### Cadence First Encounter Tutorial

Files for this tutorial can be downloaded from: www.cs.wright.edu/~emmert/tutorials/enc\_files.tar.gz

# **Configuration File**

- This file contains information used to setup your design for synthesis from RTL to the layout or gds levels of circuit abstraction
- Some (not all) of the file options:
  - Configuration file name: map.conf
  - RTL verilog input filename: map.v
    - This file can come from Cadence RTL compiler, Synopsys DC Ultra, or other HDL compiler
  - Top level design name: CHIP
    - This is your top level design name
  - gds libraries if available for gds extraction
  - Timing library files: \*.tlf files
  - IO file that defines the asic pin locations: map.io
  - Timing file generated by the synthesis tool: map.sdc
  - Library exchange format or lef library file names
  - Footprints for buffers and inverters
    - For example, {buf} would include all cells whose names started with "buf"
  - The names for the vdd and vss pins

*****	4#
#	#
# FirstEncounter Input configuration file : map.conf	#
#	#
#######################################	##
global rda_Input	
set cwd.	
set rda_Input(import_mode) {-treatUndefinedCellAsBbox 0	)}
set rda_Input(ui_netlist) "map.v"	
set rda_Input(ui_netlisttype) {Verilog}	
set rda_Input(ui_rtIlist) ""	
set rda_Input(ui_iImdir) ""	
set rda_Input(ui_ilmlist) ""	
set rda_Input(ui_iImspef) ""	
set rda_Input(ui_settop) {0}	
set rda_Input(ui_topcell) {CHIP}	
set rda_Input(ui_celllib) ""	
set rda_Input(ui_iolib) ""	
set rda_Input(ui_areaiolib) ""	
set rda_Input(ui_blklib) ""	
set rda_Input(ui_kboxlib) ""	
set rda_Input(ui_gds_file)	
set rda_Input(ui_oa_oa2lefversion) {}	
set rda_Input(ui_view_definition_file) ""	
set rda_Input(ui_timelib,max) ""	
set rda_Input(ui_timelib,min) ""	
set rda_Input(ui_timelib)	
set rda_Input(ui_smodDef) ""	
set rda_Input(ui_smodData) ""	

# Starting Encounter

- To start the tool, first you must source the environment file source set\_cadence\_soc\_env <CR>
  - This file sets up the paths and license file access to run First Encounter
- Then on the command line type

encounter <CR>

Encounter – /usr/local1/emmert/rx	_asic/tx_jul_07	7/enc_25u			
<u>D</u> esign <u>E</u> dit S <u>v</u> nthesis Partitio <u>n</u> <u>E</u> loorplan Po <u>w</u> er <u>P</u> lace <u>C</u> loc	< <u>R</u> oute <u>T</u> imir	ng <u>S</u> I <u>V</u> erif	y Too <u>l</u> s	<u>H</u> elp	cādence
	🖬 🕵 Design	is: Not in	Memory	All Colo	ors
💫 🖶 📜 💶 🖳 🖳 😹 🖴 🔩				· · · ·	v e
				Plan View	
			N	lodule ence	
			Ċ		
			C F	lbstruct Region	
			4	area Density	
			L N		
			0	special Net	
			F		
			F	lext Rel. FPlan	
			2	(ield Cell	
			_		
	Q (	-0.129,	0.047)		
					1

## Read In Your Design

- From the Design tab Design → Import Design
- From the Design Import window Load
- Type in or select the configuration file name (eg: map.conf) and select

Open

• In the Design Import window select OK

— Design Import	•
Basic Advanced	
Verilog Netlist:	
Files: map.v	J.
Top Cell: 🔶 Auto Assign 💠 By User: <u>CHIP</u>	
- Timing Libraries:	
Max Timing Libraries:	
Min Timing Libraries:	
Common Timing Libraries: vtvtlib25.tlf	
LEF Files: <u>vtvt25_base.lef vtvtlib25.lef</u>	<u></u>
Timing Constraint File: map.sdc	
IO Assignment File: <u>map.io</u>	
<u>O</u> K <u>Save Load Cancel Help</u>	1



### Floor Plan Your Design

• From the Floorplan tab

Floorplan  $\rightarrow$  Specify Floorplan

- There are many options for defining the floorplan
- Example below shows
  - Size
  - Core Utilization of 75%
  - Core space for Power Rings 100.0 from Core to IO boundary

— Specify Floc	orplan	•
Basic Advanced		
		[
Specify By: 🔶 Size 🔷 Die/IO/Core	Coordinates	
🔶 Core Size by: 🔶 Aspect Ratio:		0.9781659
	🔶 Core Utilization:	0.75
	💠 Cell Utilization:	0.751365
🔷 Dimension:		247.32
		241.92
🔷 Die Size by:		247.32
		241.92
Core Margins by: 🔶 Core to IO	Boundary	
🔷 Core to Die	Boundary	
Core to Left: 100	Core to Top:	100
Core to Right: 100	Core to Bottom:	100
Die Size Calculation Us 💠	Max IO Height 🔶 M	in IO Height
📔 Floorplan Origin at: 🔶	Lower Left Comer 🐟	Center
		Jnit: Micron
<u>О</u> К <u>А</u> рріу	<u>C</u> ancel	Help



Note: Select the layout tab button to remove the pink box on the left.

### Power Rails

- In this step we add Power Rails
- From the Power Tab

Power  $\rightarrow$  Power Planning  $\rightarrow$  Add Rings



Net(s): <u>yss vdd</u> Ring Type Conring(s) contouring Avound core boundary Exclude sel ected objects Buck fing(s) around Back fing(s) around Back find(s) around Bac

Add Rings

Basic Advanced Via Generatio



### Power Stripes

- In this step we add Power Stripes
- From the Power Tab

Power  $\rightarrow$  Power Planning  $\rightarrow$  Add Stripes

Add Stripes	
Basic Advanced Via Generation	
	-
Set Configuration	
Net(s): vss vdd	
Layer: metal2 –	
Direction: 🔶 Vertical 🔷 Horizontal	
Spacing: <u>5</u>	
- Set Pattern	
Set-to-set distance: 80	
Number of sets: <u>1</u>	
Bumps $\diamond$ Over $\diamond$ Between	
◆ Over P/G pins Pin layer:	
♦ Master name: ♦ Selected blocks	
- Strine Boundary	
◆ Core ring	
💠 Pad ring 💠 Inner 🔶 Outer	
🔷 Design boundary 📕 Create pins	
Each selected block/domain/fence	
All domains	
Specify rectangular area	
First/Last Stripe	
A Relative from one or selected area	
X from left: 20 X from right: 0	
Absolute locations	
Use option set:	
	_ ك
OK Veriebles Andre Defeution Conset	
<u>v</u> rancei <u>H</u> eip	

- Encounte	r – /usr/local1/emm	ert/rx_asic/tx_ju	l_07/enc_25u	– Top Cell: (CH	IIP)	•
Design Edit Synthesis Partit	io <u>n E</u> loorplan Po <u>w</u> e	er <u>P</u> lace <u>C</u> lock	<u>R</u> oute <u>T</u> iming	I <u>S</u> I <u>V</u> erify T	oo <u>l</u> s <u>H</u> elp	cädence
	A 2 2 A	000000	📲 Design is	s: In Memory	All Co	lors
	= <mark>-</mark>					v s
			<u></u>		FPlan View	<u>_</u> [?]
					Fence	
					Guide Obstruct	
					Region	
					Instance	
					Net Special Net	
					Terminal	
					Text	
					Rel. FPlan Yield Cell	
					Yield Map	
! !						
						T
······································						
			Q (	-40.61, 228	43 )	

# **Routing Power Stripes**

### Route vdd and vss

Route  $\rightarrow$  Special Route(SRoute)  $\rightarrow$  OK

- SRoute	
Basic Advanced Via Generation	
Net(s): vss vdd	
Route	
Stripe Layer: Width: Pitch:	
Routing Control         Layer Change Control         Top layer:       M5 m         Bottom layer::       M1 m         ♦ Straight connections and allow jogging       ♦ Straight connections only       ♦ Same layer routing only         ■ Prefer straight with layer change       ■ DRC clean         ♦ Prefer different layer jog       ■ Allow layer change	
Area       X1:     Y1:       X2:     Y2:       © Connect to target inside the area only	
☐ Delete existing routes ☐ Generate progress messages 0 ☐ Extra config file: ☐	
2K Apply Defaults Cancel Help	



### Placement

- Next we perform placement
- If required, insert well contacts first

Place  $\rightarrow$  Fill  $\rightarrow$  Add Well Tap

- Hit the Select button to view available well tap cells in the library
- Set the space between taps to the distance required for the target technology
- If well taps are included within the cells or for SOI designs, continue on to placement (next page)

### Placement

• Next, place the cells

#### Place $\rightarrow$ Standard Cells and Blocks

OK

ļ	- Place / 💷
l	$\diamond$ Run Full Placement $\diamondsuit$ Run Incremental Placement $\diamondsuit$ Run Placement in Hoorplan Mode
	Optimization Options Include Pre-Place Optimization Include In-Place Optimization
ĺ	<u>OK</u> <u>Apply</u> <u>Mode</u> <u>Defaults</u> <u>Cancel</u> <u>Help</u>



### **Clock Tree Generation**

- For sequential circuits, add clock trees
- The map.ctstch file describes the requirements for the clock tree

```
\begin{array}{l} \text{Clock} \rightarrow \text{Design Clock} \\ \text{OK} \end{array}
```

—	Synthesize Clock Tree	•
1	Basic Advanced	
	Clock Specification Files: map.ctstch Gen Spec	
	Results Directory: clock_report	
Г	OK Apply Mode Load Spec Clear Spec Cancel He	stp [
		<u> </u>



# Routing

- Now use nanoroute to route the design
  - Route  $\rightarrow$  NanoRoute  $\rightarrow$  Route

OK

- NanoRoute	• 🗆
Routing Phase Global Route Detail Route Start Iteration default End Iteration default	
Post Route Optimization 🗆 Optimize Via 🖾 Optimize Wire	
Concurrent Routing Features	
Timing Driven     Effort 2     S.M.A.R.T.	
Post Route SI SI Victim File	
Routing Control Selected Nets Only Bottom Layer default Top Layer default	
Area Route Area Select Area and Route	
Job Control       Auto Stop       Auto Stop       Multi Threading       Edit Clents	
OK Apply Albibute Mode Save Load Cancel Help	

Encounter – /usr/local1/emmert/rx_asic/tx_jul_07/enc_25u – Top Cell: (CHIP)	
Design Edit Synthesis Partition Eloorplan Power Place Clock Route Timing SI Verify Tools	∐elp cädence
🖻 및 및 및 및 🖉 😺 👘 🔊 🖉 💿 🏔 🗊 📋 😭 🖬 🕵 Design is: Placed	All Colors
	V S FPlan View 🗖 🗖 🎇
	Module
	Fence L
	Obstruct
	Area Density 🖽 🗖 🗖
	Instance
	Special Net 🛄 🗖 📕
	Terminal 🛄 🗖
	Text
	Rel. FPlan 📕 Yield Cell 🗖
	Yield Map 📕 🗖
	17
Q (86.56, 288.26)	

### Fill

### • Now you will add fill cells and metal fill

```
Place \rightarrow Filler \rightarrow Add Filler
Select to select fill cells
OK
Route \rightarrow Metal Fill \rightarrow Add
OK
```

- Add Filler	· · □
Cell Name(s) filler	Select
Prefix FILLER	
Power Domain	Select
No DRC	
Mark Fixed	
Fill Boundary	
🗆 Fill Area Draw View Area	
11× 11y	
urx ury	
OK Apply Mode Cancel	Help

Model Selection Shape: 🔶 Rectangle 😞 Square Connection: - Tie High/Low to net(s): vss vdd Connection shape: 🔶 Tree 🗠 Mesh Keep unconnected metal fill(s) □ Square shape Incremental Control □ Delete metal fill before creating new metal fill Layer Selection 🗖 M1 📮 M2 📮 M3 📮 M4 📮 M5 All Layers Timing Aware Critical nets from timing analysis Slack threshold: 0.0 💷 Area Draw 0.000 0.000 0.000 0.000 OK. Apply Defaults Cancel Help

Add Metal Fill

# Placed, Routed, and Filled



### Verification

• Verify the design and view the errors

Verify  $\rightarrow$ 

Verify Geometry Verify Metal Density Verify Connectivity Violation Browser

- Violation Browser 👘 🗖	Veri	y Geometry
Display Settings	Basic Advanced	
Description:	Varification Area	
	🔶 Entire area	
	Specify Draw	View Area
	X1: • Y1:	
	X2: 0 Y2:	0
	Minimum Width	Minimum Spacing
	🗖 Minimum Area	Same Net Spacing
	Short	Geometry Antenna
	Cell Overlap	□ Off Routing Grid
	🗖 Insufficient Metal 0	verlap 🗖 Off Manufacturing Gr
	🗖 MinHole	📕 Implant Check
	🗖 Minimum Cut	🖬 MinStep
	Via Enclosure	
	-Allow	
	🗖 Pin In Blockage	
	📕 🖬 Same Cell Violati	
	🔲 🗖 Different Cell Vi	
	🔲 🔲 Overlap of Pad Fi	
	🔲 🗖 Overlap of Routin	
🈼 😼 🗶 😳 🛆 Report File: CHIP.viols.r 🛅 Save	🔲 Överlap of Routin	
	- <u>-</u>	
Apply Close Defaults Help		

-	Verify Process Antenna 🗾 🚪
	utput Files
	ntenna LEF: CHIP.antenna.lef 🛛 📔
	eport: CHIP.antenna.rpt 🗾 🖻
	🗖 Generate detailed report
	Cui Selected
	Named:
	Report Limits
	rror: 1000 🔲 🗆 pgnet 🗖 noIOPinDefault
<u> </u>	
	<u>Ok Apply C</u> ancel <u>H</u> elp
_	

Metal		Report: <u>C</u>	HIP.densit	y.rpt	<b>D</b>	
🗖 Gen						
Lay	ers All					
<b>♦</b>	Specify					
- Ver	ification	n Area —				
<b>~</b>	Entire area					
♦ 1	Specify					
	X1: 0	Y1 :	0			
	NO 0		0			

Verify Connectivity	-
◆ All	
Regular Only	
Special Only	
_Nets	
🔶 All	
♦ Selected	
Amed:	
Check	
🗖 Open 📃 UnConnected Pin 📮 Unrouted Net	
🗖 Connectivity Loop 🗖 Antenna 🛛 🗖 Weakly Connected	Pin
Geometry Loop 🗖 Geometry Connectivity	
Connect Pad Special Ports	
Verify Connectivity Report: CHIP.conn.rpt	
-Report Limits	
Error: 1000 Warning: 50	
<u>O</u> k <u>Apply</u> <u>Cancel Help</u>	

### Save

• Save the design and write out the design exchange format (def)

```
Design \rightarrow Save Design As \rightarrow SoCE
Save
Design \rightarrow Save \rightarrow DEF
OK
```

_	Save Design
<u>D</u> .	irectory: //usr/local1/emmert/rx_asic/tx_jul_07/enc_25u 🗖 💼
	gdsinfo _appOption.dat _clock_report _figs _nanoroute_tmp_Pgaa1X
	File <u>n</u> ame: CHIP.enc Save
F	iles of <u>t</u> ype: Design files (*.enc) - Cancel





### Notes

- For instructions on tool functionality and capabilities, after sourcing the environment file, type cdnshelp <CR>
- For help while in encounter just type help command\_name <CR>
   Wild cards (\*) can also be used
- Make sure you save at regular intervals during the design process
- All commands can be run on the command line without starting the GUI by starting encounter with the nowin option encounter -nowin <CR>
  - At the command line you can source prewritten command files or enter the command