北京郵電大學
Beijing University of Posts and Telecommunications

片上集成系统分析与设计
System-on-Chip Design Methodology

集成电路学院

陈黎明

2024年11月

第六章

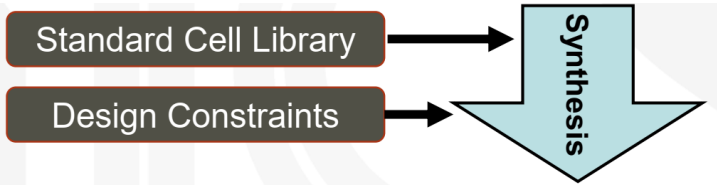
逻辑综合与时序分析

目 录

第一节 逻辑综合基本概念

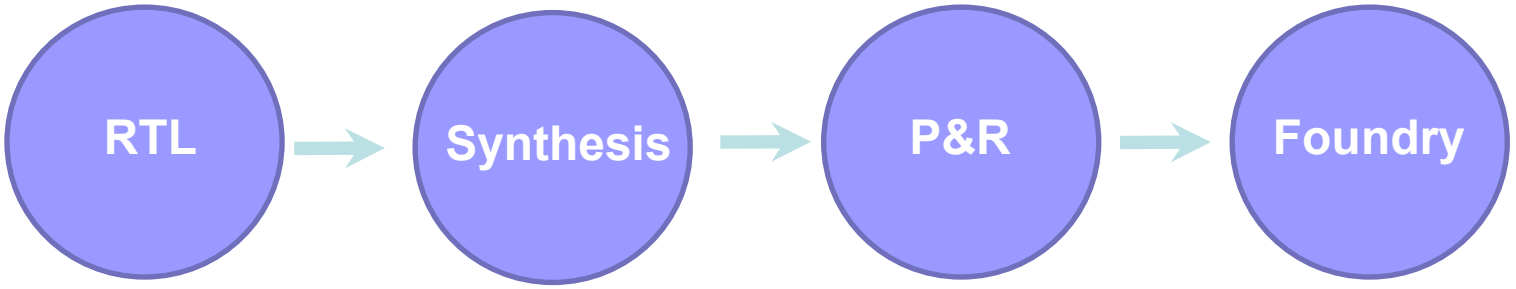
- 逻辑综合： Logic Synthesis
  - 将RTL转换为基于目标工艺的门级网表
  - 同时满足预先设定的约束。
- 输入
  - RTL代码
  - 标准单元库standard cell library
  - 设计约束constraints
- 输出
  - 基于目标工艺库的门级网表netlis
  - 满足性能、功耗和面积约束
- Synopsys DesignCompiler

```
module counter(
     input clk, rstn, load,
     input  [1:0] in,
     output reg [1:0] out);
always @(posedge clk)
  if (!rstn) out <= 2'b0;
  else if (load) out <= in;
  else out <= out + 1;
endmodule
```

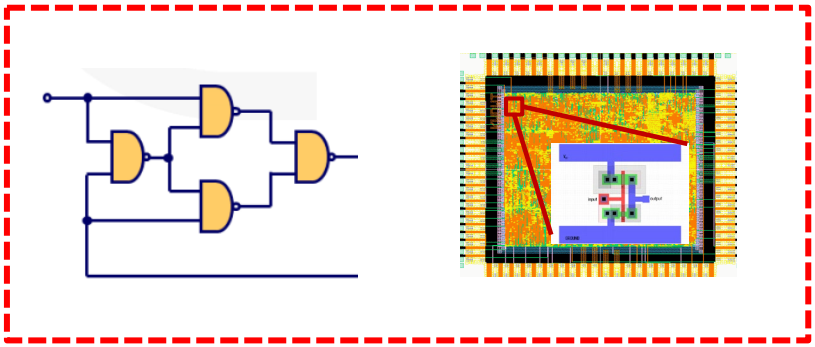


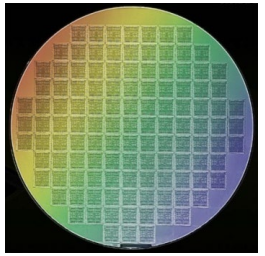

```
module counter ( clk, rstn, load, in, out );
  input [1:0] in;
  output [1:0] out;
  input clk, rstn, load;
  wire   N6, N7, n5, n6, n7, n8;

  FFPQ1 out_reg_1 (.D(N7),.CK(clk),.Q(out[1]));
  FFPQ1 out_reg_0 (.D(N6),.CK(clk),.Q(out[0]));
  NAN2D1 U8 (.A1(out[0]),.A2(n5),.Z(n8));
  NAN2D1 U9 (.A1(n5),.A2(n7),.Z(n6));
  INVD1 U10 (.A(load),.Z(n5));
  OA211D1 U11 (.A1(in[0]),.A2(n5),.B(rstn),.C(n8),.Z(N6));
  OA211D1 U12 (.A1(in[1]),.A2(n5),.B(rstn),.C(n6),.Z(N7));
  EXNOR2D1 U13 (.A1(out[1]),.A2(out[0]),.Z(n7));
endmodule
```

为什么要逻辑综合?




```
module counter(
     input clk, rstn, load,
     input  [1:0] in,
     output reg [1:0] out);
always @(posedge clk)
  if (!rstn) out <= 2'b0;
  else if (load) out <= in;
  else out <= out + 1;
endmodule
```



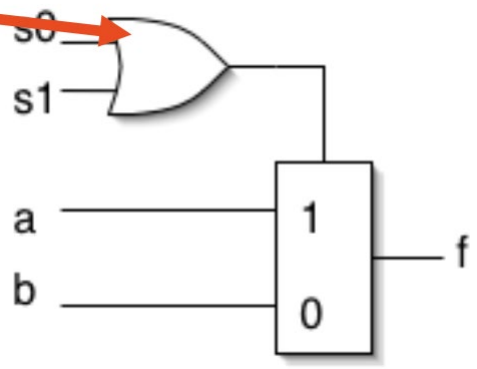


基于目标工艺的netlist和Layout

- 由EDA完成逻辑综合，省时省力，保证不出错
- RTL设计时不关心目标工艺，移植性强
- 比人工做，结果更优化，尤其设计空间巨大时
- 小规模电路，人工做也许更好

为什么要逻辑综合?

```
module foo (a,b,s0,s1,f);
 input [3:0] a;
 input [3:0] b;
 input s0,s1;
 output [3:0] f;
 reg f;

 always @ (a or b or s0 or s1)
   if (!s0 && s1 || s0)
       f=a;
   else
       f=b;
endmodule
```



逻辑综合的目标

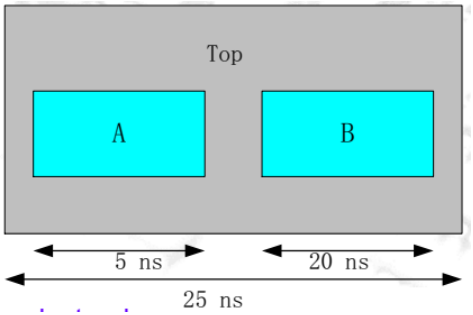
- **Minimize area**
  - In terms of literal count, cell count, register count, etc.
- **Minimize power**
  - In terms of switching activity in individual gates, deactivated circuit blocks, etc.
- **Maximize performance**
  - In terms of maximal clock frequency of synchronous systems, throughput for asynchronous systems
- **Any combination of the above**
  - Combined with different weights
  - Formulated as a constraint problem
    - Minimize area for a clock speed > 300MHz
- **More global objectives**
  - Feedback from layout
    - Actual physical sizes, delays, placement and routing

逻辑综合Strategy: Top-dwon

- **The top-level design and all its sub design are compiled together**
- **Advantages:**
  - Only top level constraints are needed
  - Better results due to optimization across entire design
- **Disadvantages**
  - Long compile times
  - Incremental changes to the sub-blocks require complete resynthesis
  - Tool limit for handle “large” design

逻辑综合Strategy: Bottom-up

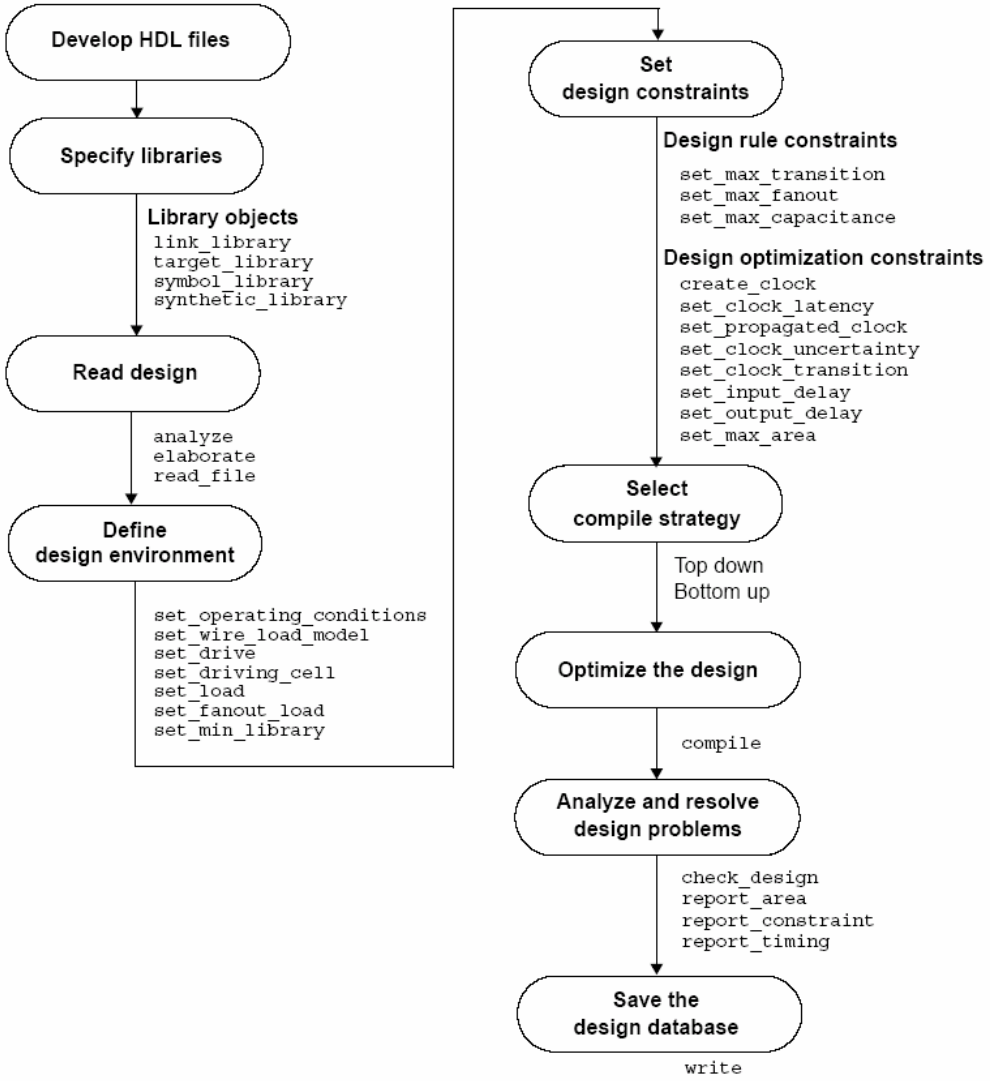
- **The individual sub designs are compiled separately**
  - You may need to do Time-budgeting at top level to get constraint file for each sub design
- **Starting from the bottom of the hierarchy and proceeding up through the levels of the hierarchy until the top-level design is compiled**
- **Advantages**
  - Easier to manage the design because of individual scripts
  - Good quality results in general because of flexibility in targeting and optimizing individual blocks
- **Disadvantages**
  - Tedious to update and maintain multiple scripts
  - Critical paths seen at the top level may not be critical at lower level



逻辑综合优化设计

- Architectural optimization
  - e.g. Selecting DesignWare implementations
- Logic-Level optimization
  - Structuring -- for reduced design area
  - Flattening -- for speed optimization
- Gate-Level optimization
  - Mapping
  - Delay optimization
  - Design rule fixing
  - Area optimization
- Timing correction is most effective with placement information
  - E.g., Physical synthesis

逻辑综合流程Flow


Develop HDL files
Specify libraries
Library objects
link_library
target_library
symbol_library
synthetic_library
Read design
analyze
elaborate
read_file
Define
design environment
set_operating_conditions
set_wire_load_model
set_drive
set_driving_cell
set_load
set_fanout_load
set_min_library
Set
design constraints
Design rule constraints
set_max_transition
set_max_fanout
set_max_capacitance
Design optimization constraints
create_clock
set_clock_latency
set_propagated_clock
set_clock_uncertainty
set_clock_transition
set_input_delay
set_output_delay
set_max_area
Select
compile strategy
Top down
Bottom up
Optimize the design
compile
Analyze and resolve
design problems
check_design
report_area
report_constraint
report_timing
Save the
design database
write

第二节 工艺库

Libraries for Setup

- **Target_library**
  - The technology libraries which contain standard cells that Design Compiler maps to during optimization
- **Link_library**
  - All technology libraries which contain hardmacros, standard cells that Design Compiler uses to resolve cell references
- **Synthetic_library**
  - A DesignWare library is a collection of reusable circuit-design building blocks (component). Technology independent.
- **Symbol_library**
  - Symbol libraries contain definitions of the graphic symbols that represent library cells in the design schematics.

Using Search Path

- Multiple db paths searching
  - set search_path [list path_a \
    path_b \
    path_c]
- With multiple db link library
  - set link_library [list * \
    link_a.db \
    link_b.db \
    link_c.db]

标准单元库Standard cells library

- The library definition stage tells the synthesizer where to look for leaf cells for binding and the target library for technology mapping.
  - We can provide a list of paths to search for libraries in

```
set ADDITIONAL_SEARCH_PATH        "     .  \
                                ../../../lib/SMIC180/SCC018UG_UHD_RVT_V0p4a/liberty/1.8v       \
                                ../../../lib/SMIC180/SP018_V1.5b/SP018_V1p5b/SP018_V1p5b/syn   \
                                ../../../lib/SMIC180/MEM_IP_20MHz/db"
```

  - And we have to provide the name of a specific library, usually characterized for a single corner:

```
set TARGET_LIBRARY_FILES          "scc018ug_uhd_rvt_ss_v1p62_125c_basic.db             \
                                              SP018_V1p4_max.db
                                     "  ;
```

- We also need to provide .lib files for IPs, such as memory macros, I/Os, and others.

```
set ADDITIONAL_LINK_LIB_FILES     "
                                              RA1SHD8192X64_slow_syn.db       \
                                              RA1SHD8192X32_slow_syn.db
                                               " ;
```

Standard cells library是什么？

Standard cell例子

```
`timescale 1ns/1ps
`celldefine
module NAND2X4MTH (Y, A, B);
output Y;
input A, B;

  nand (Y, A, B);

specify
if (B==1'b1)
(A => Y) = (`ARM_PROP_DELAY,`ARM_PROP_DELAY);
if (A==1'b1)
(B => Y) = (`ARM_PROP_DELAY,`ARM_PROP_DELAY);

endspecify
endmodule // NAND2X4MTH
```

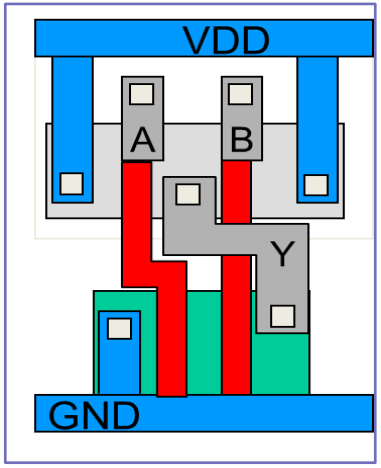
verilog model (.v)

```
  rise_power(pwr_tin_oload_7x7) {
    index_1("0.00798, 0.0657587, 0.268575, 0.655881, 1.25928, \
             2.10593, 3.22");
    index_2("0.0001, 0.00433474, 0.0191996, 0.0475863, \
             0.0918112, 0.153864, 0.235517");
    values("0.0117772, 0.0118517, 0.0116508, 0.0113309, 0.0106195, 0.0105827, 0.00929425",\
           "0.011714, 0.0117947, 0.0115816, 0.011275, 0.0110544, 0.0105073, 0.00907154",\
           "0.0115482, 0.0116591, 0.0114293, 0.0110867, 0.0109354, 0.0103546, 0.00900365",\
           "0.0115279, 0.0115298, 0.0114435, 0.01094, 0.0108024, 0.00949909, 0.0097665",\
           "0.0116652, 0.0116761, 0.0116072, 0.0109595, 0.0108039, 0.0103089, 0.00989486",\
           "0.01198, 0.0119591, 0.0119525, 0.0116753, 0.0109602, 0.00954039, 0.00911302",\
           "0.0124893, 0.0124372, 0.0123762, 0.0123677, 0.0119291, 0.0098829, 0.00914072");
  }
}
```

power model (.lib)

```
 cell_rise(tmg_ntin_oload_7x7) {
   index_1("0.00798, 0.0657587, 0.268575, 0.655881, 1.25928, \
            2.10593, 3.22");
   index_2("0.0001, 0.00433474, 0.0191996, 0.0475863, \
            0.0918112, 0.153864, 0.235517");
   values("0.353231, 0.395024, 0.513235, 0.726838, 1.05711, 1.51907, 2.12734",\
          "0.377289, 0.419056, 0.537257, 0.750857, 1.08111, 1.54313, 2.15129",\
          "0.442568, 0.484422, 0.60243, 0.815533, 1.14612, 1.60789, 2.21648",\
          "0.525777, 0.567669, 0.685418, 0.89842, 1.22859, 1.69039, 2.29865",\
          "0.601322, 0.643303, 0.761057, 0.973391, 1.30302, 1.7659, 2.37415",\
          "0.666823, 0.709, 0.826627, 1.03904, 1.36811, 1.82982, 2.43804",\
          "0.721588, 0.763783, 0.881251, 1.09362, 1.42253, 1.88416, 2.49111");
 }
```

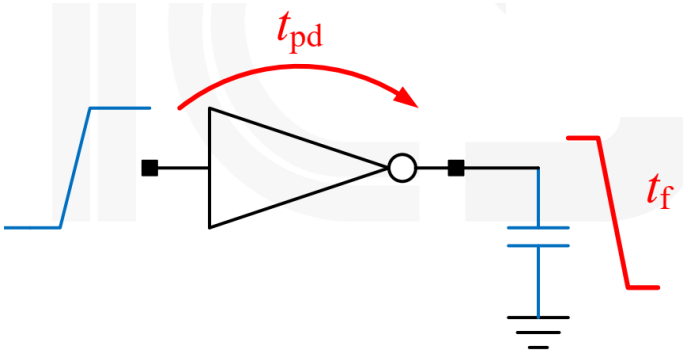
Timing model (.lib)



Layout

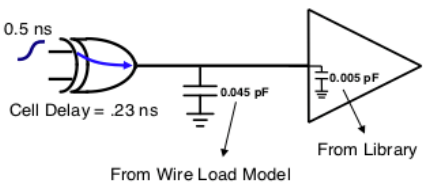
How to calculate the gate delay?

- How do we know the delay through a gate in a logic path?
  - Running SPICE is way too complex.
  - Instead, create a timing model that will simplify the calculation.
- Goal
  - For every timing arc, calculate:
    - Propagation Delay (tpd)
    - Output transition (trise, tfall)
  - Based on:
    - Input net transition (trise, tfall)
    - Output Load Capacitance (Cload)



How to calculate the gate delay?

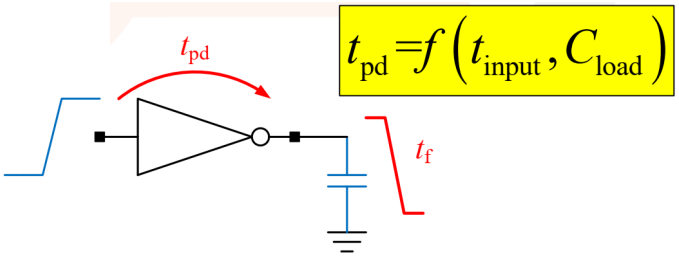
- Non-Linear Delay Model (NLDM)
  - Driver model:
    - Ramp voltage source
    - Fixed drive resistance
  - Receiver model:
    - Min/max rise/fall input caps
  - Very fast
  - Doesn’t model cap variation during transition.
  - Loses accuracy beyond 130nm



SPICE	Output Load (pF)			
	.005	.05	.10	.15
Input Trans (ns) 0.0	.1	.15	.2	.25
0.5	.15	.23	.3	.38
1.0	.25	.4	.55	.75

Cell Delay (ns)

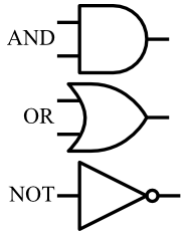
```
lu_table_template(delay_template_5x5) {
  variable_1 : input_net_transition;
  variable_2 : total_output_net_capacitance;
  index_1 ("1000.0, 1001.0, 1002.0, 1003.0, 1004.0");
  index_2 ("1000.0, 1001.0, 1002.0, 1003.0, 1004.0");
}
```



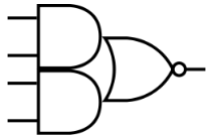

```
cell (INVX1) {
  pin(Y) {
    timing() {
      cell_rise(delay_template_5x5) {
        values ( \
          "0.147955, 0.218038, 0.359898, 0.922746, 1.76604", \
          "0.224384, 0.292903, 0.430394, 0.991288, 1.83116", \
          "0.365378, 0.448722, 0.584275, 1.13597, 1.97017", \
          "0.462096, 0.551586, 0.70164, 1.24437, 2.08131", \
          "0.756459, 0.874246, 1.05713, 1.62898, 2.44989"); }
```

library一般有哪些cell

- Inverters and Buffer Cells of various strengths
- Cells with basic logic functions
  - AND　　NAND　　AND_OR
  - OR　　NOR　　OR_AND
  - XOR　　XNOR　　MUX



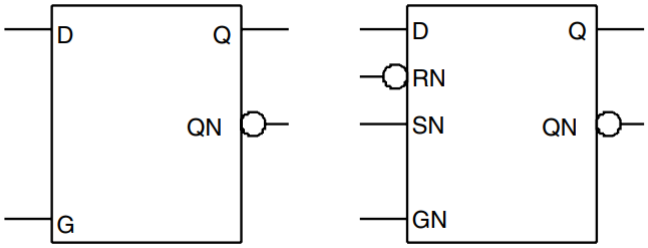
  - Various input configurations of each
    - Two, three or four input cells
    - Example: NAND2, NAND3 and NAND4 cells



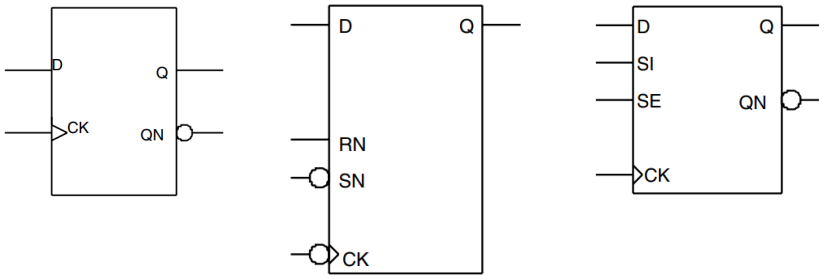
  - Various output drive configurations of each
    - Drive strengths of one, two and four
    - Example: NAND2X1, NAND2X2 and NAND2X4
  - Inverted and non-inverted output versions

library一般有哪些cell

- Latches with various combinations of:
  - Output drive strengths 1, 2 or 4
  - Active High or Low Enable
  - Q and / or QB outputs
  - Clear or no Clear
  - Preset or no Preset

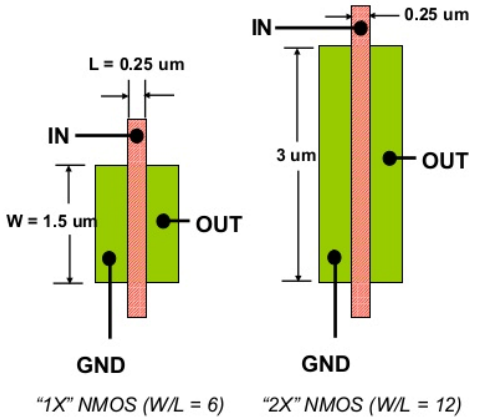


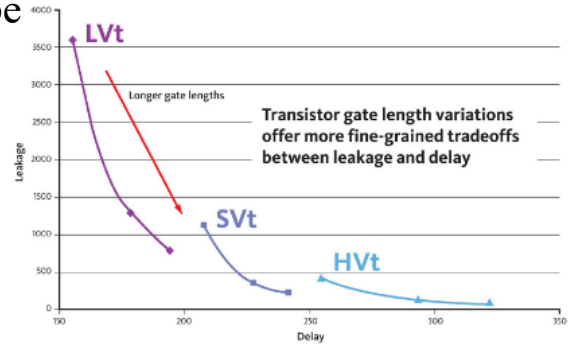
- D-Flip-Flops with various combinations of:
  - Output drive strengths 1, 2 or 4
  - Rising Edge
  - Clear or no Clear
  - Preset or no Preset
  - Scan Logic or no Scan Logic



Multiple Drive Strengths and VTs

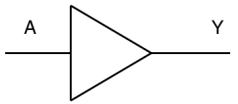
- **Multiple Drive Strength**
  - Each cell will have various sized output stages
  - Larger output stage → better at driving fanouts/loads
  - Smaller drive strength → less area, leakage, input cap
  - Often called X2, X3, or D2, D3, etc
- **Multiple Threshold (MT-CMOS)**
  - A single additional mask can provide more or less doping in a transistor channel, shifting the threshold voltage
  - Most libraries provide equivalent cells with three or more VTs: SVT, HVT, LVT. This enables tradeoff between speed vs. leakage
  - All threshold varieties have same footprint and therefore can be swapped without any placement/routing iterations.

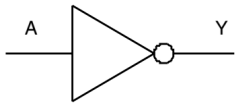


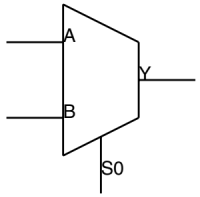


Clock cells

- **CLKBUF cell**
  - provides the logical buffer of a single input (A), with balanced delays for clock signals
- **CLKINV cell**
  - provides the logical inversion of a single input (A), with balanced delays for clock signals
- **CLKMX2 cell**
  - The CLKMX2 cell is a non-inverting 2-to-1 multiplexer with balanced delays for clock signals
- **CLKANAD2**
- **CLKXOR2**







Need to know everything?

- **So, what is a cell?**
  - I guess that the detailed layout is sufficient to know (guess) anything and everything about a standard cell.
  - Or it would be easier, if we got the whole Open Access database of the cell…
- **But do we really need to know everything?**
  - For example, does logic simulation need to know if your inverter is CMOS or Pseudo-NMOS?
  - And does a logic synthesizer need to know what type of transistors you used?
- **No!**
  - To make life (and calculations) simpler, we will abstract away this info.
  - Each tool will get only the data it really needs.

Technology Library Files

- **Verilog models (.v)**
  - Used for gate level simulation
  - Used for netlist generation of design
- **Timing and power Information (.lib)**
  - Allows synthesis tool to do timing optimization
- **Layout Information (.LEF .GDSII)**
  - Allows synthesis tool to do area optimization
  - Also used for place & route tool
- **Transistor Information (.CDL)**
  - Spice/Spectre netlist for LVS, transistor-level simulation
- **Documentation and Datasheets**

Figure 66-2 Functional Schematic

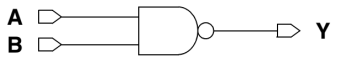


Table 66-2 Cell Size

Cell	Height (µm)	Width (µm)
NAND2X1MTH	2.87	1.64
NAND2X2MTH	2.87	1.64
NAND2X3MTH	2.87	2.46

Table 66-1 Function Table

A	B	Y
0	x	1
x	0	1
1	1	0

Table 66-5 Power - tt_typical_max_1p20v_25c, 25.0°C, VNW 1.2V, VDD 1.2V, VSS 0V, VPW 0V

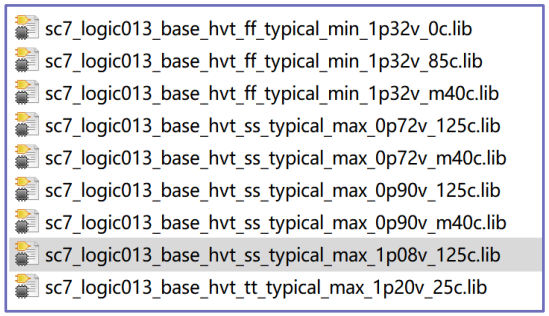
Path	Condition	Power Level	Power (µW/MHz)		
			X1M	X2M	X3M
A → Y	B	n/a	0.002499790	0.003708175	0.004894670
B → Y	A	n/a	0.003052903	0.004668434	0.006346265

Table 66-4 Delay - tt_typical_max_1p20v_25c, 25.0°C, VNW 1.2V, VDD 1.2V, VSS 0V, VPW 0V

Description	Condition	Intrinsic Delay (ns)			
		X1M	X2M	X3M	X4M
A → Y ↑	B	0.027148	0.028417	0.025229	0.025332
A → Y ↓	B	0.029729	0.024451	0.025222	0.022006
B → Y ↑	A	0.033363	0.036105	0.033125	0.034492
B → Y ↓	A	0.033858	0.028220	0.030308	0.027023

SMIC 130nm Technology Library

SMIC LOGIC013 1.2 Volt SC7 Ultra High Density Base HVT Standard Cell Library




```
library(sc7_logic013_base_hvt_ss_typical_max_1p08v_125c) {
  nom_process : 1 ;
  nom_temperature : 125 ;
  nom_voltage : 1.08 ;

  lu_table_template(tmg_ntin_oload_7x7) {
    variable_1 : input_net_transition ;
    variable_2 : total_output_net_capacitance ;
    index_1("1, 2, 3, 4, 5, 6, 7");
    index_2("1, 2, 3, 4, 5, 6, 7");
  }
```

```
fall_transition(tmg_ntin_oload_7x7) {
  index_1("0.00798, 0.0657587, 0.268575, 0.655881, 1.25928, \
          2.10593, 3.22");
  index_2("0.0001, 0.0249215, 0.11205, 0.278435, 0.537655, \
          0.90137, 1.37997");
  values("0.0139613, 0.0389668, 0.127045, 0.294918, 0.557105, 0.924613, 1.40792",\
         "0.0228581, 0.0417289, 0.126902, 0.295064, 0.557142, 0.922862, 1.40829",\
         "0.0539915, 0.0863082, 0.155019, 0.296396, 0.556861, 0.924429, 1.40778",\
         "0.0964878, 0.146906, 0.247907, 0.373954, 0.572477, 0.923423, 1.40827",\
         "0.147687, 0.22051, 0.358588, 0.524895, 0.714565, 0.978304, 1.41192",\
         "0.203745, 0.304989, 0.485177, 0.697598, 0.932012, 1.19329, 1.52739",\
         "0.267275, 0.398977, 0.627578, 0.887839, 1.17046, 1.4832, 1.82194");
}
```

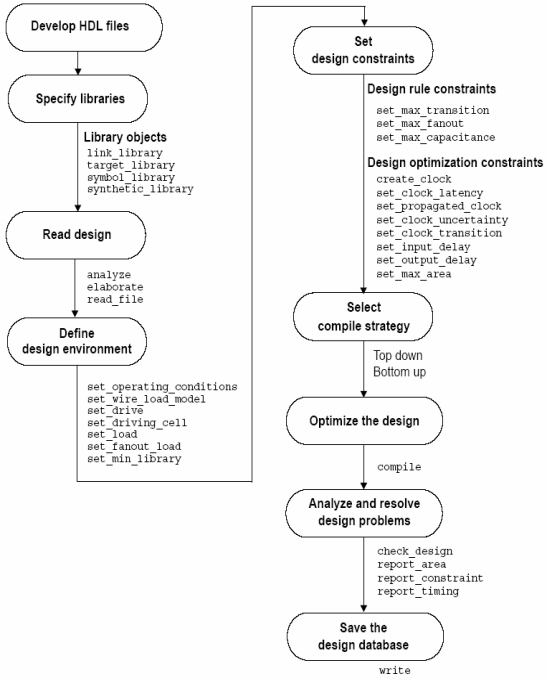
```
pin(Y) {
  direction : output ;
  function : "(!A)" ;
  max_capacitance : 1.37997 ;
  max_transition : 3.22 ;
  min_capacitance : 0.0001 ;
  output_voltage : default ;
  related_ground_pin : VSS ;
  related_power_pin : VDD ;
  power_down_function : "!VDD + VSS" ;

  timing() {
    related_pin : "A" ;
    timing_sense : negative_unate ;
    timing_type : combinational ;

    cell_fall(tmg_ntin_oload_7x7) {
      index_1("0.00798, 0.0657587, 0.268575, 0.655881, 1.25928, \
              2.10593, 3.22");
      index_2("0.0001, 0.0249215, 0.11205, 0.278435, 0.537655, \
              0.90137, 1.37997");
      values("0.0170721, 0.0317859, 0.0821555, 0.178041, 0.32753, 0.537185, 0.812992",\
             "0.0347173, 0.0537646, 0.104679, 0.20077, 0.350319, 0.559796, 0.835919",\
             "0.060298, 0.101539, 0.18069, 0.280643, 0.430275, 0.640067, 0.916022",\
             "0.0804321, 0.147875, 0.27594, 0.421854, 0.583278, 0.792984, 1.06913",\
             "0.091911, 0.188333, 0.371295, 0.577381, 0.797069, 1.03049, 1.30739",\
             "0.093878, 0.220015, 0.462598, 0.734747, 1.0224, 1.3234, 1.63818",\
             "0.0854083, 0.241359, 0.546688, 0.890597, 1.2517, 1.62716, 2.01639");
    }
```

Read Design

- analyze -format verilog ${RTL_SOURCE_VERILOG}
- analyze -format VHDL ${RTL_SOURCE_VHDL}
- elaborate ${DESIGN_NAME}
- check out the warning and error information

```
set RTL_SOURCE_VHDL   "
                        ../pre_sim/rtl/i2s/i2s_codec.vhd        \
                        ../pre_sim/rtl/i2s/rx_i2s_top.vhd       \
                        ../pre_sim/rtl/i2s/tx_i2s_top.vhd       \
```

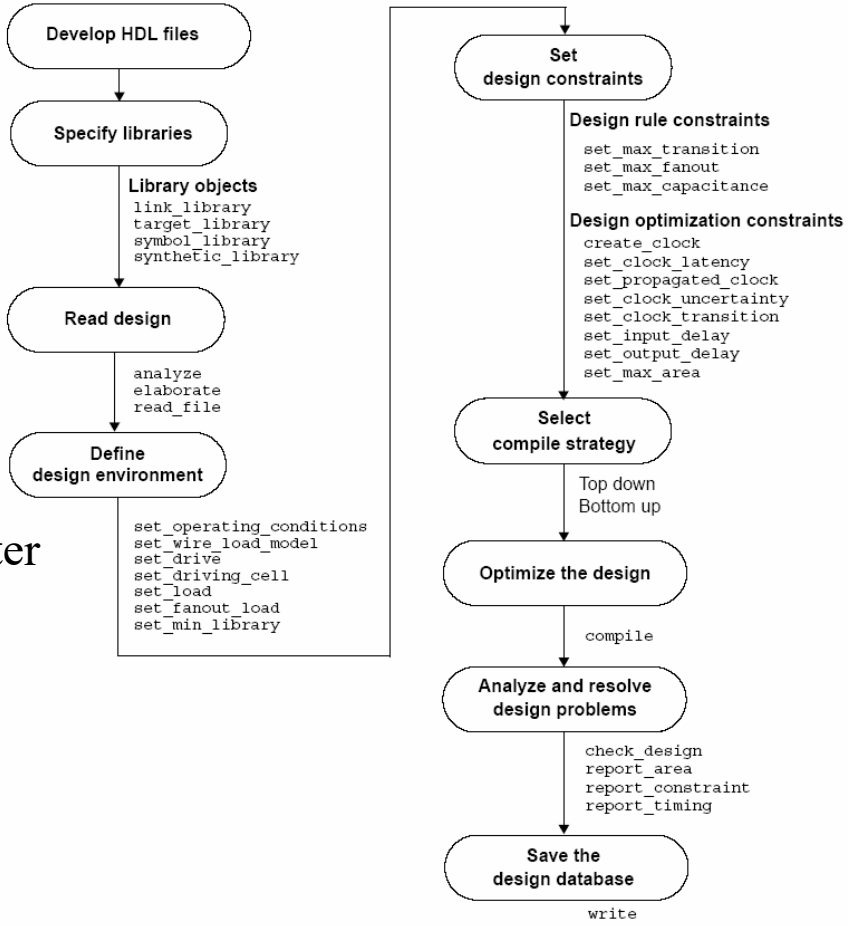
```
set RTL_SOURCE_VERILOG                    "
                      ../pre_sim/rtl/e203/soc/full_chip.v             \
                      ../pre_sim/rtl/e203/core/e203_biu.v             \
                      ../pre_sim/rtl/e203/core/e203_clk_ctrl.v        \
                      ../pre_sim/rtl/e203/core/e203_clkgate.v         \
                      ../pre_sim/rtl/e203/core/e203_core.v            \
                      ../pre_sim/rtl/e203/core/e203_cpu_top.v         \
```



第三节 逻辑综合约束

面积Area约束

- *set_max_area value*
- To set max are to be 0
  - run-time is not a concern
  - Performing area optimization only after timing optimization halt



Design Rule Constraints

- You can set specific design rules that should be met, for example:
  - Maximum transition through a net

```
set_max_transition $MAX_TRAN_IN_NS
```

  - Maximum Capacitive load of a net

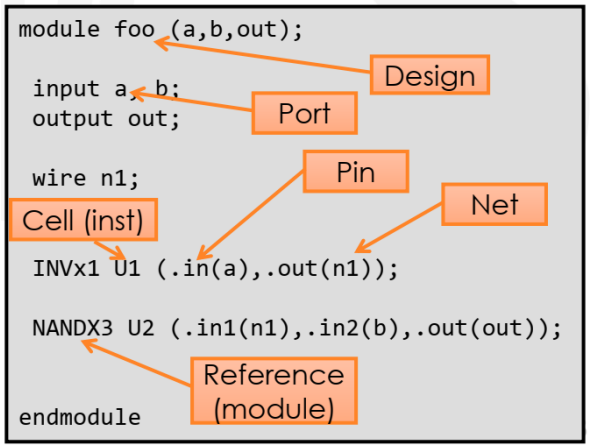
```
set_max_capacitance $MAX_CAP_IN_PF
```

  - Maximum fanout of a gate

```
set_max_fanout $MAX_FANOUT
```

设计约束

- **Design Objects：**
  - **Design:** A circuit description that performs one or more logical functions (i.e Verilog module).
  - **Cell:** An instantiation of a design within another design (i.e Verilog instance).
  - **Reference:** The original design that a cell "points to" (i.e Verilog sub-module)
  - **Port:** The input, output or inout port of a Design.
  - **Pin:** The input, output or inout pin of a Cell in the Design.
  - **Net:** The wire that connects Ports to Pins and/or Pins to each other.
  - **Clock:** Port of a Design or Pin of a Cell explicitly defined as a clock source.



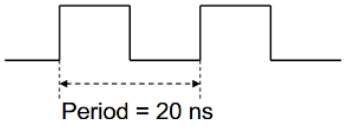
设计约束

- **useful commands**
- **“get” commands:**
  - [get_ports string] – returns all ports that match string.
  - [get_pins string] – returns all cell/macro pins that match string.
  - [get_nets string] – returns all nets that match string.
  - get_clocks, get_cells, get_designs...
- **“all” commands:**
  - [all_inputs] – returns all the primary inputs (ports) of the block.
  - [all_outputs] – returns all the primary outputs (ports) of the block.
  - [all_registers] – returns all the registers in the block.
  - all_clocks, all_registers, all_dont_touch...

时序约束——Clock定义

- 时钟定义Clock Definitions
  - Where does the clock come from? (i.e., input port, or PLL, etc.)
  - What is the clock period? (=operating frequency)
  - What is the duty-cycle of the clock?

```
create_clock -period 20 -name my_clock [get_ports clk]
```



- Can there be more than one clock in a design?
  - Yes, but be careful about clock domain crossings!
  - If a clock is produced by a clock divider, define a “generated clock”:

```
create_generated_clock -name gen_clock \
       -source [get_ports clk] -divide_by 2 [get_pins FF1/Q]
```

时序约束——Clock定义

- 逻辑综合期间，假设时钟是理想的ideal，驱动能力无穷大

```
set_ideal_network [get_ports clk]
```

- 为了模拟真实时钟，设置transition:

```
set_clock_transition 0.2 [get_clocks my_clock]
```

- 还会增加一些uncertainty，即jitter或skew

```
set_clock_uncertainty 0.2 [get_clocks my_clock]
```

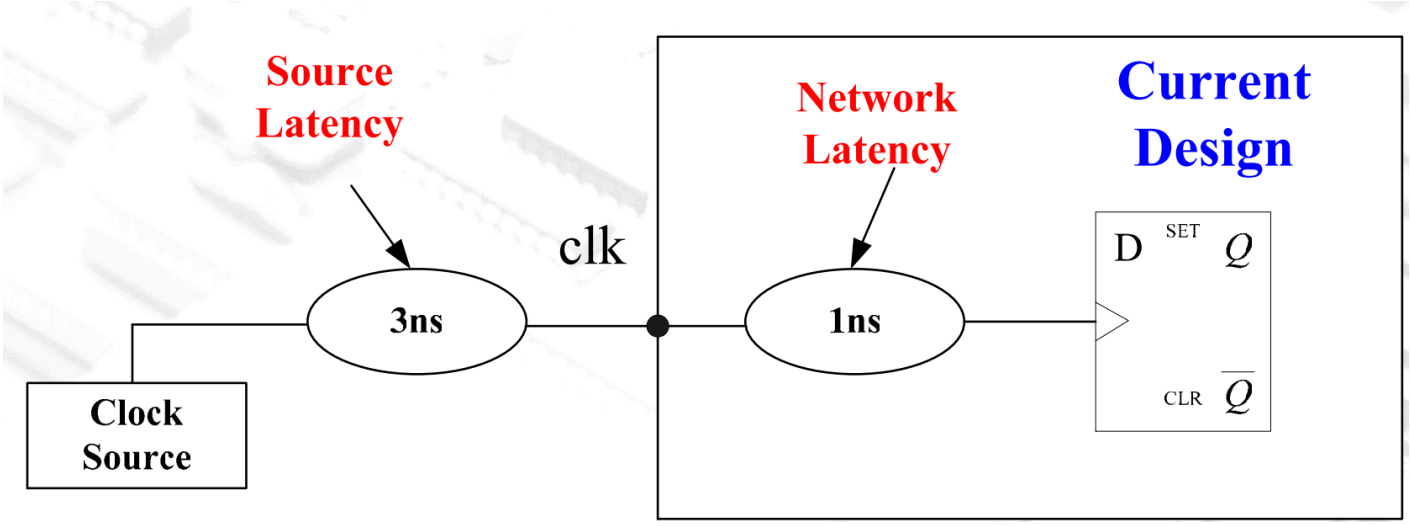
- 后端布局布线后，做STA，要设置为真实的时钟网络

```
set_propagated_clock [get_clocks my_clock]
```

时序约束——Clock定义

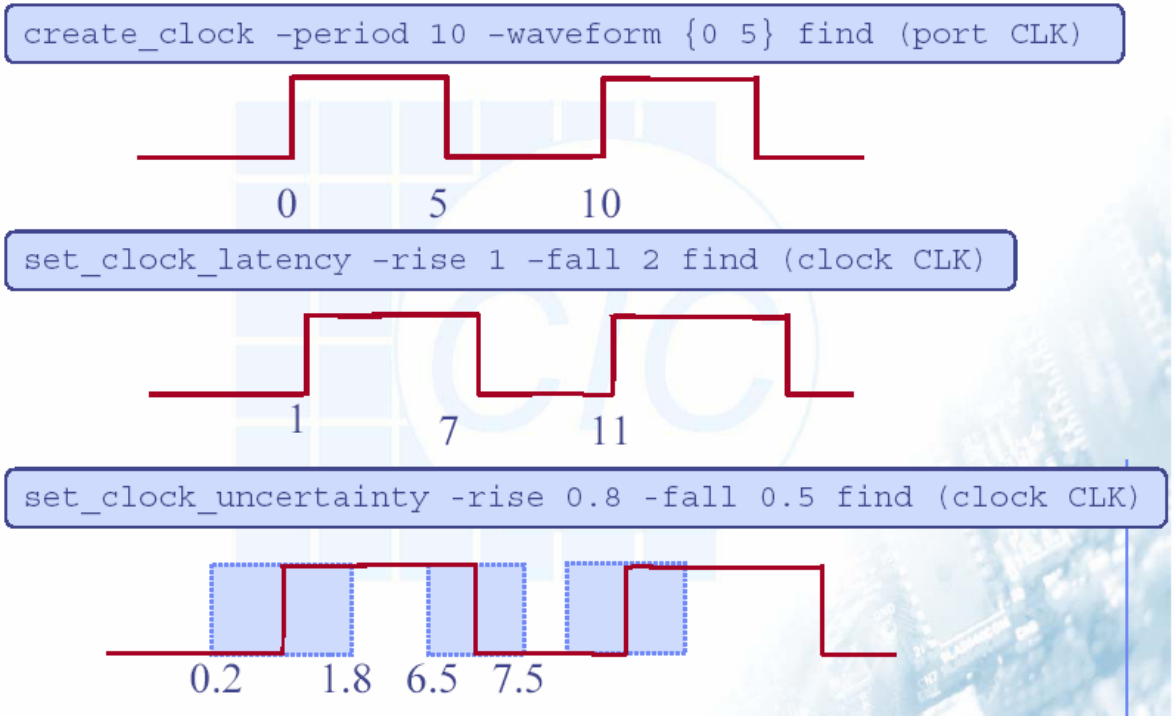
- 设置时钟网络延时

```
set_clock_latency –source –max 3 [get_clock CLK]
set_clock_latency –network –max 1 [get_clock CLK]
```



时序约束——Clock定义

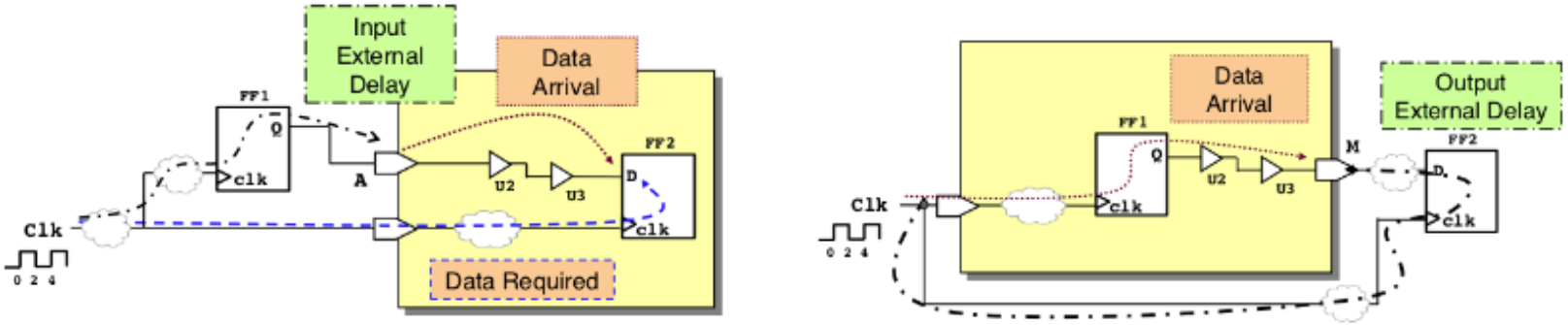
- 例子



时序约束——I/O

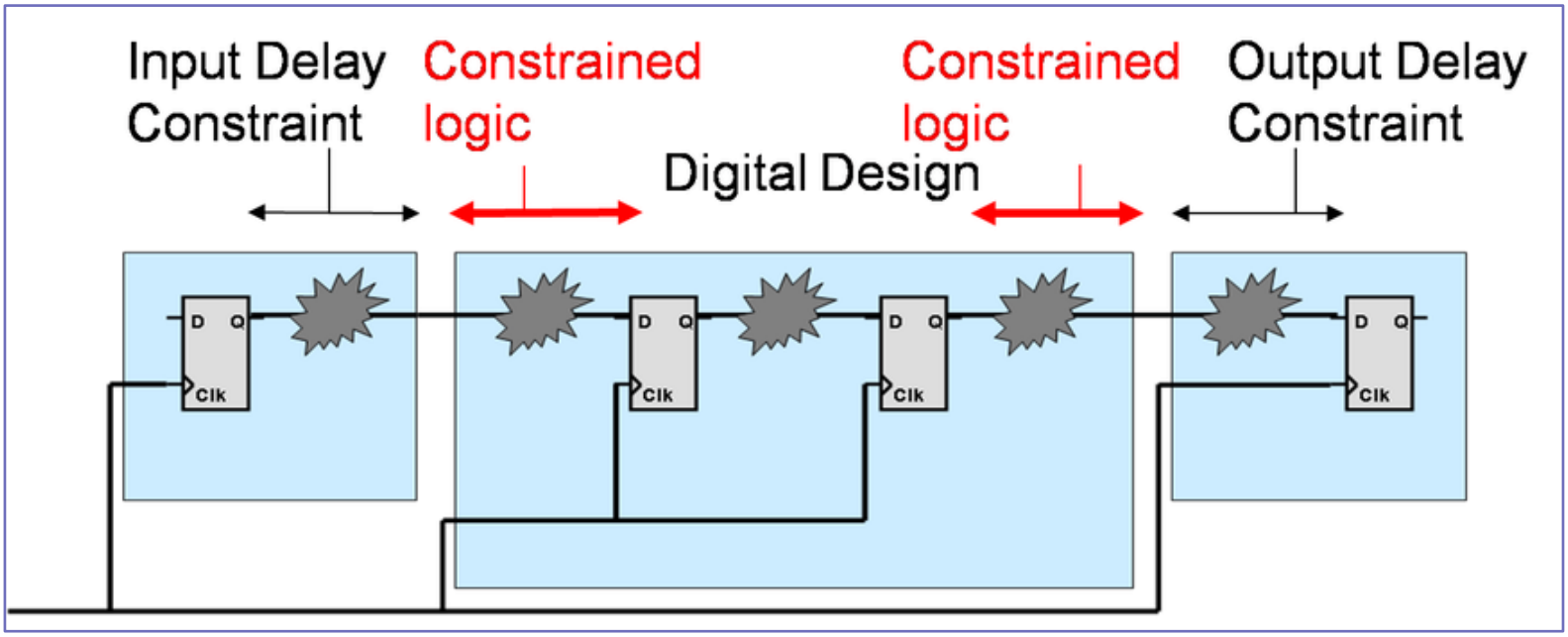
- Now that the clock is defined, reg2reg paths are constrained.
- However, what about in2reg, reg2out, and in2out paths?
  - First, what clock toggles an I/O port?
  - And what about the time needed outside the chip?
- Define I/O constraints:
  - Input and output delays model the length of the path outside the block

```
set_input_delay 0.8 -clock clk \
       [remove_from_collection [all_inputs] [get_ports clk]]
set_output_delay 2.5 -clock clk [all_outputs]
```



时序约束——I/O

- **Input Delay** – An input delay is the specification of an arrival time at input port relative to a clock edge.
- **Output delay** – An output delay represents an external timing path from an output port to a register



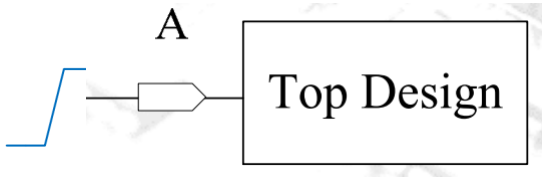
设计环境Design Environment

- **Need to take into account**
  - Input drivers/transition times
  - Output loading
  - PVT corners
  - Parasitic RCs

设计环境Design Environment

- Input Transition

```
create_clock       –name   CLK    –period 2          [get_ports clk]
set_input_delay    –max    0.6    –clock    CLK      [get_ports A]
set_output_delay   –max    0.8    –clock    CLK      [get_ports A]
set_input_transition       0.12   [get_ports A]
```

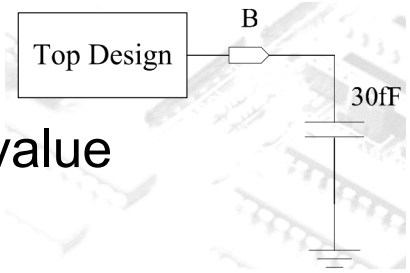


transition time

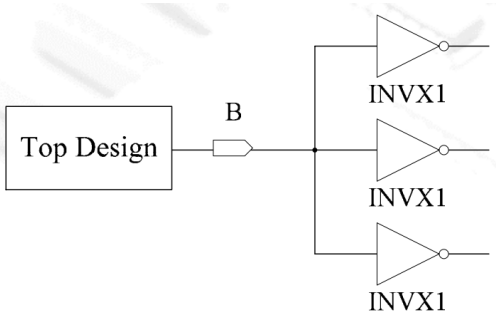
设计环境Design Environment

- Capacitive load value of the output port B is 30fF

```
create_clock   –name     CLK      –period 2 [get_ports clk]
set_input_delay          –max     0.6       –clock  CLK        [get_ports A]
set_output_delay         –max     0.8       –clock  CLK        [get_ports A]
set_load       [expr 30/1000]                [get_ports B]
```



- When no exact value provided
  - Use load_of lib/cell/pin as load estimated value



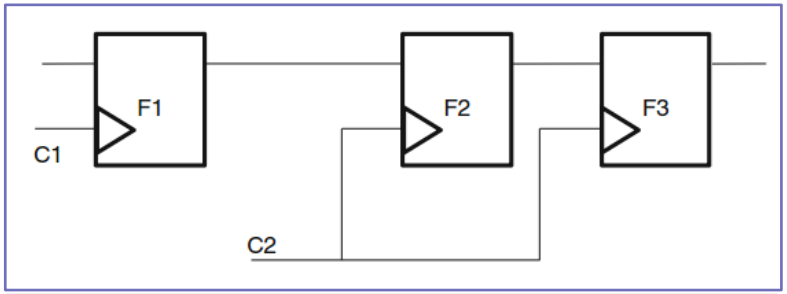

```
set_load [expr [load_of my_lib/INVX1/A]*3] [get_ports B]
```

设计环境Design Environment

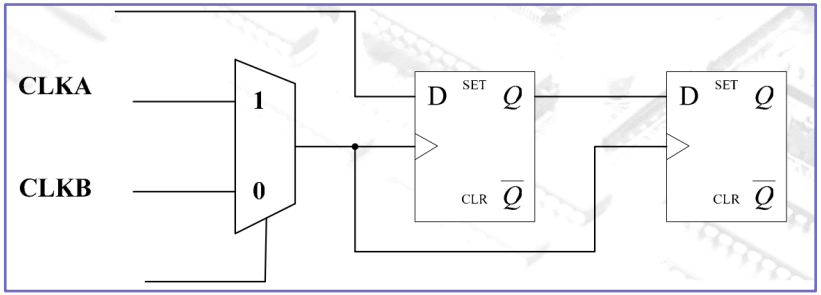
- **PVT corners:** Represented in technology libraries with different cases
  - Process
  - Temperature
  - Voltage
- Always in three cases: Best/fast, Worst/slow, Typical
- Set by target/link library definition and operation condition
  - Written in .synopsys_dc.setup file
  - Since always only one operation condition in one technology library, no need to set operation condition

Timing Exceptions

- **False Path**
  - A false path is the path which is never sensitized/cared due to the logic configuration, expected data sequence, or operating mode.



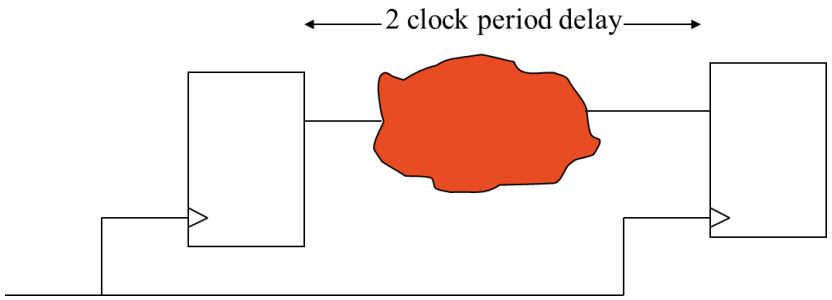

```
set_false_path -from F1/CP -to F2/D
```

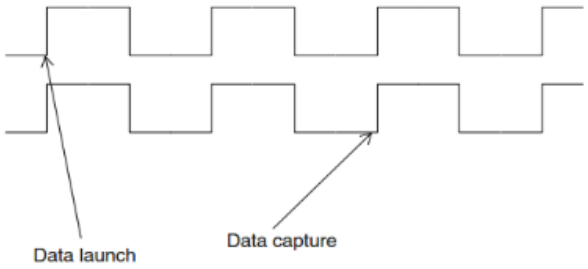


set_false_path –from [get_clocks CLKA] –to [get_clocks CLKB]

Timing Exceptions

- **Multi-cycle Path**
  - A multi-cycle path is A path designed to take more than one clock cycle from launch to capture





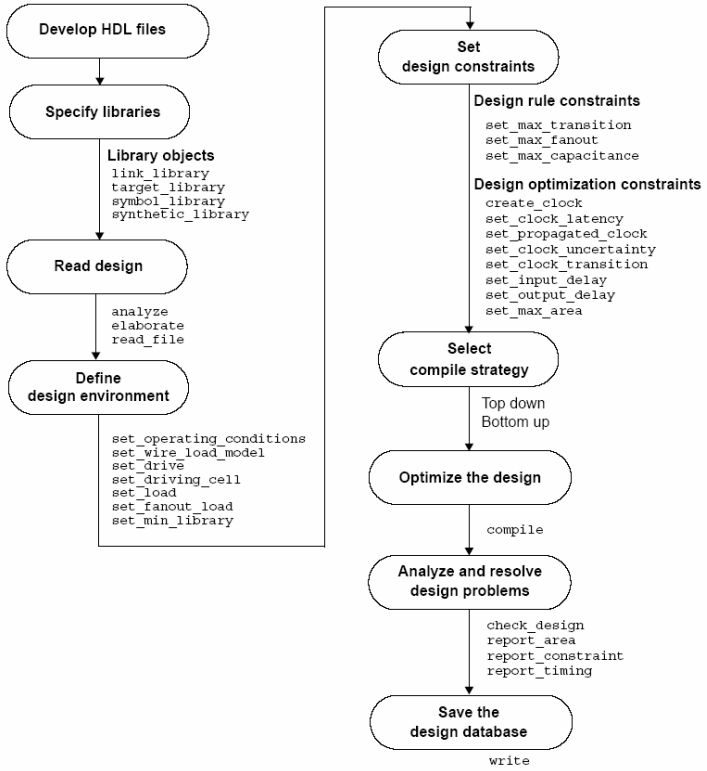

```
set_multicycle_path -setup -from F1/CP -to F2/D   2
set_multicycle_path -hold  -from F1/CP -to F2/D   1
```

第四节 逻辑综合命令

High Map Effort

- Applies maximum optimization effort during gate-level optimization
- Invokes Critical Path Resynthesis as needed

compile –map_effort high



Incremental Mapping

- An incremental compile (-incremental_mapping compile option) allows you to incrementally improve your design by experimenting with different approaches.
- An incremental compile performs only gate-level optimization and does not perform logic-level optimization. The resulting design's performance is the same or better than the original design's.

```
compile -map_effort high -incremental_mapping
```

Others

set_fix_multiple_port_nets -all -buffer_constants

remove_unconnected_ports [get_cells -hierarchical]

set power_driven_clock_gating true

compile -map_effort high -incremental_mapping -gate_clock

Report Timing/Area/Power...

- report_timing
  [-to to_list]
  [-from from_list]
  [-through through_list]
  [-path short | full | full_clock | only | end]
  [-delay min | min_rise | min_fall | max | max_rise | max_fall]
  [-nworst paths_per_endpoint]
  [-max_paths max_path_count]
  [-input_pins]
  [-nets]
- report_area
- report_power

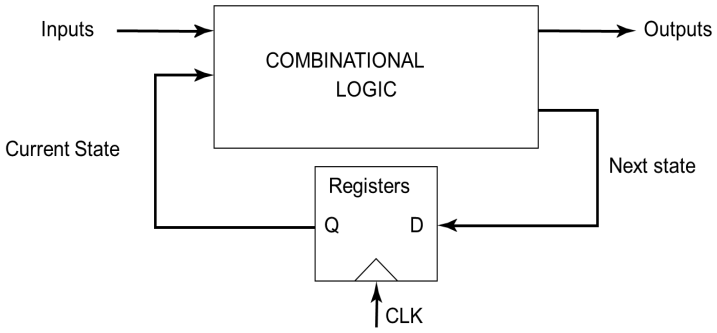
Writing Results

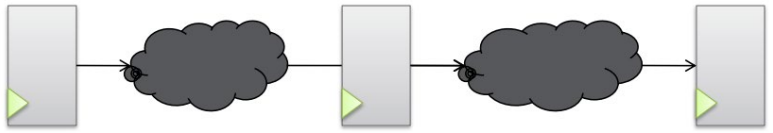
```
write -f verilog -hierarchy -output ${DESIGN_NAME}_mapped.v
write_sdc -nosplit /${DESIGN_NAME}_mapped.sdc
```

第五节 静态时序分析

同步设计Synchronous Design

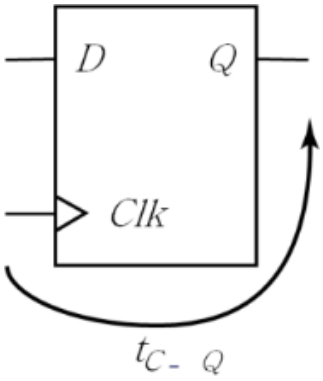
- The majority of digital designs are Synchronous and constructed with Sequential Elements.
  - Synchronous design eliminates races
  - Pipelining increases throughput.
- We will assume that all sequentials are Edge-Triggered, using D-Flip Flops as registers.
- D-Flip Flops have three critical timing parameters:
  - *tcq* – clock to output: propagation delay
  - *tsetup* – setup time: the time the data needs to arrive before the clock
  - *thold* – hold time: the time data has to be stable after clock

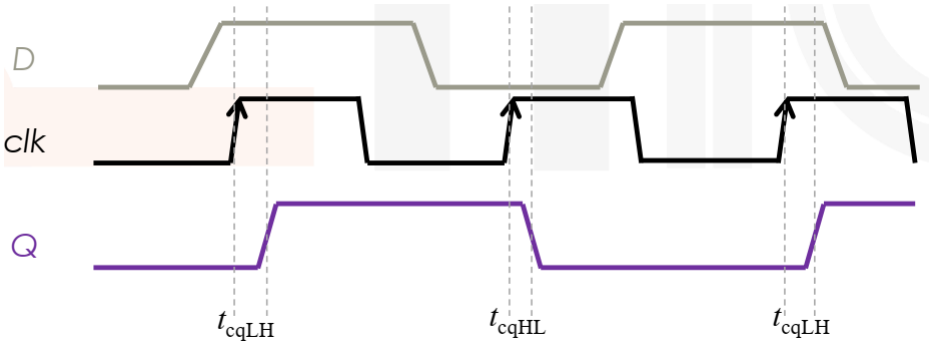




Timing Parameters- *tcq*

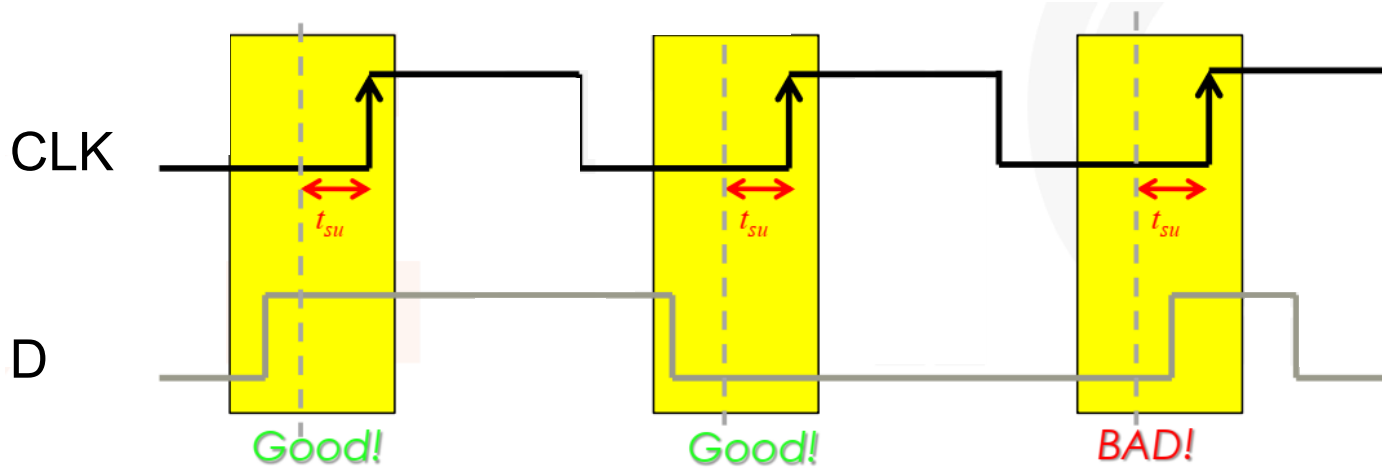
- *tcq* is the time from the clock edge until the data appears at the output.
- The *tcq* for rising and falling outputs is different.





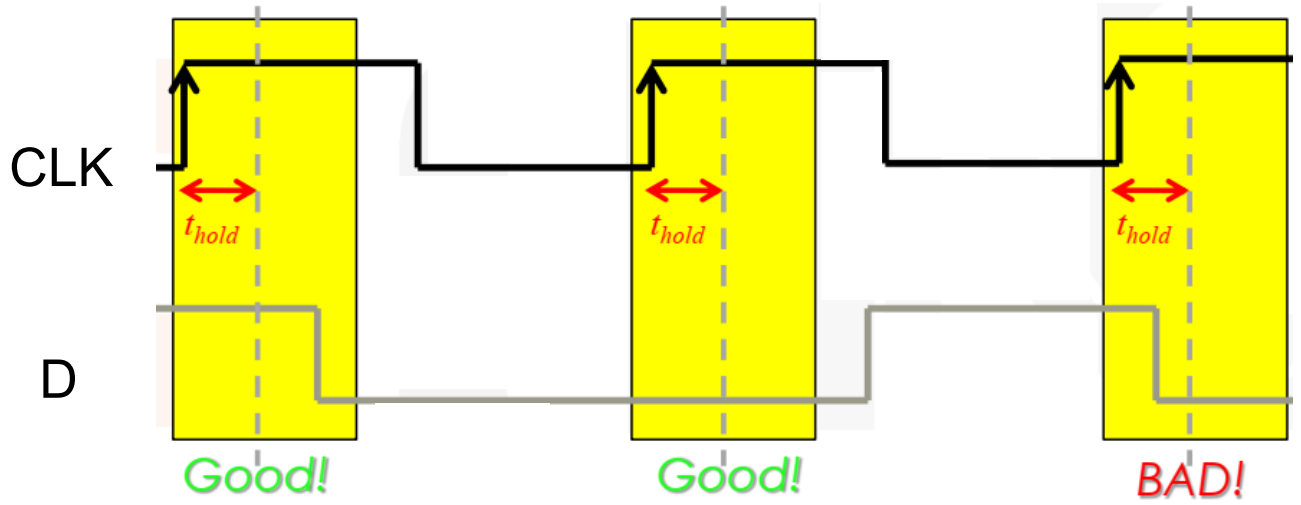
Timing Parameters- *tsetup*

- *tsetup* - Setup time is the time the data has to arrive before the clock to ensure correct sampling



Timing Parameters- *thold*

- thold - Hold time is the time the data has to be stable after the clock to ensure correct sampling.

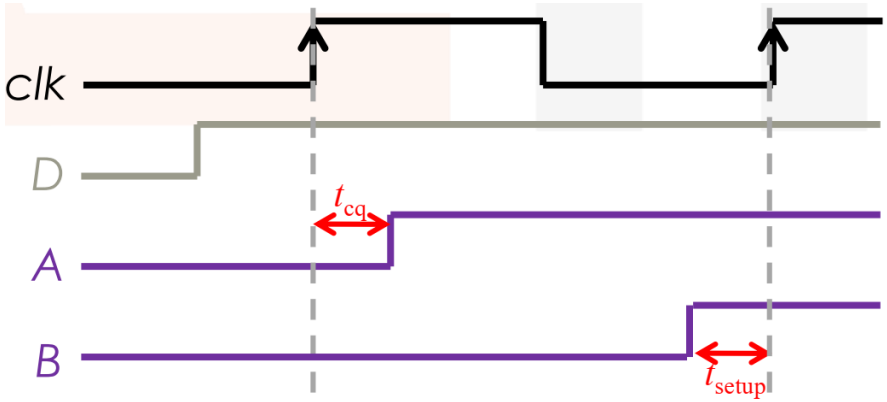


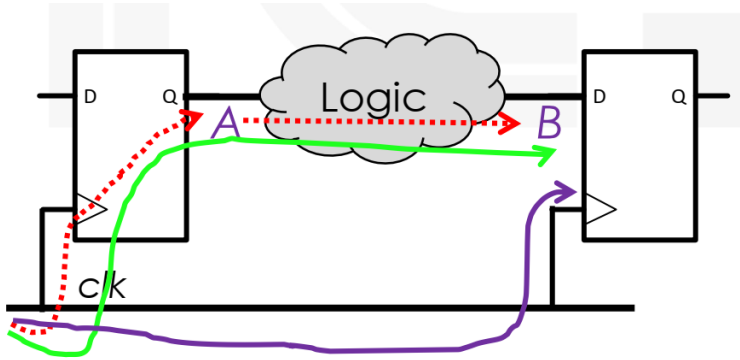
Timing Constraints

- There are two main problems that can arise in synchronous logic:
  - **Max Delay**: The data doesn't have enough time to pass from one register to the next before the next clock edge.
  - **Min Delay**: The data path is so short that it passes through several registers during the same clock cycle.
- Max delay violations are a result of a slow data path, including the registers' tsetup, therefore it is often called the "Setup" path.
- Min delay violations are a result of a short data path, causing the data to change before the thold has passed, therefore it is often called the "Hold" path.

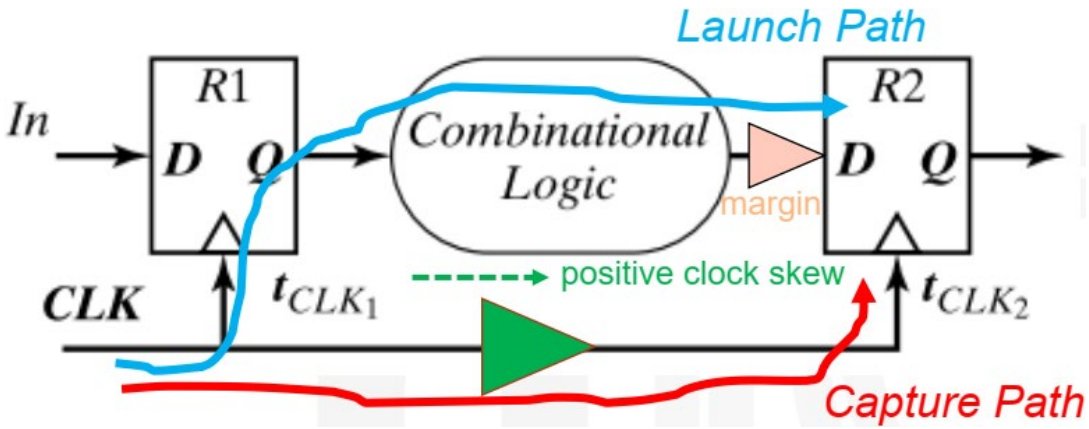
Setup Constraint

- Let's see what makes up our clock cycle:
  - After the clock rises, it takes tcq for the data to propagate to point A.
  - Then the data goes through the delay of the logic to get to point B.
  - The data has to arrive at point B, tsetup before the next clock.
- In general, our timing path is a race:
  - Between the Data Arrival, starting with the launching clock edge.
  - And the Data Capture, one clock period later





Setup Constraint



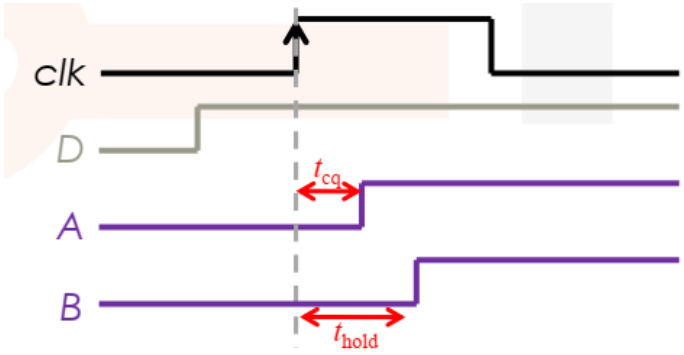
$$T > tcq + tlogic + tsetup$$

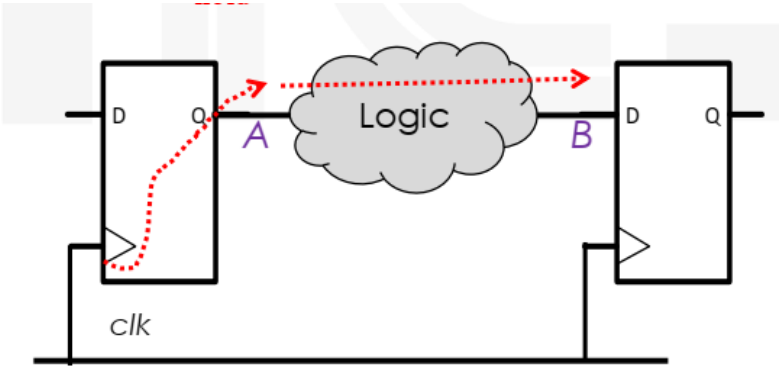
Adding in clock skew and other guardbands:

$$T + tskew > tcq + tlogic + tsetup + tmargin$$

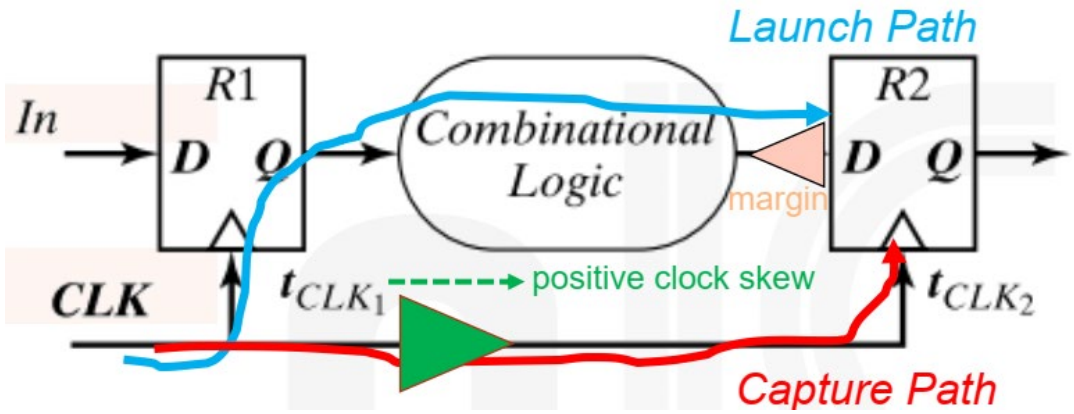
Hold Constraint

- Hold problems occur due to the logic changing before thold has passed.
- This is not a function of cycle time – it is relative to a single clock edge!
- Let's see how this can happen:
  - The clock rises and the data at A changes after tcq.
  - The data at B changes tpd(logic) later.
  - Since the data at B had to stay stable for thold after the clock (for the second register), the change at B has to be at least thold after the clock edge.





Hold Constraint



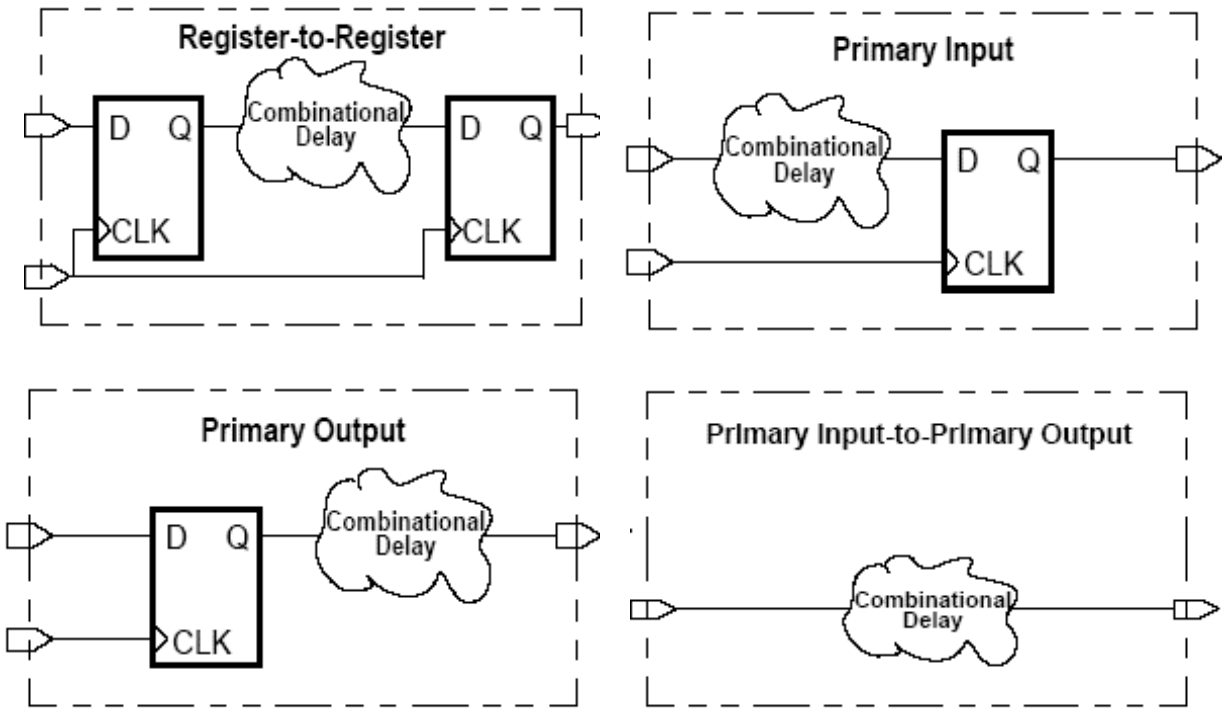
$$tcq + tlogic > thold$$

Adding in clock skew and other guardbands:

$$tcq + tlogic - tmargin > thold + tskew$$

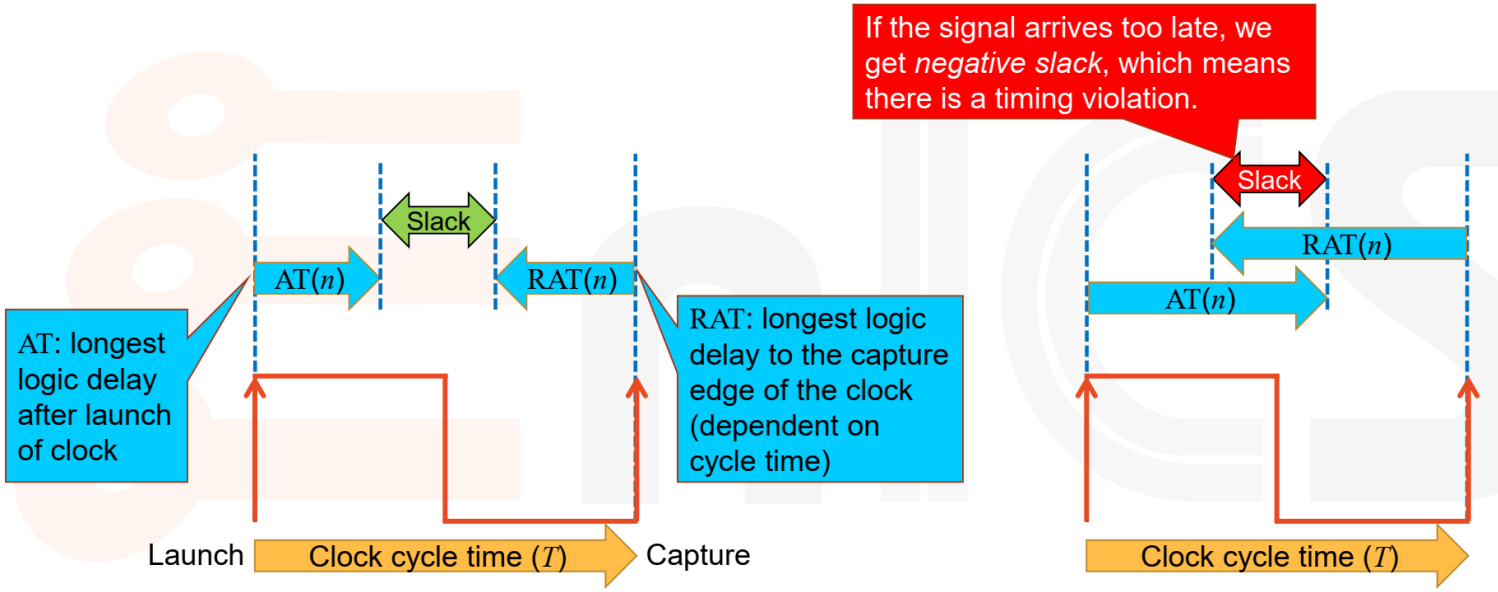
Timing Paths

- Four categories of timing paths
  - Register to Register (reg2reg)
  - Register to Output (reg2out)
  - Input to Register (in2reg)
  - Input to Output (in2out)



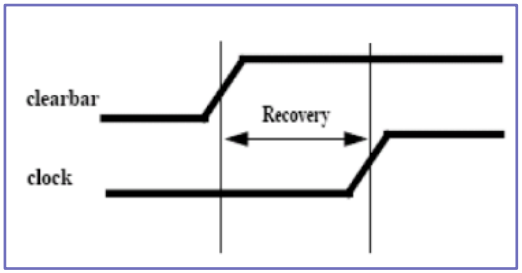
STA

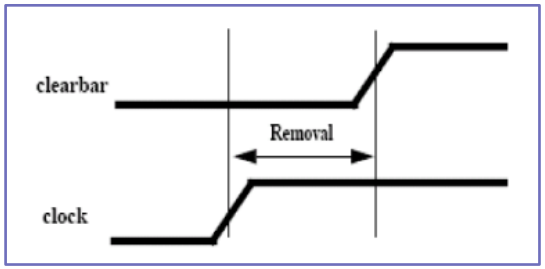
- Arrival Time at a node (AT): the longest path from the source to the node.
- Required Arrival Time at node (RAT): the latest time the signal is allowed to leave the node to make it to the sink in time.
- Slack is defined as: Slack = RAT – AT

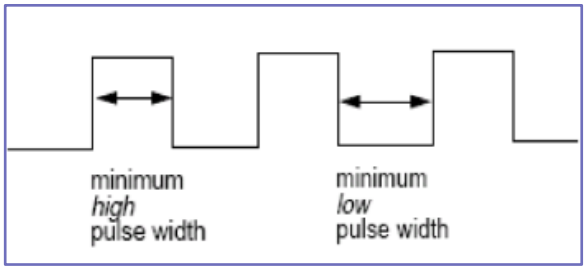


Recovery, Removal and MPW

- Recovery Check
  - The minimum time that an asynchronous control input pin must be stable after being deasserted and before the next clock transition (active-edge)
- Removal Check
  - The minimum time that an asynchronous control input pin must be stable before being deasserted and after the previous clock transition (active edge)
- Minimum Clock Pulse Width (MPW)
  - The amount of time after the rising/falling edge of a clock that the clock signal must remain stable.







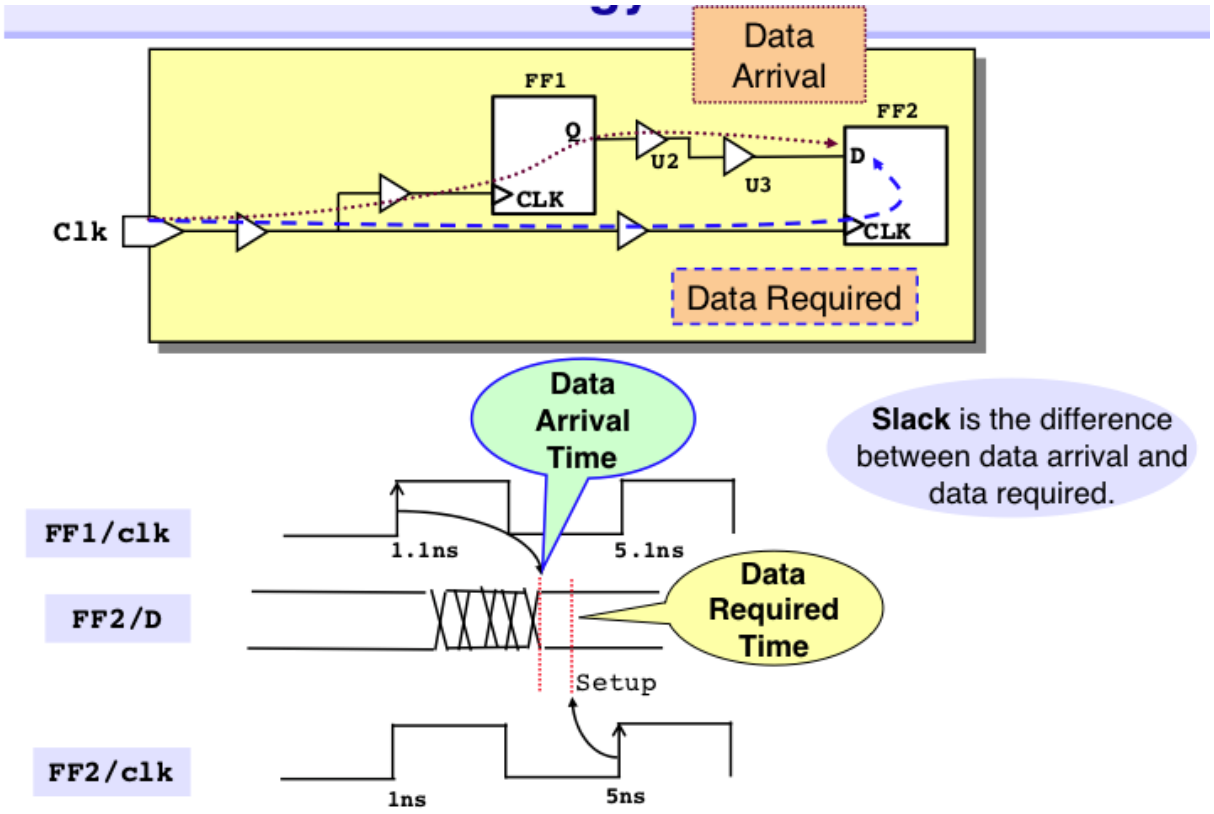
Checking your design

- **report_analysis_coverage**
  - checks that you have fully constrained your design
- **check_timing**
  - performs a variety of consistency and completeness checks on the timing constraints specified for a design.

Type of Check	Total	Met	Violated	Untested
setup	11694	11673 (100%)	0 (0%)	21 (0%)
hold	11680	11047 (95%)	612 (5%)	21 (0%)
recovery	9113	9111 (100%)	0 (0%)	2 (0%)
removal	9113	9111 (100%)	0 (0%)	2 (0%)
min_period	75	75 (100%)	0 (0%)	0 (0%)
min_pulse_width	28348	28343 (100%)	0 (0%)	5 (0%)
clock_gating_setup	591	297 (50%)	0 (0%)	294 (50%)
clock_gating_hold	591	297 (50%)	0 (0%)	294 (50%)
out_setup	9	9 (100%)	0 (0%)	0 (0%)
out_hold	9	9 (100%)	0 (0%)	0 (0%)
All Checks	71223	69972 (98%)	612 (1%)	639 (1%)

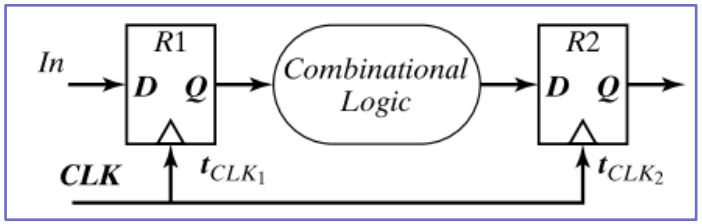
Report Timing

- most often used command: *report_timing*



Report Timing-Setup

header

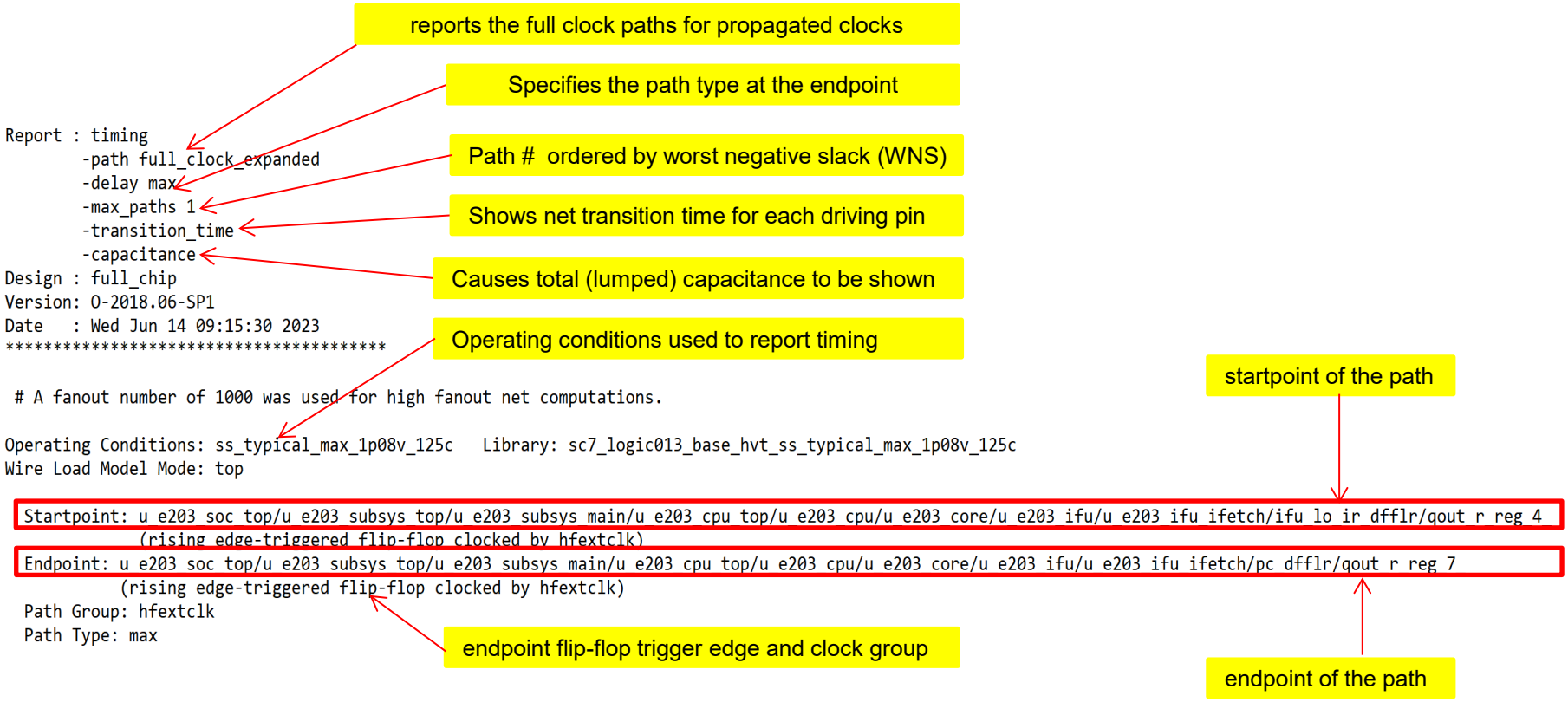



```
Report : timing
        -path full_clock_expanded
        -delay max
        -max_paths 1
        -transition_time
        -capacitance
Design : full_chip
Version: O-2018.06-SP1
Date   : Wed Jun 14 09:15:30 2023
****************************************

 # A fanout number of 1000 was used for high fanout net computations.

Operating Conditions: ss_typical_max_1p08v_125c   Library: sc7_logic013_base_hvt_ss_typical_max_1p08v_125c
Wire Load Model Mode: top

  Startpoint: u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_
              (rising edge-triggered flip-flop clocked by hfextclk)
  Endpoint: u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_
            (rising edge-triggered flip-flop clocked by hfextclk)
  Path Group: hfextclk
  Path Type: max

  Des/Clust/Port     Wire Load Model       Library
  ------------------------------------------------
  full_chip          Small                 sc7_logic013_base_hvt_ss_typical_max_1p08v_125c

  Point                                                 Cap      Trans      Incr       Path
  ---------------------------------------------------------------------------------------
  clock hfextclk (rise edge)                                                 0.00       0.00
  clock network delay (ideal)                                                6.00       6.00
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/CK (DFFRQX1MTH)
                                                                    1.00       0.00       6.00 r
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/Q (DFFRQX1MTH) <-
                                                    0.06       1.15       1.51       7.51 r
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout[4] (sirv_gnrl_dfflr_DW16_0)
                                                                               0.00       7.51 r
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_o_ir[4] (e203_ifu_ifetch)
                                                                               0.00       7.51 r
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/ifu_o_ir[4] (e203_ifu)
                                                                               0.00       7.51 r
                                                                                .
                                                                                .
                                                                                .
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_/D (DFFRQX2MTH)
                                                                    0.24       0.00     102.45 r
  data arrival time                                                                     102.45

  clock hfextclk (rise edge)                                                  100.00     100.00
  clock network delay (ideal)                                                   6.00     106.00
  clock uncertainty                                                            -0.50     105.50
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_/CK (DFFRQX2MTH)
                                                                               0.00     105.50 r
  library setup time                                                           -0.35     105.15
  data required time                                                                     105.15
  ---------------------------------------------------------------------------------------
  data required time                                                                     105.15
  data arrival time                                                                     -102.45
  ---------------------------------------------------------------------------------------
  slack (MET)                                                                             2.70
```

Launch Path

Capture Path

Report Timing-Header



Report Timing-Launch Path


```
Des/Clust/Port     Wire Load Model       Library
------------------------------------------------
full_chip          Small                 sc7_logic013_base_hvt_ss_typical_max_1p08v_125c

Point                                                  Cap     Trans     Incr      Path
-------------------------------------------------------------------------------------
clock hfextclk (rise edge)                                                0.00      0.00
clock network delay (ideal)                                               6.00      6.00
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/CK (DFFRQX1MTH)
                                                                  1.00      0.00      6.00 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/Q (DFFRQX1MTH) <-
                                                0.06      1.15      1.51      7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout[4] (sirv_gnrl_dfflr_DW16_0)
                                                                            0.00      7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_o_ir[4] (e203_ifu_ifetch)
                                                                            0.00      7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/ifu_o_ir[4] (e203_ifu)
                                                                            0.00      7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_exu/i_ir[4] (e203_exu)
                                                                            0.00      7.51 r
                                                                               .
                                                                               .
                                                                               .
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/lden (sirv_gnrl_dfflr_DW32_15)
                                                                            0.00    100.29 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U16/Y (CLKBUFX2MTH)
                                                0.02      0.26      0.41    100.70 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U13/Y (CLKBUFX1MTH)
                                                0.02      0.60      0.50    101.20 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U8/Y (CLKBUFX1MTH)
                                                0.04      1.03      0.80    102.00 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U31/Y (AND2X1MTH)
                                                0.01      0.24      0.45    102.45 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_/D (DFFRQX2MTH)
                                                0.24      0.00    102.45 r
data arrival time                                                           102.45
```

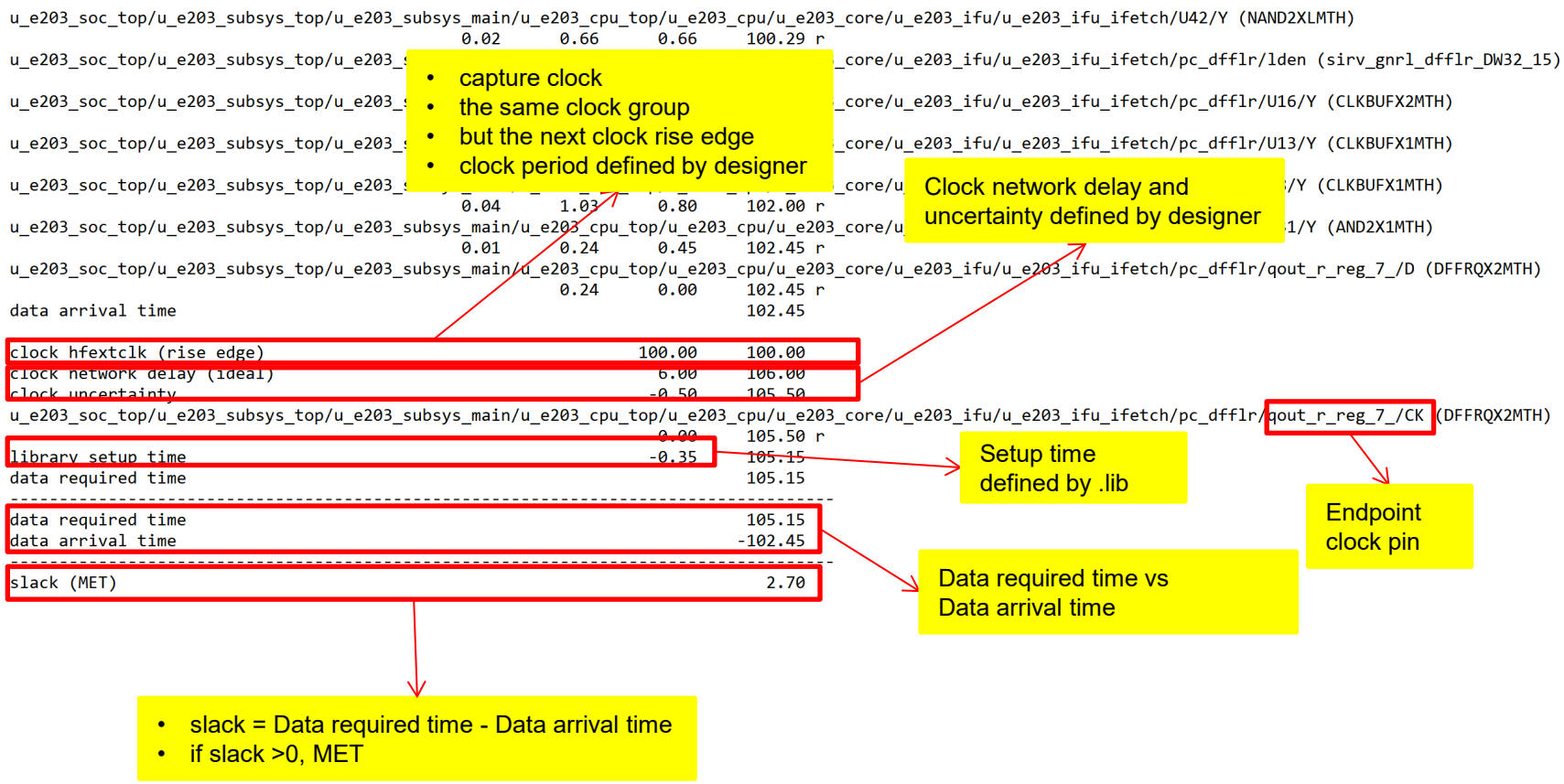
Report Timing-Launch Clock

```
Des/Clust/Port     Wire Load Model       Library
------------------------------------------------
full_chip          Small                 sc7_logic013_base_hvt_ss_typical_max_1p08v_125c

Point                                          Cap     Trans      Incr       Path
-------------------------------------------------------------------------------
clock hfextclk (rise edge)                                        0.00       0.00
clock network delay (ideal)                                       6.00       6.00
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/CK (DFFRQX1MTH)
                                                                  1.00       0.00       6.00 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/Q (DFFRQX1MTH) <-
                                               0.06       1.15       1.51       7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout[4] (sirv_gnrl_dfflr_DW16_0)
                                                                  0.00       7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_o_ir[4] (e203_ifu_ifetch)
                                                                  0.00       7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/ifu_o_ir[4] (e203_ifu)
                                                                  0.00       7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_
                                                                  0.00       7.51 r
                                                                                 .
                                                                                 .
                                                                                 .
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/lden (sirv_gnrl_dfflr_DW32_15)
                                                                  0.00     100.29 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U16/Y (CLKBUFX2MTH)
                                               0.02       0.26       0.41     100.70 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U13/Y (CLKBUFX1MTH)
                                               0.02       0.60       0.50     101.20 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U8/Y (CLKBUFX1MTH)
                                               0.04       1.03       0.80     102.00 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U31/Y (AND2X1MTH)
                                               0.01       0.24       0.45     102.45 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_/D (DFFRQX2MTH)
                                               0.24       0.00     102.45 r
data arrival time                                                           102.45
```

- Launch clock
  - clock group: hfextclk
  - clock network delay: defined by designer
  - launch the startpoint flip-flop's CK pin

Endpoint data pin

Report Timing-Capture Clock



Report Timing-Hold

```
Report : timing
       -path_full_clock_expanded
       -delay_min
       -max_paths 1
       -transition_time
       -capacitance
Design : full_chip
Version: O-2018.06-SP1
Date   : Wed Jun 14 13:48:44 2023
****************************************

 # A fanout number of 1000 was used for high fanout net computations.

Operating Conditions: ss_typical_max_1p08v_125c   Library: sc7_logic013_base_hvt_ss_typical_max_1p08v_125c
Wire Load Model Mode: top

  Startpoint: u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_
              (rising edge-triggered flip-flop clocked by hfextclk)
  Endpoint: u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_
            (rising edge-triggered flip-flop clocked by hfextclk)
  Path Group: hfextclk
  Path Type: min

  Des/Clust/Port     Wire Load Model       Library
  ------------------------------------------------
  full_chip          Small                 sc7_logic013_base_hvt_ss_typical_max_1p08v_125c

  Point                                             Cap     Trans      Incr       Path
  -----------------------------------------------------------------------------------
  clock hfextclk (rise edge)                                           0.00       0.00
  clock network delay (ideal)                                          6.00       6.00
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/CK (DFFRQX1MTH)
                                                    1.00      0.00       6.00 r
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/Q (DFFRQX1MTH) <-
                                                    0.06      1.15       1.51       7.51 r
                                                                                    .
                                                                                    .
                                                                                    .
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/dnxt[7] (sirv_gnrl_dfflr_DW32_15)
                                                                         0.00      14.07 r
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U31/Y (AND2X1MTH)
                                                    0.01      0.23       0.40      14.47 r
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_/D (DFFRQX2MTH)
                                                    0.23      0.00       0.00      14.47 r
  data arrival time                                                                 14.47

  clock hfextclk (rise edge)                                           0.00       0.00
  clock network delay (ideal)                                          6.00       6.00
  clock uncertainty                                                    0.50       6.50
  u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_/CK (DFFRQX2MTH)
                                                                         0.00       6.50 r
  library hold time                                                     -0.20       6.30
  data required time                                                                6.30
  -----------------------------------------------------------------------------------
  data required time                                                                6.30
  data arrival time                                                               -14.47
  -----------------------------------------------------------------------------------
  slack (MET)                                                                       8.17
```

header

Launch Path

Capture Path

Report Timing-Launch/Capture clk

- Launch clock
  - clock group: hfextclk
  - clock network delay: defined by designer
  - launch the startpoint flip-flop's CK pin

```
Point                                                    Cap     Trans      Incr       Path
-----------------------------------------------------------------------------------------------------------
clock hfextclk (rise edge)                                                  0.00       0.00
clock network delay (ideal)                                                 6.00       6.00
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/CK (DFFRQX1MTH)
                                                                            1.00       0.00       6.00 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout_r_reg_4_/Q (DFFRQX1MTH) <-
                                                         0.06       1.15       1.51       7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_lo_ir_dfflr/qout[4] (sirv_gnrl_dfflr_DW16_0)
                                                                            0.00       7.51 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/ifu_o_ir[4] (e203_ifu_ifetch)
                                                                            0.00       7.51 r
.
.
.
u_e203_soc_top/u_e203_subsys                                                                  u_e203_core/u_e203_ifu/pipe_flush_pc[7] (e203_ifu)
                                                                                              .50 r
u_e203_soc_top/u_e203_subsys                                                                  u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pipe_flush_pc[7] (e203_ifu_ifetch)
                                                                                              .50 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/U199/Y (OAI2BB1XLMTH)
                                                         0.01       0.60       0.57      14.07 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/dnxt[7] (sirv_gnrl_dfflr_DW32_15)
                                                                            0.00      14.07 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/U31/Y (AND2X1MTH)
                                                         0.01       0.23       0.40      14.47 r
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_/D (DFFRQX2MTH)
                                                                            0.23       0.00      14.47 r
data arrival time                                                                               14.47

clock hfextclk (rise edge)                                                  0.00       0.00
clock network delay (ideal)                                                 6.00       6.00
clock uncertainty                                                           0.50       6.50
u_e203_soc_top/u_e203_subsys_top/u_e203_subsys_main/u_e203_cpu_top/u_e203_cpu/u_e203_core/u_e203_ifu/u_e203_ifu_ifetch/pc_dfflr/qout_r_reg_7_/CK (DFFRQX2MTH)
                                                                            0.00       6.50 r
library hold time                                                          -0.20       6.30
data required time                                                                      6.30

data required time                                                                      6.30
data arrival time                                                                     -14.47
-----------------------------------------------------------------------------------------------------------
slack (MET)                                                                            8.17
```

- capture clock
- the same clock group
- but the **SAME** clock rise edge
- **independent of clock period!**

Reach the endpoint from fastest way

uncertainty is positive, unlike max path

hold time defined by .lib

Data required time vs Data arrival time

- slack = Data required time - Data arrival time
- if slack >0, MET

Report Timing-option

- To debug timing, we would like more information, just use options
  - -capacitance: display net capacitance for each net or driving pin
  - -transition_time: display net transition time for each driving pin
  - -nets: include nets in combinational path
  - -path_type: full, end, only, short, start, full_clock, full_clock_expanded
  - -variation: show the mean and variation value of cell and net delays
- By default, report_timing shows you the most critical path, use options to analyze a specific path or set of paths
  - -from/-to: select a startpoint/endpoint (pins, ports, nets or clocks)
  - -through: select a <through_list> in path (pins, ports, nets or clocks)
  - -delay_type: analyze <delay_type> path (max/min, maxmin_rise/fall)
  - -nworst: number of paths per endpoint, default is 1
  - -greater_path: display paths with length greater than this value

Report Timing-option

```
report_timing
        [-to <to_list>]                 (to pins, ports, nets or clocks)
        [-rise_to <rise_to_list>]       (to pins, ports, nets or clocks)
        [-fall_to <fall_to_list>]       (to pins, ports, nets or clocks)
        [-from <from_list>]             (from pins, ports, nets or clocks)
        [-rise_from <rise_from_list>]   (from pins, ports, nets or clocks with rise sense)
        [-fall_from <fall_from_list>]   (from pins, ports, nets or clocks with fall sense)
        [-exclude <exclude_list>]       (exclude pins, ports, nets or cells)
        [-rise_exclude <rise_exclude_list>]     (exclude pins, ports, nets or cells with rise sense)
        [-fall_exclude <fall_exclude_list>]     (exclude pins, ports, nets or cells with fall sense)
        [-through <through_list>]       (through pins, ports, nets or clocks)
        [-rise_through <rise_through_list>]     (to pins, ports, nets or clocks)
        [-fall_through <fall_through_list>]     (to pins, ports, nets or clocks)
        [-path_type <path_type>](path type: Values: full, end, only, short, start, full_clock, full_clock_expanded)
        [-delay_type <delay_type>](delay type: Values: max, max_rise, max_fall, min, min_rise, min_fall)
        [-nworst <paths_per_endpoint>]  (number of paths per endpoint, default is 1)
        [-max_paths <max_path_count>]   (maximum number of paths per path group)
        [-input_pins]           (include input pins in combinational path)
        [-nets]                 (include nets in combinational path)
        [-transition_time]      (display net transition time for each driving pin)
        [-crosstalk_delta]      (display delta delay for each input pin)
        [-capacitance]          (display net capacitance for each ne or driving pin)
        [-effective_capacitance](display net capacitance for each net or driving pin that is calculated with Ceff)
        [-variation]            (show the mean and variation value of cell and net delays)
        [-lesser_path <max_path_delay>]         (display paths with length less than this value: Value >= 0)
        [-greater_path <min_path_delay>]        (display paths with length greater than this value: Value >= 0)
        [-slack_lesser_than <max_slack>]        (display paths with slack less than this value)
        [-slack_greater_than <min_slack>]       (display paths with slack greater than this value)
        [-loops]                                (report timing loops)
        [-enable_preset_clear_arcs]             (enable async arcs for this report)

          .....
```