

Discussion 06

SDS, Logic, FSM

Aditya Balasubramanian

`aditbala [at] berkeley [dot] edu`

Announcements

Agenda

- SDS
- FSM
- Logic

Definitions (Pt. 1)

- Clk
 - Central timing unit of the entire SDS; usually only one clock per system
- State element
 - Any clocked element: stores values
 - Only does computation things at the rising edge of the clock
 - E.g. registers
- Logic element
 - Any unclocked elements: does not store value
 - Computes ALL THE TIME!
 - E.g. combinatorial logic elements (AND gates, OR gates, etc.)

Definitions (Pt. 2)

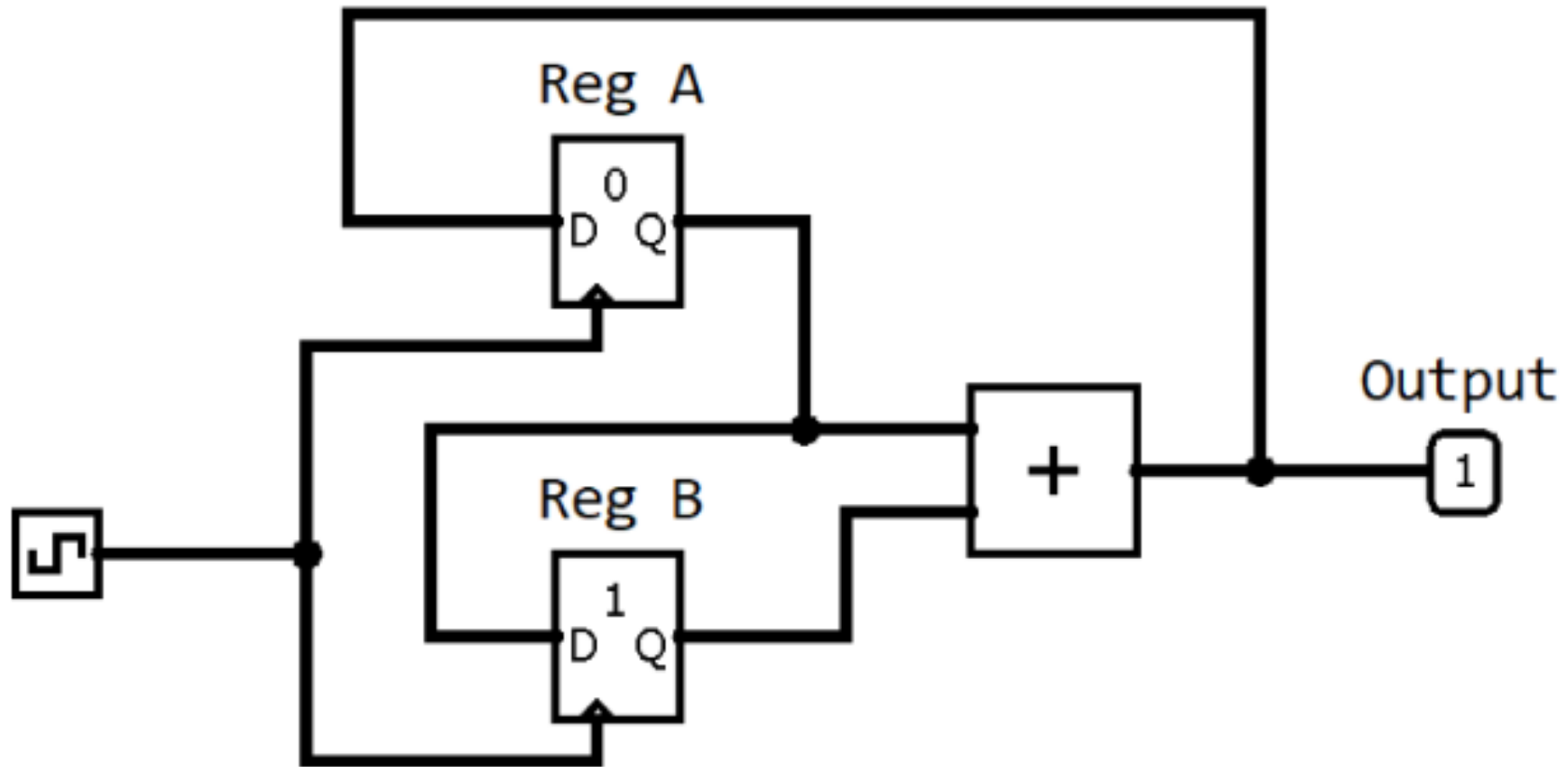
- Flip-flop
 - state element that stores 1 bit's value (0/1)
- Register
 - **n**-bit state element; created with **n** chained FFs
 - **D** = input; **Q** = output
 - Reads value from **D** at clock tick, puts value into **Q**

Definitions (Pt. 3)

- Rising clock edge (RCE)
 - clk goes from 0 \Rightarrow 1; Usually instantaneous
 - Triggers all the state elements dependent directly on the clock
- Falling clock edge
clk goes from 1 \Rightarrow 0
- Setup time
 - Time BEFORE RCE where input must be stable
- Hold time
 - Time AFTER RCE where input must be stable
- Clock-to-q time (c2q)
 - Time after RCE needed for value in Q to change

Definitions (Pt. 4)

- Combinational logic delay
 - Combinatorial delay between 2 state elements
 - Usually Sum of total delays within the path from the Q of one register to the D of another (or the same) register
- Critical path
 - Total delay between 2 state elements
 - Clk-to-q (reg1) + longest CL + setup time (reg2)
- Maximum clock frequency
 - $1 / \text{minimum clock period}$



Equations

$$\text{max hold time} \leq t_{\text{clock to q}} + \text{shortest CL}$$

- Any longer of a hold time means that value has potential to change

$$\text{cycle time} \geq t_{\text{clock to q}} + \text{longest CL} + t_{\text{setup}}$$

- Cycle time = clock period
- Any shorter cycle time means the values may not finish computing correctly in time

FSM (Finite State Machine)

- An FSM takes in a sequence of characters (bits), and outputs another sequence of characters (bits).
- We represent a FSM by a number of states, plus transitions between the states
 - State labels don't usually have meanings
 - Arrow going from state A to B labeled input/output
 - When we're currently in state A and see input, we move on to state B and print output

Thank you!

Feedback