Timing Analysis

Timing analysis is the process of verifying that the timing requirements of each chip in a circuit are met. Unless all timing requirements are met the circuit may fail to operate properly under some conditions.
After this lecture you should be able to draw a timing diagram for a simple circuit, derive expressions for a chip's timing requirements from the timing diagram and compute the margin for each requirement based on clock periods and the guaranteed responses of the other components.

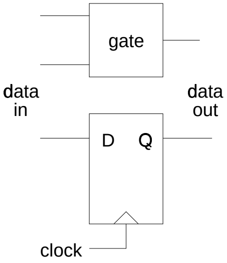
Timing Specifications

A chip's timing specifications are of two types: (1) timing requirements and (2) guaranteed responses. All timing specifications are measured between transitions from low to high (rising edge) or high to low (falling edge) – of a chip's inputs and outputs.

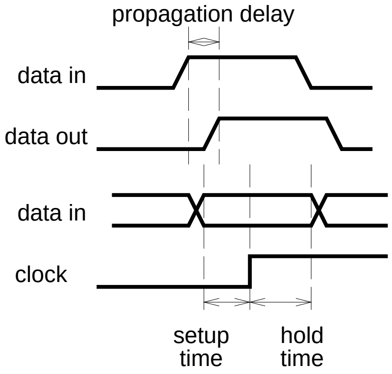
Guaranteed responses are delays between a transition on a an input (or output) signal and the transition to the correct level on an *output* signal. A chip's manufacturer guarantees this specification will always be true if the chip is operated within it's recommended limits. A typical guaranteed response is a propagation delay.

Timing *requirements* are the time relationships between a transition on an output (or input) signal and the transition to the correct level on an *input* signal. A chip's manufacturer guarantee the correct logical operation of the chip if all of the requirements are met. Typical examples of timing requirements are setup and hold times.

The diagram below shows the simplest examples of the two types of circuits: a logic gate (an example of a combinational circuit) and a D flip-flop (an example of a sequential circuit):



and the diagram below shows the three most common timing specifications:



The most common guaranteed response is the *propagation delay* which is the maximum delay between a change in the input and the correct value appearing at the output. A similar guaranteed response during a memory device read cycle is the *access time*. This is typically measured from the address, chip-select, or output-enable signals changing to when the data outputs become valid.

The most common timing requirements are the *setup time* and *hold time* which are the minimum durations that the data input to a flip-flop has to be at the desired value before and after the relevant clock edge. Setup and hold times also apply to memory device write cycles. These are typically measured between the address, data, or control signals changing to the edge of the write strobe that ends the write cycle.

During read cycles the data output by a memory device is loaded into a CPU register. Thus the CPU read cycle timing specifications usually include requirements similar to setup and hold times. Similarly, during write cycles the data output by the CPU is loaded into a memory device. Thus CPU write cycle timing specifications usually include guaranteed responses similar to propagation delays. Often the CPU manufacturer will quote timing requirements relative to clock edges rather than to read or write strobe edges.

In addition to the three fundamental specifications many chips may either require or guarantee a minimum/maximum *pulse width*s on certain signals and/or minimum/maximum *cycle time*s (waveform period) or frequencies.

Exercise: Which of the three basic specifications (delay, setup and hold times) would apply to a multiplexer? To a RAM chip? To a ROM?

Timing Analysis

Timing analysis should be part of every digital system design. After a preliminary circuit design the designer must verify that all of the timing requirements for each device will be met. It is not sufficient to build a prototype and demonstrate that it works properly since the actual timing characteristics will vary from chip to chip and as a function of temperature and supply voltage. If a timing analysis is not done before a design is put into production the consequences could be serious.

A timing analysis is most conveniently summarized in the form of a table. The first step in a timing analysis is to consult the data sheets for all of the devices in the circuit and prepare one table for each chip and for each possible type of cycle (read, write, etc). Each table line should list one timing *requirement* and it's minimum or maximum value.

Next, draw timing diagrams that show the waveforms of the relevant signals with labels indicating the timing specifications. This may include signals generated by clocks, by the microprocessor, by memories and by interface circuits such as address decoders and buffers. Often there will be several timing diagrams for different parts of the circuit or for different sequences of signals (e.g. different bus cycles). The timing diagrams are then used to derive expressions for each timing requirement in terms of the guaranteed timing responses of the other devices.

The expressions are derived by expressing each requirement in terms of variables representing clock periods and other chips' guaranteed responses. These expressions are then entered into the table.

When values are substituted for the variables in the equations a minimum (or possibly maximum) value is obtained for that requirement in that specific circuit. The difference between the computed requirement and the manufacturer's specified requirement is called the *margin*. For example, if a manufacturer specifies that a certain flip-flop requires a 10 ns setup time and in a particular circuit the setup time is guaranteed to be at least 50 ns then the margin is 40 ns.

On the other hand, if any of the margins are negative then the chip's timing requirements are not met and the design must be changed. Typical changes include:

- adding CPU wait states
- latching signals to extend them
- using redundant logic gates to add small delays (poor practice)

Timing Diagrams

Timing diagrams help to clarify the meanings of timing specifications by labeling the times between signal transitions ("edges") using symbols from tables of timing specifications.

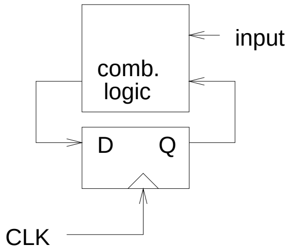
Some conventions used in timing diagrams are:

- high and low levels shown at the same time indicate the signal is not changing but can have either value (e.g. a data signal)
- 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
- arrows drawn between transitions on different signals show that one signal transition causes or affects another
- sloped transitions between levels allow references to the signal reaching a low (V_{OL}/V_{IL}) or high (V_{OH}/V_{IH}) value

It is important to understand that timing diagrams are *not* drawn to scale. This allows chips with different specifications to share the same timing diagram, allows small delays to be shown more clearly and also allows the same label on the diagram to refer to both maximum and minimum values. You can't even rely on the timing diagram to show the order in which signal transitions will happen.

Example

As a simple but complete example, consider a simple state machine where a combinational circuit computes the next state based on the current state and the input:



Exercise: Draw separate timing diagrams for the flip-flop and the combinational circuit. Assume the flip-flops require a minimum setup time, t_s, of 20ns and a minimum hold time, t_h of 0 ns. Assume the maximum clock-to-output propagation delay for the flip-flop is $t_{CO} = 5$ ns (again, with no minimum). Assume that the maximum propagation delay through the combinational logic circuit is guaranteed to be a maximum of $t_{PD} = 20$ ns, and there is no minimum for t_{PD}. Label the timing diagrams with each of these specifications.

Exercise: Draw a timing diagram for the complete circuit. It should include the clock CLK, the flip-flop's output, Q, and its input, D. Indicate cause–effect relationships between the signal edges using arrows.

Derive expressions for each timing requirement in terms of the clock period and guaranteed timing specifications for a clock frequency of 10 MHz. Substitute the actual values and compute the remaining margin. Will this circuit operate properly as far as timing is concerned? What if the hold time requirement was 5 ns?