# Quartus® II Software Design Series: Timing Analysis

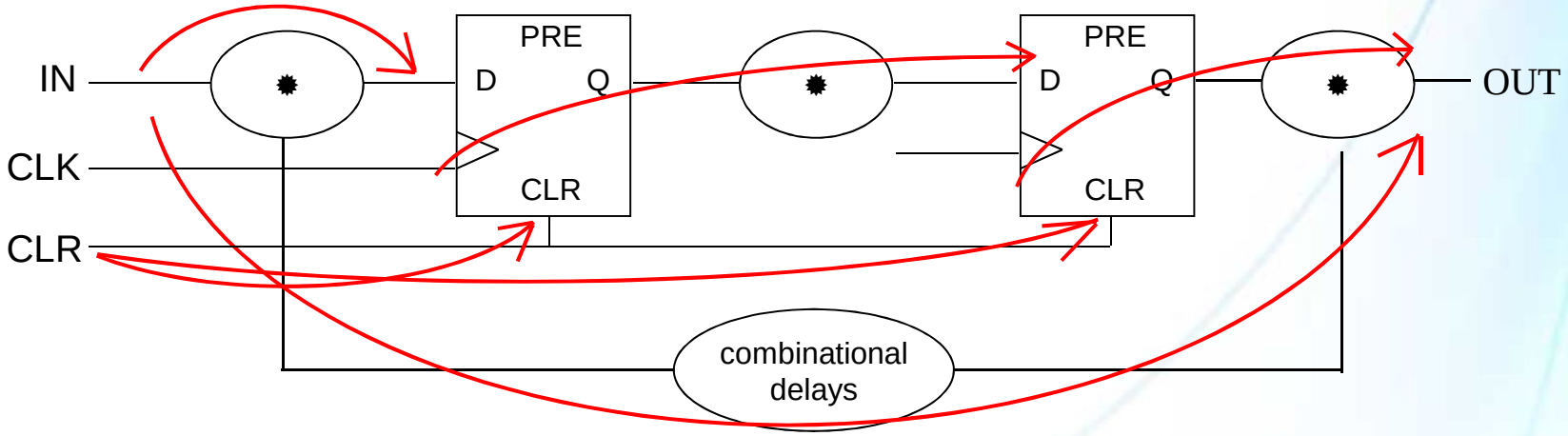
## *- Timing analysis basics*

# Objectives

- Display a complete understanding of timing analysis

# How does timing verification work?

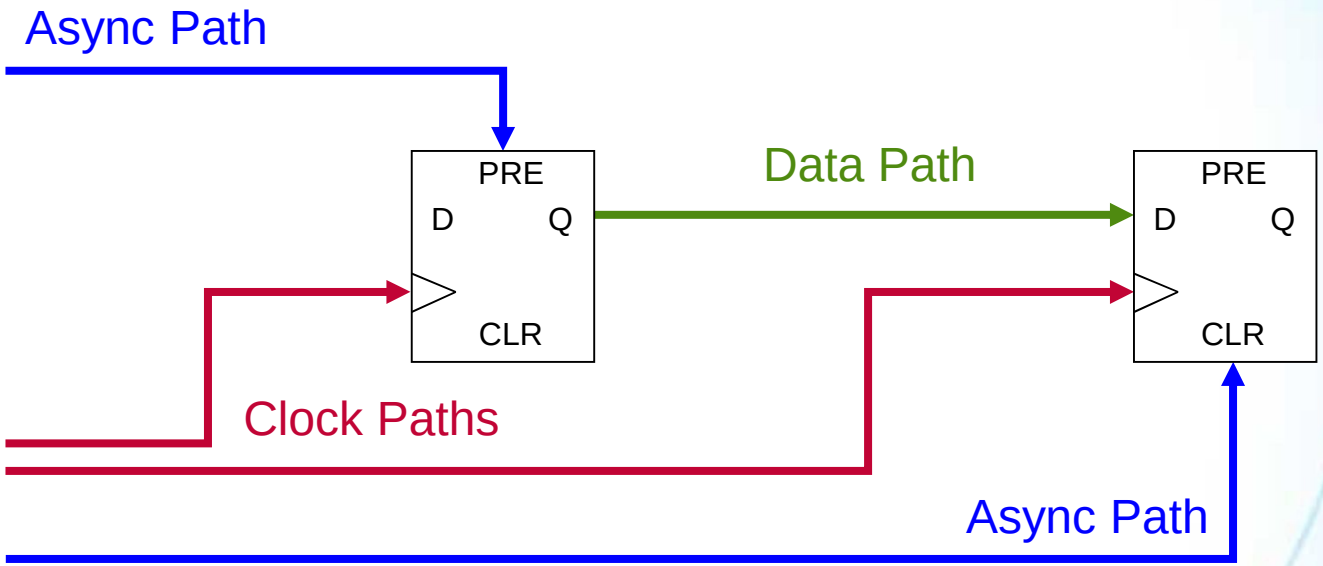
- Every device path in design must be analyzed with respect to timing specifications/requirements
  - Catch timing-related errors faster and easier than gate-level simulation & board testing
- Designer must enter timing requirements & exceptions
  - Used to guide fitter during placement & routing
  - Used to compare against actual results



# Timing Analysis Basics

- Launch vs. latch edges
- Setup & hold times
- Data & clock arrival time
- Data required time
- Setup & hold slack analysis
- I/O analysis
- Recovery & removal
- Timing models

# Path & Analysis Types




Three types of Paths:

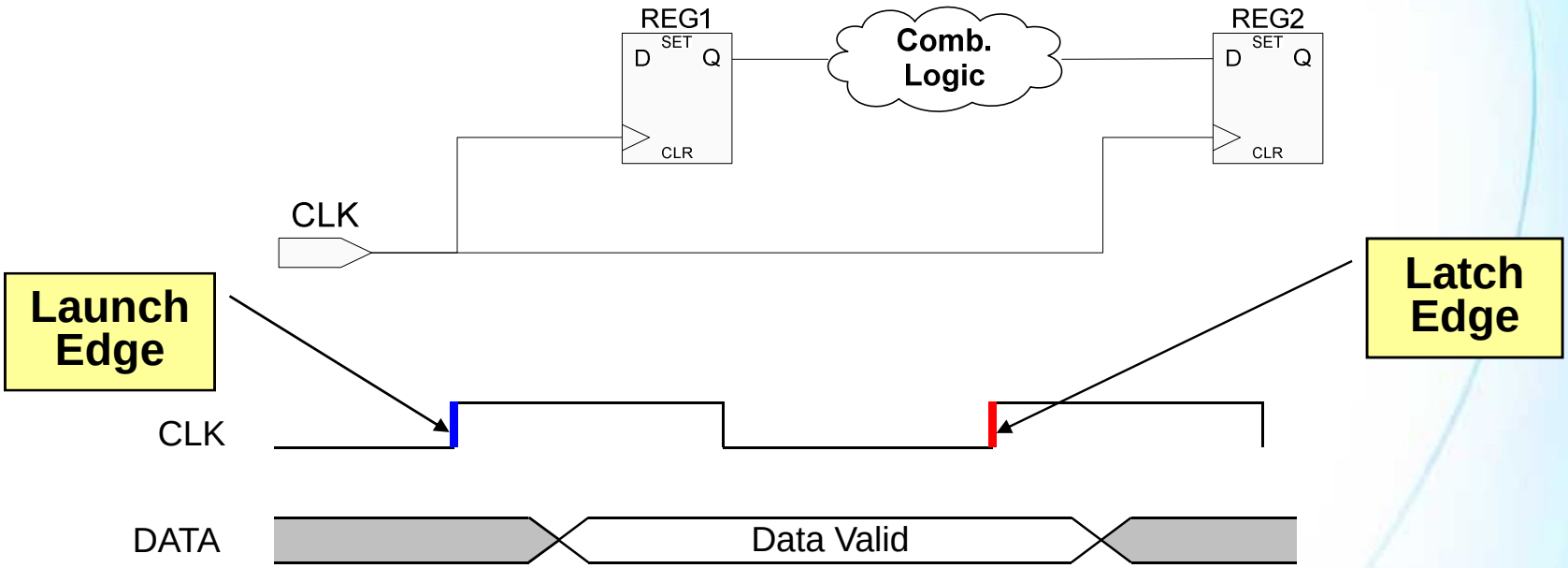
1. Clock Paths
2. Data Path
3. Asynchronous Paths*

Two types of Analysis:

1. Synchronous – clock & data paths
2. Asynchronous* – clock & async paths

***Asynchronous refers to signals feeding the asynchronous control ports of the registers***

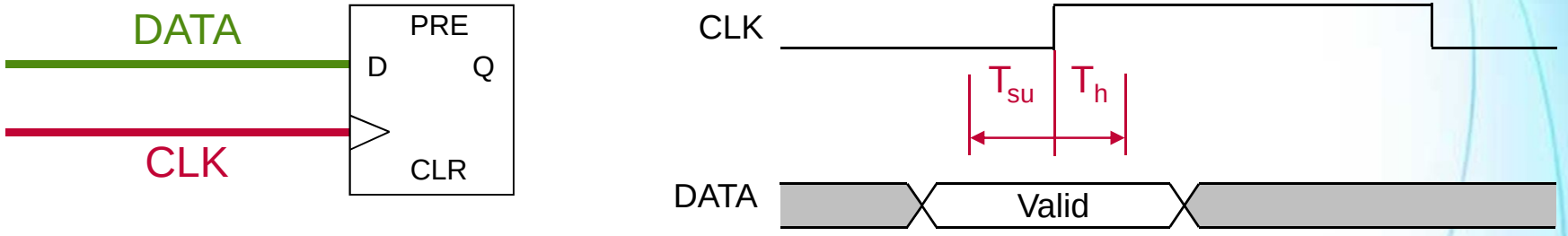
# Launch & Latch Edges




Launch Edge: the edge which “launches” the data from source register

Latch Edge: the edge which “latches” the data at destination register (with respect to the launch edge, selected by timing analyzer; typically 1 cycle)

# Setup & Hold



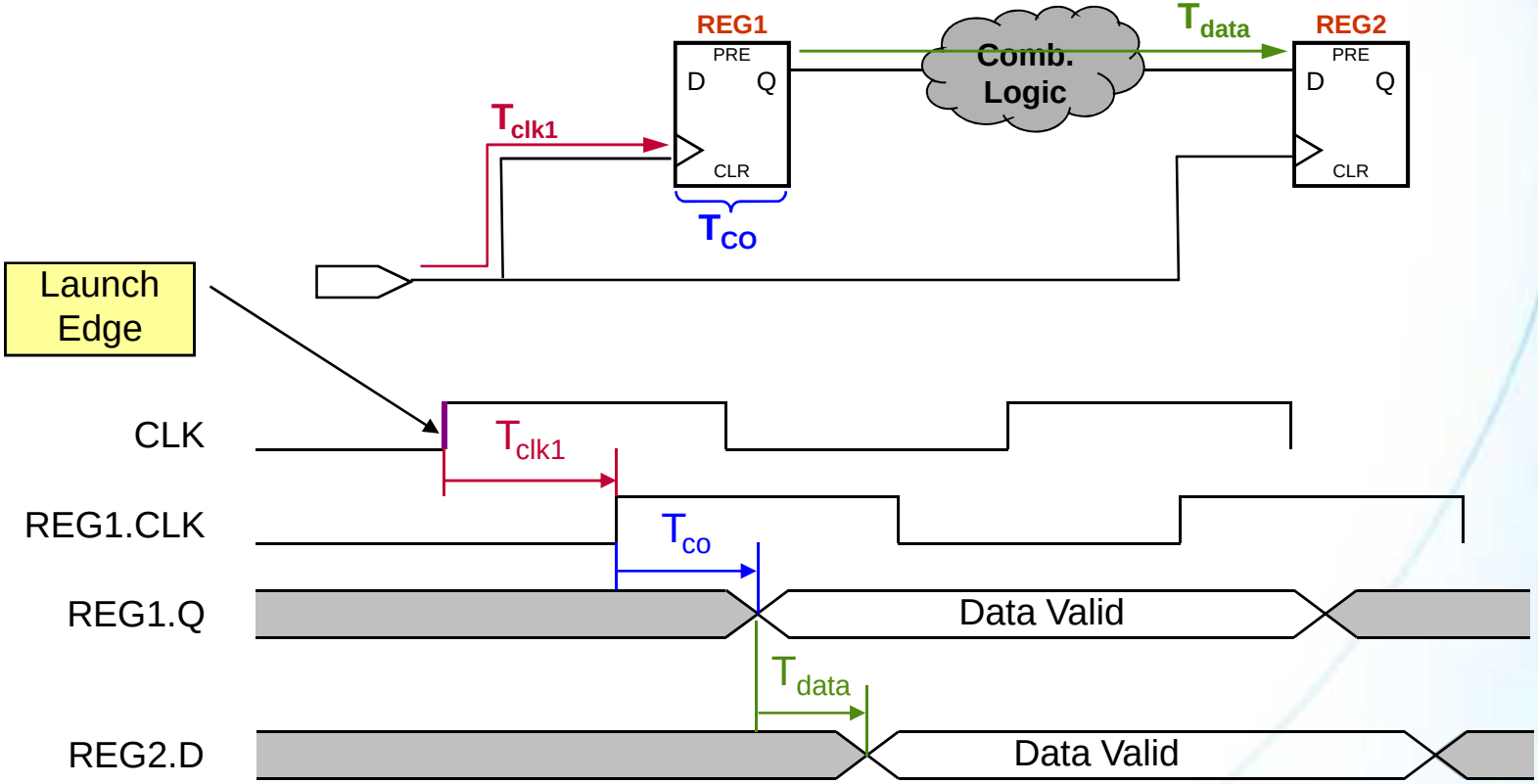

Setup: The minimum time data signal must be stable BEFORE clock edge

Hold: The minimum time data signal must be stable AFTER clock edge

***Together, the setup time and hold time form a Data Required Window, the time around a clock edge in which data must be stable.***

# Data Arrival Time

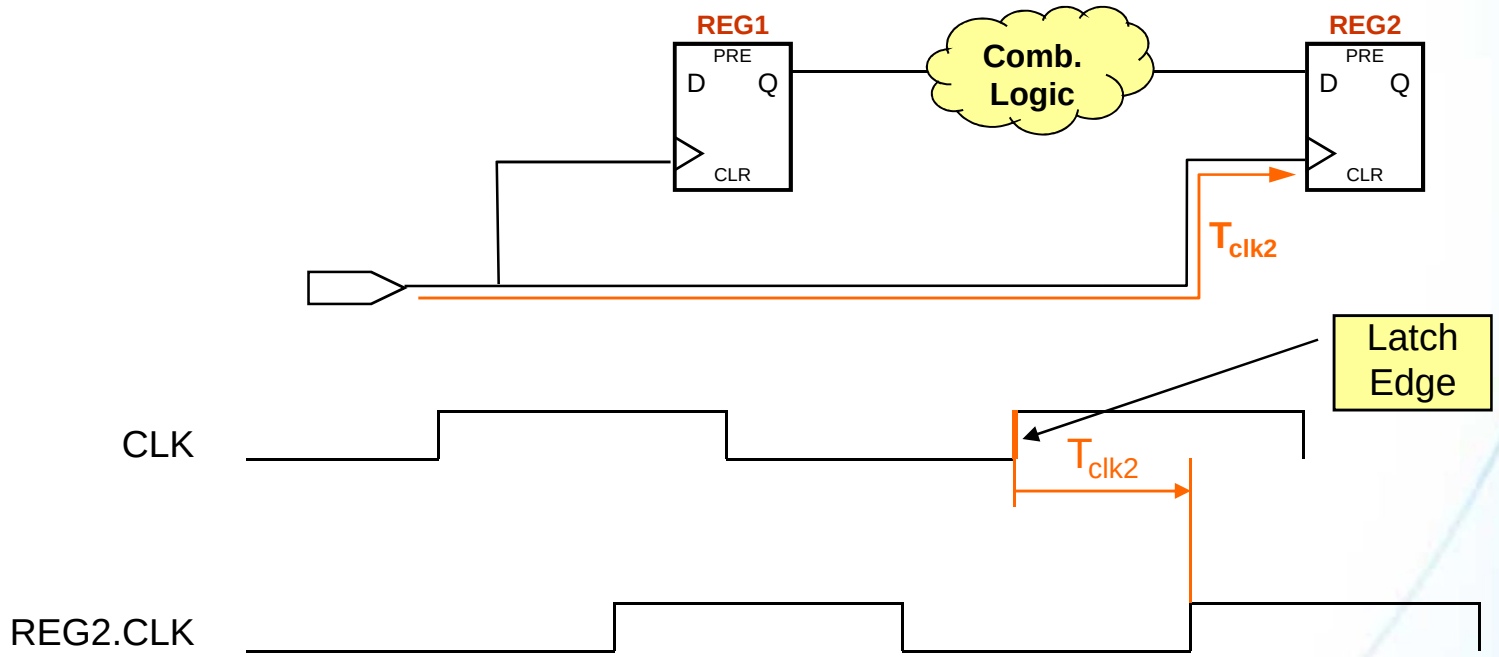
- The time for data to arrive at destination register's D input




Data Arrival Time = launch edge + $T_{clk1}$ + $T_{co}$ +$T_{data}$

# Clock Arrival Time

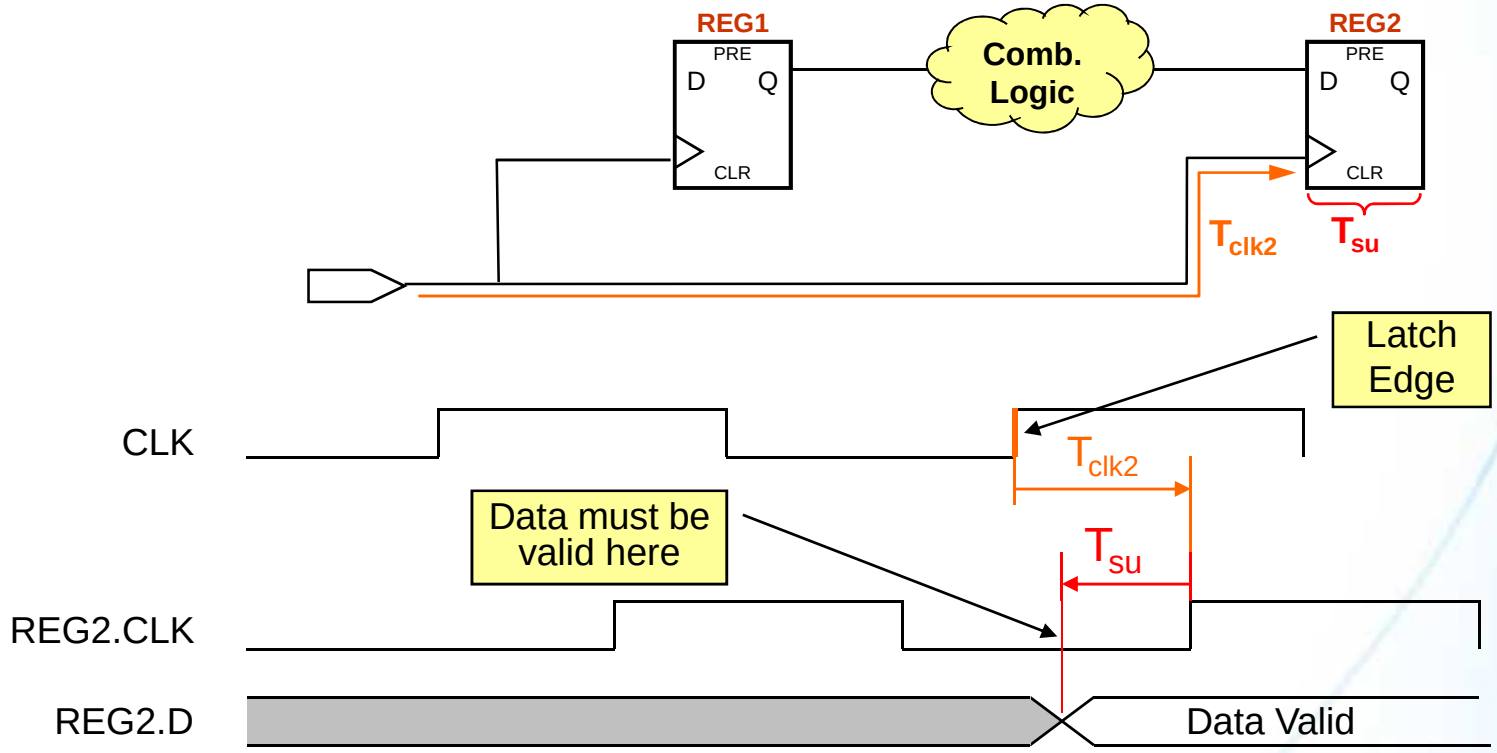
- The time for clock to arrive at destination register's clock input




Clock Arrival Time = latch edge + $T_{clk2}$

# Data Required Time - Setup

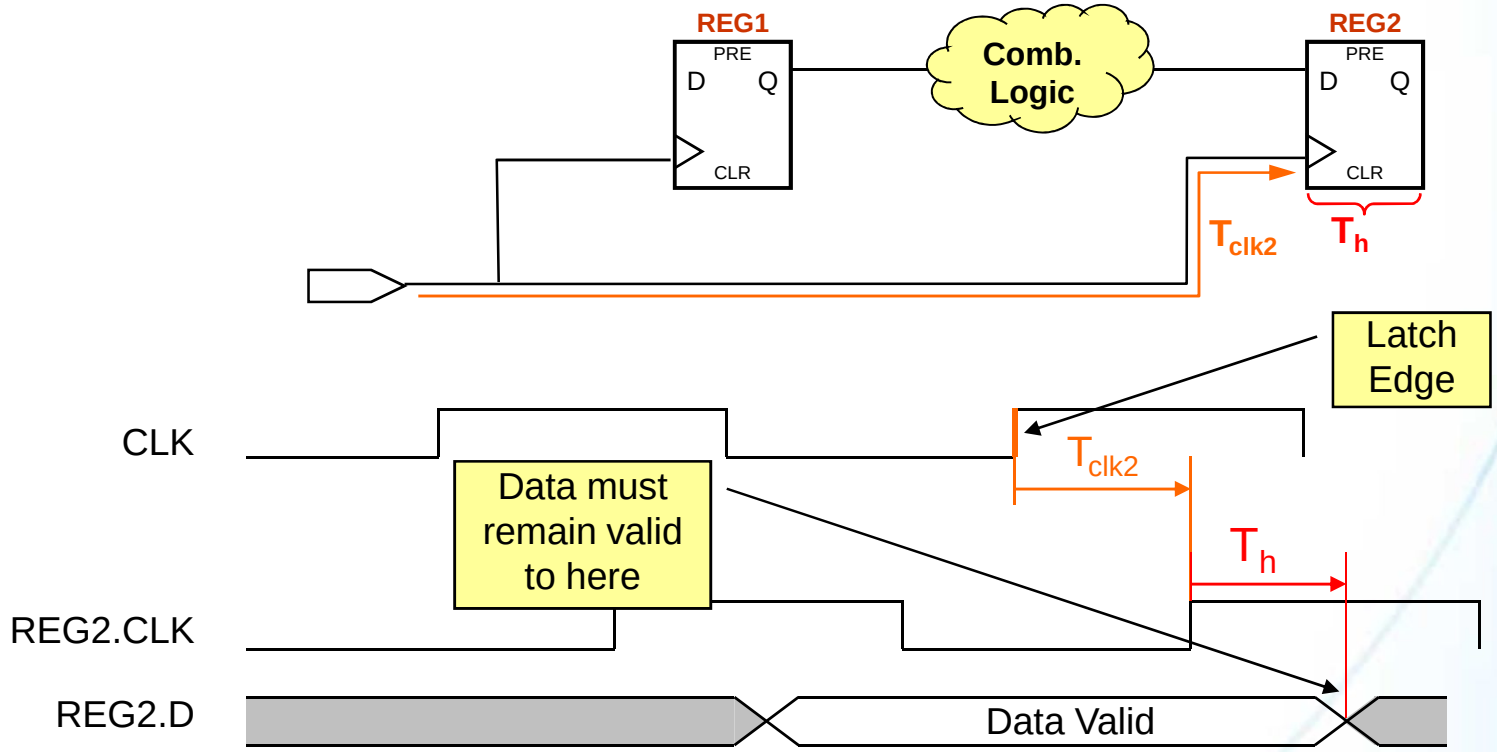
- The minimum time required for the data to get latched into the destination register




Data Required Time = Clock Arrival Time - $T_{su}$ - Setup Uncertainty

# Data Required Time - Hold

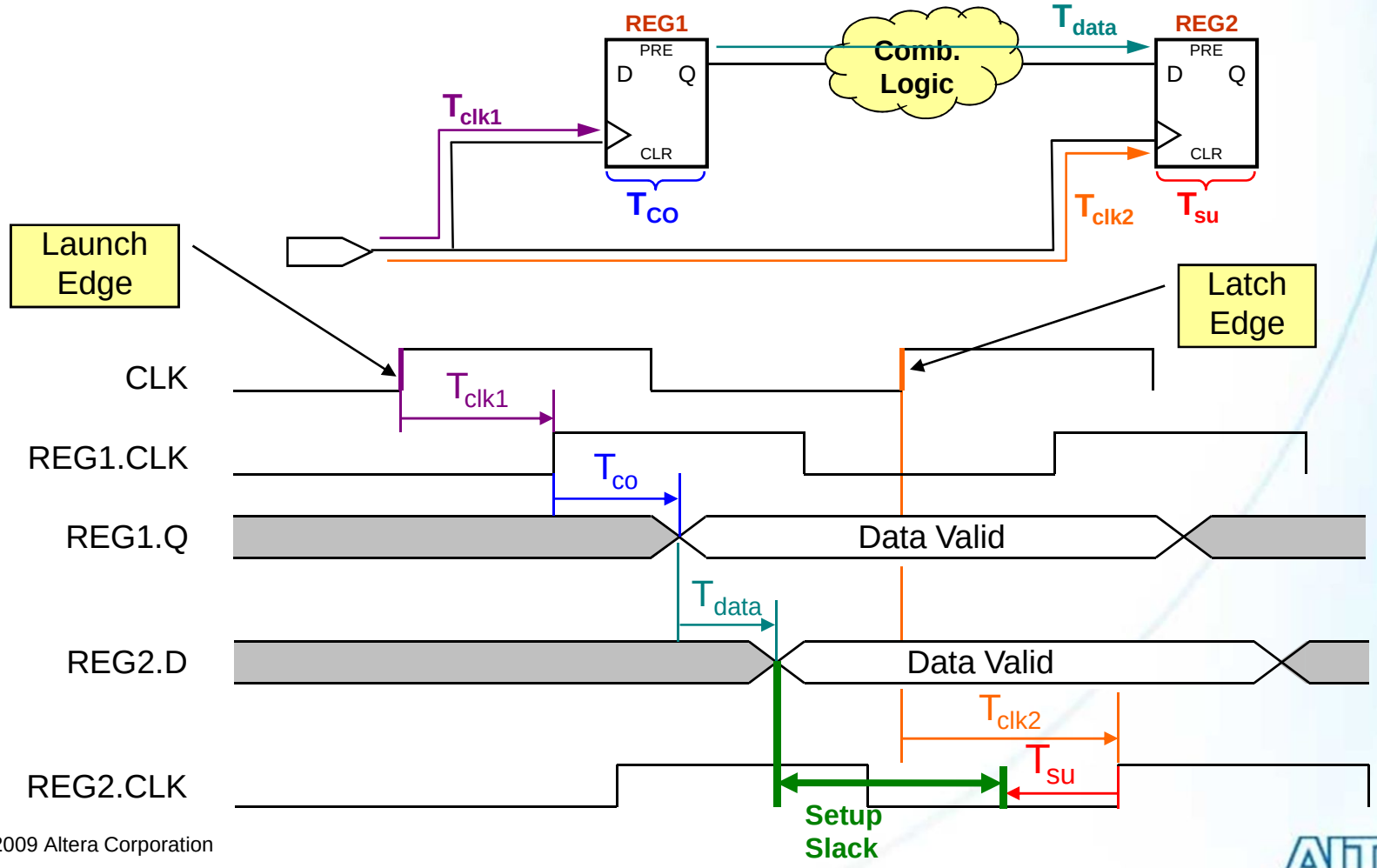
- The minimum time required for the data to get latched into the destination register




Data Required Time = Clock Arrival Time + $T_h$ + Hold Uncertainty

# Setup Slack

- The margin by which the setup timing requirement is met. It ensures launched data arrives in time to meet the latching requirement.



# Setup Slack (cont'd)

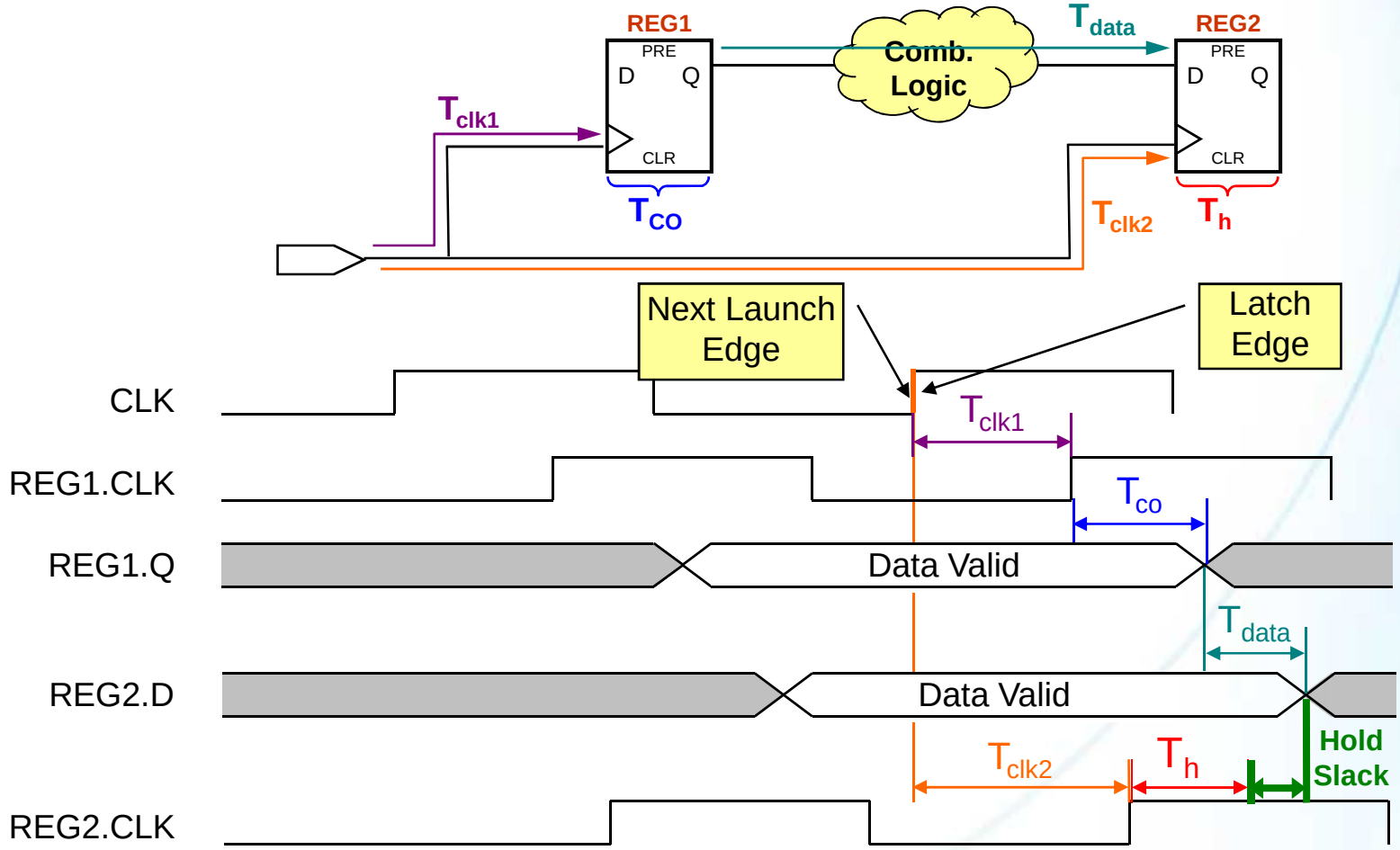
*Setup Slack = Data Required Time*
*– Data Arrival Time*

Positive slack
- Timing requirement met

Negative slack
- Timing requirement not met

# Hold Slack

- The margin by which the hold timing requirement is met. It ensures latch data is not corrupted by data from another launch edge.



# Hold Slack (cont'd)

*Hold Slack = Data Arrival Time*
*– Data Required Time*

Positive slack

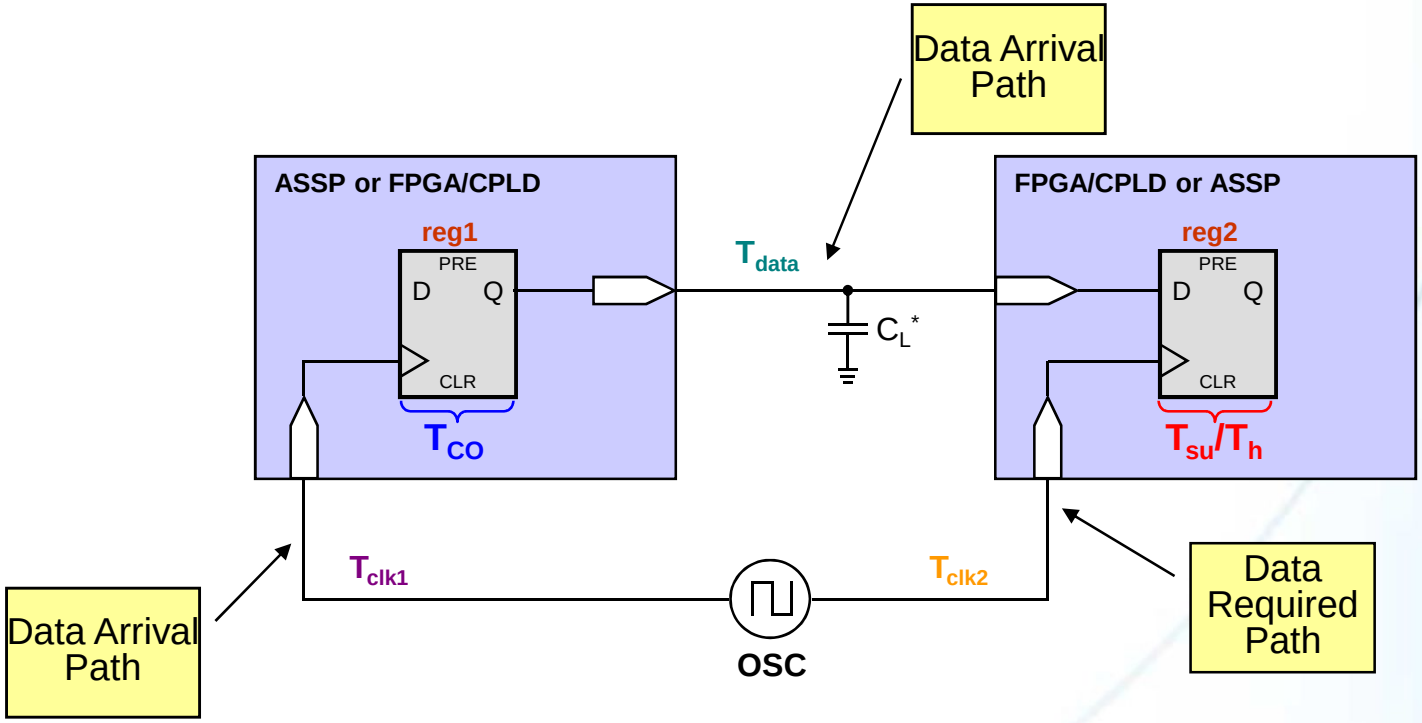
– Timing requirement met

Negative slack

– Timing requirement not met

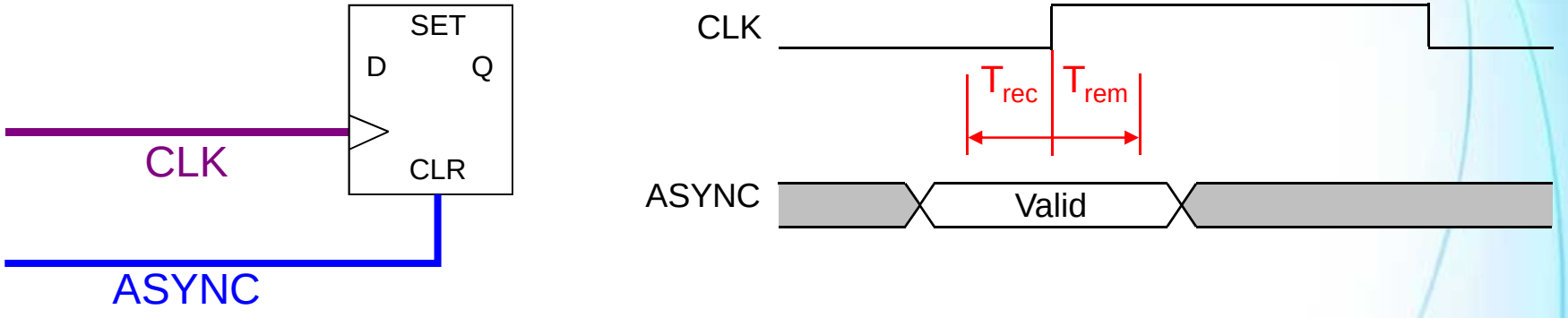
# I/O Analysis

- Analyzing I/O performance in a synchronous design uses the same slack equations
  - Must include external device & PCB timing parameters




** Represents delay due to capacitive loading*

# Recovery & Removal

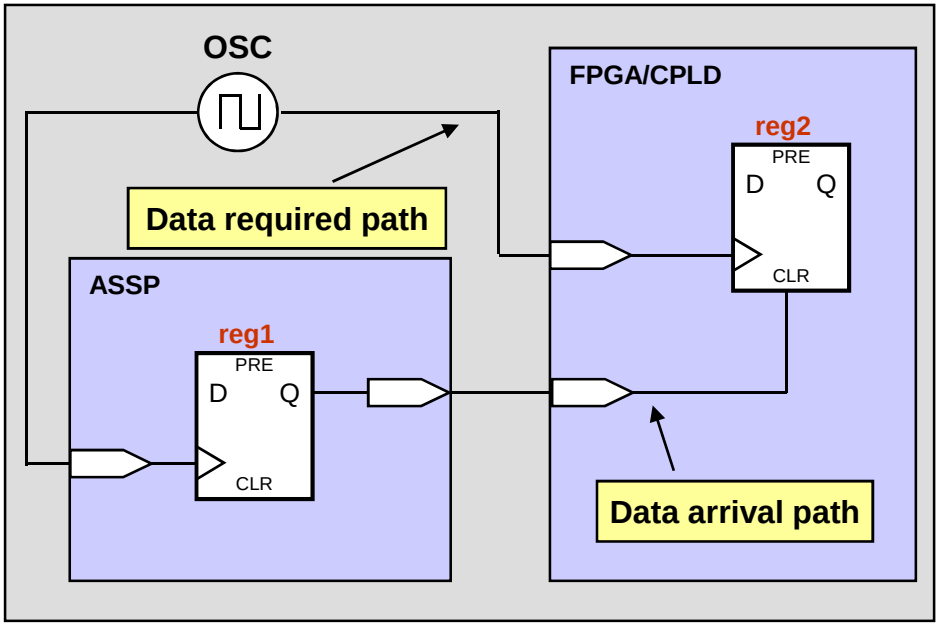



Recovery: The minimum time an asynchronous signal must be stable BEFORE clock edge

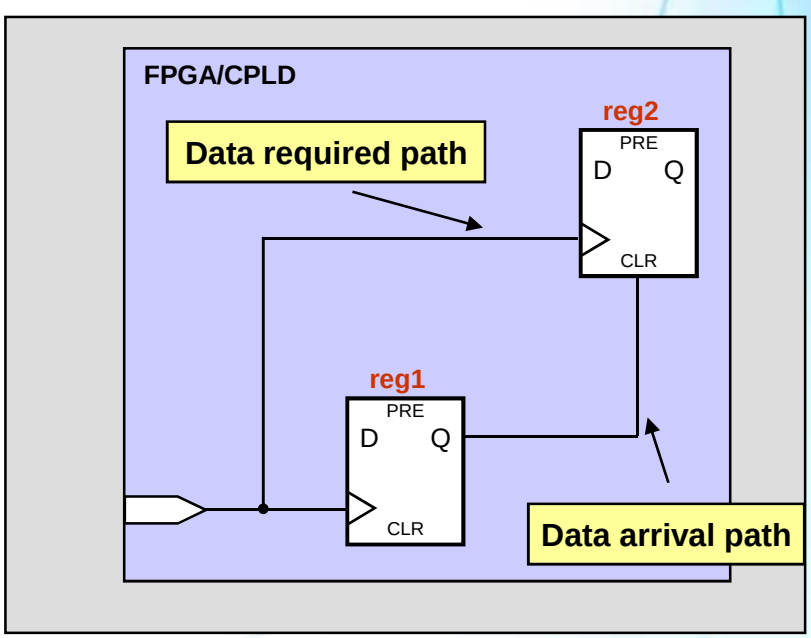
Removal: The minimum time an asynchronous signal must be stable AFTER clock edge

# Asynchronous = Synchronous?

- Asynchronous control signal source is assumed synchronous
  - Slack equations still apply
    - data arrival path = asynchronous control path
    - $T_{su} \approx T_{rec}$; $T_h \approx T_{rem}$
  - External device & board timing parameters may be needed (Ex. 1)




Example 1




Example 2

# Why Are These Calculations Important?

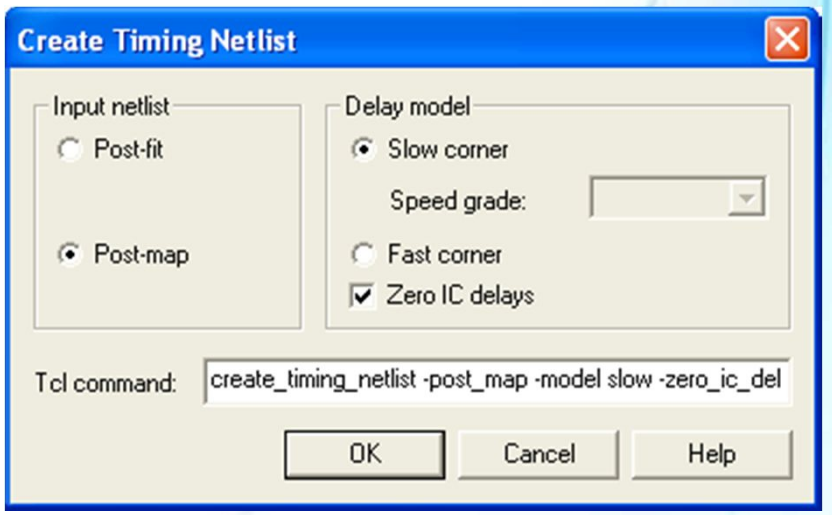
- Calculations are important when timing violations occur
  - Need to be able to understand cause of violation
- Example causes
  - Data path too long
  - Requirement too short (incorrect analysis)
  - Large clock skew signifying a gated clock, etc.
- TimeQuest timing analyzer uses them
  - Equations to calculate slack
  - Terminology (launch and latch edges, Data Arrival Path, Data Required Path, etc.) in timing reports

# Timing Models in Detail

- Quartus II software models device timing at two PVT conditions by default
  - ***Slow Corner*** Model
    - Indicates slowest possible performance for any single path
    - Timing for slowest device at maximum operating temperature and $VCC_{MIN}$
  - ***Fast Corner*** Model
    - Indicates fastest possible performance for any single path
    - Timing for fastest device at minimum operating temperature and $VCC_{MAX}$
- Why two corner timing models?
  - Ensure **setup** timing is met in **slow** model
  - Ensure **hold** timing is met in **fast** model
    - Essential for source synchronous interfaces
- Third model (slow, min. temp.) available only for 65 nm and smaller technology devices (temperature inversion phenomenon)

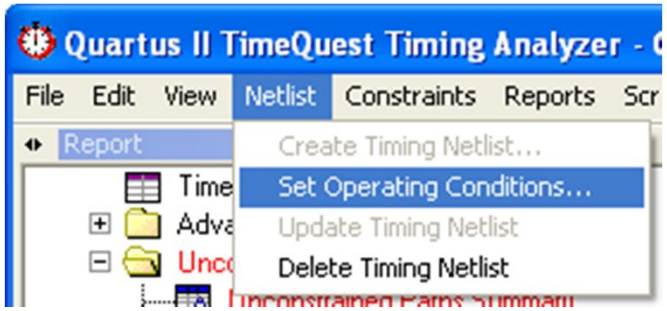
# Generating Fast/Slow Netlist

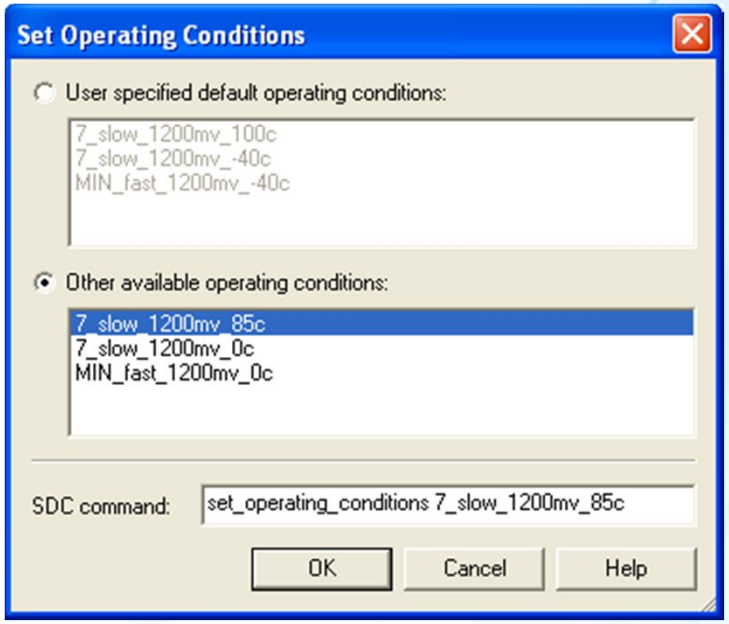
- Specify one of the default timing models to be used when creating your netlist
- Default is the slow timing netlist
- To specify fast timing netlist
  - Use `-fast_model` option with `create_timing_netlist` command
  - Choose **Fast corner** in GUI when executing **Create Timing Netlist** from **Netlist** menu
  - CANNOT select fast corner from Tasks Pane



# Specifying Operating Conditions

- Perform timing analysis for different delay models without recreating the existing timing netlist
- Takes precedence over already generated netlist
- Required for selecting slow, min. temp. model and other models (industrial, military, etc.) depending on device
- Use `get_available_operating_conditions` to see available conditions for target device



# Reference Documents

- Quartus II Handbook, Volume 3, Chapter 7 The Quartus II TimeQuest Timing Analyzer

  http://www.altera.com/literature/hb/qts/qts_qii53018.pdf
- Quick Start Tutorial

  http://www.altera.com/literature/hb/qts/ug_tq_tutorial.pdf
- Cookbook
  - http://www.altera.com/literature/manual/mnl_timequest_cookbook.pdf

# Reference Documents

- SDC and TimeQuest API Reference Manual
  - http://www.altera.com/literature/manual/mnl_sdctmq.pdf
- AN 481: Applying Multicycle Exceptions in the TimeQuest Timing Analyzer
  - http://www.altera.com/literature/an/an481.pdf
- AN 433: Constraining and Analyzing Source-Synchronous Interfaces
  - http://www.altera.com/literature/an/an433.pdf

# Learn More Through Technical Training

## Instructor-Led Training

With Altera's instructor-led training courses, you can:

- Listen to a lecture from an Altera technical training engineer (instructor)
- Complete hands-on exercises with guidance from an Altera instructor
- Ask questions & receive real-time answers from an Altera instructor
- Each instructor-led class is one or two days in length (8 working hours per day).

## Online Training

With Altera's online training courses, you can:

- Take a course at any time that is convenient for you
- Take a course from the comfort of your home or office (no need to travel as with instructor-led courses)

Each online course will take approximate one to three hours to complete.

http://www.altera.com/training

View training class schedule & register for a class

# Altera Technical Support

- Reference Quartus II software on-line help
- Quartus II Handbook
- Consult Altera applications (factory applications engineers)
  - MySupport: http://www.altera.com/mysupport
  - Hotline: (800) 800-EPLD (7:00 a.m. - 5:00 p.m. PST)
- Field applications engineers: contact your local Altera sales office
- Receive literature by mail: (888) 3-ALTERA
- FTP: ftp.altera.com
- World-wide web: http://www.altera.com
  - Use solutions to search for answers to technical problems
  - View design examples