

CMP305

Very Large Scale Integrated

Circuit

Lab 1

ASIC Flow (Part 1)

Objectives

- Understand:
 - Difference between Chip Design Flows
 - ASIC Architecture
 - ASIC Design Flow
 - How VHDL maps to Hardware
- Learn:
 - How to use LeonardoSpectrum
 - Design Synthesis
 - Post-Synthesis Simulation

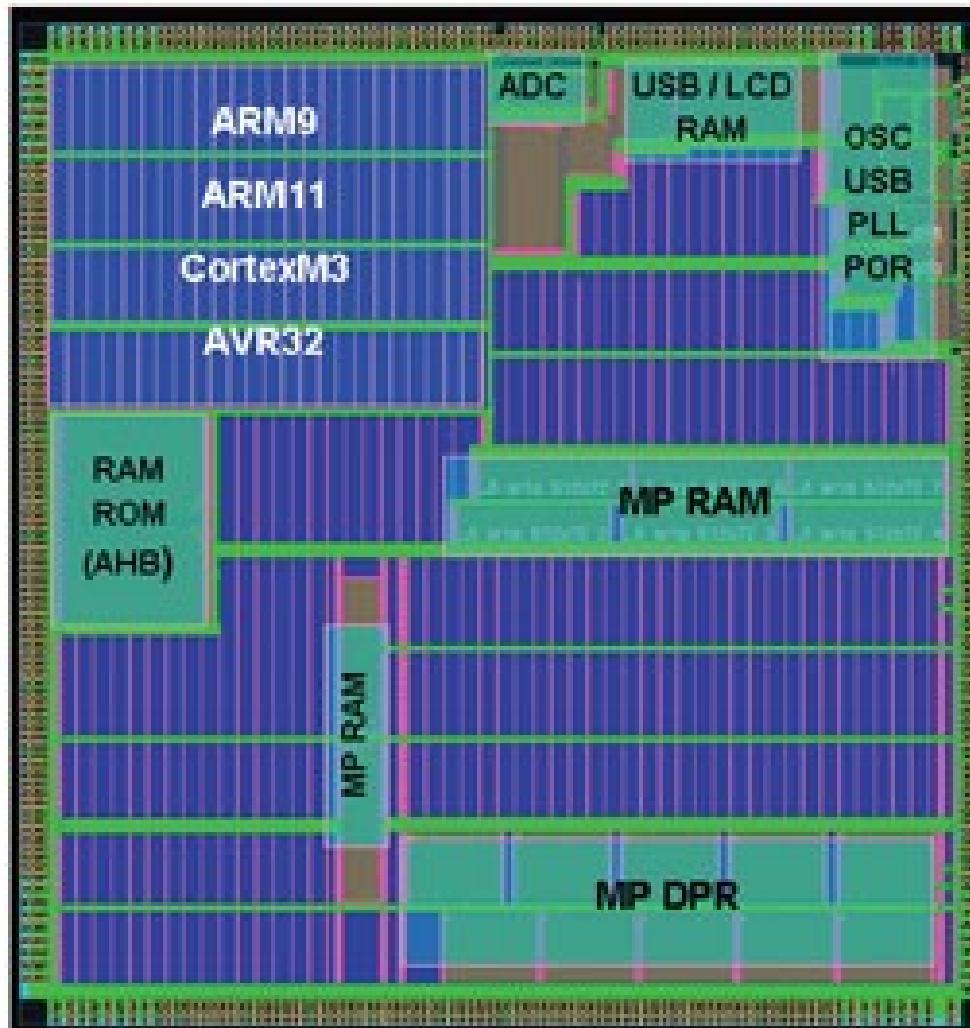
Chip Design

- There are 3 different flows:
 - Full Custom Design
 - ASIC Design
 - FPGA Design

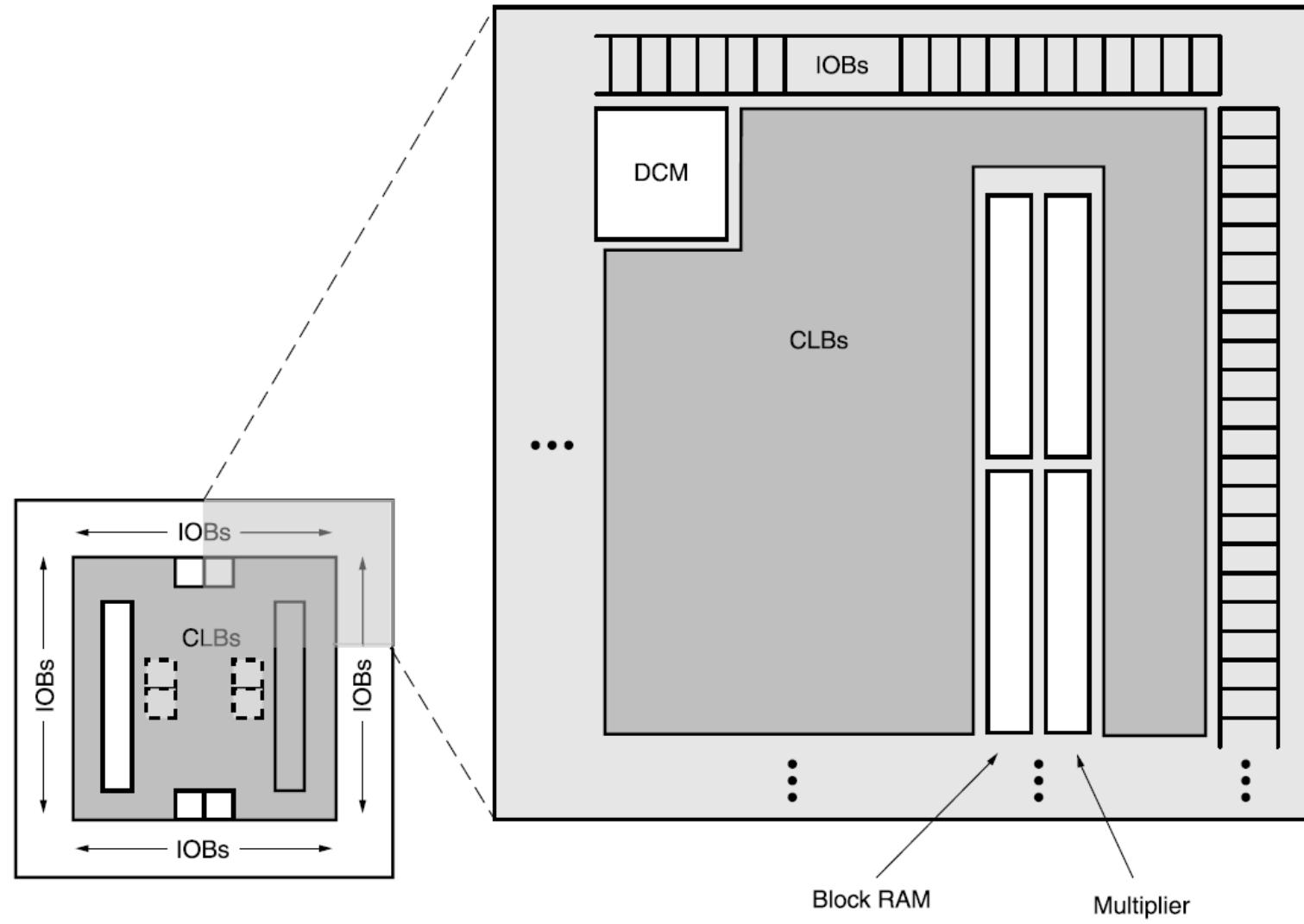
FPGA vs ASIC

FPGA	ASIC
Gates already Exists	Gates will be fabricated according to your design
Reusability (Field Programmable)	Low Power
Special Hardware	Low Cost
Prototyping	Analog/MEMS Designs

ASIC Architecture



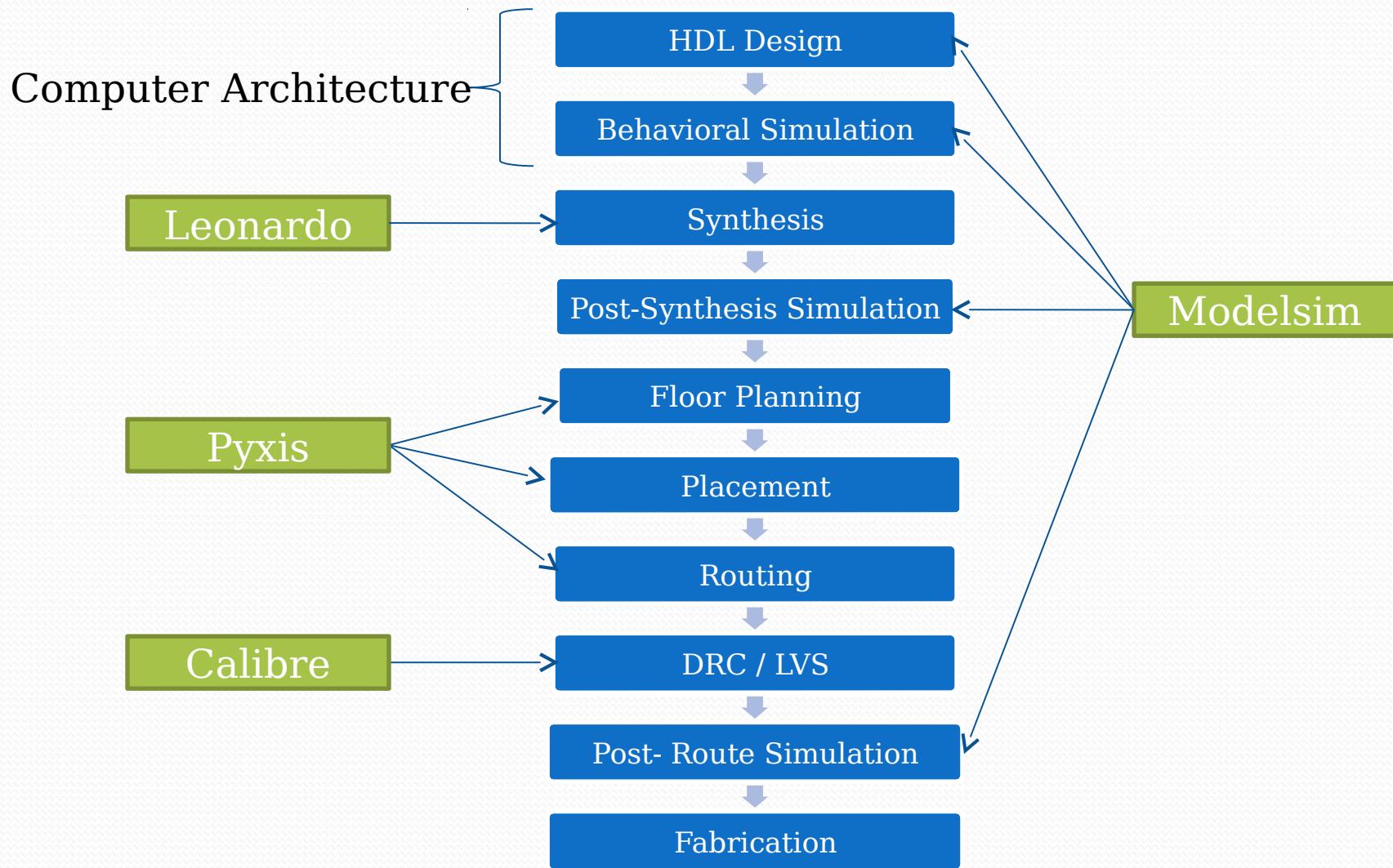
FPGA Architecture



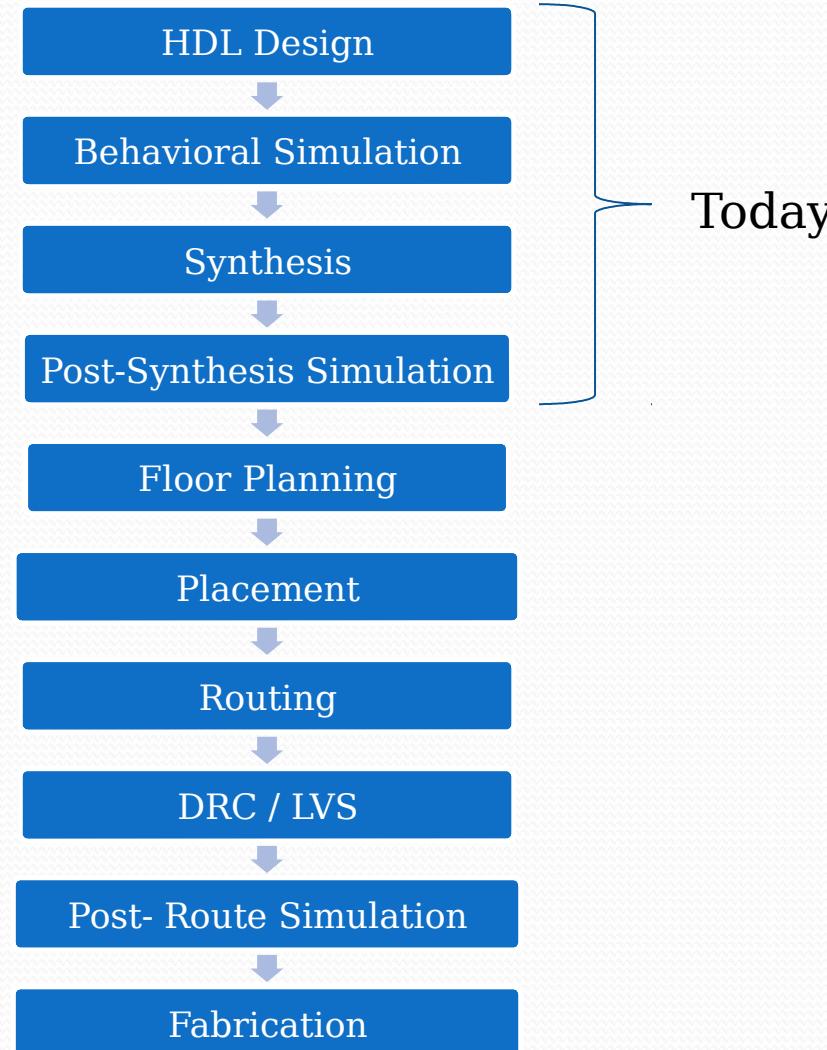
EDA Vendors

- The most popular EDA tools vendors
 - Synopsys
 - Cadence
 - Mentor Graphics
- We will use Mentor tools for our ASIC tutorial

ASIC Design Flow



ASIC Design Flow



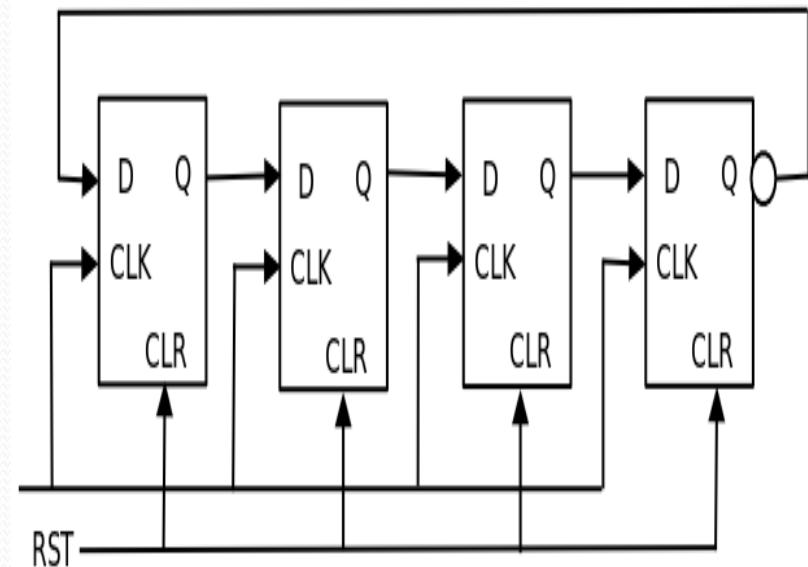
Mapping VHDL to Hardware

- When writing design code, put in your mind what we talked about last term
 - If .. Elsif .. Else / When .. Else (Cascaded Mux / Priority Encoder)
 - Case .. When / With .. Select .. When (Large Mux)
 - Combinational Design using Processes
 - Sensitivity List
 - Conditions of “If” statement
 - Always use “Else” or “Default”

Johnson Counter

- We will exercise on 8-bit Johnson Counter

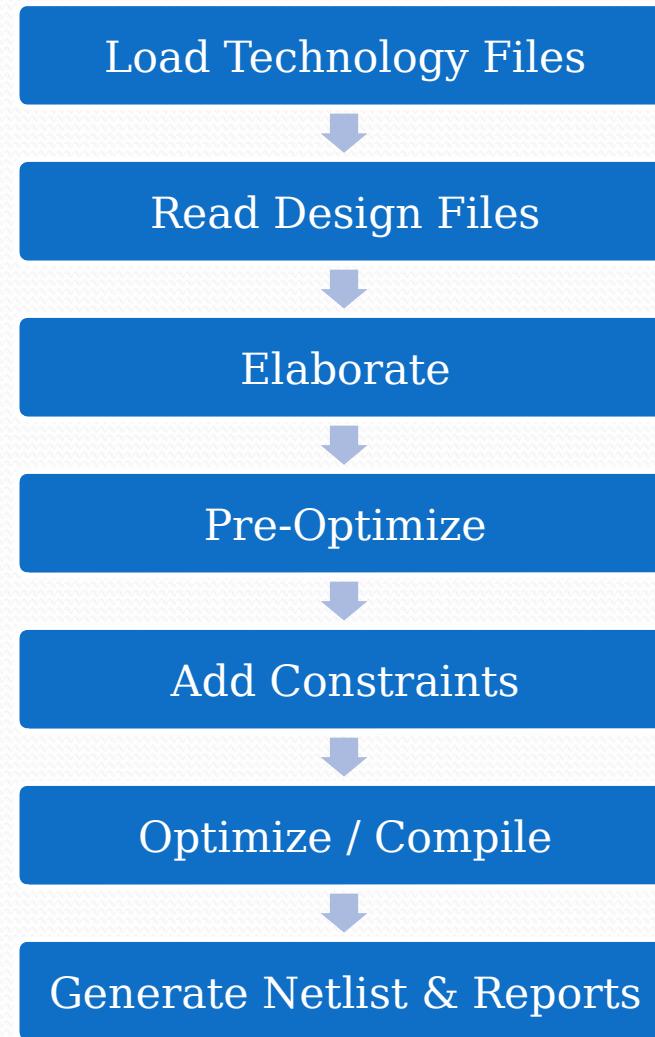
State	Q0	Q1	Q2	Q3
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1
0	0	0	0	0



Synthesis

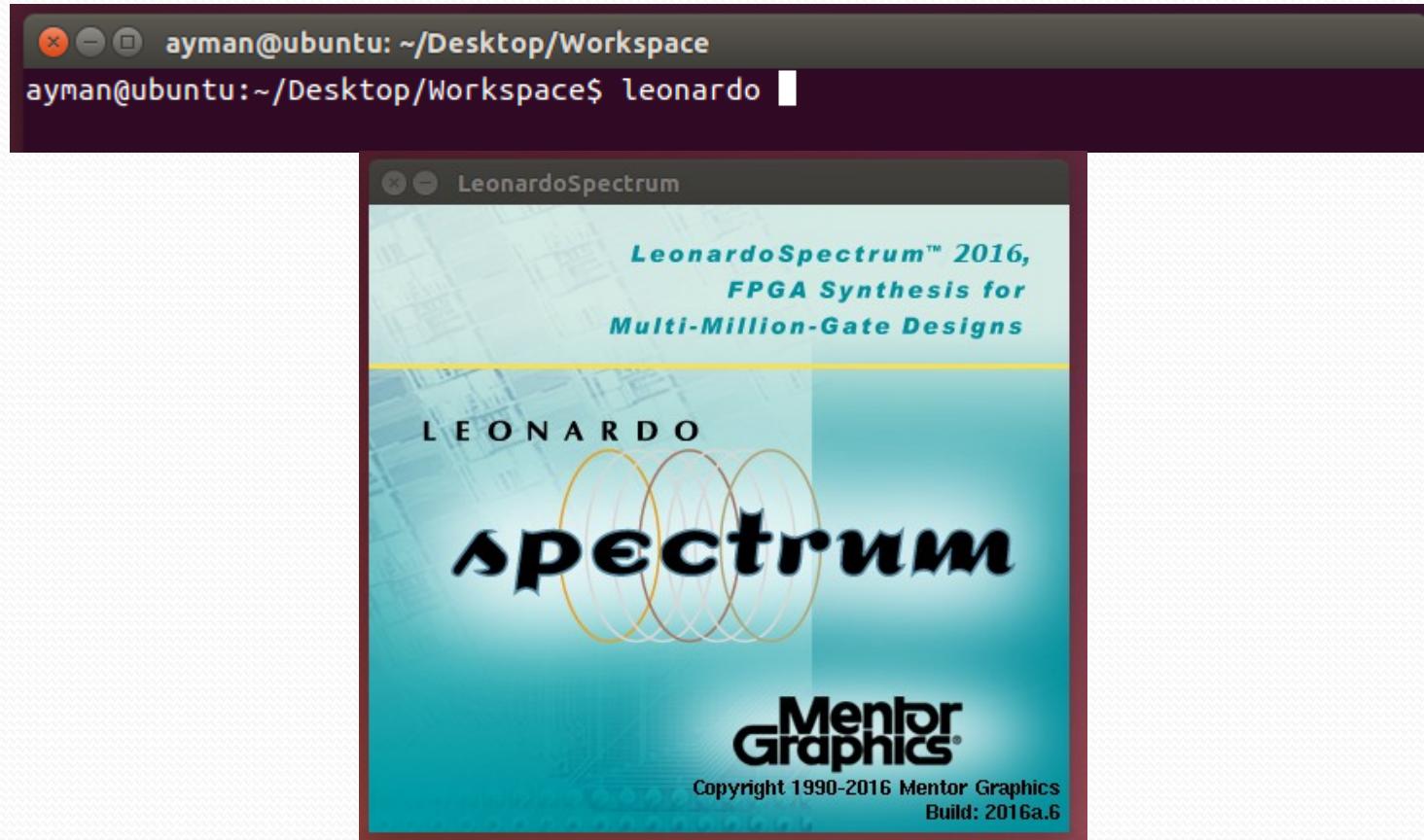
- Check the Design files & Syntax
- Translate Behavioral Design to Gate-Level Netlist

Synthesis Flow

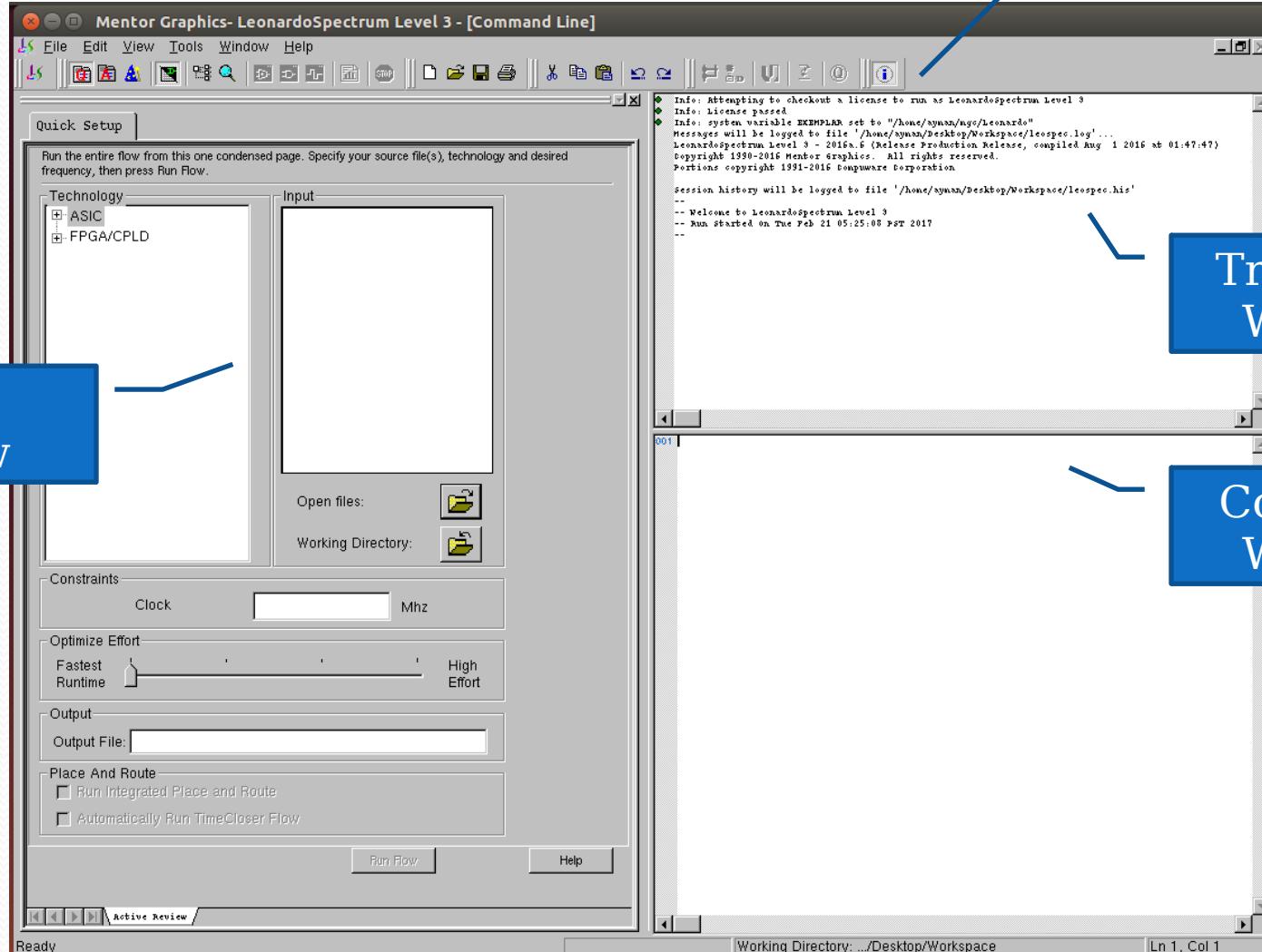


Leonardo

- Open Leonardo from the terminal



Leonardo



Task Bar

Transcript
Window

Action
Window

Command
Window

Transcript Window

- Shows Leonardo messages

```
--Run.Started.On.Tue.Feb.21.06:04:23.PST.2017
-->set.register2register.100.000000
Info::setting.register2register.to.100.000000
100.000000
->set.input2register.100.000000
Info::setting.input2register.to.100.000000
100.000000
->set.register2output.100.000000
Info::setting.register2output.to.100.000000
100.000000
->set.bubble_tristates.FALSE
Info::setting.bubble_tristates.to.FALSE
FALSE
->load_library.ami05_typ
Reading.library.file.~/home/ayman/mgc/Leonardo/lib/ami05_typ.syn...
Library.version.=..Release...Patch.(a)...(Sep.23,.2005)
Delays.assume:.Process=typical.Temp=..0.0.C..Voltage=5.00.V..
Info::setting.encoding.to.auto
Info,.Command.'load_library'.finished.successfully
->read..{./home/ayman/Desktop/Workspace/Code/Johnson_count.vhd.}
---Reading.file./home/ayman/mgc/Leonardo/data/standard.vhd.for.unit.standard
---Loading.package.standard.into.library.std
---Reading.vhdl1.file./home/ayman/Desktop/Workspace/Code/Johnson_count.vhd.into.library.work
---Reading.file./home/ayman/mgc/Leonardo/data/std_1164.vhd.for.unit.std_logic_1164
---Loading.package.std_logic_1164.into.library.ieee
```

Command Window

- Enter Direct Commands

```
001 set.register2register.100.000000
002 set.input2register.100.000000
003 set.register2output.100.000000
004 set.bubble_tristates.FALSE
005 load_library.ami05_typ
006 read..{./home/ayman/Desktop/Workspace/Code/Johnson_count.vhd..}
007 pre_optimize--common_logic--unused_logic--boundary--xor_comparator_optimize.
008 pre_optimize--extract.
009 pd
010 optimize..work.Johnson_count.J2--target.ami05_typ--macro--auto--effort.quick--hierarchy.auto.
011 optimize_timing..work.Johnson_count.J2.
012 report_area--cell_usage--all_leafs.
013 set.report_delay_slack_threshold.0.000000
014 report_delay--num_paths.1--critical_paths--clock_frequency
015 set.novendor_constraint_file.FALSE
016 auto_write.Johnson_count_Gate.vhd
017
```

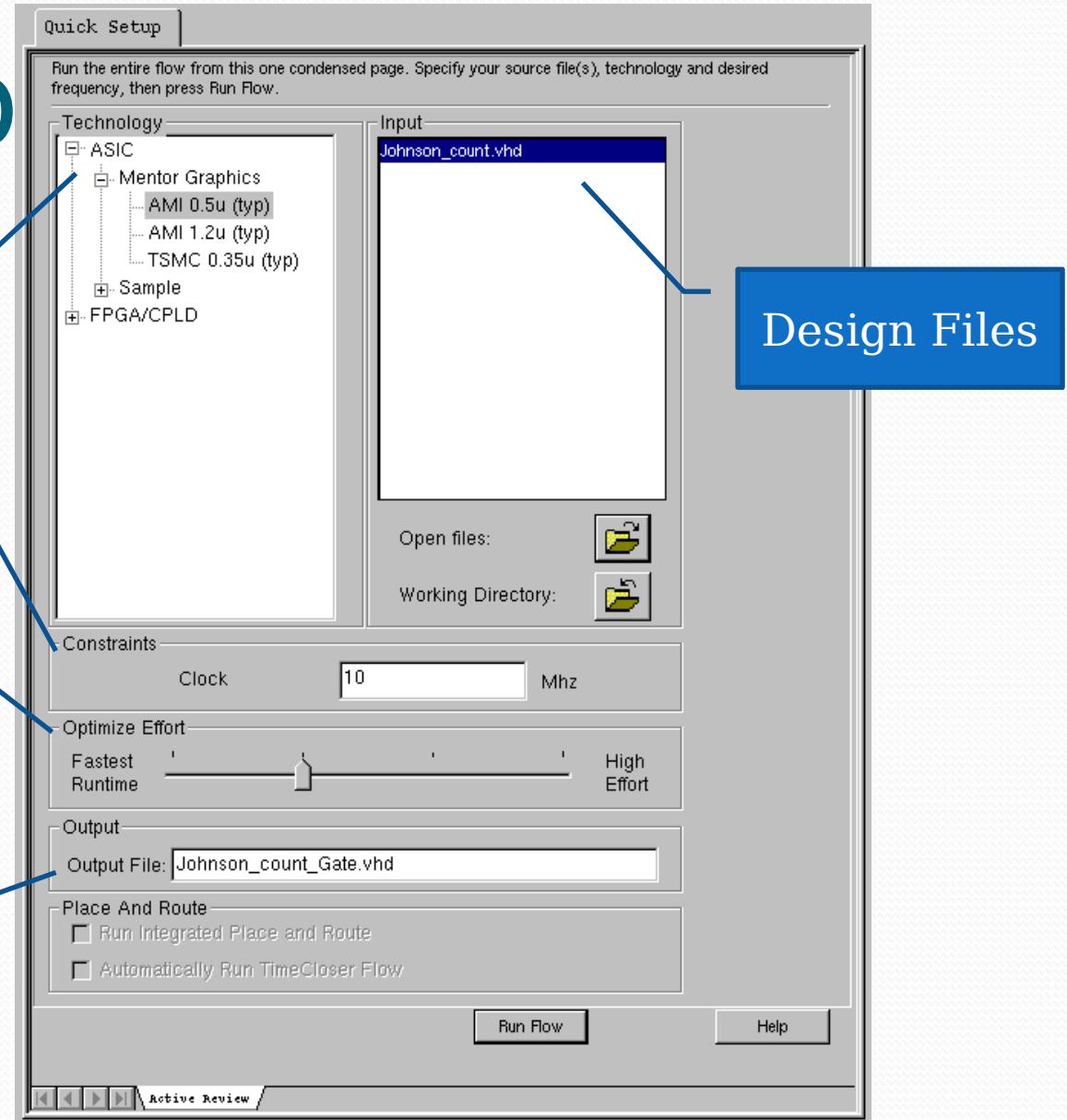
Quick Tab

Technology
Files

Clock Freq

Optimization

Output
Netlist



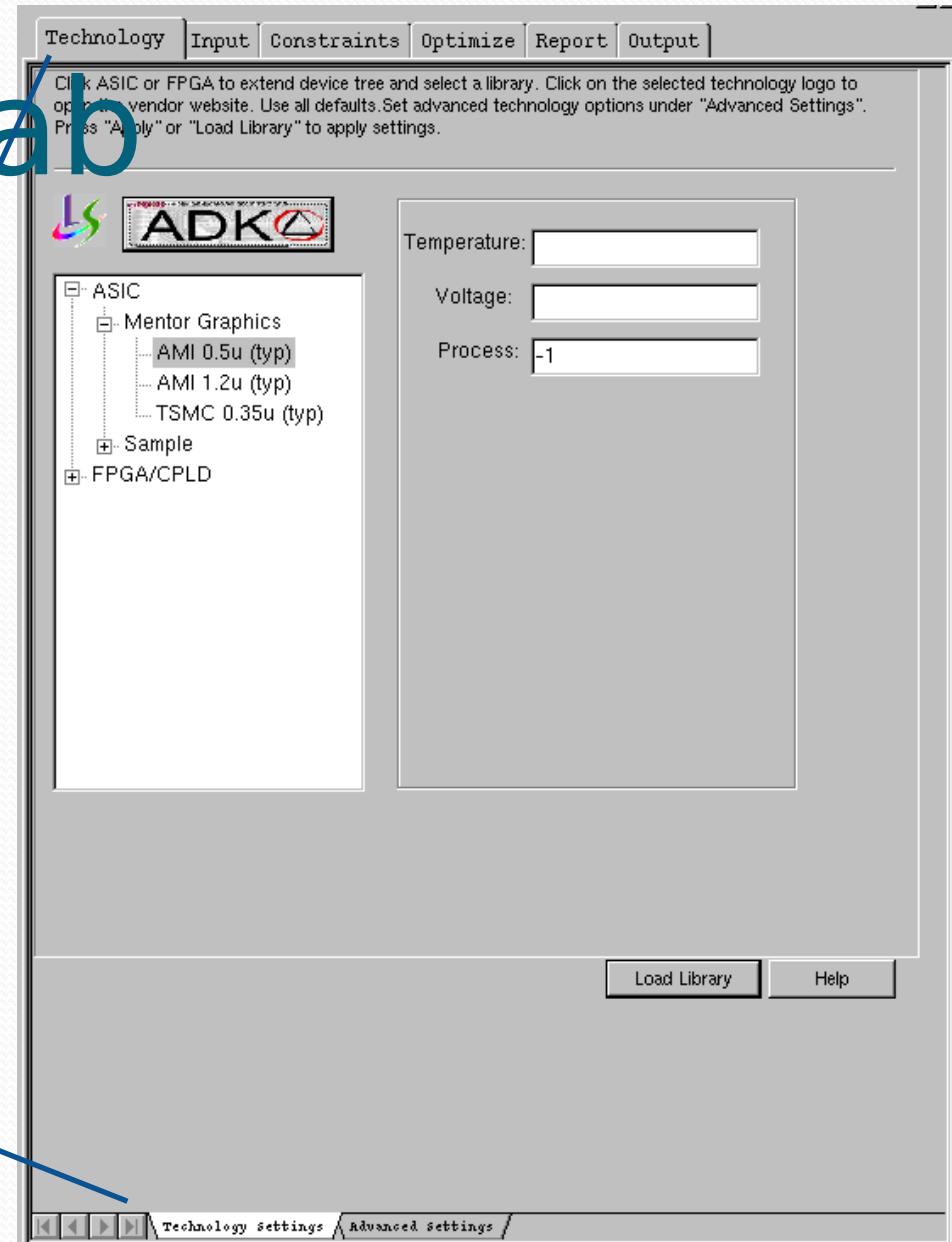
Design Files

Advanced Tab

- Detailed Steps

Steps Tabs

Power Tabs



Synthesis

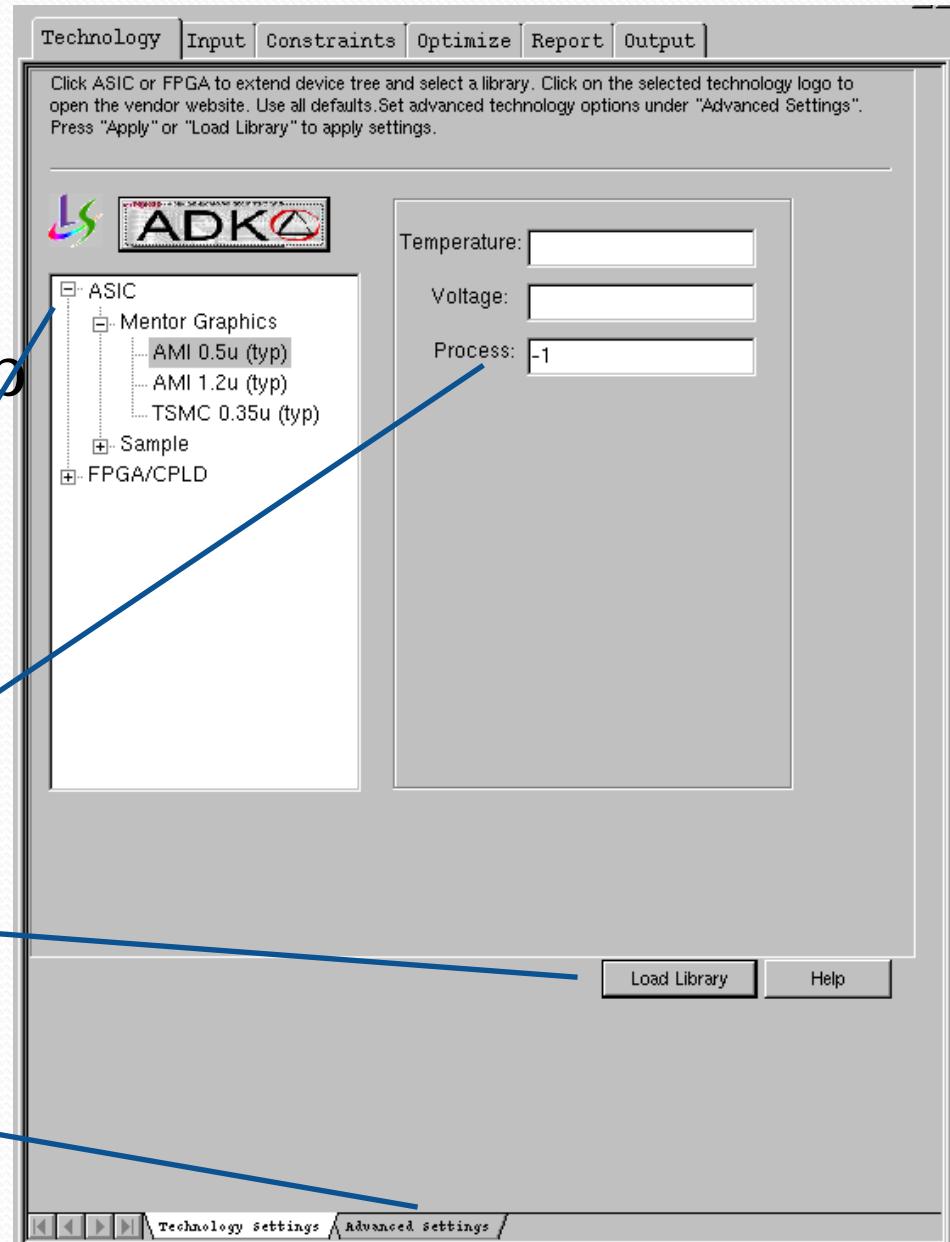
- Step 1:
 - Load Technology Lib

Technology
Files

Tech Details

Load Button

Advanced
Settings

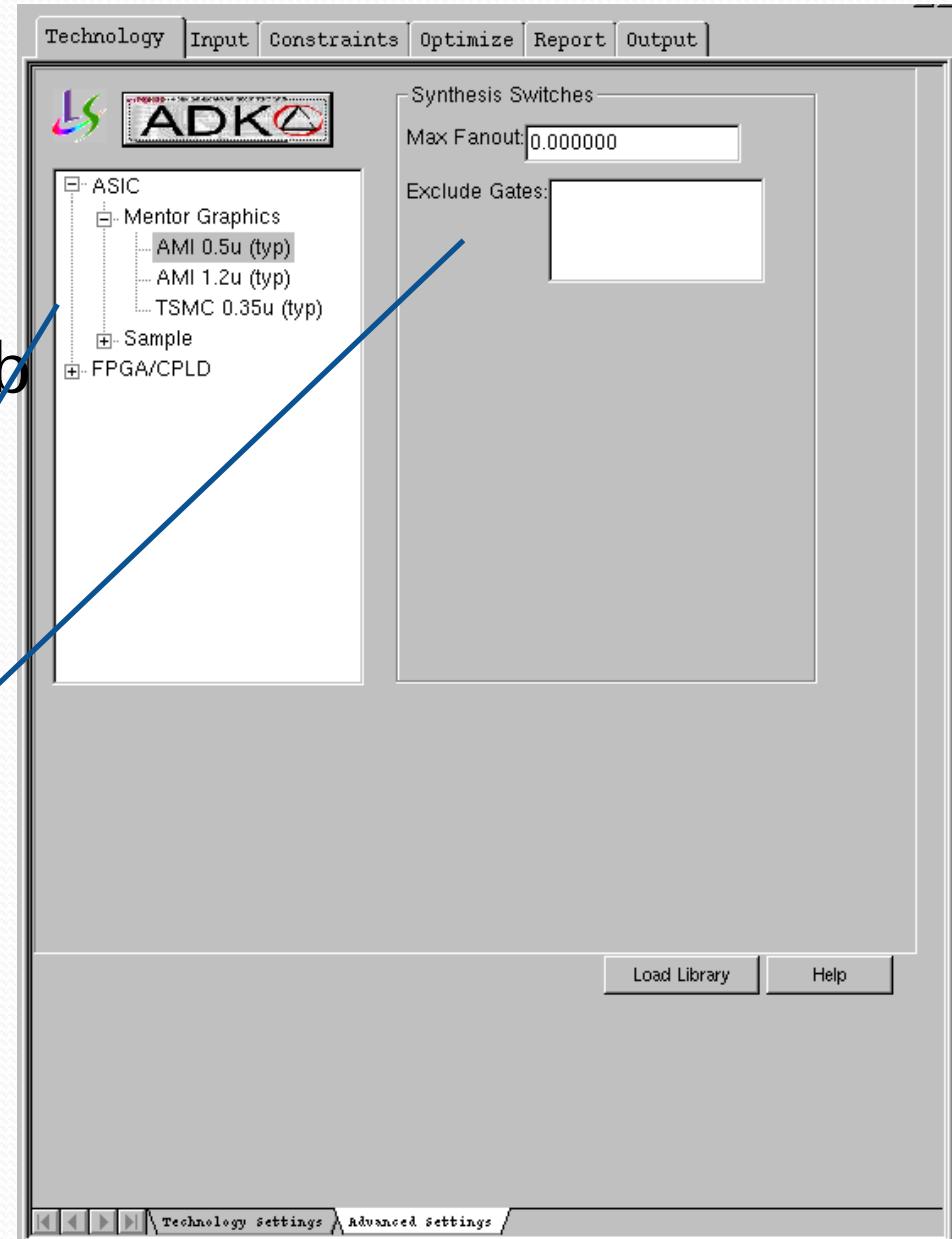


Synthesis

- Step 1:
 - Load Technology Lib

Technology
Files

Fanout &
Exclude
Gates



Synthesis

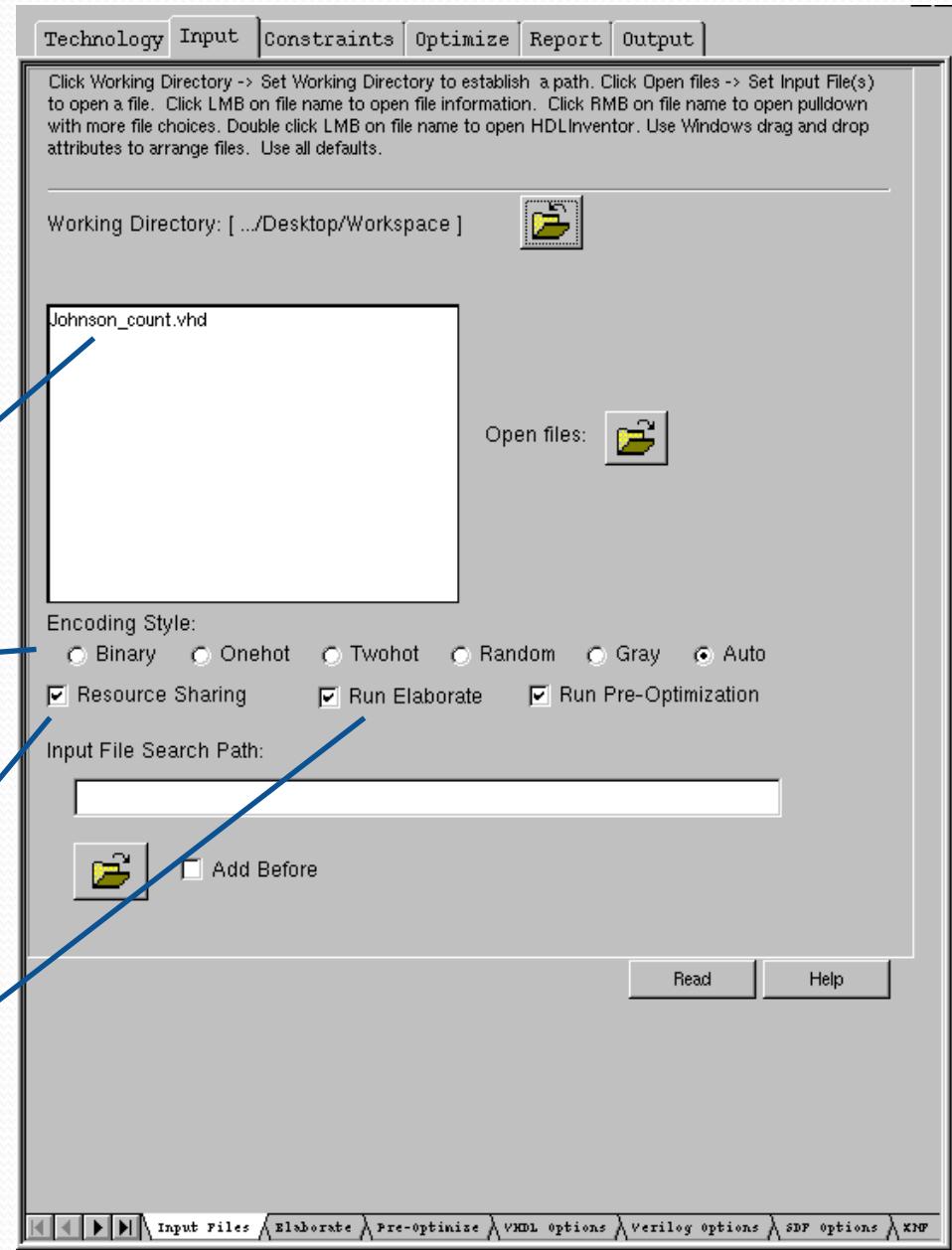
- Step 2:
 - Read Design Files

Design Files

Encoding of
FSM

Common
Logic

Elaborate

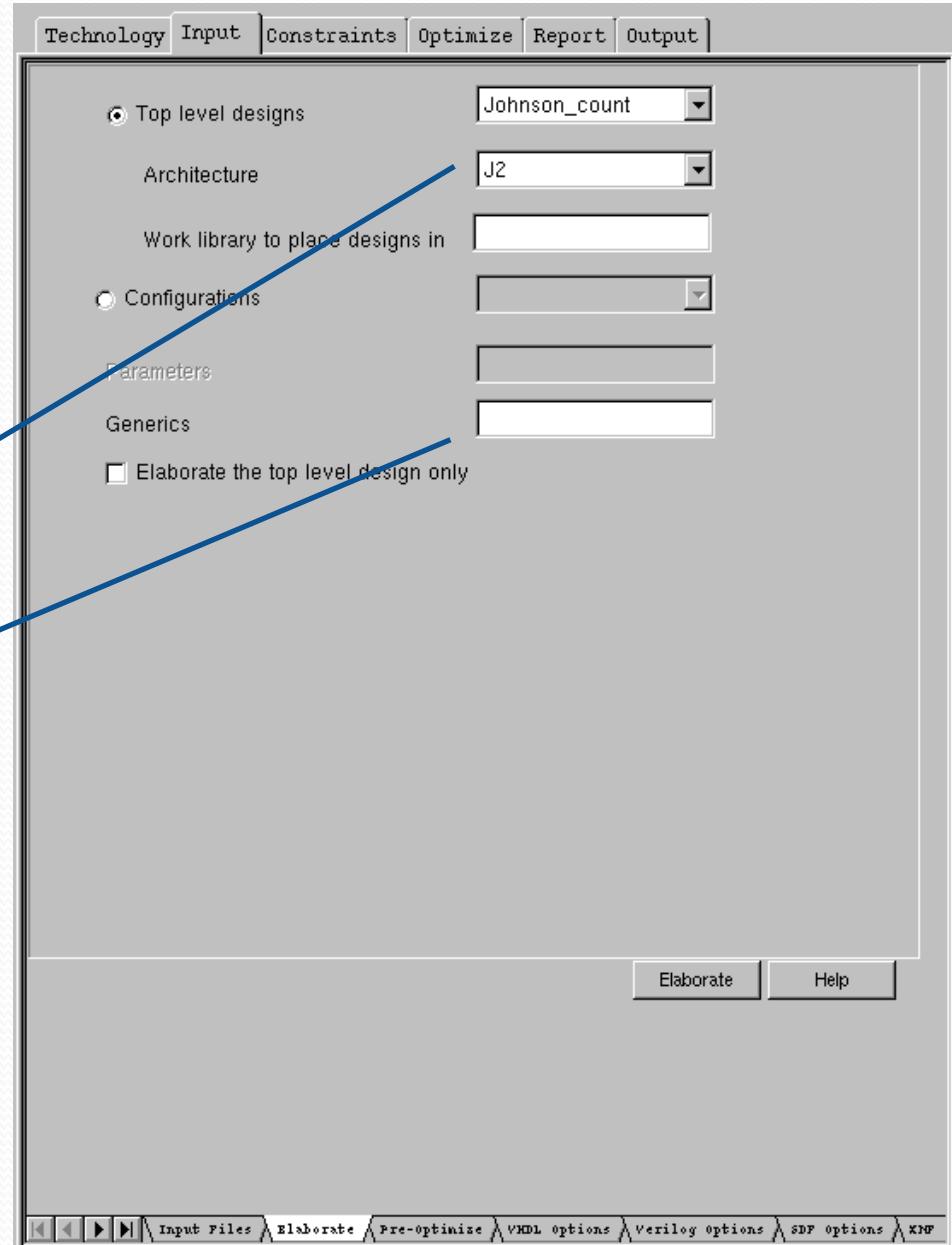


Synthesis

- Step 3:
 - Elaborate

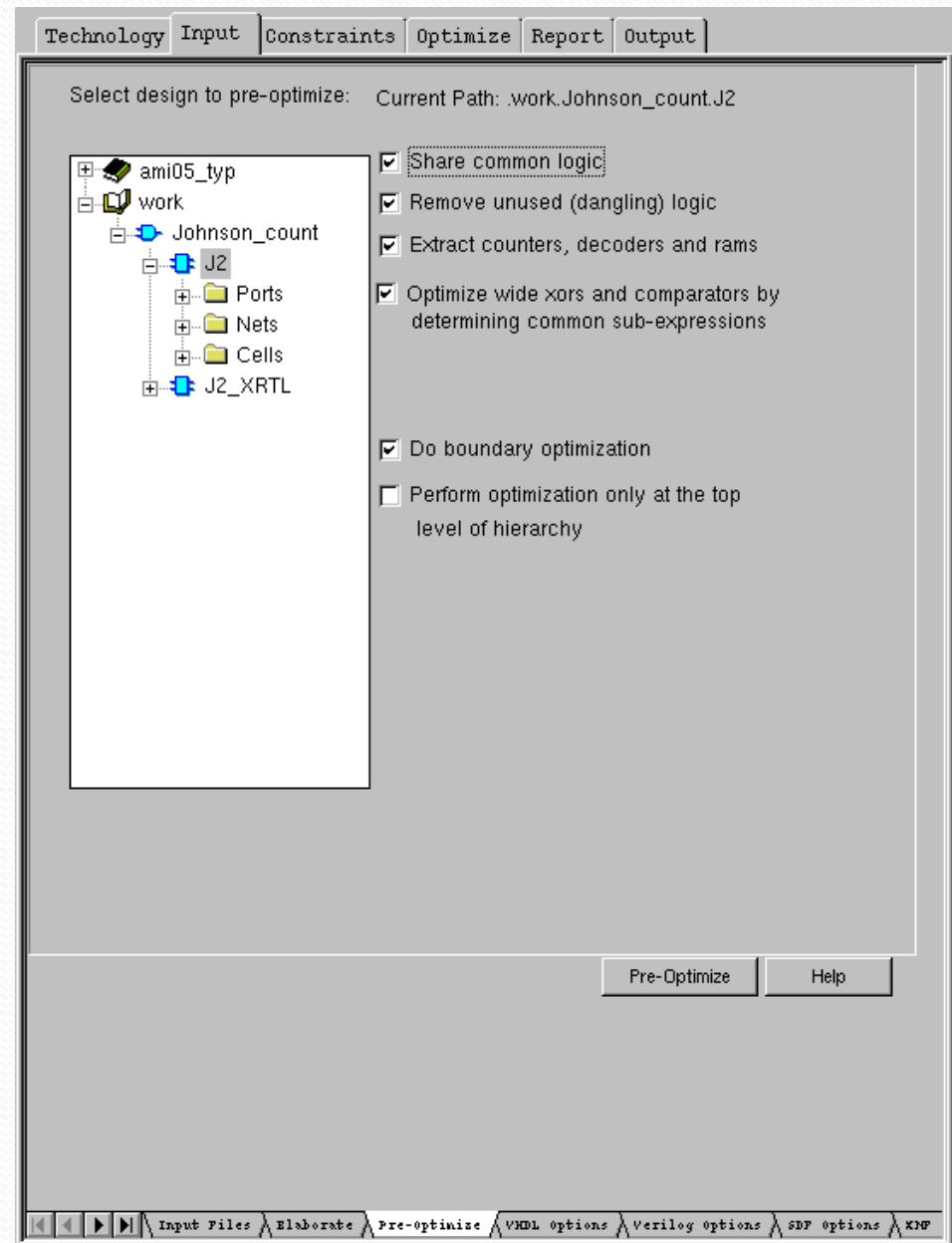
Select Top Level

Set Value to Generic Signals



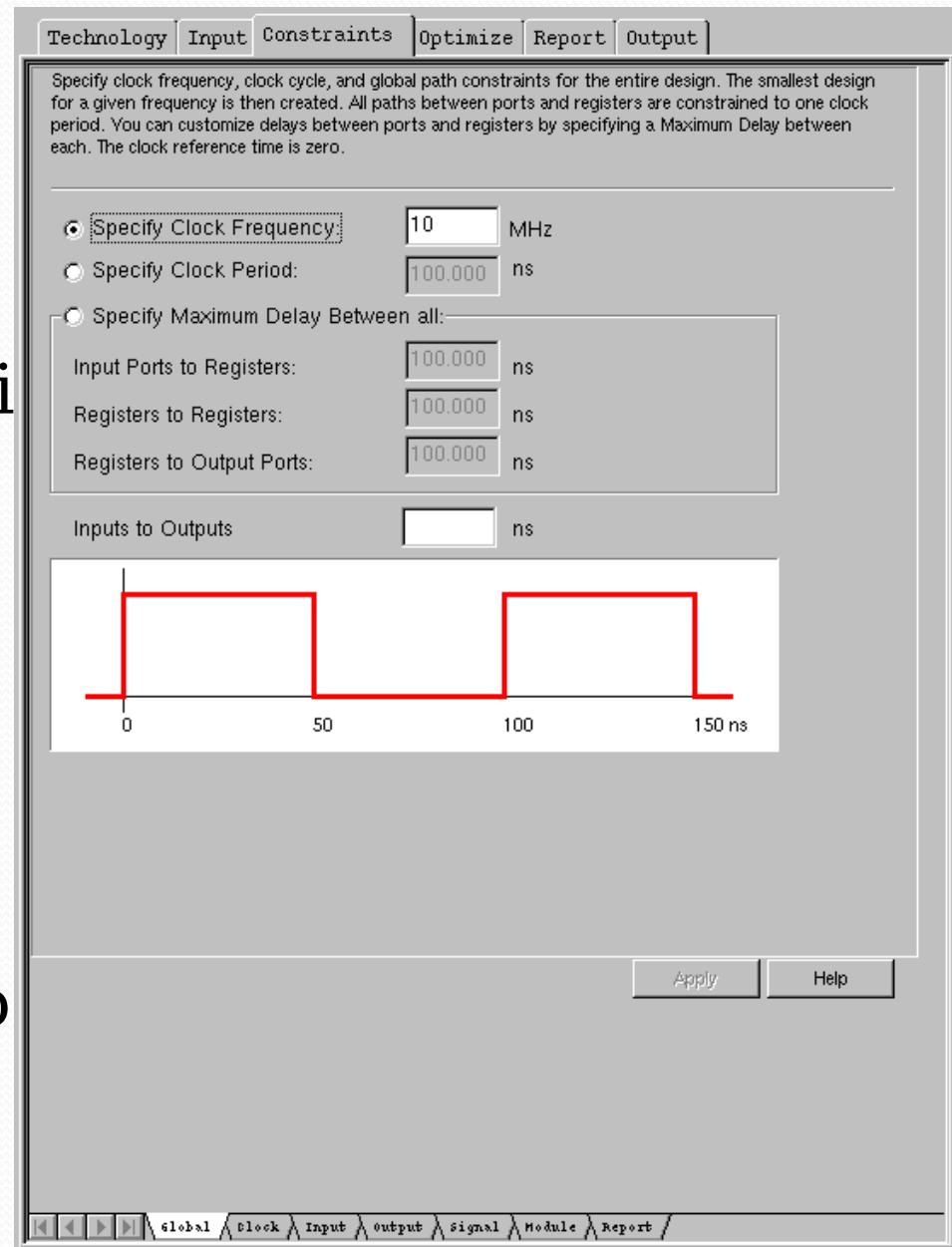
Synthesis

- Step 4:
 - Pre-Optimize



Synthesis

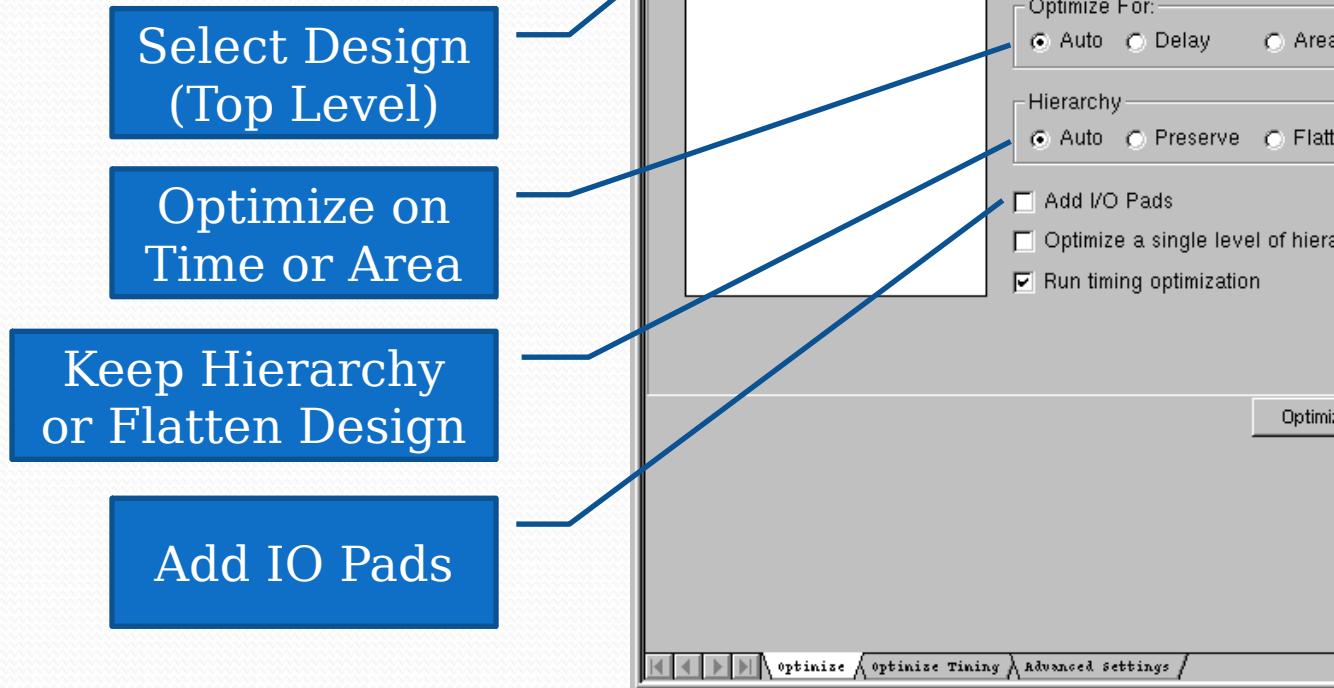
- Step 5:
 - Add Design Constraints
 - Specify in General



- We will see more about constraints next lab

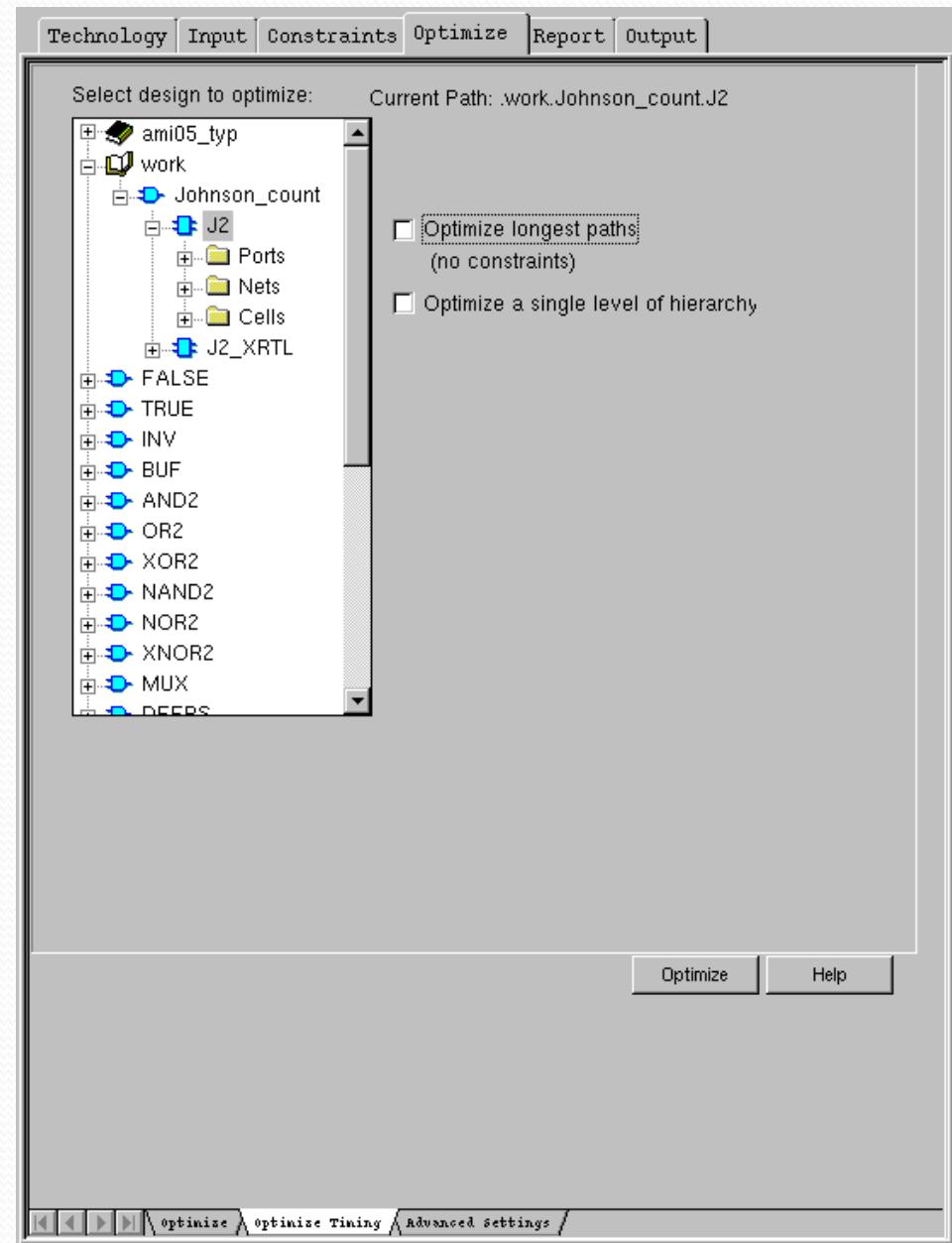
Synthesis

- Step 6:
 - Optimize
 - Synthesis the design



Synthesis

- Step 6:
 - Optimize (Timing)



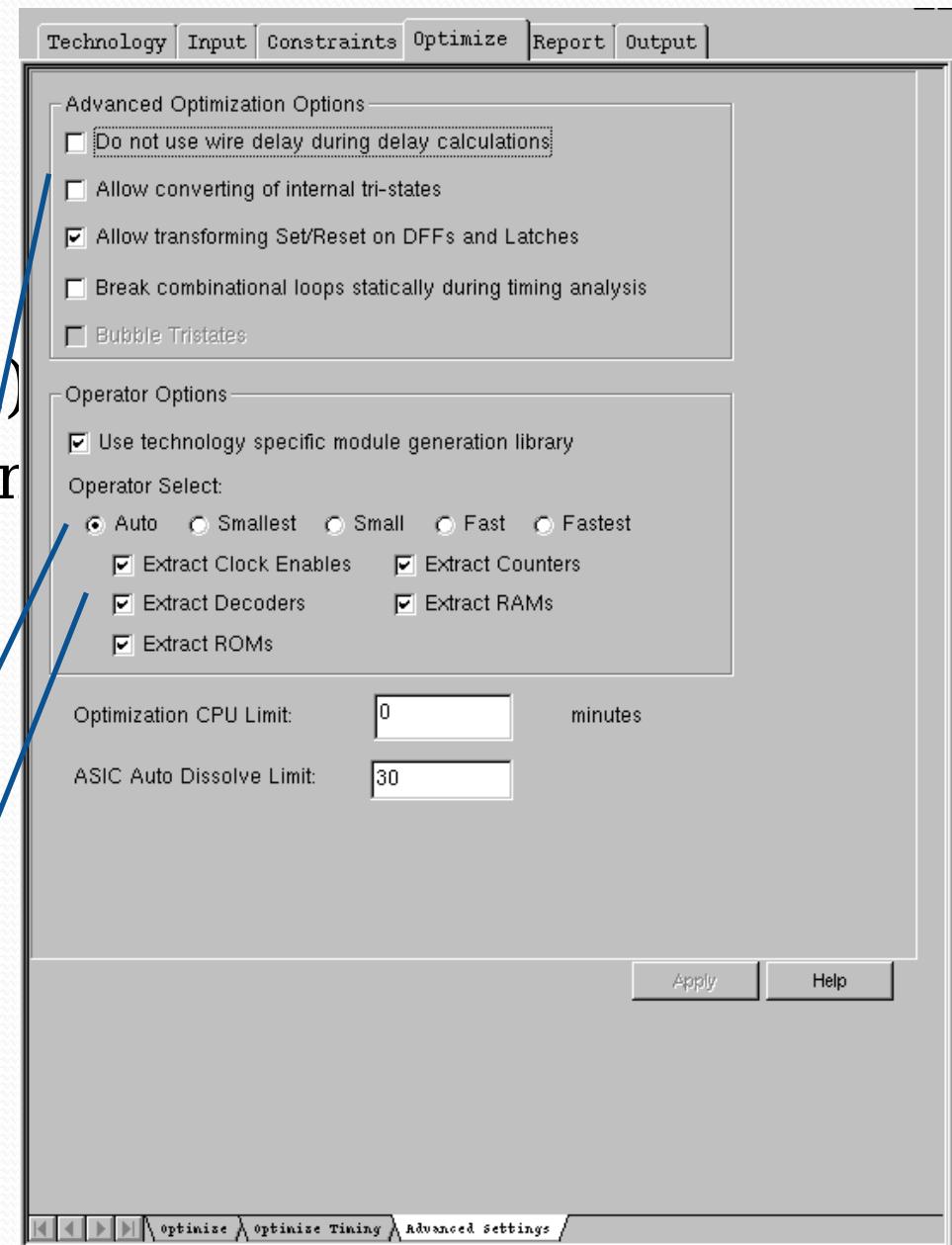
Synthesis

- Step 6:
 - Optimize (Advanced)
 - Synthesis the design

Optimization Options

Synthesized Gates

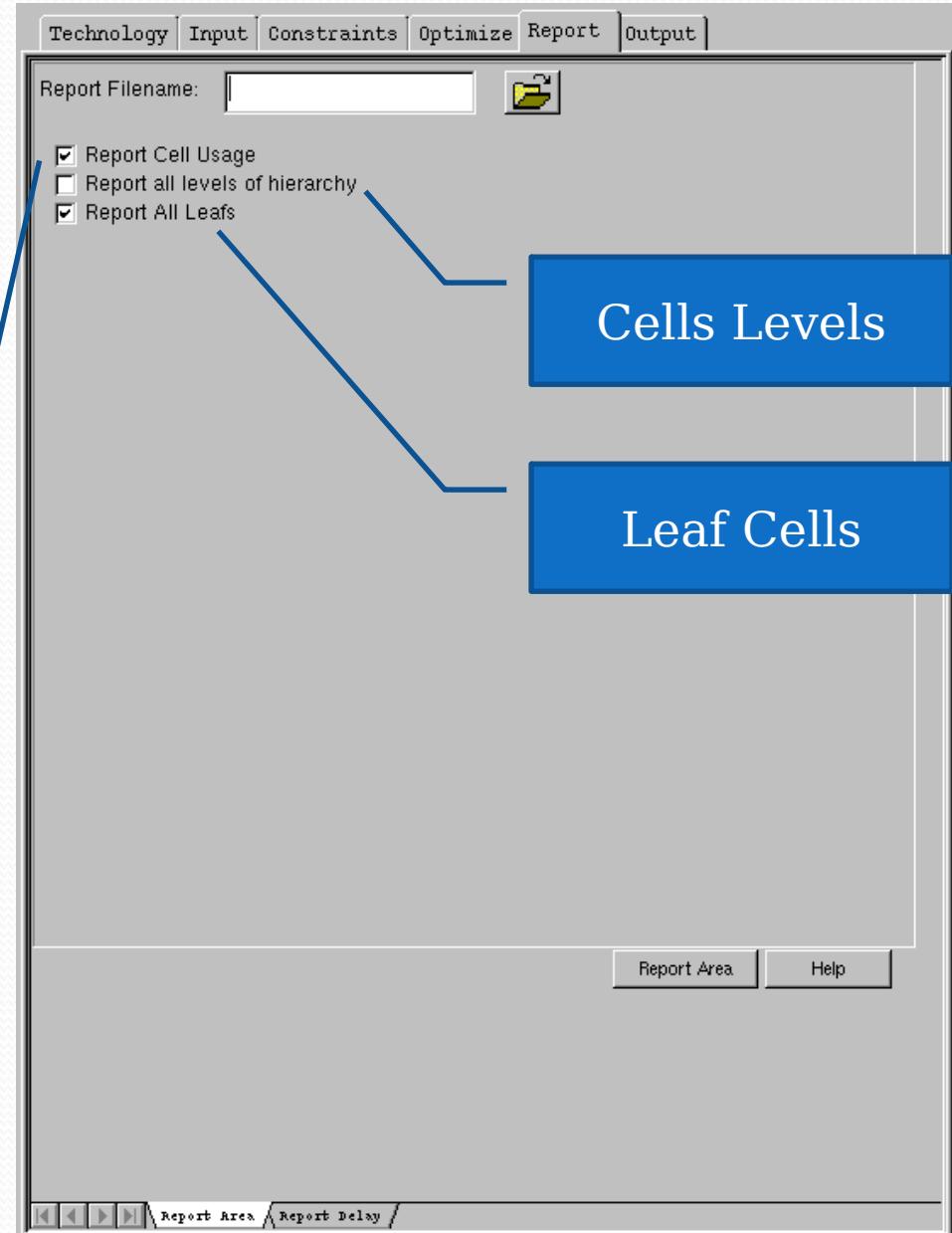
Use Technology Modules



Synthesis

- Step 7:
 - Generate Reports & Netlist
 - Report Area

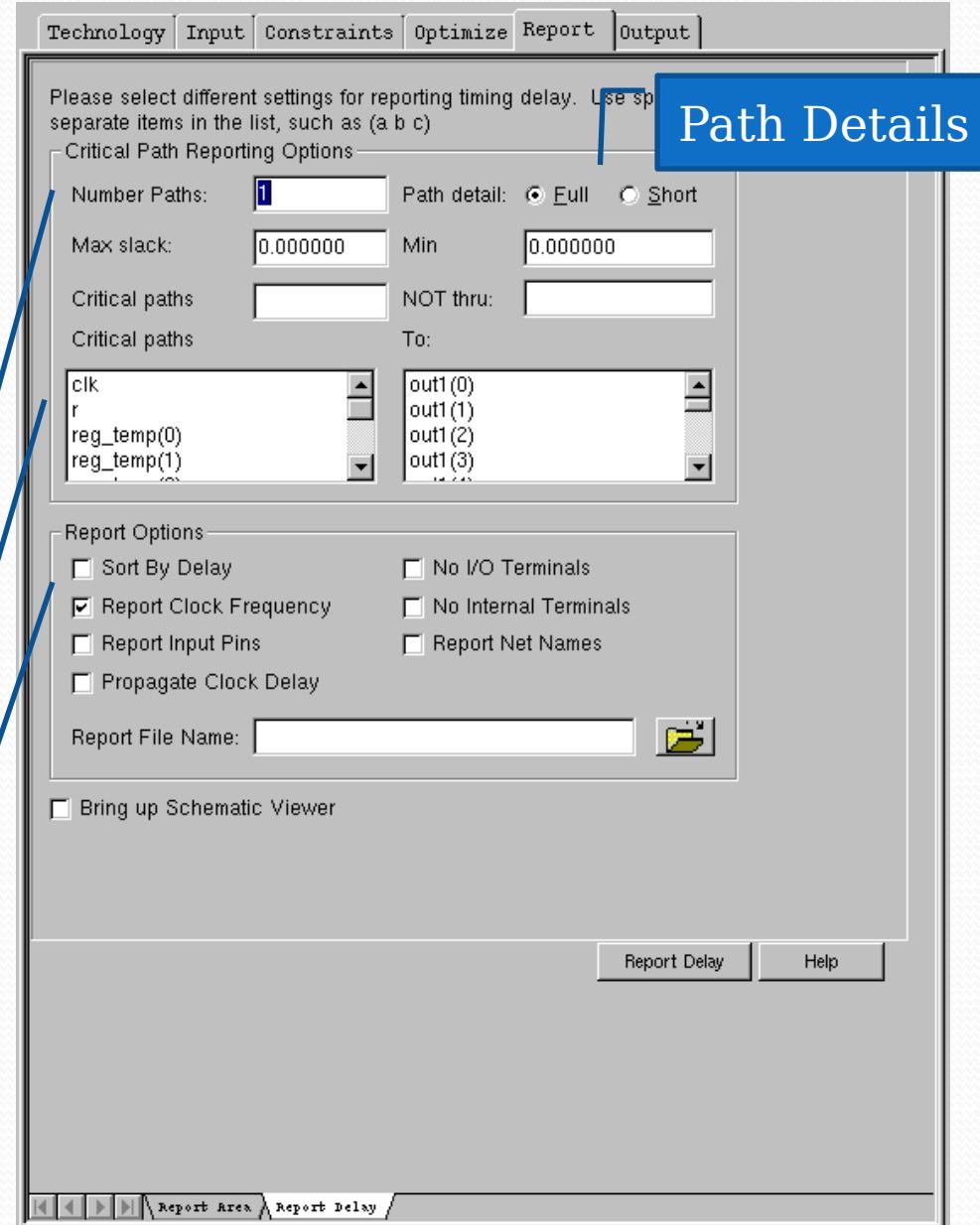
Cells Summary



Synthesis

- Step 7:
 - Generate Reports & Netlist
 - Report Timing

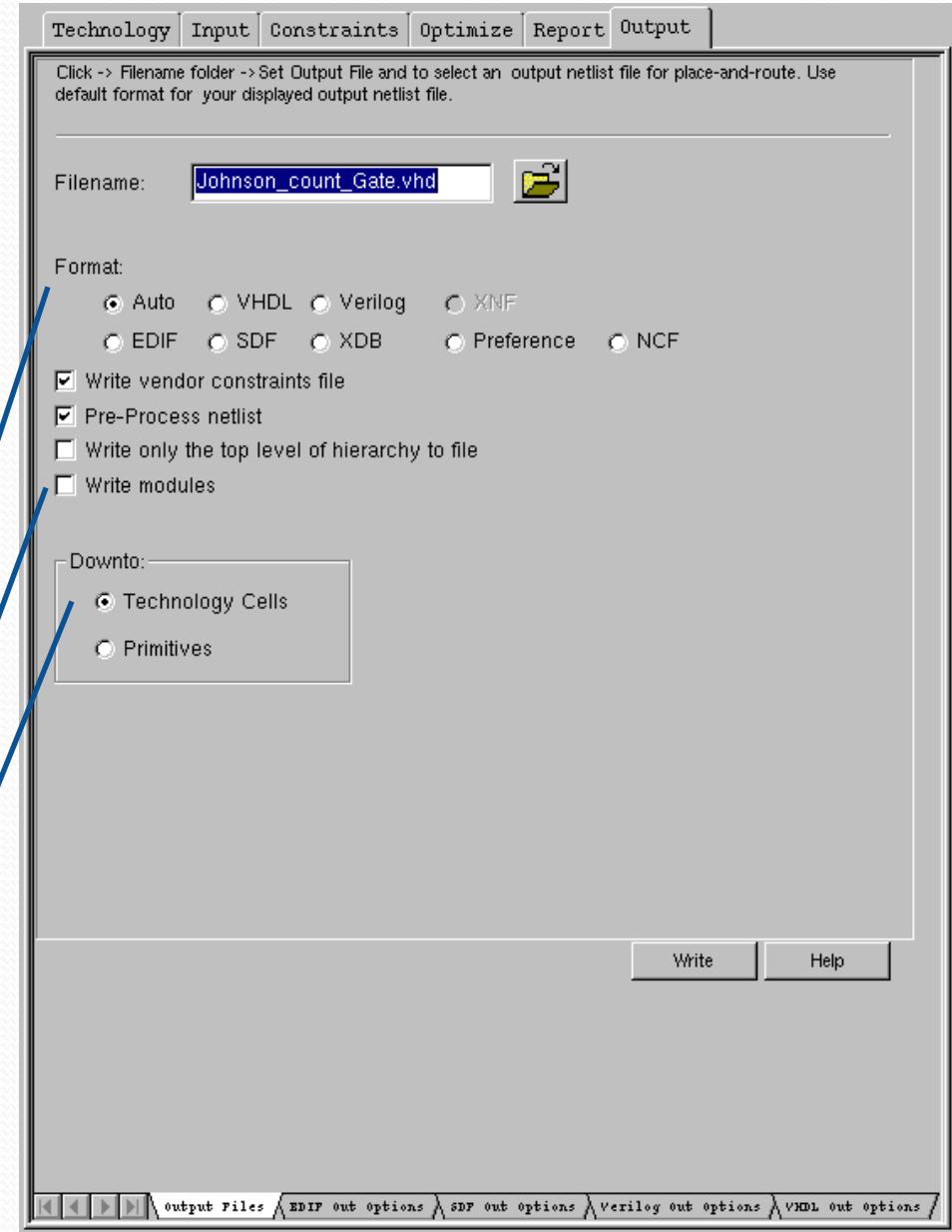
- Number of Paths
- Specific Path
- More Details



Synthesis

- Step 7:
- Generate Reports & Netlist
 - Netlist File

File Format
Write as single Module
Technology Cells



Synthesis

- Check the synthesis output
 - Area Report
 - Timing Report
 - RTL Schematic
 - Technology Schematic
 - Critical Path Schematic



Synthesis

- Area Report

```
*****
Cell::Johnson_count....View::J2....Library::work
*****
.Cell.....Library..References.....Total.Area
PadInC.....ami05_typ.....8.x.....1.....8.PadInC
dff.....ami05_typ.....8.x.....5.....38.gates
nand02.....ami05_typ.....7.x.....1.....7.gates
nor02_2x.....ami05_typ.....1.x.....1.....1.gates

.Number.of.ports.....10
.Number.of.nets.....26
.Number.of.instances.....24
.Number.of.references.to.this.view.....0

Total.accumulated.area::
.Number.of.PadInC.....8
.Number.of.gates.....46
.Number.of.accumulated.instances.....24
Info,.Command.'report_area'.finished.successfully
```

Synthesis

- Timing Report

```
.....Clock.Frequency.Report
>> Clock.....Frequency
-----
>> clk.....45.6.MHz

.....Critical.Path.Report

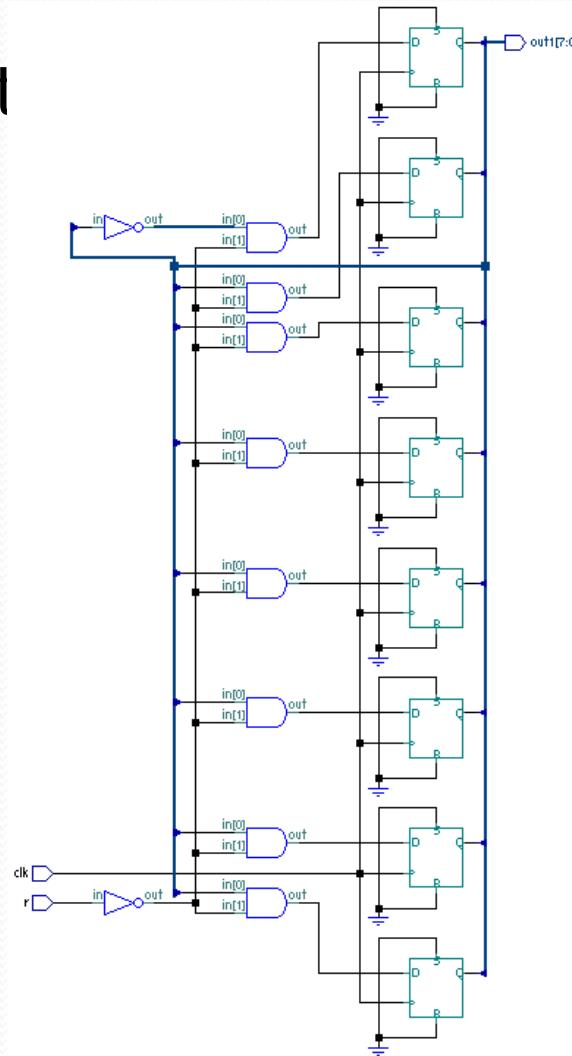
Critical.path.#1,(path.slack.=.78.1):
NAME.....GATE.....ARRIVAL.....LOAD
-----
clock.information.not.specified
delay.thru.clock.network.....0.00.(worst.case)

reg_temp(1)/Q.....dff.....0.00..0.70.up.....0.01
ix128/Y.....nand02.....17.60..18.30.dn.....4.46
ix3/DataInB.....PadInC.....3.27..21.57.up.....0.03
reg_temp(0)/D.....dff.....0.00..21.57.up.....0.00
data.arrival.time.....21.57

data.required.time..(default.specified..setup.time)...99.62
-----
data.required.time.....99.62
data.arrival.time.....21.57
-----
slack.....78.05
```

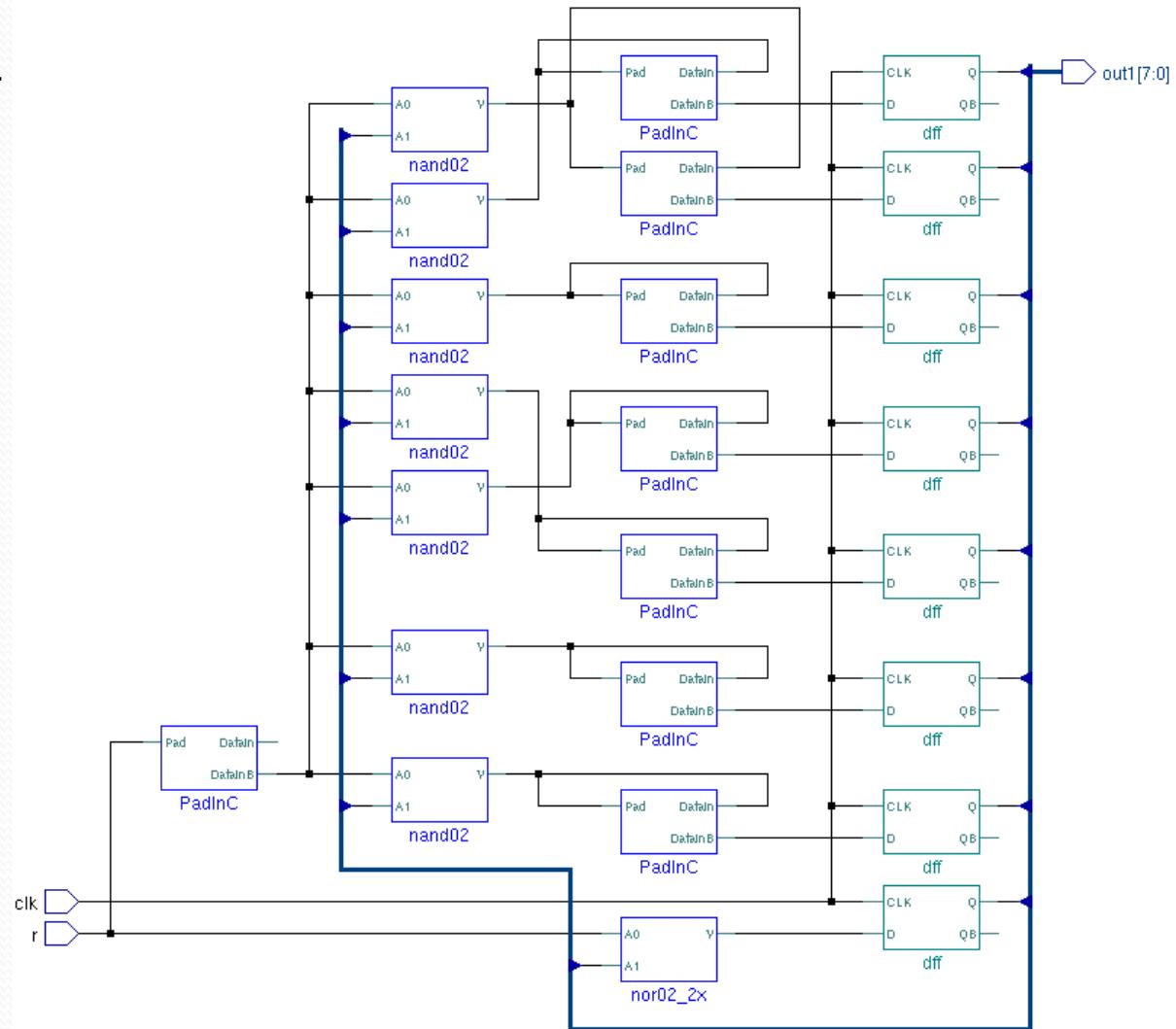
Synthesis Notes

- RTL Schematic



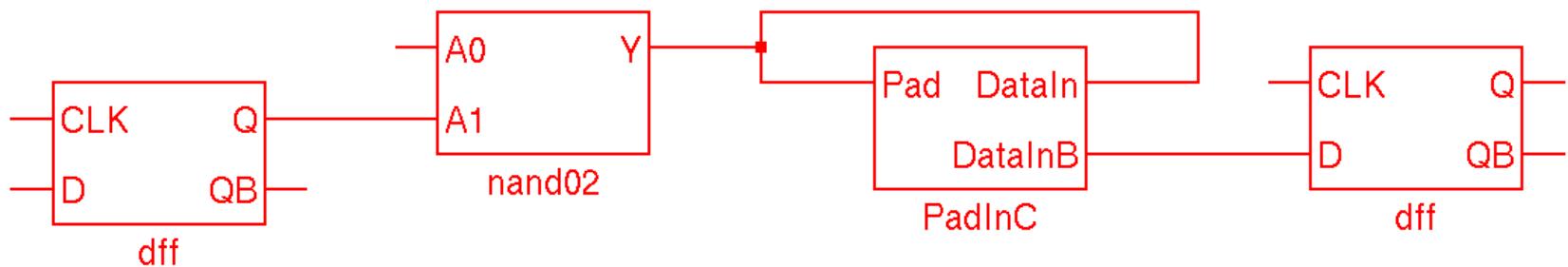
Synthesis Notes

- Technology Schematic



Synthesis Notes

- Critical Path Schematic
 - Hover with the mouse to see details



Post-Synthesis Simulation

- We want to check if the design is synthesized correctly
 - Un-optimized & badly written Designs could generate wrong functionality
 - Generate Verilog Netlist

Post-Synthesis Simulation

- Create New Project in Modelsim
- Compile Technology File in work library
- Compile the Synthesized design in work library
- Run

