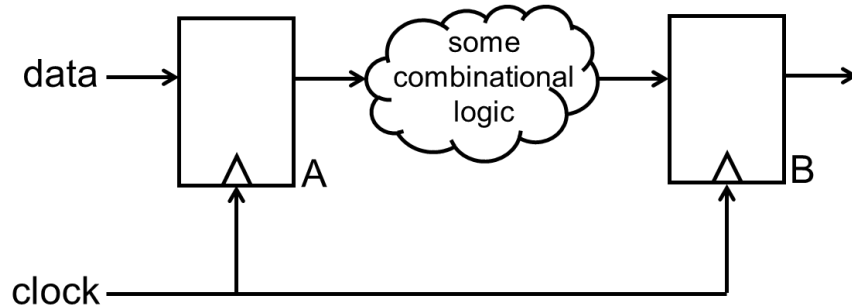# Supplementary Notes   Timing Analysis

We will reference the following diagram to discuss common timing constraints in digital circuits:

1. Setup Time Constraint

Setup time = minimum amount of time the data signal should be held steady **before** the clock edge so that the data can be reliably sampled.

Think of the setup time constraint as a race between the data signal and the clock. These two race between one flip-flop to the other – such as between flip-flops A and B in the diagram above.

	Full Name	Think of it as …
$t_{ffpd(max)}$	Flip-flop (maximum) propagation delay	Time taken for signal to leave flip-flop A
$t_{comb(max)}$	(Maximum) propagation delay of combinational logic	Time taken for signal to pass through combinational logic block
t_{setup}	Setup time	Time that the data has to be held steady before the clock reaches flip-flop B
t_{clk}	Clock period	Time taken for clock to get from flip-flop A to flip-flop B
t_{skew}	Clock skew	Head start (positive or negative) that the clock has in this race

Satisfying the setup time constraint = preparing the data (using combinational logic) in time so that the prepared data is steady for the specified setup time before the clock gets to the receiving flip-flop.

Hence, satisfying the setup time constraint means that the data signal has to win the race between two flip-flops by t_{setup} or more **and** the data signal has to win even in the **worst-case** scenario.

With setup time constraint, the worst case means that the data signal is slowed down (maximum propagation delays) and the clock signal is sped up (maximum skew).

1.1. Without clock skew
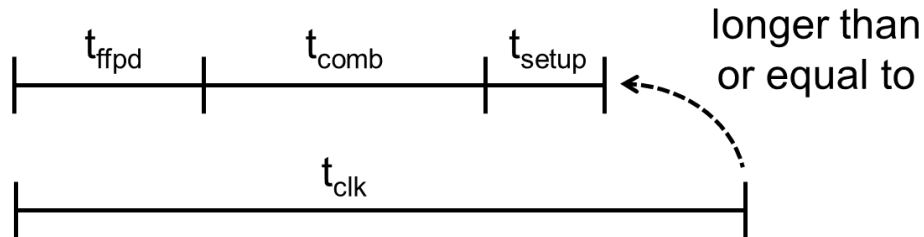
As shown in the above diagram, t_{clk} should be longer than or equal to the sum of t_{ffpd}, t_{comb}, and t_{setup}. In this sense, the data signal would have "won" the race to flip-flop B, since it got there before t_{clk}.

The data signal must win even if the propagation delays are at their maximums – the data signal must win even in the worst case when everything is against the data signal getting to flip-flop B first.

The following inequality should hold for the setup time constraint to be met:

$$t_{ffpd(max)} + t_{comb(max)} + t_{setup} \leq t_{clk}$$

1.2. With clock skew

Clock skew could arise because of various reasons, usually to do with imperfections of the real world. Some examples:

- Clock distribution wire length
- Temperature variations
- Material imperfections
- Direction of clock distribution (whether in the direction of data flow or opposite in direction)

Hence, the clock could reach flip-flops at different times.

In the following diagrams, t_{skew} is annotated as a magnitude (no positive or negative signs). Zero is labeled on the time axis to show the sign of the skew.

Negative Skew: Sending flip-flop (FF A) receives clock after receiving flip-flop (FF B)

If the clock reaches FF B before FF A, then the clock connected to FF B is given a head start, and the data now has a tighter window to reach FF B if the setup time constraint is still to be satisfied.
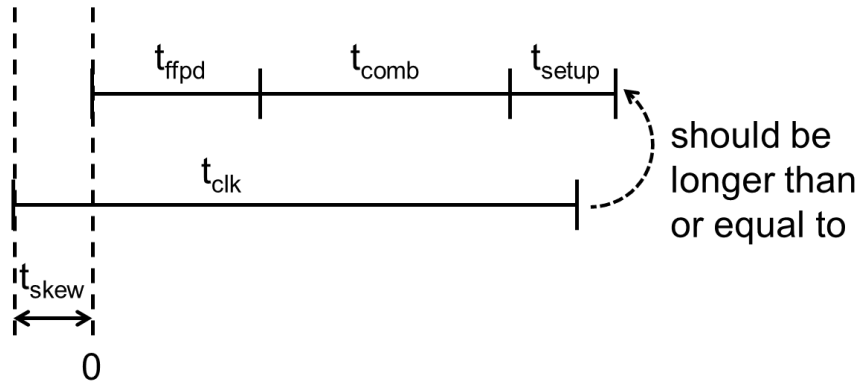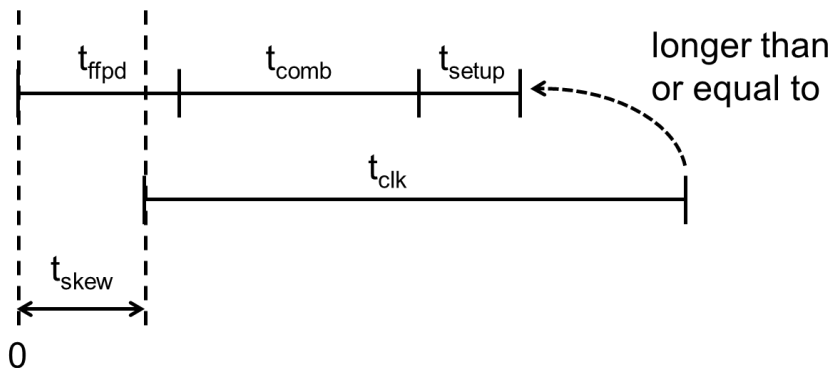

The following inequality should hold for the setup time constraint to be met:

$$t_{ffpd(max)} + t_{comb(max)} + t_{setup} \leq t_{clk} - t_{skew}$$

The minimum required t_{clk} is increased for this clock skew, which means that the circuit can only operate at a longer clock period (i.e., lower frequency). Since this clock skew requires a lower clock frequency, we say that this clock skew is a **harmful** clock skew.

Positive Skew: Receiving flip-flop (FF A) receives clock after sending flip-flop (FF B)

When the clock is skewed such that the clock signal reaches FF B after FF A, it means that the clock cycle is "stretched", as the clock connected to FF B starts later by t_{skew}.

Given these conditions, the data signal still has to win the race, but because of clock skew, it does not have to run as quickly.

The following inequality should hold for the setup time constraint to be met:

$$t_{ffpd(max)} + t_{comb(max)} + t_{setup} \leq t_{clk} + t_{skew}$$

If the clock signal reaches flip-flop B after flip-flop A, the minimum required t_{clk} is reduced and hence the circuit as a whole can operate at a shorter clock period (i.e., higher frequency). This allows a higher clock frequency; we say that this clock skew is a **beneficial** clock skew.

2. Hold Time Constraint

Hold time = minimum amount of time that the data signal must be held steady **after** the clock edge so that the data can be reliably sampled.

You can think of this as passing the baton between two members of a relay team. The goal is for the data signal to be *slower* than a certain time known as the hold time, so that the member holding the baton does not drop it before the next member of the team (flip-flop B) can firmly grab on to the baton (data is not contaminated).

In determining whether the hold time constraint is satisfied, we are concerned with the minimum propagation delays (= contamination delay).

	Full Name	Think of it as…
$t_{ffpd(min)}$	Flip-flop (minimum) propagation delay = Flip-flop contamination delay	Time taken for signal at Q of flip-flop A to be contaminated if the signal at D gets contaminated
$t_{comb(min)}$	(Minimum) propagation delay of combinational logic = Contamination delay of combinational logic	Time taken for signal at the end of the combinational logic block to be contaminated if the signal at the start of the block gets contaminated
t_{hold}	Hold time	Time that the data has to be held steady after the clock reaches flip-flop B
t_{skew}	Clock skew	Deadline extension (positive or negative) that is given to pass the baton

Satisfying the hold time constraint = data must stay valid for the specified hold time after the clock edge

Hence, satisfying the hold time constraint means that the data signal can be contaminated only after the required hold time **and** this has to happen even in the **worst-case** scenario.

With hold time constraints, the worst case means that the data signal is sped up (minimum contamination delays) and the required deadline is later (maximum skew).

2.1. Without clock skew
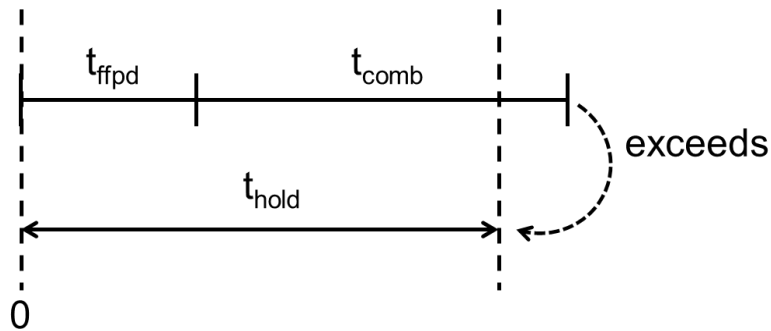
As shown in the above diagram, the sum of t_{ffpd} and t_{comb} has to be greater than t_{hold}. Otherwise, it would be as if the earlier member of the relay team (flip-flop A) did not give the later member (flip-flop B) enough time to grab on, and the baton was dropped.

The data signal must succeed in passing the baton even in the worst case where the hold time is lengthened by clock skew and the contamination delays are particularly short.

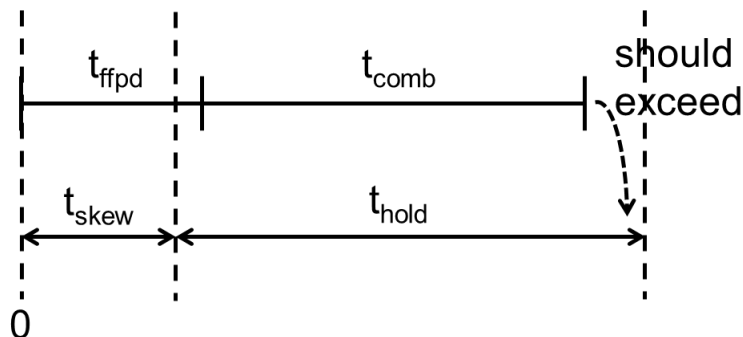The following inequality should hold for the hold constraint to be met:

$$t_{ffpd(min)} + t_{comb(min)} \geq t_{hold}$$

2.2. With clock skew

Again, t_{skew} is annotated as a magnitude (no positive or negative signs). Zero is labeled on the time axis to show the sign of the skew.

Positive Skew: Receiving flip-flop (FF B) receives clock after sending flip-flop (FF A)

When the clock reaches FF B after FF A, this means that the time required to pass the baton is lengthened (namely, wider hold time window), and the data signal must be held steady for a longer time. Hence this is a **harmful** skew for meeting hold time.
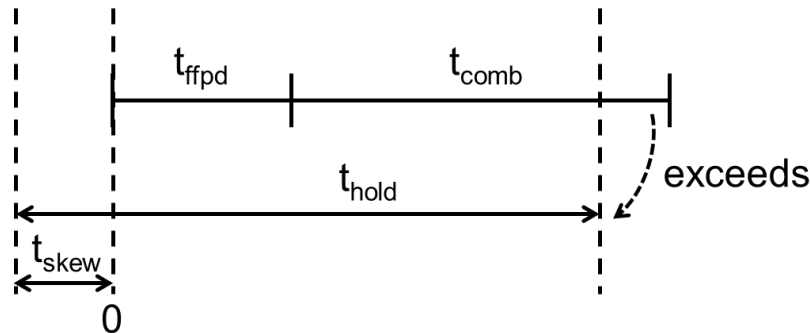
The following inequality should hold for the setup time constraint to be met:

$$t_{ffpd(min)} + t_{comb(min)} \geq t_{hold} + t_{skew}$$

Negative Skew: Sending flip-flop (FF A) receives clock after receiving flip-flop (FF B)

If the clock reaches FF A after FF B, then the time required to pass the baton is shortened (namely, narrower hold time window), since the t_{ffpd} and t_{comb} start counting only when the clock reaches FF A. Hence this is a **beneficial** skew for meeting hold time.

The following inequality should hold for the hold time constraint to be met:

$$t_{ffpd(min)} + t_{comb(min)} \geq t_{hold} - t_{skew}$$

Appendix

To learn more about why setup and hold time constraints matter for flip-flops, check out this link: https://www.edn.com/understanding-the-basics-of-setup-and-hold-time/