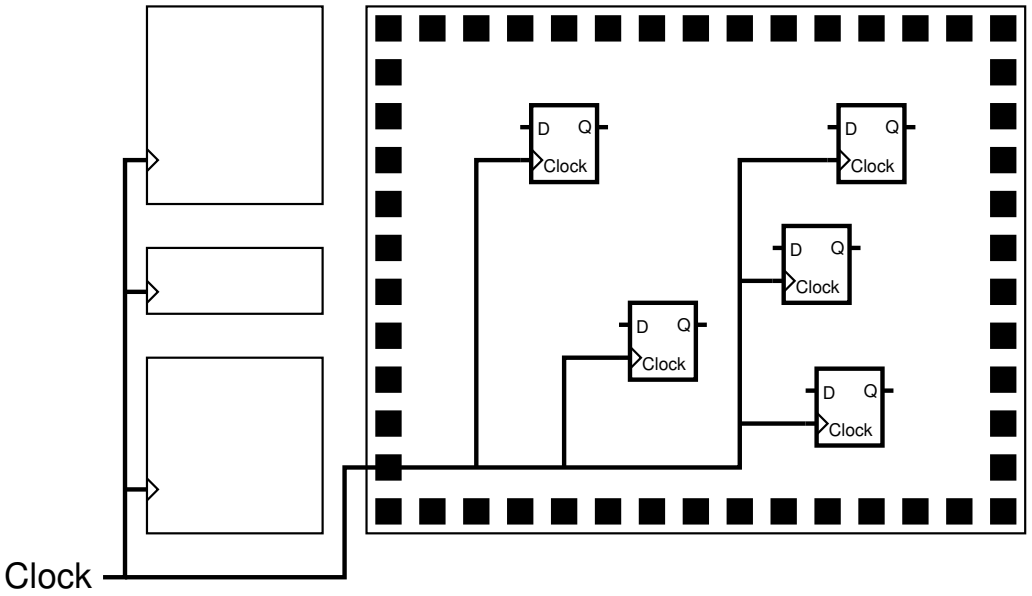
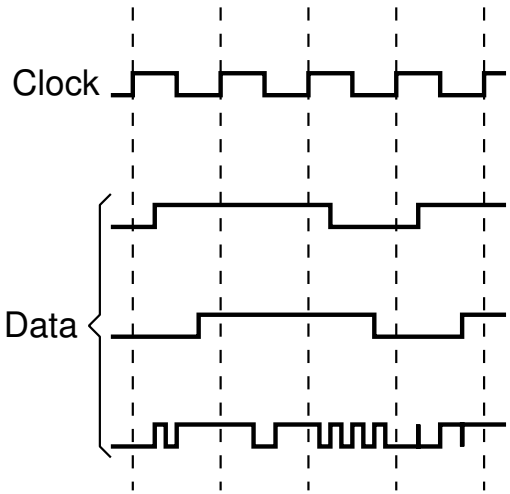
Synchronous Systems

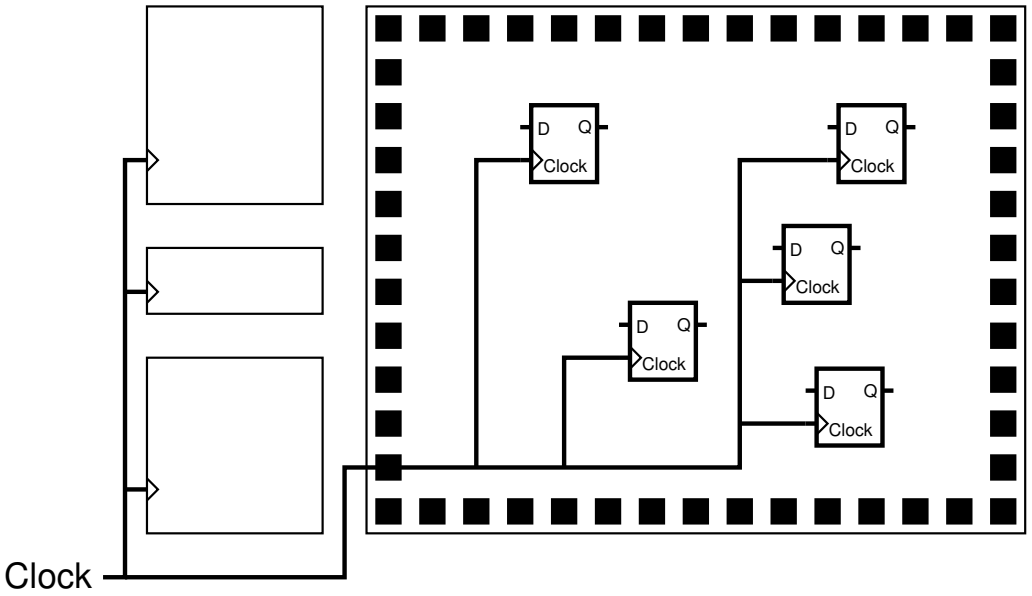


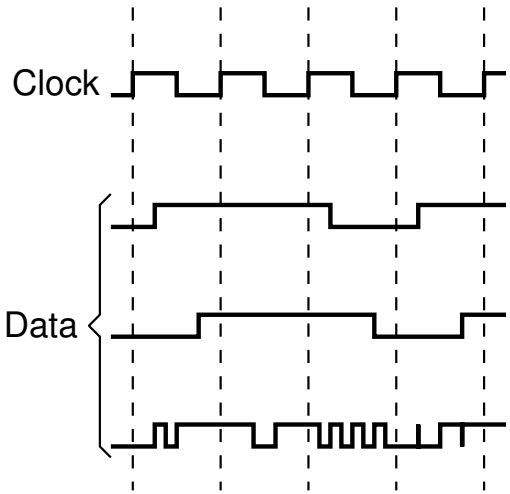


- All parts of the system share the same clock .
- The data may change between active clock transitions but must be stable by the time the next active transition occurs[1].

[1] for most systems the *active transition* is the rising edge of the clock

Synchronous Systems

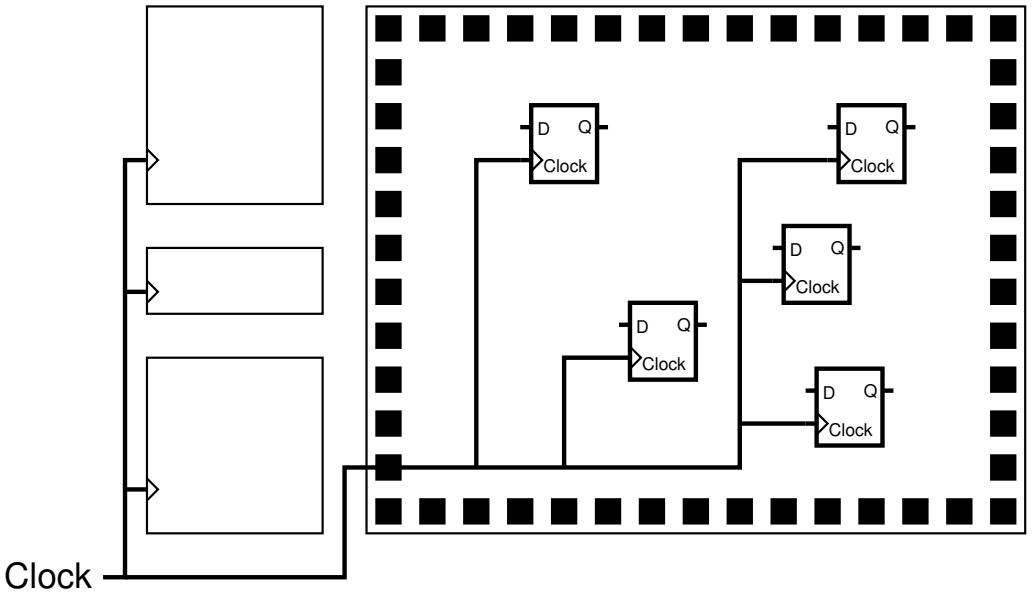


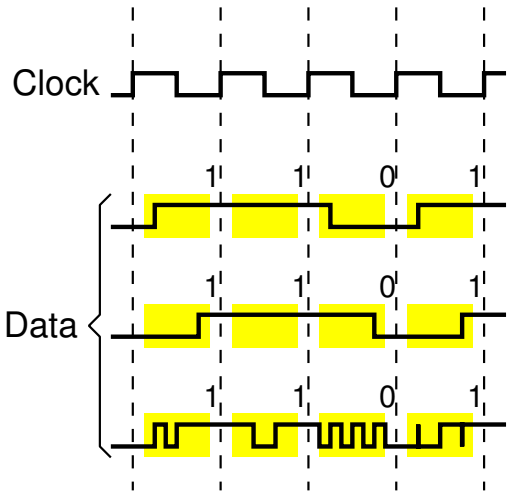


- All parts of the system share the same clock and the same clock edge sensitivity.
- The data may change between active clock transitions but must be stable by the time the next active transition occurs[1].

[1] for most systems the *active transition* is the rising edge of the clock

Synchronous Systems

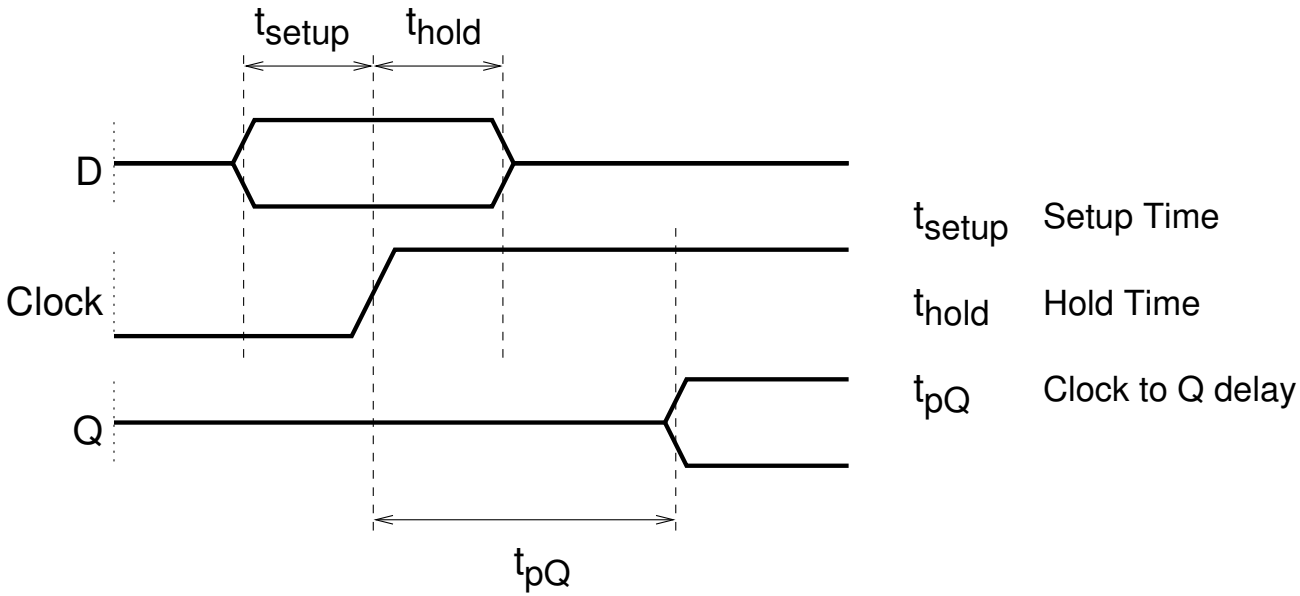




- All parts of the system share the same clock and the same clock edge sensitivity.
- The data may change between active clock transitions but must be stable by the time the next active transition occurs[1].

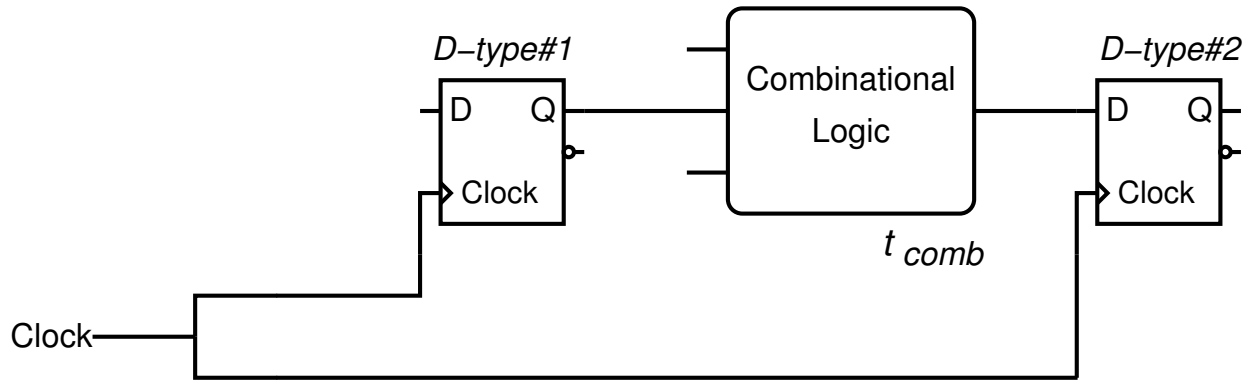
[1] for most systems the *active transition* is the rising edge of the clock

Synchronous Systems - D-type Timing



- Valid data should be present on D input for at least t_{setup} before the active clock edge and at least t_{hold} after the clock edge.
- The minimum D-type cycle time will be limited by the sum $t_{setup} + t_{pQ}$.

Synchronous Systems - Static Timing Analysis[2]



- To avoid a setup violation:

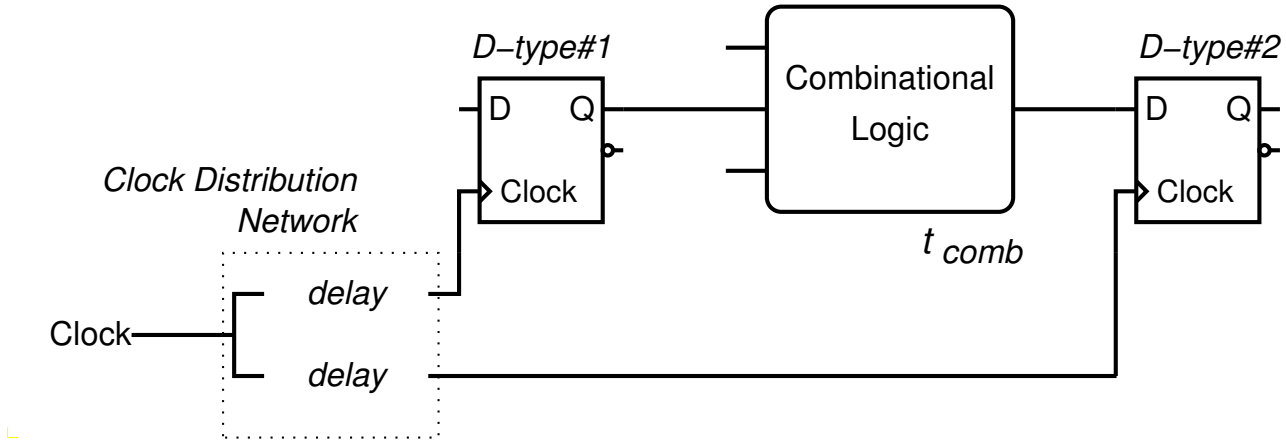
$$ClockPeriod > t_{pQ} + t_{comb} + t_{setup}$$

- To avoid a hold violation:

$$t_{pQ} + t_{comb} > t_{hold}$$

[2]with ideal clock

Synchronous Systems - Clock Skew



- Clock Distribution

  The process of distributing clocks from a central source gives rise to delays.

  Clock skew is the difference between the arrival times of the clock at different points in the circuit.

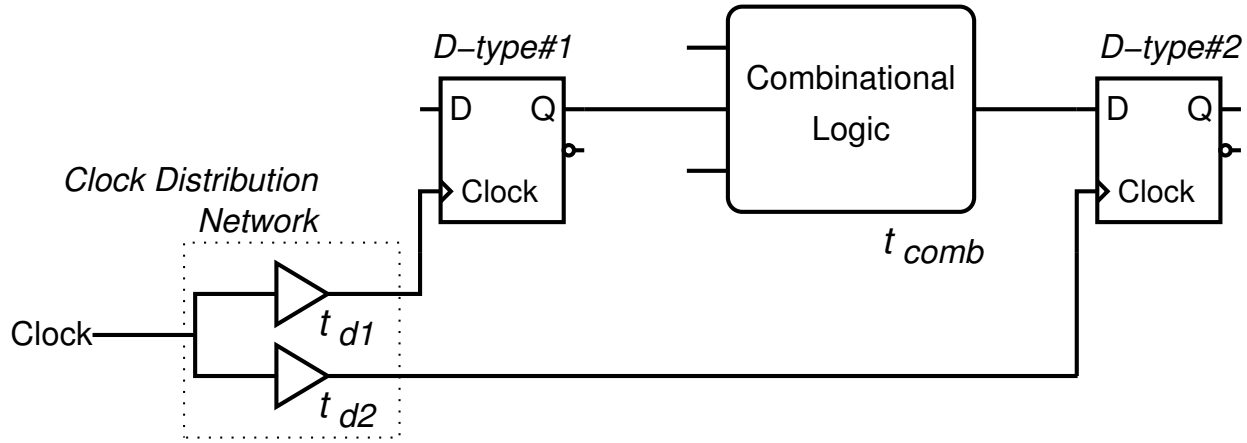
- Clock Skew may cause unexpected timing violations

  Hold: if *D-type #1* clocks first, D input of *D-type #2* may change too early[3]

  Setup: if *D-type #1* clocks second, D input of *D-type #2* may change too late

[3]most likely where combinational logic is minimal or absent

Synchronous Systems - Clock Skew



- Clock Distribution

  The process of distributing clocks from a central source gives rise to delays.

  Clock skew is the difference between the arrival times of the clock at different points in the circuit.

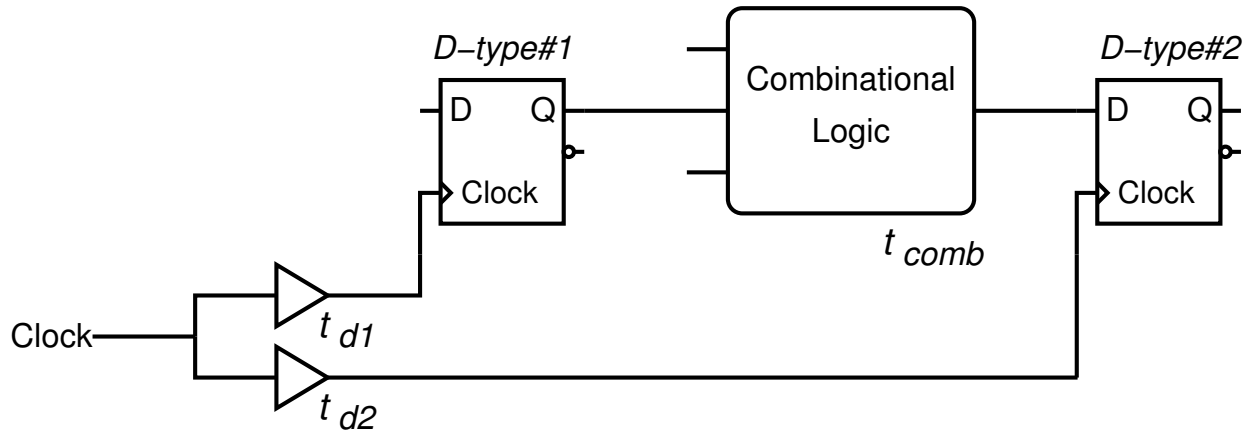
- Clock Skew may cause unexpected timing violations

  Hold: if *D-type #1* clocks first, D input of *D-type #2* may change too early[3]

  Setup: if *D-type #1* clocks second, D input of *D-type #2* may change too late

[3]most likely where combinational logic is minimal or absent

Synchronous Systems - Static Timing Analysis[4]



- To avoid a setup violation:

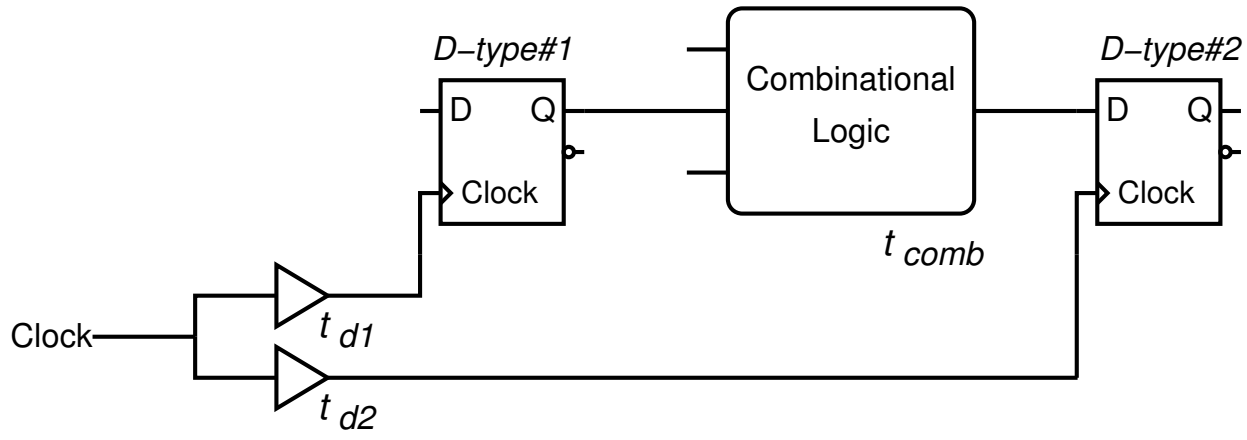
$$ClockPeriod + t_{d2} > t_{pQ} + t_{comb} + t_{setup} + t_{d1}$$

- To avoid a hold violation:

$$t_{pQ} + t_{comb} + t_{d1} > t_{hold} + t_{d2}$$

[4] with clock skew

Synchronous Systems - Static Timing Analysis[5]



- To avoid a setup violation:

$$ClockPeriod > t_{pQ} + t_{comb} + t_{setup} + (t_{d1} - t_{d2})$$

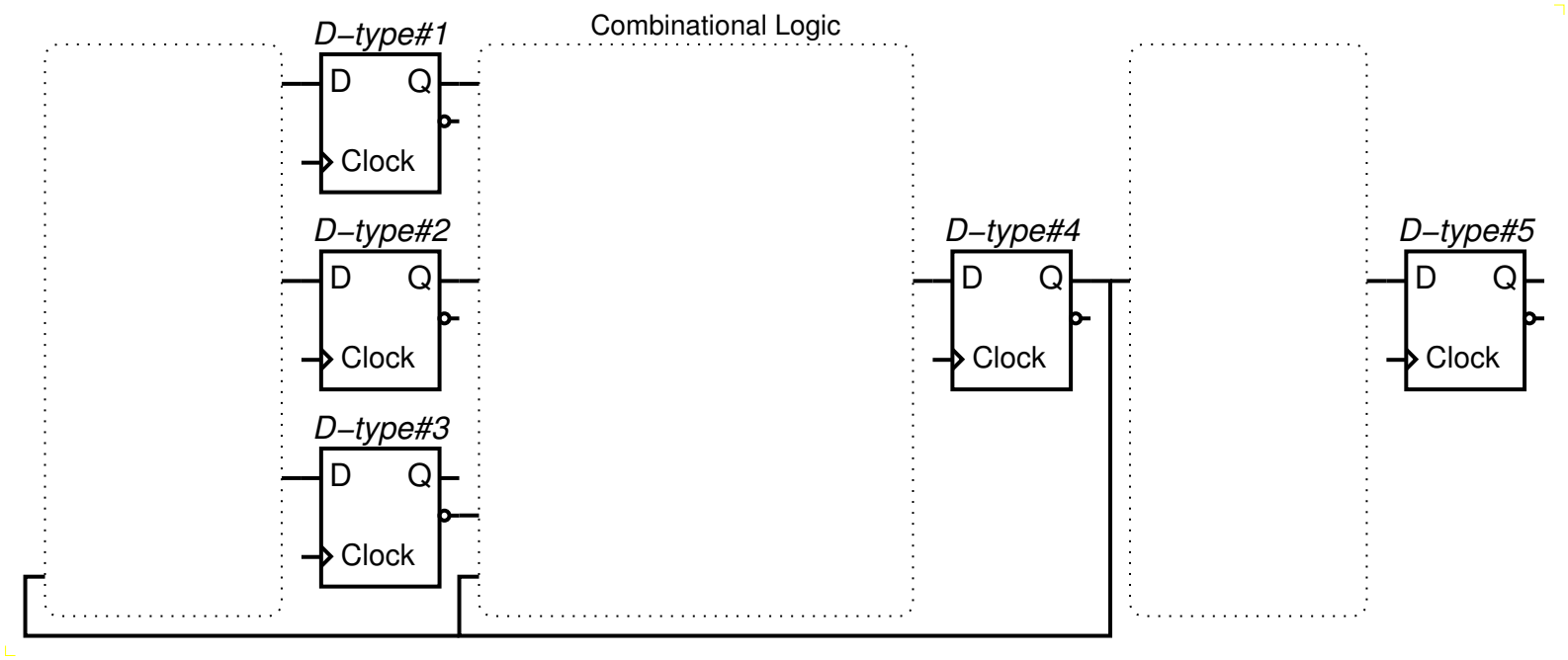
- To avoid a hold violation:

$$t_{pQ} + t_{comb} > t_{hold} + (t_{d2} - t_{d1})$$

[5] with clock skew

Synchronous Systems - Static Timing Analysis

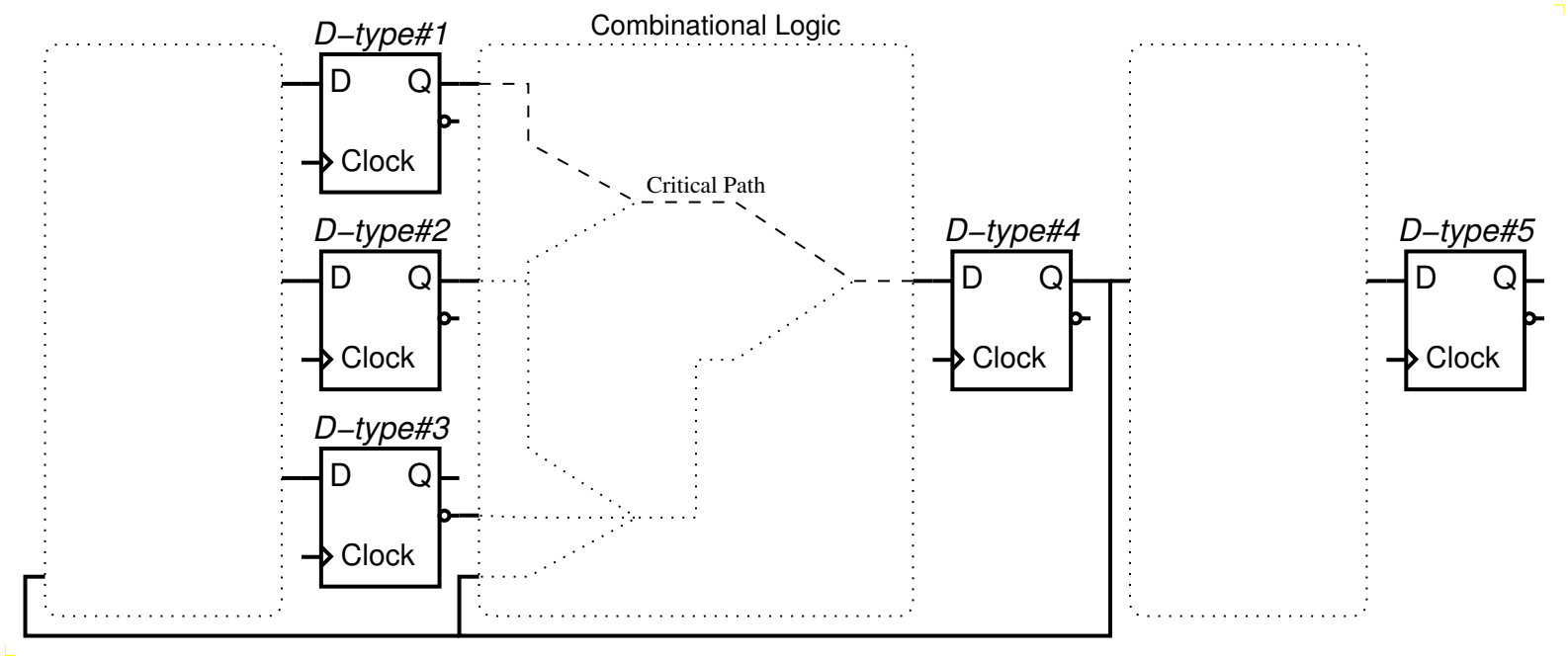
Critical Path



- To avoid a setup violation:

$$ClockPeriod > t_{pQ} + t_{critical_path} + t_{setup} + t_{skew}$$

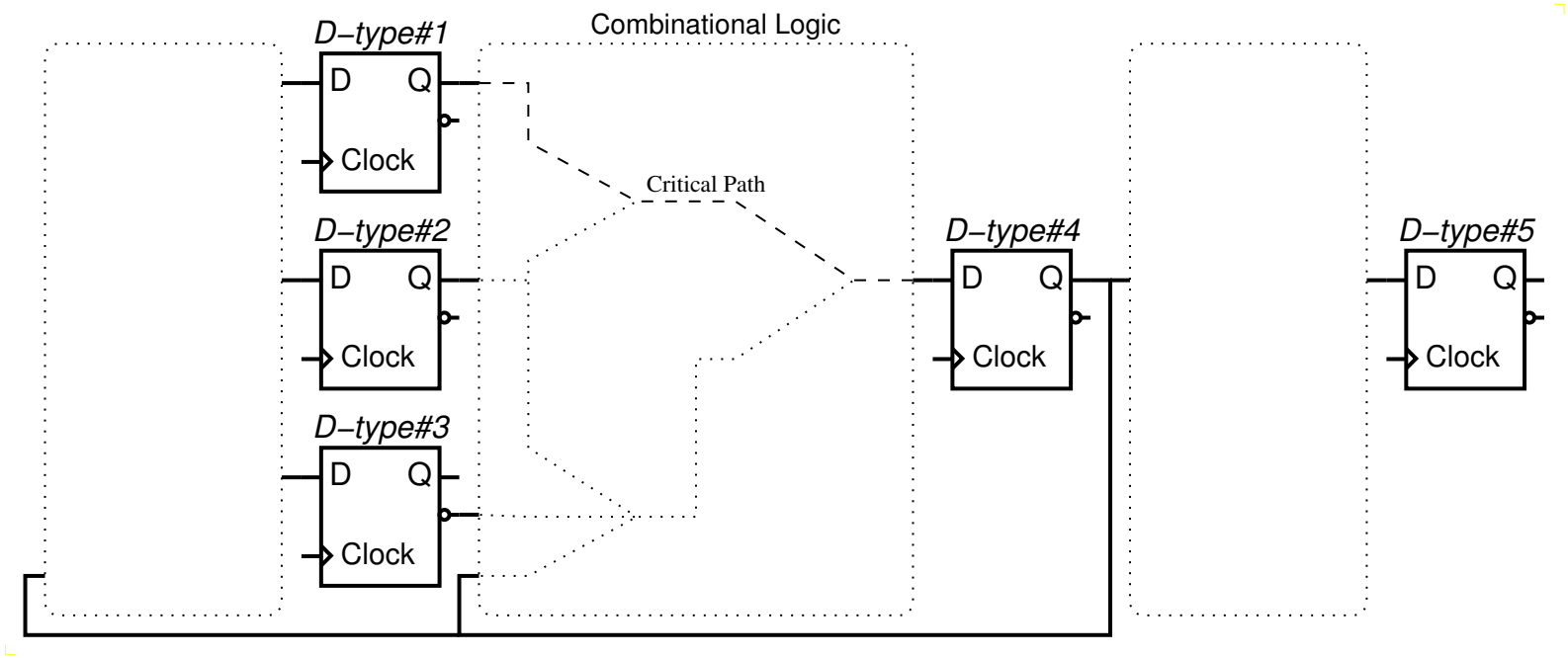
Critical Path



- To avoid a setup violation:

$$ClockPeriod > t_{pQ} + t_{critical_path} + t_{setup} + (t_{d1} - t_{d4})$$

Critical Path

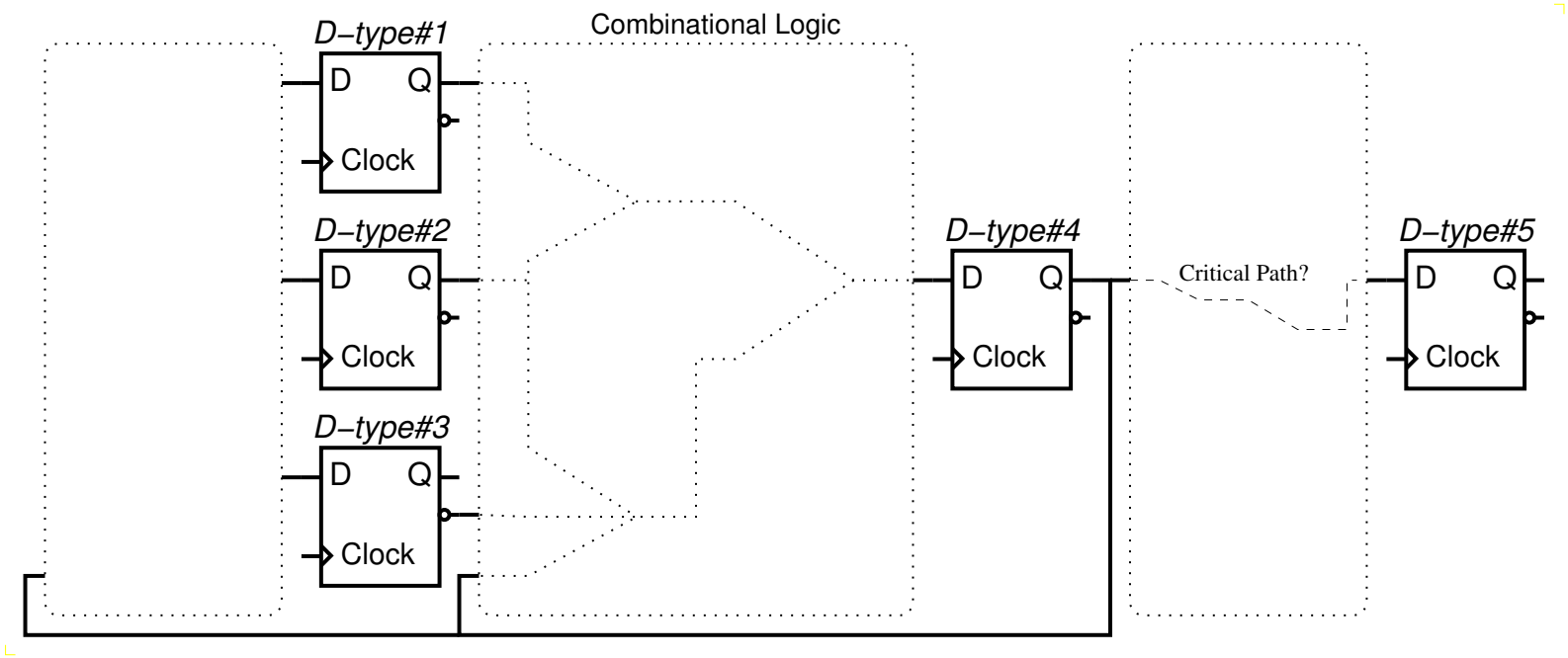


$$ClockPeriod > t_{pQ} + t_{critical_path} + t_{setup} + (t_{d1} - t_{d4})$$

- If we can control the skew (e.g. by increasing t_{d4}), we can ease the timing constraint.[5]

[5] this may result in the critical path moving to another part of the circuit

Critical Path

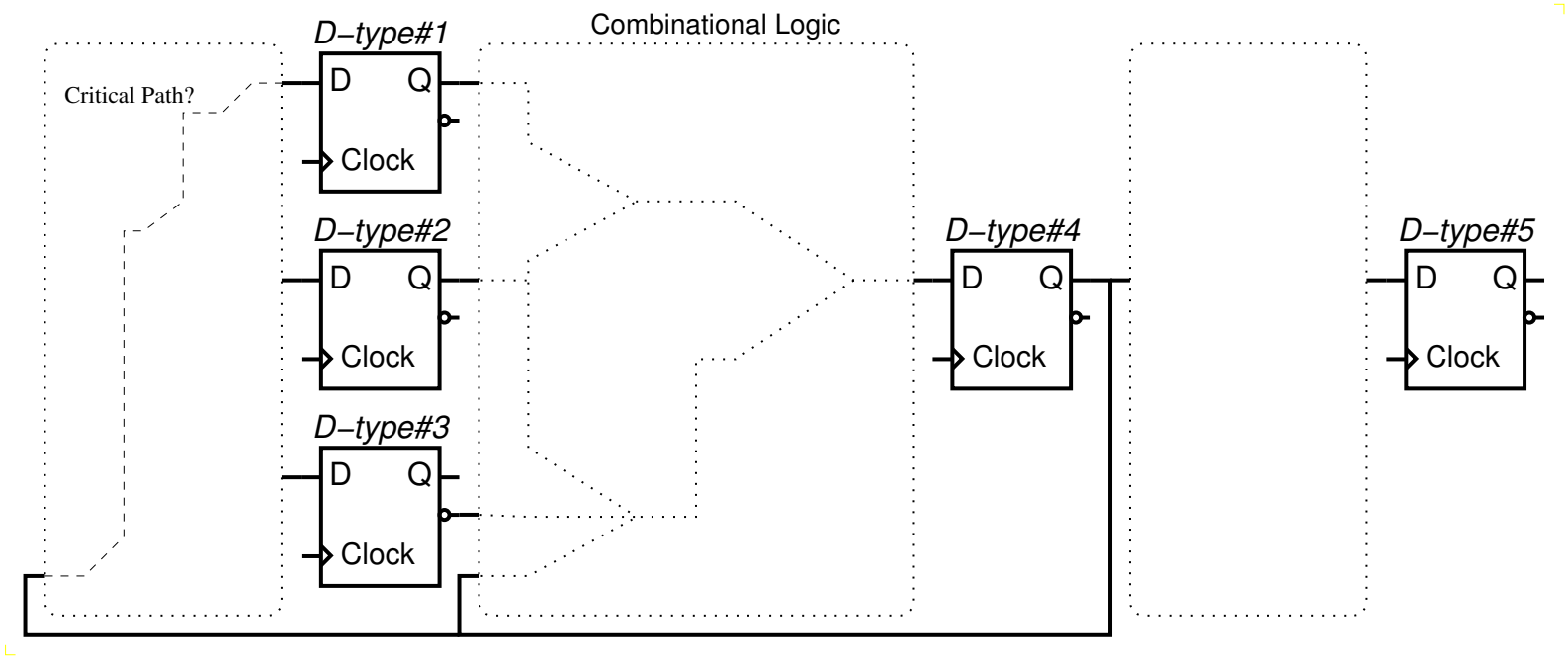


$$ClockPeriod > t_{pQ} + t_{critical_path} + t_{setup} + (t_{d4} - t_{d5})$$

- If we can control the skew (e.g. by increasing t_{d4}), we can ease the timing constraint.[5]

[5] this may result in the critical path moving to another part of the circuit

Critical Path

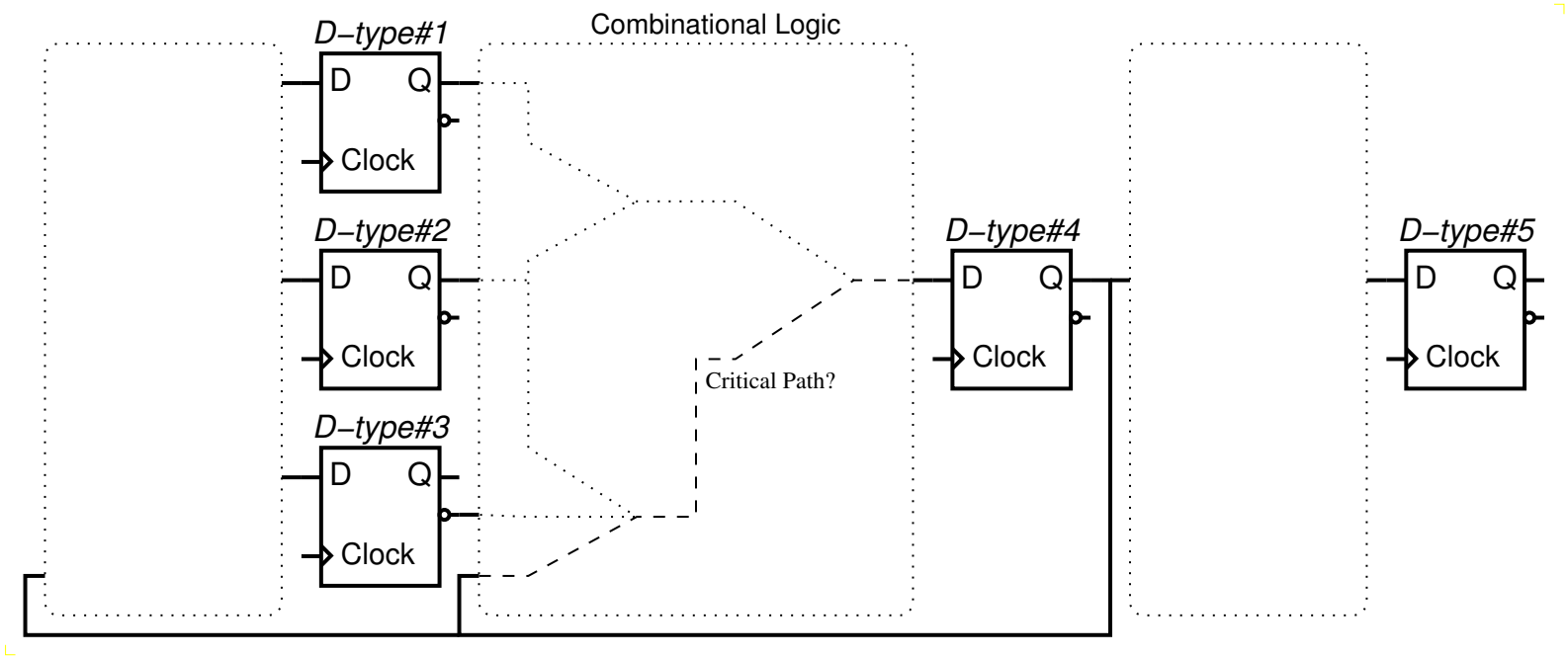


$$ClockPeriod > t_{pQ} + t_{critical_path} + t_{setup} + (t_{d4} - t_{d1})$$

- If we can control the skew (e.g. by increasing t_{d4}), we can ease the timing constraint.[5]

[5] this may result in the critical path moving to another part of the circuit

Critical Path

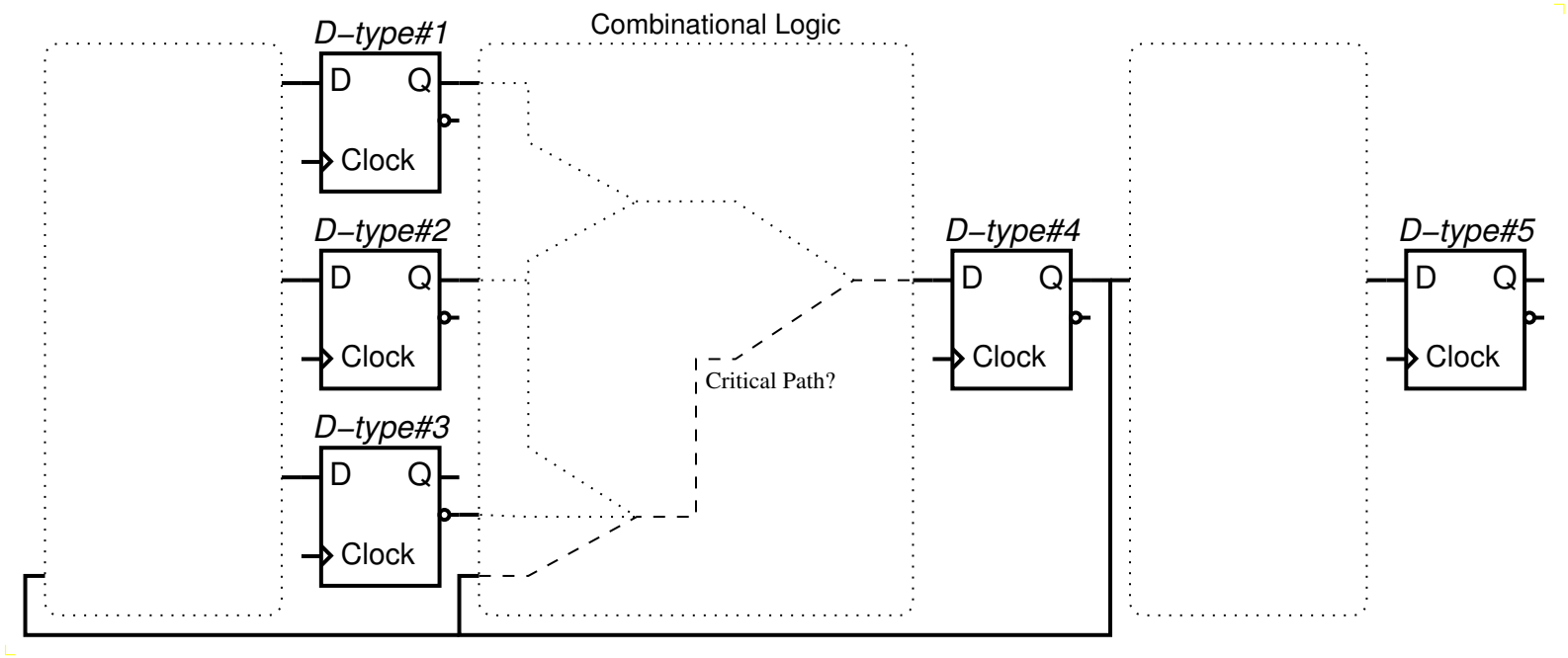


$$ClockPeriod > t_{pQ} + t_{critical_path} + t_{setup} + (t_{d4} - t_{d4})$$

- If we can control the skew (e.g. by increasing t_{d4}), we can ease the timing constraint.[5]

[5] this may result in the critical path moving to another part of the circuit

Critical Path

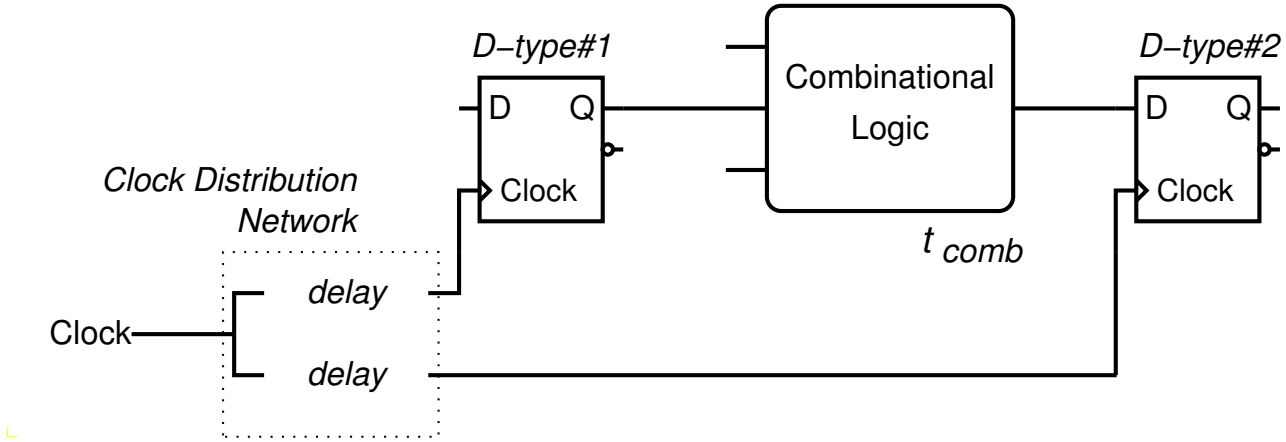


$$ClockPeriod > t_{pQ} + t_{critical_path} + t_{setup}$$

- If we can control the skew (e.g. by increasing t_{d4}), we can ease the timing constraint.[5]

[5] this may result in the critical path moving to another part of the circuit

Synchronous Systems - Jitter



- Jitter

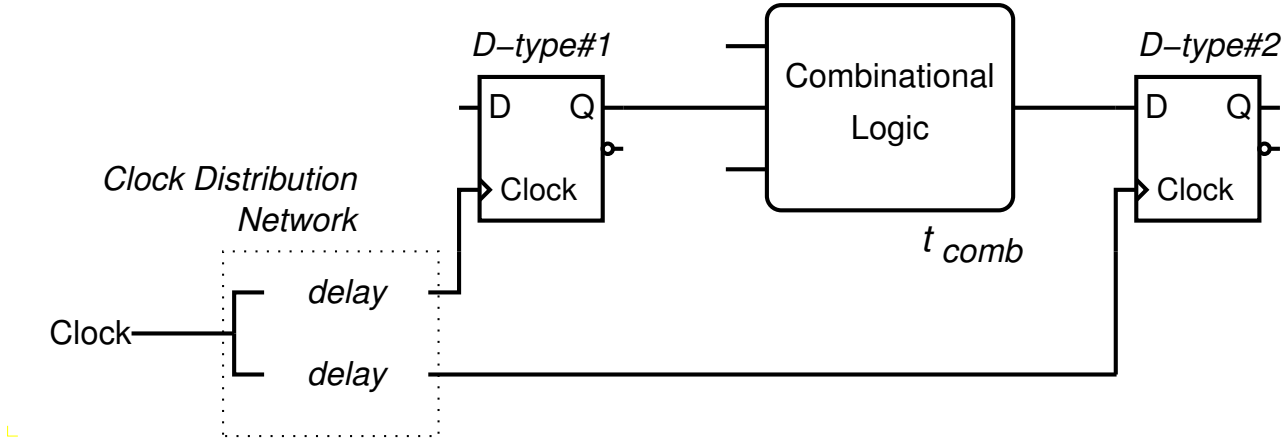
  Jitter is the cycle-by-cycle variation in the arrival time of the clock.

- Caused by
  - Variation in frequency/phase of clock source[6]
  - Power supply noise affecting clock distribution
  - Cross-talk affecting clock distribution

- Jitter may cause unexpected timing violations

[6] primary clock source or Phase-Locked Loop (PLL)

Synchronous Systems - Jitter



- To avoid a setup violation:

$$ClockPeriod > t_{pQ} + t_{comb} + t_{setup} + (t_{d1} - t_{d2}) + 2 \times t_{jitter}$$

- To avoid a hold violation:

$$t_{pQ} + t_{comb} > t_{hold} + (t_{d2} - t_{d1}) + 2 \times t_{jitter}$$