数字集成电路静态时序分析基础

邸志雄 博士, zxdi@home.swjtu.edu.cn

西南交通大学信息科学与技术学院

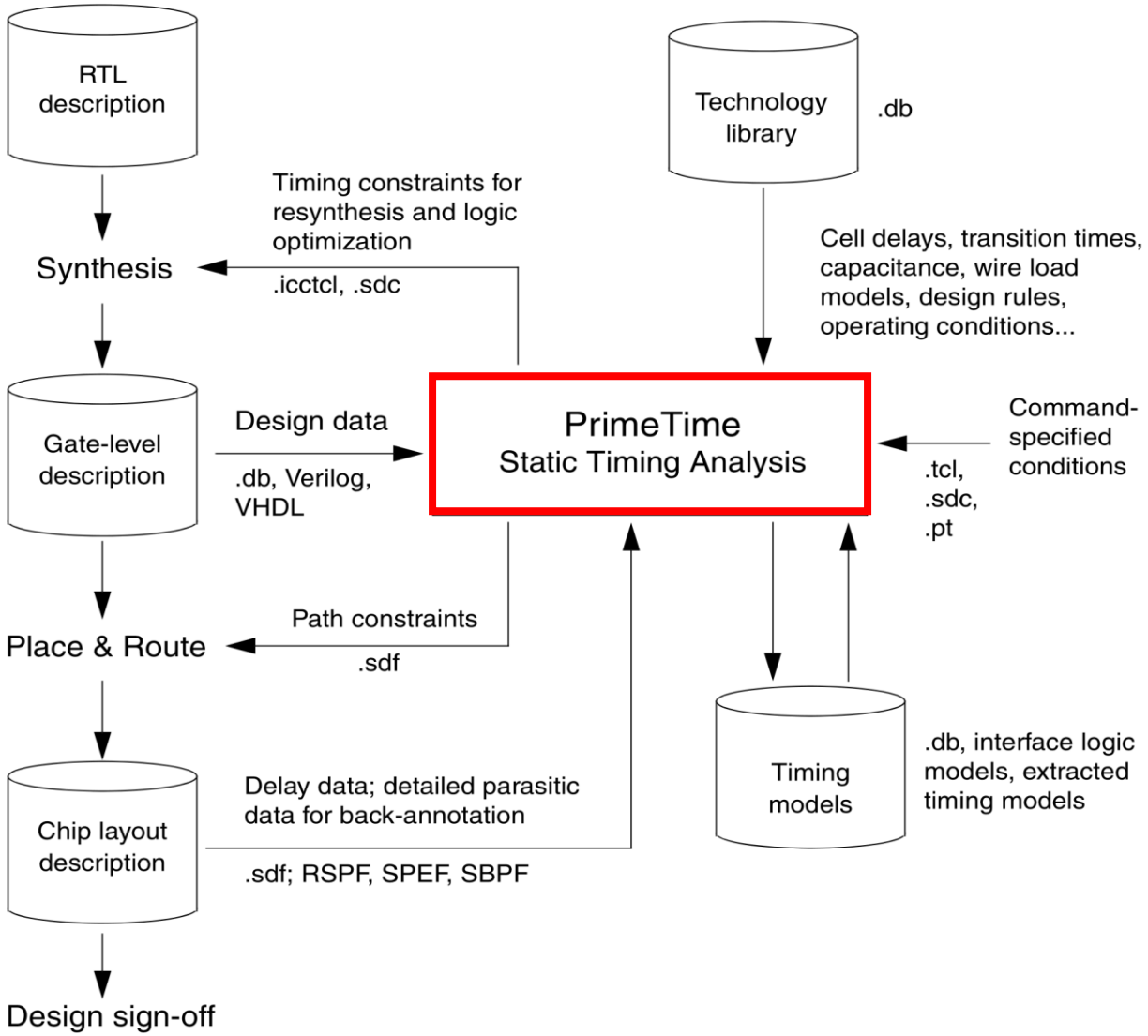
Part 2: Overview of Static Timing Analysis

CONTENT

01-PrimeTime

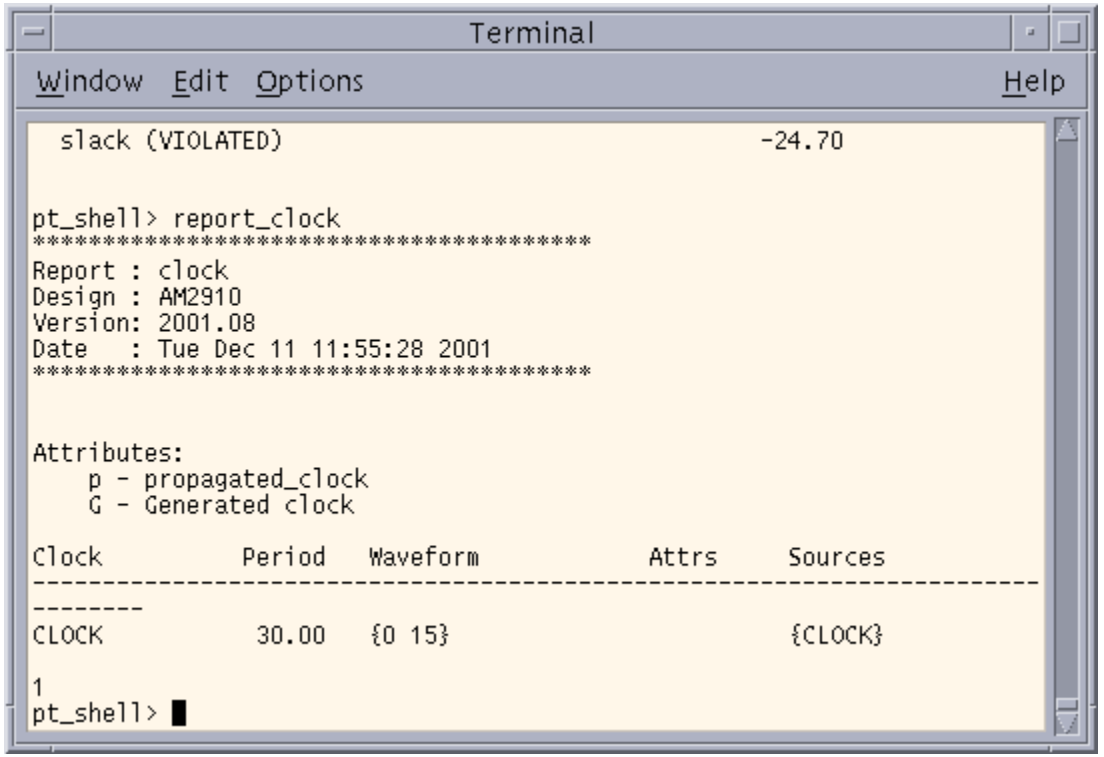
- PrimeTime is a full-chip, gate-level static timing analysis tool that is an essential part of the design and analysis flow for today's large chip designs.
- PrimeTime exhaustively validates the timing performance of a design by checking all possible paths for timing violations, without using logic simulation or test vectors.
- PrimeTime fits ideally into the Synopsys physical synthesis flow because it uses many of the same libraries, databases, and commands as other Synopsys tools such as Design Compiler. It can also operate as a standalone static timing analyzer in other design flows.

01-PrimeTime



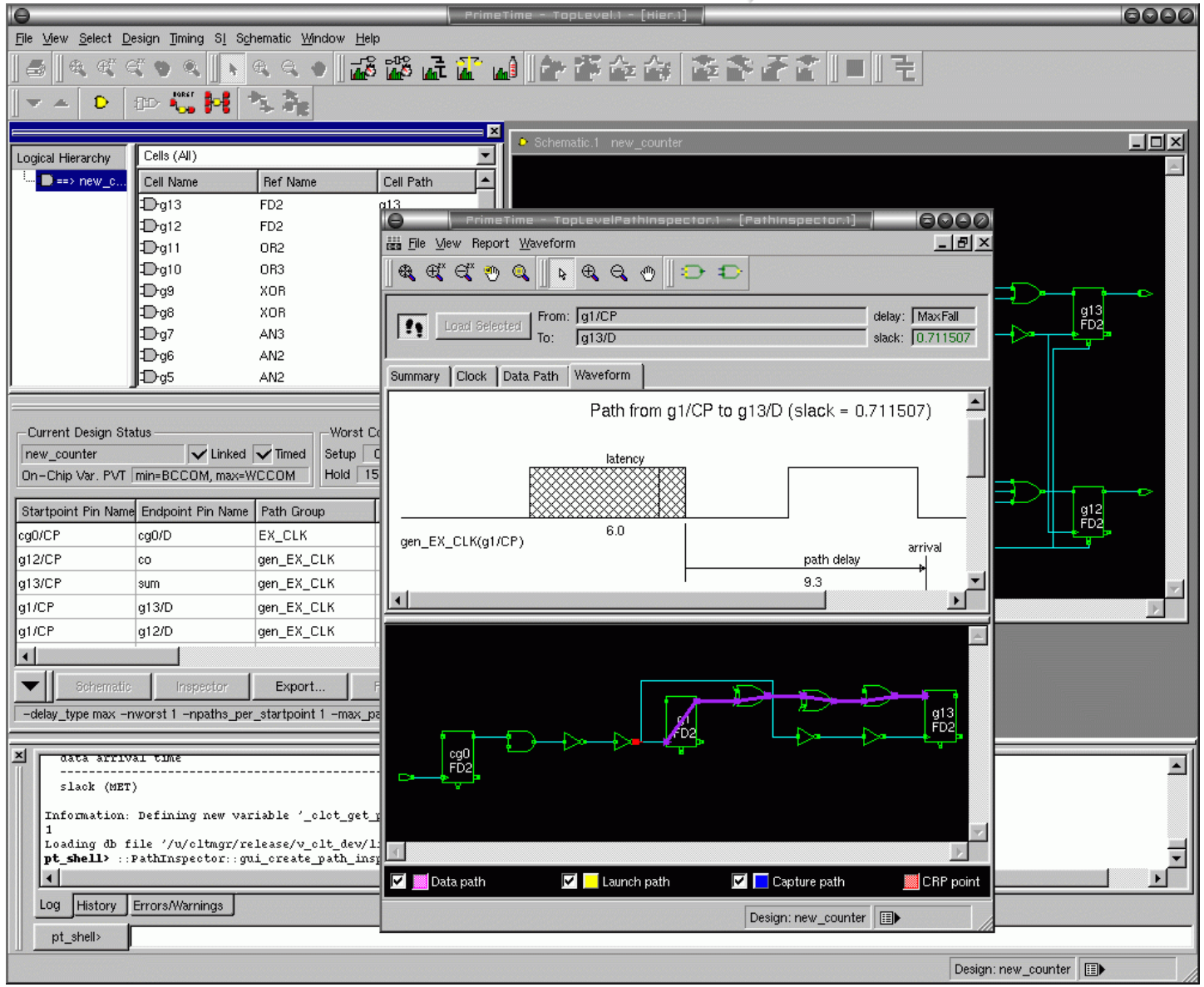
01-PrimeTime

PrimeTime offers two command environments for timing analysis: pt_shell and the graphical user interface (GUI).



The pt_shell interface is a command-line and script-execution environment based on the tool command language (Tcl) programming language.

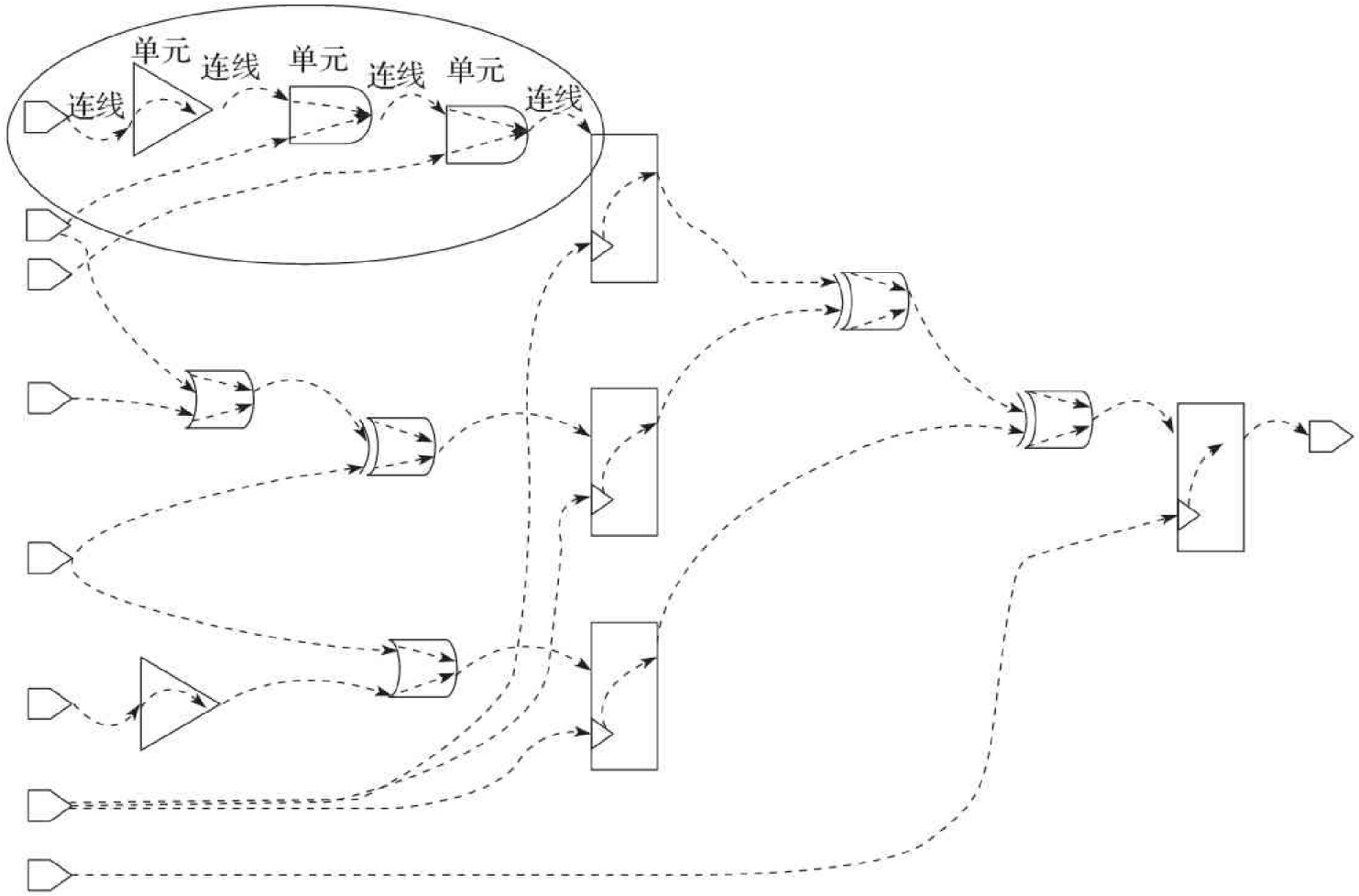
1-PrimeTime


PrimeTime - TopLevel.1 - [Hier.1]
File View Select Design Timing SI Schematic Window Help
Logical Hierarchy
Cells (All)
Cell Name
Ref Name
Cell Path
g13 FD2
g12 FD2
g11 OR2
g10 OR3
g9 XOR
g8 XOR
g7 AN3
g6 AN2
g5 AN2
Schematic.1 new_counter
PrimeTime - TopLevelPathInspector.1 - [PathInspector.1]
File View Report Waveform
Load Selected
From: g1/CP
To: g13/D
delay: MaxFall
slack: 0.711507
Summary Clock Data Path Waveform
Path from g1/CP to g13/D (slack = 0.711507)
latency
6.0
gen_EX_CLK(g1/CP)
path delay
9.3
arrival
Current Design Status
new_counter
Linked
Timed
On-Chip Var. PVT
min=BCCOM, max=WCCOM
Setup
Hold
Startpoint Pin Name
Endpoint Pin Name
Path Group
cg0/CP cg0/D EX_CLK
g12/CP co gen_EX_CLK
g13/CP sum gen_EX_CLK
g1/CP g13/D gen_EX_CLK
g1/CP g12/D gen_EX_CLK
Schematic
Inspector
Export...
-delay_type max -nworst 1 -npaths_per_startpoint 1 -max_pa
slack (MET)
Information: Defining new variable '_clct_get_p
Loading db file '/u/cltmgr/release/v_clt_dev/l
pt_shell> ::PathInspector::gui_create_path_insp
Log
History
Errors/Warnings
pt_shell>
cg0 FD2
g1 FD2
g13 FD2
g12 FD2
Data path
Launch path
Capture path
CRP point
Design: new_counter
Design: new_counter

Part 2: Overview of Static Timing Analysis

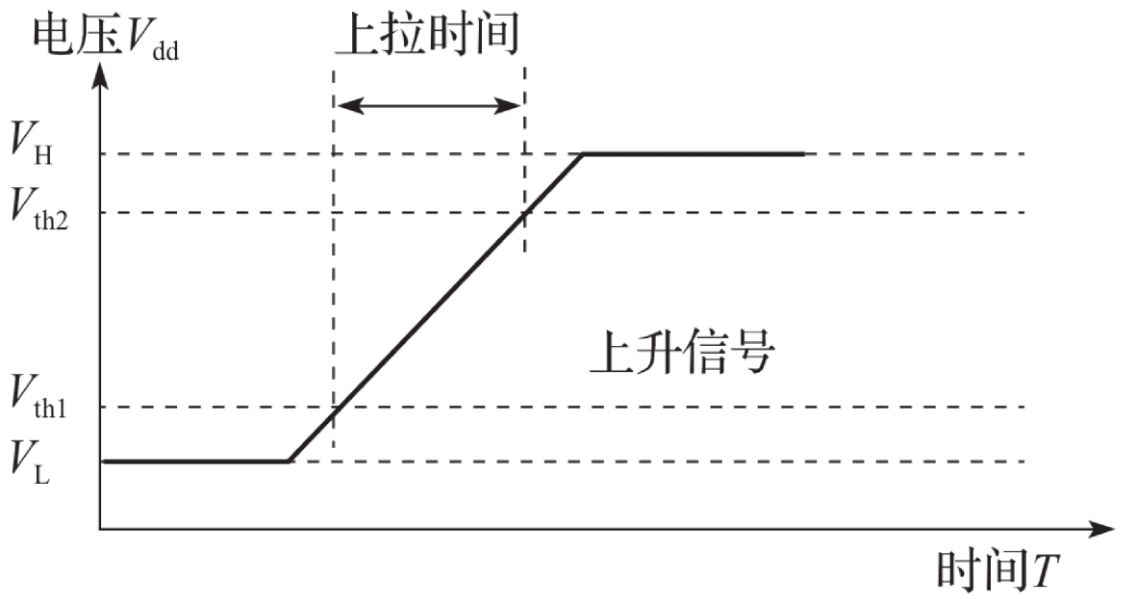
CONTENT

Timing Arc



Cell delay

(1) Transition delay



(1) slew_lower_threshold_pct_fall

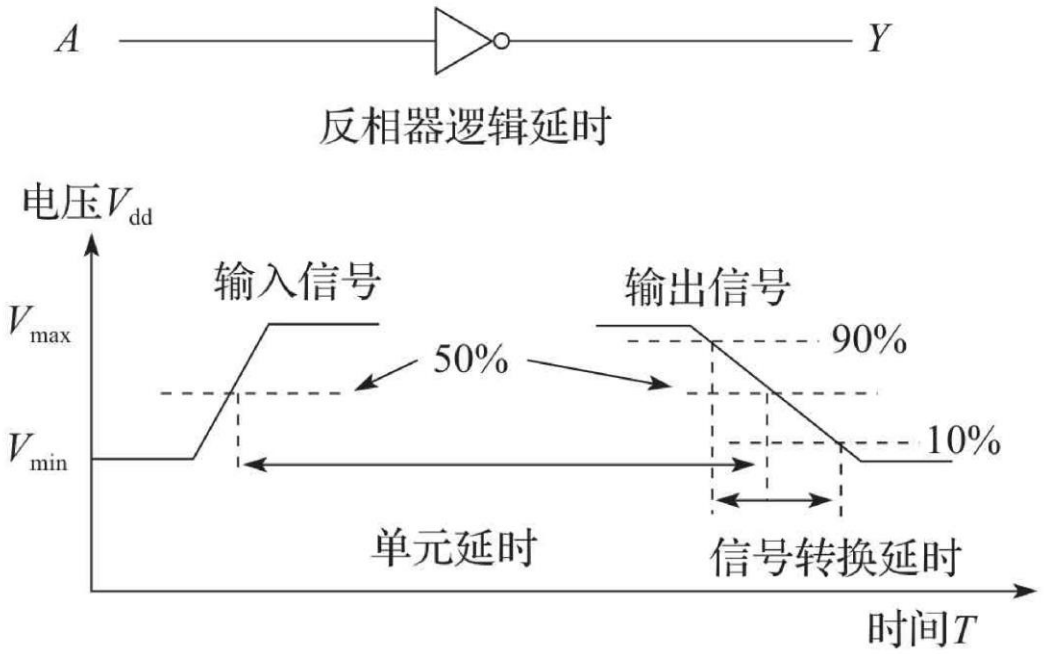
(2) slew_upper_threshold_pct_fall

(3) slew_lower_threshold_pct_rise

(4) slew_upper_threshold_pct_rise

Cell delay

(2) Logic gate delay



(1) input_threshold_pct_rise

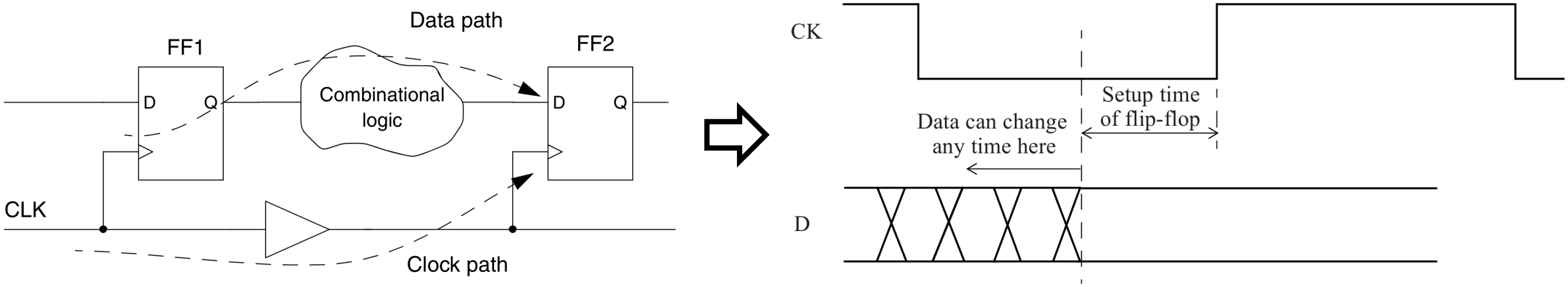
(2) output_threshold_pct_rise

(3) output_threshold_pct_fall

(4) input_threshold_pct_fall

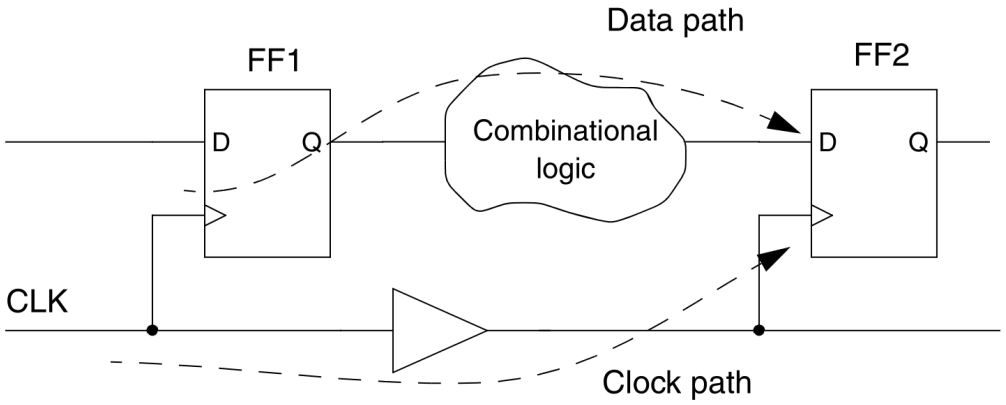
Setup time and hold time

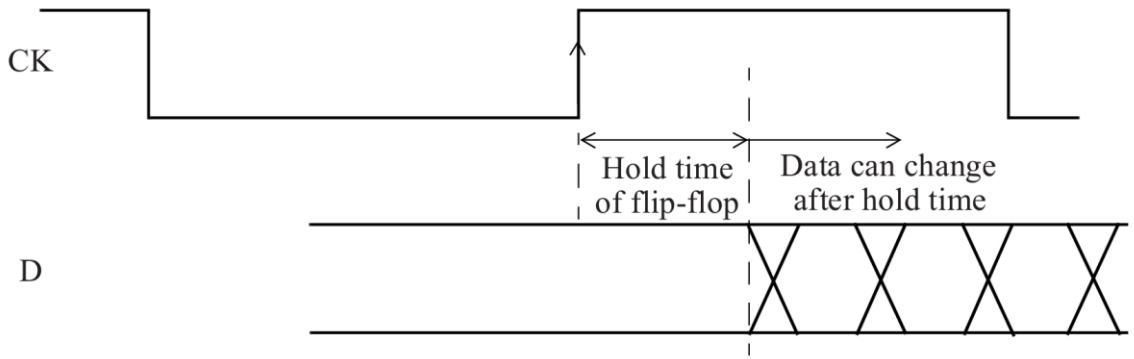
- A setup constraint specifies how much time is necessary for data to be available at the input of a sequential device **before** the clock edge that captures the data in the device.
- This constraint enforces a **maximum** delay on the data path relative to the clock path.



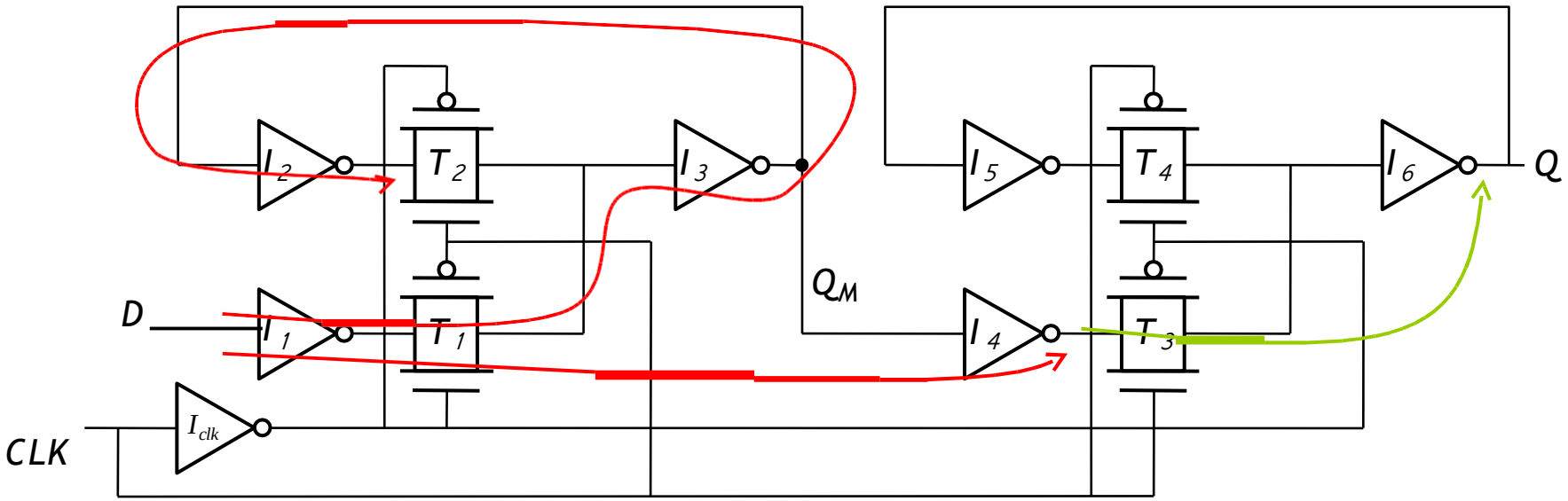
Setup time and hold time

- A hold constraint specifies how much time is necessary for data to be stable at the input of a sequential device **after** the clock edge that captures the data in the device.
- This constraint enforces a **minimum** delay on the data path relative to the clock path.





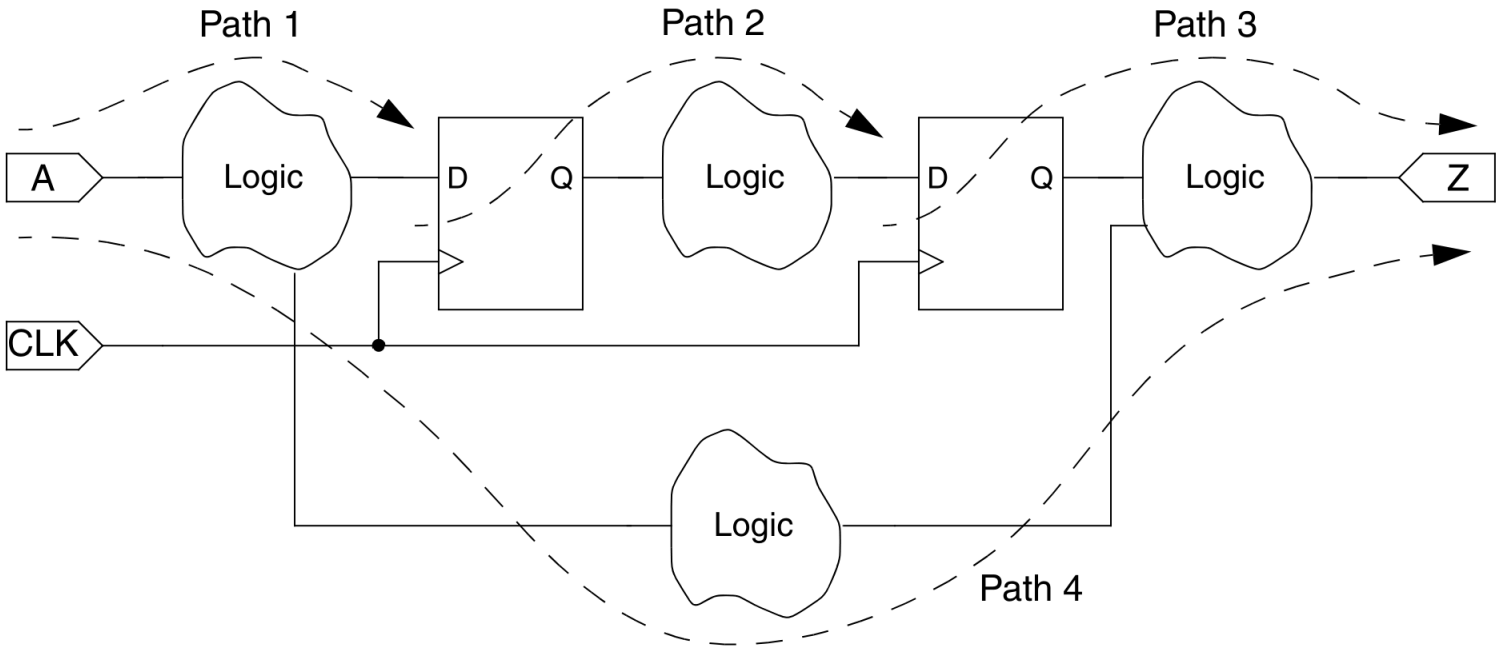
Setup time and hold time



Timing path

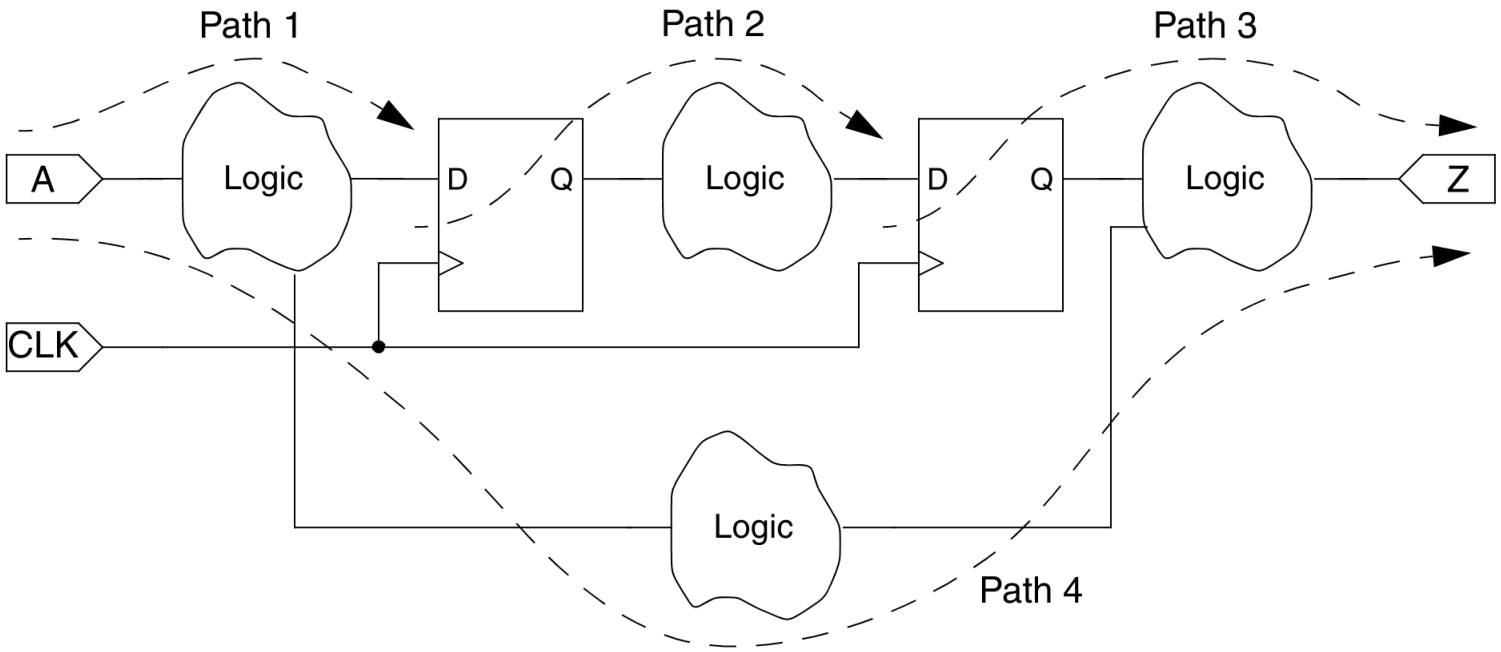
The first step performed by PrimeTime for timing analysis is to break the design down into a set of timing paths.

Each path has **a startpoint** and **an endpoint**.



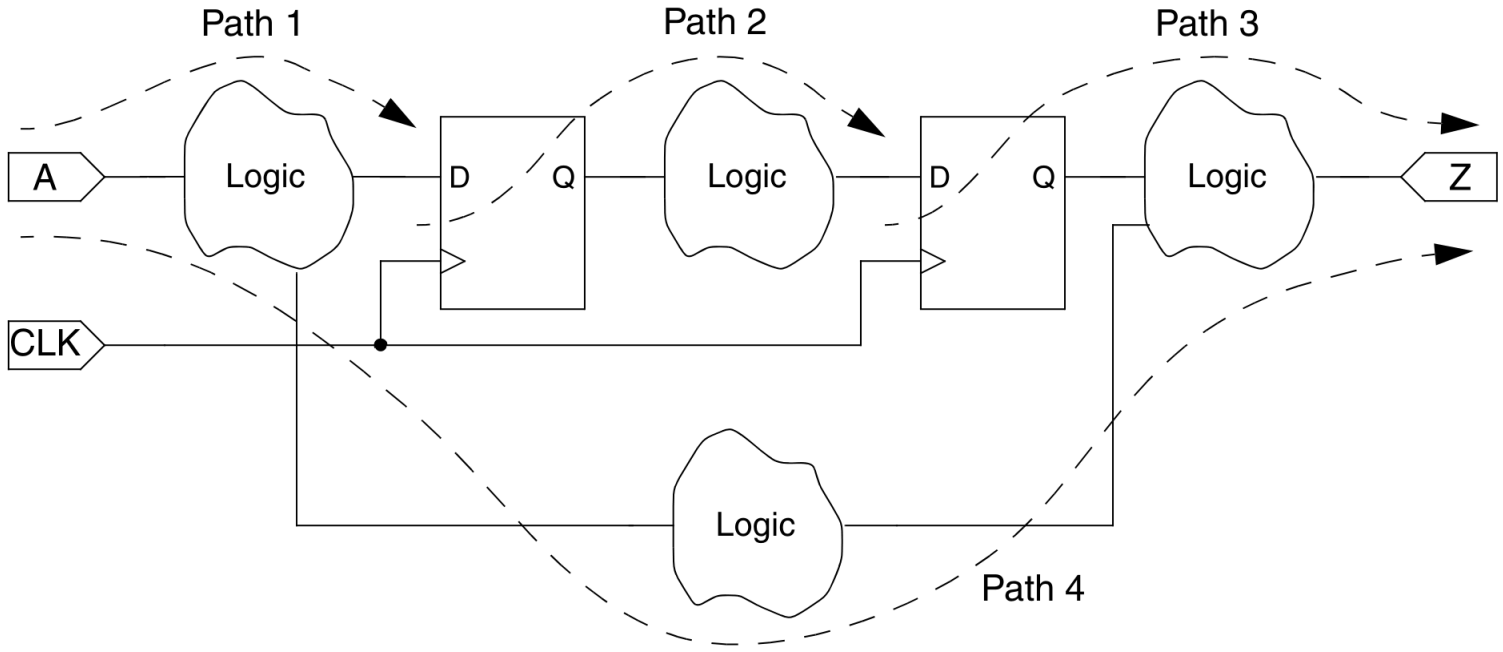
Timing path

The **startpoint** of a path is a **clock pin** of a sequential element, or possibly an **input port** of the design (because the input data can be launched from some external source).

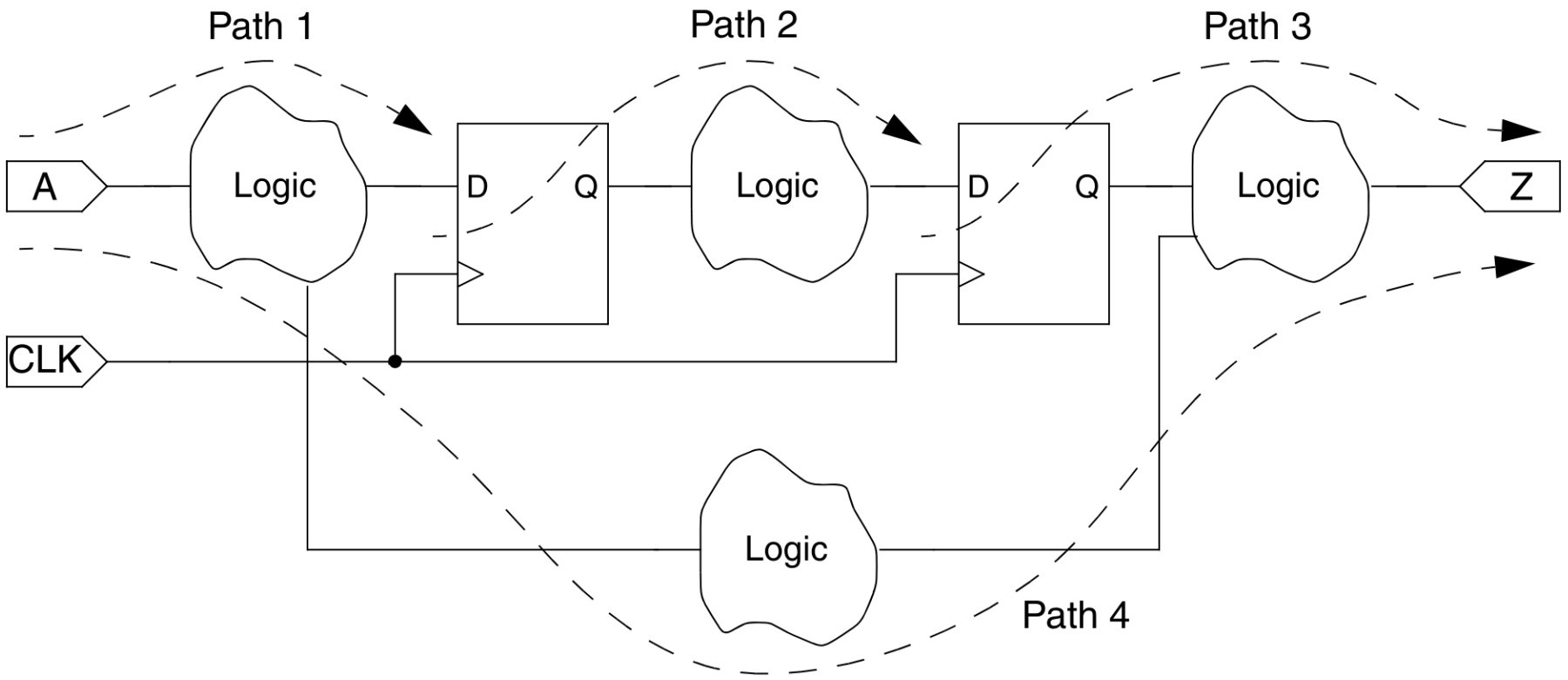


Timing path

The **endpoint** of a path is a **data input pin** of a sequential element, or possibly an **output port** of the design (because the output data can be captured by some external sink).

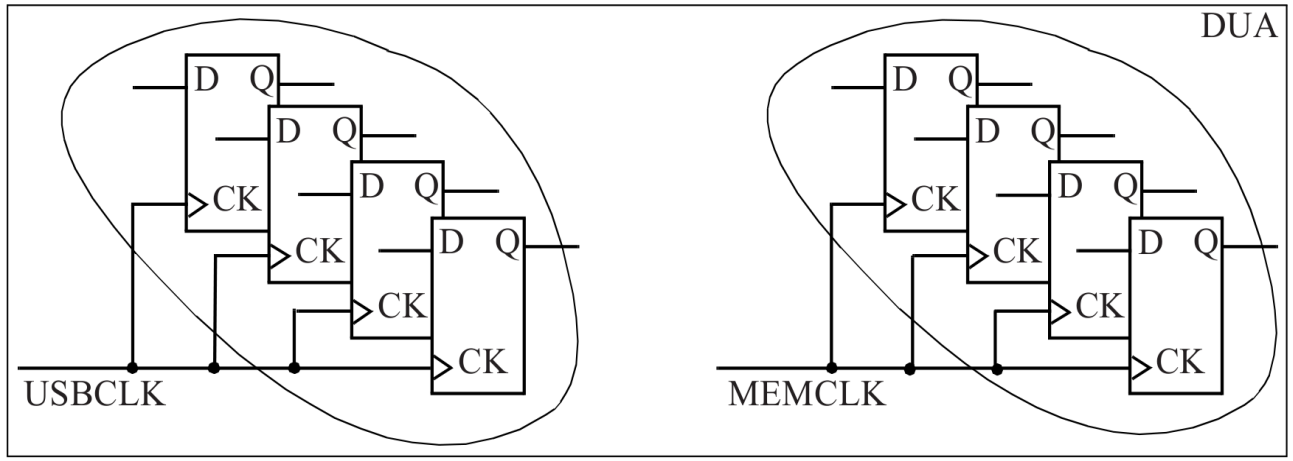


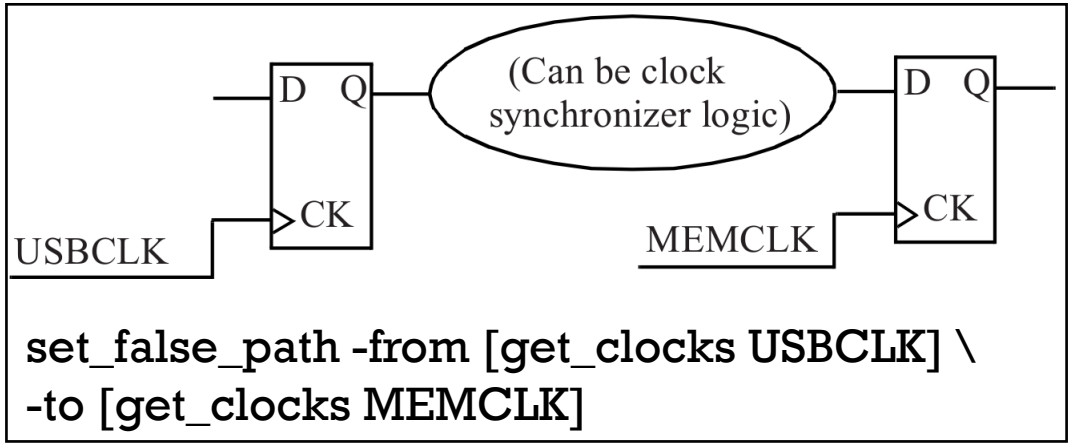
Timing path



Clock Domains

The set of flip-flops being fed by one clock is called its clock domain.






```
set_false_path -from [get_clocks USBCLK] \
-to [get_clocks MEMCLK]
```

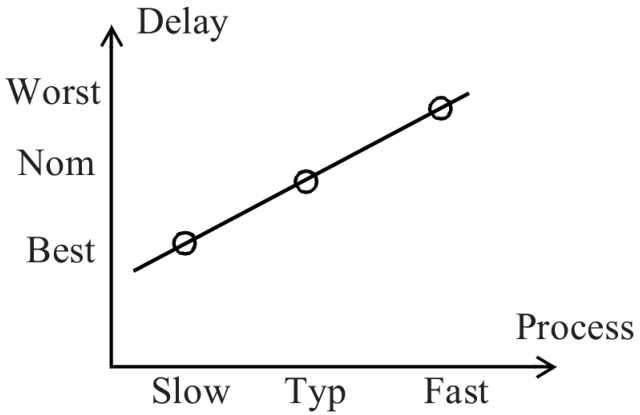
Operating Conditions

- Static timing analysis is typically performed at a specific operating condition.
- An operating condition is defined as a combination of **Process, Voltage and Temperature** (PVT).
- Cell delays and interconnect delays are computed based upon the specified operating condition.

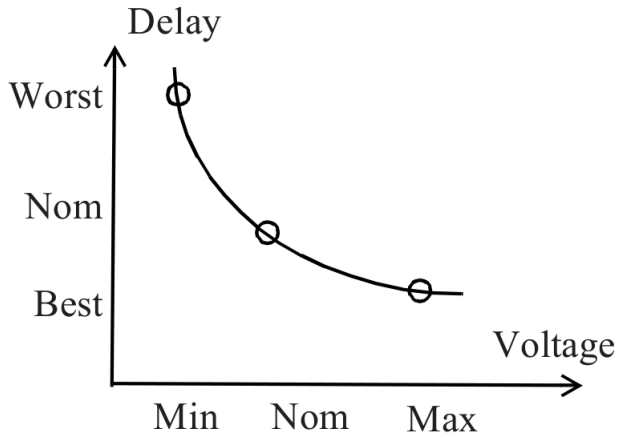
Operating Conditions

- There are **three** kinds of manufacturing process models that are provided by the semiconductor foundry for digital designs: **slow** process models, **typical** process models, and **fast** process models.
- The slow and fast process models represent the **extreme corners** of the manufacturing process of a foundry.
- For **robust** design, the design is validated at the extreme corners of the manufacturing process as well as environment extremes for temperature and power supply.

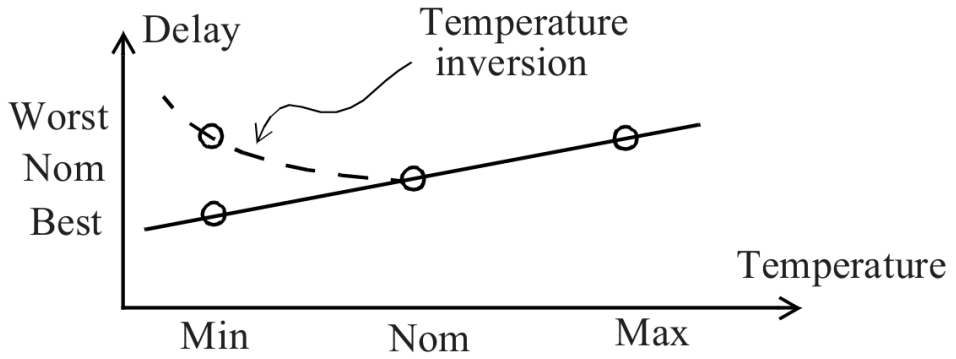
Operating Conditions



(a) Delay vs process.



(b) Delay vs voltage.



(c) Delay vs temperature.

It is important to decide the operating conditions that should be used for various static timing analyses.

Operating Conditions

The choice of what operating condition to use for STA is also governed by the operating conditions under which cell libraries are available. Three standard operating conditions are:

- **WCS (Worst-Case Slow):** Process is slow, temperature is highest (say 125C) and voltage is lowest (say nominal 1.2V minus 10%).
- **TYP (Typical):** Process is typical, temperature is nominal (say 25C) and voltage is nominal (say 1.2V).
- **BCF (Best-Case Fast):** Process is fast, temperature is lowest (say -40C) and voltage is highest (say nominal 1.2V plus 10%).

```
set_operating_conditions “WCCOM” -library mychip
# Use the operating condition called WCCOM defined in the cell library mychip.
```

参考书目

- Static Timing Analysis for Nanometer Designs: A Practical Approach. J. Bhasker, Rakesh Chadha. Springer Science Business Media, LLC 2009. Chaper 2.
- 集成电路静态时序分析与建模. 刘峰, 机械工业出版社.出版时间：2016-07-01. 第二章.
- PrimeTime ® Fundamentals User Guide. Synopsys. Version F-2011.12, December 2011.

谢谢聆听！

个人教学工作主页https://customizablecomputinglab.github.io/