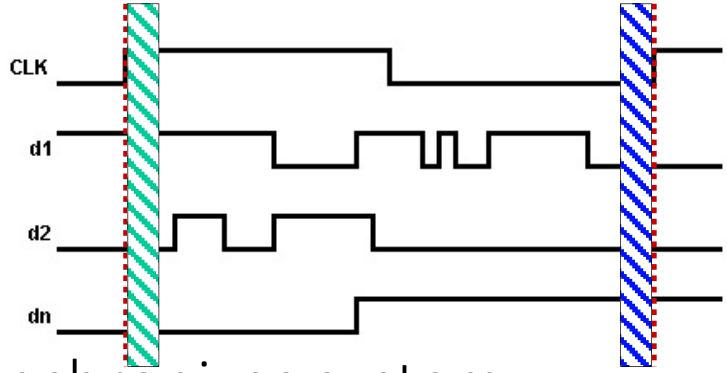
"… The primary finding: 87 percent of people who responded to the survey reported that they liked the idea of drone delivery."

"Interest in drone delivery is rising. The service in Christiansburg, run by Wing, Alphabet's drone-delivery subsidiary, is the most advanced of the handful of trial services operating today. But drone technology—and the laws that regulate it—are maturing, and it's expected that services like these could become routine in the next few years."

https://techxplore.com/news/2021-04-population-drone-delivery.html

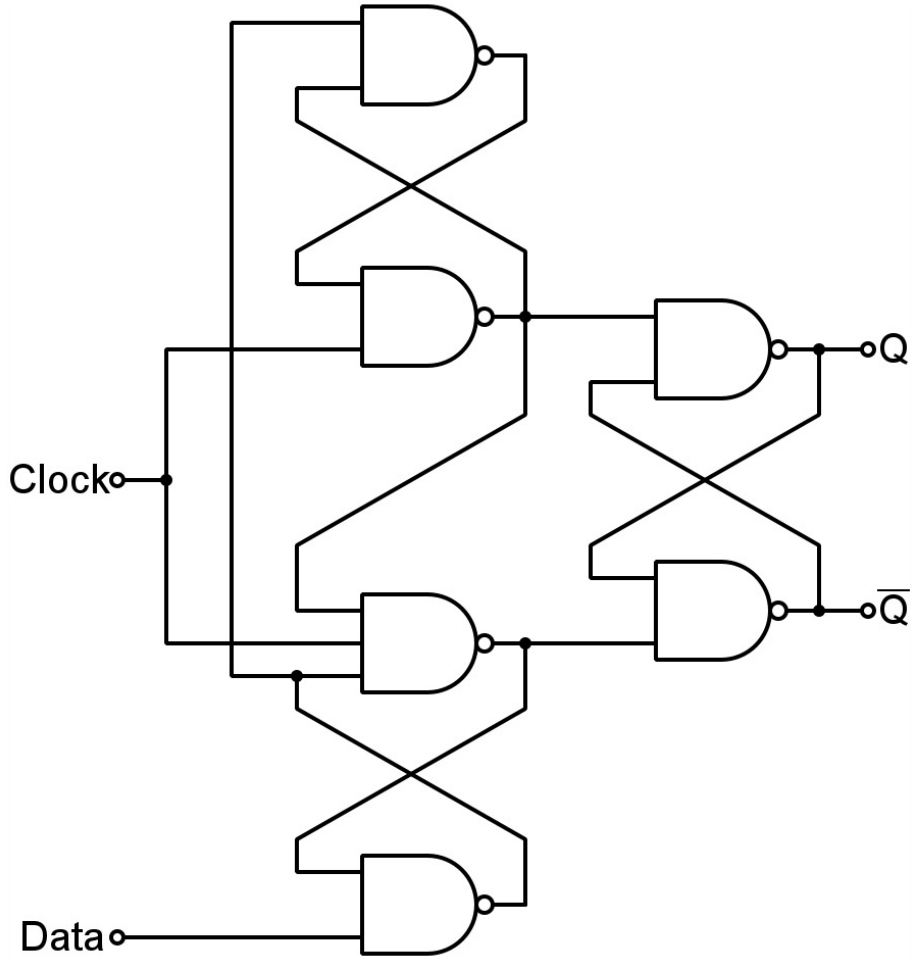
Review of Timing Terms



- Clock: steady square wave that synchronizes system
- Flip-flop: one bit of state that samples every rising edge of CLK (positive edge-triggered)
- Register: several bits of state that samples on rising edge of CLK (positive edge-triggered); often has a RESET
- Setup Time: when input must be stable *before* CLK trigger
- Hold Time: when input must be stable *after* CLK trigger
- CLK-to-Q Delay: how long it takes output to change from CLK trigger

Edge-triggered D flip-flop

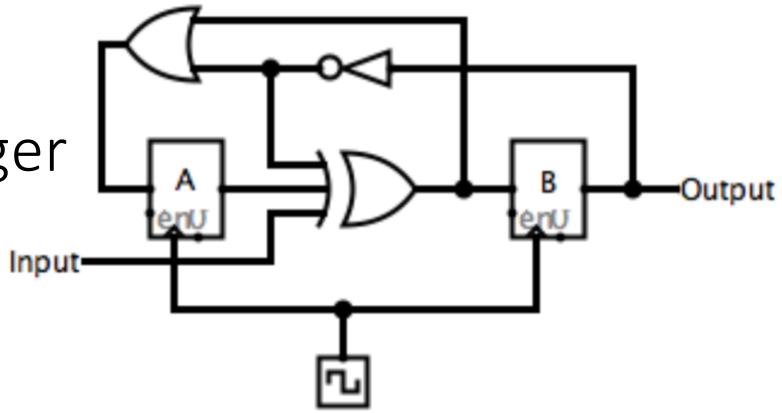
Edge-triggered
D flip-flop:



By Nolanjshettle at English Wikipedia, CC BY-SA 3.0,
https://commons.wikimedia.org/w/index.php?curid=40852354

SDS Timing Question (all times in ns)

- The circuit below has the following timing parameters
  - $t_{\text{period}} = 20, t_{\text{setup}} = 2$
  - $t_{\text{XOR}} = t_{\text{OR}} = 5, t_{\text{NOT}} = 4$
  - Input changes 1 ns after clock trigger



- What is the max t_{C2Q}?

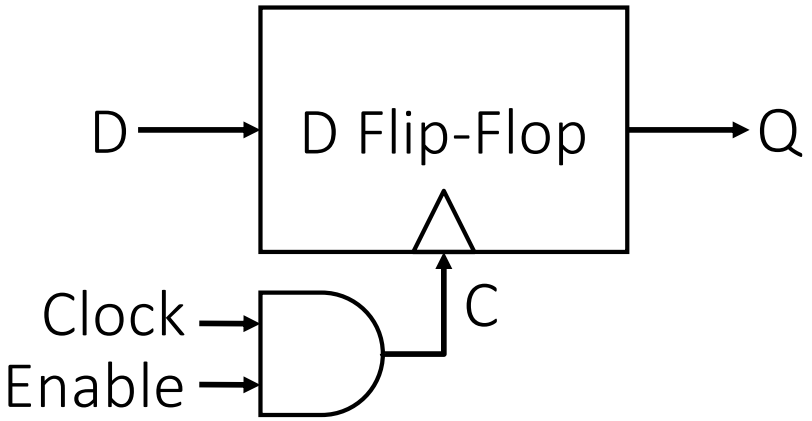
- If $t_{\text{C2Q}} = 3$, what is the max t_{hold}?

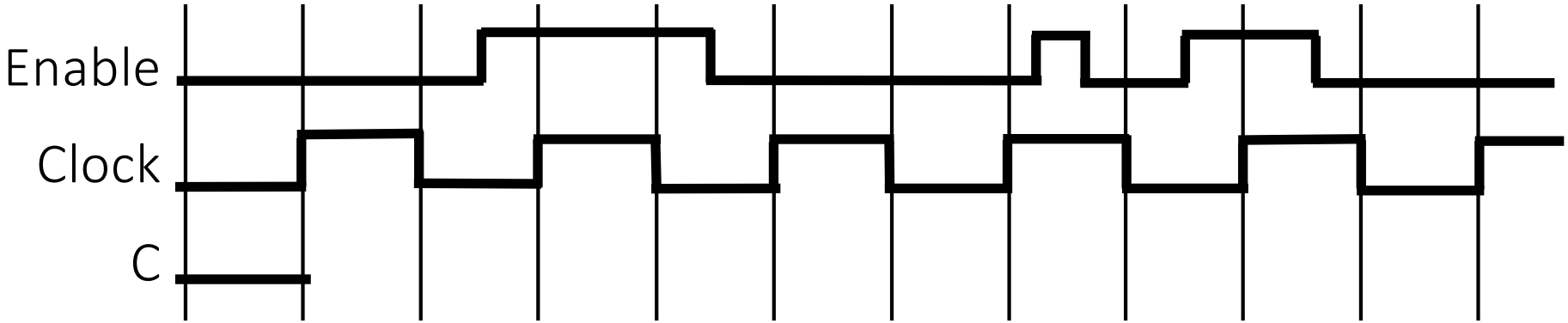
Outline

- Flip-Flop Realities
- Finite State Machines
- FSMs in Verilog

Flip-Flop Realities: Gating the Clock

- Delay can cause part of circuit to get out of sync with rest
  - More timing headaches!
  - Adds to *clock skew*
- Hard to track non-uniform triggers



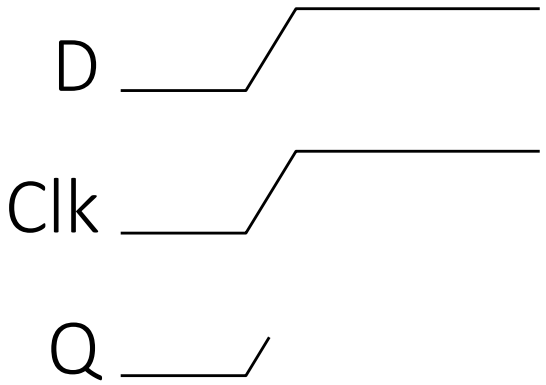


- NEVER GATE THE CLOCK!!!

Flip-Flop Realities: External Inputs

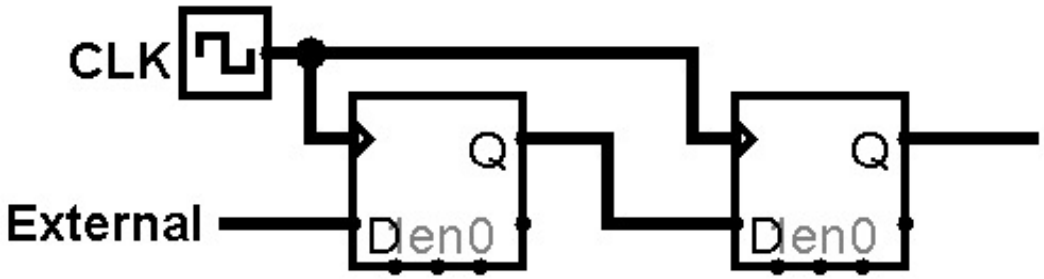
- External inputs aren't synchronized to the clock
  - If not careful, can violate timing constraints

- What happens if input changes around clock trigger?



Flip-Flop Realities: Metastability

- **Metastability** is the ability of a digital system to persist for an unbounded time in an unstable equilibrium or metastable state
  - Circuit may be unable to settle into a stable '0' or '1' logic level within the time required for proper circuit operation
  - Unpredictable behavior or random value
  - https://en.wikipedia.org/wiki/Metastability_in_electronics
- State elements can help reject transients
  - Longer chains = more rejection, but longer signal delay

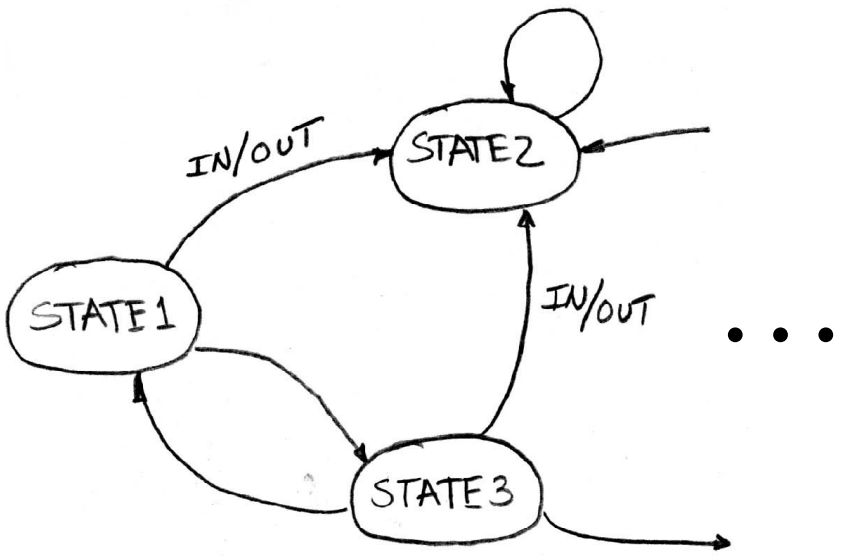


Outline

- Flip-Flop Realities
- Finite State Machines
- FSMs in Verilog

Finite State Machines (FSMs)

- A convenient way to conceptualize computation over time
  - Function can be represented with a *state transition diagram*
  - You've seen these before in CSE311
- **New for CSE369:** Implement FSMs in hardware as synchronous digital systems
  - Flip-flops/registers hold "state"
  - Controller (state update, I/O) implemented in combinational logic



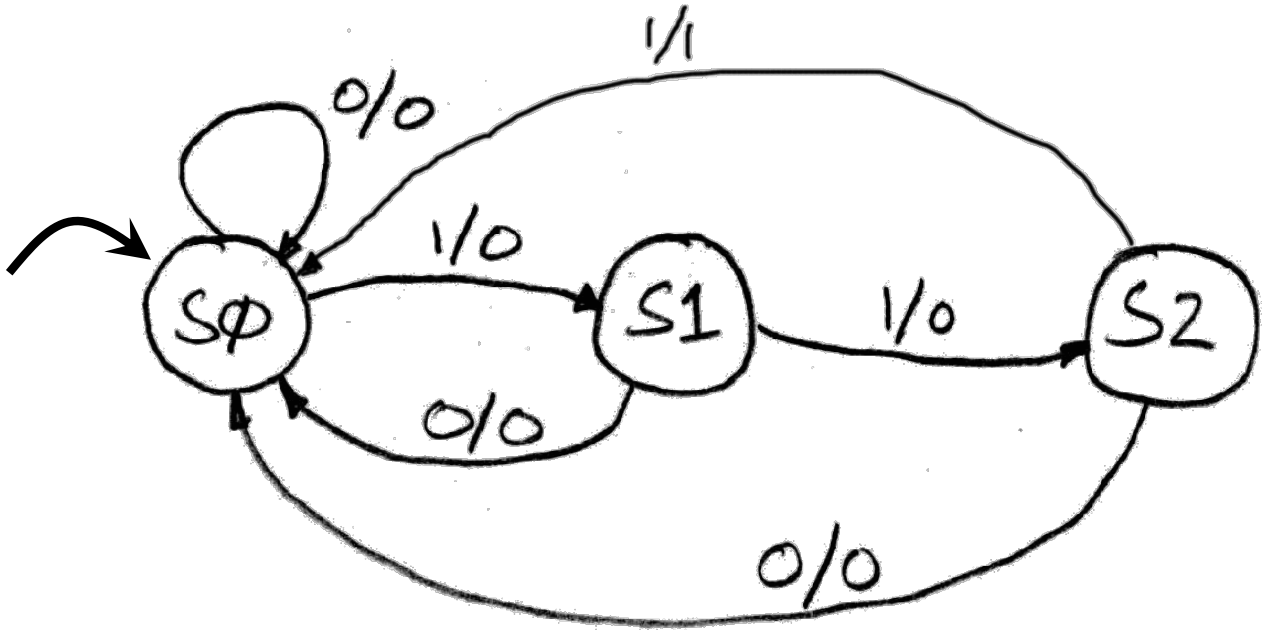
State Diagrams

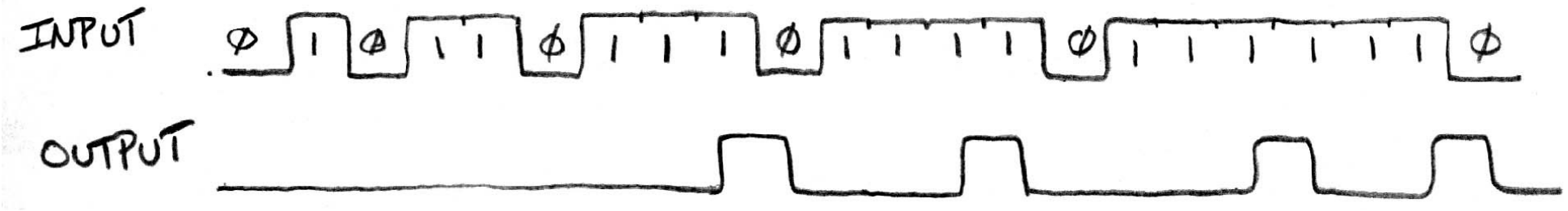
- An state diagram (in this class) is defined by:
  - A set of *states* S (circles)
  - An *initial state* s_0 (only arrow not between states)
  - A *transition function* that maps from the current input and current state to the output and the next state (arrows between states)
    - **Note:** We cover Mealy machines here; Moore machines put outputs on states, not transitions

- State transitions are controlled by the clock:
  - On each clock cycle the machine checks the inputs and generates a new state (could be same) and new output

Example: Buggy 3 Ones FSM

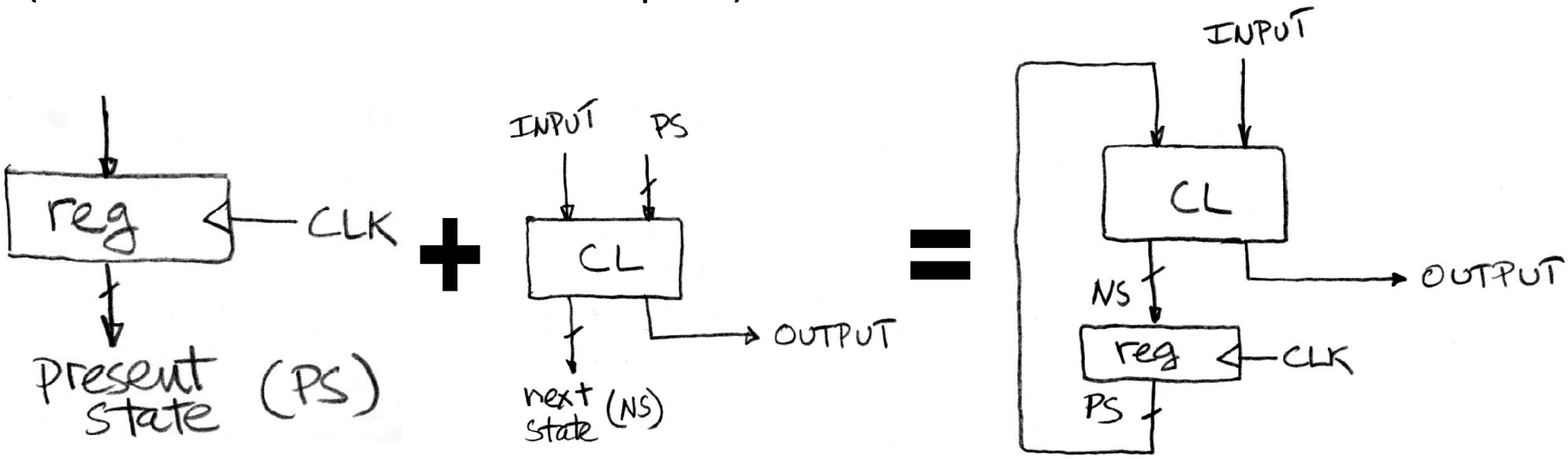
- FSM to detect 3 consecutive 1's in the Input





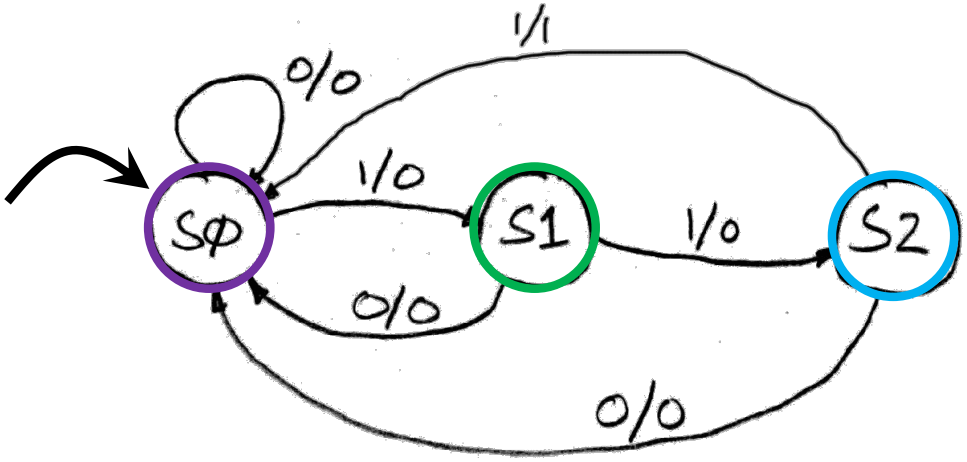
Hardware Implementation of FSM

- Register holds a representation of the FSM's state
  - Must assign a *unique* bit pattern for each state
  - Output is *present/current state* (PS/CS)
  - Input is *next state* (NS)
- Combinational Logic implements transition function (state transitions + output)



FSM: Combinational Logic

- Read off transitions into Truth Table!
  - **Inputs:** Present State (PS) and Input (In)
  - **Outputs:** Next State (NS) and Output (Out)



PS	In	NS	Out
00	0	00	0
00	1	01	0
01	0	00	0
01	1	10	0
10	0	00	0
10	1	00	1

- Implement logic for *EACH* output (2 for NS, 1 for Out)