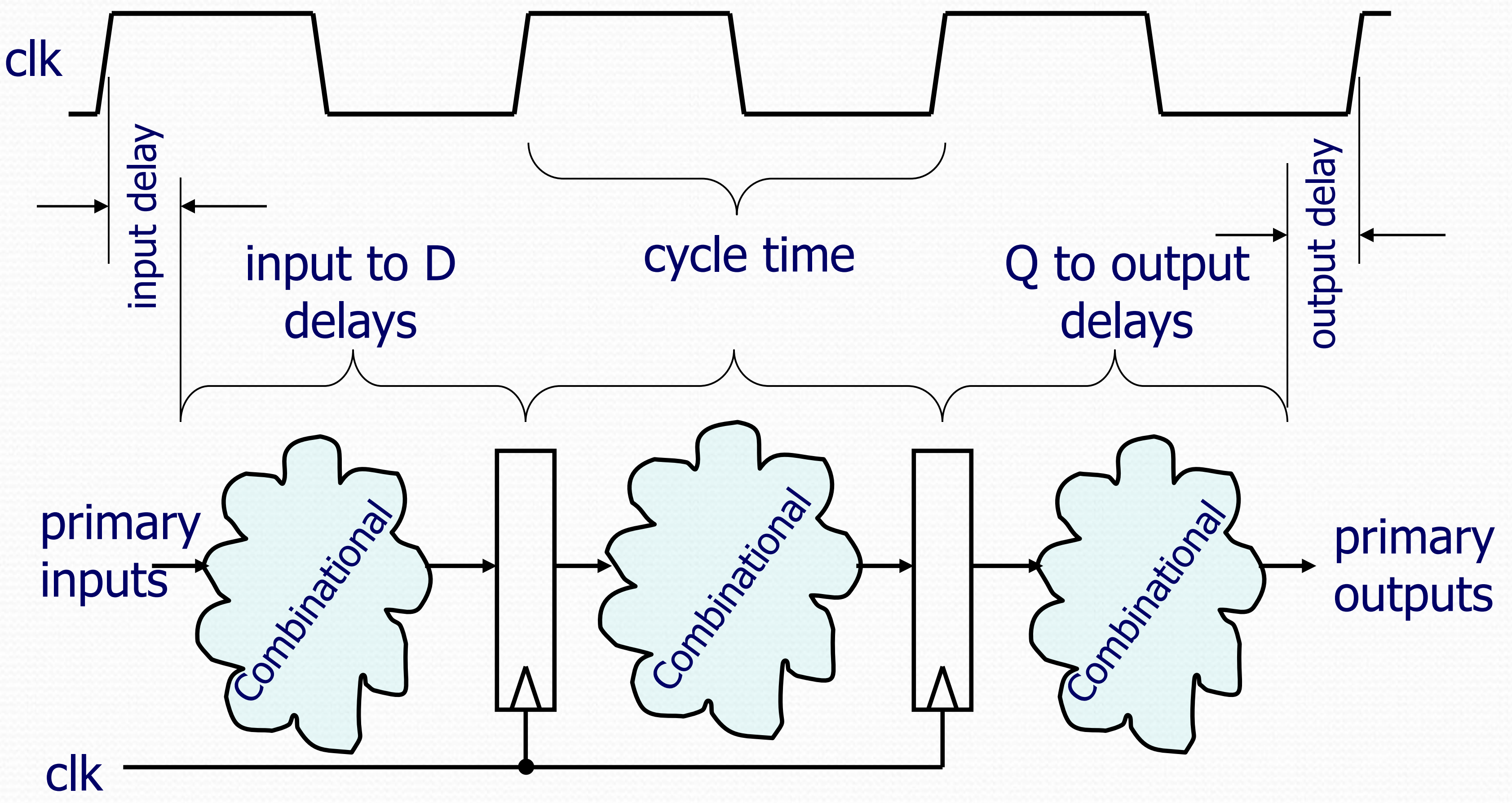
Exercise 23 Static Timing Analysis

(we have flops…and combinational logic…this stuff is not hard. Its just simple arithmetic.



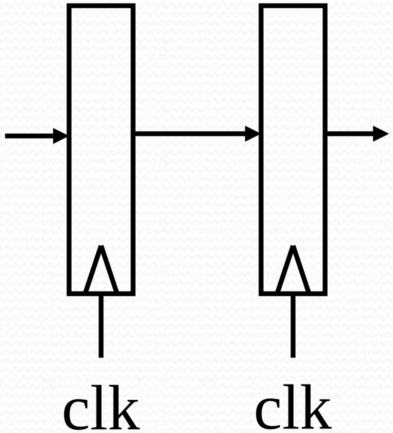
Exercise 23: Know your definitions.

- **clk2q** delay ➔ The delay from a active edge of clock (positive edge in our case) to when the output (Q output) is known valid. The name tells you what it is…how hard is that to remember?

- **setup** time ➔ The amount of time the D input of a flop needs to be valid prior to the active edge of clock (positive edge in our case). You already knew this right?

Exercise 23: Know your definitions.

- **hold** time ➔ The minimum amount of time the D input of a flop has to be held after the active edge of clock (positive edge in our case). If the D input changes prior to this time there is the danger that the flop will grab the new value the D input is assuming, not the value it was just prior to the clock edge. OK…this one is a little trickier to understand…still it is not that hard of a concept.



Back to back flops is the worst case topology for hold time (min delay). Do you want it to go through both flops in one clock edge?

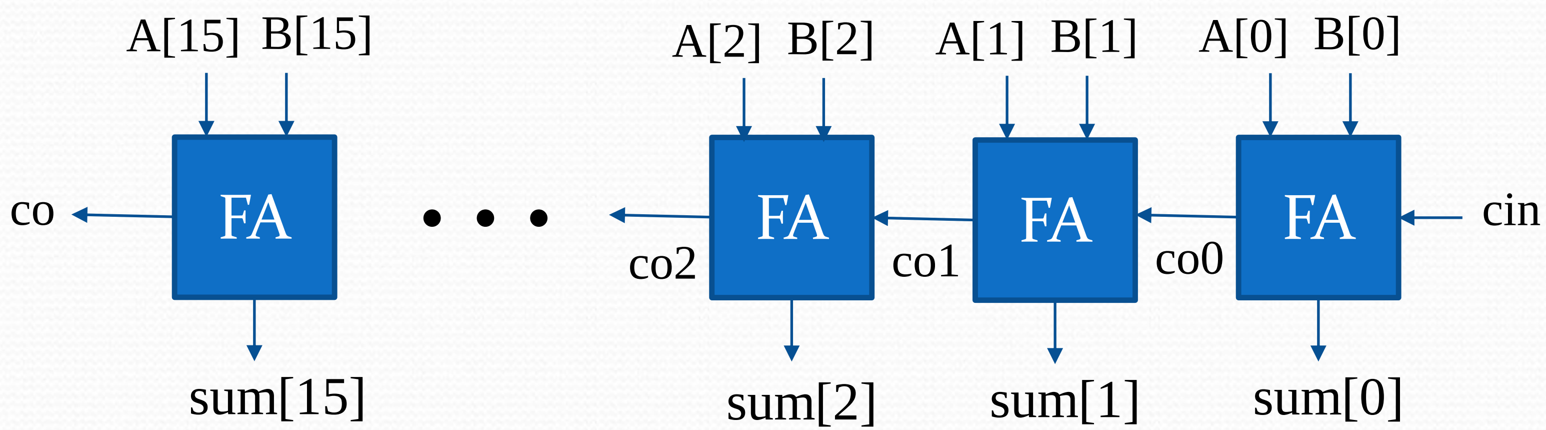
if (clk2q > T_{HOLD}) we are golden

(what about clock skew?)

Exercise 23: Know your definitions.

- max path ➔ The worst case (maximum) amount of time it takes to propagate from an input of a combinational block of logic to the output. This can be vector dependent.

16-bit Ripple Carry Adder

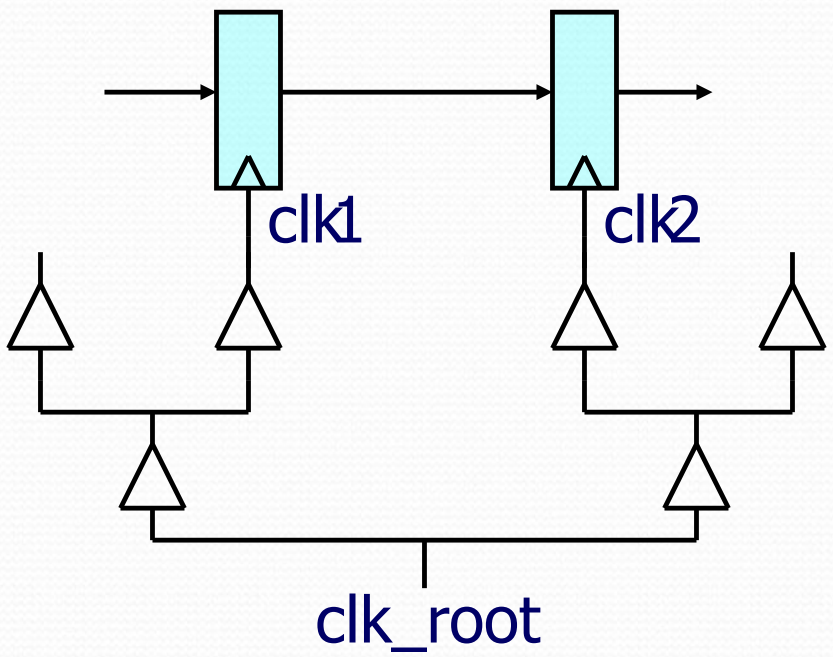


Consider the delay for inputs:	Consider the delay for inputs:

`A[15:0] = 16'hAAAA;`
`B[15:0] = 16'h5555;`
`cin = 1'b0;`
	`A[15:0] = 16'hAAAA;`
	`B[15:0] = 16'h5555;`
	`cin = 1'b1;`

Exercise 23: Know your definitions.

- min path ➔ The minimum amount of time it takes to propagate from an input of a combinational block of logic to the output.

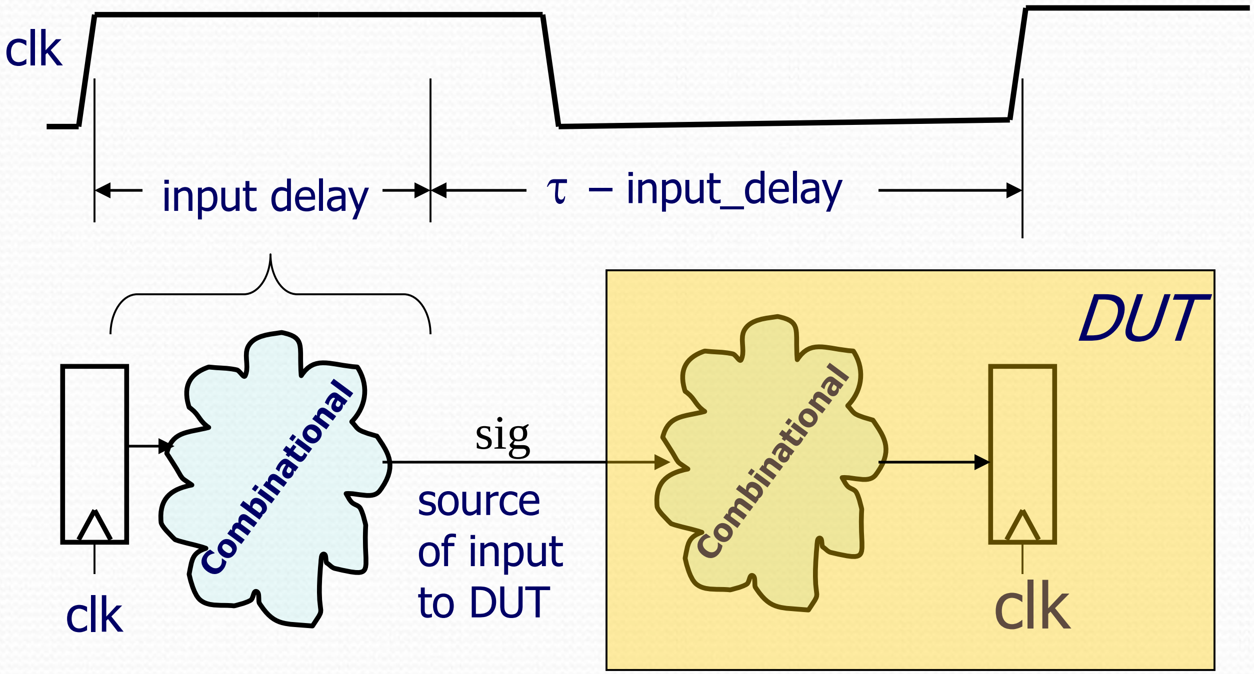


Clock skew ➔ try as hard as you might…it is impossible to distribute a clock to 100k+ flops on chip and have the clock arrive precisely at the same time to all the flops.

There is uncertainty (a +/- margin) in a clocks arrival time.

Exercise 23: Know your definitions.

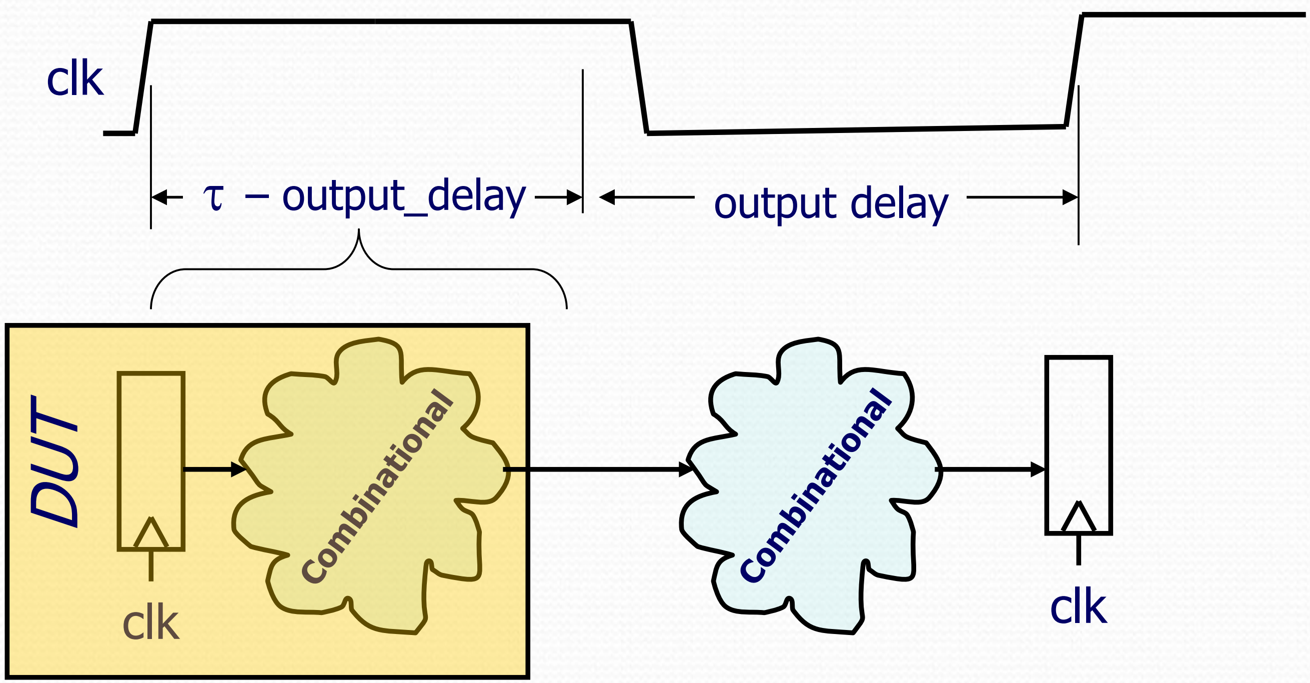
- Remember how Synopsys defines input delay



input delay is specified as time after the clock edge (of prior clock cycle) that the input to the DUT is valid.

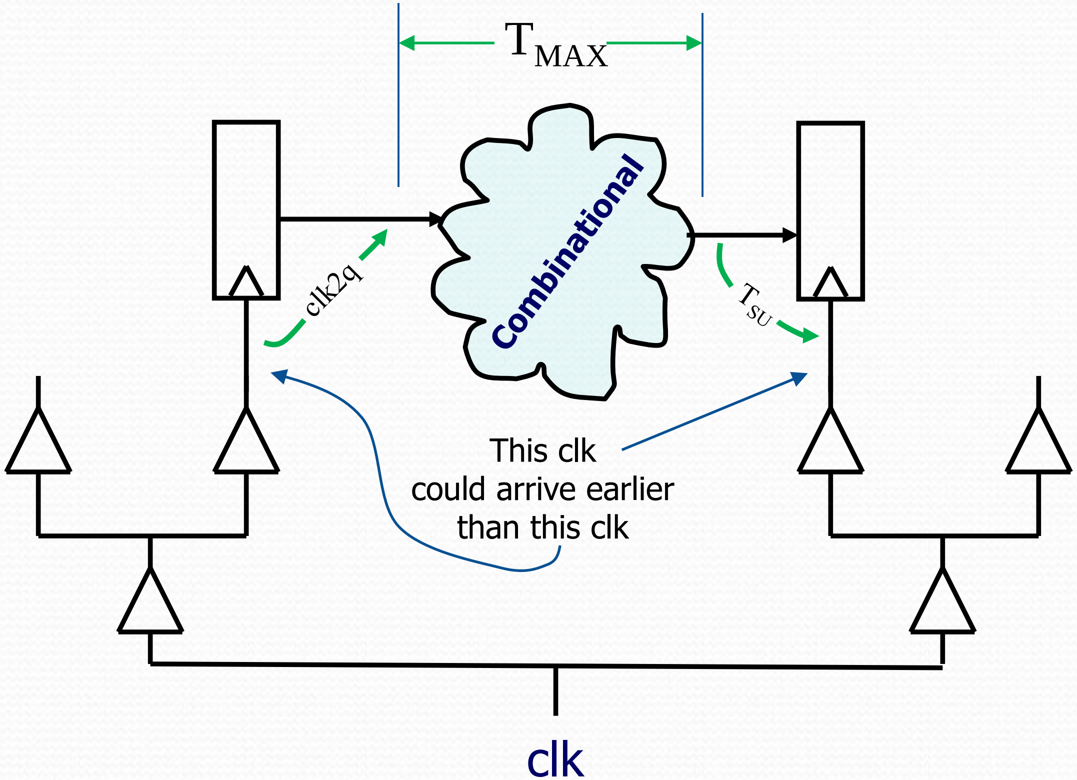
Exercise 23: Know your definitions.

- Remember how Synopsys defines output delay



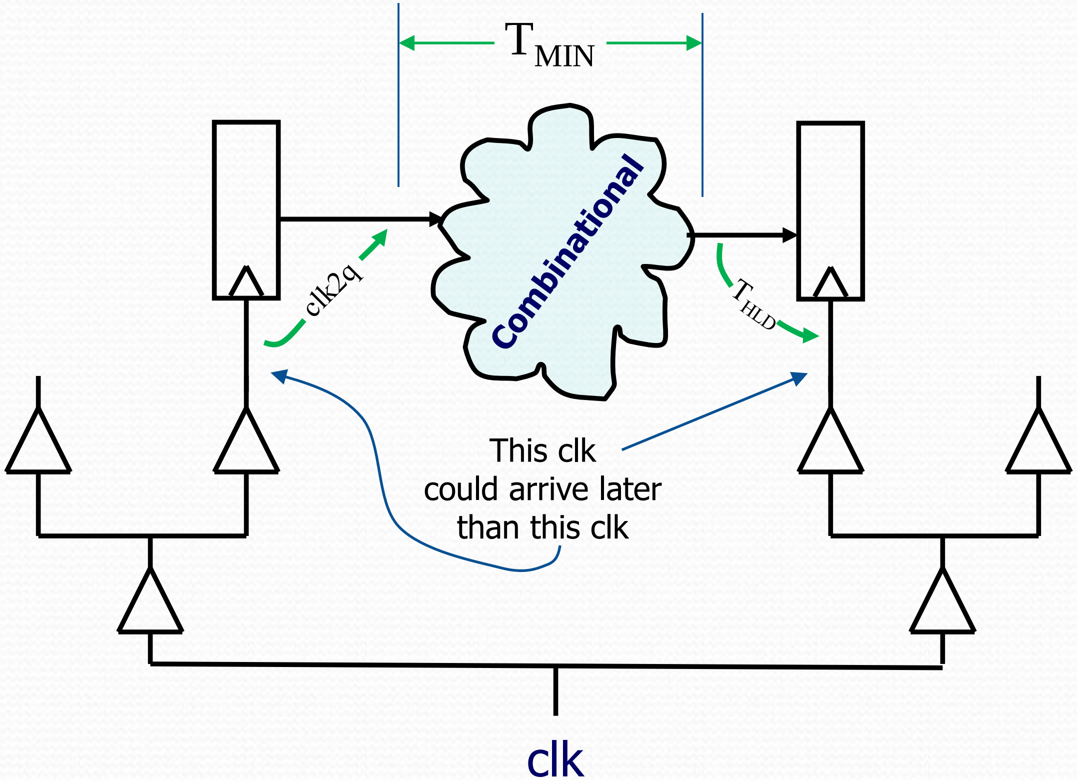
output delay is specified as time prior to next rising edge that the output has to be valid.

Exercise 23: When Considering Max Delay *(Is my circuit fast enough?)*



$$MaxDelaySlack = ClockPeriod - clk2q - T_{MAX} - T_{SU} - clkSkew$$

Exercise 23: When Considering Min Delay *(Is my circuit too fast. I am going to have a shoot through problem)*



$$\text{MinDelaySlack} = \text{clk2q} + T_{MIN} - T_{HLD} - \text{clkSkew}$$

If you have to write these formulas down on your cheat sheet for the final…then you had better be really good looking to get by in life. If you understand the definitions, these formulas should be obvious.

Exercise 23:

There is an online quiz for Exercise23. Take the quiz.