DVLSI PROJECT REPORT

BIOREALISTIC SPIKING NEURAL NETWORKS

Sai Vamshi Kandukuri ,Naveen Chander

Design Objective : TO implement a bio-realistic Izhikevich Neural Network.

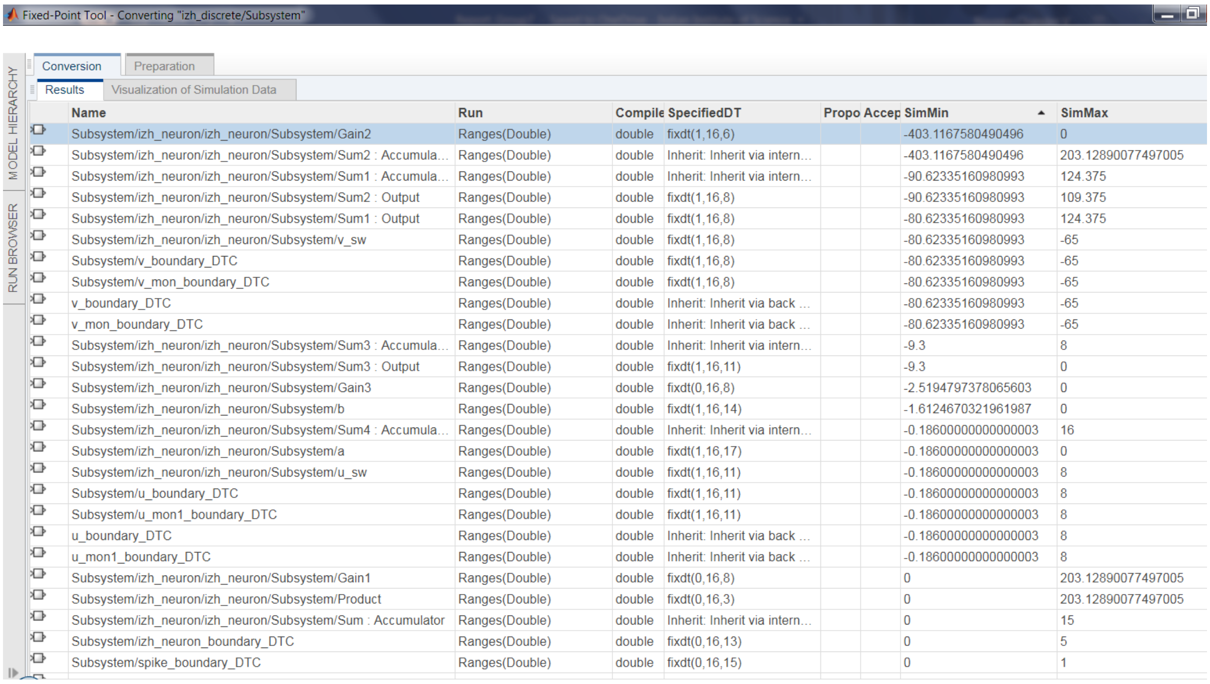
Design focusses on implementing the neural network using the least power and area by leveraging the relatively relaxed timing constraints due to bio-realistic time scales.

A linear neural network of 4 Regular Spiking Neurons has been implemented.

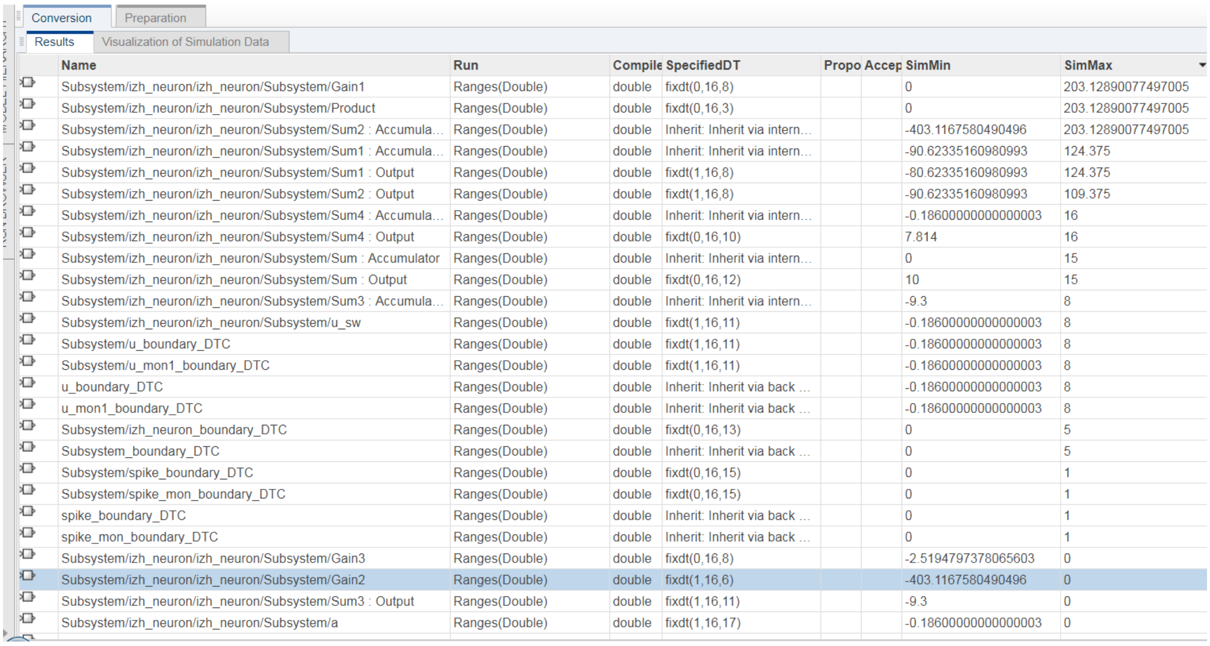
Verilog Code written is presently scalable upto 200 neurons.

After implementing the algorithm in MATLAB, the fixed point conversion values were obtained.

Fixed Point Conversion


Fixed-Point Tool - Converting "izh_discrete/Subsystem"
MODEL HIERARCHY
RUN BROWSER
Conversion
Preparation
Results
Visualization of Simulation Data

Name	Run	Compile	SpecifiedDT	Propo	Accep	SimMin	SimMax
Subsystem/izh_neuron/izh_neuron/Subsystem/Gain2	Ranges(Double)	double	fixdt(1,16,6)			-403.1167580490496	0
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum2 : Accumula...	Ranges(Double)	double	Inherit: Inherit via intern...			-403.1167580490496	203.12890077497005
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum1 : Accumula...	Ranges(Double)	double	Inherit: Inherit via intern...			-90.62335160980993	124.375
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum2 : Output	Ranges(Double)	double	fixdt(1,16,8)			-90.62335160980993	109.375
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum1 : Output	Ranges(Double)	double	fixdt(1,16,8)			-80.62335160980993	124.375
Subsystem/izh_neuron/izh_neuron/Subsystem/v_sw	Ranges(Double)	double	fixdt(1,16,8)			-80.62335160980993	-65
Subsystem/v_boundary_DTC	Ranges(Double)	double	fixdt(1,16,8)			-80.62335160980993	-65
Subsystem/v_mon_boundary_DTC	Ranges(Double)	double	fixdt(1,16,8)			-80.62335160980993	-65
v_boundary_DTC	Ranges(Double)	double	Inherit: Inherit via back ...			-80.62335160980993	-65
v_mon_boundary_DTC	Ranges(Double)	double	Inherit: Inherit via back ...			-80.62335160980993	-65
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum3 : Accumula...	Ranges(Double)	double	Inherit: Inherit via intern...			-9.3	8
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum3 : Output	Ranges(Double)	double	fixdt(1,16,11)			-9.3	0
Subsystem/izh_neuron/izh_neuron/Subsystem/Gain3	Ranges(Double)	double	fixdt(0,16,8)			-2.5194797378065603	0
Subsystem/izh_neuron/izh_neuron/Subsystem/b	Ranges(Double)	double	fixdt(1,16,14)			-1.6124670321961987	0
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum4 : Accumula...	Ranges(Double)	double	Inherit: Inherit via intern...			-0.18600000000000003	16
Subsystem/izh_neuron/izh_neuron/Subsystem/a	Ranges(Double)	double	fixdt(1,16,17)			-0.18600000000000003	0
Subsystem/izh_neuron/izh_neuron/Subsystem/u_sw	Ranges(Double)	double	fixdt(1,16,11)			-0.18600000000000003	8
Subsystem/u_boundary_DTC	Ranges(Double)	double	fixdt(1,16,11)			-0.18600000000000003	8
Subsystem/u_mon1_boundary_DTC	Ranges(Double)	double	fixdt(1,16,11)			-0.18600000000000003	8
u_boundary_DTC	Ranges(Double)	double	Inherit: Inherit via back ...			-0.18600000000000003	8
u_mon1_boundary_DTC	Ranges(Double)	double	Inherit: Inherit via back ...			-0.18600000000000003	8
Subsystem/izh_neuron/izh_neuron/Subsystem/Gain1	Ranges(Double)	double	fixdt(0,16,8)			0	203.12890077497005
Subsystem/izh_neuron/izh_neuron/Subsystem/Product	Ranges(Double)	double	fixdt(0,16,3)			0	203.12890077497005
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum : Accumulator	Ranges(Double)	double	Inherit: Inherit via intern...			0	15
Subsystem/izh_neuron_boundary_DTC	Ranges(Double)	double	fixdt(0,16,13)			0	5
Subsystem/spike_boundary_DTC	Ranges(Double)	double	fixdt(0,16,15)			0	1



Name	Run	Compile	SpecifiedDT	Propo	Accep	SimMin	SimMax
Subsystem/izh_neuron/izh_neuron/Subsystem/Gain1	Ranges(Double)	double	fixdt(0,16,8)			0	203.12890077497005
Subsystem/izh_neuron/izh_neuron/Subsystem/Product	Ranges(Double)	double	fixdt(0,16,3)			0	203.12890077497005
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum2 : Accumula…	Ranges(Double)	double	Inherit: Inherit via intern…			-403.1167580490496	203.12890077497005
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum1 : Accumula…	Ranges(Double)	double	Inherit: Inherit via intern…			-90.62335160980993	124.375
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum1 : Output	Ranges(Double)	double	fixdt(1,16,8)			-80.62335160980993	124.375
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum2 : Output	Ranges(Double)	double	fixdt(1,16,8)			-90.62335160980993	109.375
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum4 : Accumula…	Ranges(Double)	double	Inherit: Inherit via intern…			-0.18600000000000003	16
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum4 : Output	Ranges(Double)	double	fixdt(0,16,10)			7.814	16
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum : Accumulator	Ranges(Double)	double	Inherit: Inherit via intern…			0	15
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum : Output	Ranges(Double)	double	fixdt(0,16,12)			10	15
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum3 : Accumula…	Ranges(Double)	double	Inherit: Inherit via intern…			-9.3	8
Subsystem/izh_neuron/izh_neuron/Subsystem/u_sw	Ranges(Double)	double	fixdt(1,16,11)			-0.18600000000000003	8
Subsystem/u_boundary_DTC	Ranges(Double)	double	fixdt(1,16,11)			-0.18600000000000003	8
Subsystem/u_mon1_boundary_DTC	Ranges(Double)	double	fixdt(1,16,11)			-0.18600000000000003	8
u_boundary_DTC	Ranges(Double)	double	Inherit: Inherit via back …			-0.18600000000000003	8
u_mon1_boundary_DTC	Ranges(Double)	double	Inherit: Inherit via back …			-0.18600000000000003	8
Subsystem/izh_neuron_boundary_DTC	Ranges(Double)	double	fixdt(0,16,13)			0	5
Subsystem_boundary_DTC	Ranges(Double)	double	Inherit: Inherit via back …			0	5
Subsystem/spike_boundary_DTC	Ranges(Double)	double	fixdt(0,16,15)			0	1
Subsystem/spike_mon_boundary_DTC	Ranges(Double)	double	fixdt(0,16,15)			0	1
spike_boundary_DTC	Ranges(Double)	double	Inherit: Inherit via back …			0	1
spike_mon_boundary_DTC	Ranges(Double)	double	Inherit: Inherit via back …			0	1
Subsystem/izh_neuron/izh_neuron/Subsystem/Gain3	Ranges(Double)	double	fixdt(0,16,8)			-2.5194797378065603	0
Subsystem/izh_neuron/izh_neuron/Subsystem/Gain2	Ranges(Double)	double	fixdt(1,16,6)			-403.1167580490496	0
Subsystem/izh_neuron/izh_neuron/Subsystem/Sum3 : Output	Ranges(Double)	double	fixdt(1,16,11)			-9.3	0
Subsystem/izh_neuron/izh_neuron/Subsystem/a	Ranges(Double)	double	fixdt(1,16,17)			-0.18600000000000003	0

MATLAB proposed (16,6) .However, for safety margins, (18,8) was considered.

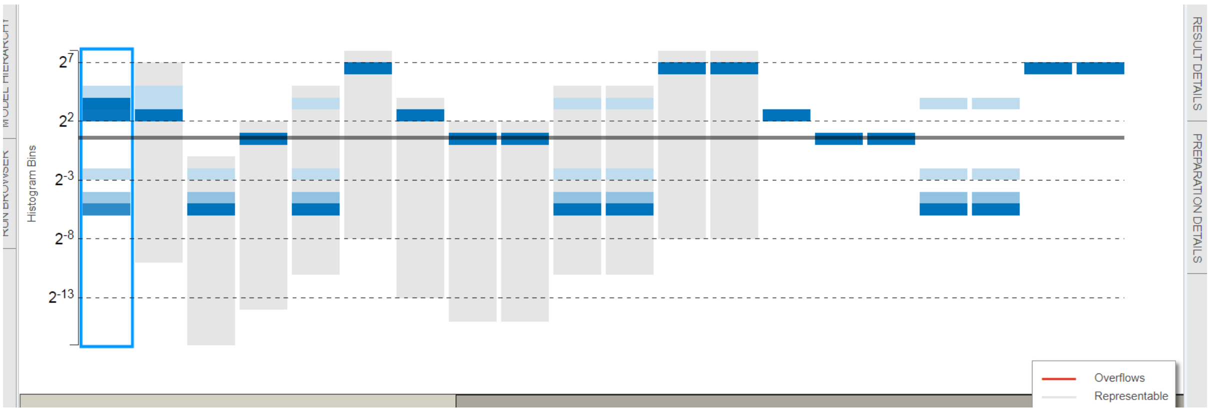


Figure 1Histogram for OUtput magnitudes

ASIC DESIGN FLOW

A)RTL Code Development:

The first stage of ASIC design flow is RTL coding. This is where we

Specify the behaviour of our design in a hardware description language.

We have divided our logic into 4 main modules:

1.Top

2.uv_main

3.uv_fsm

4.current_compu_fsm.

And also a multiplier module using booth's algorithm.

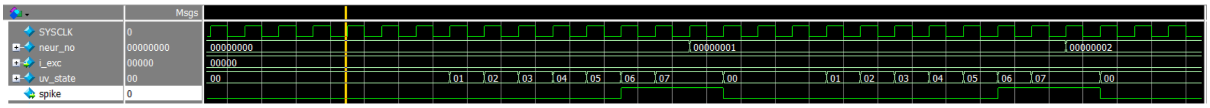
Module Descriptions

1. UV_FSM : Solves the Izhikevich Neuron Equation with inputs as present u,v state and i_exc. It is modelled using FSM
2. Curr_compu_fsm :Computes synaptic current. Modelled using FSM.
3. UV_MAIN_FSM : schedules the neuron aned synaptic current computations. Modelled as an FSM. Computes all 4 neuron states and synaptic current using the resource of that of a single neuron. Achieved by time multiplexing of coimputations.
4. Top: Integrates the above modules. Top Level FSM uses 4 states : UV_COMPUTE, CURR_COMPTE, WAIT_STATE, START_OVER.

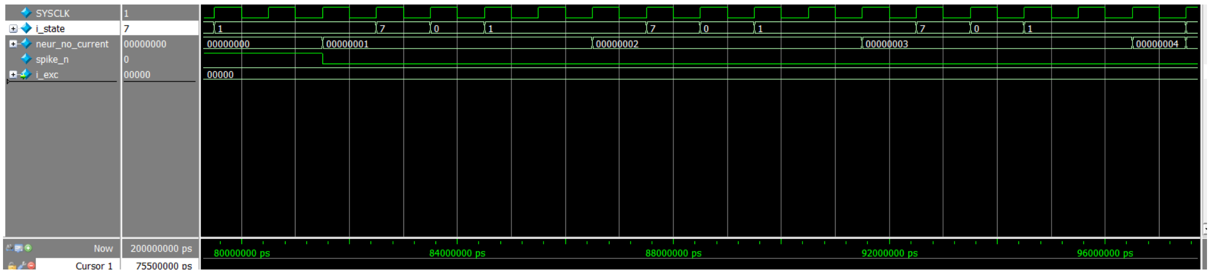
After the RTL development simulation is carried out and results verified.

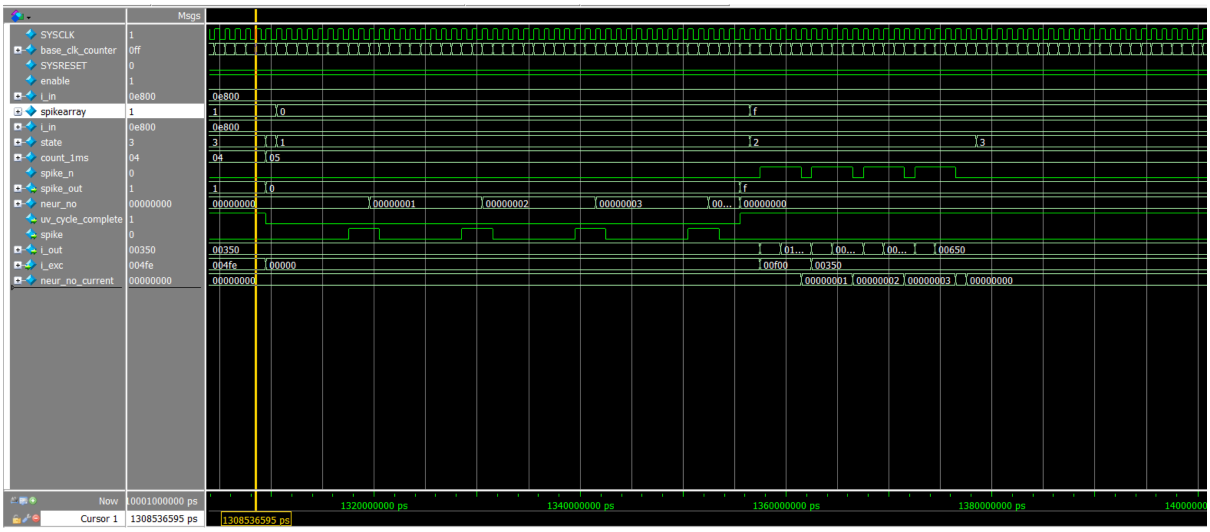
/////////Simulation Results

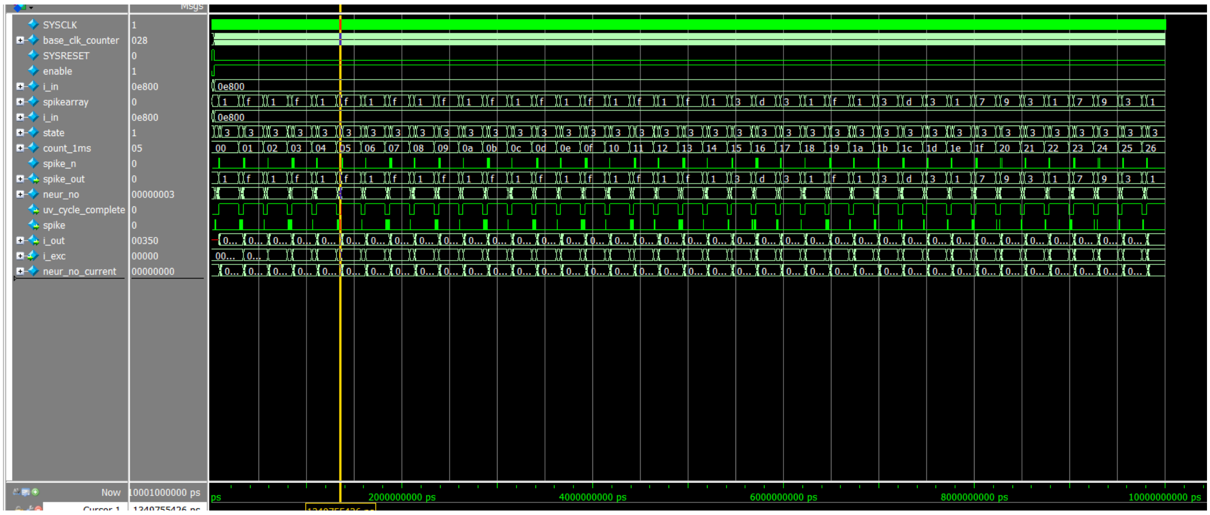
1. UV FSM



2. CUREENT FSM







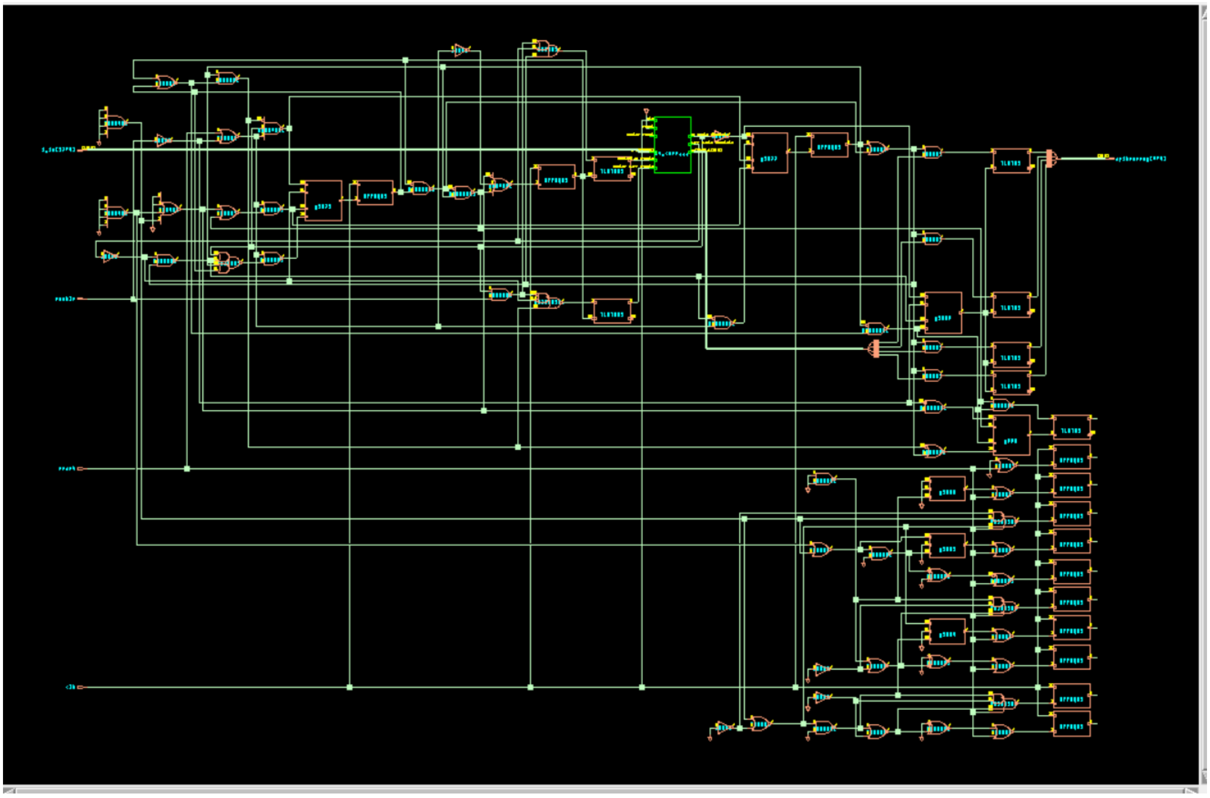
Once the RTL is developed the next step is synthesis.

B) Synthesis:

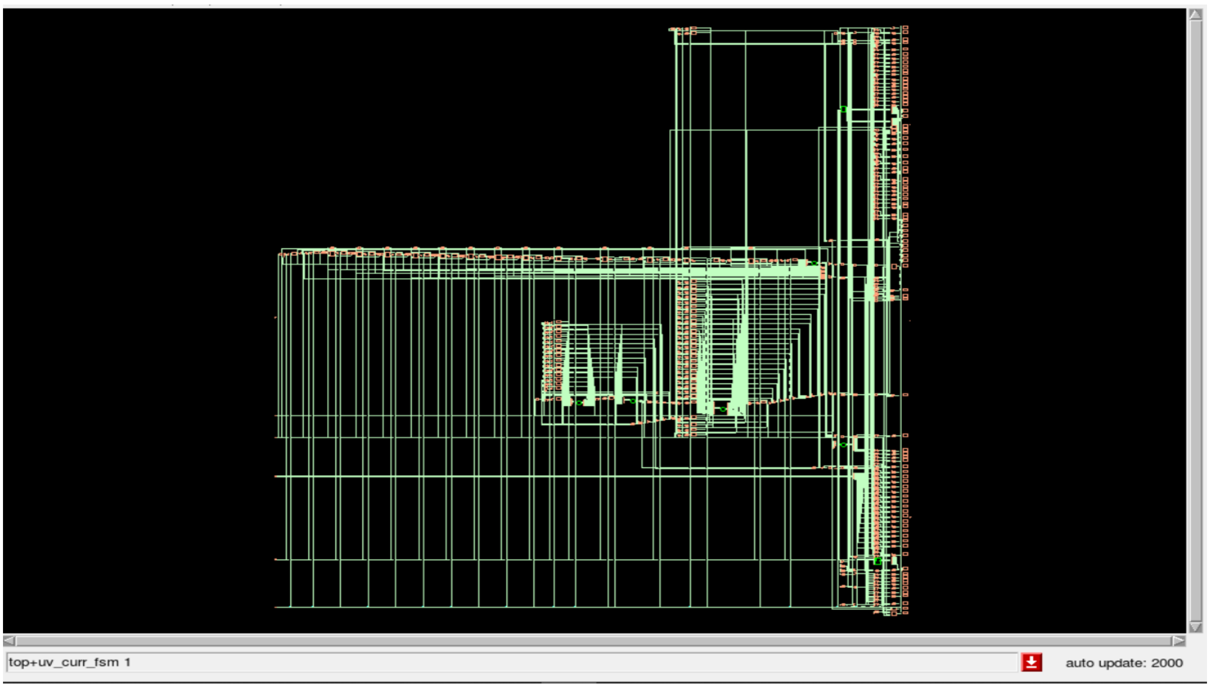
In this step the RTL code is elaborated and translated into standard cell library. After the translation we need to make a rough estimate of timing ,area and power consumption. In this step we need to mention constraints like the clock frequency at which the circuit operates and also expected clock jitter,input delays,output delays and library files.

The synthesised design of our logic is as shown in the following figures

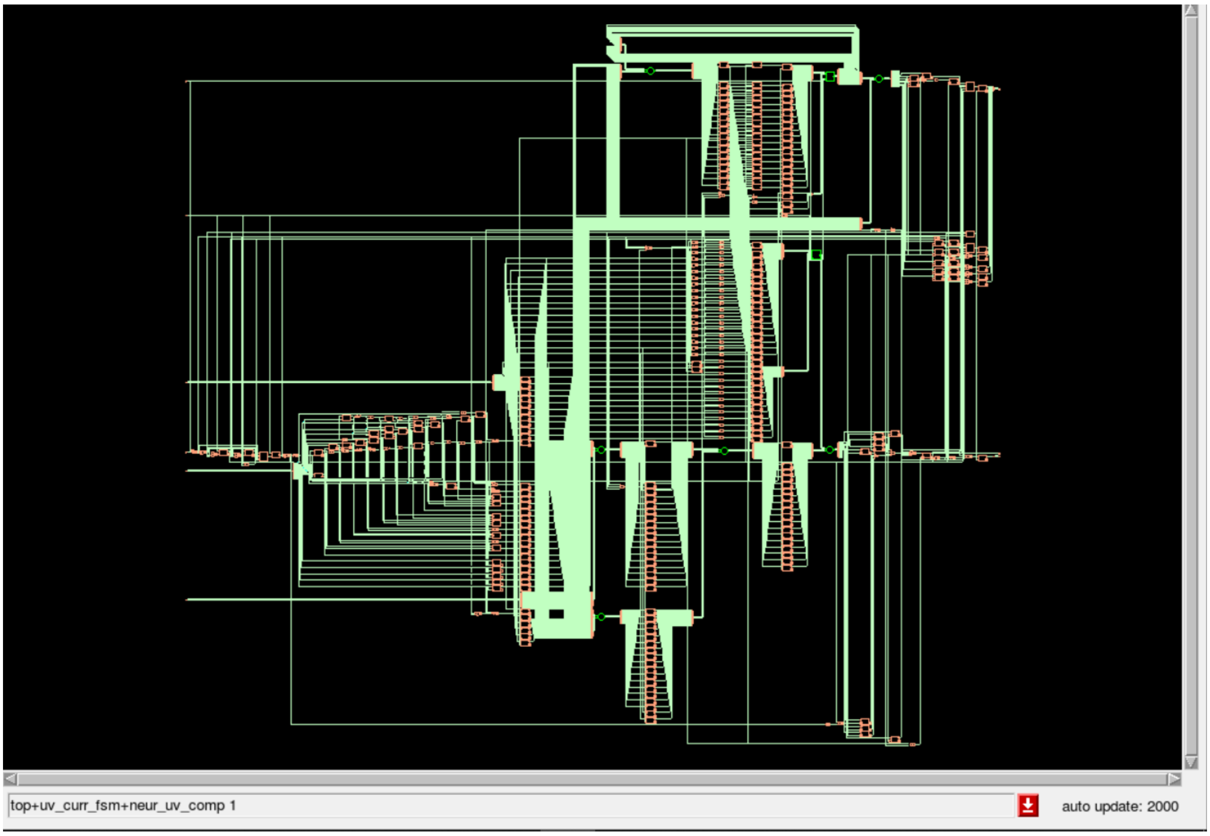
The top module is synthesised as follows:



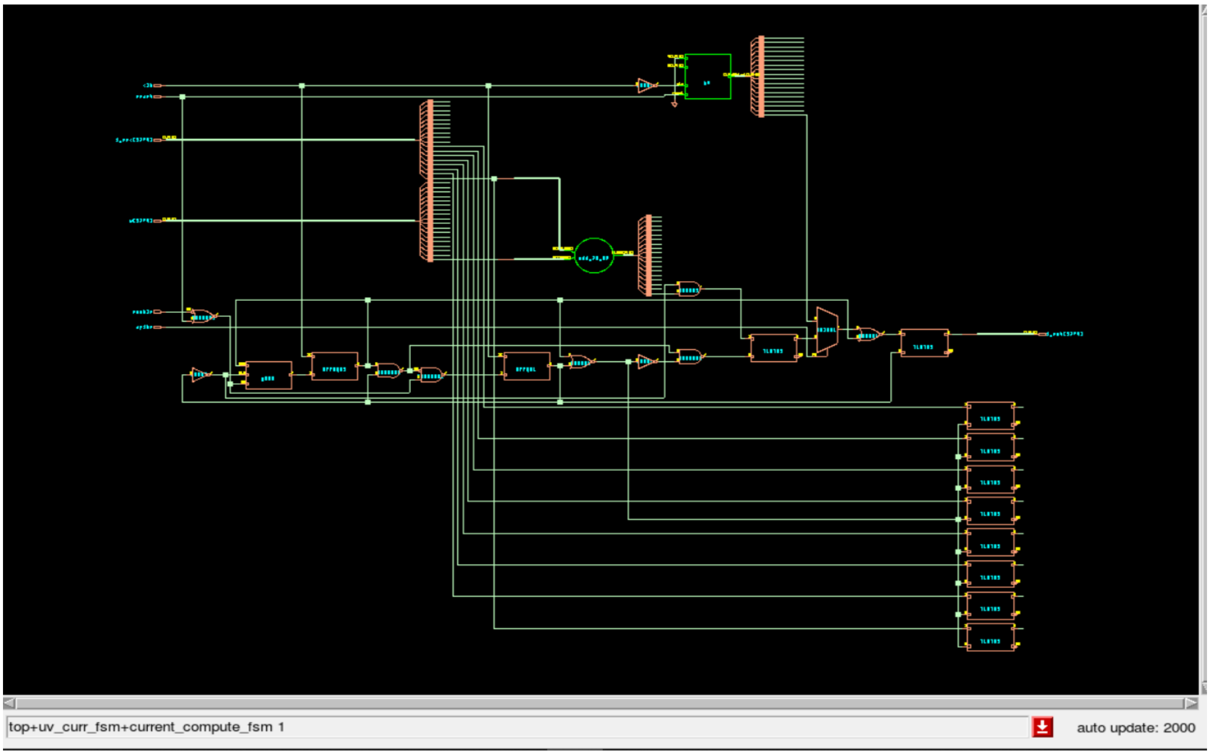
This module contains uv_curr_fsm module which is synthesised as follows:



The uv_curr fsm instantiates uv_fsm and current_compu_fsm.The uv fsm is synthesised as follows:



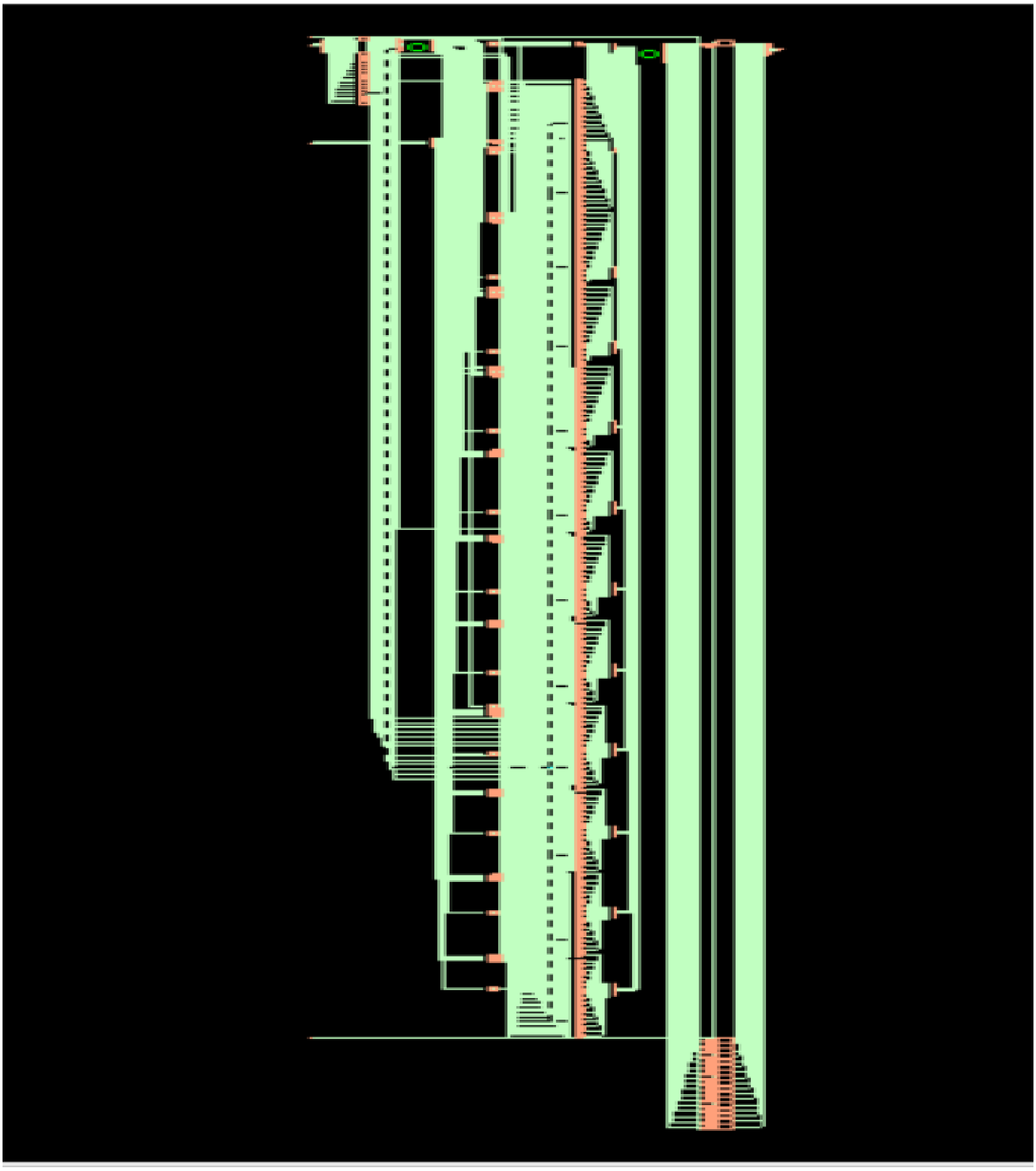
The current_compu_fsm is synthesised as follows:



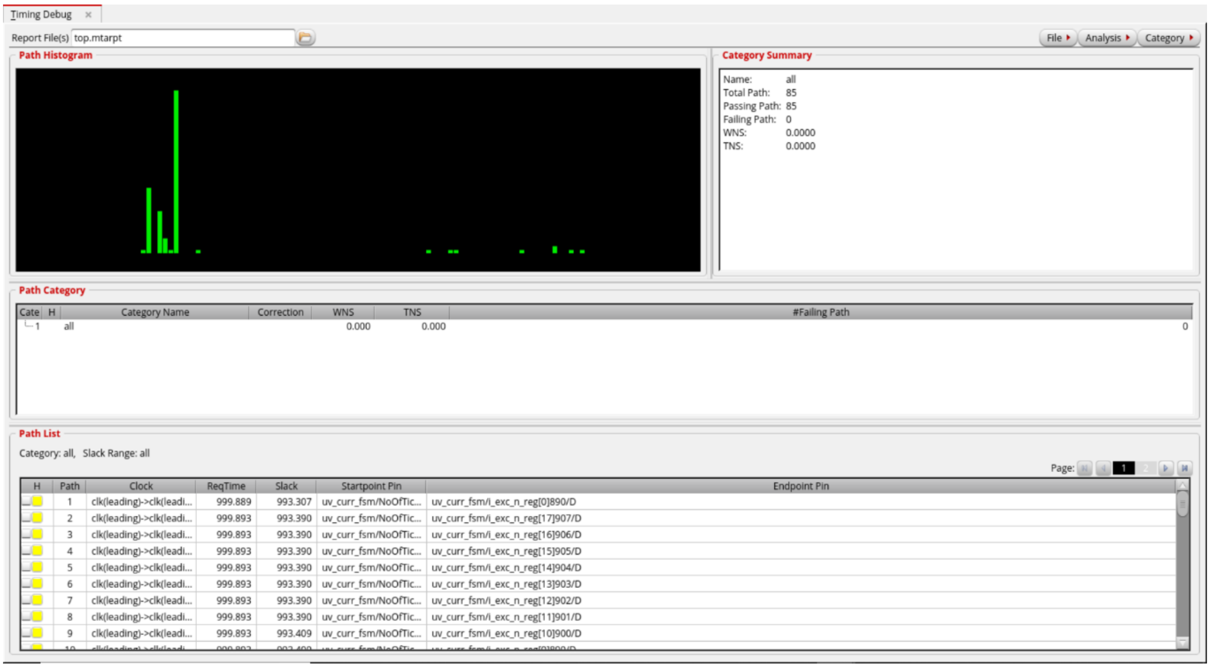
top+uv_curr_fsm+current_compute_fsm 1

auto update: 2000

The booth multiplier is synthesised as follows:



After the synthesis the first level analysis of timing and power were carried out. The results are as follows:



Report File(s) top.mtarpt

Category Summary

Name: all
Total Path: 85
Passing Path: 85
Failing Path: 0
WNS: 0.0000
TNS: 0.0000

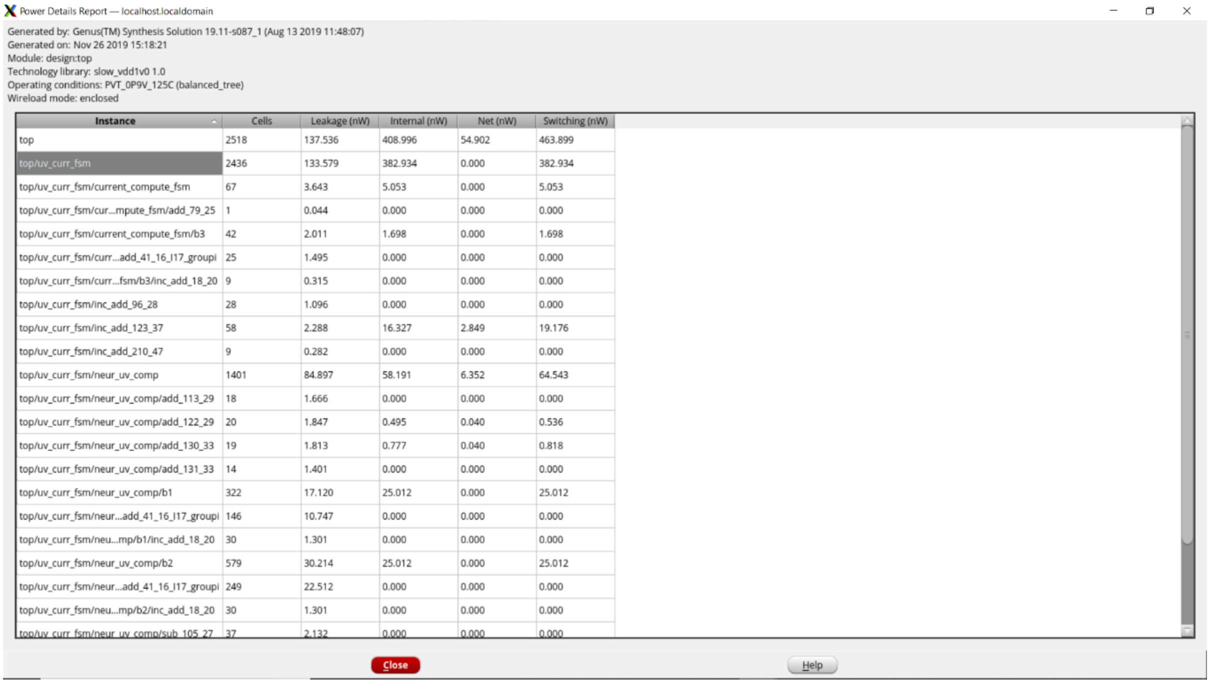
Path Category

Cate	H	Category Name	Correction	WNS	TNS	#Failing Path
1		all		0.000	0.000	0

Path List

Category: all, Slack Range: all

H	Path	Clock	ReqTime	Slack	Startpoint Pin	Endpoint Pin
🟨	1	clk(leading)->clk(leadi...	999.889	993.307	uv_curr_fsm/NoOfTic...	uv_curr_fsm/i_exc_n_reg[0]890/D
🟨	2	clk(leading)->clk(leadi...	999.893	993.390	uv_curr_fsm/NoOfTic...	uv_curr_fsm/i_exc_n_reg[17]907/D
🟨	3	clk(leading)->clk(leadi...	999.893	993.390	uv_curr_fsm/NoOfTic...	uv_curr_fsm/i_exc_n_reg[16]906/D
🟨	4	clk(leading)->clk(leadi...	999.893	993.390	uv_curr_fsm/NoOfTic...	uv_curr_fsm/i_exc_n_reg[15]905/D
🟨	5	clk(leading)->clk(leadi...	999.893	993.390	uv_curr_fsm/NoOfTic...	uv_curr_fsm/i_exc_n_reg[14]904/D
🟨	6	clk(leading)->clk(leadi...	999.893	993.390	uv_curr_fsm/NoOfTic...	uv_curr_fsm/i_exc_n_reg[13]903/D
🟨	7	clk(leading)->clk(leadi...	999.893	993.390	uv_curr_fsm/NoOfTic...	uv_curr_fsm/i_exc_n_reg[12]902/D
🟨	8	clk(leading)->clk(leadi...	999.893	993.390	uv_curr_fsm/NoOfTic...	uv_curr_fsm/i_exc_n_reg[11]901/D
🟨	9	clk(leading)->clk(leadi...	999.893	993.409	uv_curr_fsm/NoOfTic...	uv_curr_fsm/i_exc_n_reg[10]900/D



Generated by: Genus(TM) Synthesis Solution 19.11-s087_1 (Aug 13 2019 11:48:07)
Generated on: Nov 26 2019 15:18:21
Module: design:top
Technology library: slow_vdd1v0 1.0
Operating conditions: PVT_0P9V_125C (balanced_tree)
Wireload mode: enclosed

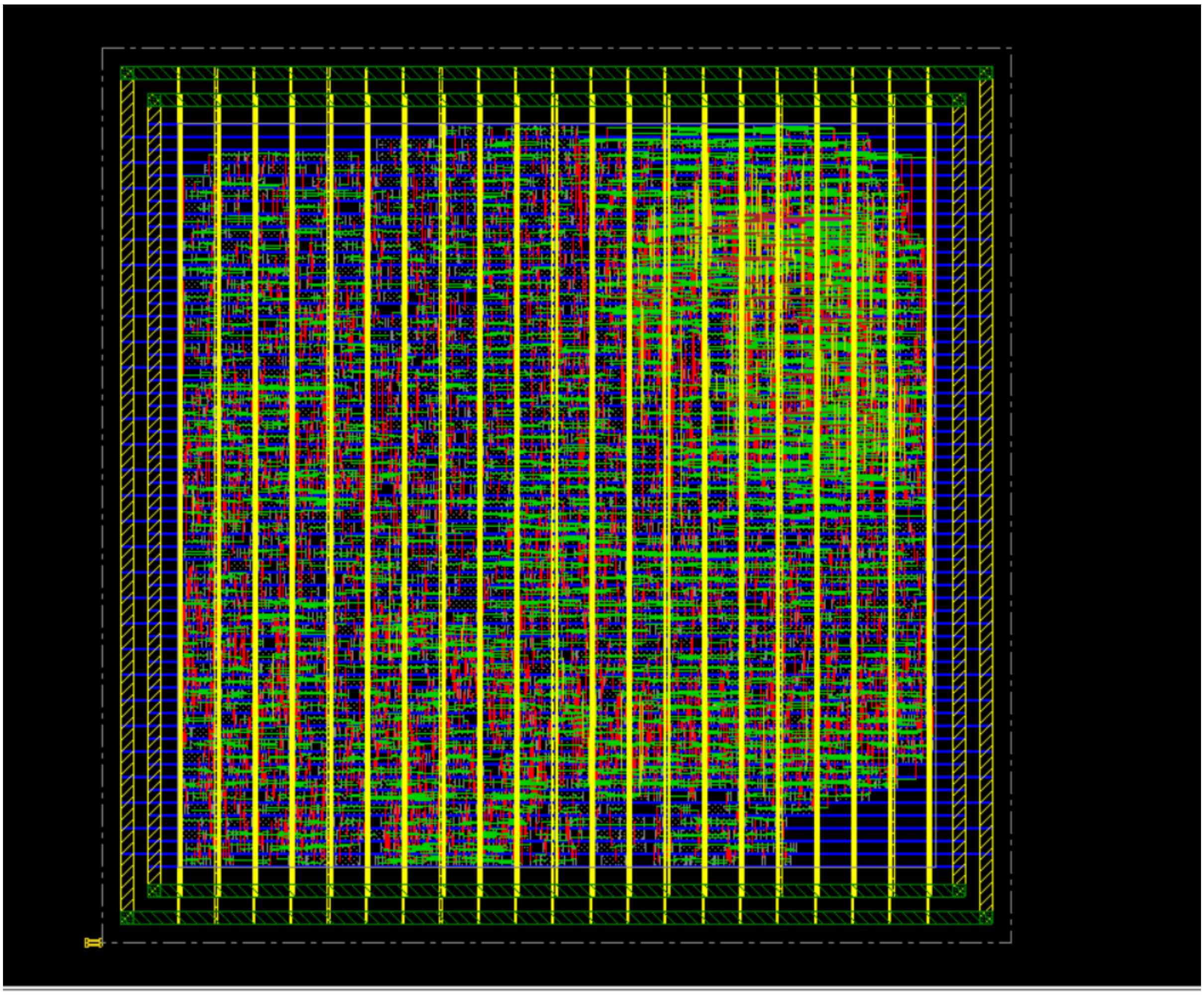
Instance	Cells	Leakage (nW)	Internal (nW)	Net (nW)	Switching (nW)
top	2518	137.536	408.996	54.902	463.899
top/uv_curr_fsm	2436	133.579	382.934	0.000	382.934
top/uv_curr_fsm/current_compute_fsm	67	3.643	5.053	0.000	5.053
top/uv_curr_fsm/cur...mpute_fsm/add_79_25	1	0.044	0.000	0.000	0.000
top/uv_curr_fsm/current_compute_fsm/b3	42	2.011	1.698	0.000	1.698
top/uv_curr_fsm/curr...add_41_16_I17_groupi	25	1.495	0.000	0.000	0.000
top/uv_curr_fsm/curr...fsm/b3/inc_add_18_20	9	0.315	0.000	0.000	0.000
top/uv_curr_fsm/inc_add_96_28	28	1.096	0.000	0.000	0.000
top/uv_curr_fsm/inc_add_123_37	58	2.288	16.327	2.849	19.176
top/uv_curr_fsm/inc_add_210_47	9	0.282	0.000	0.000	0.000
top/uv_curr_fsm/neur_uv_comp	1401	84.897	58.191	6.352	64.543
top/uv_curr_fsm/neur_uv_comp/add_113_29	18	1.666	0.000	0.000	0.000
top/uv_curr_fsm/neur_uv_comp/add_122_29	20	1.847	0.495	0.040	0.536
top/uv_curr_fsm/neur_uv_comp/add_130_33	19	1.813	0.777	0.040	0.818
top/uv_curr_fsm/neur_uv_comp/add_131_33	14	1.401	0.000	0.000	0.000
top/uv_curr_fsm/neur_uv_comp/b1	322	17.120	25.012	0.000	25.012
top/uv_curr_fsm/neur...add_41_16_I17_groupi	146	10.747	0.000	0.000	0.000
top/uv_curr_fsm/neu...mp/b1/inc_add_18_20	30	1.301	0.000	0.000	0.000
top/uv_curr_fsm/neur_uv_comp/b2	579	30.214	25.012	0.000	25.012
top/uv_curr_fsm/neur...add_41_16_I17_groupi	249	22.512	0.000	0.000	0.000
top/uv_curr_fsm/neu...mp/b2/inc_add_18_20	30	1.301	0.000	0.000	0.000
top/uv_curr_fsm/neur_uv_comp/sub_105_27	37	2.132	0.000	0.000	0.000

There was no negative slack observed in this step and also the total power consumption was reported as around 1µW.

c)Floor planning and placement:

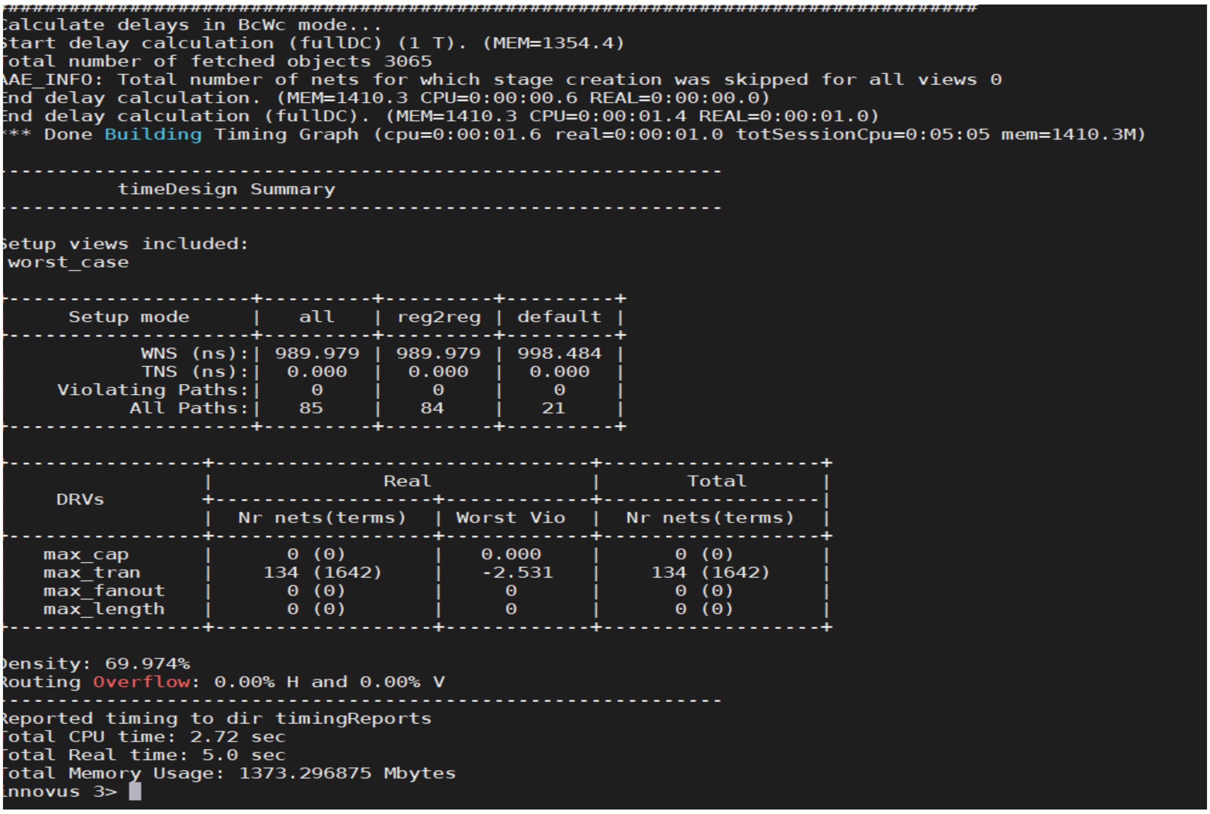
The next step is floor planning where power rails and standard cells are placed.

The design after floor planning and placement of standard cells is as follows:



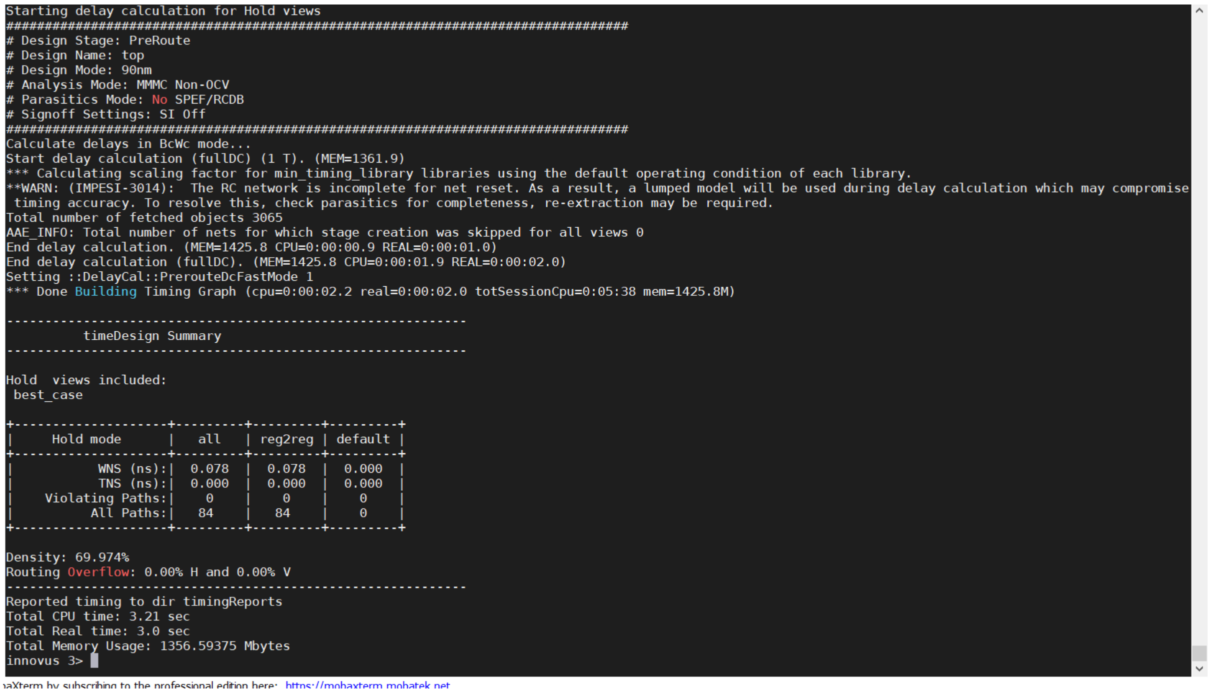
Once the floor planning and placement is done we should run pre CTS timing analysis to check for any setup and violations. The results of pre CTS timing analysis are as follows

The pre CTS setup timing analysis is as follows:



```
################################################################################
alculate delays in BcWc mode...
tart delay calculation (fullDC) (1 T). (MEM=1354.4)
otal number of fetched objects 3065
AE_INFO: Total number of nets for which stage creation was skipped for all views 0
nd delay calculation. (MEM=1410.3 CPU=0:00:00.6 REAL=0:00:00.0)
nd delay calculation (fullDC). (MEM=1410.3 CPU=0:00:01.4 REAL=0:00:01.0)
** Done Building Timing Graph (cpu=0:00:01.6 real=0:00:01.0 totSessionCpu=0:05:05 mem=1410.3M)

------------------------------------------------------------
         timeDesign Summary
------------------------------------------------------------

etup views included:
worst_case

-------------------+---------+---------+---------+
     Setup mode    |   all   | reg2reg | default |
-------------------+---------+---------+---------+
           WNS (ns):| 989.979 | 989.979 | 998.484 |
           TNS (ns):|  0.000  |  0.000  |  0.000  |
    Violating Paths:|    0    |    0    |    0    |
          All Paths:|   85    |   84    |   21    |
-------------------+---------+---------+---------+

----------------+-------------------------------+------------------+
                |             Real              |      Total       |
    DRVs        +------------------+------------+------------------|
                |  Nr nets(terms)  | Worst Vio  |  Nr nets(terms)  |
----------------+------------------+------------+------------------+
   max_cap      |     0 (0)        |   0.000    |     0 (0)        |
   max_tran     |   134 (1642)     |  -2.531    |   134 (1642)     |
   max_fanout   |     0 (0)        |     0      |     0 (0)        |
   max_length   |     0 (0)        |     0      |     0 (0)        |
----------------+------------------+------------+------------------+

ensity: 69.974%
outing Overflow: 0.00% H and 0.00% V
------------------------------------------------------------
eported timing to dir timingReports
otal CPU time: 2.72 sec
otal Real time: 5.0 sec
otal Memory Usage: 1373.296875 Mbytes
nnovus 3> █
```

The pre CTS hold timing analysis is as follows:



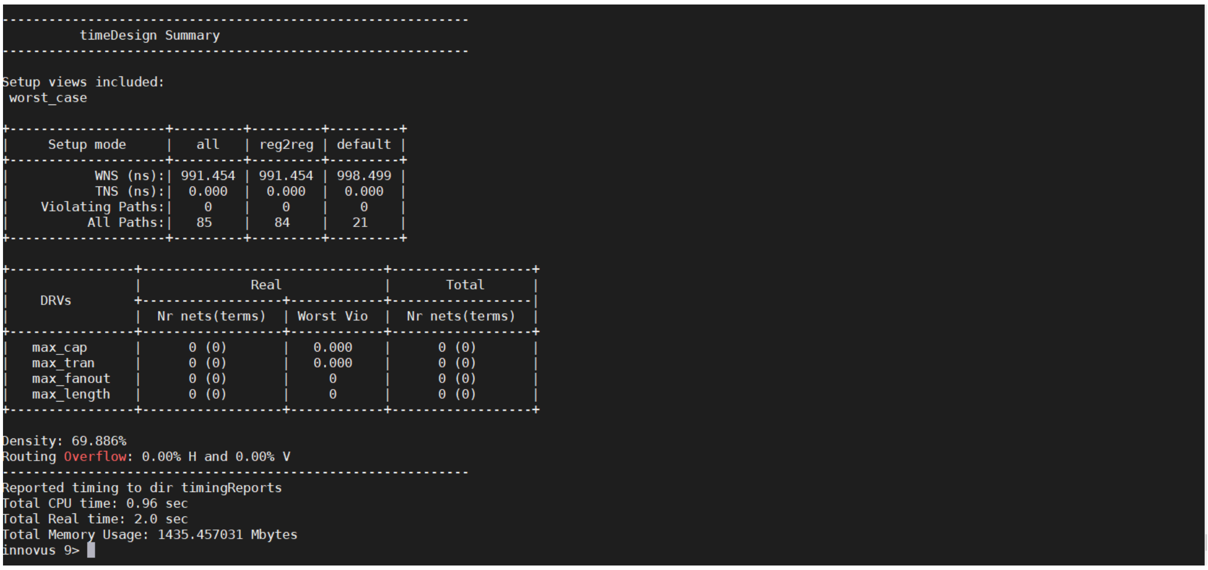
```
Starting delay calculation for Hold views
################################################################################
# Design Stage: PreRoute
# Design Name: top
# Design Mode: 90nm
# Analysis Mode: MMMC Non-OCV
# Parasitics Mode: No SPEF/RCDB
# Signoff Settings: SI Off
################################################################################
Calculate delays in BcWc mode...
Start delay calculation (fullDC) (1 T). (MEM=1361.9)
*** Calculating scaling factor for min_timing_library libraries using the default operating condition of each library.
**WARN: (IMPESI-3014):  The RC network is incomplete for net reset. As a result, a lumped model will be used during delay calculation which may compromise
 timing accuracy. To resolve this, check parasitics for completeness, re-extraction may be required.
Total number of fetched objects 3065
AAE_INFO: Total number of nets for which stage creation was skipped for all views 0
End delay calculation. (MEM=1425.8 CPU=0:00:00.9 REAL=0:00:01.0)
End delay calculation (fullDC). (MEM=1425.8 CPU=0:00:01.9 REAL=0:00:02.0)
Setting ::DelayCal::PrerouteDcFastMode 1
*** Done Building Timing Graph (cpu=0:00:02.2 real=0:00:02.0 totSessionCpu=0:05:38 mem=1425.8M)

------------------------------------------------------------
          timeDesign Summary
------------------------------------------------------------

Hold  views included:
 best_case

+--------------------+---------+---------+---------+
|     Hold mode      |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  0.078  |  0.078  |  0.000  |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |
|    Violating Paths:|    0    |    0    |    0    |
|          All Paths:|   84    |   84    |    0    |
+--------------------+---------+---------+---------+

Density: 69.974%
Routing Overflow: 0.00% H and 0.00% V
------------------------------------------------------------
Reported timing to dir timingReports
Total CPU time: 3.21 sec
Total Real time: 3.0 sec
Total Memory Usage: 1356.59375 Mbytes
innovus 3> █
```

aXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

The next step is clock tree synthesis. This is done by inserting buffers in clock path in order to reduce the clock skew and latency.

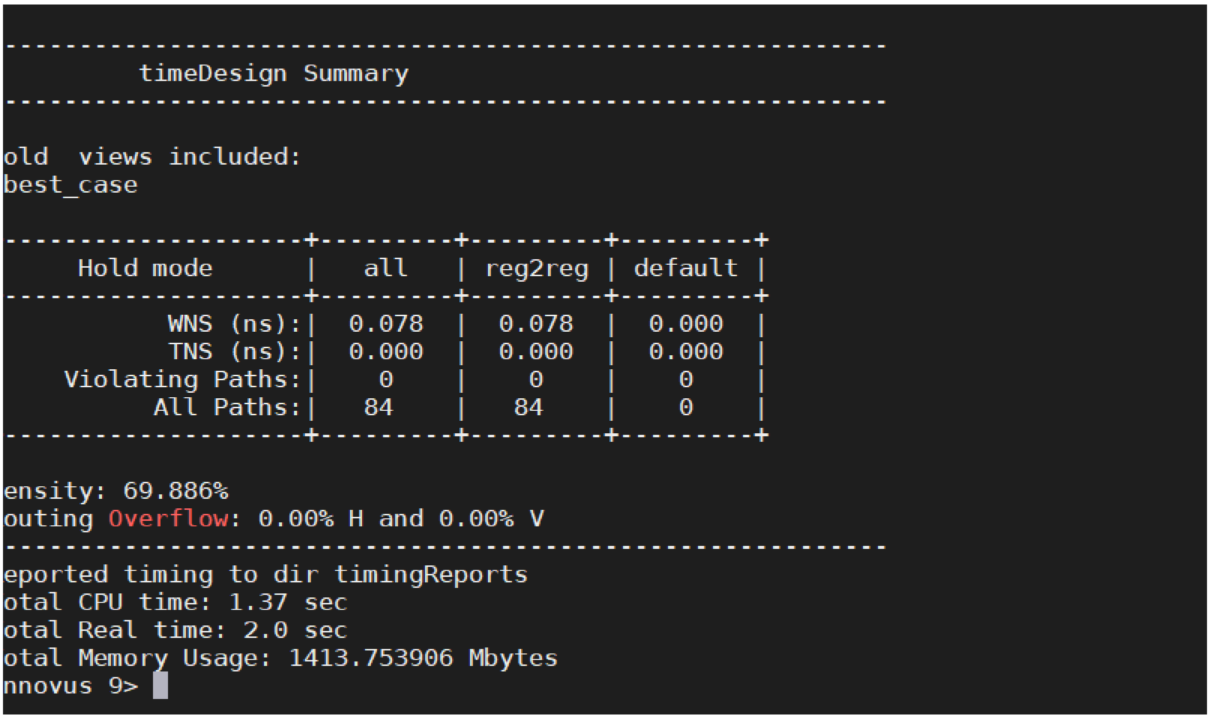
Once the clock tree synthesis is done, we need to verify if there are any setup and hold time violation. Till this step the clock was assumed to reach each flop at the same time. Now that the clock tree synthesis is done we need to check for setup and hold violations with the clock network setup.

The post CTS setup analysis report is as follows:



```
------------------------------------------------------------
         timeDesign Summary
------------------------------------------------------------

Setup views included:
 worst_case

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):| 991.454 | 991.454 | 998.499 |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |
|    Violating Paths:|    0    |    0    |    0    |
|          All Paths:|   85    |   84    |   21    |
+--------------------+---------+---------+---------+

+----------------+-------------------------------+------------------+
|                |              Real             |       Total      |
|    DRVs        +------------------+------------+------------------|
|                |  Nr nets(terms)  | Worst Vio  |  Nr nets(terms)  |
+----------------+------------------+------------+------------------+
|   max_cap      |      0 (0)       |   0.000    |      0 (0)       |
|   max_tran     |      0 (0)       |   0.000    |      0 (0)       |
|   max_fanout   |      0 (0)       |     0      |      0 (0)       |
|   max_length   |      0 (0)       |     0      |      0 (0)       |
+----------------+------------------+------------+------------------+

Density: 69.886%
Routing Overflow: 0.00% H and 0.00% V
------------------------------------------------------------
Reported timing to dir timingReports
Total CPU time: 0.96 sec
Total Real time: 2.0 sec
Total Memory Usage: 1435.457031 Mbytes
innovus 9>
```

The post CTS hold analysis is as follows:

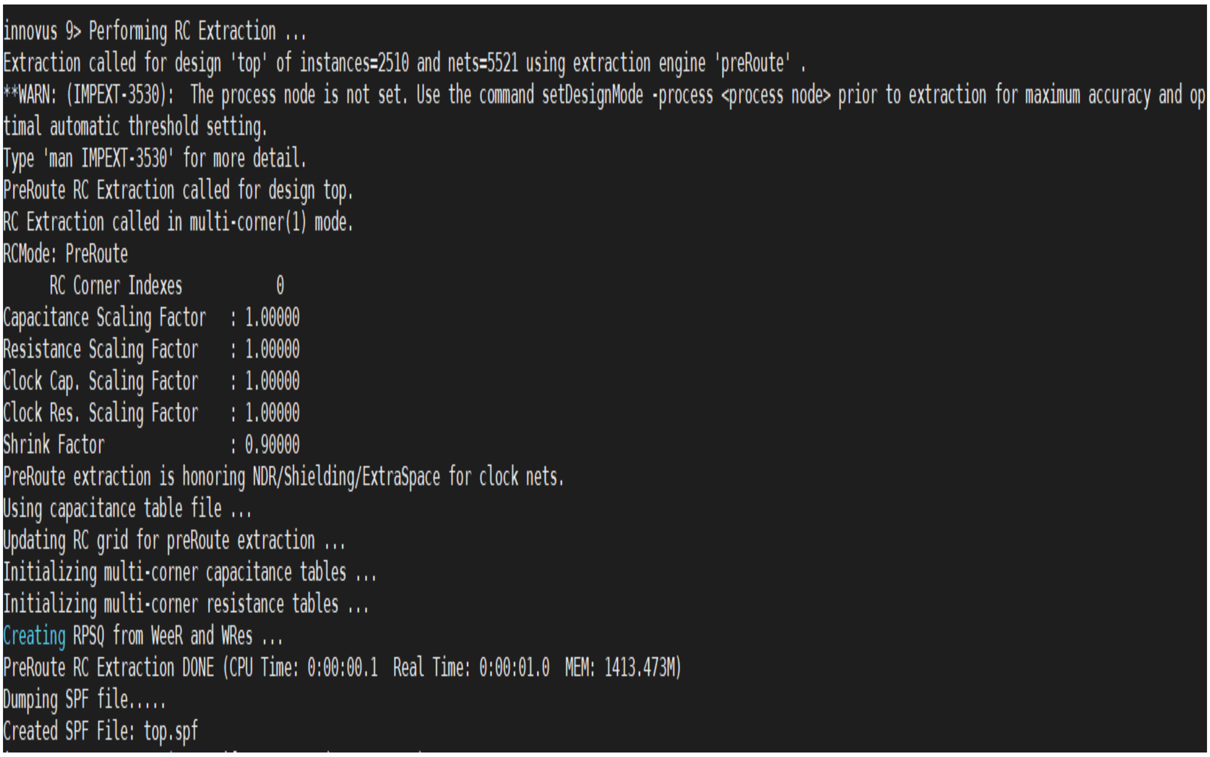


```
------------------------------------------------------------
         timeDesign Summary
------------------------------------------------------------

old  views included:
best_case

--------------------+---------+---------+---------+
     Hold mode      |   all   | reg2reg | default |
--------------------+---------+---------+---------+
           WNS (ns):|  0.078  |  0.078  |  0.000  |
           TNS (ns):|  0.000  |  0.000  |  0.000  |
    Violating Paths:|    0    |    0    |    0    |
          All Paths:|   84    |   84    |    0    |
--------------------+---------+---------+---------+

ensity: 69.886%
outing Overflow: 0.00% H and 0.00% V
------------------------------------------------------------
eported timing to dir timingReports
otal CPU time: 1.37 sec
otal Real time: 2.0 sec
otal Memory Usage: 1413.753906 Mbytes
nnovus 9>
```

All the cases reported have either positive or zero Negative slack, which implies that there are no setup and hold violations after clock tree synthesis.

The next step is routing where the interconnections between different modules are made and optimised. Once the routing is done we need to extract interconnect rc delays in order to perform the post routing timing analysis.

The RC extraction was done and the statistics reported are as follows:



```
innovus 9> Performing RC Extraction ...
Extraction called for design 'top' of instances=2510 and nets=5521 using extraction engine 'preRoute' .
**WARN: (IMPEXT-3530):  The process node is not set. Use the command setDesignMode -process <process node> prior to extraction for maximum accuracy and op
timal automatic threshold setting.
Type 'man IMPEXT-3530' for more detail.
PreRoute RC Extraction called for design top.
RC Extraction called in multi-corner(1) mode.
RCMode: PreRoute
      RC Corner Indexes            0
Capacitance Scaling Factor   : 1.00000
Resistance Scaling Factor    : 1.00000
Clock Cap. Scaling Factor    : 1.00000
Clock Res. Scaling Factor    : 1.00000
Shrink Factor                : 0.90000
PreRoute extraction is honoring NDR/Shielding/ExtraSpace for clock nets.
Using capacitance table file ...
Updating RC grid for preRoute extraction ...
Initializing multi-corner capacitance tables ...
Initializing multi-corner resistance tables ...
Creating RPSQ from WeeR and WRes ...
PreRoute RC Extraction DONE (CPU Time: 0:00:00.1  Real Time: 0:00:01.0  MEM: 1413.473M)
Dumping SPF file.....
Created SPF File: top.spf
```

After routing the post timing analysis was carried out and found that all the paths are meeting the timing requirements.

Then the final stages require physical verification which include verification of geometry and connectivity.

And finally, the gds file is generated.