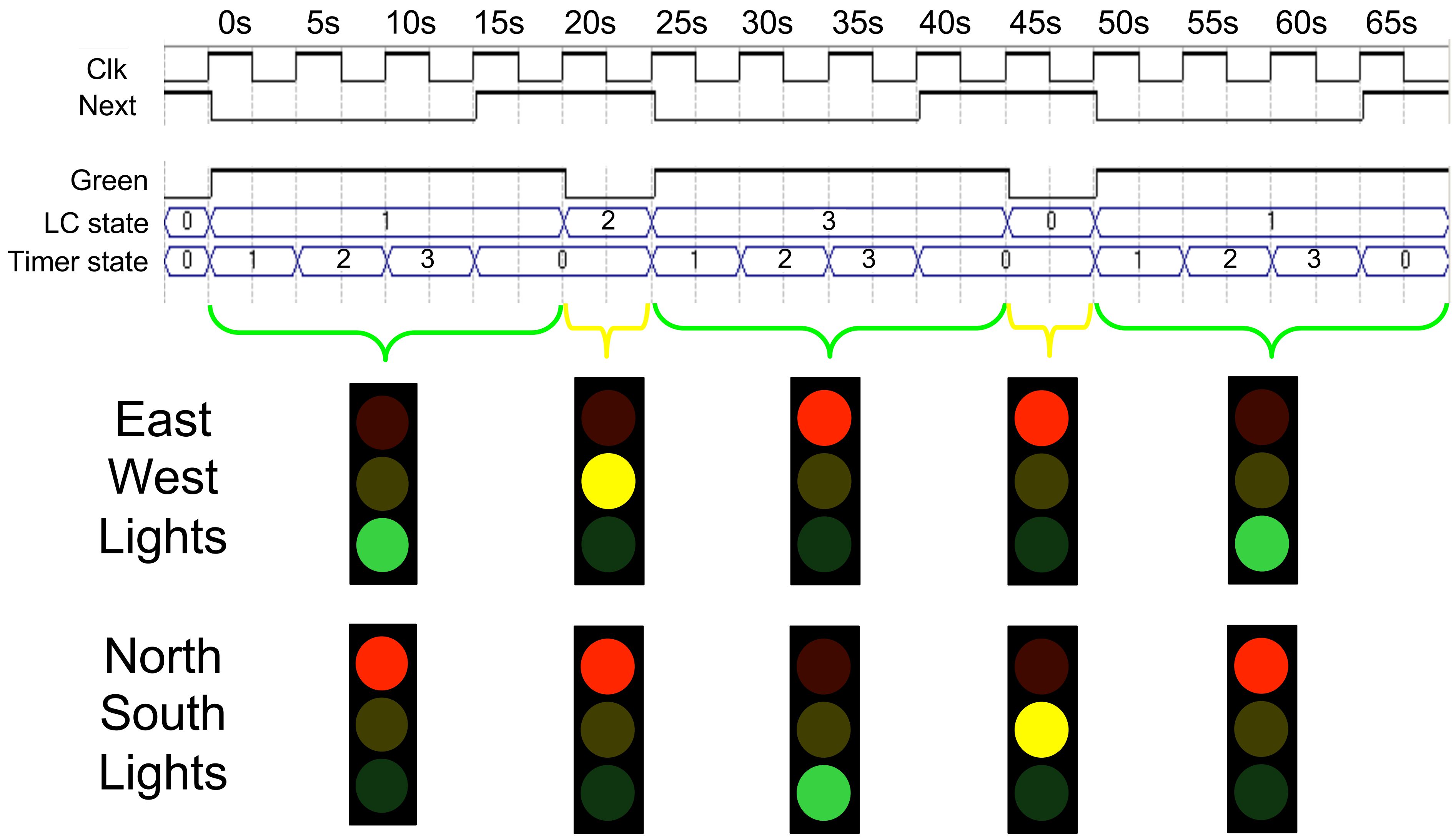
G4: Last Green

NG: Not Green (yellow light is on)

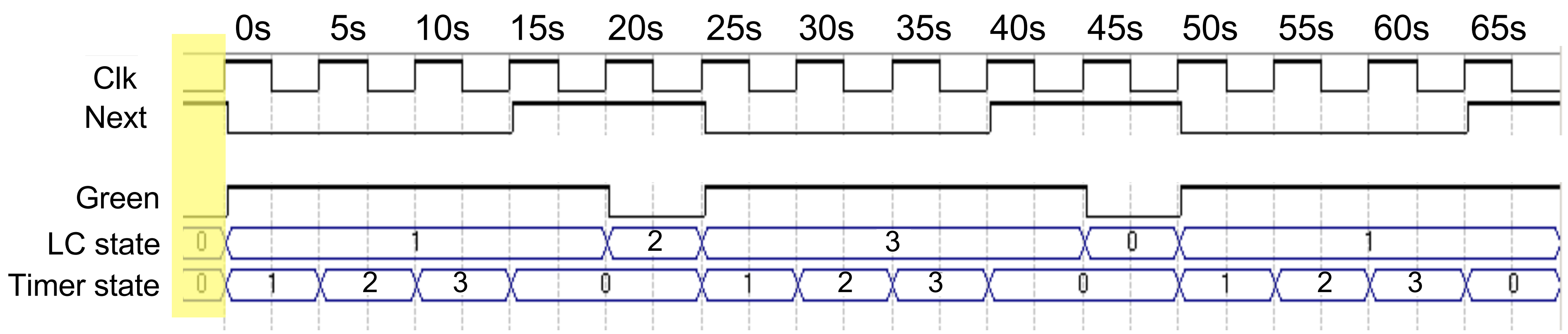


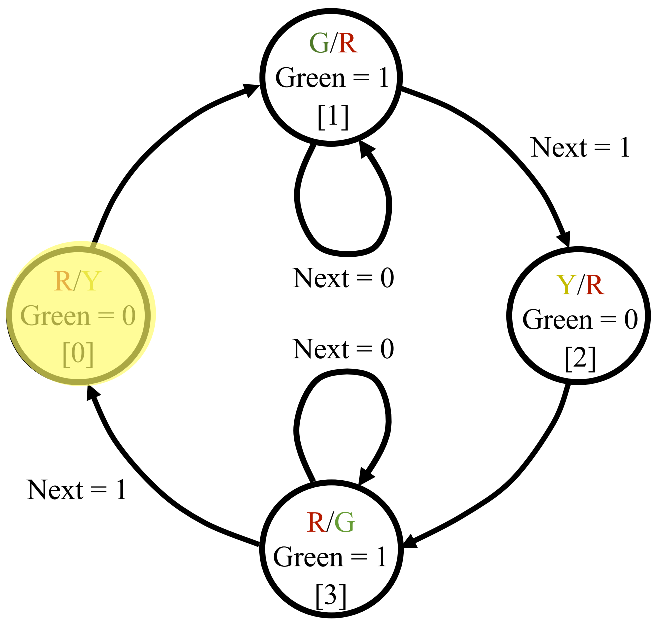


Traffic Light Controller Operation

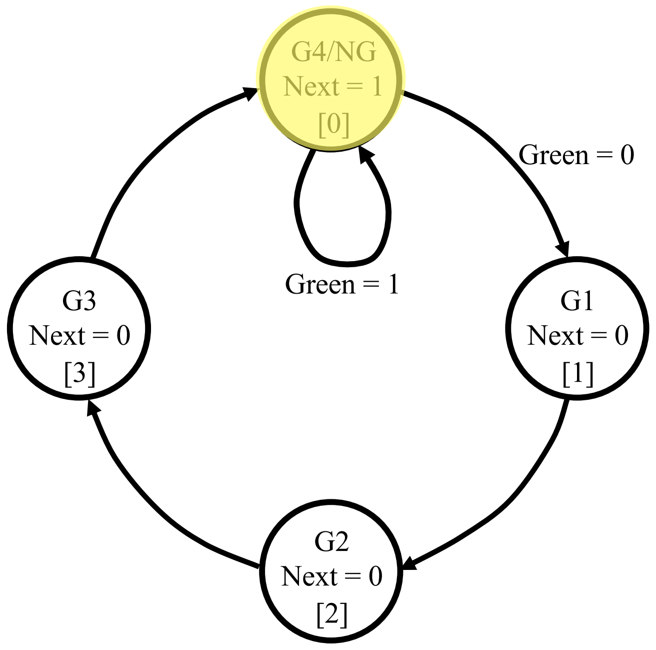


Traffic Light Controller Operation



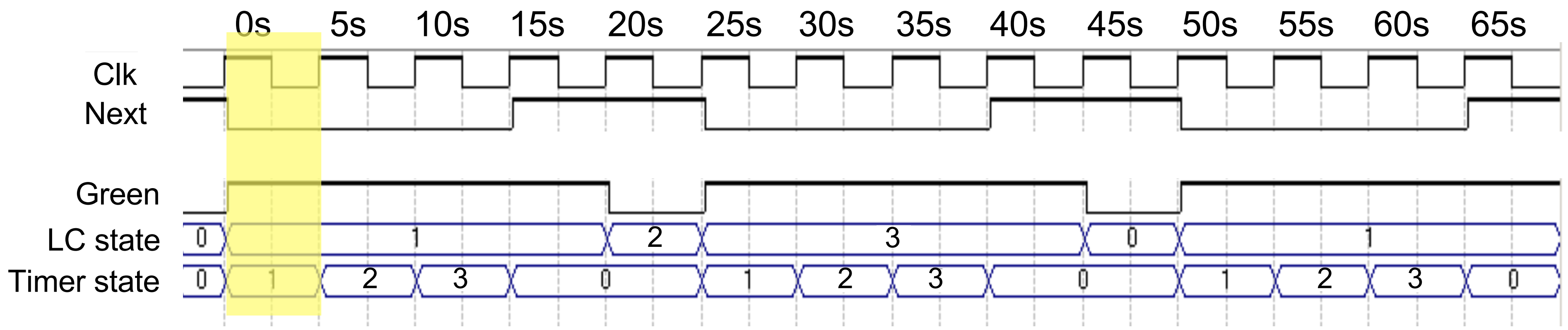


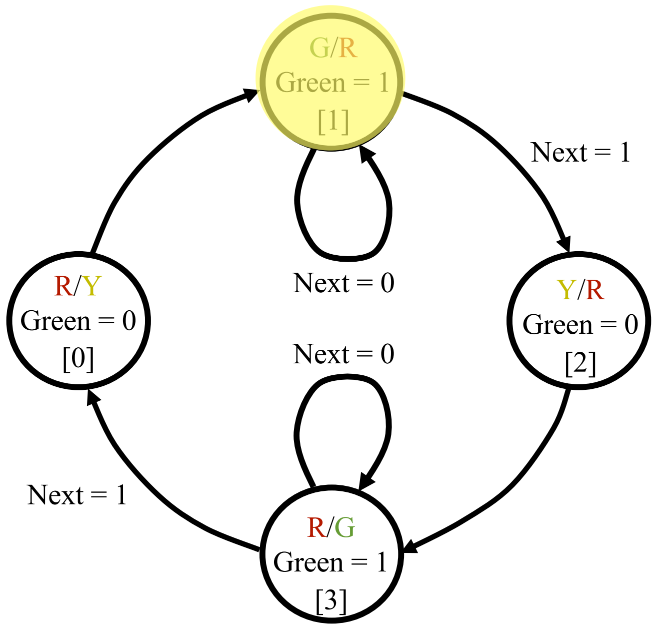
Light Controller FSM



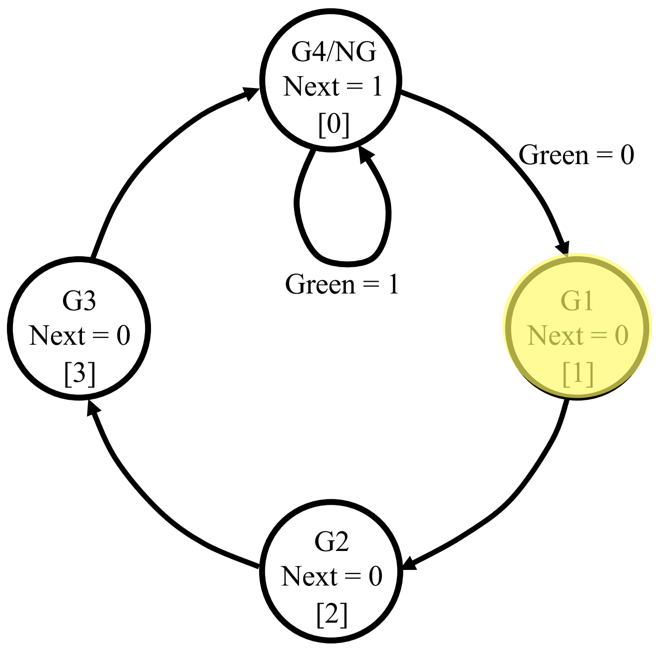
Timer FSM

Traffic Light Controller Operation



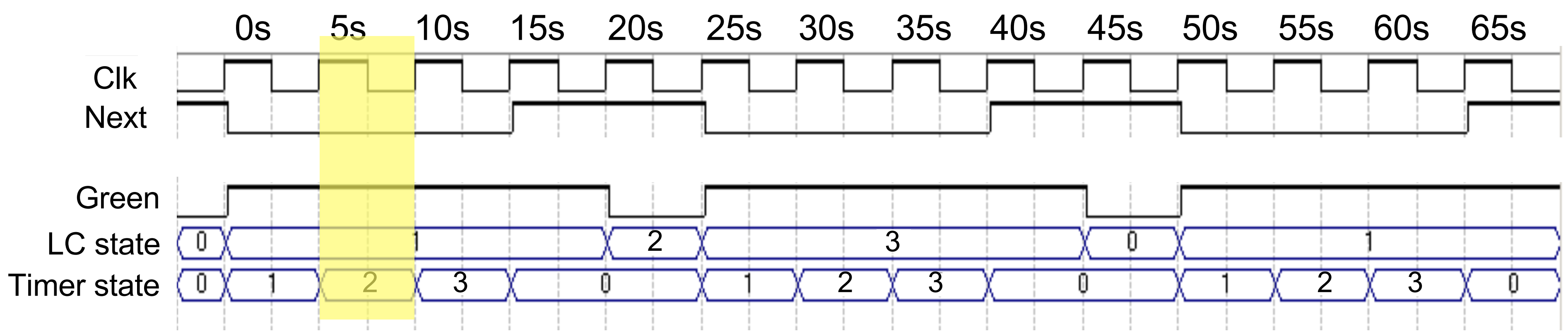


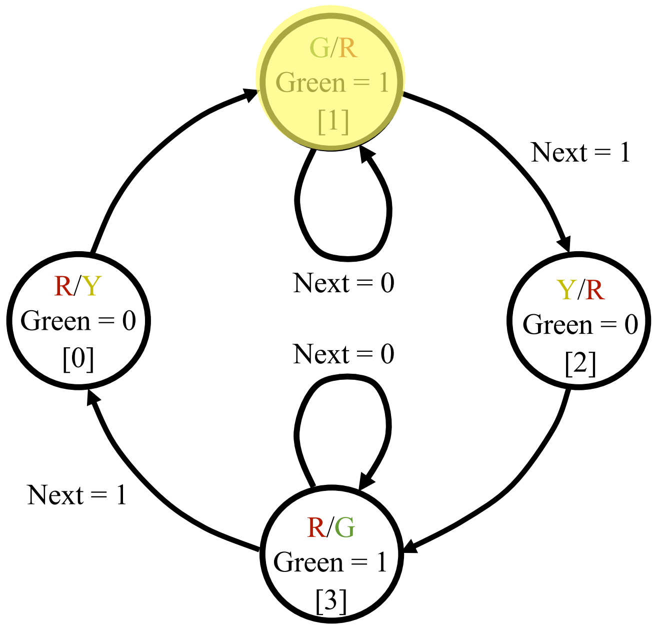
Light Controller FSM



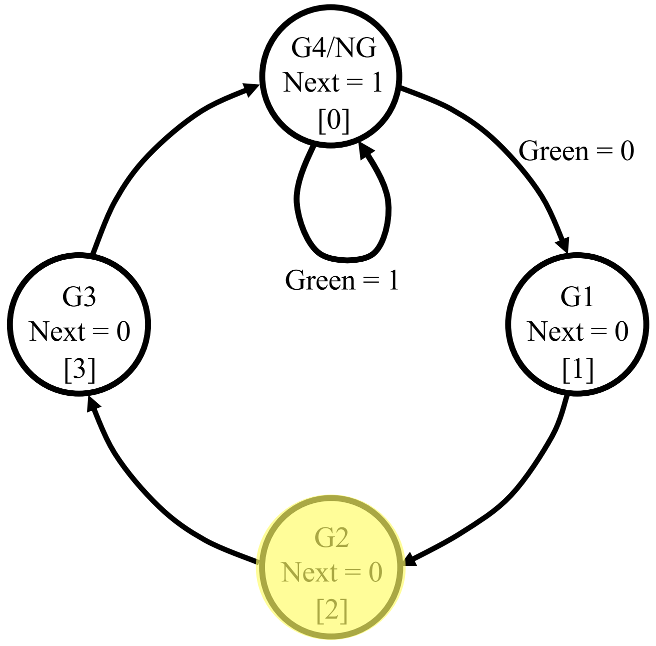
Timer FSM

Traffic Light Controller Operation



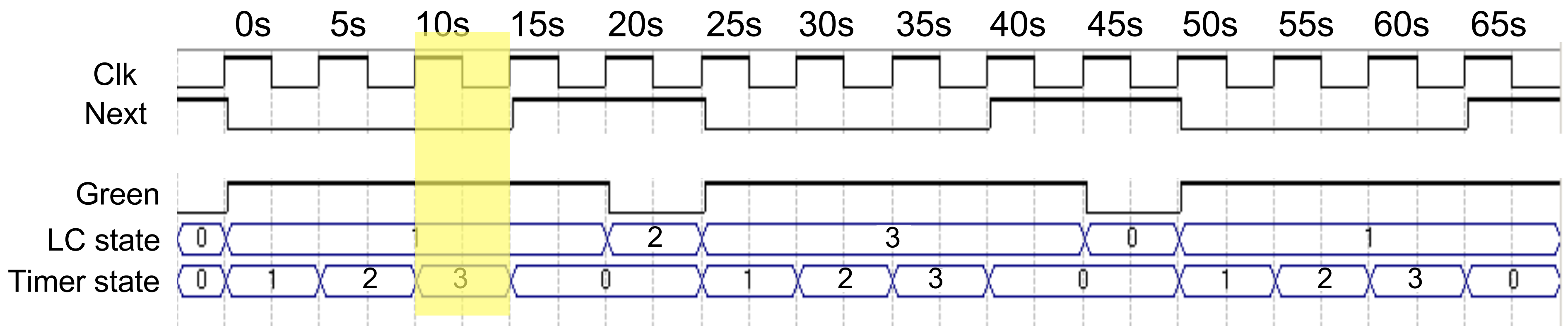


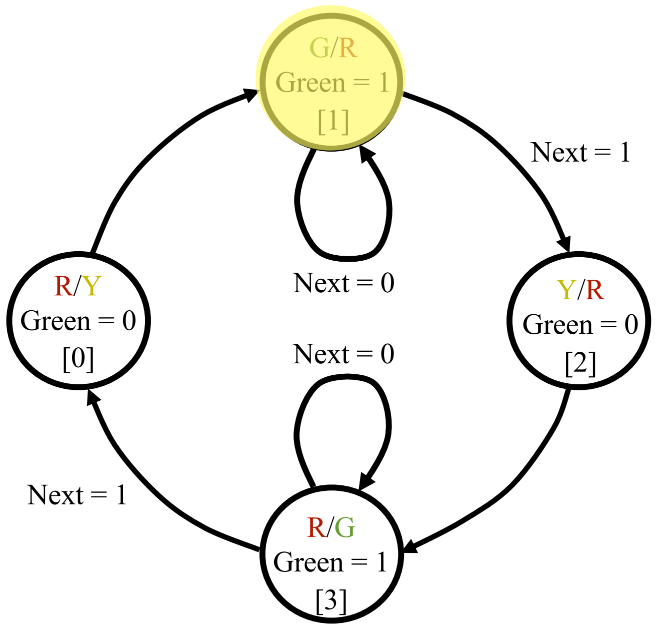
Light Controller FSM



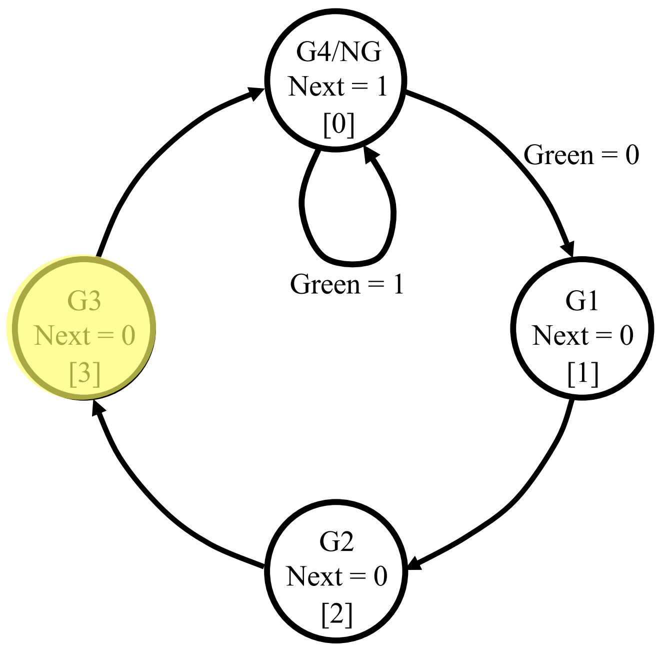
Timer FSM

Traffic Light Controller Operation



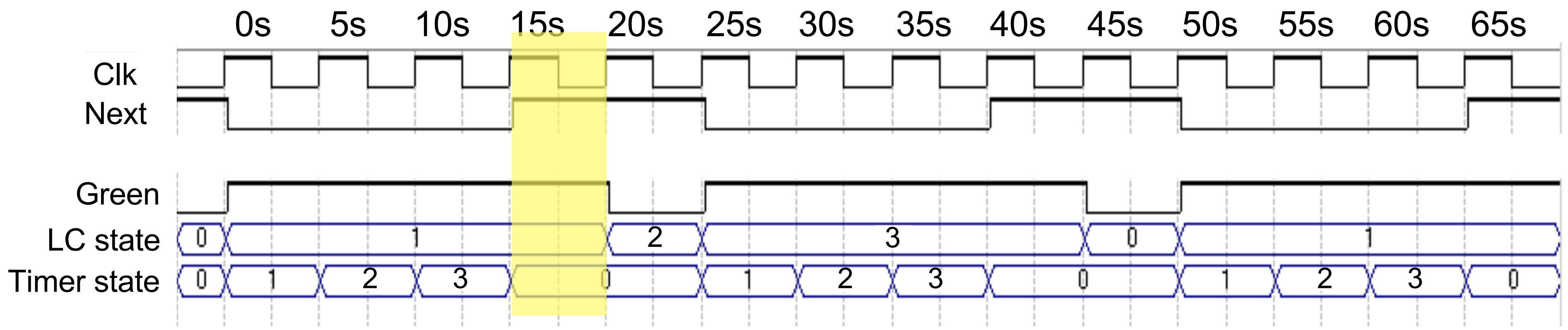


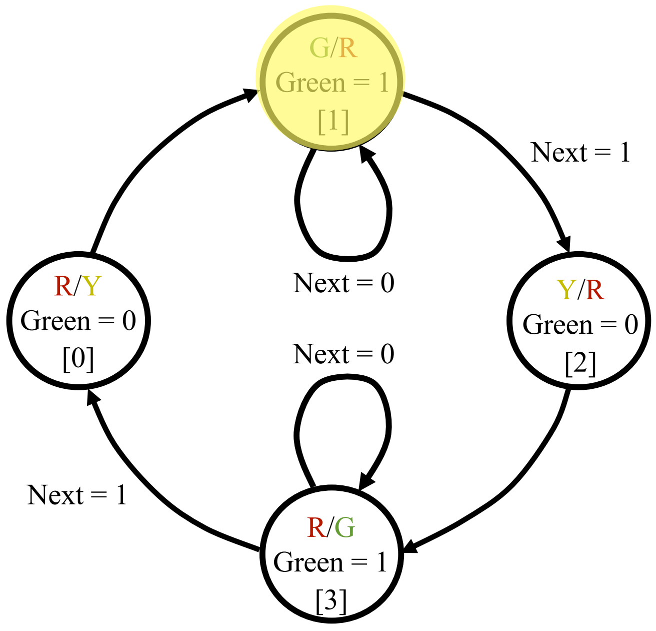
Light Controller FSM



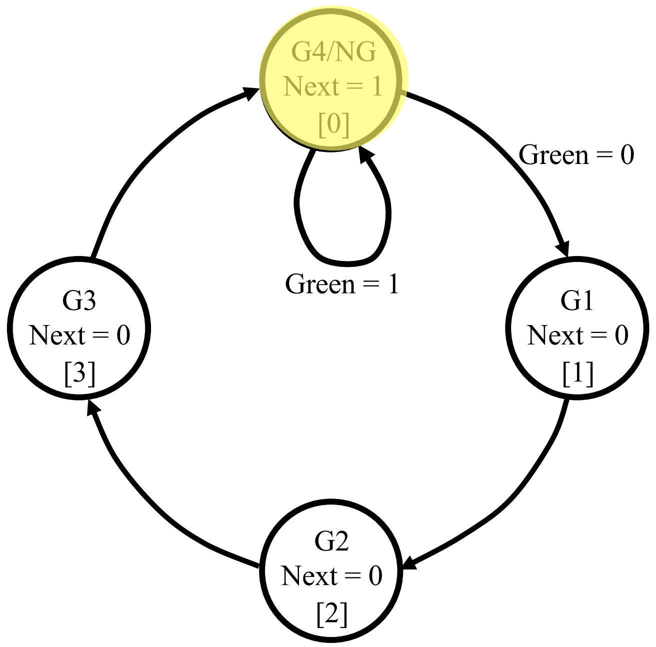
Timer FSM

Traffic Light Controller Operation



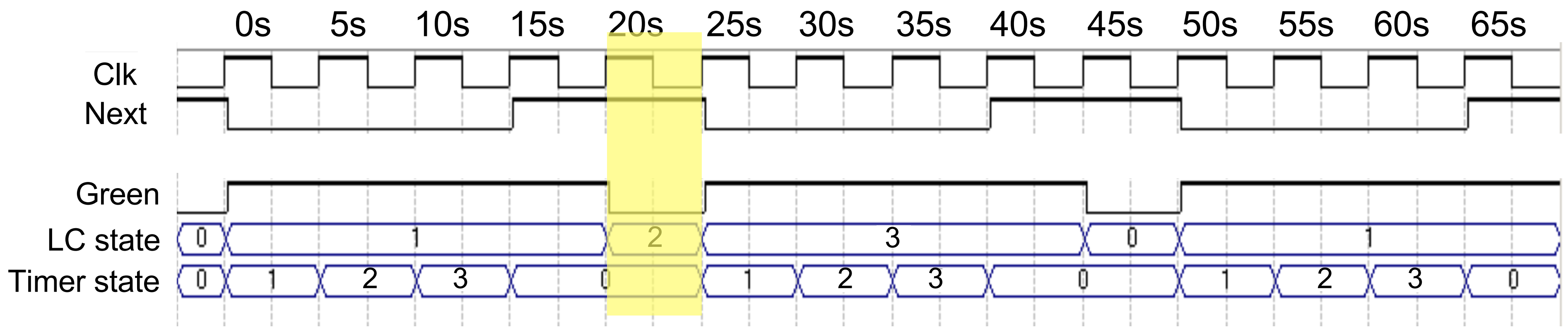


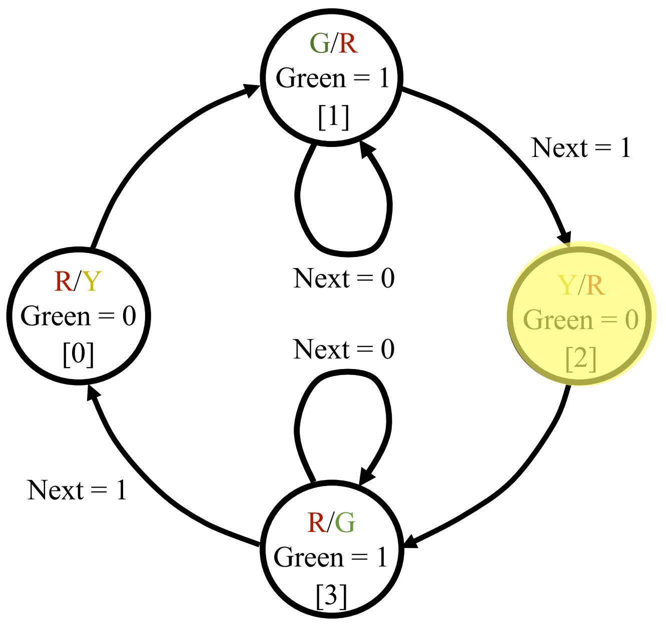
Light Controller FSM



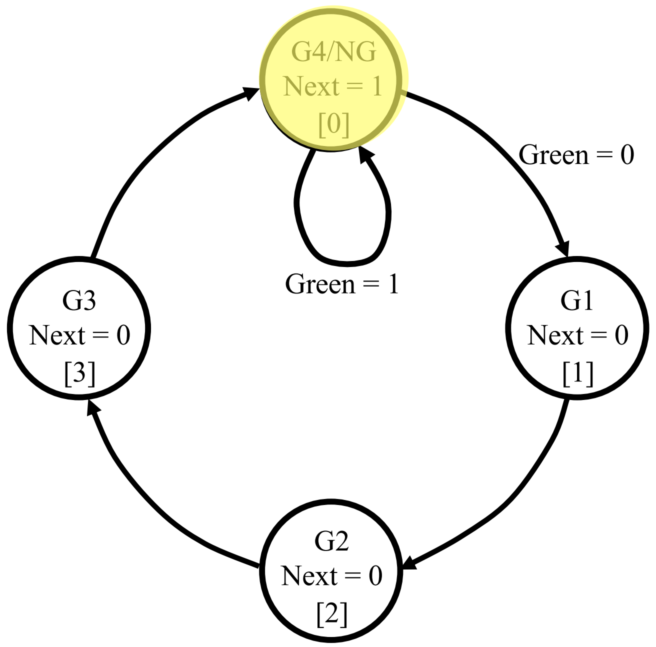
Timer FSM

Traffic Light Controller Operation



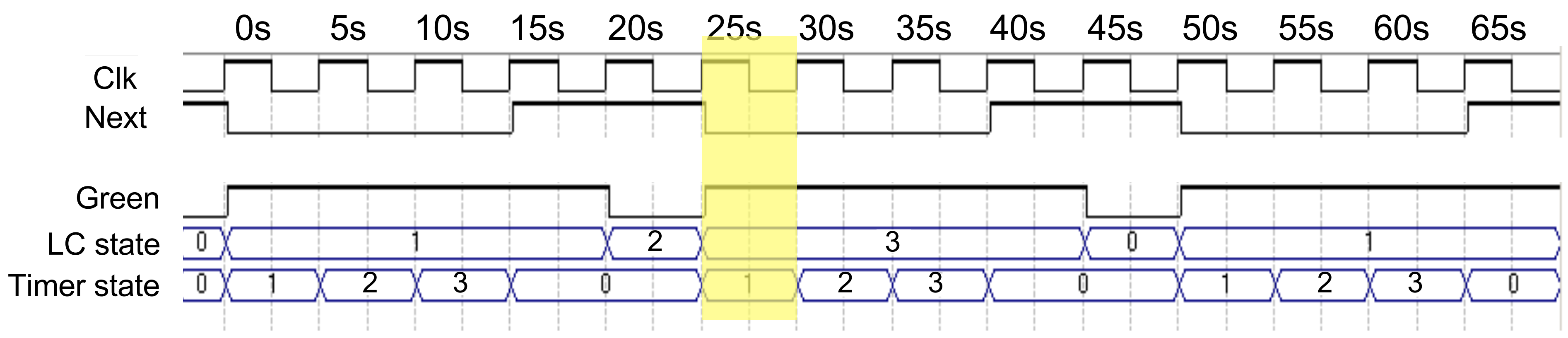


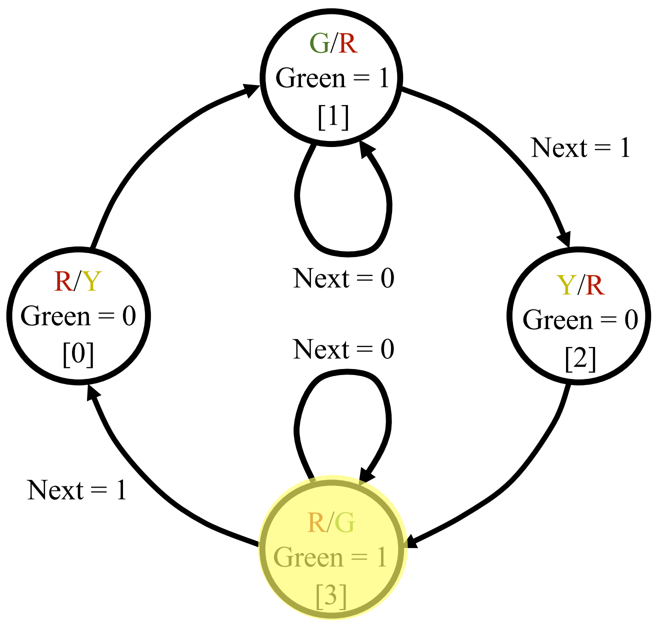
Light Controller FSM



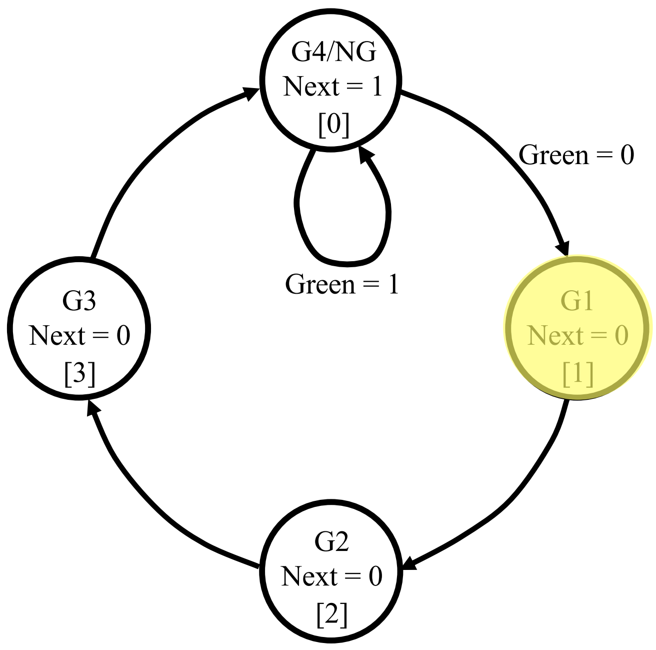
Timer FSM

Traffic Light Controller Operation





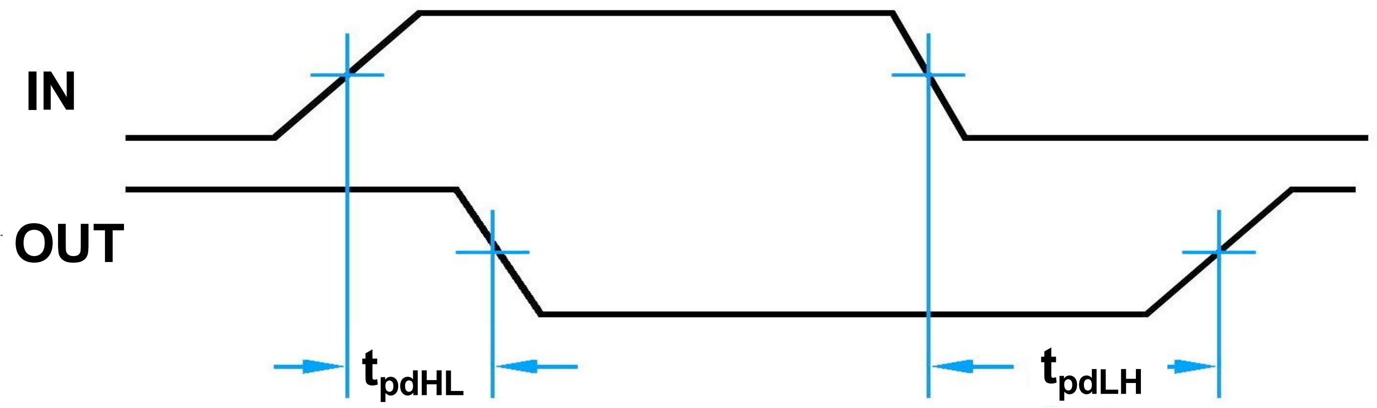
Light Controller FSM



Timer FSM

Propagation Delay (t_{pd})

- Time for change in input to change the output
  - Typically specified between 50% points



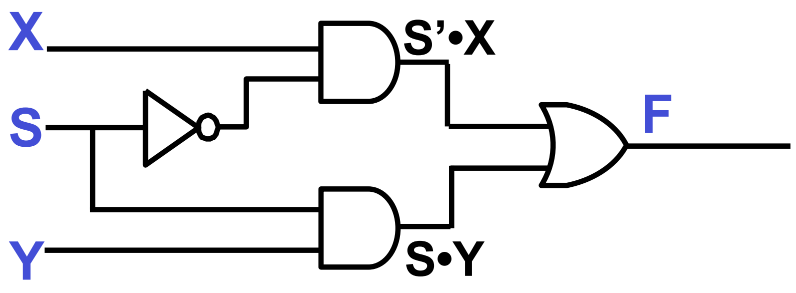
delay for transition from high to low

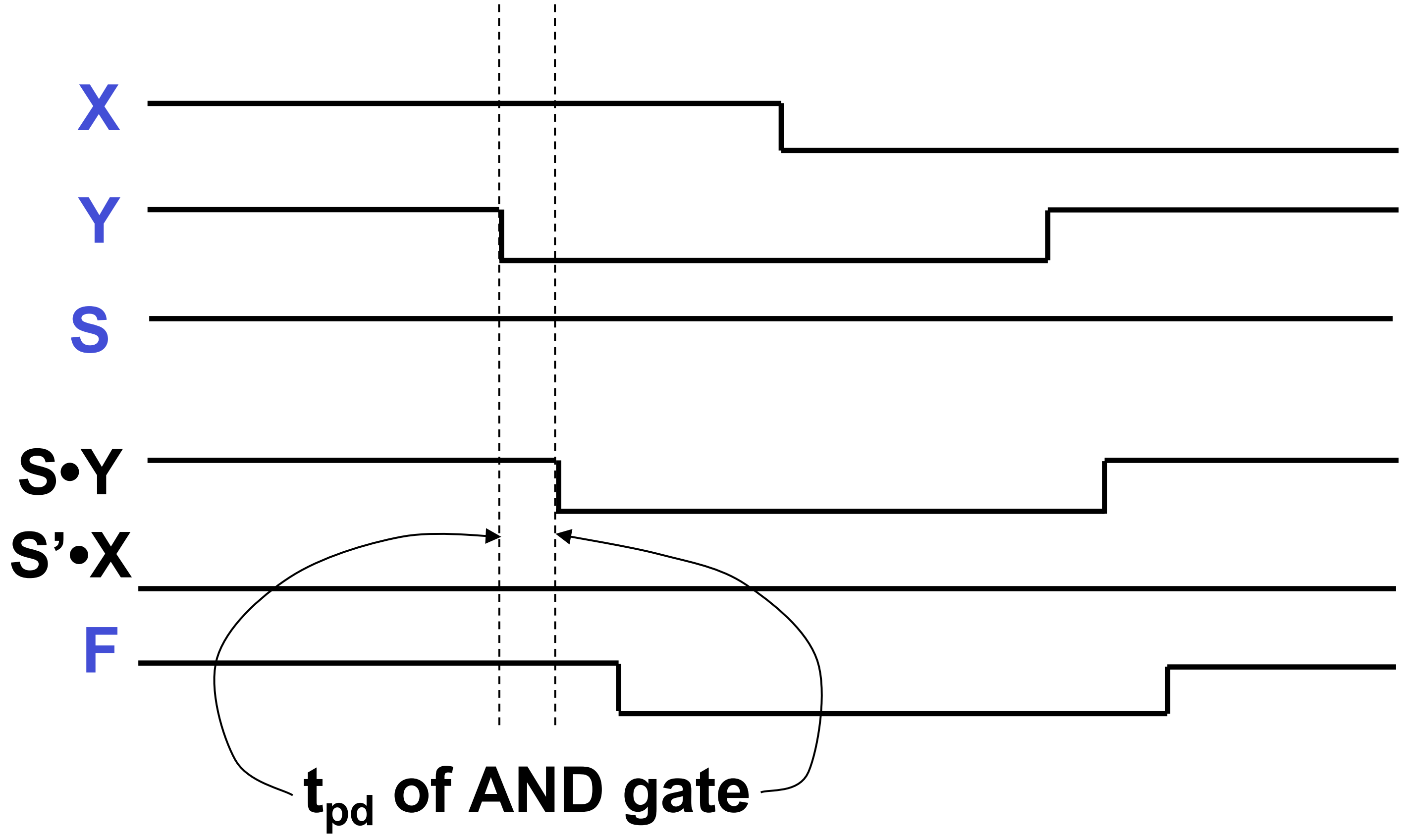
delay for transition from low to high

- Circuits have minimum and maximum propagation delays
  - Minimum sometimes called the *contamination delay* and maximum the *propagation delay*

Timing Diagram

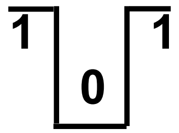
- Shows how outputs respond to changes in inputs over time



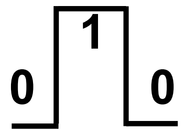


Glitch (Hazard)

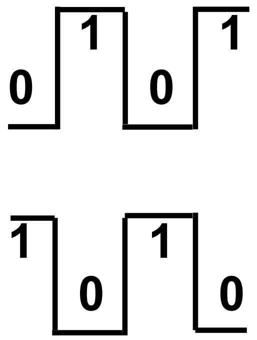
- Unplanned momentary switching of an output
- Types of glitches
  - Static 1-hazard: Input change causes output to go from 1 to 0 to 1 (should have stayed 1)



  - Static 0-hazard: Input change causes output to go from 0 to 1 to 0 (should have stayed 0)



  - Dynamic hazards: Input change causes a change from 0 to 1 to 0 to 1 or from 1 to 0 to 1 to 0 (there should be just one change)

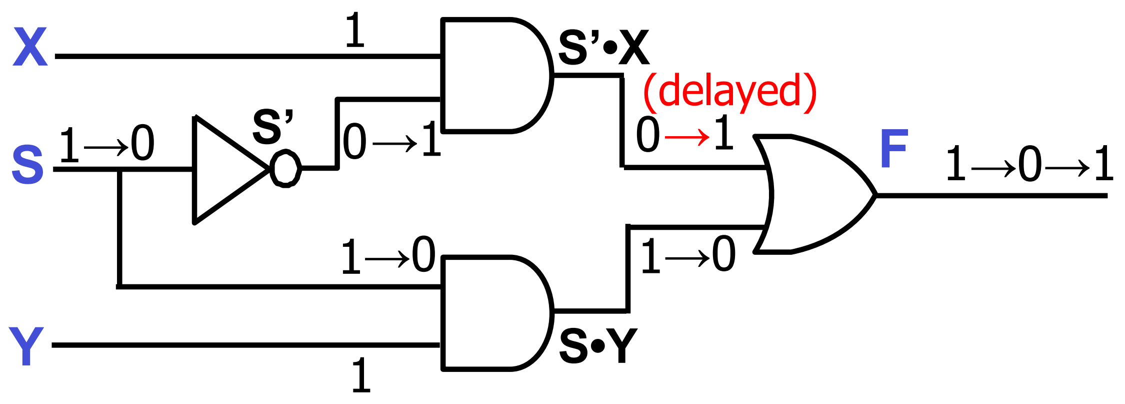


Glitch Example

- **Glitches are typically caused by unequal signal propagation delays through the circuit**

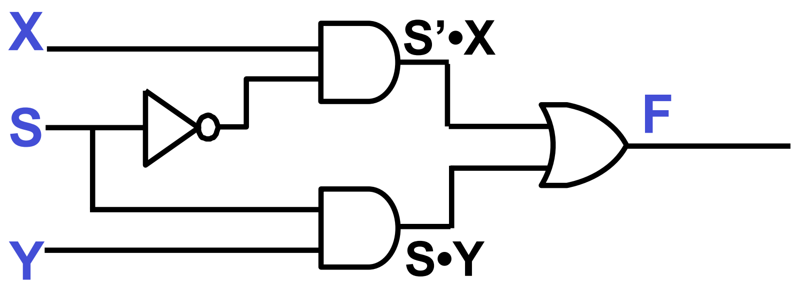
Assume X and Y are 1
S changes from 1 →0

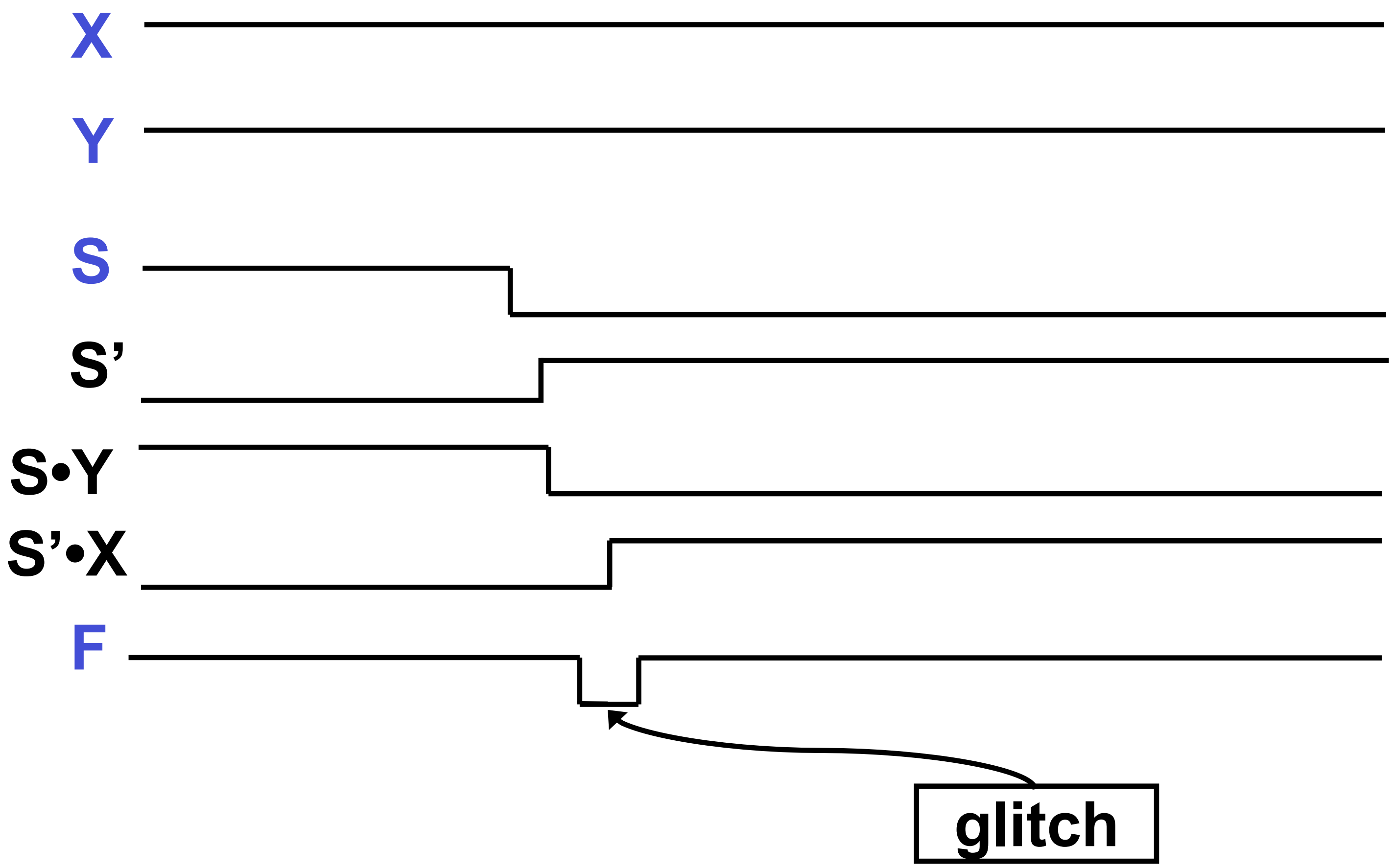
Expected behavior without considering delays: This is a 2:1 mux. Both inputs are 1, so F should have remained 1 regardless of the S value



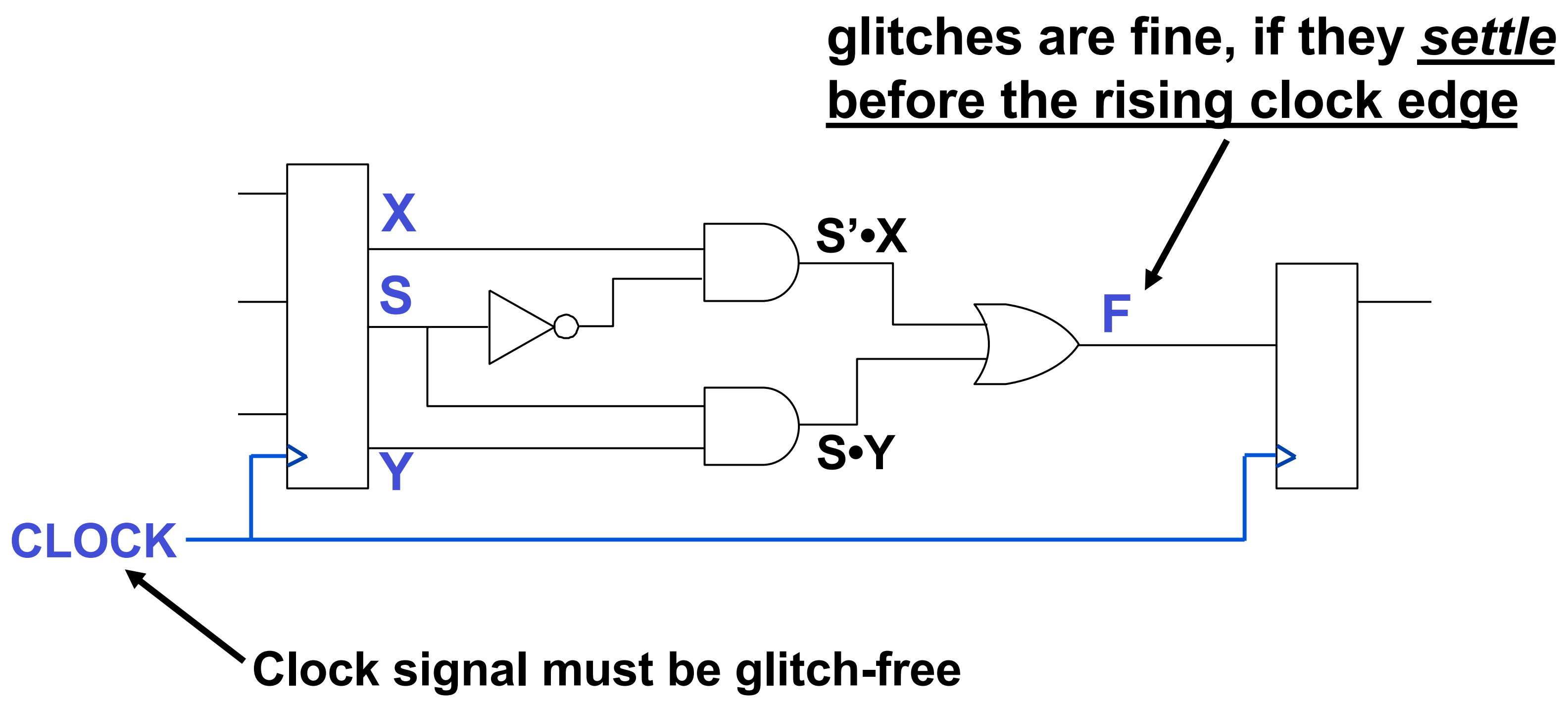
Actual behavior: Transition of S'•X is slower than that of S•Y because of the additional propagation delay on the top path ➔ F shows a transient 0 value

Timing Diagram Showing Glitch

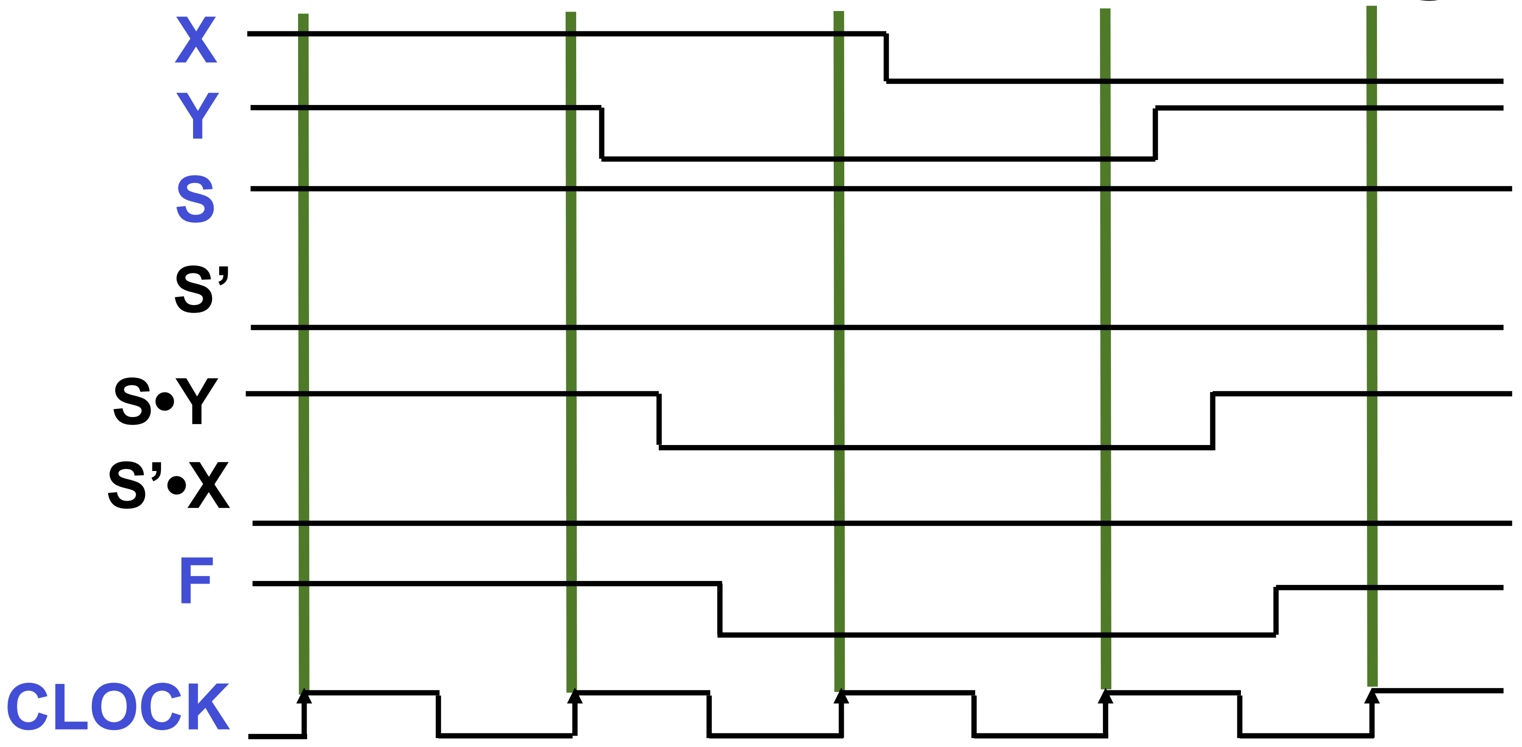


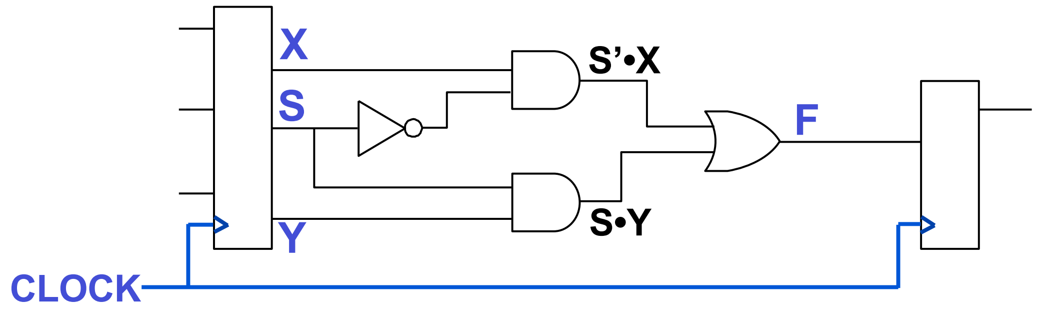


Do Glitches Matter?

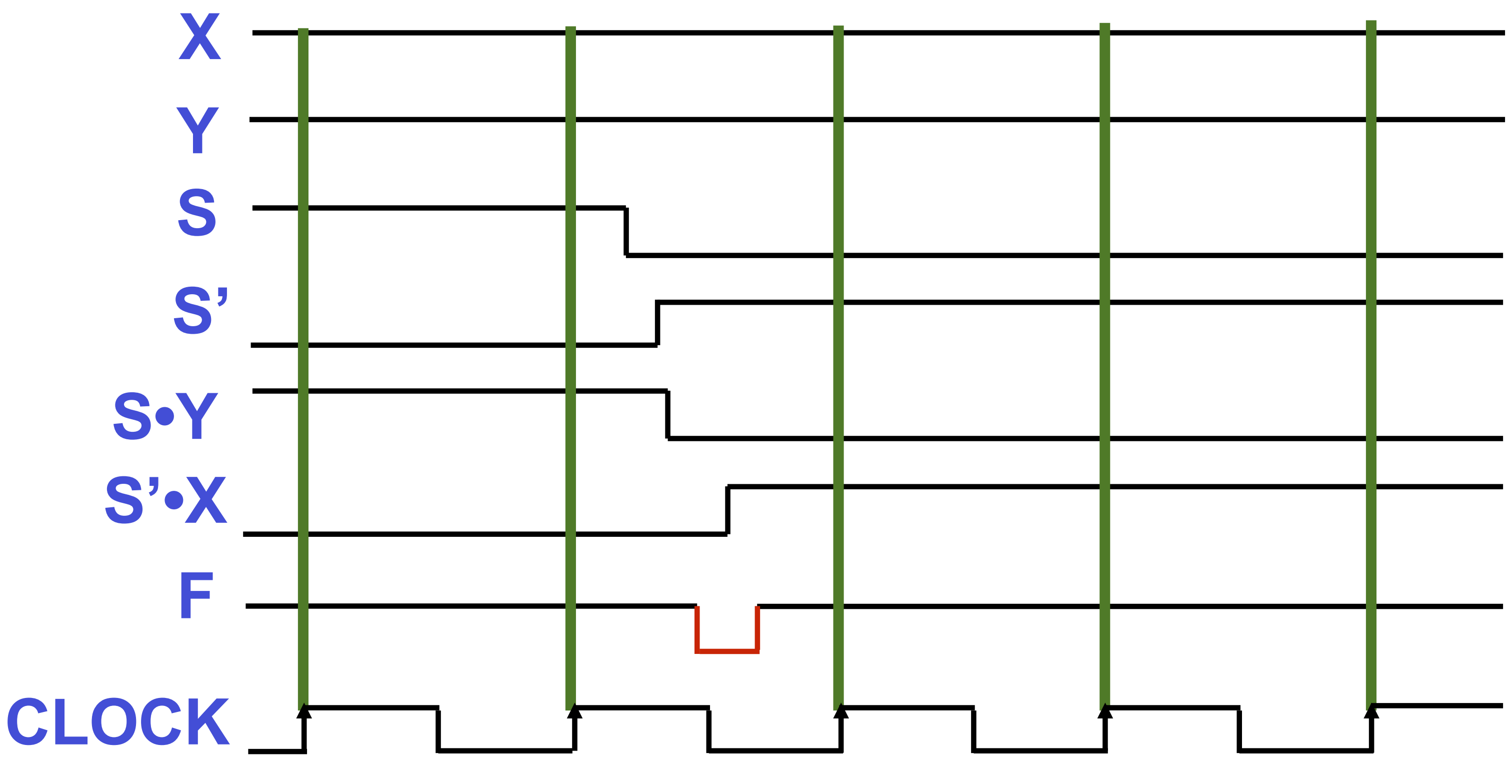

glitches are fine, if they ***settle*** **before the rising clock edge**
X
S
Y
S'•X
S•Y
F
CLOCK
Clock signal must be glitch-free

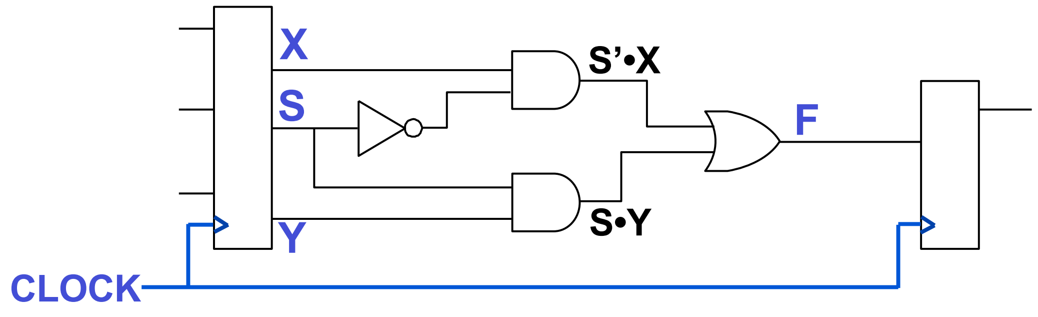
Sequential Circuit Timing



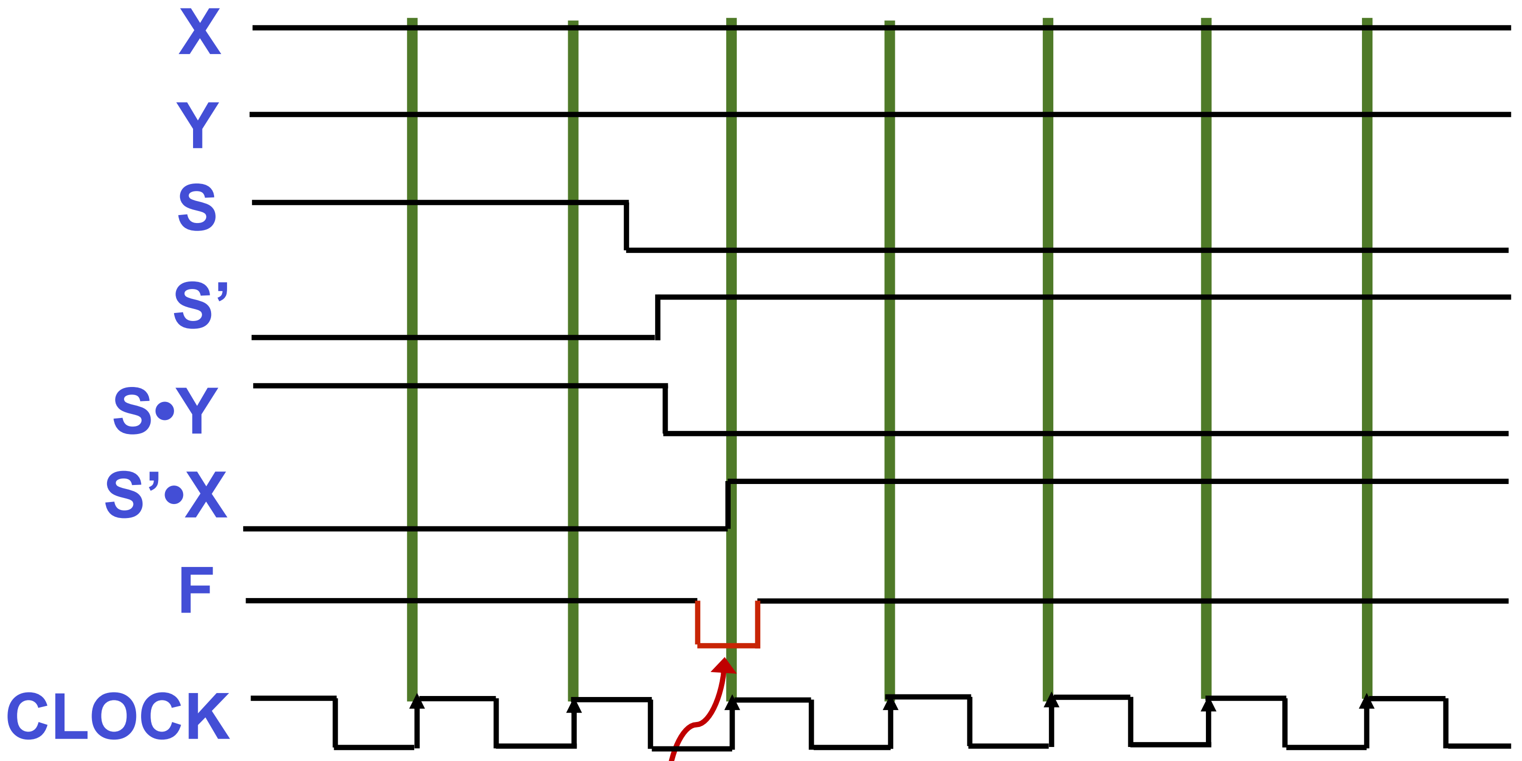


Glitch on F: No Problem

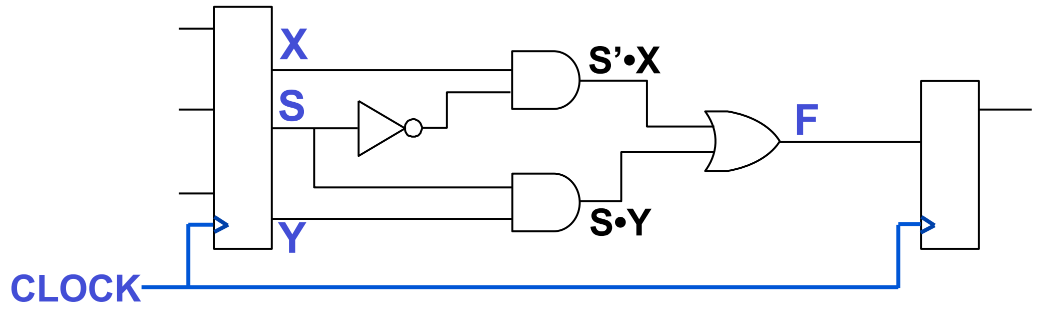




But What About This Situation?



Incorrect value captured; we'll go over how to prevent this in the next lecture



Next Class

Timing Analysis

(H&H 3.5)