Static Timing Analysis

This lecture describes how static timing analysis is used to ensure that timing constraints for a digital design are met. After this lecture you should be able to be able to: identify features and specifications on a timing diagram, identify a specification as a requirement or guaranteed response, apply the terms defined in this lecture, and do calculations involving clock rate, propagation delays and setup/hold time requirements.

Introduction

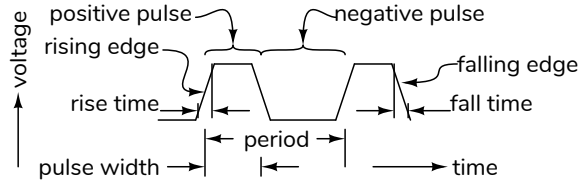
Logic *devices* have specifications that must be met in order for the device to operate properly. These are called "requirements." Some requirements are related to voltages (e.g. supply voltages or logic levels). Some, called timing requirements, are requirements related to time.

A *circuit* using these logic devices may also have timing requirements. A typical requirement is that the circuit must operate at a specific clock frequency. Meeting these requirements is often as difficult as ensuring that a design is logically correct.

A designer must specify the timing requirements for the circuit such as the clock rate (or period) and external circuit delays. This allows the design software to check whether every logic device's timing requirements (e.g. flip-flop setup and hold times) will be met. If they are not met, then the designer must change the design or relax the requirements in order to meet the device timing requirements.

The performance of ICs varies from device to device and with changes in temperature and voltage. Building one example of a circuit that works properly is not enough to ensure that a design will work reliably. This is because the same design may fail on a different device, at a different temperature, or with a different supply voltage.

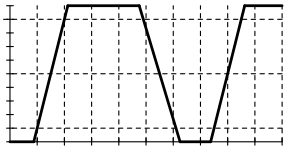
Digital Waveforms



A transition from low to high is called a rising edge. The time it takes is called the rise time. A transition from high to low is called a falling edge. The time it takes is called the fall time. Two adjacent edges define a pulse, which can be negative or positive. The time between these is called the pulse width. The duty cycle is the active pulse width divided by the period.

Rise and fall times are typically measured between 10% and 90% of the swing. Other measurements are typically made between 50% levels. But there are exceptions.

A signal consisting of periodic pulses is a clock. The inverse of the clock period is the clock frequency.



Exercise 1: The diagram above shows an oscilloscope screen capture that includes one period of an *active-low* digital waveform. The scale on the horizontal axis is 20 ns per division. What are: the rise time, period, positive pulse width and duty cycle?

Timing Specifications

Timing specifications can be:

requirements: the manufacturer requires that these specifications be met for a device to operate properly, or

guaranteed responses: the manufacturer guarantees that these specifications will be met.

Since the device manufacturer cannot control the timing on inputs but can guarantee the timing of outputs, a simple rule to distinguish requirements from responses is:

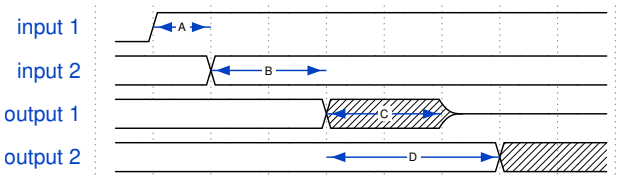
- requirements are times measured *ending on a transition on an input*.

- guaranteed responses are times measured *ending on a transition on an output*.

Timing Diagrams

Timing diagrams show the relationship between transitions on inputs and outputs. However, they are not drawn to scale and transitions may not always happen in the order shown.

Other conventions used in timing diagrams include: both high and low levels are shown when the specification applies to both, shading between two levels indicates that the value is allowed to change during this time, a line half-way between the two logic levels indicates that the signal is in high-impedance ("tri-state") state and arrows drawn between transitions show that one signal transition causes or affects another.

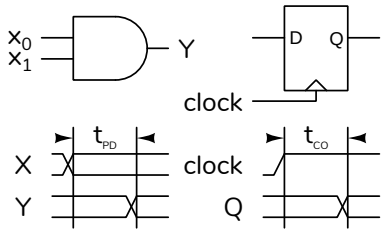


Exercise 2: Label the specifications A through D as requirements or guaranteed responses. Which specifications are measured to a signal being in a high-impedance state? Which are measured from a rising edge only? From either?

Propagation Delays

The most important specification for combinational logic is *propagation delay*, t_{PD}. This is the delay from a change at an input to the corresponding change at an output.

The clock-to-output delay of a flip-flop, t_{CO} is also a type of propagation delay. It is the delay from the rising edge at a flip-flop clock input to the change at the Q output.

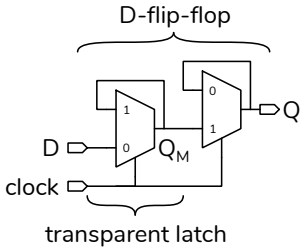


These main contribution to these delays is the time required to charge the parasitic capacitances of transistors and the connections between them.

Exercise 3: Is t_{PD} a requirement or a guaranteed response?

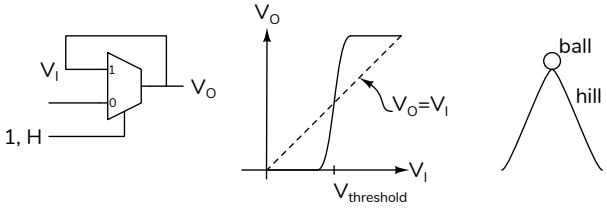
Metastability, Setup and Hold Times

Consider the following implementation of an edge-triggered D flip-flop:

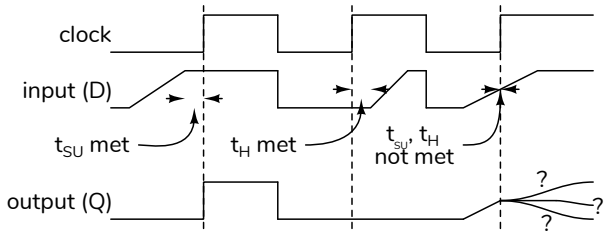


When the clock input is low, the output of the first multiplexer follows the input – the latch is "transparent". When the clock input is high, the output level is fed back to the input and held at that level. The second multiplexer is transparent when the clock is high and holds the value captured by the first multiplexer when the clock is low. This results in the output only changing on the rising edge of the clock.

However, if the first multiplexer's output (Q_M) is at the logic input threshold voltage when the clock changes from 0 to 1 – the rising edge – the value fed back would be indeterminate.



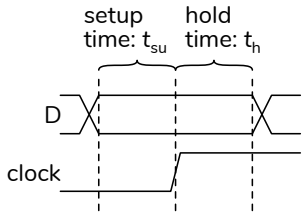
The multiplexer output (Q_M) could remain at at the threshold level for a long time – longer than the t_{CO} specification. This behaviour is called "metastability"[1] and can result in incorrect operation of the circuit.



To avoid metastability we must ensure the voltage at the latch input is at valid level long enough to drive the latch output to a valid voltage level before the rising edge of the clock. The time required for this is called the "setup" time, t_{SU}.

The input level must also be held at the correct level until the transparent latch has finished switching to the feedback mode. This is typically a much shorter time – typically zero – and is called the "hold" time, t_H.

D flip-flops thus have two important timing requirements:



setup time the D input must be at a valid logic level for at least t_{SU} before the rising edge of the clock

hold time the D input must remain a valid logic level for at least t_H after the rising edge of the clock

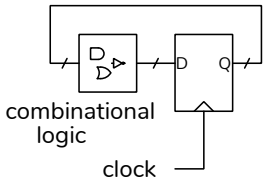
Exercise 4: Is t_{SU} a requirement or a guaranteed response? How about t_H?

[1]The output is "meta" stable because it appears to be stable at a level that is not one of the true stable states (H or L).

Synchronous Design and Static Timing Analysis

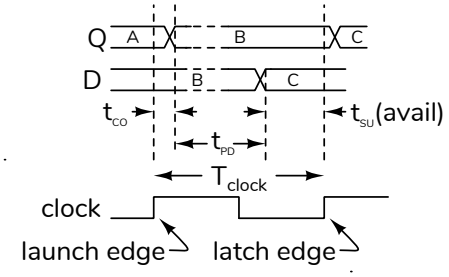
A circuit that uses only edge-triggered flip-flops and a single clock is called a "synchronous" circuit. Synchronous circuits can be easily designed to avoid metastable behaviour. For this reason they are almost always used.

We can imagine replacing all the flip-flops in a circuit with one – potentially very wide – register. The circuit now consists of only this one register and combinational logic between this register's Q and D (ignoring inputs and outputs for now). We can draw this as:



On the rising edge of the clock the register output changes, and it takes t_{PD}(max) for these changes to propagate through all of the logic and for the register input to become stable.

The timing diagram below shows the relationship between two adjacent clock edges, the output (Q) and input (D) of the register:



Q becomes valid t_{CO} after the first ("launching") rising clock edge. After t_{PD}, the maximum propagation delay through the combinational logic, D will have the correct and valid logic level.

In most cases the timing requirement that is most difficult to meet is the minimum setup time.

From the timing diagram above we can write an expression for the minimum available setup time before the next ("latching") rising clock edge. :

$$t_{SU}\text{ (avail)} = T_{clock} - t_{CO}\text{ (max)} - t_{PD}\text{ (max)}$$

Exercise 5: Which of the specifications in the formula above decrease the available setup time as they increase? Which increase it?

The amount by which the minimum available setup time exceeds the minimum required setup time is known as the "slack":

$$\text{slack} = t_{SU}\text{ (available) - }t_{SU}\text{ (required)}$$

T_{clock} is a design choice. t_{CO} (max) is the maximum delay specified by the manufacturer. Each logic function (multiplexer, adder, gate, etc.) between the output and input of a flip-flop increases t_{PD} for that path.

Static timing analysis is a step in logic synthesis that computes the maximum t_{PD} for any possible path between any register output and any register input. This t_{PD}(max) is then used to compute the slack.

If the slack is positive then the available setup time exceeds the required setup time and the t_{SU} requirement is met, otherwise it is not and the circuit may not operate correctly due to metastable behaviour: excessively long clock-to-output delays.

Exercise 6: For a particular circuit f_{clock} is 50 MHz, t_{CO} is 2 ns (maximum), the worst-case (maximum) t_{PD} in a circuit is 15 ns and the minimum setup time requirement is 5 ns. What is the setup time slack? Will this circuit operate reliably? If not, what it the maximum clock frequency at which it will?

Exercise 7: What is the maximum clock frequency for a counter using flip-flops with 200 ps setup times, 50 ps clock-to-output delays and adder logic that has a 250 ps propagation delay?

Note that the clock signal itself may have a propagation delay and may arrive at different flip-flops at different times. This is known as "clock skew" and must also be taken into account in the analysis.