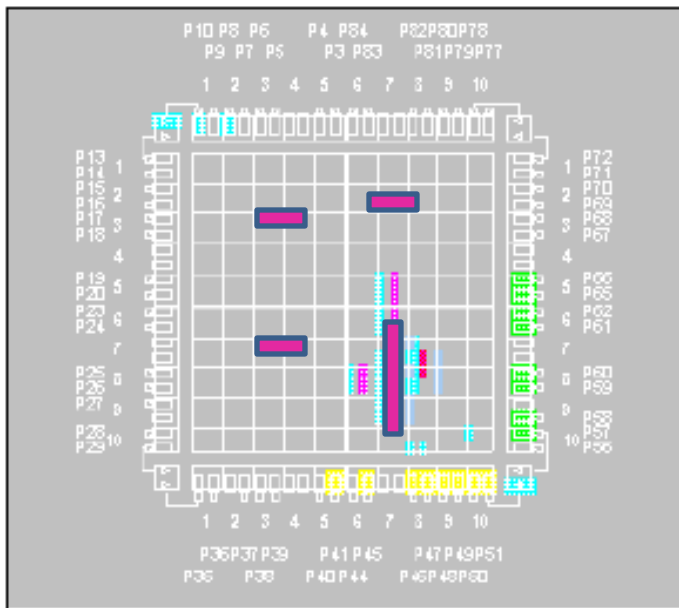


# FPGAs-3

# Design Without and With Constraints

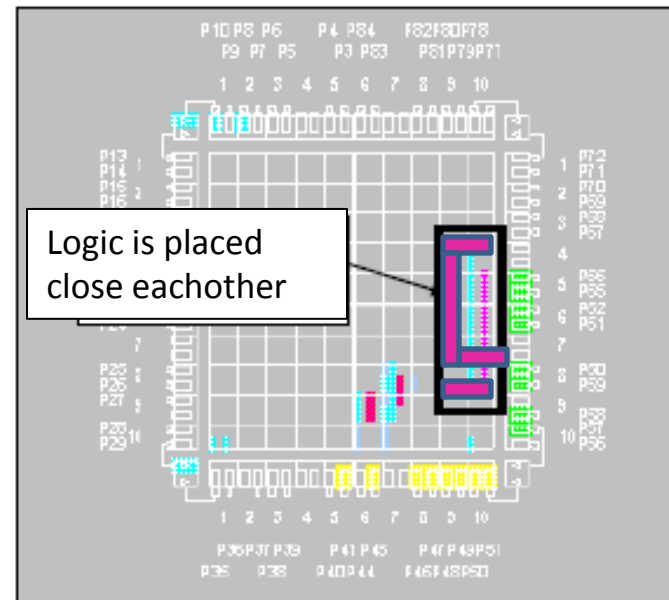
Logic Placement Can Be Very Different

Without global timing constraints



Logic is placed randomly

With global timing constraints



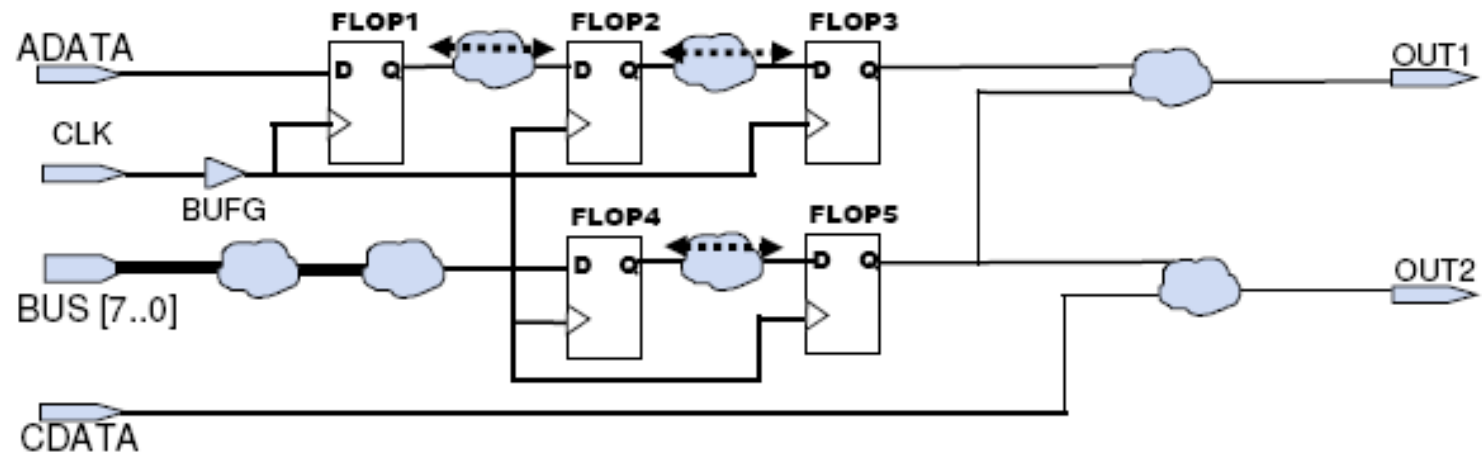
Logic is placed to result in a faster design

# Setup and hold time

- ***Setup time***: The data must become valid at least a setup time before the arrival of the active clock edge at its pin.
- ***Hold Time***: The data must stay valid at least a hold time after the arrival of the active clock edge at its pin.

# Period Constraints

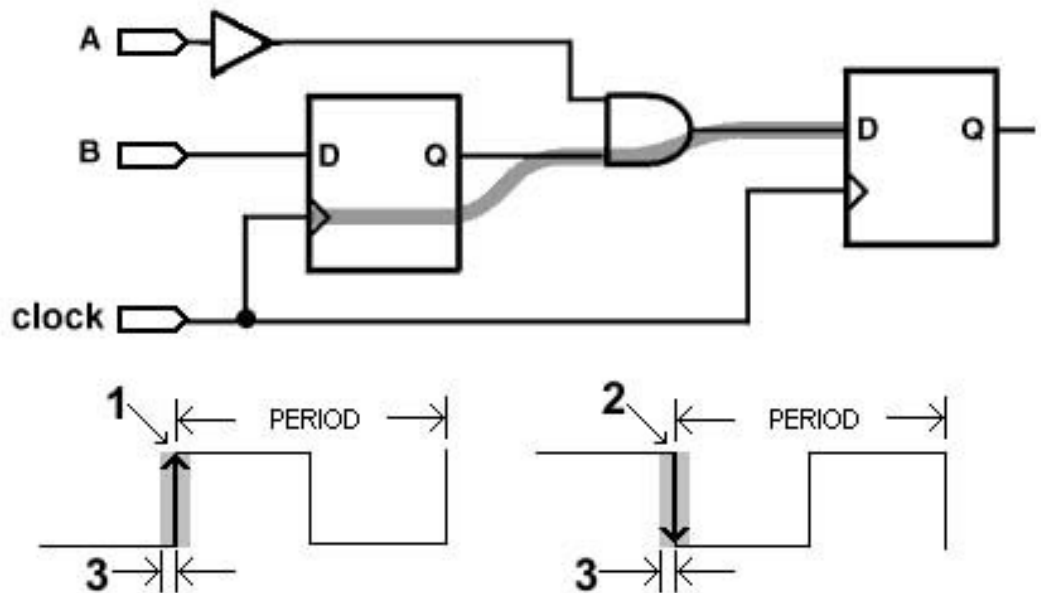
Cover Paths Between Synchronous Elements



# PERIOD Constraint

Use the Most Accurate Timing Information

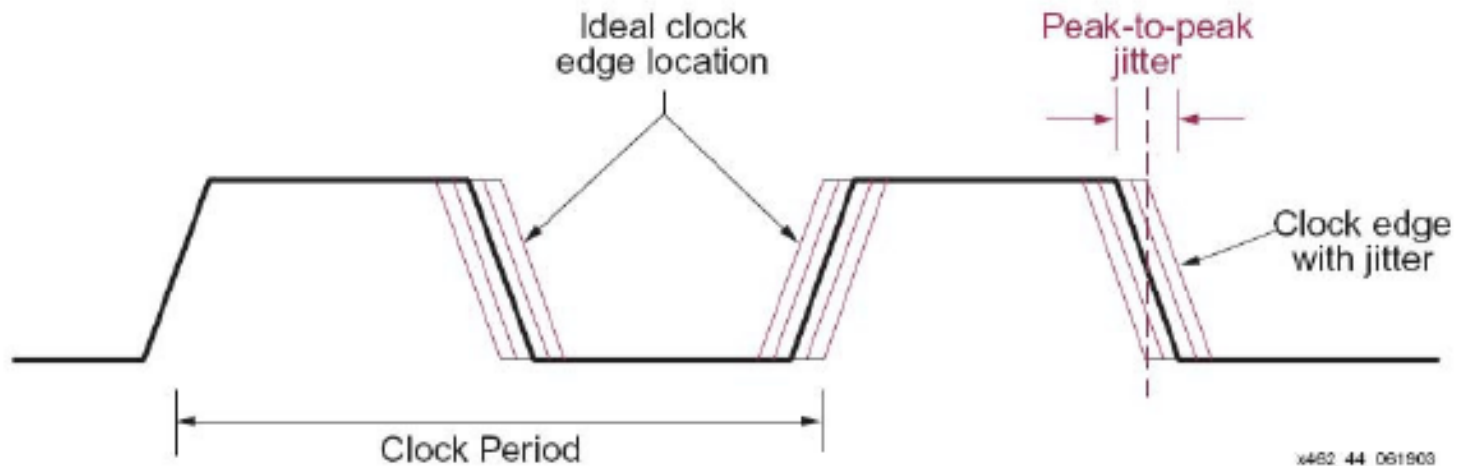
- ✓ Clock skew between the source and destination flip-flops
- ✓ Synchronous elements clocked on the negative edge
- ✓ Unequal clock duty cycles
- ✓ Clock input jitter



# Period Constraint

Clock uncertainty is automatically accounted for in global constraint calculations

Clock jitter is a form of clock uncertainty



# Period Constraint

Timing Analyzer calculation accounts for most accurate timing information

Timing constraint: [TS Inst clockgen\\_CLK0\\_BUF\\_0 = PERIOD TIMEGRP "Inst clockgen\\_CLK0\\_BUF\\_0" TS clk HIGH 50%](#)  
[INPUT\\_JITTER 0.001 ns;](#)

10 paths analyzed, 4 endpoints analyzed, 0 failing endpoints

0 timing errors detected. (0 setup errors, 0 hold errors)

Minimum period is 2.015ns.

Slack: [7.984 ns](#) (requirement - (data path - clock path skew + uncertainty))

Source: <a href="#">bincount/EU2/U0/q_i_0</a> (FF)	clk: clkgen_out rising at 0.000ns
Destination: <a href="#">bincount/EU2/U0/q_i_3</a> (FF)	clk: clkgen_out rising at 10.000ns

Equation takes into account data path delay, clock skew and clock uncertainty

Requirement	Data Path Delay	Clock Path Skew:	Clock Uncertainty
10.000ns	2.015ns (Levels of Logic = 2)	0.000ns	0.001ns

Clock Uncertainty: 0.001ns  $((TSJ^2 + TJ^2)^{1/2} + DJ) / 2 + PE$

Total System Jitter (TSJ):	0.000ns
----------------------------	---------

Total Input Jitter (TJ):	0.001ns
--------------------------	---------

Discrete Jitter (DJ):	0.000ns
-----------------------	---------

Phase Error (PE):	0.000ns
-------------------	---------

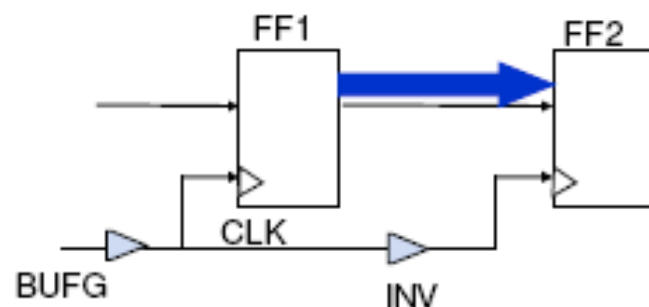
[Maximum Data Path: bincount/EU2/U0/q\\_i\\_0 to bincount/EU2/U0/q\\_i\\_3](#)

Delay type	Delay(ns)	Logical Resource
<a href="#">Tcko</a>	0.370	<a href="#">bincount/EU2/U0/q_i_0</a>
net (fanout=2)	0.246	<a href="#">dout_0_CBUF</a>

# PERIOD Constraint

Calculation takes into account inverted clock edges

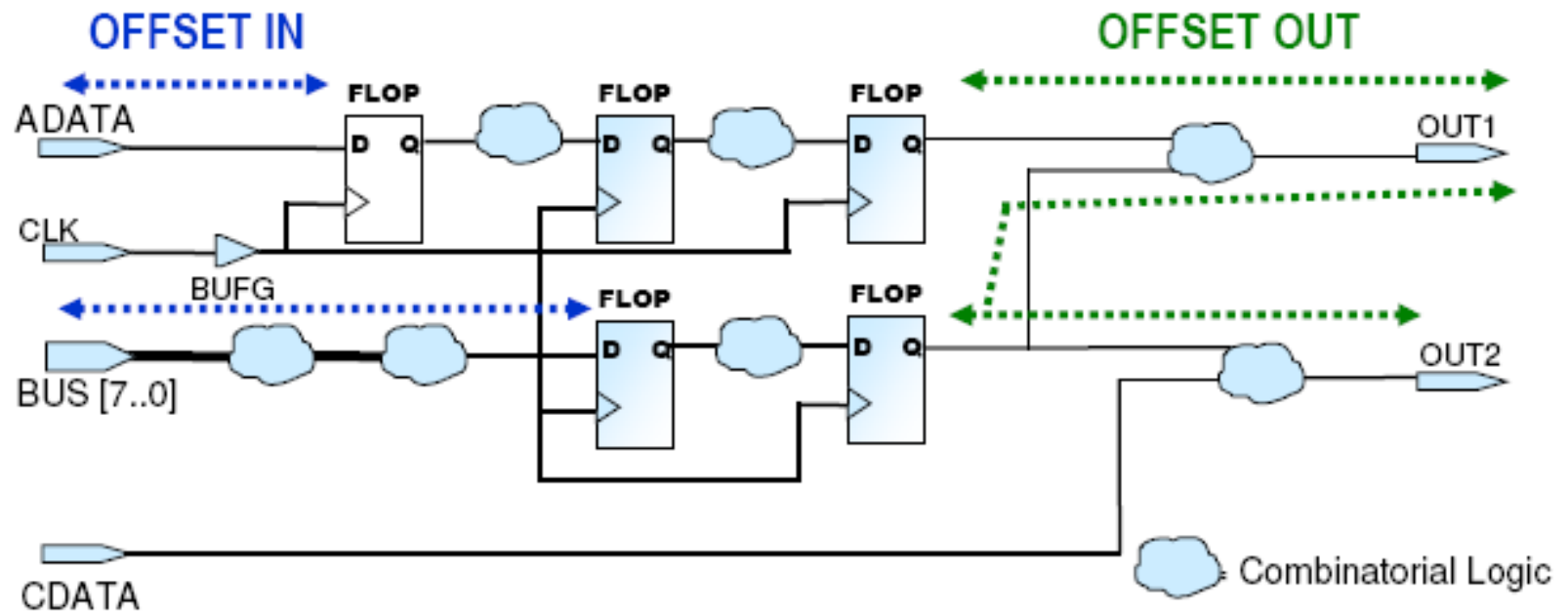
- Assume:
  - 50 percent duty cycle on CLK
  - PERIOD constraint of 10 ns
  - Because FF2 will be clocked on the falling edge of CLK, the path between the two flip-flops will be constrained to 50 percent of 10 ns = 5 ns





# Offset Constraints

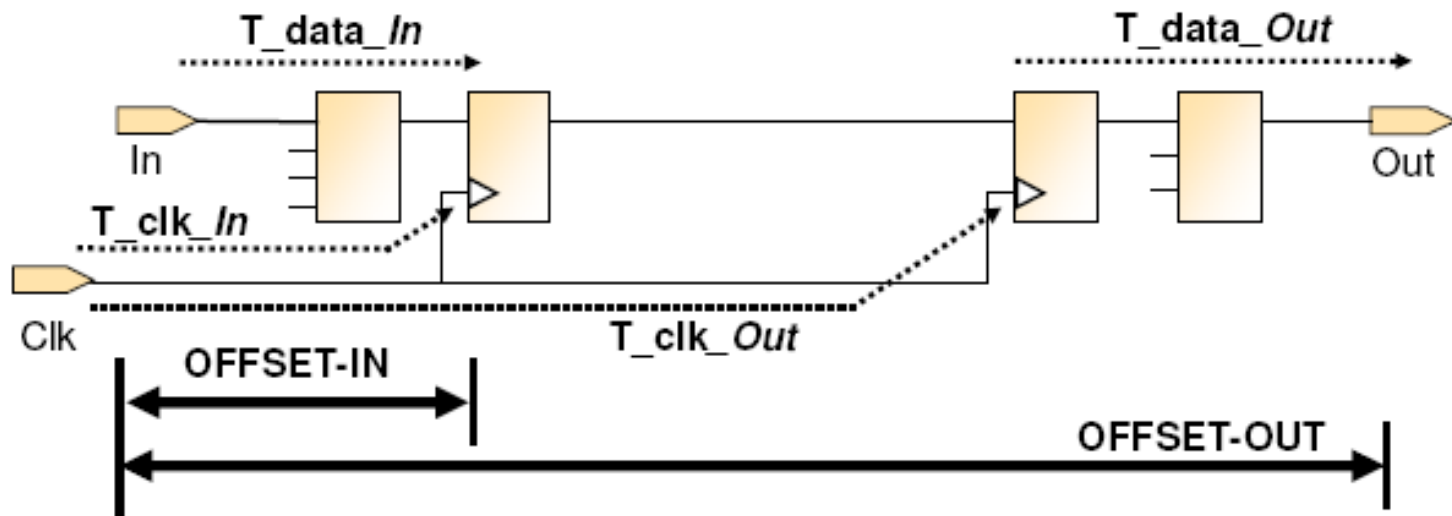
Constrains I/O Pads To/From Synchronous Elements relative to associated clock signal



# Offset Constraints

Accounts for Clock Delay

- $OFFSET\ IN = T\_data\_In - T\_clk\_In$
- $OFFSET\ OUT = T\_data\_Out + T\_clk\_Out$



# Offset Constraints

Timing Analyzer calculation accounts for most accurate timing information

Timing constraint: **OFFSET = III 5 ns BEFORE COMP "clk"**

30 paths analyzed, 8 endpoints analyzed, 0 failing endpoints

0 timing errors detected. (0 setup errors, 0 hold errors)

Minimum allowable offset is 3.693ns.

Slack: **1.307ns** (requirement - (data path - clock path - clock arrival + uncertainty))

Source: load (PAD)

Destination: bincount/EU2/U0/q\_j\_3 (FF)

clk: clkgen\_out rising at 0.000ns

Requirement

5.000ns

Data Path Delay

2.877ns (Levels of Logic = 2)

Clock Path Del

-0.815ns (Levels

Equation takes into account clock path, clock arrival, and clock uncertainty

Clock Uncertainty: 0.001ns  $((TSJ^2 + TIJ^2)^{1/2} + DJ) / 2 + PE$

Total System Jitter (TSJ):

0.000ns

Total Input Jitter (TIJ):

0.001ns

Discrete Jitter (DJ):

0.000ns

Phase Error (PE):

0.000ns

Maximum Data Path: load to bincount/EU2/U0/q\_j\_3

Delay type	Delay(ns)	Logical Resource
------------	-----------	------------------

<a href="#">Ttopi</a>	0.736	<a href="#">load</a> <a href="#">load_IBUF</a>
-----------------------	-------	---

net (fanout=5)	1.147	<a href="#">load_IBUF</a>
----------------	-------	---------------------------

# PERIOD Constraint Options

- TIMESPEC name
- Specific constraint value
  - Active clock edge
  - Duty cycle
- Relative to other PERIOD TIMESPEC
  - Useful for designs with multiple clock signals
  - Can define both frequency and phase relationships
- Input jitter

Initial active edge used for OFFSET value is set to HIGH

PERIOD

INPUT\_JITTER

\* TIMESPEC name: TS\_clk\_pin

\* Clock net name: clk\_pin

Clock signal definition

Specify time

Time: 37 Units: ns

Initial clock edge:  Rising (HIGH)  Falling (LOW)

Rising duty cycle: 50 Units: %

Relative to other period TIMESPEC

Reference TIMESPEC:

Factor

Operand:  Multiply by  Divide by

Value: 1

Phase shift

Phase:  Plus  Minus

Value: 0.0 Units: ns

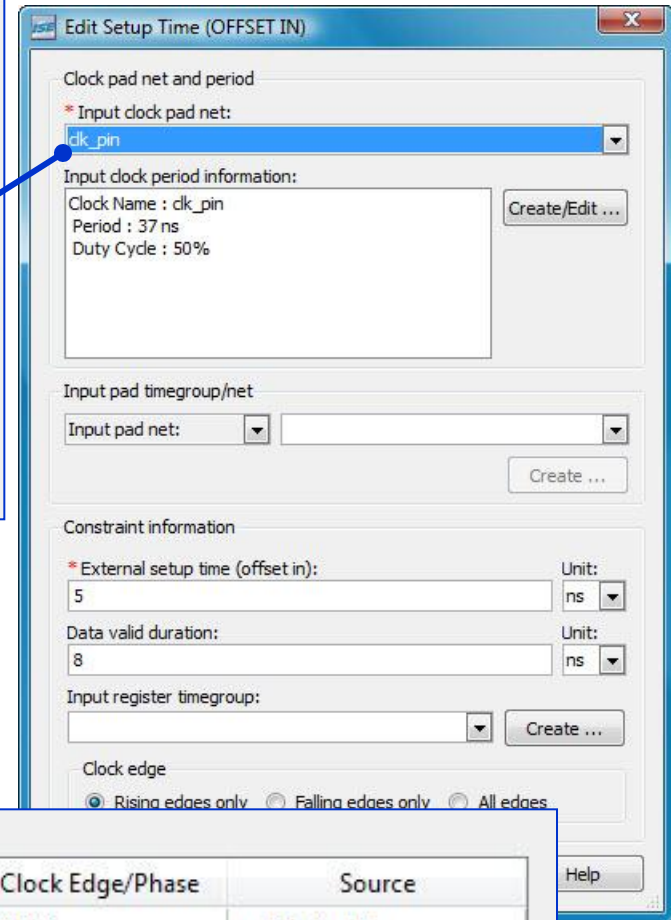
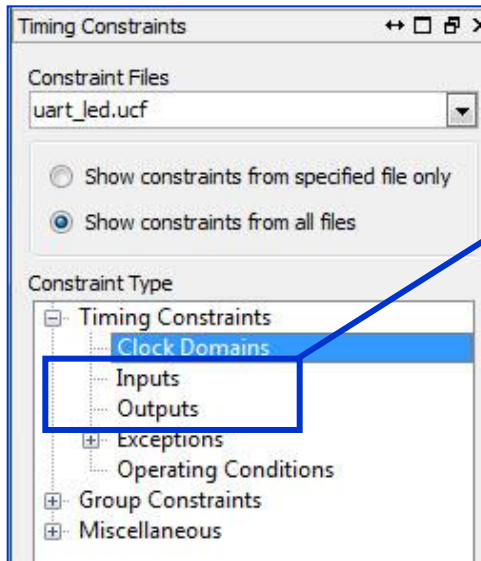
Input jitter: Units: ps

Priority:

OK Close Create Help

# Entering OFFSET Constraints

- Global OFFSET IN and OFFSET OUT constraints can be made from Inputs or Outputs
- Right-click here and select **Create Constraint** to make an OFFSET constraint

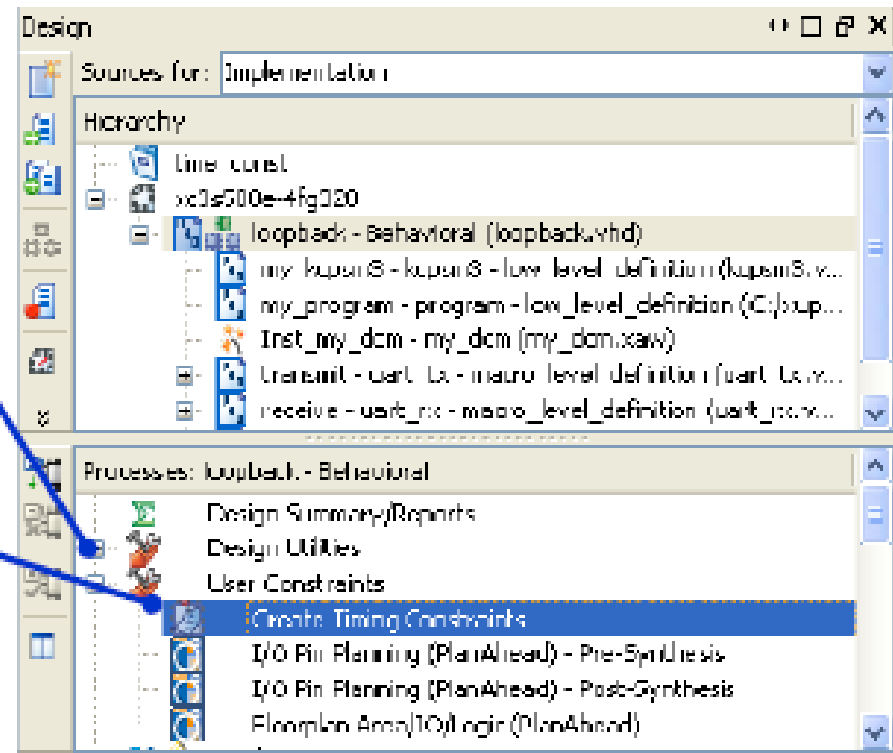


Pad Group	Port	Value *	Valid Duration	Clock *	Register Group	Clock Edge/Phase	Source
1		5 ns	8 ns	clk_pin		RISING	uart_led.ucf
2							

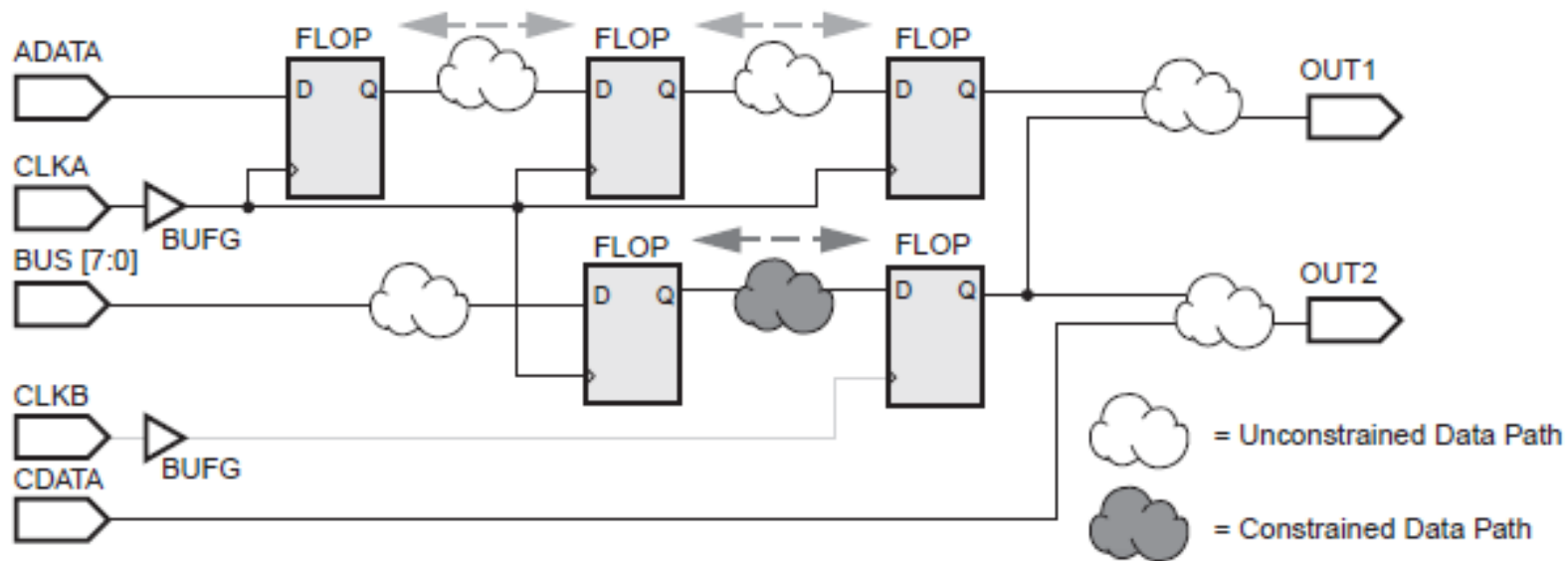
# Access the Constraints Editor

Enter constraints in the Constraints Editor GUI

- Expand **User Constraints** in the Processes for Source window
- Double-click **Create Timing Constraints**



```
NET "clk" TNM_NET = clk;  
TIMESPEC TS_clk = PERIOD "clk" 20 ns HIGH 50%;  
OFFSET = IN 7 ns VALID 20 ns BEFORE "clk" RISING;  
OFFSET = OUT 7.5 ns AFTER "clk";
```

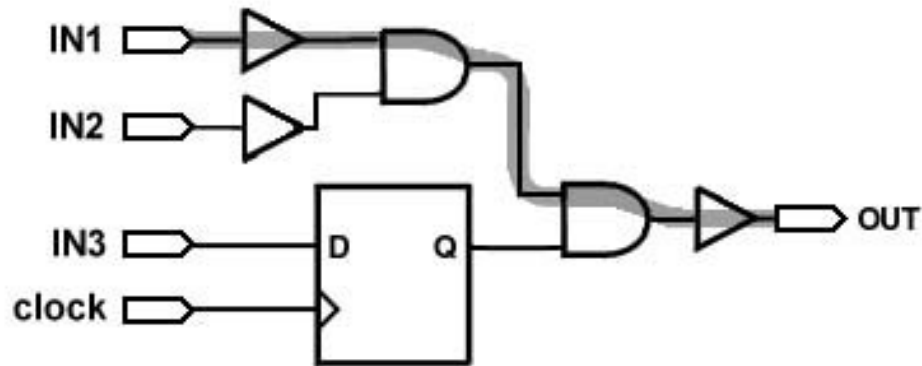


TIMESPEC TS\_clk1\_to\_clk2 = FROM clk1 TO clk2 8 ns;  
 Constrain from time group **clkA** to time group **clkB** to be 8 ns.



# Pad to Pad Constraints

Covers Purely Combinatorial Paths that start and end at I/O pads



TIMESPEC TS\_Pad2Pad = FROM PADS TO PADS 14.4 ns;

# Latches

- Level-sensitive storage
  - Data transmitted while enable is '1'
    - *transparent* latch
  - Data stored while enable is '0'

