Design of Large Scale Digital Circuits
What we have done so far in class ...

- Device Physics
- MOSFET
- Inverter
- Combinatorial Logic
- Sequential Logic
Now.. Digital design

Front End Digital Design Flow

Specifications

HDL Coding

Verilog
VHDL

Mentor Modelsim
Xilinx Isim

Functional Verification

Logic Synthesis

Synopsys Design Compiler
Cadence RTL compiler

Static Timing Analysis

Synopsys Primetime

Back End Digital Design Flow

Floorplanning

Clock Tree Synthesis

Place & Route

RC Extraction

Layout verification

Cadence Encounter
Synopsys IC compiler

Synopsys Star-RXCT
DRC/LVS Tools
Synopsys Hercules
Co-simulation tools: Cadence Incisive

Cadence Incisive
Let’s design a counter using this method

- VHDL Behavioral modeling

```vhdl
library IEEE;
use IEEE.std_logic_1164.all;
use ieee.numeric_std.all;

entity counter is
port (clk: in std_logic;
     reset: in std_logic;
     output: out std_logic_vector(3 downto 0));
end counter;

architecture behavioural of counter is

signal out_i: unsigned(3 downto 0);

begin
  output <= std_logic_vector(out_i);

  process(clk, reset)
  begin
    if reset = '1' then
      out_i <= (others=>'0');
    elsif rising_edge(clk) then
      out_i <= out_i + 1;
    end if;
  end process;

end behavioural;
```
library IEEE;
use IEEE.std_logic_1164.all;
use ieee.numeric_std.all;

entity counter is
port (clk: in std_logic;
    reset: in std_logic;
    output: out std_logic_vector(3 downto 0)
    );
end counter;

architecture behavioural of counter is
begin
    signal out_i: unsigned(3 downto 0);
    begin
    output <= std_logic_vector(out_i);
    process(clk, reset)
    begin
    if reset = '1' then
        out_i <= (others=>'0');
    elsif rising_edge(clk) then
        out_i <= out_i + 1;
    end if;
    end process;
end behavioural;

module counter ( clk, reset, \output 
    output [3:0] \output 
    input clk, reset;
    wire n14, n15, n16, n17, N2, N3, N4, n4, n6, n7, n8, n9;
    DFFSR \out_i_reg[0] ( .D(n6), .CLK(clk), .R(n4), .S(1'b1), .Q(n17) );
    DFFSR \out_i_reg[1] ( .D(n2), .CLK(clk), .R(n4), .S(1'b1), .Q(n16) );
    DFFSR \out_i_reg[2] ( .D(n3), .CLK(clk), .R(n4), .S(1'b1), .Q(n15) );
    DFFSR \out_i_reg[3] ( .D(n4), .CLK(clk), .R(n4), .S(1'b1), .Q(n14) );
    INVX1 U6 ( .A(reset), .Y(n4) );
    INVX2 U7 ( .A\output [3], .B(n9), .Y(n4) );
    INVX2 U12 ( .A\output [2], .B(n8), .Y(n3) );
    INVX2 U13 ( .A\output [1], .B(n6), .Y(n2) );
    INVX2 U14 ( .A\output [2], .B(n8), .Y(n7) );
    INVX2 U15 ( .A(n7), .Y(n9) );
    INVX2 U16 ( .A\output [1], .B\output [0], .Y(n8) );
    BUFX2 U17 ( .A(n14), .Y\output [3] );
    BUFX2 U18 ( .A(n17), .Y\output [0] );
    BUFX2 U19 ( .A(n16), .Y\output [1] );
    BUFX2 U20 ( .A(n15), .Y\output [2] );
    INVX1 U21 ( .A\output [0], .Y(n6) );
endmodule
Synthesis Flow

Develop HDL files

Specify library
- Design object
  - Target_library
  - Link_library
  - Symbol_library
  - Synthetic_library

Read design and link
- Read_file
- analyze
- elaborate
- link

Define design environment
- set_operating_conditions
- set_wire_load_model
- set_wire_load_mode
- set_drive
- set_driving_cell
- set_load
- set_fanout_load
- set_min_library

Set design constraint
- Design Rule Constraints
  - set_max_transition
  - set_max_capacitance
  - set_max_fanout
- Design Opt. Constraints
  - create_clock
  - set_don’t_touch_network
  - set_clock_uncertainty
  - set_input_delay
  - set_output_delay
  - set_max_area

Select design 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
Specify Libraries

- **Synthetic Library (.sldb)**: Technology Independent, microarchitecture design libraries.

- **Target Library (.db)**: Technology library that Design Compiler uses to build the circuit.

Other library configuration: Link Library, symbol library: they are usually the same as the target library.
Standard Cell Technology Library

AND2_A
AND2_B
AND2_C
...
INV_A
INV_B
...
INV_F
Read Design and Link
(analyze and elaborate)

Analyze:
Reads the HDL source and checks for syntactical errors

Elaborate:
Translates the design into a technology-independent design (GTECH) from the intermediate files produced during analysis
Read Design and Link (analyze and elaborate)

Synopsys Design Compiler Command Window

```bash
design_vision> elaborate Stop_level -architecture BEHAVIORAL -library WORK
Loading db file '/synopsys/DesignCompiler/libraries/syn/gtech.db'
Loading db file '/synopsys/DesignCompiler/libraries/syn/standard.sldb'
Loading link library 'gsc145nm'
Loading link library 'gtech'
Running PRESTO HDLC

Inferred memory devices in process
in routine counter line 23 in file
'./src/counter.vhd'.

Register Name | Type  | Width | Bus | MB | AR | AS | SR | SS | ST |
--------------|-------|-------|-----|----|----|----|----|----|----|
out_i_reg     | Flip-flop | 4 | Y | N | Y | N | N | N | N |

===============================================================================

Presto compilation completed successfully.
Elaborated 1 design.
Current design is 'counter'.
```

Design Compiler GUI

Generic Register  Generic Adder
Define Design Environment

- **Set_operating_condition**: specify variation in Process, Voltage, and Temperature (PVT).

- **set_wire_load_model**: use to determine effects of interconnect on timing
- **set_load**: designate the load of an output port
- **set_driving_cell**: designate the driving cell to an input port
Define Design Constraints

• Most important step of synthesis.
• User define the constrains, and the synthesis tool optimizes the circuits given these constrains.
• There are two types of constraints
  – **Design Rule Constraints**: typically defined by the vendor for each standard cell.
    • Max Transition Time, Max fanout, Max (and Min) driving capacitance.
  – **Optimization Constraints**: timing or chip area constraints set by the user.
    • Clock constrains, Input/output delay, input/output transition and etc...
Clock Constrains

• Clock Jitter

• Clock Latency, Clock transition, Clock uncertainty ...
Setup/Hold Time

CLK

D

Q

DATA STABLE

Q1  Q2  Q3  Q4

Data In  Clock

Setup/Hold Violation?
How to determine a S/H time violation?
- Delay path

Path 1: input port to data pin of sequential cell
Path 2: input port to output port
Path 3: clock pin to data pin of next sequential cell
Path 4: clock pin to output port
Let’s calculate Setup/Hold Time Slack

- CLK Period: 30 ns
- Sequential block has output delay of 1ns
- Each logic block has different delays in nS
- No delay on the clock path
- Each Sequential block requires 1ns setup time and 0.5ns hold time

Red: \( T_d = T_{seq} + T_d4 + T_d5 + T_d6 = 9\text{ns} \)
Yellow: \( T_d = T_{seq} + T_d4 + T_d5 + T_d6 + T_d8 = 11\text{ns} \)
Purple: \( T_d = T_{seq} + T_d1 + T_d2 + T_d3 = 8\text{ns} \)
Green: \( T_d = T_{seq} + T_d7 + T_d2 + T_d3 = 8\text{ns} \)

\( T_{longest} = 11\text{ns}, \ T_{shortest} = 8\text{ns} \)

Setup time slack: \( T_{clock} – T_{longest} - T_{setup} = 18\text{ns} \)
Setup time slack: \( T_{shortest} – T_{hold} = 8 – 0.5 = 7.5\text{ns} \)
Constraining the Clock

create_clock -name "clk" -period 4 -waveform {0 2} {clk}
set_clock_uncertainty 0.1 clk
set_clock_latency 0.2 clk
set_clock_transition 0.1 clk
set_dont_touch_network clk

Be very careful if:
• You have clock dividers in the circuits
• You are using clock gating logic
Optimization

• Compile the design

Loading db file '/cadence/kits/FreePDK45/FreePDK45/osu_soc/lib/files/gscl45nm.db'

Optimization Complete

Information: Defining new variable 'compile_group_pull_control_logic'. (CMD-041)
Reports

Quality of results

Timing

Constraints

---

Quality of results

Report: qor
Design: counter
Version: E-2016.12-SP5-2
Date: Mon Nov 10 14:44:29 2014

Levels of Logic: 5.00
Critical Path Length: 0.32
Critical Path Slack: 3.46
Critical Path Clk Period: 4.00
Total Negative Slack: 0.00
No. of Violating Paths: 0.00
No. of Hold Violations: 0.00

Timing Path Group 'clk'

Levels of Logic: 5.00
Critical Path Length: 0.32
Critical Path Slack: 3.46
Critical Path Clk Period: 4.00
Total Negative Slack: 0.00
No. of Violating Paths: 0.00
No. of Hold Violations: 0.00

Operating Conditions: typical
Library: gscl45nm
Wire Load Modal Mode: top
Startpoint: out_i_reg[0]
   (rising edge-triggered flip-flop clocked by clk)
Endpoint: out_i_reg[3]
   (rising edge-triggered flip-flop clocked by clk)
Path Group: clk
Path Type: max

Point | Increment | Path
---- | --------- | ----
clk    | 0.00      | 0.06
clock network delay (Ideal) | 0.20 | 0.26
out_i_reg[0]/CLK (DFDSR) | 0.60 | 0.26
out_i_reg[0]/O (DFDSR) | 0.13 | 0.33
U21/Y (BUF2X2) | 0.04 | 0.36
U19/Y (AND2X1I) | 0.06 | 0.42
U17/Y (AND2X1I) | 0.04 | 0.46
U16/Y (INVX1) | 0.03 | 0.49
U13/Y (XNOR2X1I) | 0.03 | 0.52
out_i_reg[3]/O (DFDSR) | 0.00 | 0.52
data arrival time | 0.52
clk    | 4.00      | 4.06
clock network delay (Ideal) | 0.20 | 4.28
clock uncertainty | -0.10 | 4.16
out_i_reg[3]/CLK (DFDSR) | 0.00 | 4.16
library setup time | -0.13 | 3.97
data required time | 3.97

Constraints

Report: constraint
Design: counter
Version: E-2016.12-SP5-2
Date: Mon Nov 10 15:06:36 2014

<table>
<thead>
<tr>
<th>Group</th>
<th>Critical Range</th>
<th>Slack</th>
<th>Endpoints</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>clk</td>
<td>max_delay/setup</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>default</td>
<td>max_delay/setup</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Group</th>
<th>Critical Range</th>
<th>Slack</th>
<th>Endpoints</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>clk</td>
<td>min_delay/hold</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Constraint</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>max_capacitance</td>
<td>0.01 (VIOLATED)</td>
</tr>
<tr>
<td>max_delay/setup</td>
<td>0.00 (MET)</td>
</tr>
<tr>
<td>critical_range</td>
<td>0.00 (MET)</td>
</tr>
</tbody>
</table>

Data required time: 3.97
Data arrival time: 3.97
Slack: 3.46
What to fix and what can be ignored?

Must fix:
• Setup time violations

Can postpone to fix during Place & Route
• Hold time violations
• Secondary design rule constraints
  – max capacitance, Fanout, max transitions
Save the netlist and delay constraint

Verilog netlist (.v)

```verilog
module counter ( clk, reset, \output );
  output [3:0] \output ;
  input clk, reset;
  wire n14, n15, n16, n17, N2, N3, N4, n4, n6, n7, n8, n9;
  DFFSR \out_i_reg[0] ( .D(n6), .CLK(clk), .R(n4), .S(1'b1), .Q(n17) );
  DFFSR \out_i_reg[1] ( .D(n2), .CLK(clk), .R(n4), .S(1'b1), .Q(n16) );
  DFFSR \out_i_reg[2] ( .D(n3), .CLK(clk), .R(n4), .S(1'b1), .Q(n15) );
  DFFSR \out_i_reg[3] ( .D(n4), .CLK(clk), .R(n4), .S(1'b1), .Q(n14) );
  INVX1 U6 ( .A(reset), .Y(n4) );
  XNOR2X1 U10 ( .A(\output [3]), .B(n9), .Y(N4) );
  XOR2X1 U12 ( .A(\output [2]), .B(n8), .Y(N3) );
  XNOR2X1 U13 ( .A(\output [1]), .B(n6), .Y(N2) );
  AND2X1 U14 ( .A(\output [2]), .B(n8), .Y(n7) );
  INVX1 U15 ( .A(n7), .Y(n9) );
  AND2X1 U16 ( .A(\output [1]), .B(\output [0]), .Y(n8) );
  BUFX2 U17 ( .A(n14), .Y(\output [3]) );
  BUFX2 U18 ( .A(n17), .Y(\output [6]) );
  BUFX2 U19 ( .A(n16), .Y(\output [1]) );
  BUFX2 U20 ( .A(n15), .Y(\output [2]) );
  INVX1 U21 ( .A(\output [0]), .Y(n6) );
endmodule
```

Standard Delay Constrains (.sdc)

```
# Created by write_sdc on Mon Nov 10 15:24:18 2014

set sdc_version 1.9

set_units -time ns -resistance kOhm -capacitance pF -voltage V -current uA
create_clock [get_ports clk] -period 4 -waveform [0 2]
set_clock_latency 0.2 [get_clocks clk]
set_clock_uncertainty 0.1 [get_clocks clk]
set_clock_transition -max -rise 0.1 [get_clocks clk]
set_clock_transition -max -fall 0.1 [get_clocks clk]
set_clock_transition -min -rise 0.1 [get_clocks clk]
set_clock_transition -min -fall 0.1 [get_clocks clk]
```
Place and Route
Cadence Encounter

Command Lines

GUI
Design Import

- Verilog structural netlist (.v)
- Standard Cell Library (.lef)
- Standard Cell Library timing library (.lib)
After Design Import

Summary of the imported design:

- Total number of combinational cells: 25
- Total number of sequential cells: 4
- Total number of tristate cells: 2
- Total number of power gating cells: 8
- Total number of isolation cells: 8
- Total number of power switch cells: 8
- Total number of pulse generator cells: 0
- Total number of always on buffers: 0
- Total number of retention cells: 0
- List of usable buffers: BUFX2 BUFX4
- Total number of usable buffers: 2
- List of unusable buffers: 
- List of usable inverters: INVX1 INVX4 INVX8
- Total number of usable inverters: 4
- List of unusable inverters: 
- Total number of unusable inverters: 0
- List of identified usable delay cells: CLKBUF1 CLKBUF2 CLKBUF3
- Total number of identified usable delay cells: 3
- List of identified unusable delay cells: 
- Total number of identified unusable delay cells: 0

Grid for Std Cell Placement

Input/output pins
Floorplanning

Extra space for power rings

Standard cell placement area
Make Power network

6 core components: 6 unplaced, 6 placed, 6 fixed
Read in 6 logical pins
Read in 5 nets
Read in 2 special nets, 2 routed
Read in 12 terminals
2 nets selected.

Begin power routing...

**WARN: (ENC5-1254): Net vdd does not have block pins to be routed. Please check pin list or port class.
**WARN: (ENC5-1255): Net vdd does not have pad pins to create pad ring. Please check pin list or port class.
**WARN: (ENC5-1256): Net vdd does not have CORE class pad pins to be routed. Please check pin list or port class.
Net vdd does not have AREA10 class pad pins to be routed. Please check pin list or port class.

**WARN: (ENC5-1254): Net gnd does not have block pins to be routed. Please check pin list or port class.
**WARN: (ENC5-1255): Net gnd does not have pad pins to create pad ring. Please check pin list or port class.
**WARN: (ENC5-1256): Net gnd does not have CORE class pad pins to be routed. Please check pin list or port class.
Net gnd does not have AREA10 class pad pins to be routed. Please check pin list or port class.

CPU time for FollowPin 8 seconds
CPU time for FollowPin 8 seconds
Number of 10 ports routed: 0
Number of Block ports routed: 0
Number of Stripe ports routed: 0
Number of Core ports routed: 0
Number of Pad ports routed: 0
Number of Power Bump ports routed: 0
Number of Followpin connections: 0

End power routing: cpu: 0:00:00, real: 0:00:00, peak: 554.00 megs.

Begin updating DB with routing results...
Updating DB with 16 via definition... Extracting standard cell pins and blocks.

Pin and blockage extraction finished.

sroute post-processing starts at Mon Nov 10 16:45:28 2014
The vias is rebuilding shadow vias for net vdd.
sroute post-processing ends at Mon Nov 10 16:45:28 2014

sroute post-processing starts at Mon Nov 10 16:45:28 2014
The vias is rebuilding shadow vias for net gnd.
sroute post-processing ends at Mon Nov 10 16:45:28 2014

sroute: Total CPU time used = 0:00:10
sroute: Total Real time used = 0:01:08
sroute: Total Memory used = 0.94 megs
sroute: Total Peak Memory used = 235.93 megs
encounter 16

Vdd  Gnd
Place the Design

Standard Cells
Pre-CTS Setup time check

---

timeDesign Summary

<table>
<thead>
<tr>
<th>Setup mode</th>
<th>all</th>
<th>reg2reg</th>
<th>in2reg</th>
<th>reg2out</th>
<th>in2out</th>
<th>clkGate</th>
</tr>
</thead>
<tbody>
<tr>
<td>WNS (ns)</td>
<td>3.440</td>
<td>3.440</td>
<td>3.455</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>TNS (ns)</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Violating Paths</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>All Paths</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>DRVs</th>
<th>Real</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Nr nets(terms)</td>
<td>Worst Vio</td>
</tr>
<tr>
<td>max_cap</td>
<td>4 (4)</td>
<td>-0.003</td>
</tr>
<tr>
<td>max_tran</td>
<td>0 (0)</td>
<td>0.000</td>
</tr>
<tr>
<td>max_fanout</td>
<td>0 (0)</td>
<td>0</td>
</tr>
</tbody>
</table>

Density: 18.269%
Routing Overflow: 0.00% H and 0.00% V

Reported timing to dir :/timingReports/preCTS
Total CPU time: 0.05 sec
Total Real time: 0.0 sec
Total Memory Usage: 252.160156 Mbytes
encounter 15> encounter 15>
Clock Tree Synthesis

- Optimize the clock pin fanout
- Minimize clock skew

A typical clock configuration file

```
#-- Clock Group --
#ClkGroup
#<clockName>
#
# Clock Root : clk
# Clock Name : clk
# Clock Period : 4ns
#
AutoCtsRootPin clk
Period 4ns
MaxDelay 8.0ns # sdc driven default
MinDelay 8.0ns # sdc driven default
MaxSkew 100ps # set_clock_uncertainty
SinkMaxTran 100ps # set_clock_transition
BufMaxTran 100ps # set_clock_transition
Buffer CLKBUFF1 CLKBUFF2 CLKBUFF3 INVX1 INVX2 INVX4 INVX8
NoGating NO
DetailReport YES
#SetDPinAsSync NO
#SetIPinAsSync NO
#SetASyncSPinAsSync NO
#SetTristEnPinAsSync NO
#SetBBPinAsSync NO
RouteClkNet YES
PostOpt
OptAddBuffer YES
#RouteType specialRoute
#LeafRouteType regularRoute
END
```
Clock Tree Synthesis

Clock Analysis (CPU Time 0:00:00.0)
All RC-Corners-Per-Net-In-Memory is turned OFF...
setting up for view 'typical'...
Selecting the worst MMC view of clock tree 'clk'...
resized 0 standard cell(s).
*** Non-Gated Clock Tree Optimization (cpu=0:00:00.0 real=0:00:00.0 mem=258.8M) ***
*** Finished Clock Tree Skew Optimization (cpu=0:00:00.0 real=0:00:00.0 mem=258.8M) ***
None of the clock tree buffers/gates are modified by the skew optimization.
Switching to the default view 'typical' ...
*** None of the buffer chains at roots are modified by the fine-tune process.
All RC-Corners-Per-Net-In-Memory is turned ON...
**ERROR: (ENCSYSUTIL-96): Cannot open (for write) skew report file: "./CTSrep.skew"
Reason for error: No such file or directory.
All RC-Corners-Per-Net-In-Memory is turned OFF...
Clock gating checks

Clock gating Checks Finished, CPU=0:00:00.0

# Summary of During-Synthesis Checks

<table>
<thead>
<tr>
<th>Types of Check</th>
<th>Number of warnings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check Refine Placement move distance</td>
<td>1</td>
</tr>
<tr>
<td>Check route layer follows preference</td>
<td>0</td>
</tr>
<tr>
<td>Check route follows guide</td>
<td>0</td>
</tr>
<tr>
<td>clock gating checks</td>
<td>0</td>
</tr>
</tbody>
</table>

*** End clkSynthesis [cpu=0:00:00.2, real=0:00:00.0, mem=258.8M] ***
encounter 280.
Post CTS Setup/Hold Check

Setup Time Report

<table>
<thead>
<tr>
<th>Setup node</th>
<th>all</th>
<th>reg2reg</th>
<th>ln2reg</th>
<th>reg2out</th>
<th>ln2out</th>
<th>clkgate</th>
</tr>
</thead>
<tbody>
<tr>
<td>WNS (ns)</td>
<td>3.498</td>
<td>3.498</td>
<td>4.449</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>TNS (ns)</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Violating Paths</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>All Paths</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Hold Time Report

<table>
<thead>
<tr>
<th>Hold node</th>
<th>all</th>
<th>reg2reg</th>
<th>ln2reg</th>
<th>reg2out</th>
<th>ln2out</th>
<th>clkgate</th>
</tr>
</thead>
<tbody>
<tr>
<td>WNS (ns)</td>
<td>-0.259</td>
<td>0.091</td>
<td>-0.259</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>TNS (ns)</td>
<td>-1.036</td>
<td>0.000</td>
<td>-1.036</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Violating Paths</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>All Paths</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Density: 19.759%
Routing Overflow: 0.00% H and 0.00% V

Reported timing to dir ./timingReports/postCTS
Total CPU time: 0.05 sec
Total Real time: 0.0 sec
Total Memory Usage: 258.753986 Mbytes
encounter 21> encounter 22>
Post CTS Hold time fix

Hold Violation Fixed!

More delay cells added into the layout to fix hold violations
Routing

Routing summary

Routed Layout
### Post Routing Timing check

#### setup time/crosstalk

<table>
<thead>
<tr>
<th>Mode</th>
<th>all</th>
<th>reg2reg</th>
<th>in2reg</th>
<th>reg2out</th>
<th>in2out</th>
<th>clkgate</th>
</tr>
</thead>
<tbody>
<tr>
<td>WNS (ns)</td>
<td>3.497</td>
<td>3.497</td>
<td>4.971</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>TNS (ns)</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

- Violating Paths: 0
- All Paths: 8

#### Hold time

<table>
<thead>
<tr>
<th>Mode</th>
<th>all</th>
<th>reg2reg</th>
<th>in2reg</th>
<th>reg2out</th>
<th>in2out</th>
<th>clkgate</th>
</tr>
</thead>
<tbody>
<tr>
<td>WNS (ns)</td>
<td>0.094</td>
<td>0.094</td>
<td>0.094</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>TNS (ns)</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

- Violating Paths: 0
- All Paths: 8

---

**Density:** 27.590%

- Total number of glitch violations: 0

---

**Timing passed!**

**Crosstalk check passed!**
Filler cells placed to avoid DRC errors.
Final DRC Check

*********End: VERIFY GEOMETRY*********
*** verify geometry (CPU: 0:00:00.0  MEM: 1.5M)

encounter 28> **Info: (ENCSP-307): Design contains fractional 20 cells.
*INFO: Adding fillers to top-module.
*INFO: Added 295 filler insts (cell FILL / prefix FILLER).
*INFO: Total 295 filler insts added - prefix FILLER (CPU: 0:00:00.0).
For 295 new insts, ** Applied 2 GNC rules (cpu = 0:00:00.0).
*INFO: Checking for DRC violations on added fillers.
*INFO: Iteration 0-#1, Found 0 DRC violation (real: 0:00:00.0).
*INFO: Adding fillers to top-module.
*INFO: Added 0 filler inst of any cell-type.
For 0 new insts, ** Applied 0 GNC rules.
*INFO: End DRC Checks. (real: 0:00:00.0).
encounter 28> *** Starting Verify Geometry (MEM: 271.8) ***

VERIFY GEOMETRY ...... Starting Verification
VERIFY GEOMETRY ...... Initializing
VERIFY GEOMETRY ...... Deleting Existing Violations
VERIFY GEOMETRY ...... Creating Sub-Areas
...... bin size: 2000
VERIFY GEOMETRY ...... SubArea : 1 of 1
VERIFY GEOMETRY ...... Cells : 0 Viols.
VERIFY GEOMETRY ...... SameNet : 0 Viols.
VERIFY GEOMETRY ...... Wiring : 0 Viols.
VERIFY GEOMETRY ...... Antenna : 0 Viols.
VERIFY GEOMETRY ...... SubArea : 1 complete 0 Viols. 0 Wrngs.
VG: elapsed time: 0.80
Begin Summary ... 
Cells : 0
SameNet : 0
Wiring : 0
Antenna : 0
Short : 0
Overlap : 0
End Summary

Verification Complete : 0 Viols. 0 Wrngs

*********End: VERIFY GEOMETRY*********
*** verify geometry (CPU: 0:00:00.0  MEM: 1.3M)

DRC passed!
What’s next ... ?

• Stream the layout into a .gds file
• Stream the timing into a .sdf file.
• Stream the final netlist into a .v file.

• Import the netlist, timing, and .gds into cadence virtuoso for co-simulation
• If you are happy with the results, send the .gds file to the foundry to get your chip fabricated! (which usually takes around 2 month)