

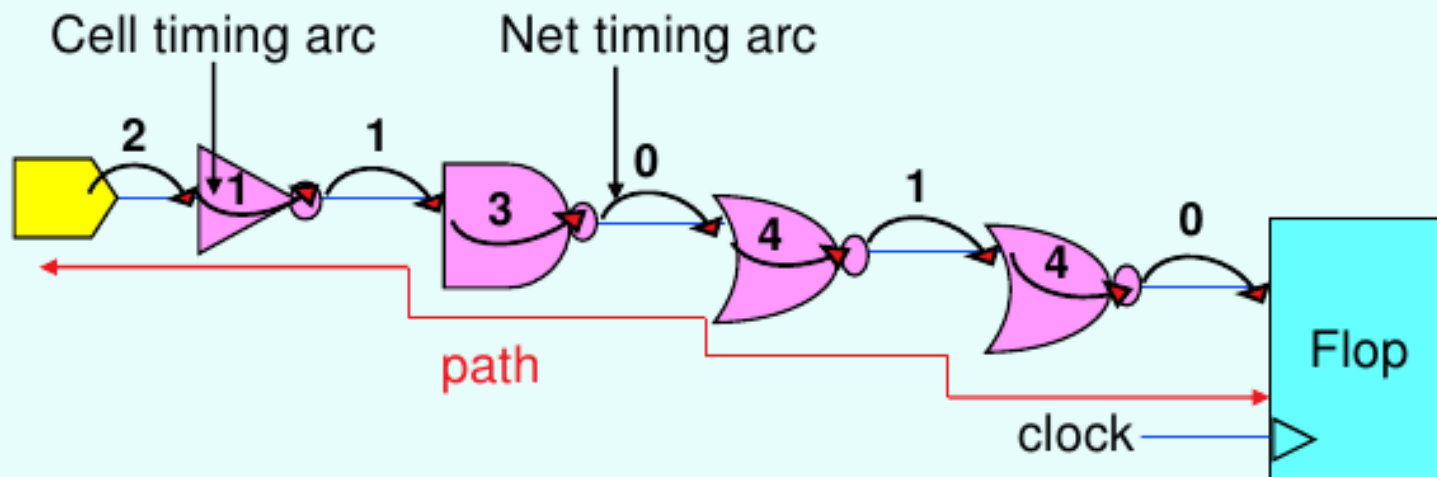
# Timing Analysis

- Introdução à Análise de Temporização
- Cell Delay
- Net Delay
- Clocks
- Checagem de temporização
- Timing Paths
- Restrições com SDC
- Restrições de temporização
- Exceções
- Relatórios de temporização

# O que é Static Timing Analysis (STA)?

Static timing analysis é processo de soma dos delays de células e das redes, obtendo assim os delays dos caminhos (path delays). Os Path Delays são então comparados à especificação de temporização.

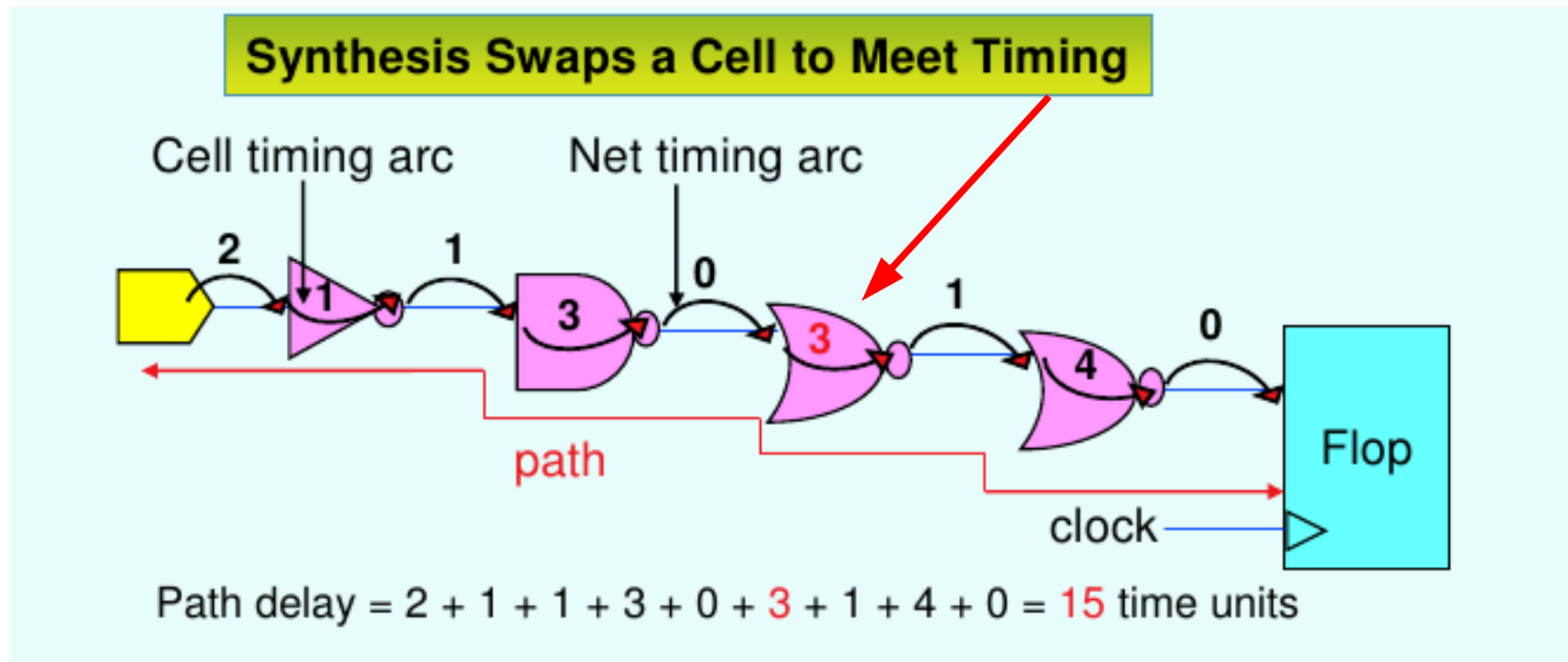
## Simple Design Showing Timing Arcs and Timing Paths



$$\text{Path delay} = 2 + 1 + 1 + 3 + 0 + 4 + 1 + 4 + 0 = 16 \text{ time units}$$

# Qual o propósito da STA?

- STA determina os path delays para síntese. Então, com base nos path delays, a ferramenta de síntese escolhe quais células utilizar para criar um circuito que atenda aos requisitos de temporização.
- STA analisa a temporização do circuito para verificar se trabalha na frequência especificada.

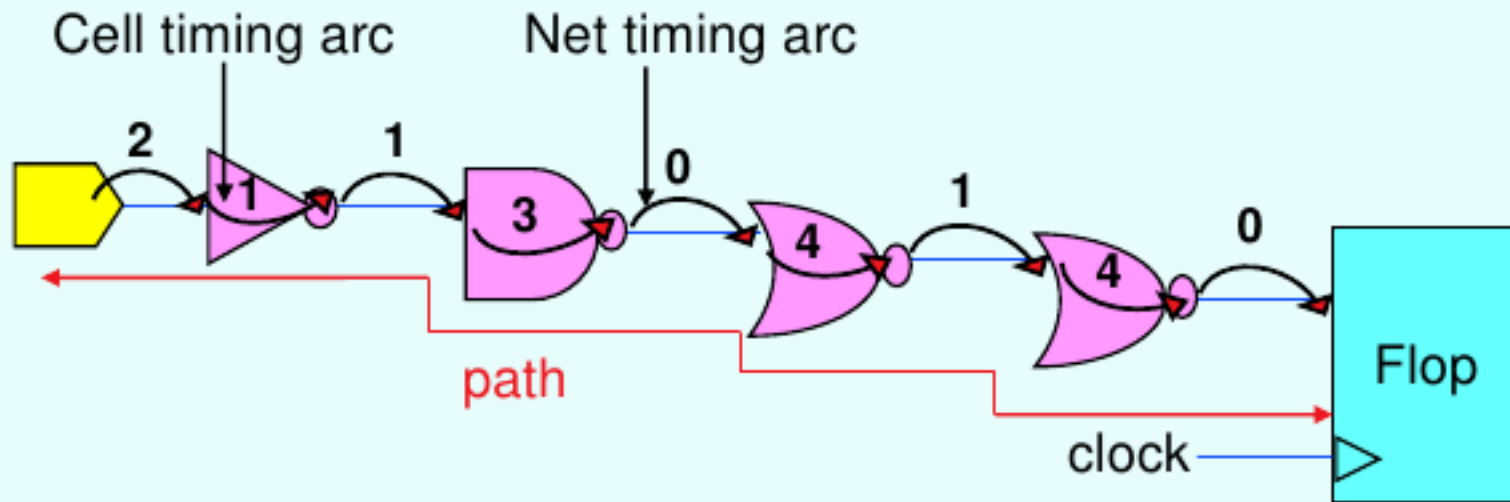


# Elementos de Static Timing

A ferramenta de STA calcula os delays dos caminhos, que consistem de dois elementos básicos:

- Arcos de tempo em células
- Arcos de tempo em redes

## A Timing Path Showing Timing Arcs

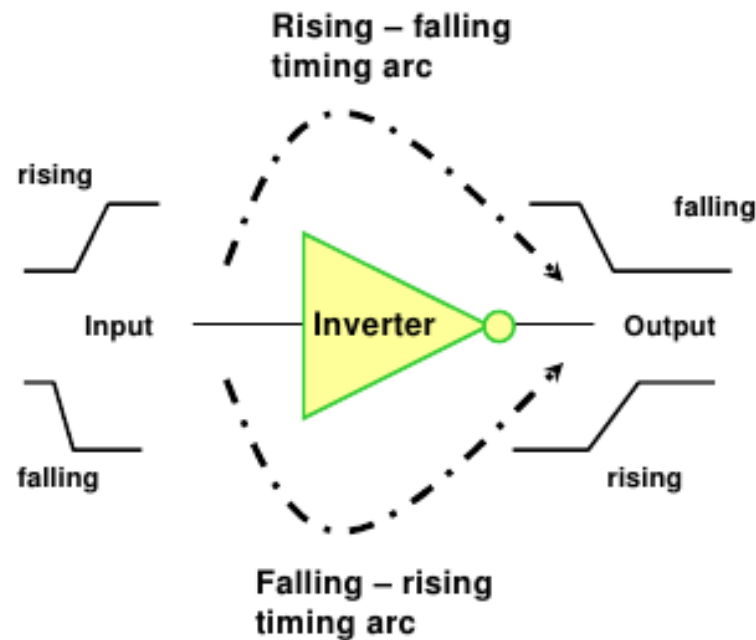


$$\text{Path delay} = 2 + 1 + 1 + 3 + 0 + 4 + 1 + 4 + 0 = 16 \text{ time units}$$

# O que são Arcos de Tempo?

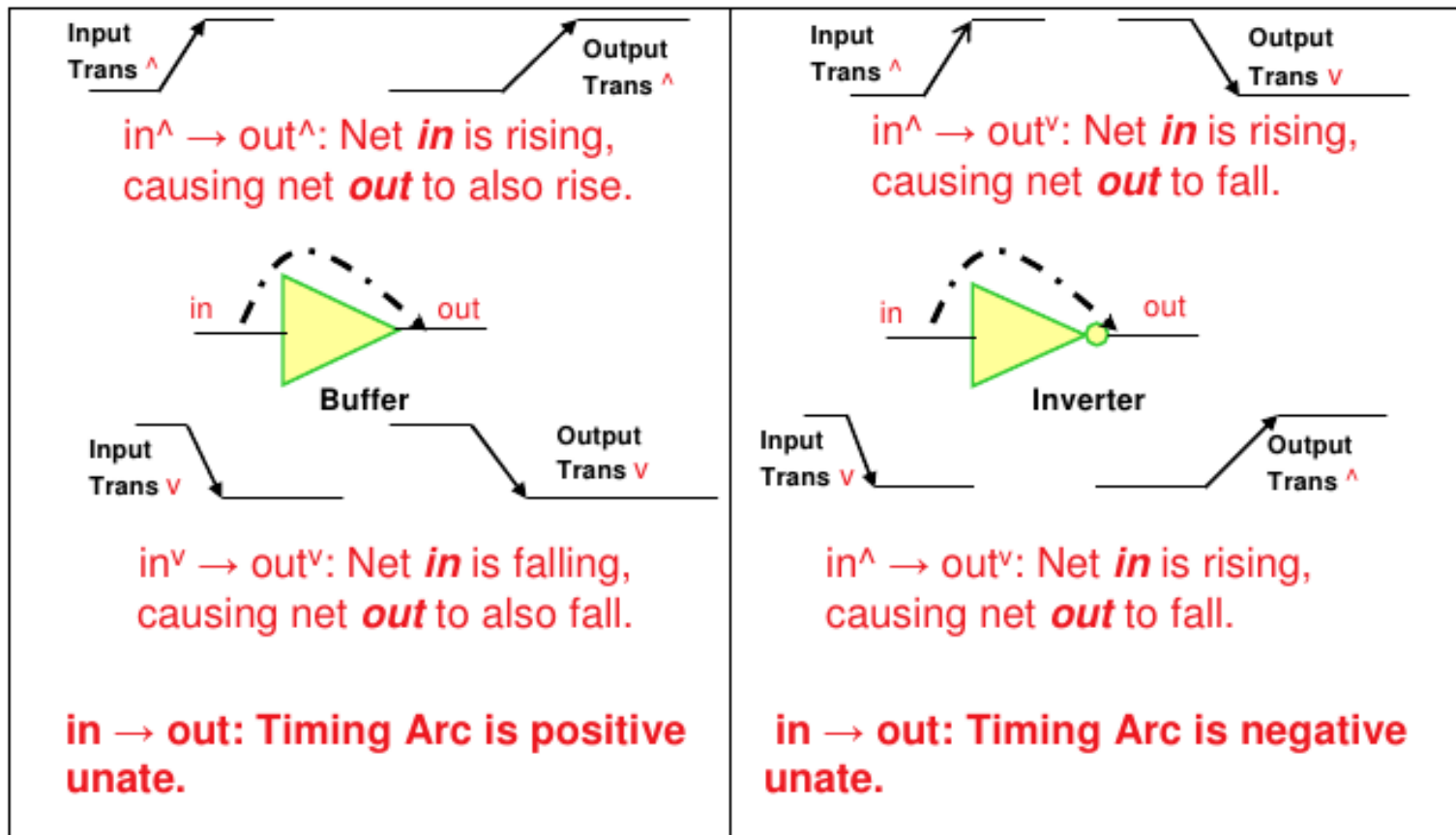
Um Arco de Tempo (Timing Arc) é um arco imaginário que representa uma relação causal simples. Se uma mudança na entrada causa uma mudança na saída, isso é conhecido como relação causal.

- Características de um Timing Arc:
  - Unateness
  - Slew
  - Delay



# Timing Arc: Unateness

Se uma variação na entrada (subida ou descida) causa a mesma variação na saída (subida ou descida), o Unateness é dito positivo. Caso uma variação na entrada (subida ou descida) cause uma variação oposta na saída (descida ou subida), o Unateness é dito negativo.



# Timing Arc: Transições de sinal

Tempo de transição é o tempo que um sinal leva para mudar seu estado e alto para baixo ou de baixo para alto.

- Transições de subida e descida são propriedades do Timing Arc, estando presentes tanto no sinal de entrada quanto no de saída.

A ferramenta de STA mede o tempo de transição usando limiares de tensão.

- Slew rate é a taxa de variação na transição.

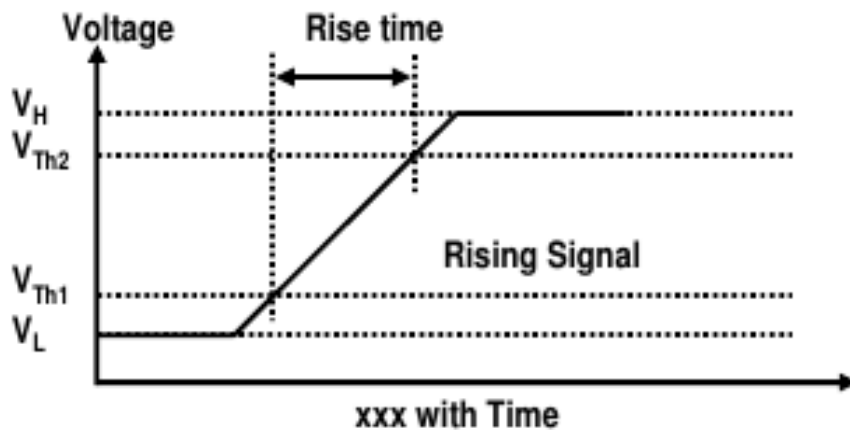
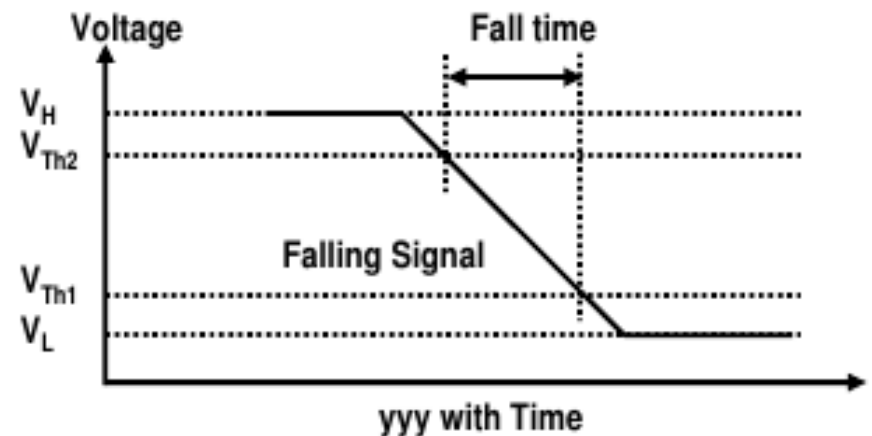


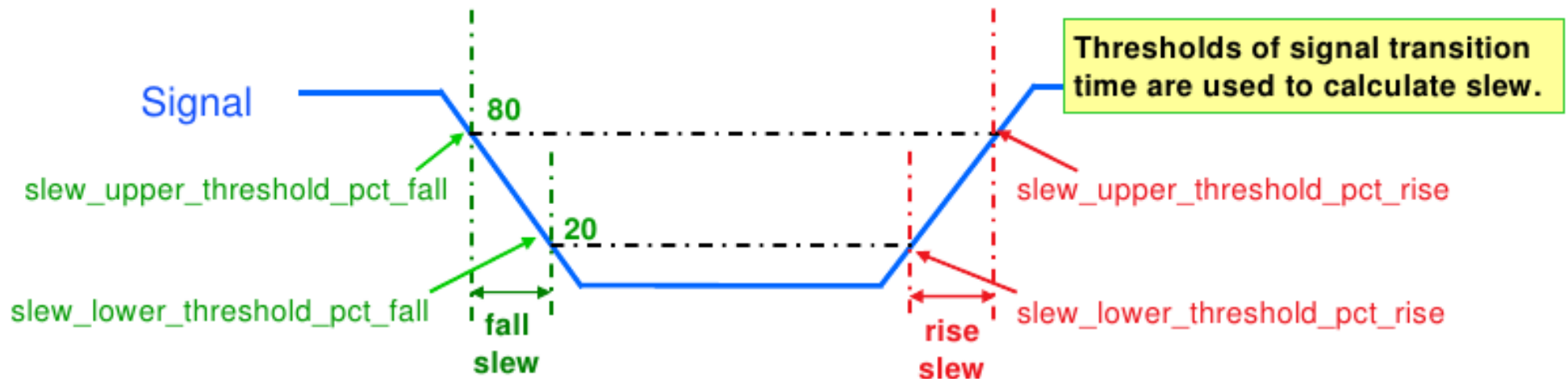
Fig. 1



# Transições: Slew de subida e de descida

As ferramentas de STA usam o limiar de tensão para calcular os tempos de transição.

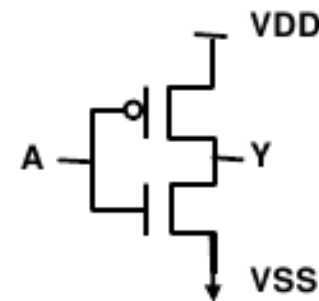
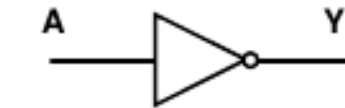
- Exemplo: A ferramenta calcula o tempo de subida utilizando o limiar superior em 90% e o inferior em 10%. Tais valores dependem da biblioteca de células do fabricante.





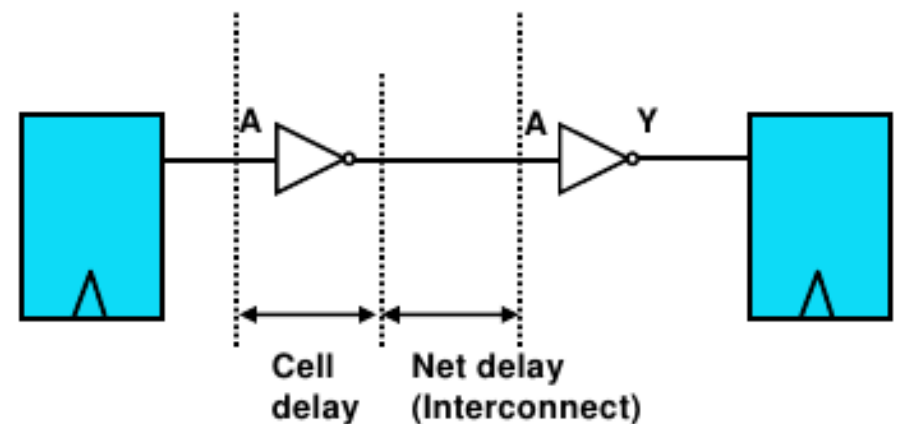
# Timing Arcs: delays de células e de rede

Cell delay: Os transistores de uma determinada célula levam um certo tempo para o chaveamento. Ou seja, uma variação na entrada leva algum tempo para ser mostrada na saída.



Transistor Representation

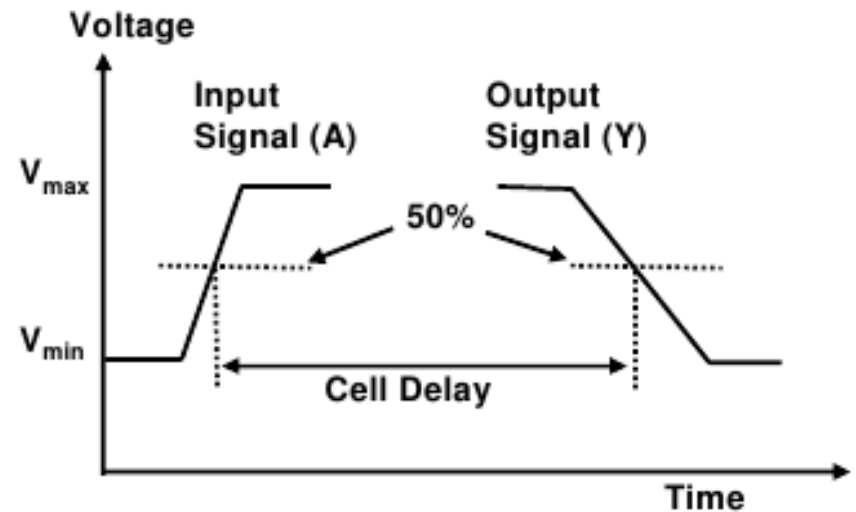
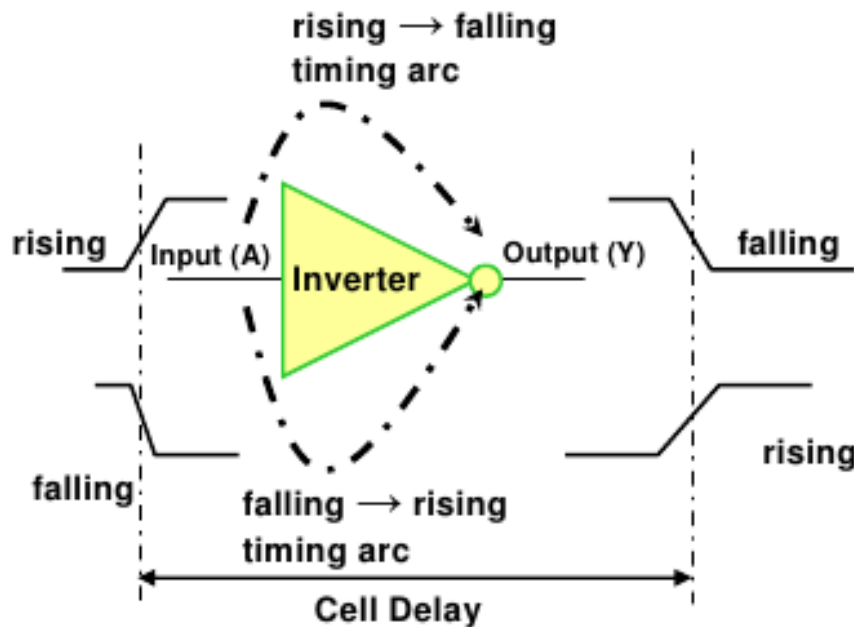
Net delay: É o intervalo de tempo entre o instante em que o sinal é aplicado à rede e instante em que ele chega ao outro elemento da rede.



# Cell delay

Cada célula tem um ou mais timing arcs, e todo timing arc tem um atraso de propagação, também conhecido como Cell Delay. O Cell delay é determinado por:

- O delay intrínseco dos timing arcs
- A carga que está sendo ativada posteriormente
- A transição de entrada (input slew)

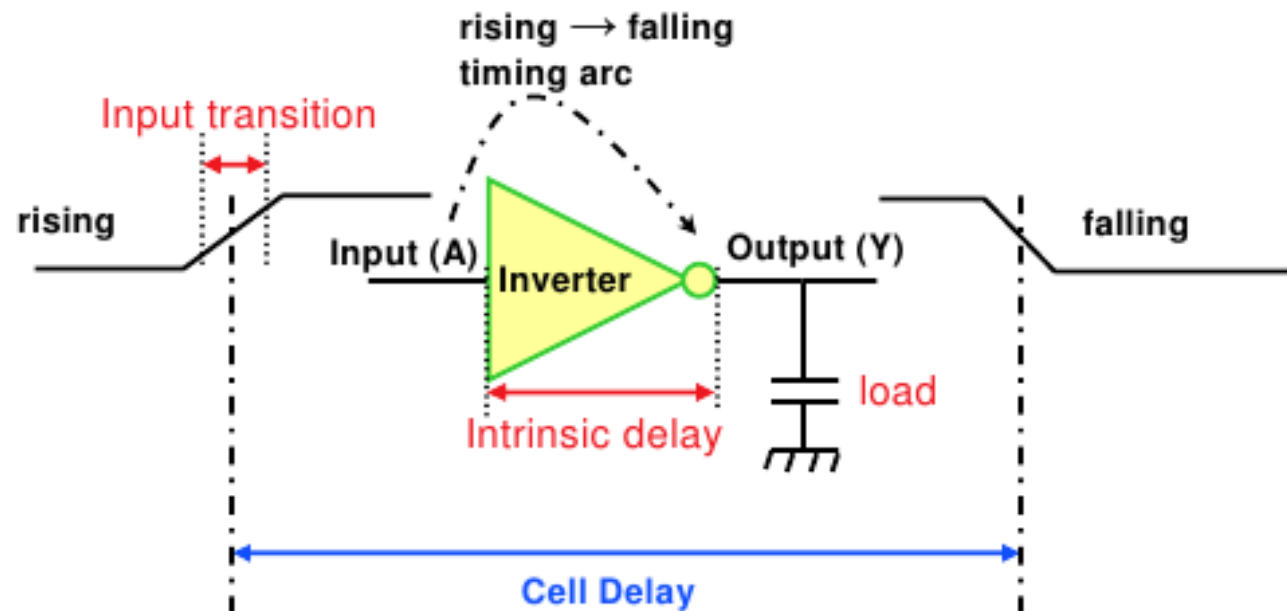


In this waveform, the cell delay is measured from 50% of the input signal to 50% of the output.

# Cell delay

## Tipos de Cell Delay:

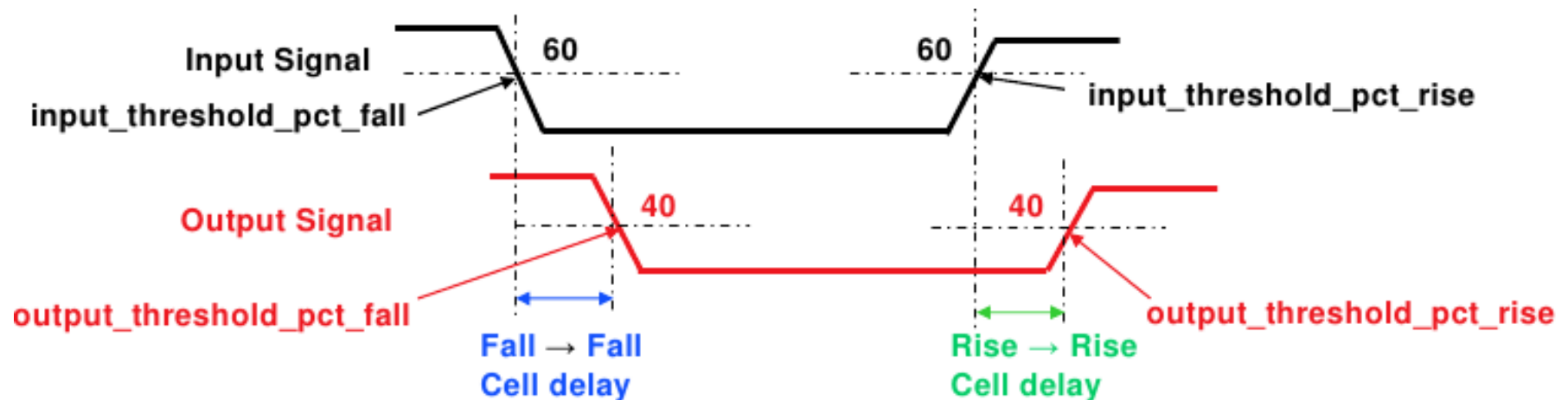
- Transition delay: tempo que o pino leva para mudar seu estado de alto/baixo para baixo/alto.
- Intrinsic delay: definido como o atraso em uma célula considerando um sinal com transition delay zero aplicado a sua entrada, e sem nenhuma carga em sua saída.
- Propagation delay: é o intervalo de tempo entre o instante em que a entrada muda de estado e o instante em que a saída muda de estado.



# Cell delay: limiares de entrada e saída

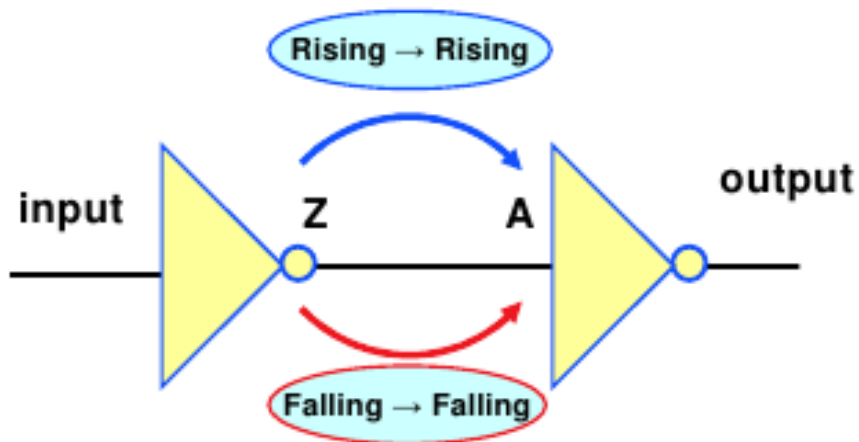
Por padrão, ferramentas de STA medem o cell delay a partir da variação de 50% do sinal de entrada até a variação de 50% do sinal de saída.

No exemplo abaixo, os delays de subida e descida são medidos de 60% da entrada a 40% da saída.

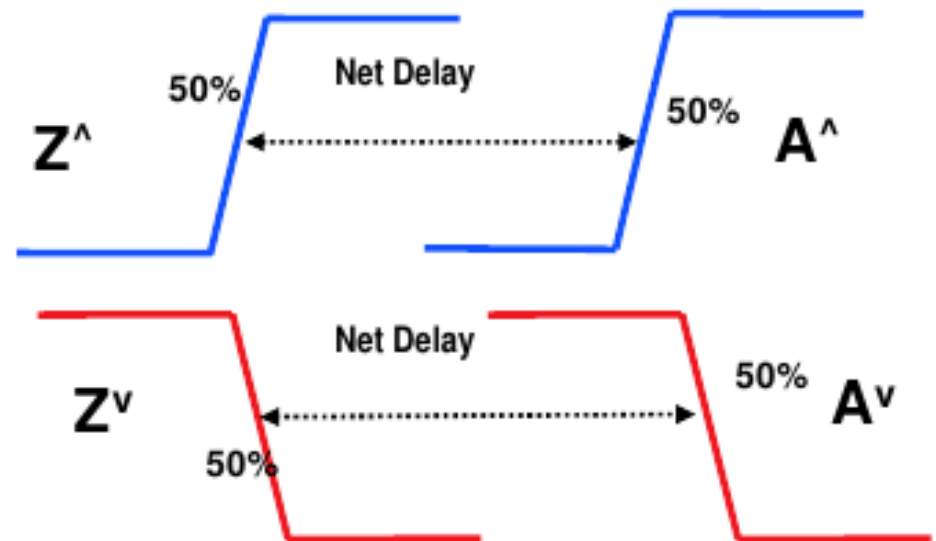


# Net delay

As interconexões são os timing arcs de um pino para outro e possuem atrasos devido a capacitâncias e resistências da interconexão. O delay de interconexão (interconnect delay) também é chamado de wire delay ou net delay.



**All interconnect arcs are positively unate.**

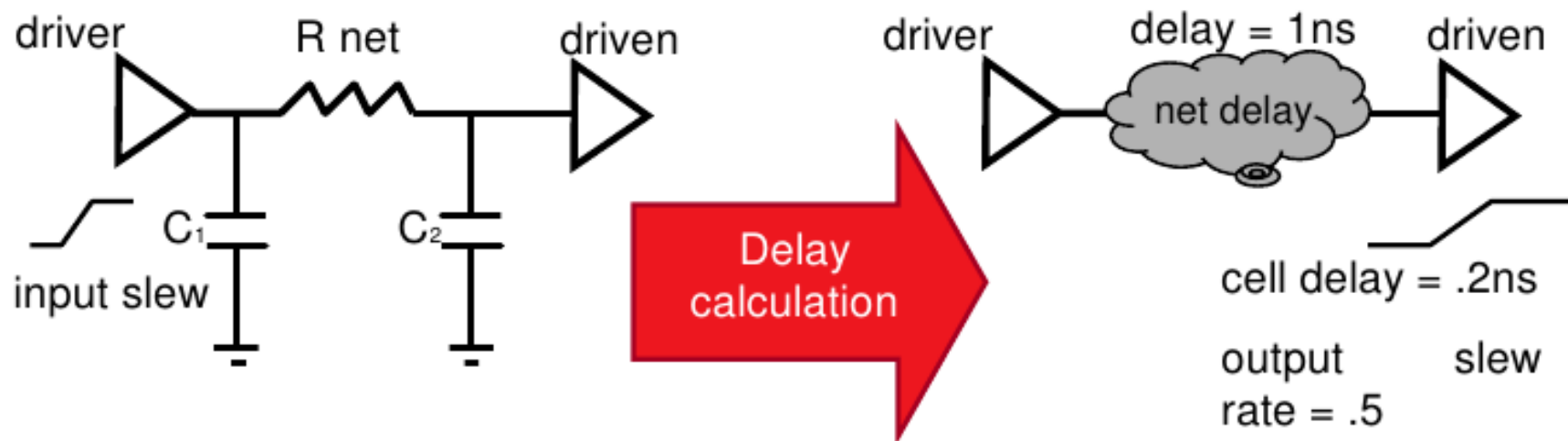


**All net delays are usually calculated from 50% of the input to 50% of output.**

# Net delay

O net delay depende de diversos parâmetros:

- Capacitância da rede
- Resistência da rede
- Fan-out : quantidade de elementos ligados à saída

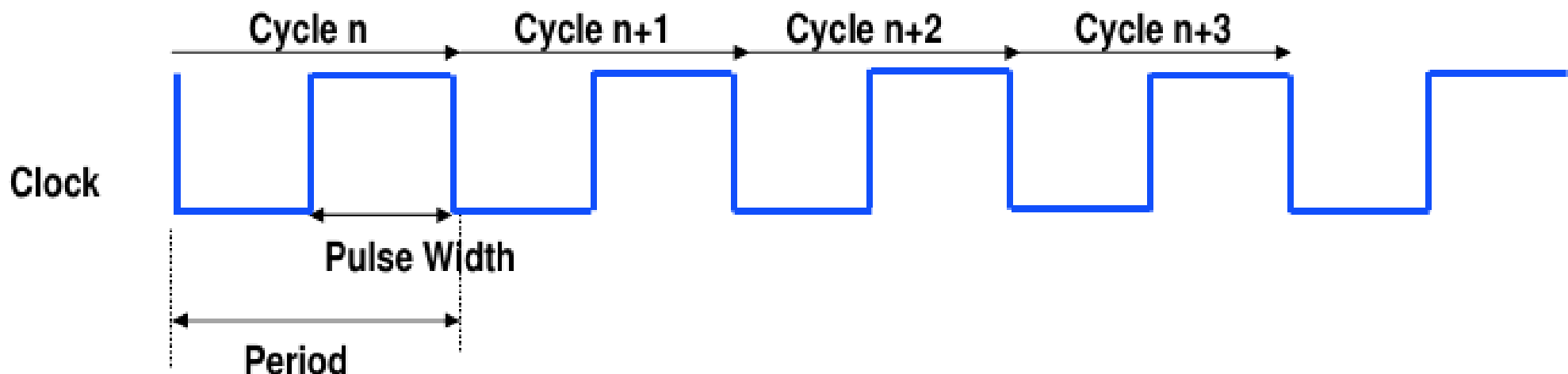


Formats: SDF

# O que é um Clock?

Após calcular os path delays, a ferramenta de STA determina os atrasos relativos ao clock, de modo que cada path possua um atraso máximo de 1 período de clock.

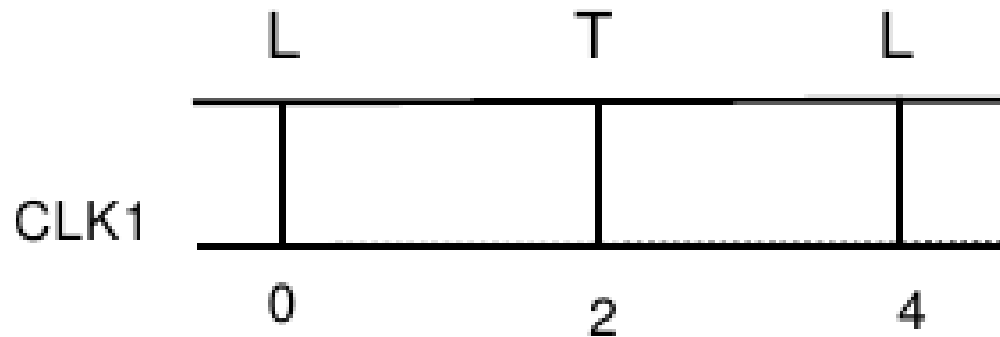
- O clock é um sinal periódico espalhado em um sistema seqüencial para sincronizar a transferência de dados entre elementos de armazenamento (flops).
- Clocks são gerados por cristais osciladores fora do chip ou por um Phase-Locked Loop (PLL) interno ao chip.



# Clock Ideal

Um clock ideal possui chaveamento instantâneo.

- Leading transition indica a primeira transição do ciclo (subida ou descida)
- Trailing transition indica a segunda transição do ciclo, oposta à primeira (descida ou subida)



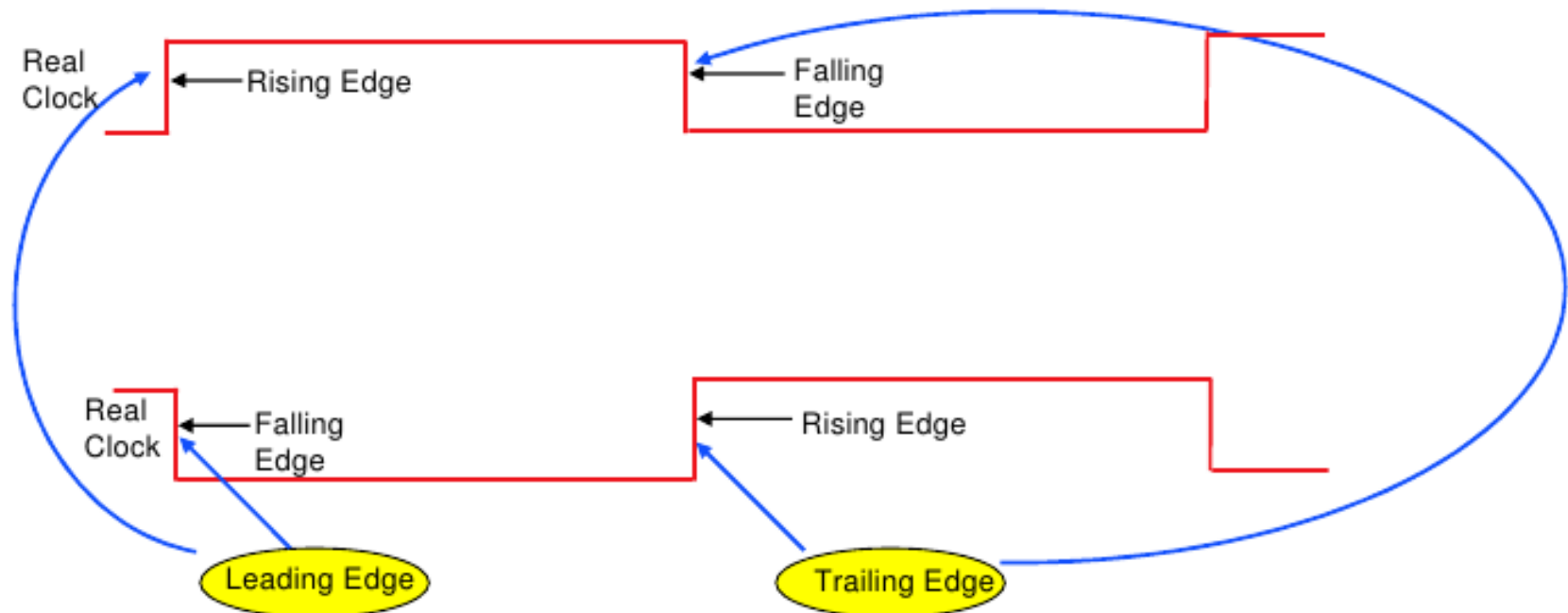
L - Leading edge  
T - Trailing edge



# Polaridade do Clock

A polaridade do sinal de clock é definida dependendo se a Leading Transition é de subida ou descida.

- O clock é positivo quando a borda de subida está associada à Leading Transition.

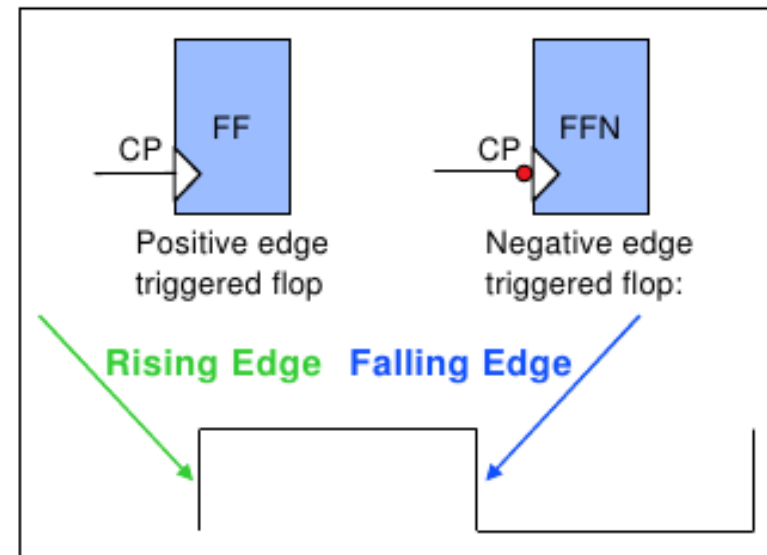
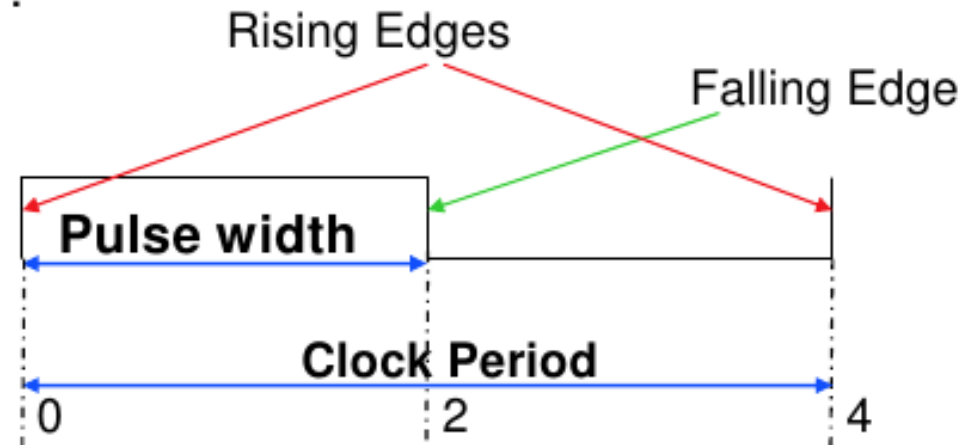


- O clock é negativo quando a borda de descida está associada à Leading Transition.

# Características do Clock

As características essenciais do Clock são:

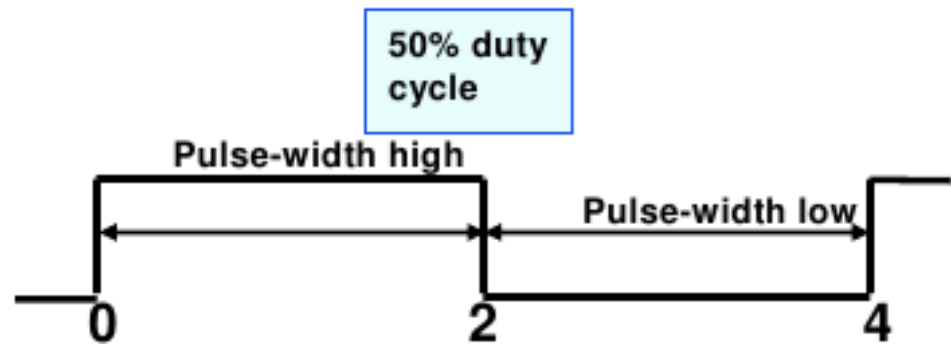
- Período: descreve o intervalo de repetição da forma de onda.
- Bordas de subida e descida.
- Largura de pulso: porcentagem do período em que o clock está no estado alto.



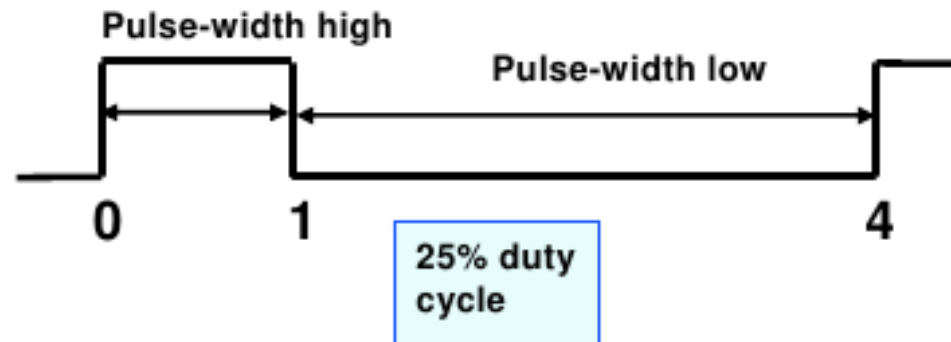
# Clock Duty Cycle

Duty Cycle é definido como a razão entre o tempo em estado alto e o período da forma de onda.

- Por exemplo, o clock da figura superior está em estado alto do instante 0 ns ao instante 2 ns, de um total de 4 ns.  $\Rightarrow 2/4 = 50\%$

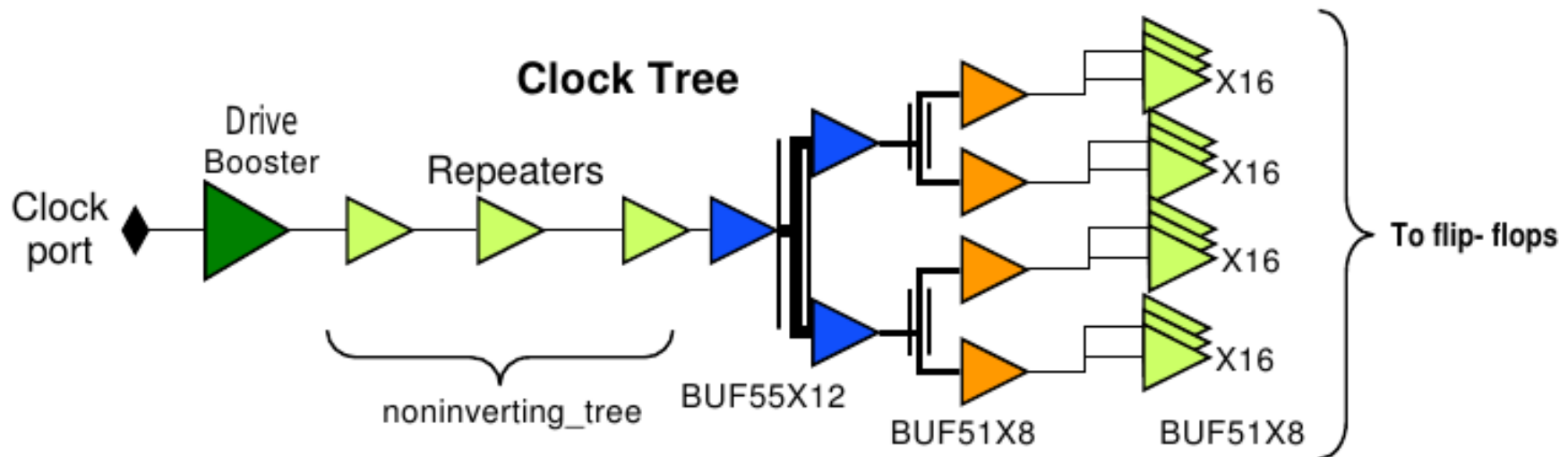


- O clock da figura inferior está em estado alto do instante 0 ns ao instante 1 ns, de um total de 4 ns.  $\Rightarrow 1/4 = 25\%$



# Propagação do Clock

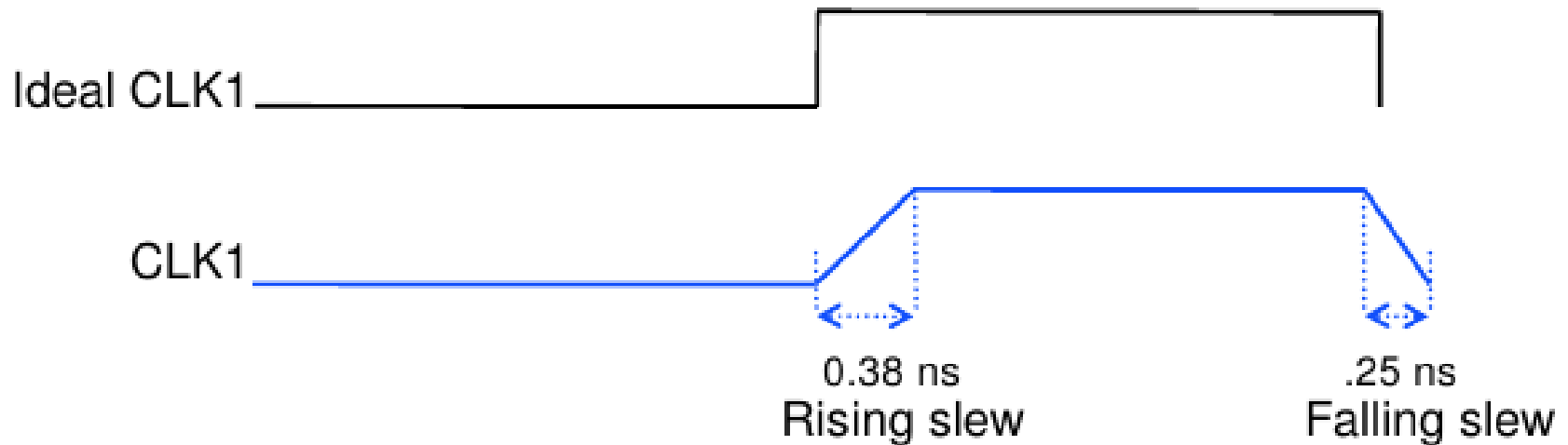
O sinal de clock tem de percorrer todo o circuito para alcançar diversos flops e latches. Para corrigir a degradação do clock, o sinal é bufferizado para manter sua integridade. Geralmente os buffers são organizados em forma de árvores binárias.



- Ideal Mode: análise feita sem a presença de clock buffers, mas considerando o atraso total da rede de clock.
- Propagated Mode: análise feita com o objetivo de inserirem-se os buffers de acordo com a análise em Ideal Mode.

# Clock slew

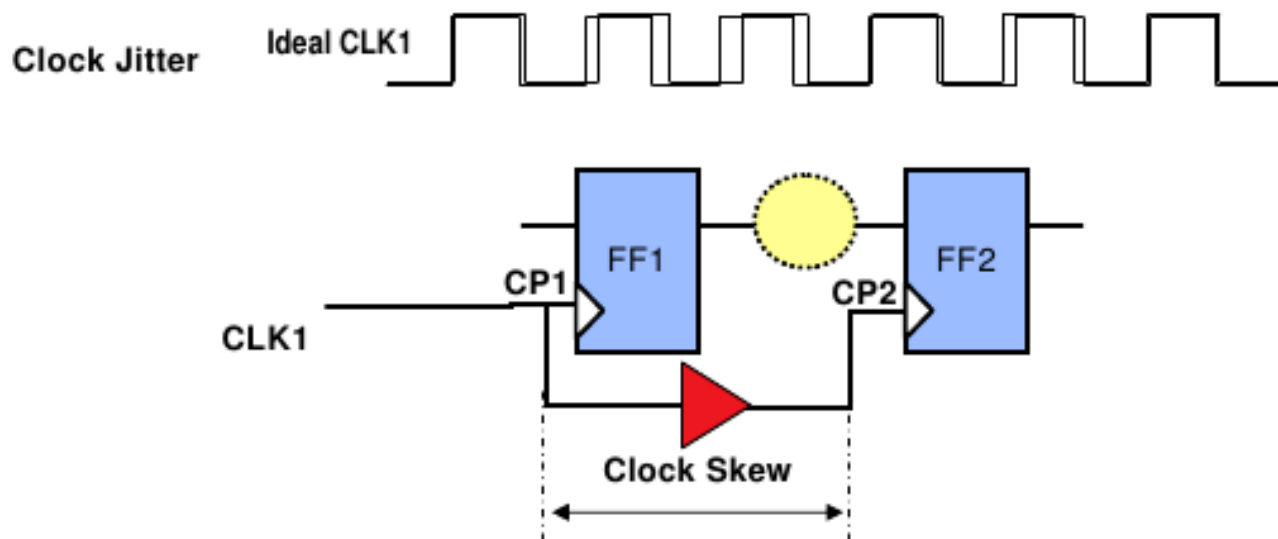
Clock Slew é o tempo necessário para a transição do sinal de clock do valor alto para o baixo e vice-versa.



# Clock Uncertainty

Entre um ciclo e outro, o período e o duty-cycle pode mudar levemente devido ao circuito de geração do clock. Esse fenômeno é conhecido como **clock jitter**.

O sinal de clock chega em componentes diferentes em instantes de tempo diferentes. A diferença entre os instantes de chegada é conhecida como **clock skew**.



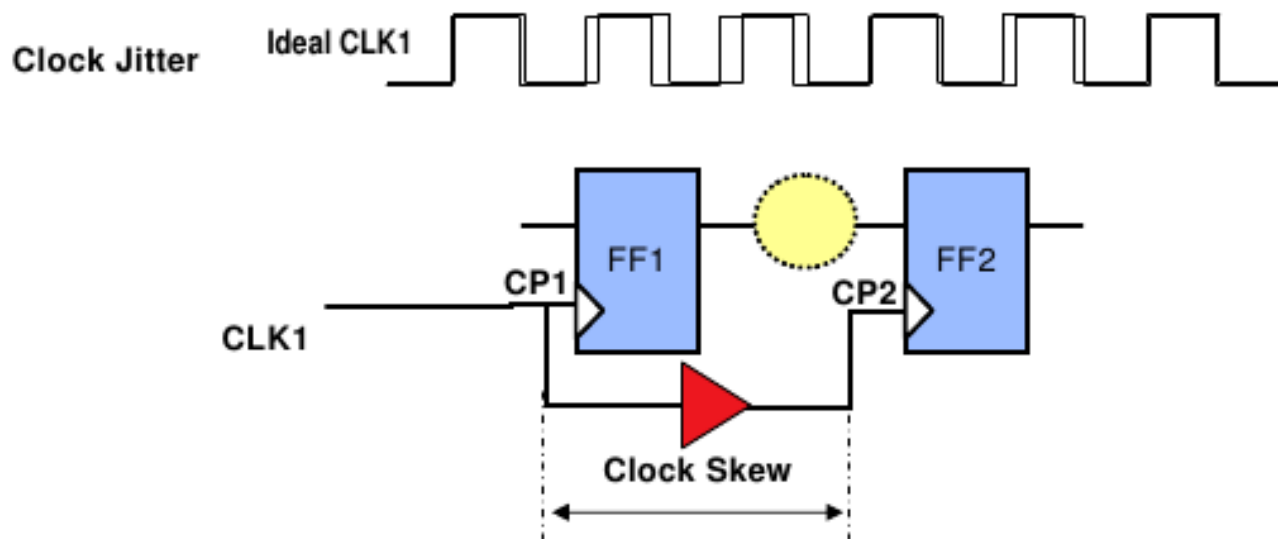
Clock Uncertainty = Clock jitter + Worst clock skew (For setup)

Clock Uncertainty = 0 + Best Clock Skew (For Hold)

# Clock Uncertainty

Entre um ciclo e outro, o período e o duty-cycle pode mudar levemente devido ao circuito de geração do clock. Esse fenômeno é conhecido como **clock jitter**.

O sinal de clock chega em componentes diferentes em instantes de tempo diferentes. A diferença entre os instantes de chegada é conhecida como **clock skew**.



Clock Uncertainty = Clock jitter + Worst clock skew (For setup)

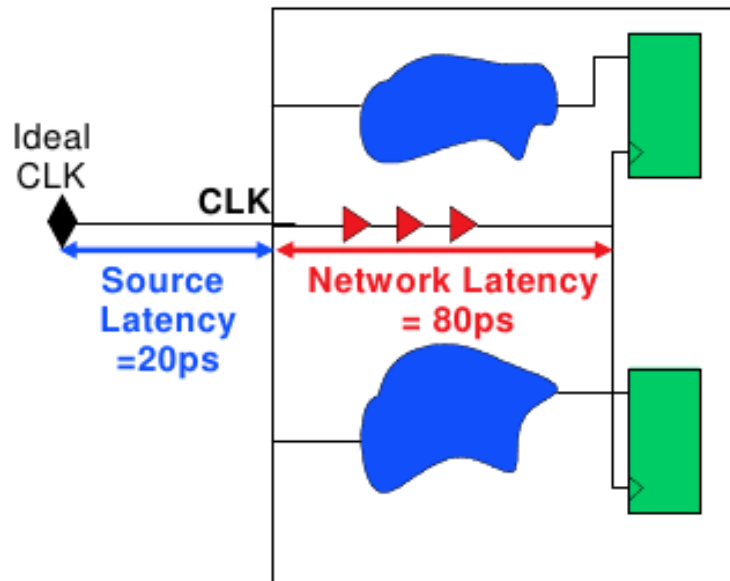
Clock Uncertainty = 0 + Best Clock Skew (For Hold)

# Latência do Clock

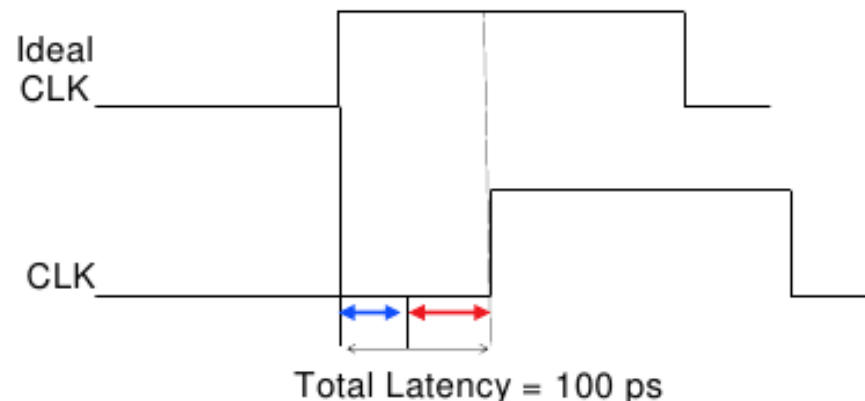
Latência do clock é o tempo que o sinal leva para propagar-se do ponto de definição ao pino de clock de um registrador. Também é conhecida como Insertion Delay.

Existem dois tipos de latência:

- Source Latency: da fonte de clock à porta de clock do projeto.
- Network Latency: da porta de clock do projeto aos registradores (clock tree delay)



Total Latency @ clock port/pin =  
Clock source latency + Clock  
network latency

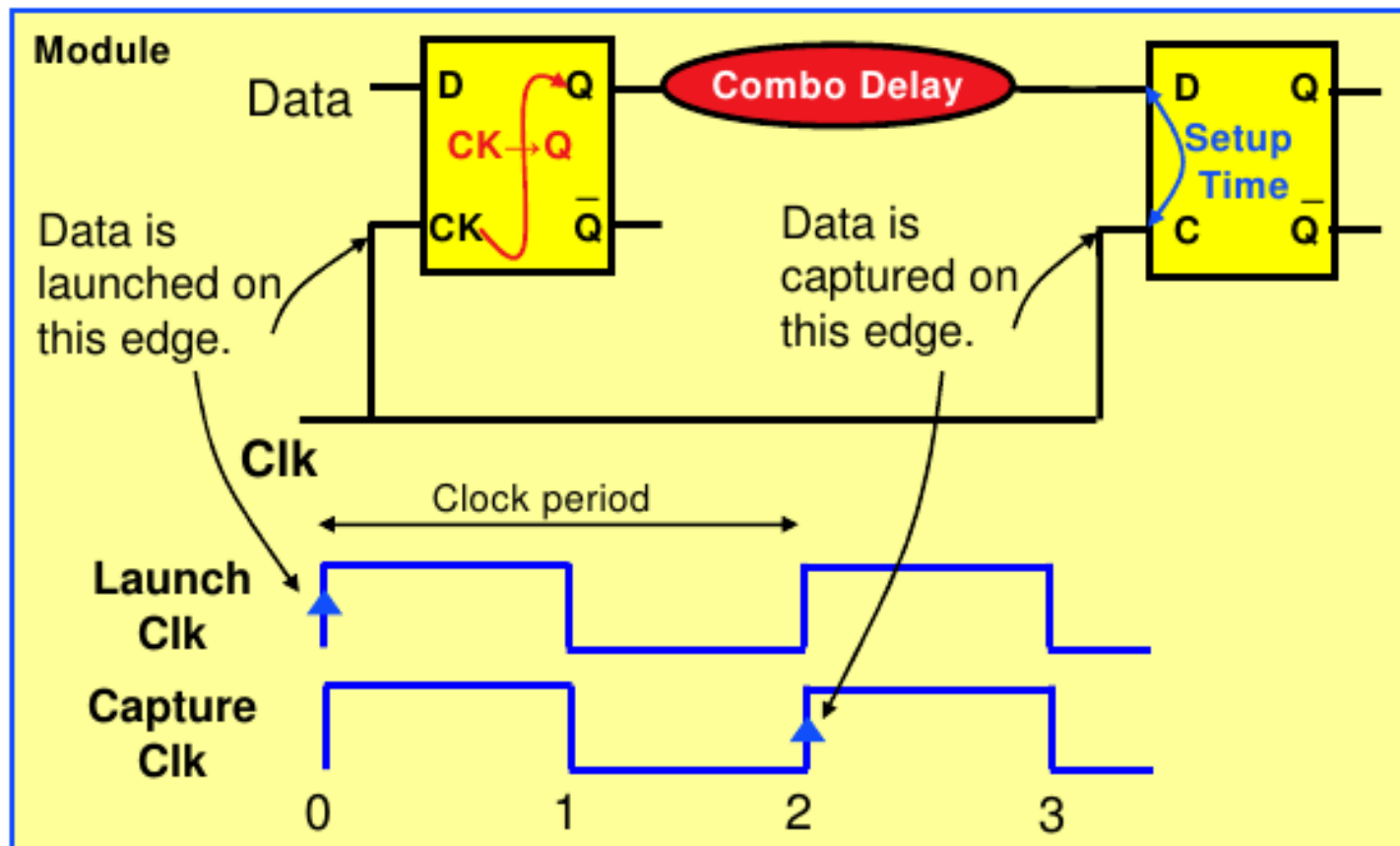




# Launch and Capture Clock Edges

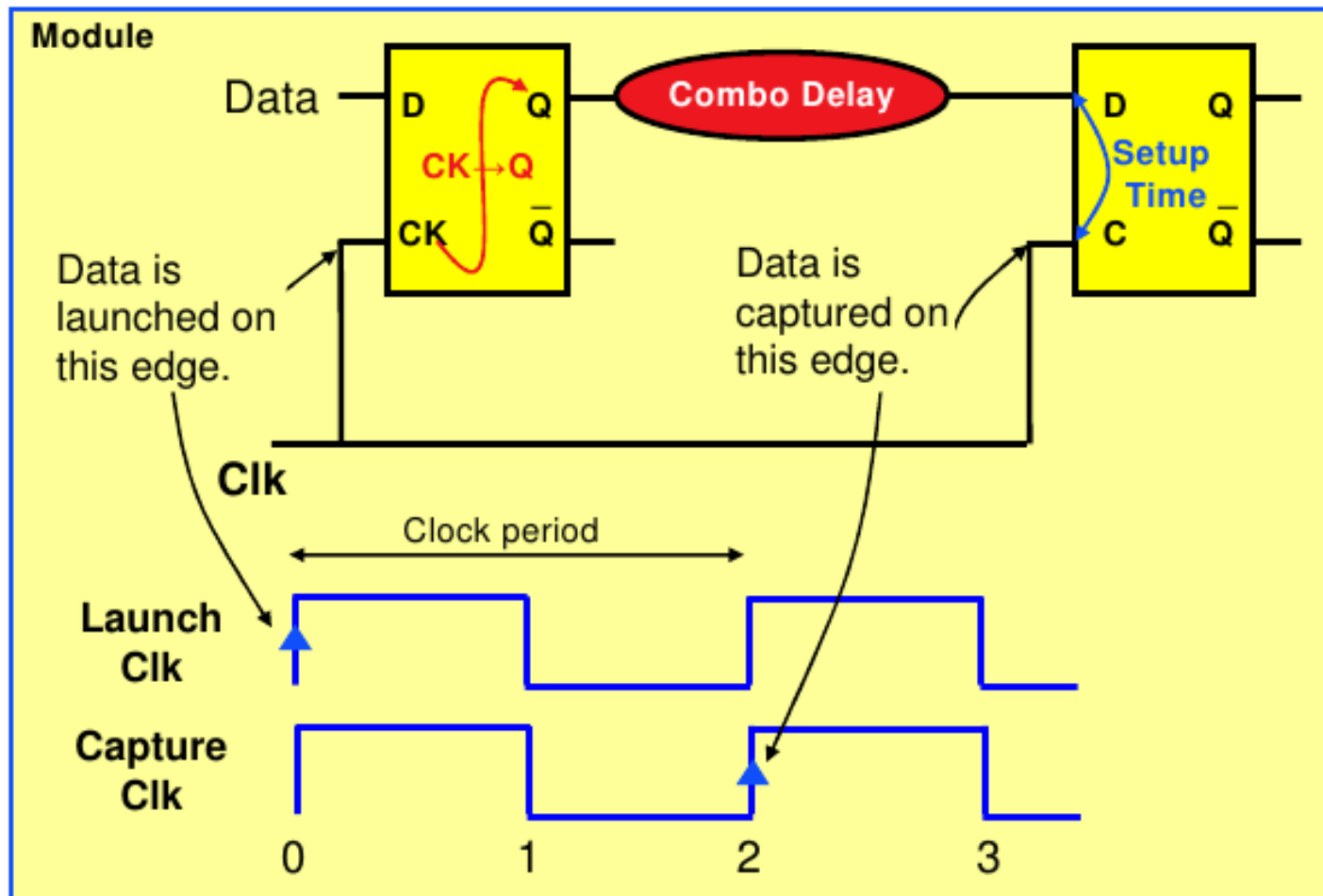
Launch clock é o sinal de “envio” dos dados. Launch edge é a borda do launch clock.

Capture clock é o sinal de “carregamento” dos dados. Capture edge é a borda do capture clock.



# Phase Shift

Phase shift é a menor diferença de tempo entre a borda de captura e a borda de disparo do clock.



# Flip-Flops

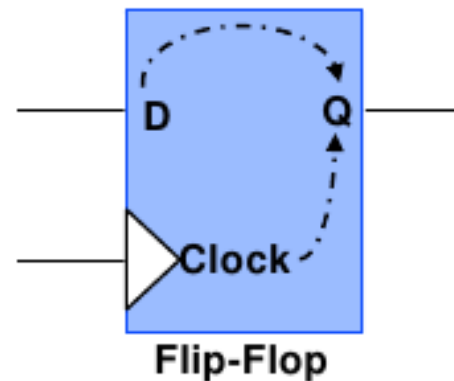
O flip-flop é um dos mais comuns elementos de armazenamento de dados. Também é referenciado por registrador.

Existem dois timing arcs básicos em um flip-flop:

- D para Q
- Clock para Q

Alguns parâmetros importantes para um flop são:

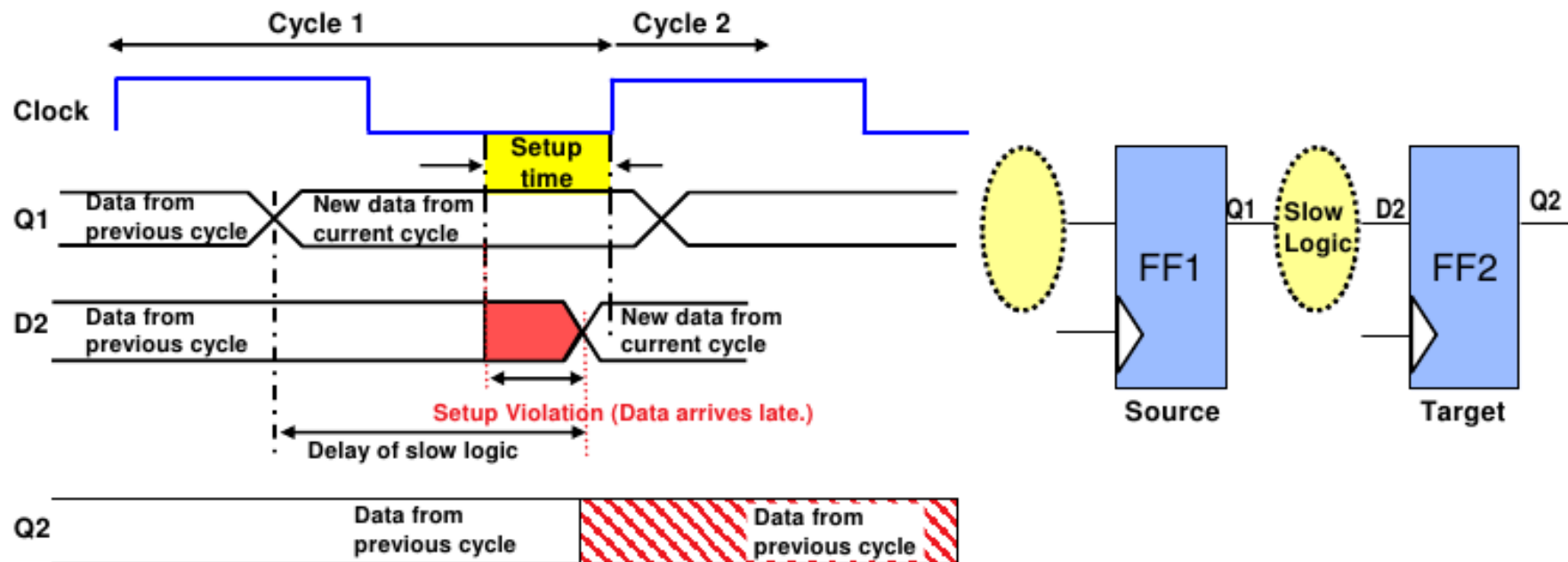
- Clock-to-Q delay
- Setup time
- Hold time



# Setup time

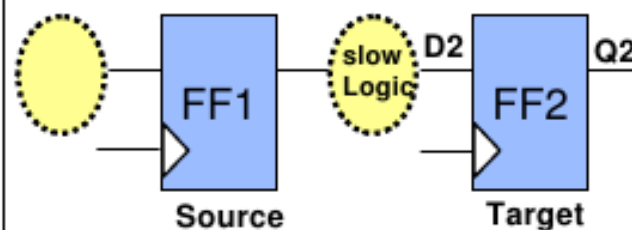
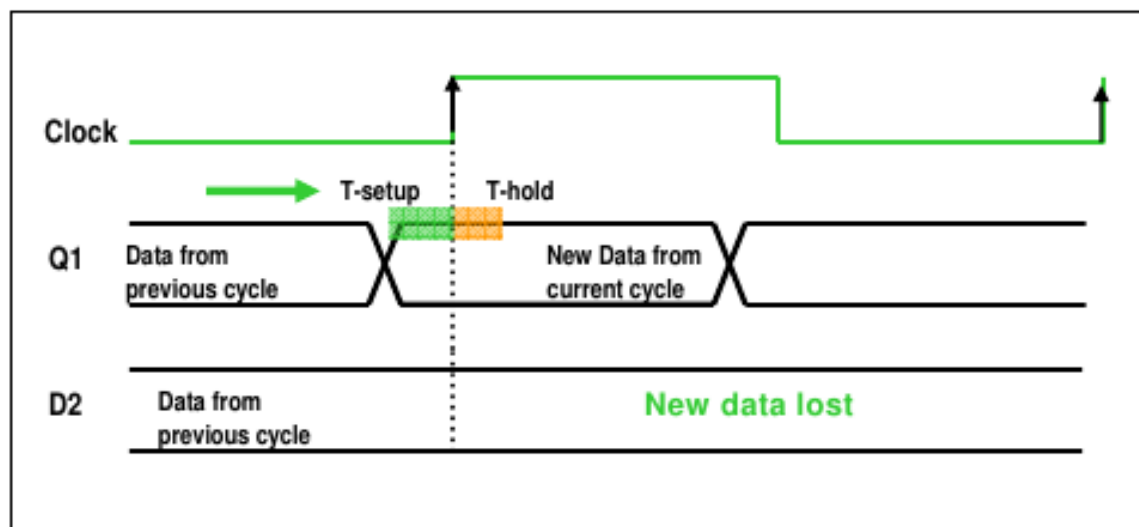
Setup time é o intervalo de tempo durante o qual o dado deve estar estável antes da borda de lançamento do clock.

- Se o Setup Time não é atendido, o dado não é armazenado no flop.
- Violações de Setup podem ser remediadas pela diminuição do clock (aumento do período) ou diminuindo o delay do data path.



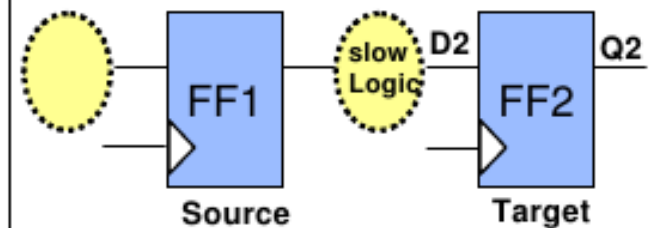
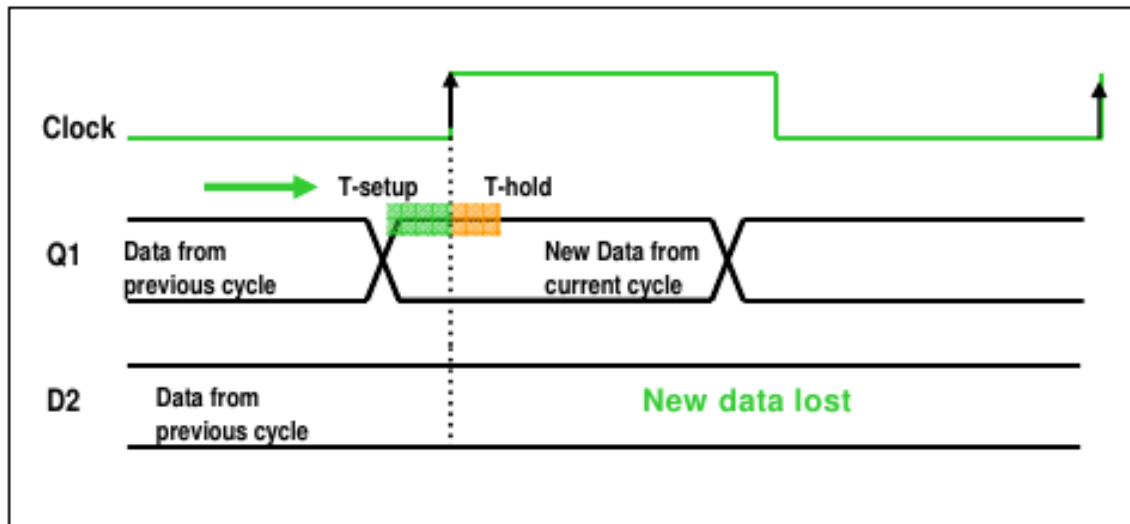
# Violações de Setup: Slow Data

O que acontece quando o dado D2 demora para chegar (slow data)?



# Violações de Setup: Slow Data

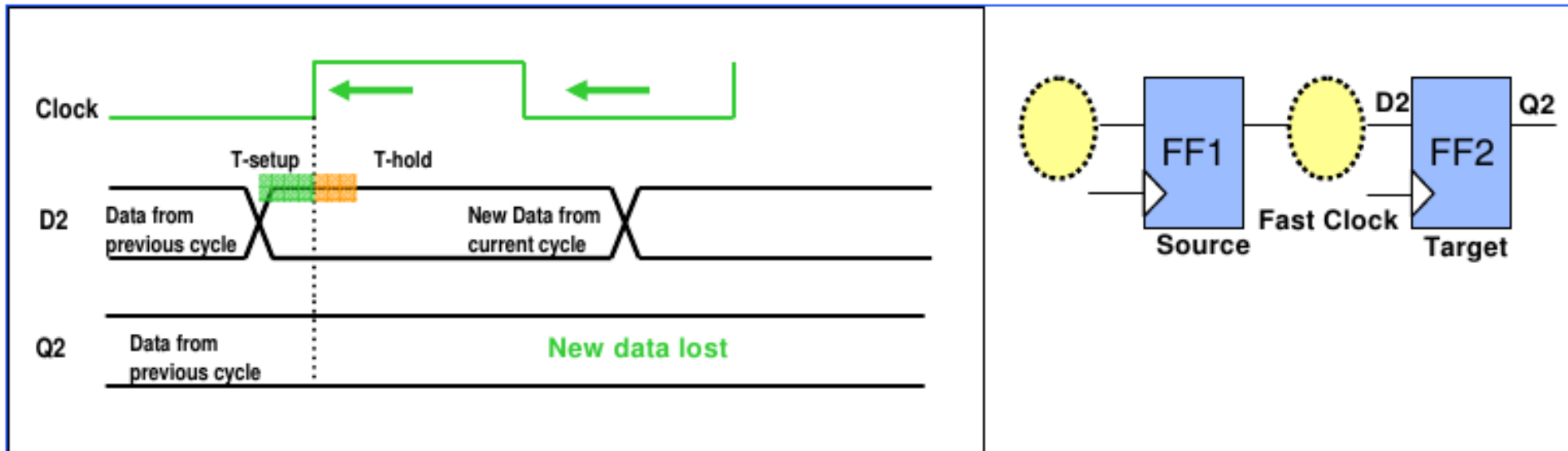
O que acontece quando o dado D2 demora para chegar (slow data)?



- Setup time é violado.
- O novo dado é perdido.

# Violações de Setup: Fast Clock

O que acontece quando o clock é acelerado?

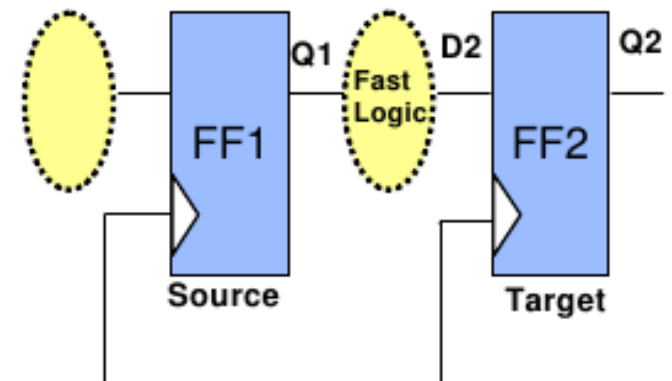
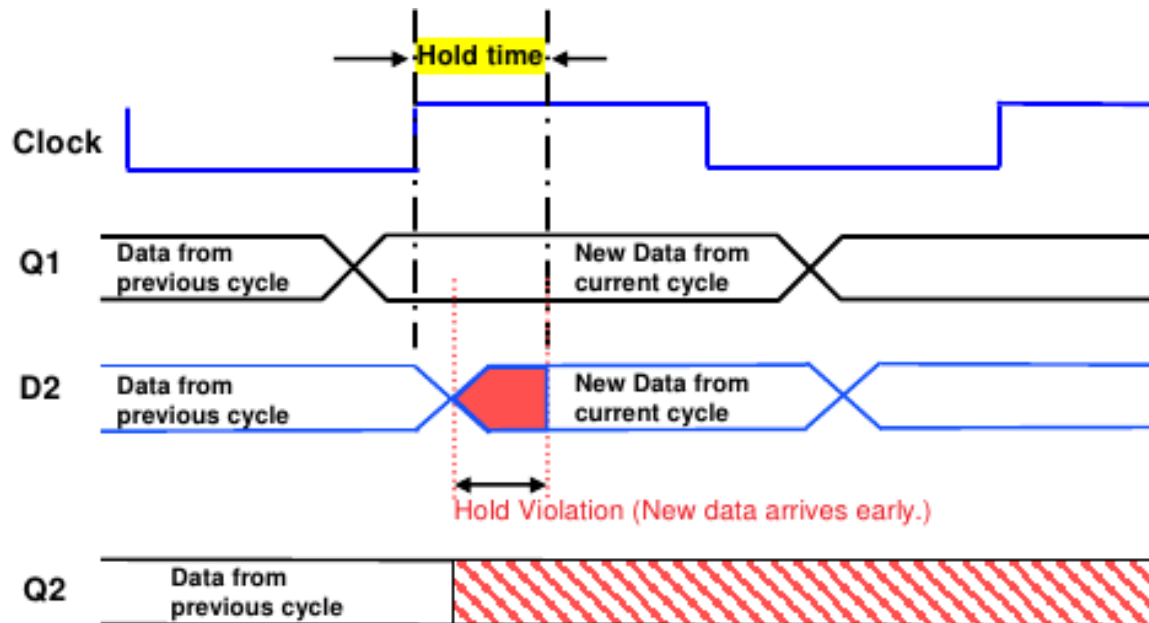


- O Setup Time é violado.
- Novos dados são perdidos.

# Hold Times

Hold time é o intervalo de tempo durante o qual a entrada síncrona D deve estar estável antes da borda de lançamento do clock.

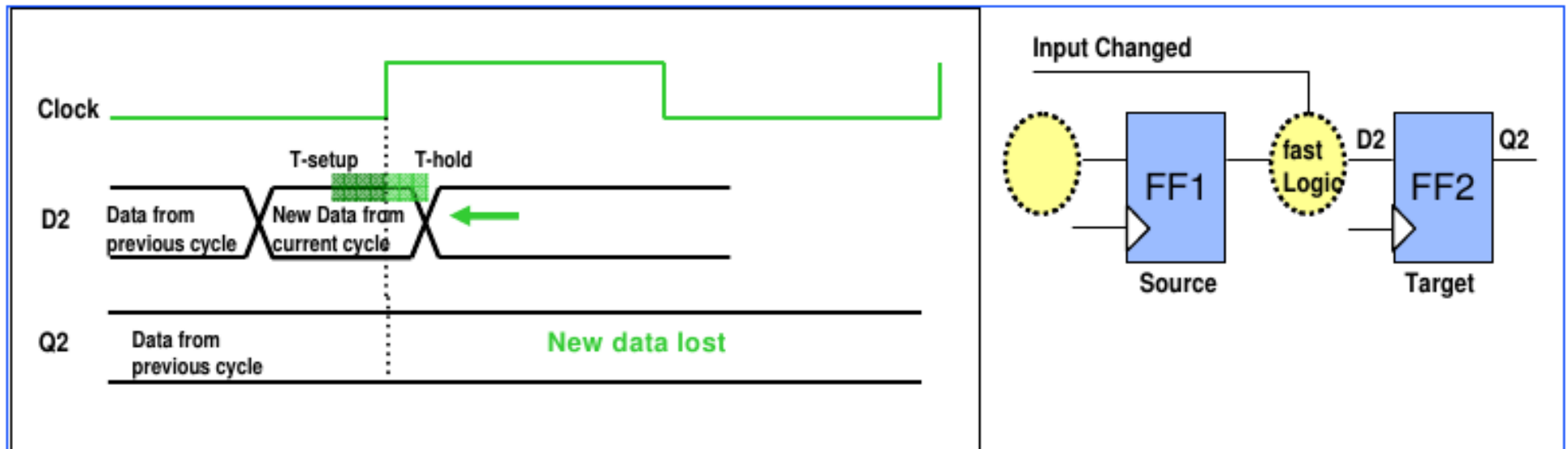
O atraso do dado deve ser maior que o hold time do target, adicionado a uma incerteza do clock entre os flops, de modo que o dado possa ser armazenado com segurança.





# Hold Time Violations: Fast Data Change

O que acontece quando o dado D2 começa a mudar imediatamente?



- Hold time é violado.
- O novo dado é perdido mesmo sendo capturado pela borda do clock.

# Exemplo

O período do clock de um projeto é 100 ps. O mínimo tempo de setup para todos os flops é de 150 ps.

O projeto constantemente não atinge os requisitos de temporização durante a STA. Qual parece ser o problema?

# Exemplo

O período do clock de um projeto é 100 ps. O mínimo tempo de setup para todos os flops é de 150 ps.

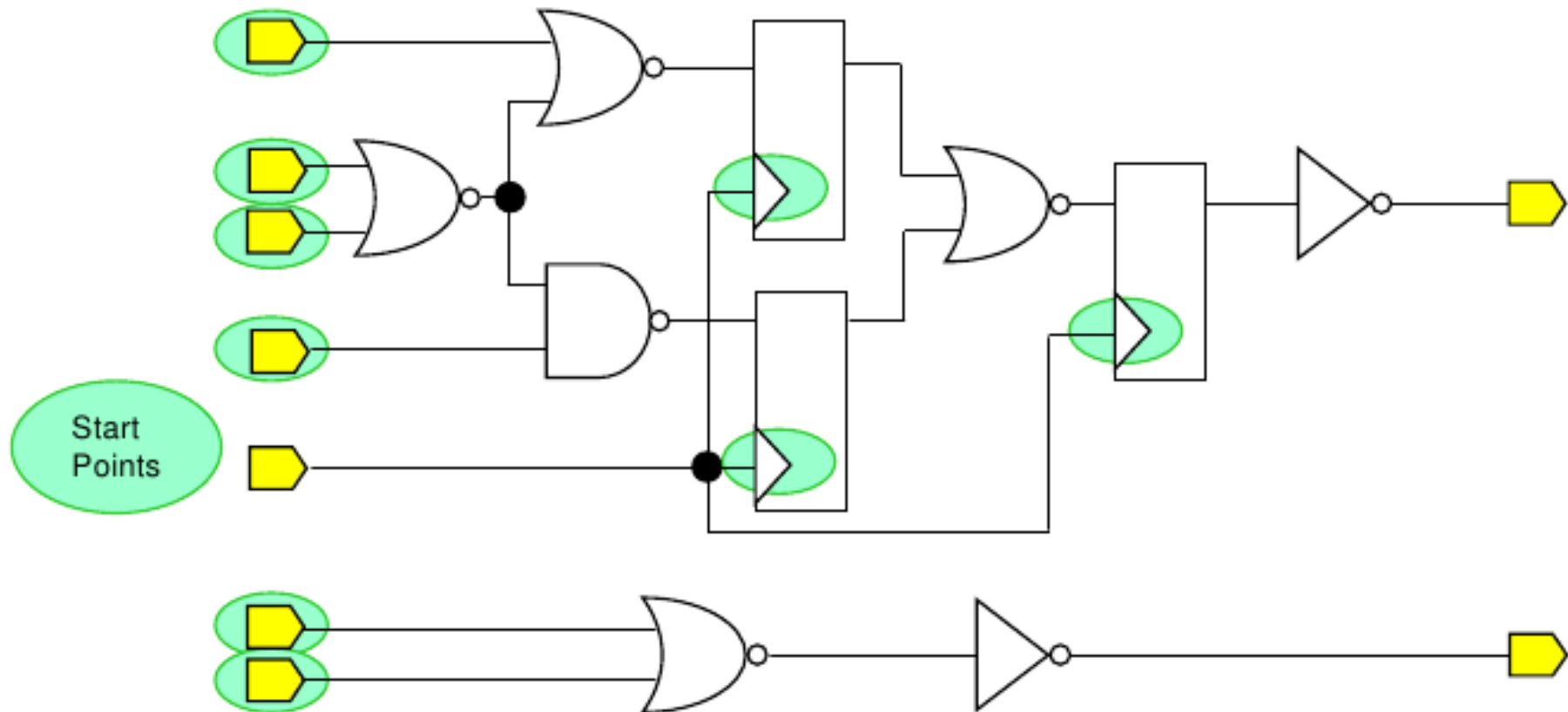
O projeto constantemente não atinge os requisitos de temporização durante a STA. Qual parece ser o problema?

O mínimo tempo de setup é de 150 ps, porém o período do clock é de apenas 100 ps. É necessário aumentar o período do clock para no mínimo o tempo de setup de 150 ps.

# Timing Paths: Start Points

Existem dois tipos de Start Points:

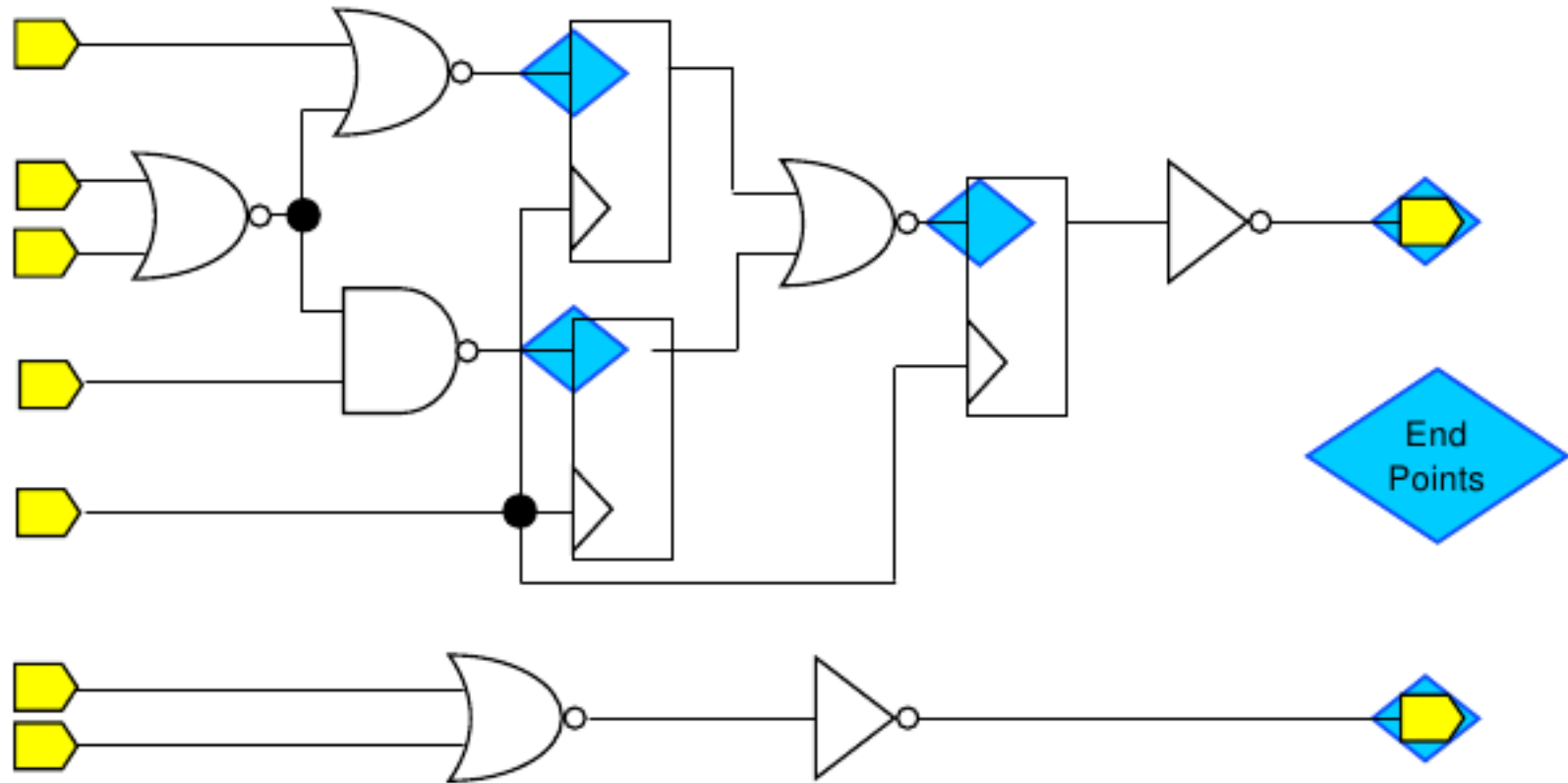
- Uma porta de entrada do sistema (que não uma porta de clock)
- O pino de clock de uma célula seqüencial



# Timing Paths: End Points

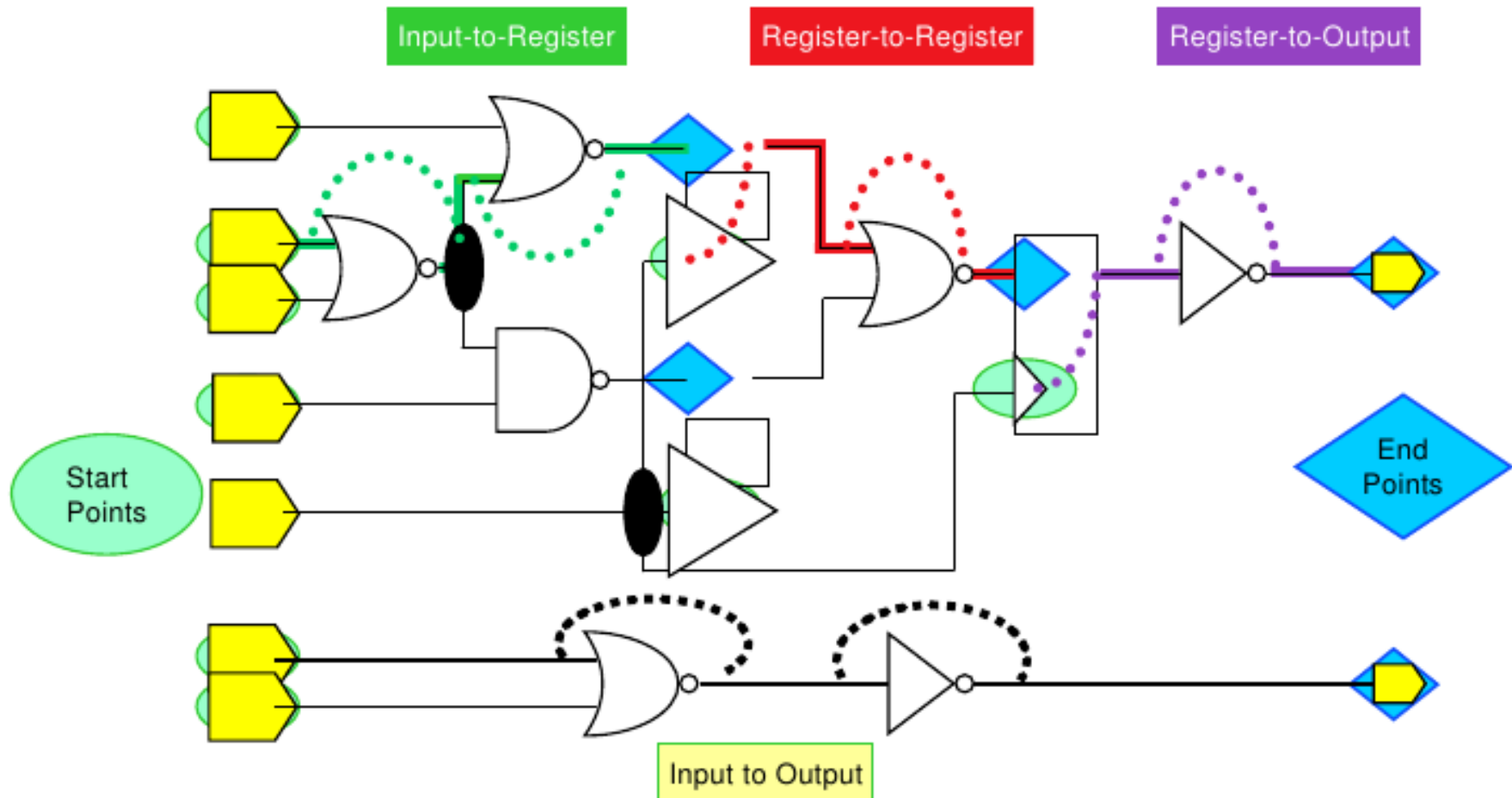
Existem dois tipos de End Points:

- Um pino de entrada de dados de uma célula seqüencial
- Uma porta de saída do sistema



# Timing Paths: Tipos

Um Timing Path é a combinação de todos os Timing Arc de um Start Point a um End Point. Existem quatro tipos:



# Timing Slack

Slack (folga) é a diferença entre o tempo requerido e o tempo de chegada de um sinal a um endpoint.

$\text{Slack} = \text{Required Time} - \text{Arrival Time}$

A ferramenta de síntese usa o slack para identificar os paths a serem otimizados.

Um slack positivo indica que a ferramenta conseguiu atender à especificação de tempo. Um slack negativo indica que a temporização não foi atendida.

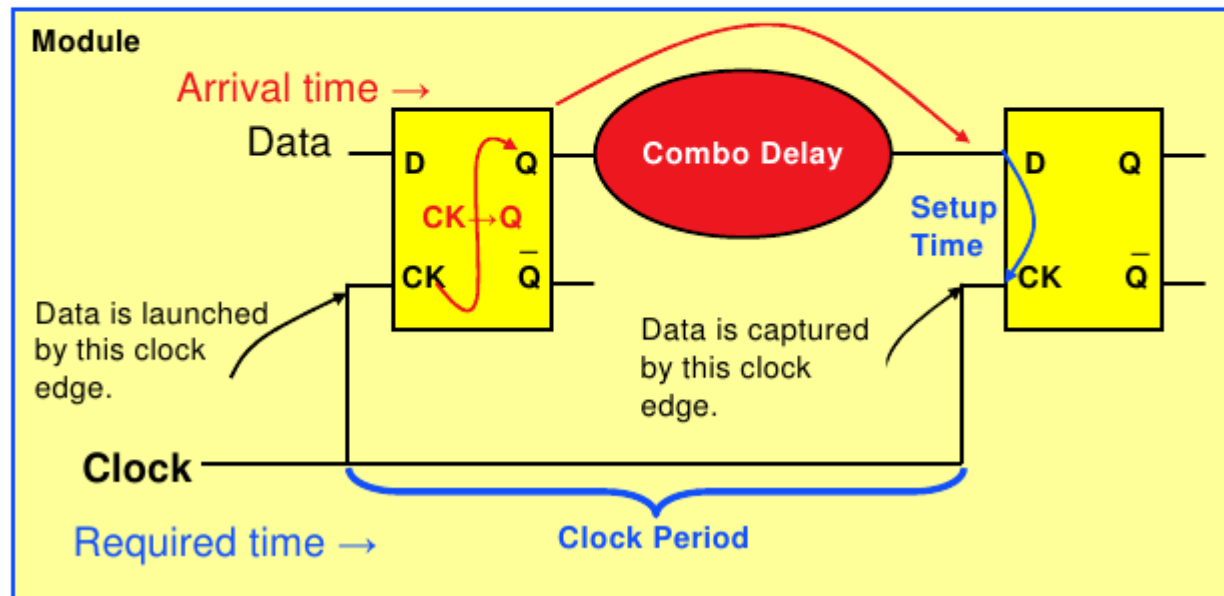
A determinação do slack é diferente para cada tipo de path.

# Register-to-Register Setup Requirement

Required time = Clock Period – Setup Time

Arrival time =  $t_{CK \rightarrow Q}$  + Combo delay

Slack = Required time – Arrival time





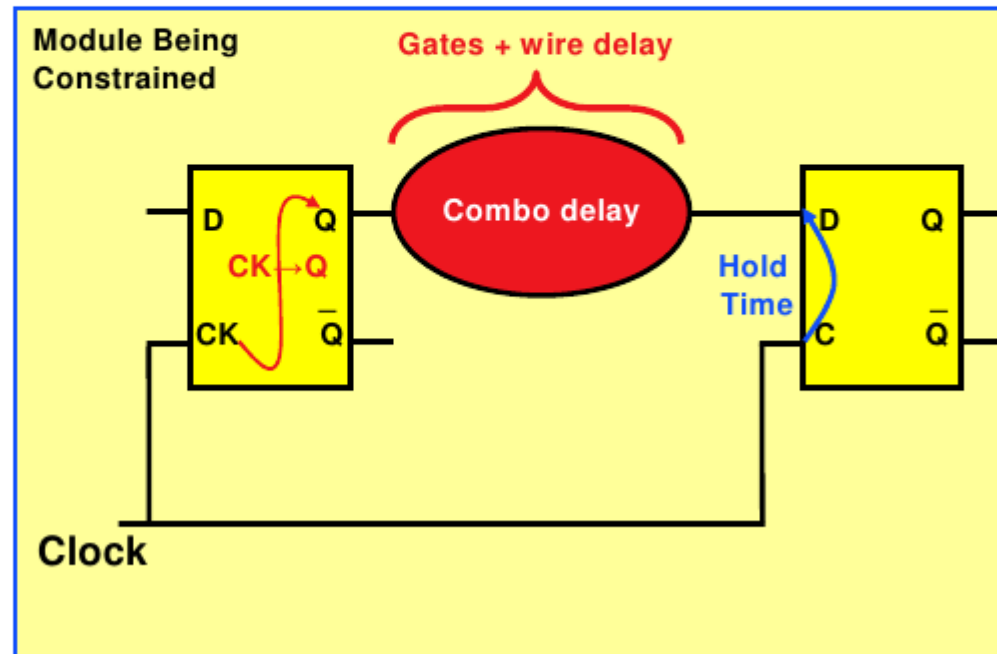
# Register-to-Register Hold Requirement

Required time = Hold Check (0) + Hold Time

Note: Default hold check is at 0.

Arrival time =  $t_{CK \rightarrow Q}$  + Combo delay

Slack = Required time – Arrival time

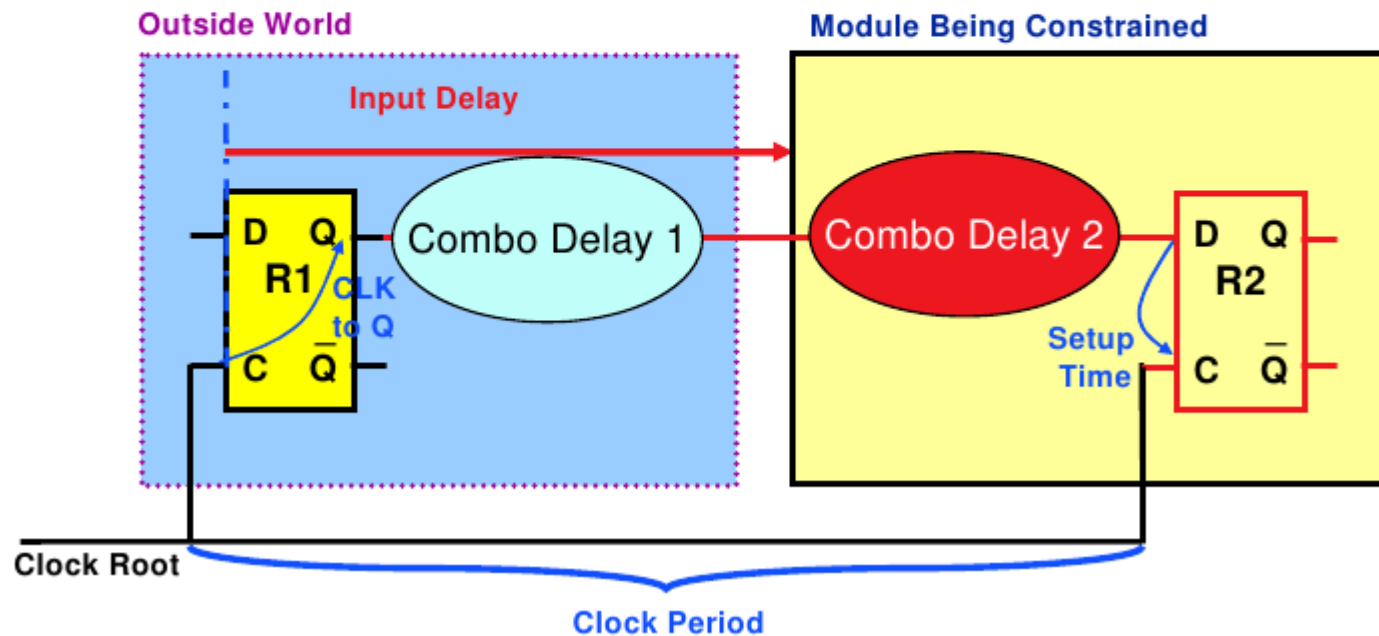


# Input-to-Register Setup Requirement

Required time = (Clock Period) – (Setup time of R2)

Arrival time = ( $t_{\text{ck} \rightarrow \text{Q}}$  of R1) + (Combo Delay 1) + (Combo Delay 2)

Slack = (Required time) – (Arrival time)

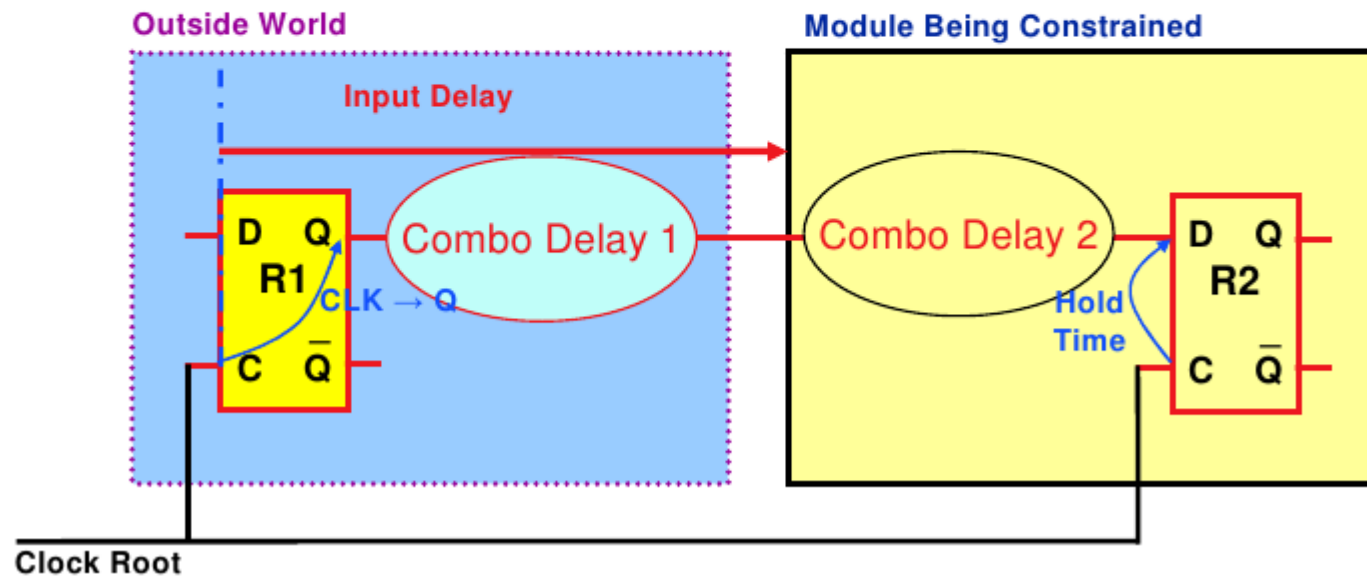


# Input-to-Register Hold Requirement

Required time = Hold Check (0) + Hold time of R2

Arrival time = Input Delay ( $t_{\text{clk-q}}$  of R1 + Combo Delay 1) + Combo Delay 2

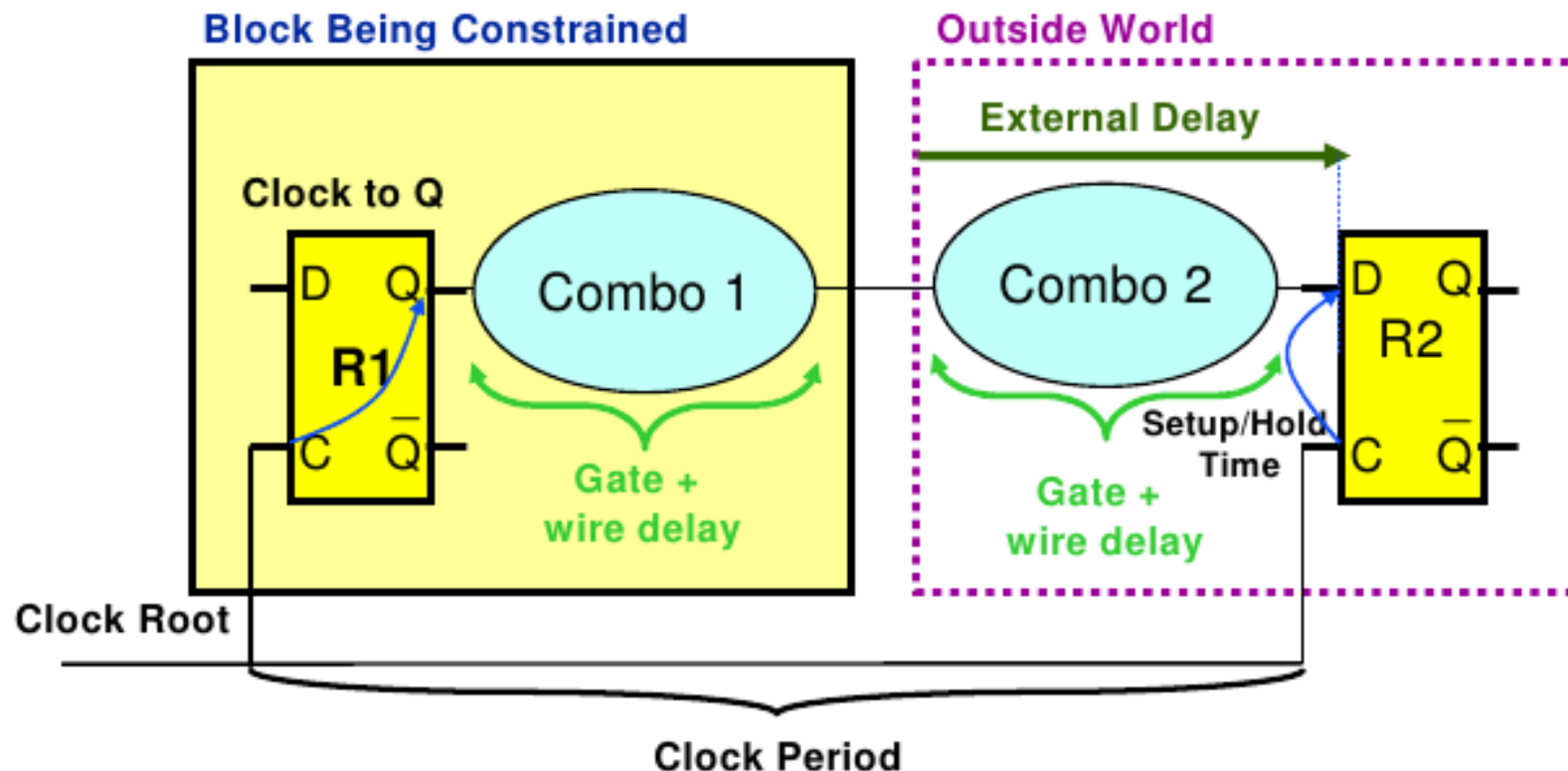
Slack = Required time – Arrival time



# Register-to-Output Path Timing Requirements

Setup slack = clock\_period – clk\_to\_q – combo\_logic\_delay – output\_delay

Hold slack = Hold check (0) + hold time of R2 + Phase Shift – Output Delay – Combo1 Delay



# Input to Output Path Timing Requirements

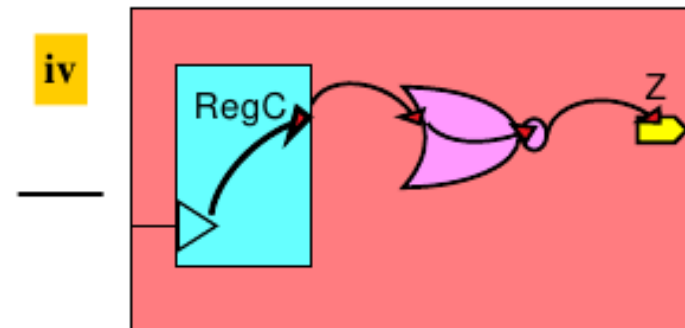
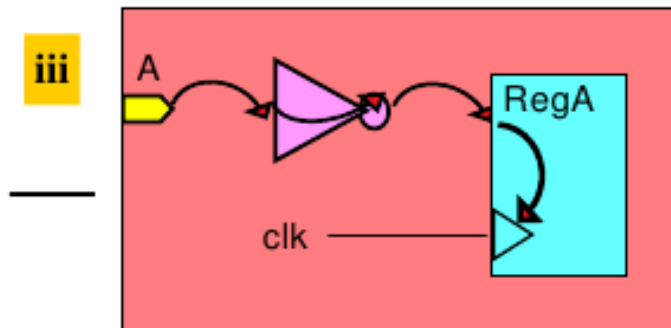
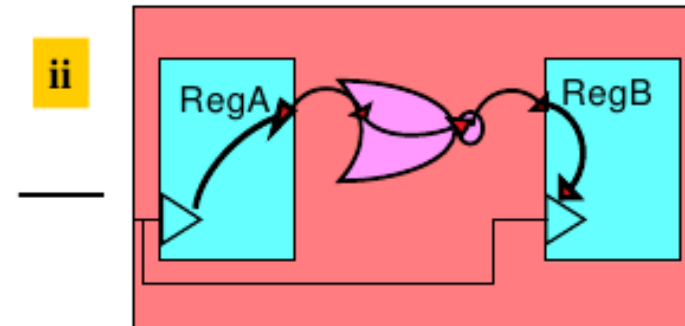
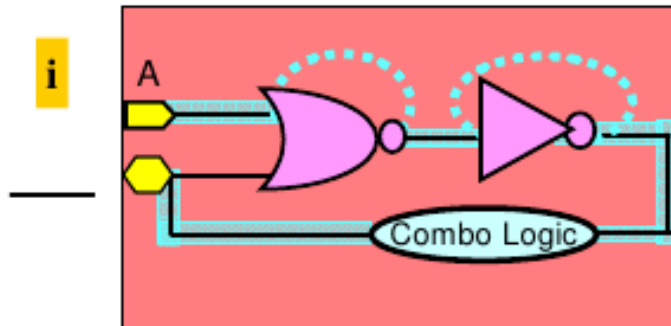
Input delay e output delay são definidos com relação ao clock.  
Para caminhos combinacionais é definido um clock virtual.

O slack é calculado como se segue:

$$\text{Slack} = \text{Clock Period} - \text{Input Delay} - \text{Output Delay} - \text{Combinational Delay}$$

# Exercício

Identify the path type in each of the following illustrations.



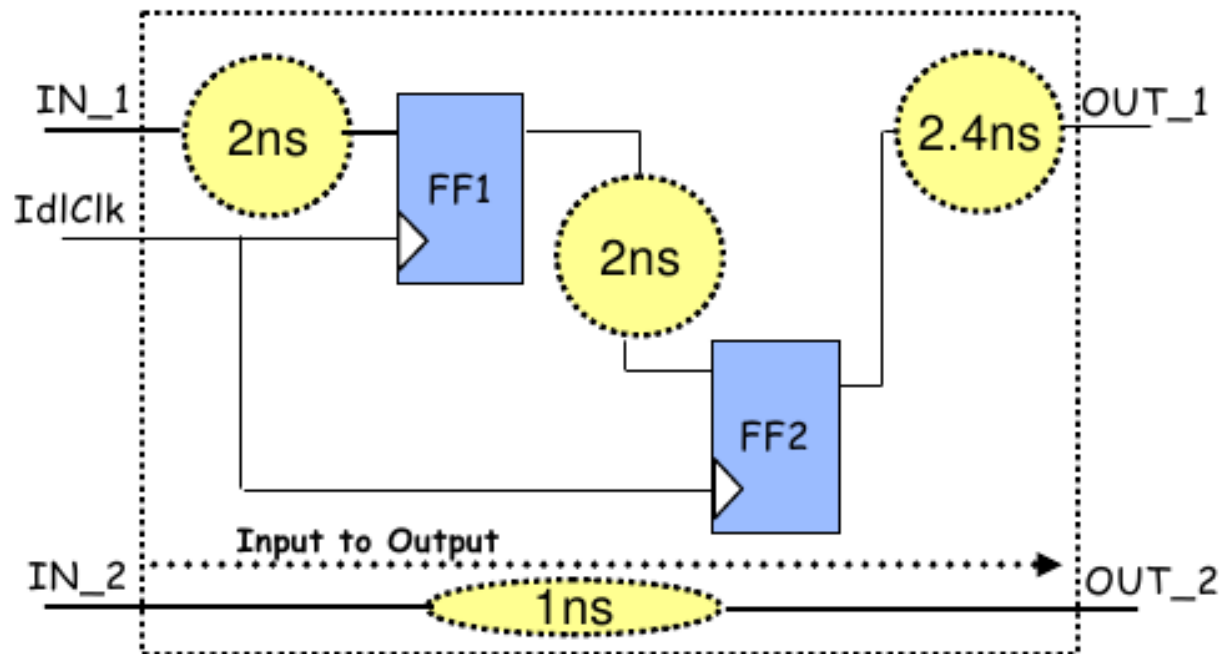
- a) Combinational   b) Register-to-Output  
c) Register-to-Register   d) Input-to-Register

# Exercício

Here are the parameters for this activity:

- ◆ The clock-to-Q delay of each flop is 0.5 ns.
- ◆ The setup of each flop is 0.5 ns.
- ◆ All input and output delays are 0.5 ns.
- ◆ The clock period of Clk is 4 ns.

What is the worst timing slack based on this illustration?



# Exercício

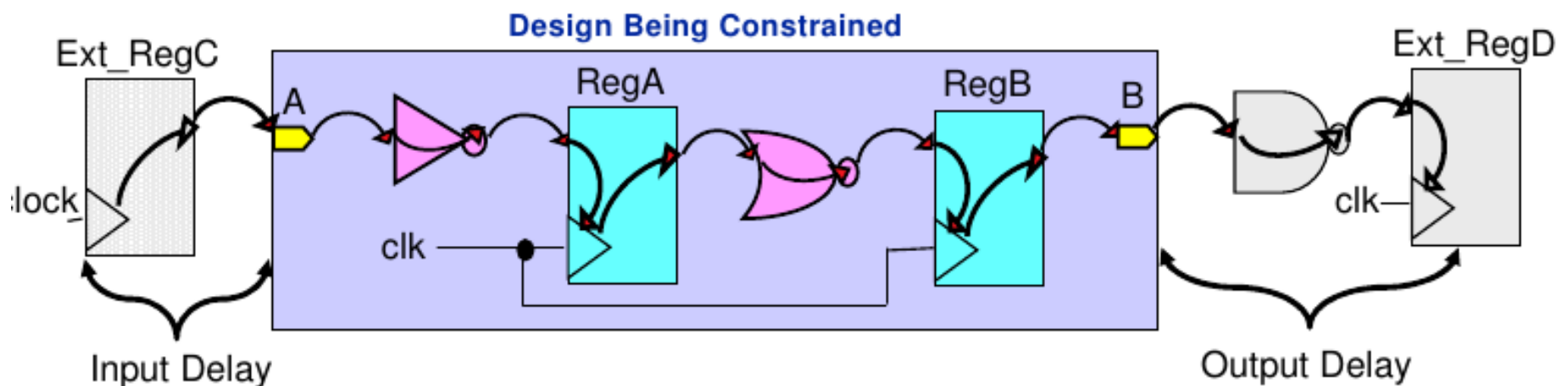
**1. A path is always:**

- a)** From an input to an output. **b)** From an input to the D-input of a flop
- c)** From a start point to an end point **d)** A single timing arc

**2. Which of the following are considered as start points for timing paths?**

- a)** Input ports of a design excluding clock ports **b)** D-input of a register
- c)** Clock pin of a register **d)** Both a and c **e)** All of the above

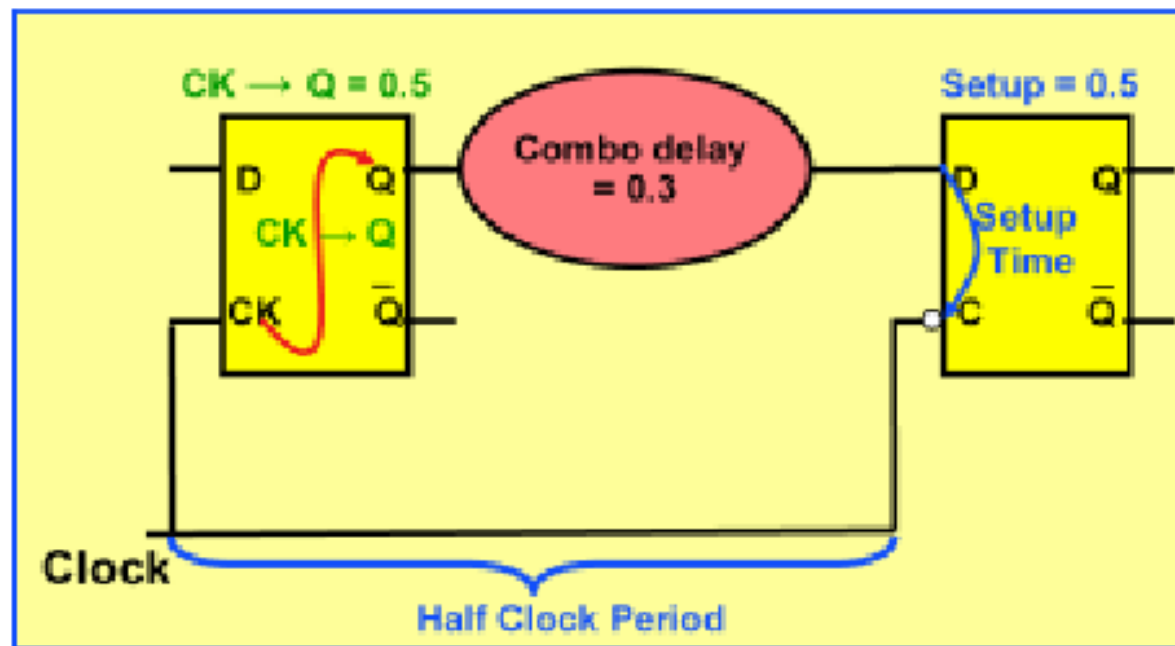
**3. The output delays are measured relative to what type of signal?**





# Exercício

4. What is the slack of the path, if the required time is 1.2 ns and the arrival time is 1.5 ns? \_\_\_\_\_
5. What is the slack of the register-to-register path in the illustration, if the clock period is 2 ns,  $\text{clk} \rightarrow \text{Q}$  delay and setup time of capture flop are 0.5 ns, and the combinational delay is 0.3 ns?



# Design Constraints

Constraints (restrições, especificações, condições de contorno...) provêm especificações que o projeto deve atender.

Exemplos:

- Clock constraints
- External constraints
- Power constraints
- Net Delay constraints
- Environmental constraints
- Design rules for manufacturing (ASICs)

# Constraints Formats

\*.SDC – Synopsys Design Constraints é o formato padrão na indústria.

Exemplos:

```
create_clock -period 100 -waveform {0 50} clk
```

```
create_clock -period 100 -waveform {0 50} {get_ports {clk}}
```

```
create_clock -period 100.000000 -waveform {0.000000\ 50.000000}[get_ports {clk}]
```

# Constraints comuns

## Operating conditions

- ◆ set\_operating\_conditions

## Wire-load models

- ◆ set\_wire\_load\_mode
- ◆ set\_wire\_load\_model
- ◆ set\_wire\_load\_selection\_group

## Environmental

- ◆ set\_drive
- ◆ set\_driving\_cell
- ◆ set\_load
- ◆ set\_fanout\_load
- ◆ set\_input\_transition
- ◆ set\_port\_fanout\_number

## Design rules

- ◆ set\_max\_capacitance
- ◆ set\_max\_fanout
- ◆ set\_max\_transition

## Timing

- ◆ create\_clock
- ◆ create\_generated\_clock
- ◆ set\_clock\_latency
- ◆ set\_clock\_transition
- ◆ set\_disable\_timing
- ◆ set\_propagated\_clock
- ◆ set\_clock\_uncertainty
- ◆ set\_input\_delay
- ◆ set\_output\_delay

## Exceptions

- set\_false\_path
- set\_max\_delay
- set\_multicycle\_path

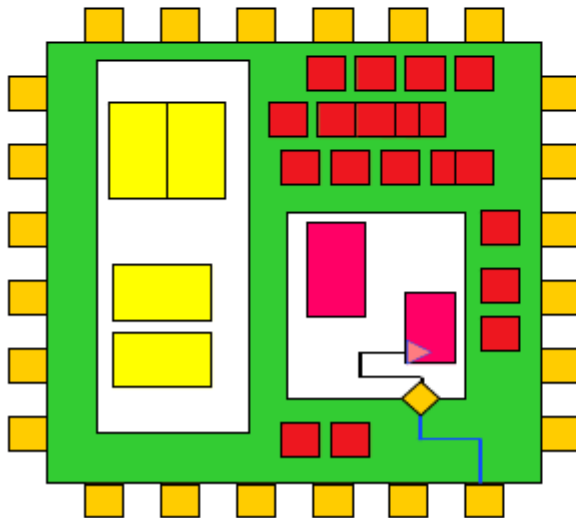
## Power

- ◆ set\_max\_dynamic\_power
- ◆ set\_max\_leakage\_power

# Elementos de Projeto

You apply certain constraints to design objects to affect different parts of the design.

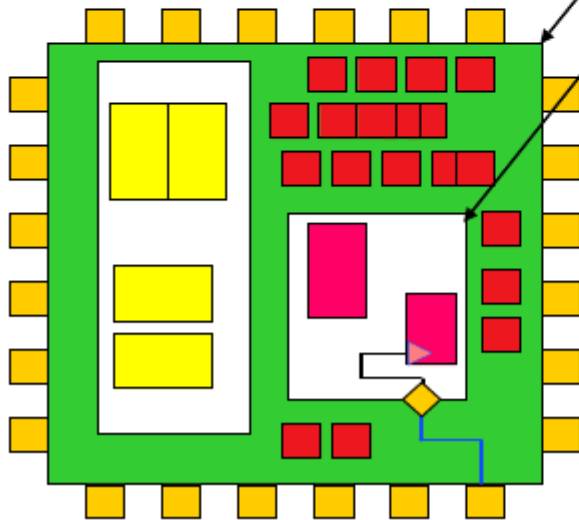
The table shows several design objects and the commands to get a list of these objects.



Object	Command	Description
Design	<code>current_design</code>	Design is a container for cells or is the entire circuit.
Cell or Block	<code>get_cells</code>	Cell is an instance of a design or is a library component.
Port	<code>get_ports</code> <code>all_inputs</code> <code>all_outputs</code>	A port is a signal entry point or exit point to a design.
Pin	<code>get_pins</code>	A pin is a hierarchical port of a design, port of an instance, or a port of a library cell.
Clock	<code>get_clocks</code> <code>all_clocks</code>	A clock is a port or a pin that drives sequential cells.
Net	<code>get_nets</code>	A net is an interconnect between cell pins and design ports.

# Elementos de Projeto

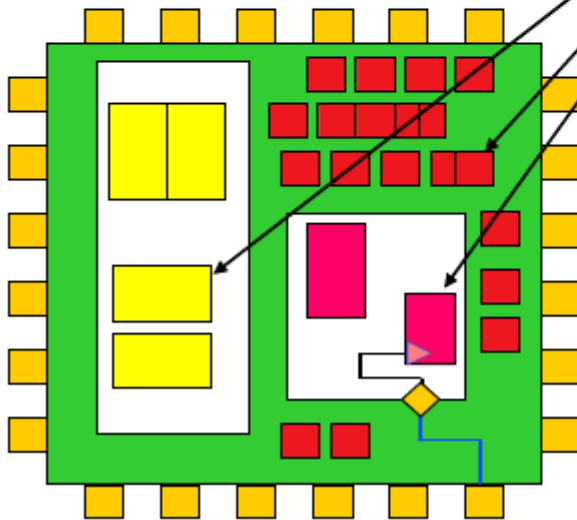
**Operating conditions affect the chip or a block.**



Object	Command	Description
Design	current_design	Design is a container for cells or is the entire circuit.
Cell or Block	get_cells	Cell is an instance of a design or is a library component.
Port	get_ports all_inputs all_outputs	A port is a signal entry point or exit point to a design.
Pin	get_pins	A pin is a hierarchical port of a design, port of an instance, or a port of a library cell.
Clock	get_clocks all_clocks	A clock is a port or a pin that drives sequential cells.
Net	get_nets	A net is an interconnect between cell pins and design ports.

# Elementos de Projeto

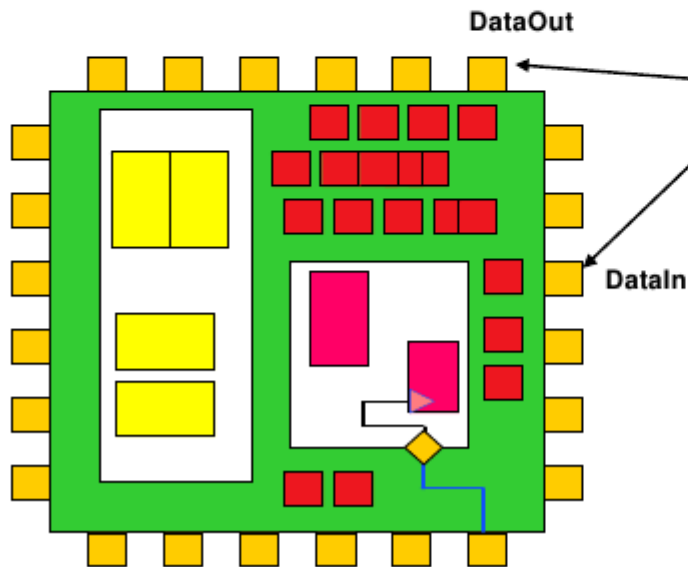
**Apply power constraints and wire-load models to the cell or block object.**



Object	Command	Description
Design	current_design	Design is a container for cells or is the entire circuit.
Cell or Block	get_cells	Cell is an instance of a design or is a library component.
Port	get_ports all_inputs all_outputs	A port is a signal entry point or exit point to a design.
Pin	get_pins	A pin is a hierarchical port of a design, port of an instance, or a port of a library cell.
Clock	get_clocks all_clocks	A clock is a port or a pin that drives sequential cells.
Net	get_nets	A net is an interconnect between cell pins and design ports.

# Elementos de Projeto

**Apply timing constraints and environmental constraints to a port object.**



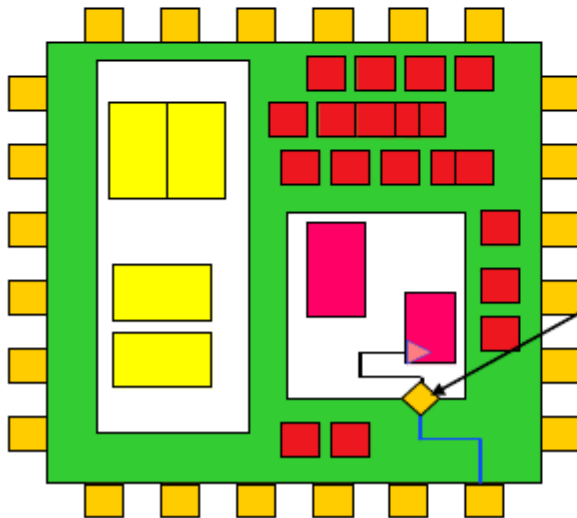
Object	Command	Description
Design	current_design	Design is a container for cells or is the entire circuit.
Cell or Block	get_cells	Cell is an instance of a design or is a library component.
Port	get_ports all_inputs all_outputs	A port is a signal entry point or exit point to a design.
Pin	get_pins	A pin is a hierarchical port of a design, port of an instance, or a port of a library cell.
Clock	get_clocks all_clocks	A clock is a port or a pin that drives sequential cells.
Net	get_nets	A net is an interconnect between cell pins and design ports.



# Elementos de Projeto

**Apply timing constraints and environmental constraints to a port object.**

**Pins are like ports of sub-designs.**

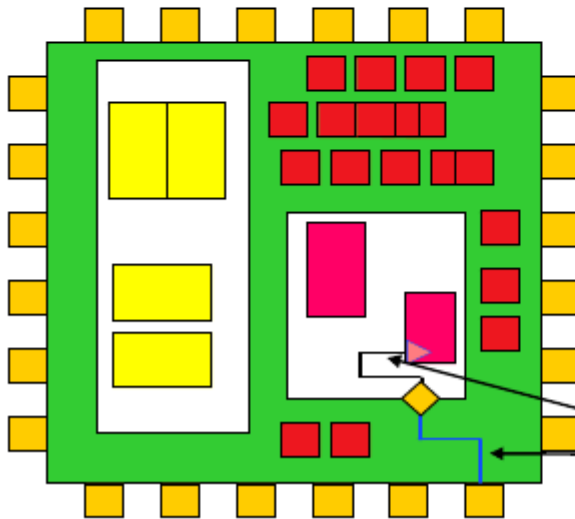


Object	Command	Description
Design	current_design	Design is a container for cells or is the entire circuit.
Cell or Block	get_cells	Cell is an instance of a design or is a library component.
Port	get_ports all_inputs all_outputs	A port is a signal entry point or exit point to a design.
Pin	get_pins	A pin is a hierarchical port of a design, port of an instance, or a port of a library cell.
Clock	get_clocks all_clocks	A clock is a port or a pin that drives sequential cells.
Net	get_nets	A net is an interconnect between cell pins and design ports.

# Elementos de Projeto

**Apply timing constraints to the net objects.**

**Wire-load models affect the delays of these net objects.**



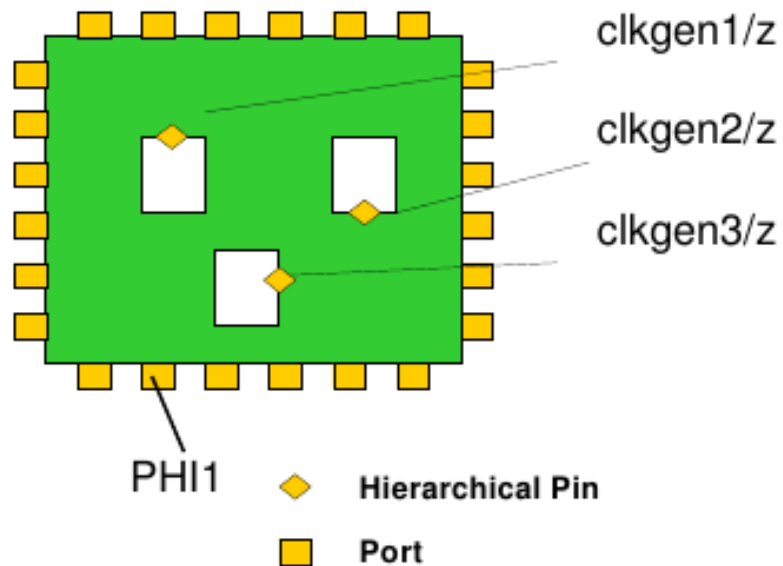
Object	Command	Description
Design	current_design	Design is a container for cells or is the entire circuit.
Cell or Block	get_cells	Cell is an instance of a design or is a library component.
Port	get_ports all_inputs all_outputs	A port is a signal entry point or exit point to a design.
Pin	get_pins	A pin is a hierarchical port of a design, port of an instance, or a port of a library cell.
Clock	get_clocks all_clocks	A clock is a port or a pin that drives sequential cells.
Net	get_nets	A net is an interconnect between cell pins and design ports.

# Criando um Clock

O comando `create_clock` define um sinal de clock ideal e especifica os ports e pins a ele conectados.

```
create_clock -name "clk1" -period 10 -waveform {0.0  
5.0} {PHI1}
```

```
create_clock -name "clk2" -period 20 -waveform {0.0  
10.0} {clkgen1/Z clkgen2/Z clkgen3/Z}
```



```
create_clock -period  
period_value  
[-name clock_name]  
[-waveform edge_list]  
[-add]  
[ source_objects]
```

# Transições de Clock

Clock Transition = Clock Slew. É definido a partir de um clock ideal já criado.

## Syntax

```
set_clock_transition [-rise] [-fall] [-min] [-max] \  
transition_value clock_list
```

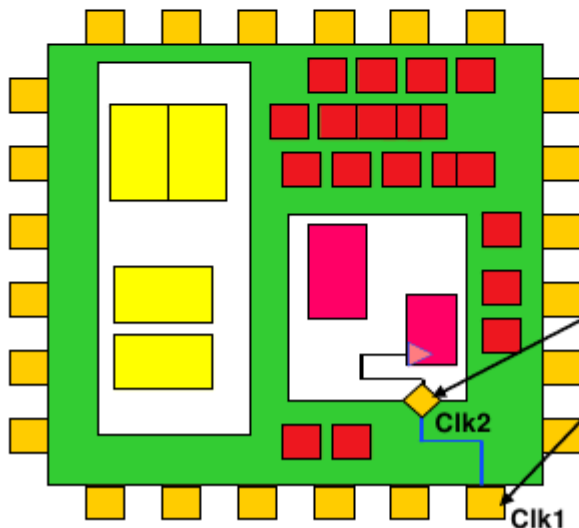
## Example

```
set_clock_transition 0.38 -rise [get_clocks clk1]  
set_clock_transition 0.25 -fall [get_clocks clk1]
```



# Elementos de Projeto

**Apply clock related timing constraints to the clock object.**



Object	Command	Description
Design	current_design	Design is a container for cells or is the entire circuit.
Cell or Block	get_cells	Cell is an instance of a design or is a library component.
Port	get_ports all_inputs all_outputs	A port is a signal entry point or exit point to a design.
Clock	get_clocks all_clocks	A clock is a port or a pin that drives sequential cells.
Pin	get_pins	A pin is a hierarchical port of a design, port of an instance, or a port of a library cell.
Net	get_nets	A net is an interconnect between cell pins and design ports.

O fanout do sinal de clock é geralmente muito maior que de outros sinais do circuito. A degradação do slew é também maior que de outros sinais, dependendo da distância da célula à fonte de clock. Para modelar a natureza do clock de forma mais eficaz, adicionam-se os tempos de transição, incertezas e latência.

# Incertezas do Clock

(uncertainty = skew + jitter)

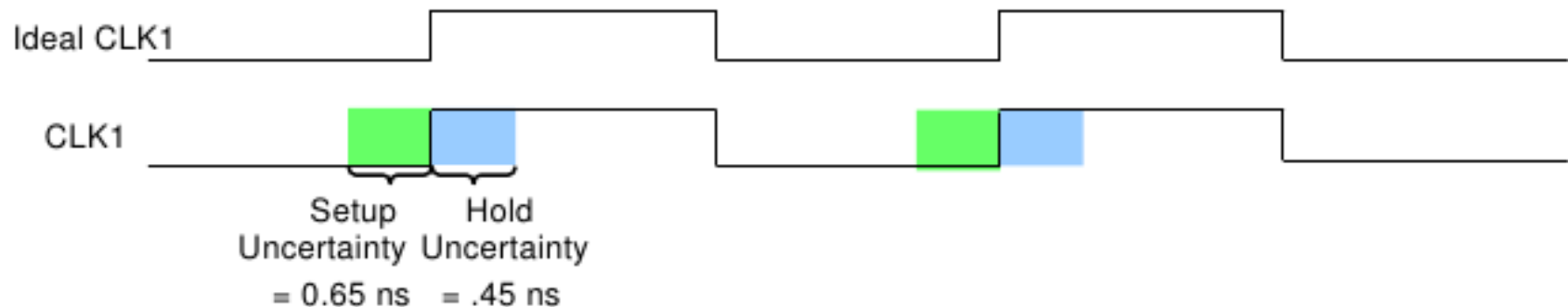
## Syntax

```
set_clock_uncertainty [-from from_clock] [-to to_clock] [-rise] [-fall] [-setup] [-hold] uncertainty [clock_objects]
```

## Example

```
create_clock -name CLK1 -period 10 -waveform { 0 4} [get_ports CLK1]
set_clock_uncertainty -setup 0.65 [get_clocks CLK1]
set_clock_uncertainty -hold 0.45 [get_clocks CLK1]
```

The above commands specify a setup uncertainty of 0.65 ns, and a hold uncertainty of 0.45 ns on CLK1.



# Incertezas do Clock

(uncertainty = skew + jitter)

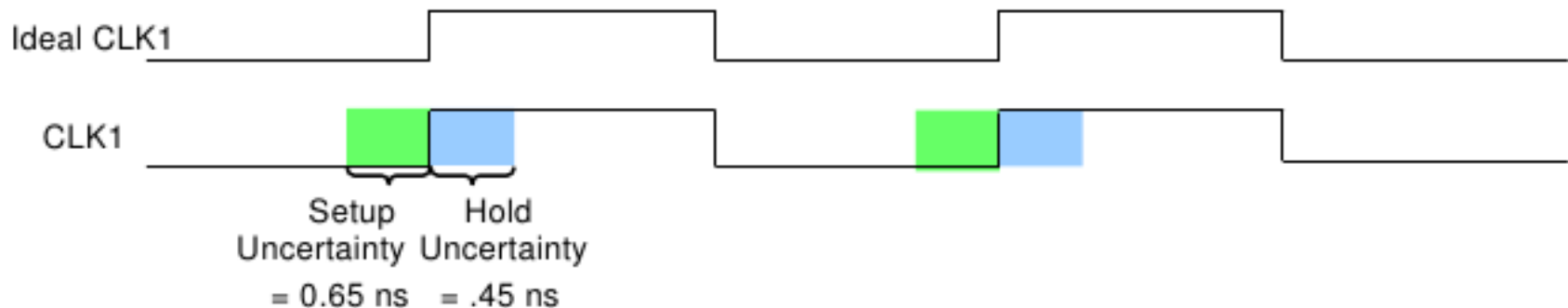
## Syntax

```
set_clock_uncertainty [-from from_clock] [-to to_clock] [-rise] [-fall] [-setup] [-hold] uncertainty [clock_objects]
```

## Example

```
create_clock -name CLK1 -period 10 -waveform { 0 4} [get_ports CLK1]  
set_clock_uncertainty -setup 0.65 [get_clocks CLK1]  
set_clock_uncertainty -hold 0.45 [get_clocks CLK1]
```

The above commands specify a setup uncertainty of 0.65 ns, and a hold uncertainty of 0.45 ns on CLK1.



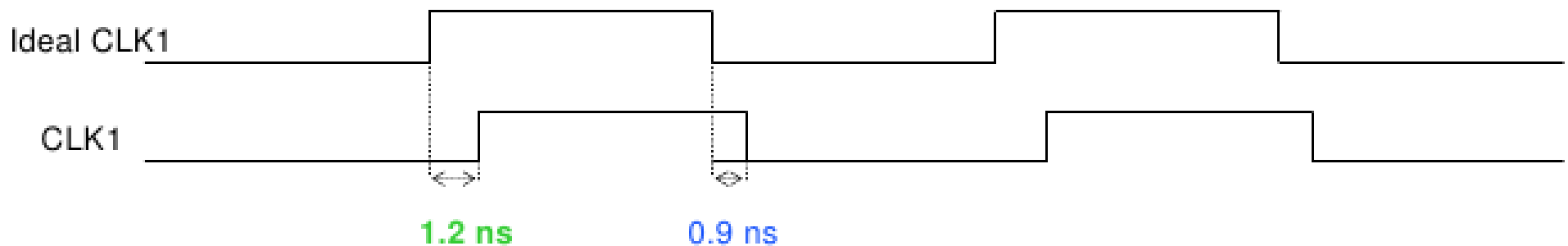
# Clock Latency

## Syntax

```
set_clock_latency [-rise] [-fall] [-min] [-max]  
                  [-source] [-late] [-early] delay object_list
```

## Example

```
set_clock_latency 1.2 -rise [get_clocks CLK1]  
set_clock_latency 0.9 -fall [get_clocks CLK1]
```





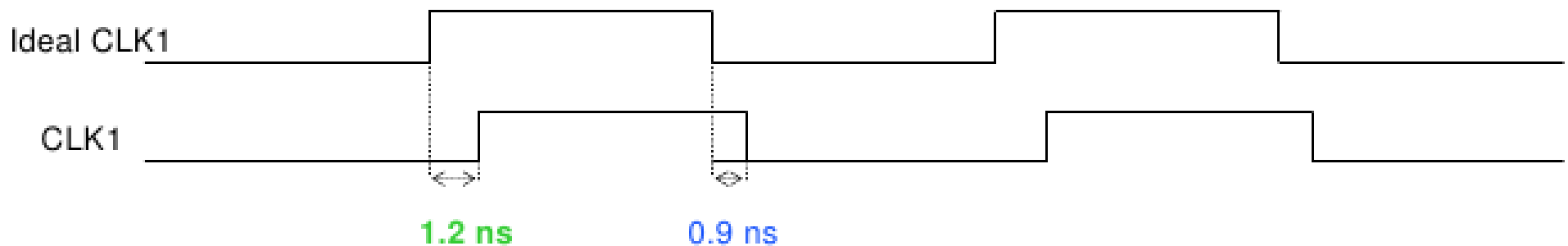
# Clock Latency

## Syntax

```
set_clock_latency [-rise] [-fall] [-min] [-max]  
                  [-source] [-late] [-early] delay object_list
```

## Example

```
set_clock_latency 1.2 -rise [get_clocks CLK1]  
set_clock_latency 0.9 -fall [get_clocks CLK1]
```



# Clock Latency: Hold e Setup

```
set_clock_latency 0.8 -source -early [get_clocks CLK1]  
set_clock_latency 0.9 -source -late [get_clocks CLK1]
```

