EDA Tool Tutorial

- Global Routing

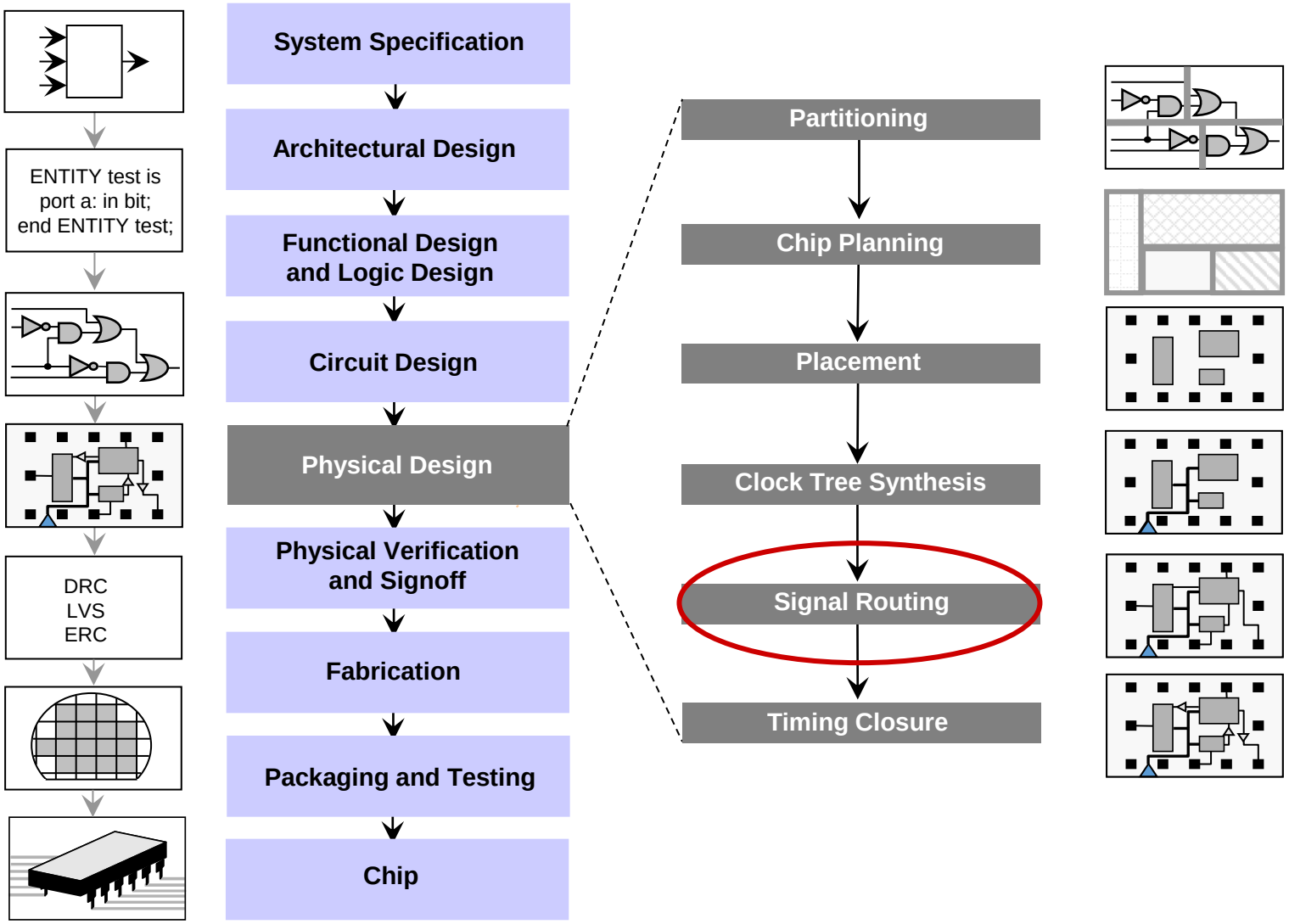
Oct 17, 2024

양자영, 김태환 교수
ECE Dept.
Seoul National University

Outline

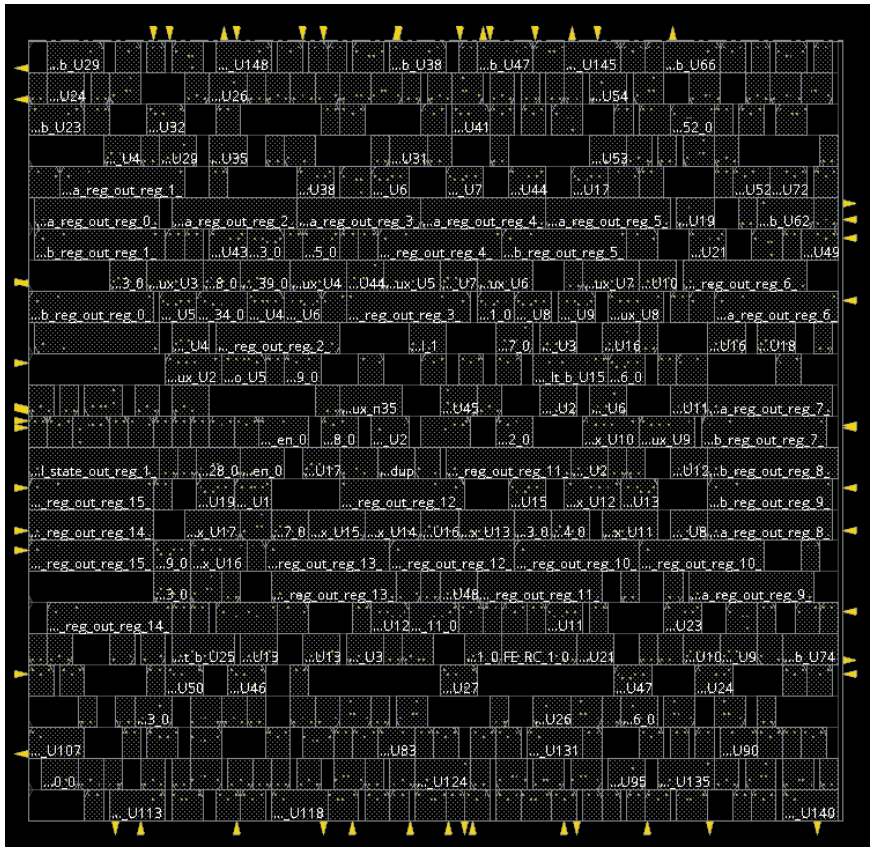
- **Theoretical Background**
  - Introduction to Routing
  - Overview of Modern Global Router Flows
- Lab
  - Base P&R Flow
  - Random Sensitivity Analysis
  - Feature Extraction

Introduction

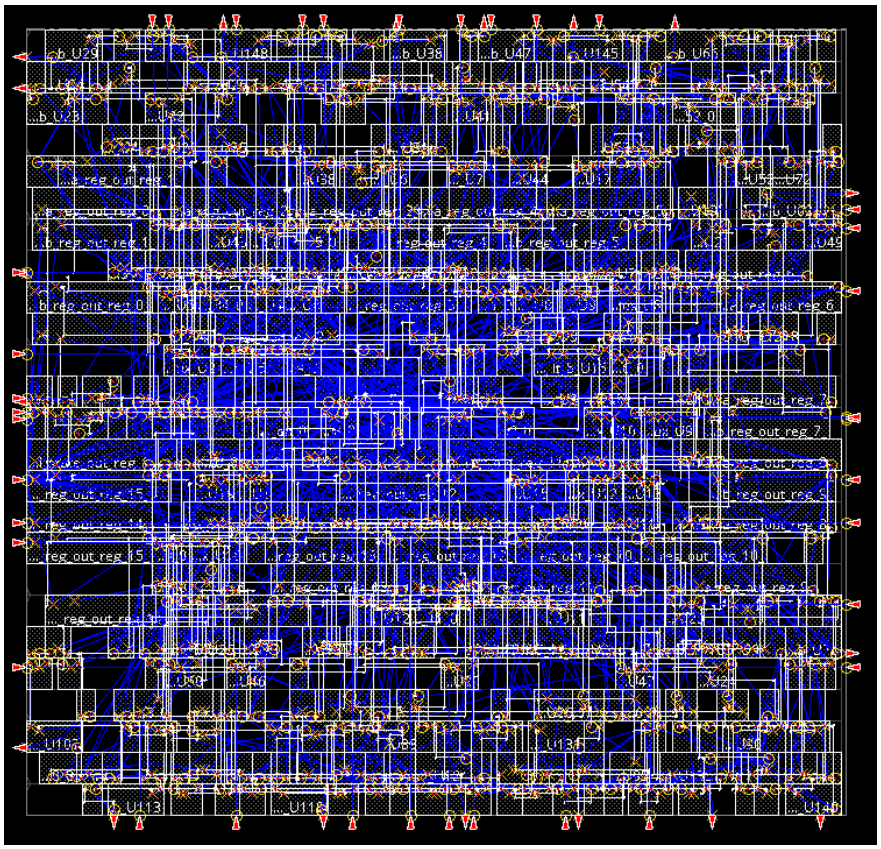


Introduction - Routing

Given a placement, a netlist and technology information,



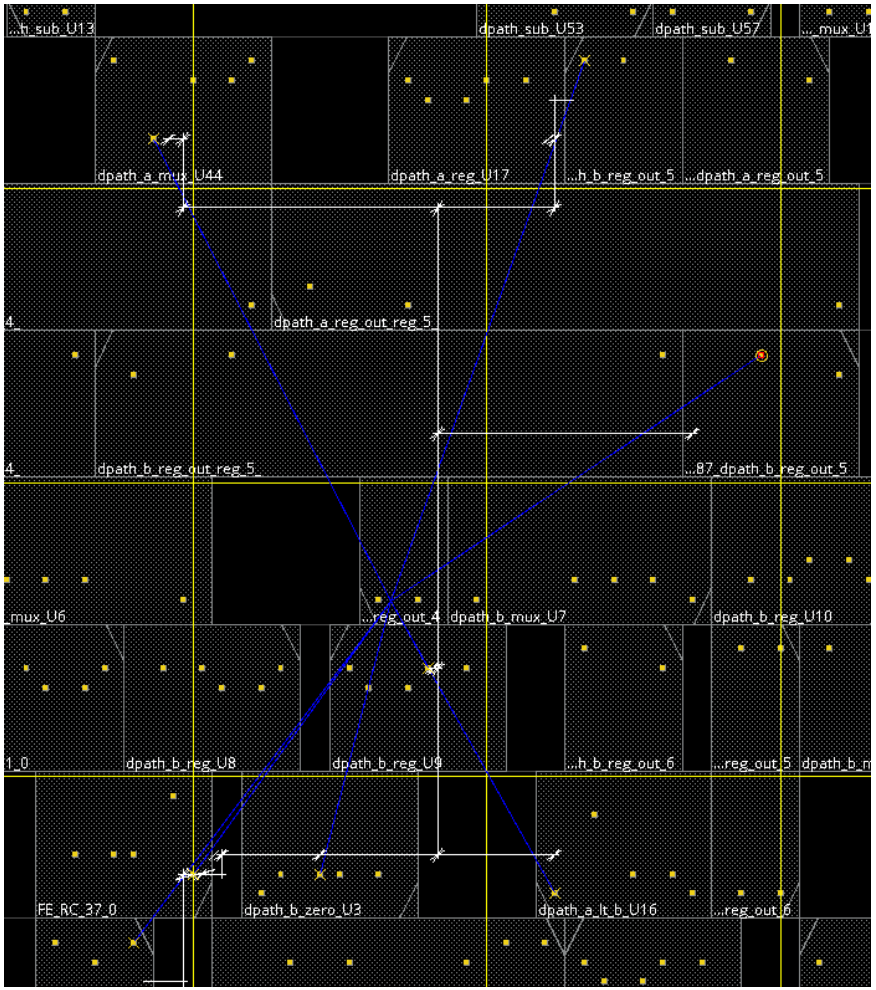
Placement



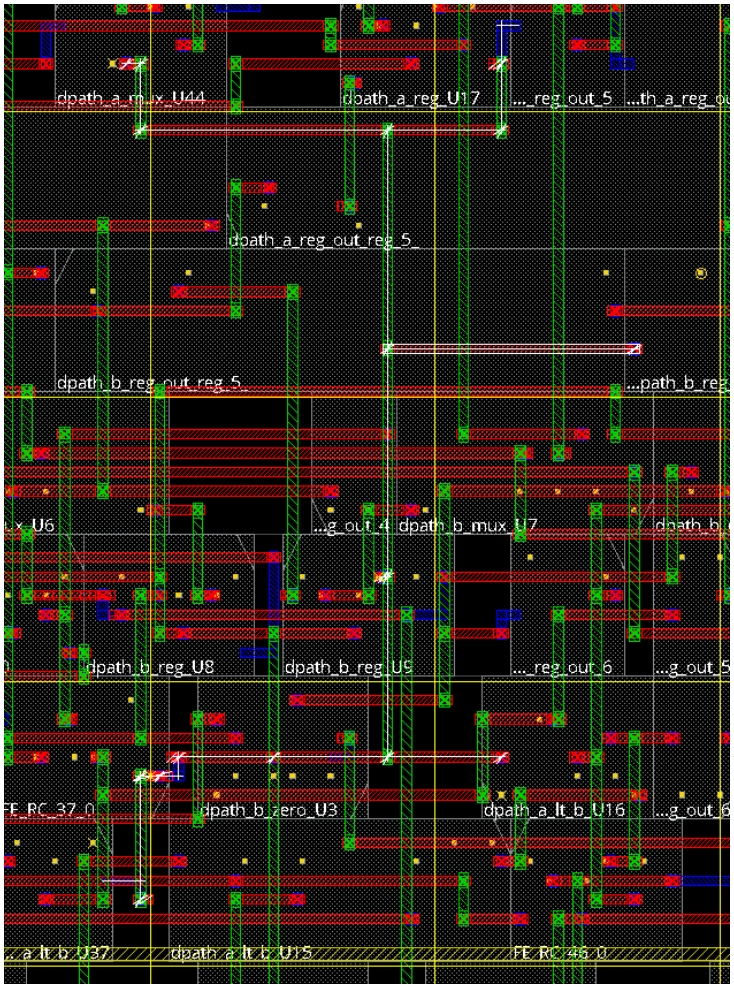
Nets

Global & Detailed Routing

Determine the necessary wiring, specific routing segments, to connect cells(pins)

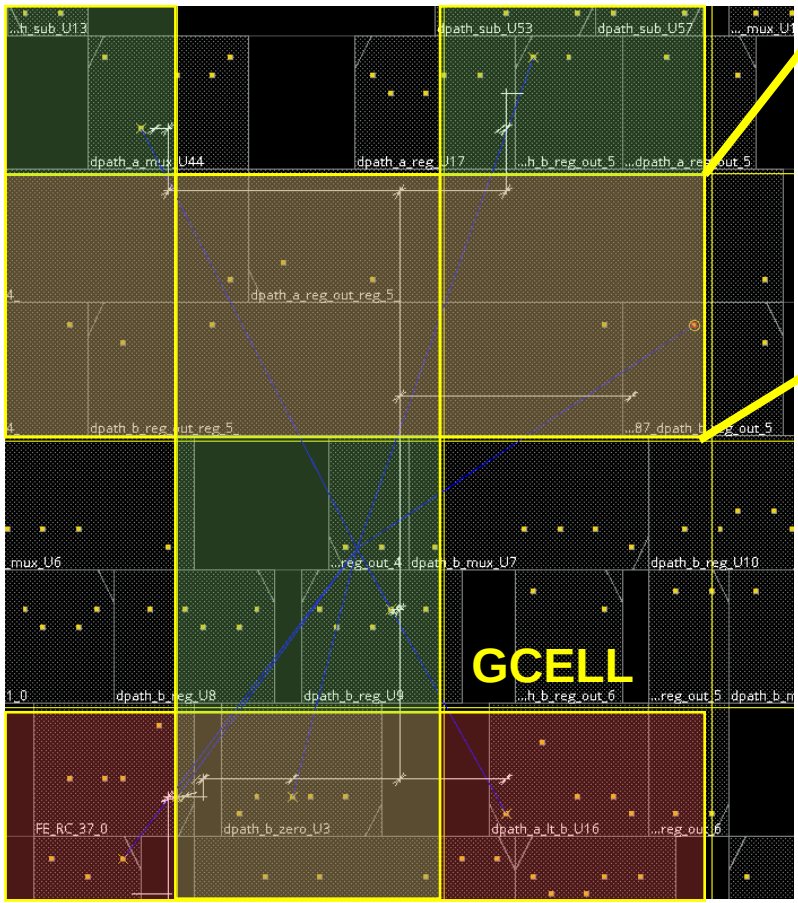


Pins on a net

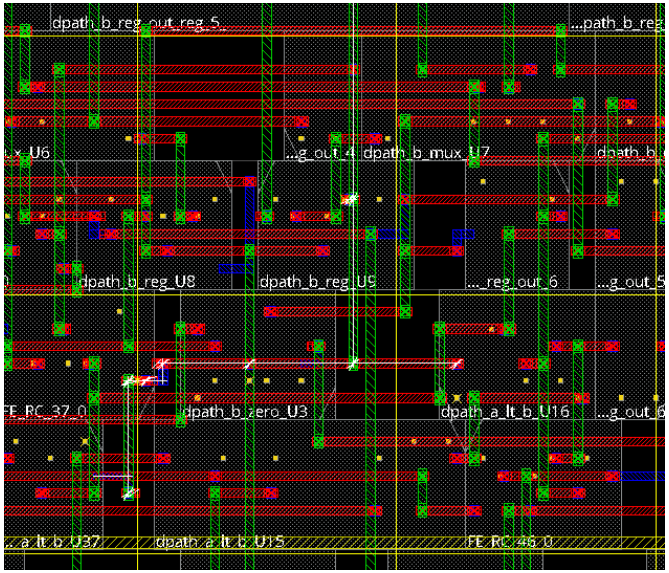


Post-route

Global & Detailed Routing



Track
(M2 Layer)

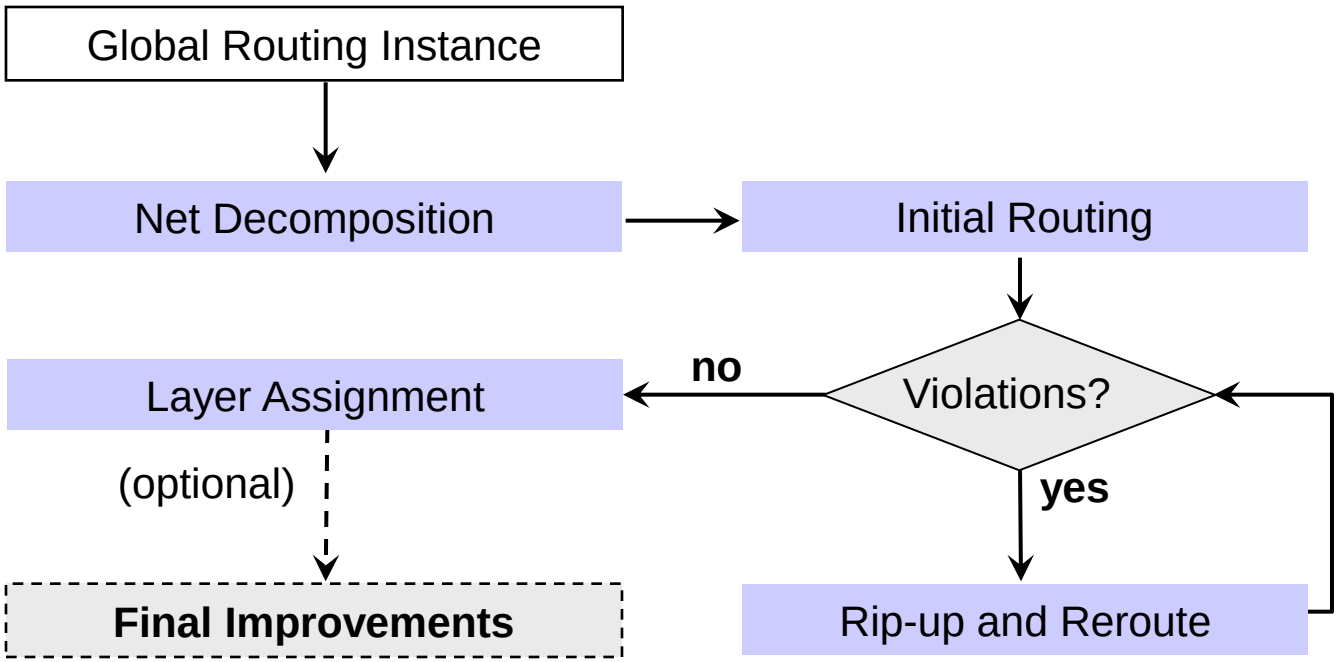


Global Routing
- Partitions the routing region into global routing cells (gcells)
- Plans routes as sequences of gcells
- Minimizes total length of routes and, possibly, routed congestion

Detailed Routing
- Seeks to implement each global route as a sequence of track segments

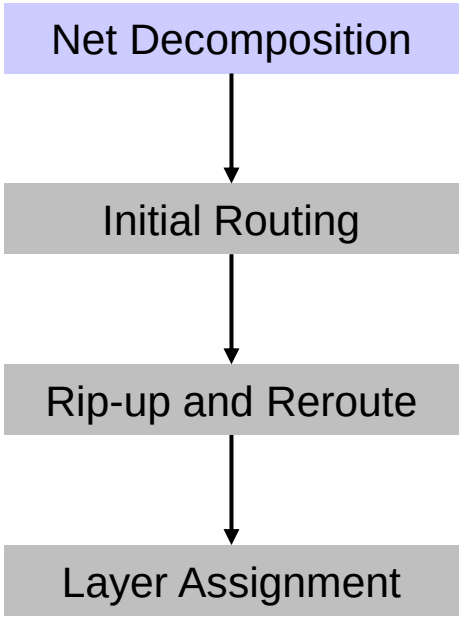
Flow for Modern Global Routers

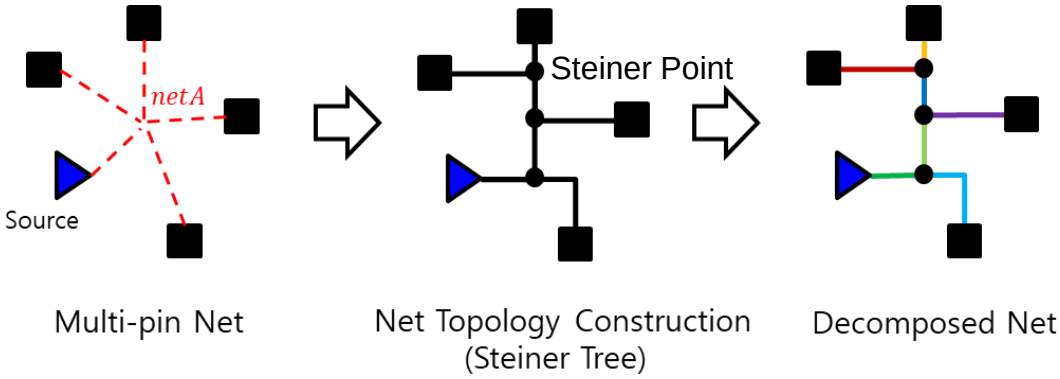
- **General flow for modern global routers, where each router uses a unique set of optimizations:**



* Reference: VLSI Physical Design: From Graph Partitioning to Timing Closure

Net Decomposition



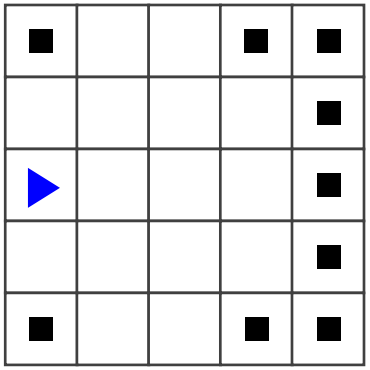


Net Topology Construction – Background

▶ Source ■ Sink ■ Critical sink

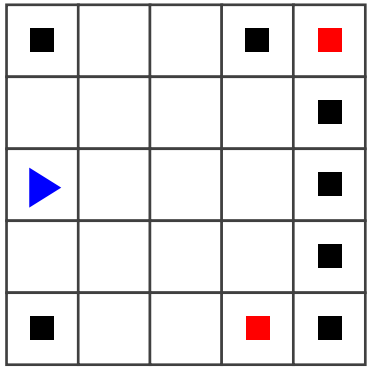
Pins on grid

Case 1



Net with non-critical pins

Case 2



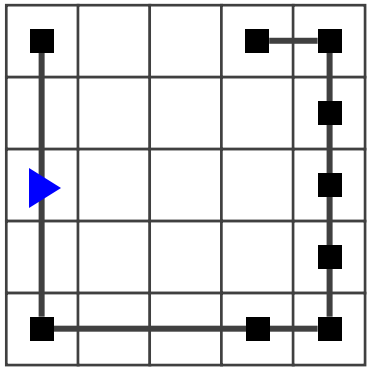
Net with critical pins

Net Topology Construction – Background

▶ Source ■ Sink ■ Critical sink

RSMT*

Case 1

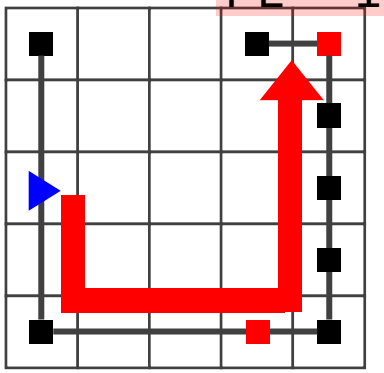


WL=13

Minimal WL

PL^{crit}=10

Case 2



WL=13

Long PL to critical pin
→ Timing violation

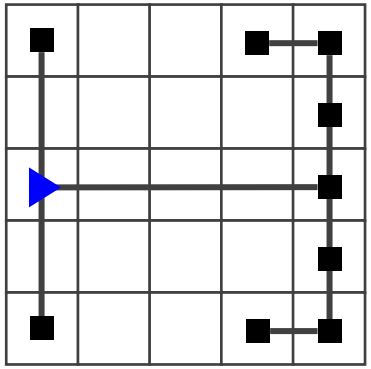
* Rectilinear Steiner Minimum Tree: **FLUTE(2008)**, REST(2021)

Net Topology Construction – Background

▶ Source ■ Sink ■ Critical sink

RSSLT**

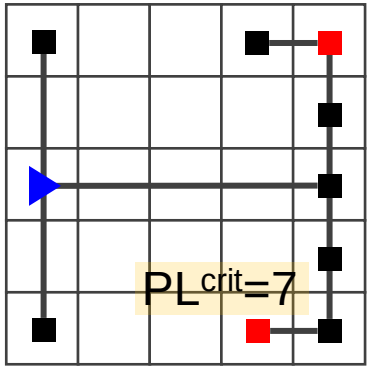
Case 1



WL=14

Longer WL
→ Increase power & routing congestion

Case 2



WL=14

Reasonable PL to all pins

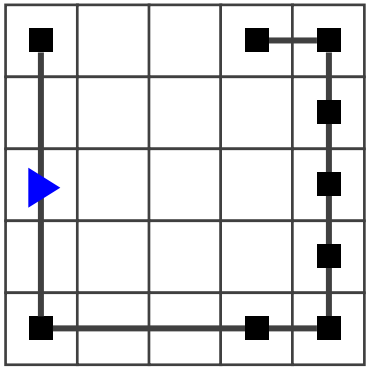
** Rectilinear Steiner Shallow-Light Tree: PD-II(2018), SALT(2020)

Net Topology Construction – Our Motivation

Source Sink Critical sink

Proposed

Case 1

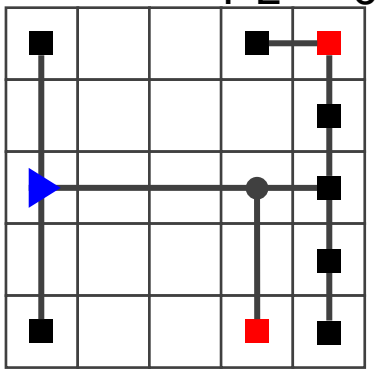


WL=13

Minimal WL

Case 2

$PL^{crit}=6$



WL=15

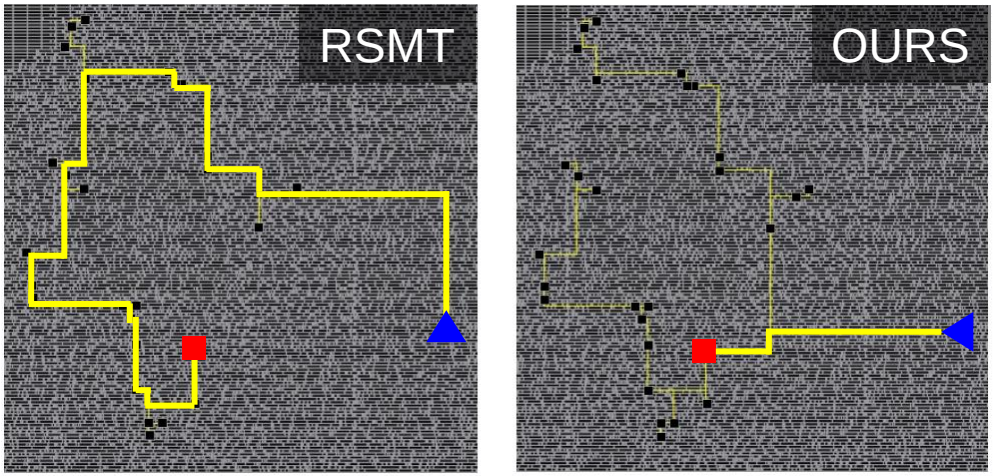
Short PL to critical pins

Robust timing
w/ minimal WL increase

Net Topology – Experimental Results

- Integrated our methodology into **OpenROAD(C++)**
- FLUTE(RSMT) vs PD-II(RSSLT) vs Ours

Example



Delay to the critical pin:

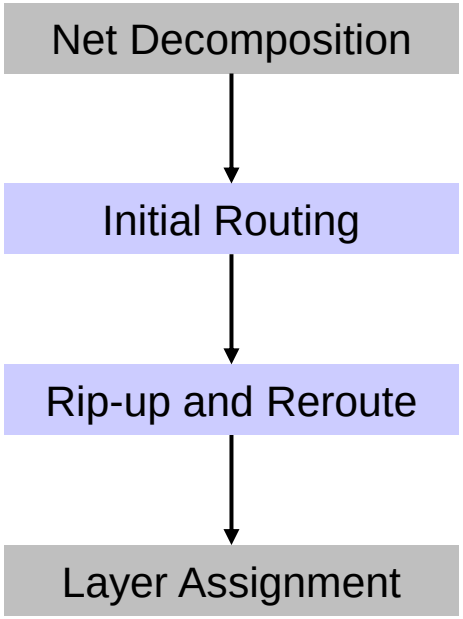
$64.6ps \rightarrow 21.7ps$

Comparison of results for the post-route final designs

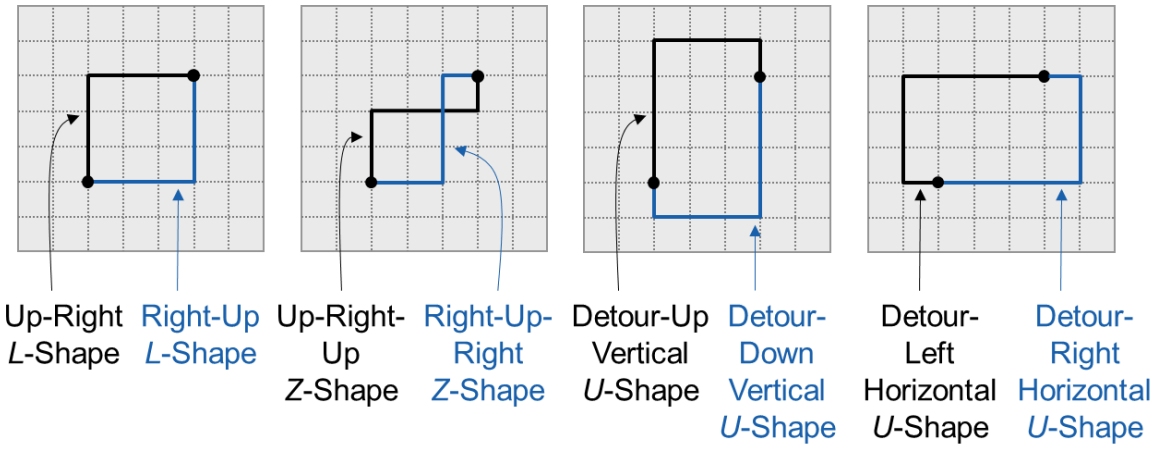
WNS: -18.28% TNS: -53.37% Power: -0.06%

Reference: Jayoung Yang and Taewhan Kim, "Improving Timing Quality Through Net Topology Optimization in Global Routing"
IEEE International System-on-Chip Conference (SOCC) 2024

Pattern Routing



- **Pattern Routing**
  - Searches through a small number of route patterns to im prove runtime
  - Topologies commonly used in pattern routing: *L*-shapes, *Z*-shapes, *U*-shapes



Pattern Routing

Net Decomposition

↓

Initial Routing

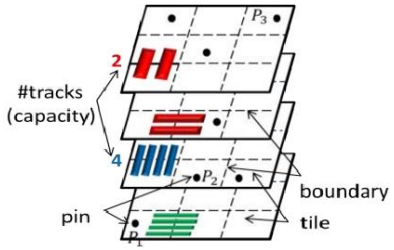
↓

Rip-up and Reroute

↓

Layer Assignment

- **Negotiated-Congestion Routing**



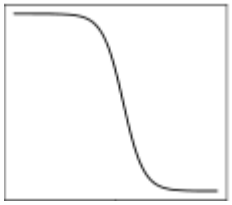
$capacity(e)$: Edge e를 지나갈 수 있는 최대 track 수
(주로 $\frac{gcell_tile_size}{track_pitch}$로 계산)

$demand(e)$: Edge e에서 routing에 사용된 track 수

The edge cost ***cost(e)*** is increased according to the *edge congestion*

Cost function of CUGR 2.0

$1/(1+e^{capacity(e)-demand(e)})$



$capacity(e) - demand(e)$

Outline

- Theoretical Background
  - Introduction to Routing
  - Overview of Modern Global Router Flows

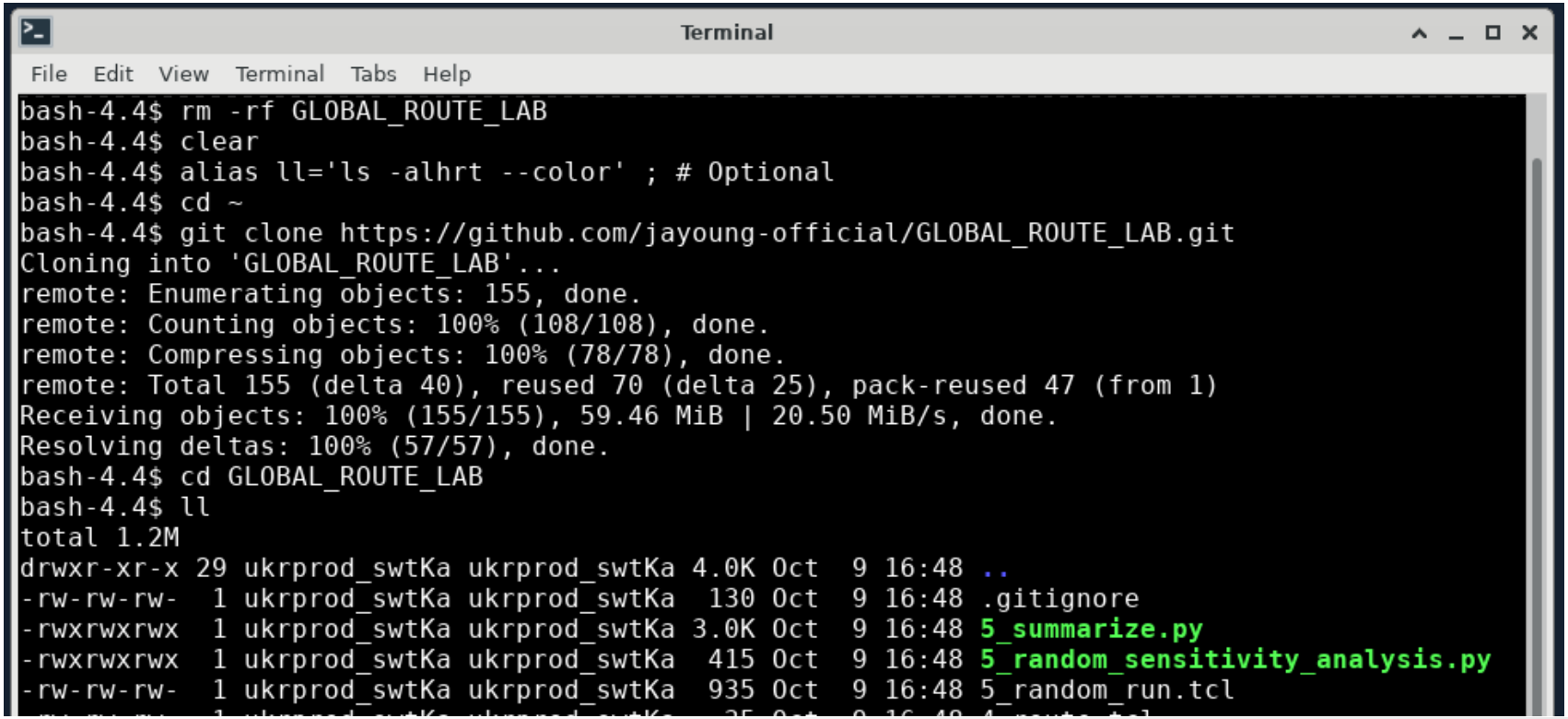
- **Lab**
  - Base P&R Flow
  - Random Sensitivity Analysis
  - Feature Extraction

[LAB1] Download LAB Materials

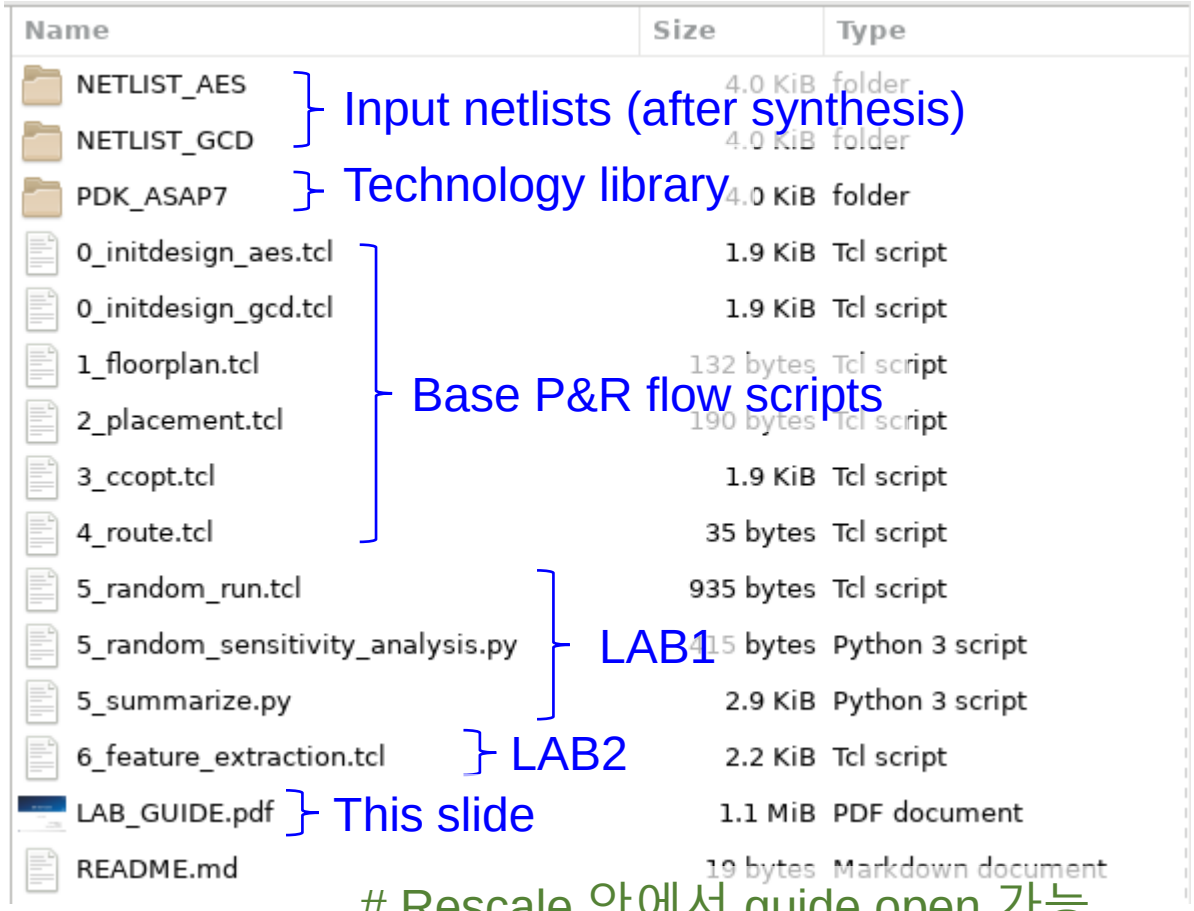
```
alias ll='ls -alhrt --color' ; # Optional
cd ~
git clone https://github.com/jayoung-official/GLOBAL_ROUTE_LAB.git
cd GLOBAL_ROUTE_LAB
```

git clone 대신 copy도 가능

```
cp -rf ~/storage_ojXWdb/LAB_Download/241017_SNUCAD/GLOBAL_ROUTE_LAB ~
```



[LAB1] LAB Materials

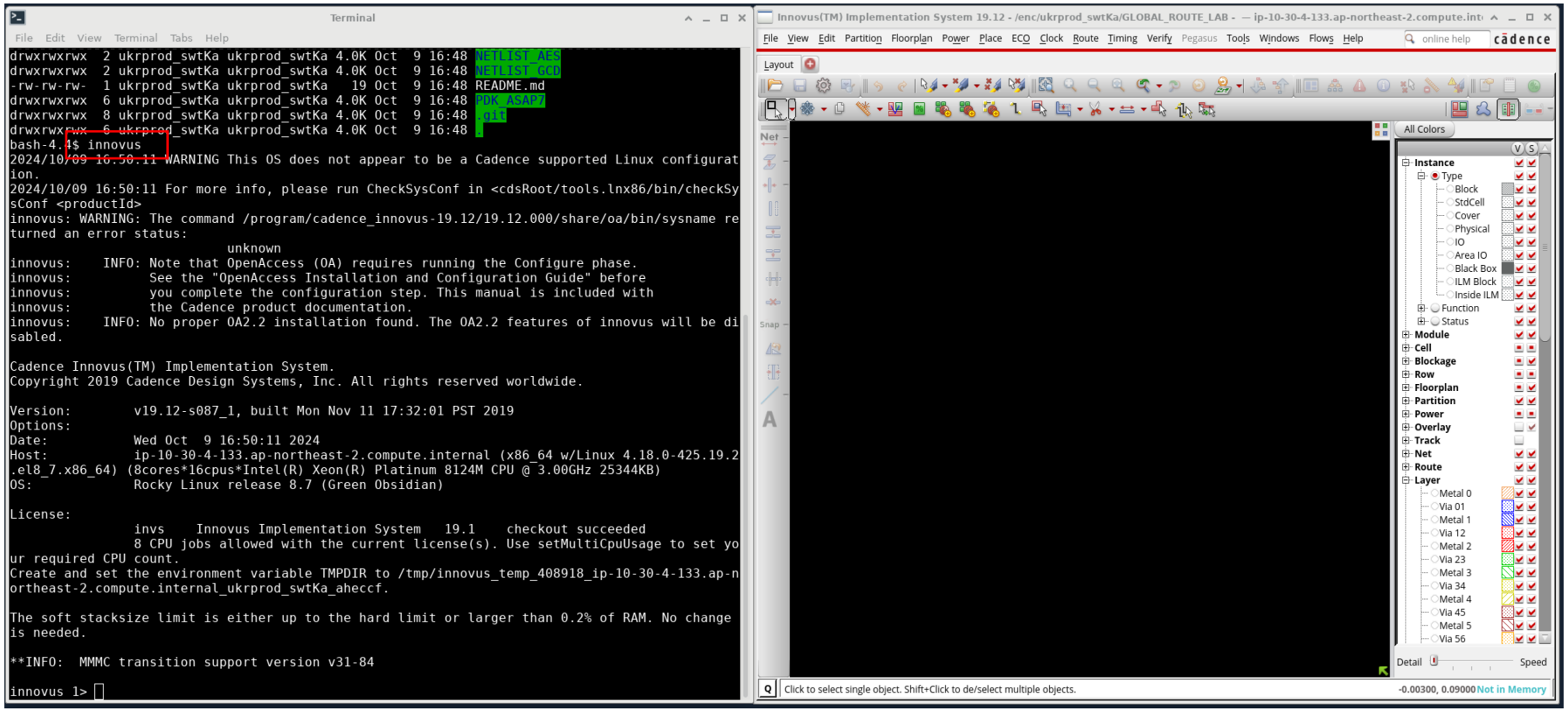


Rescale 안에서 guide open 가능

```
evince ./LAB_GUIDE.pdf &
```

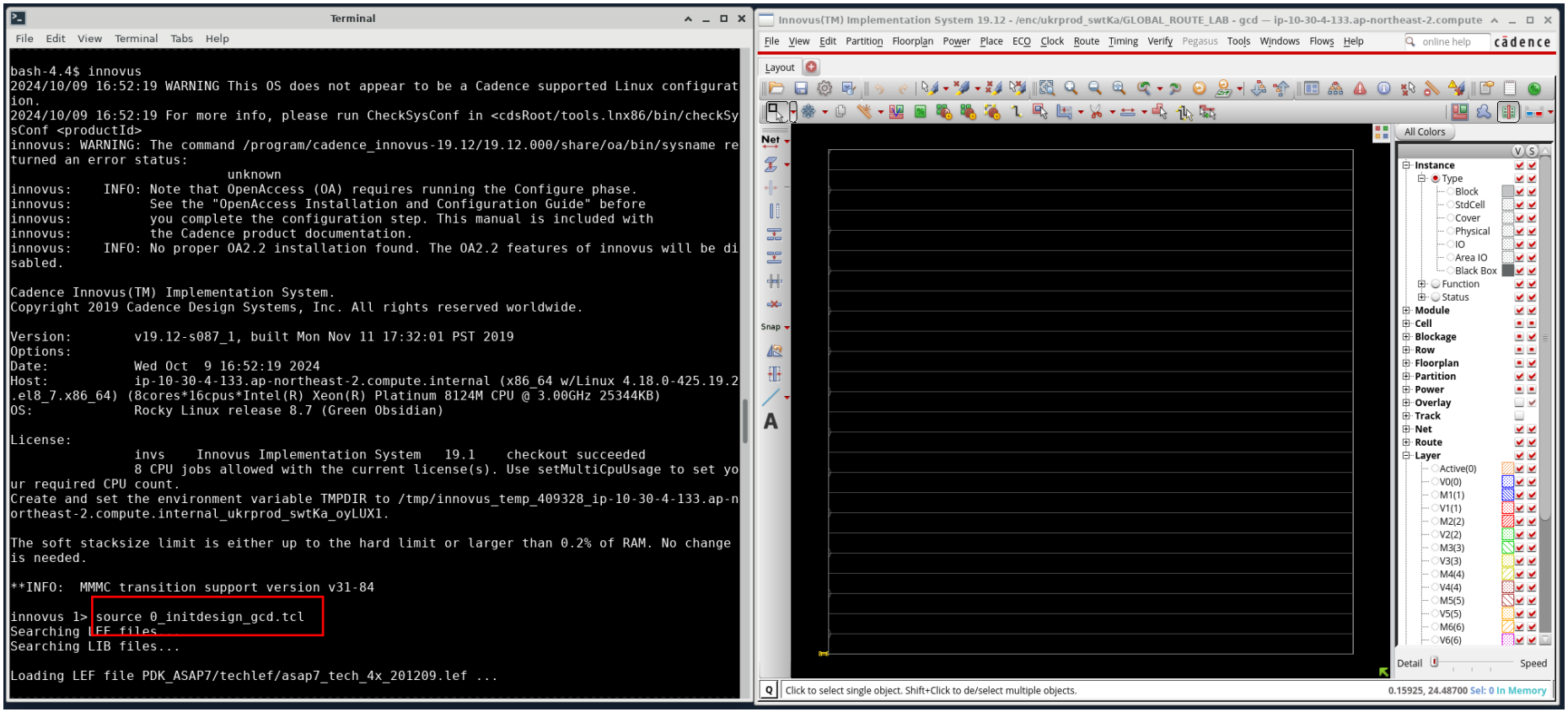
[LAB1] Run Innovus

```
innovus
```



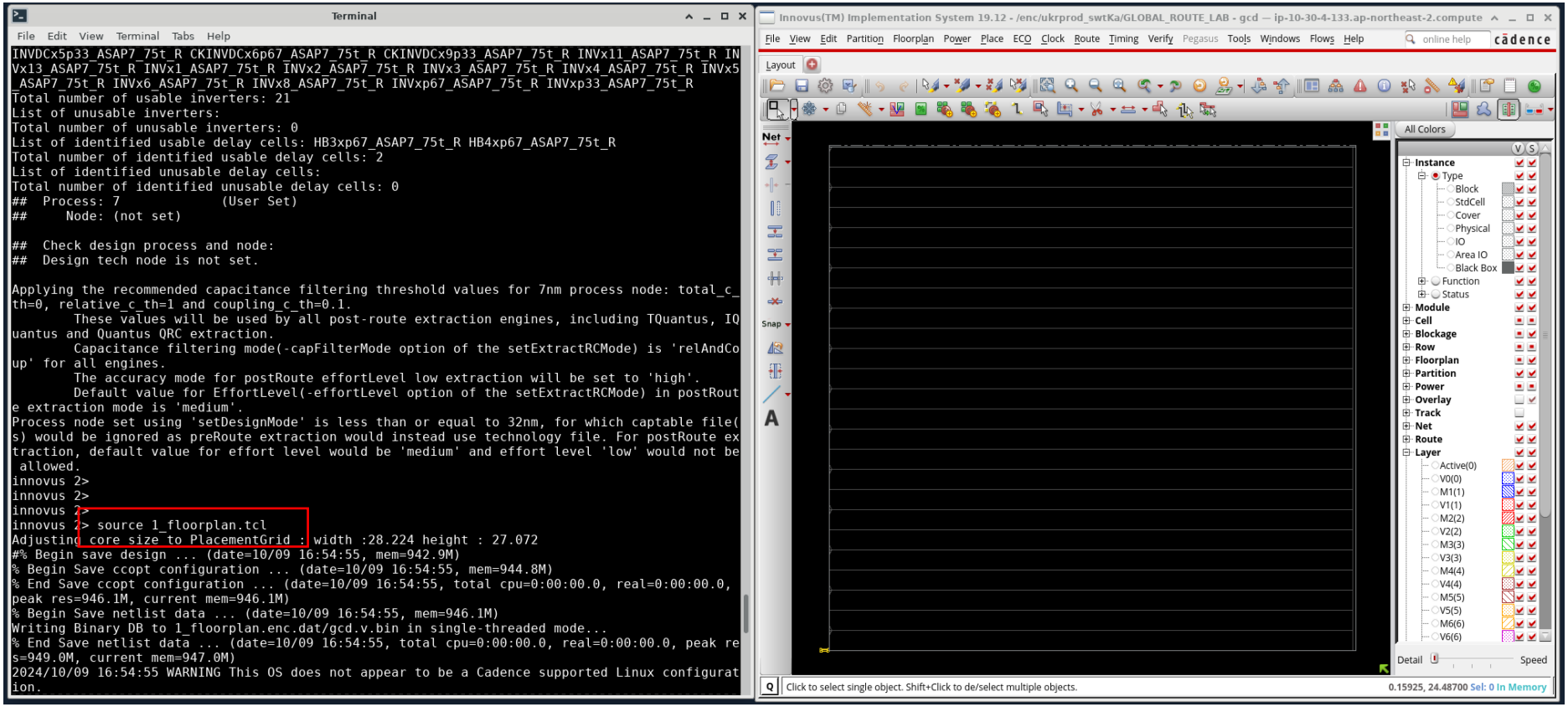
[LAB1] Run Init Design

```
source 0_initdesign_gcd.tcl
```



[LAB1] Run Floorplan

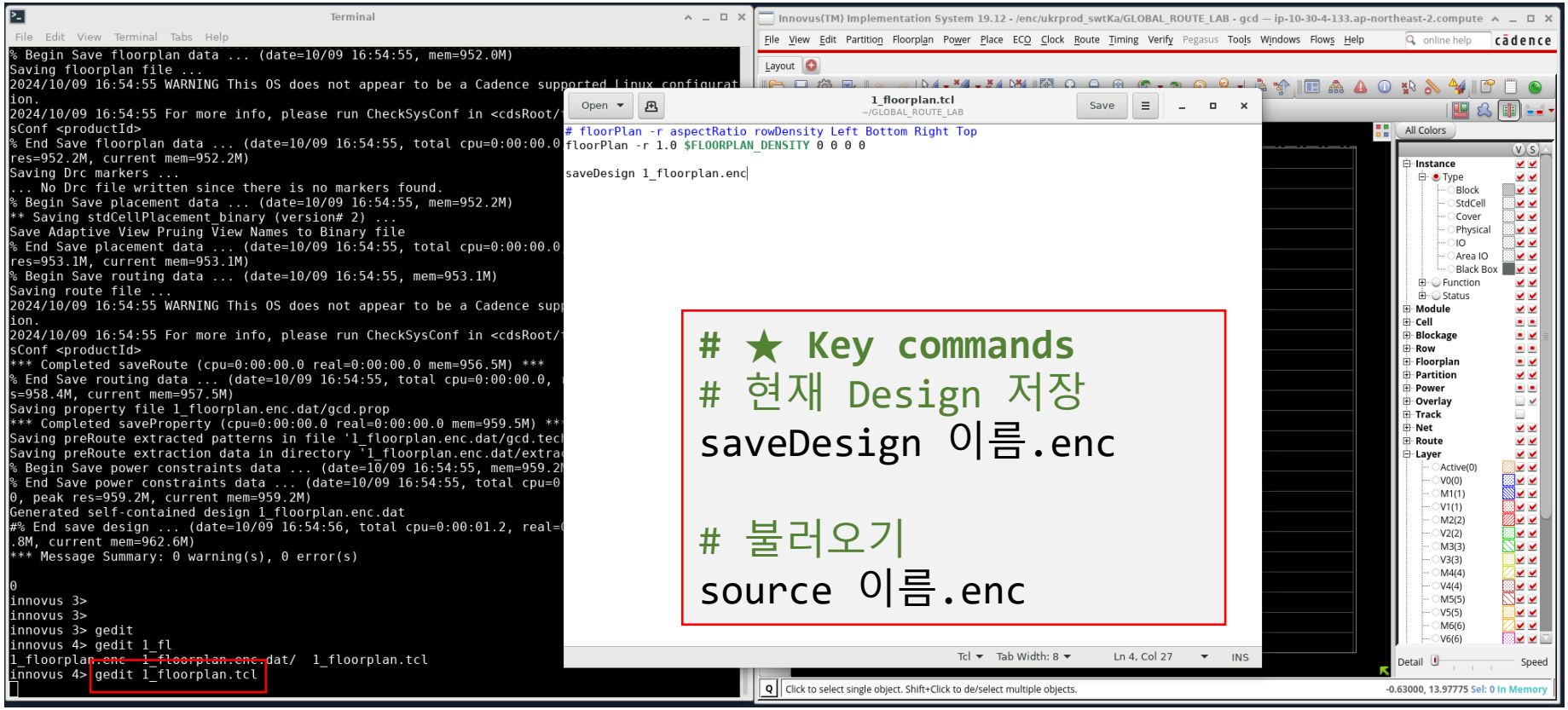
```
source 1_floorplan.tcl
```



Layout area floorplan

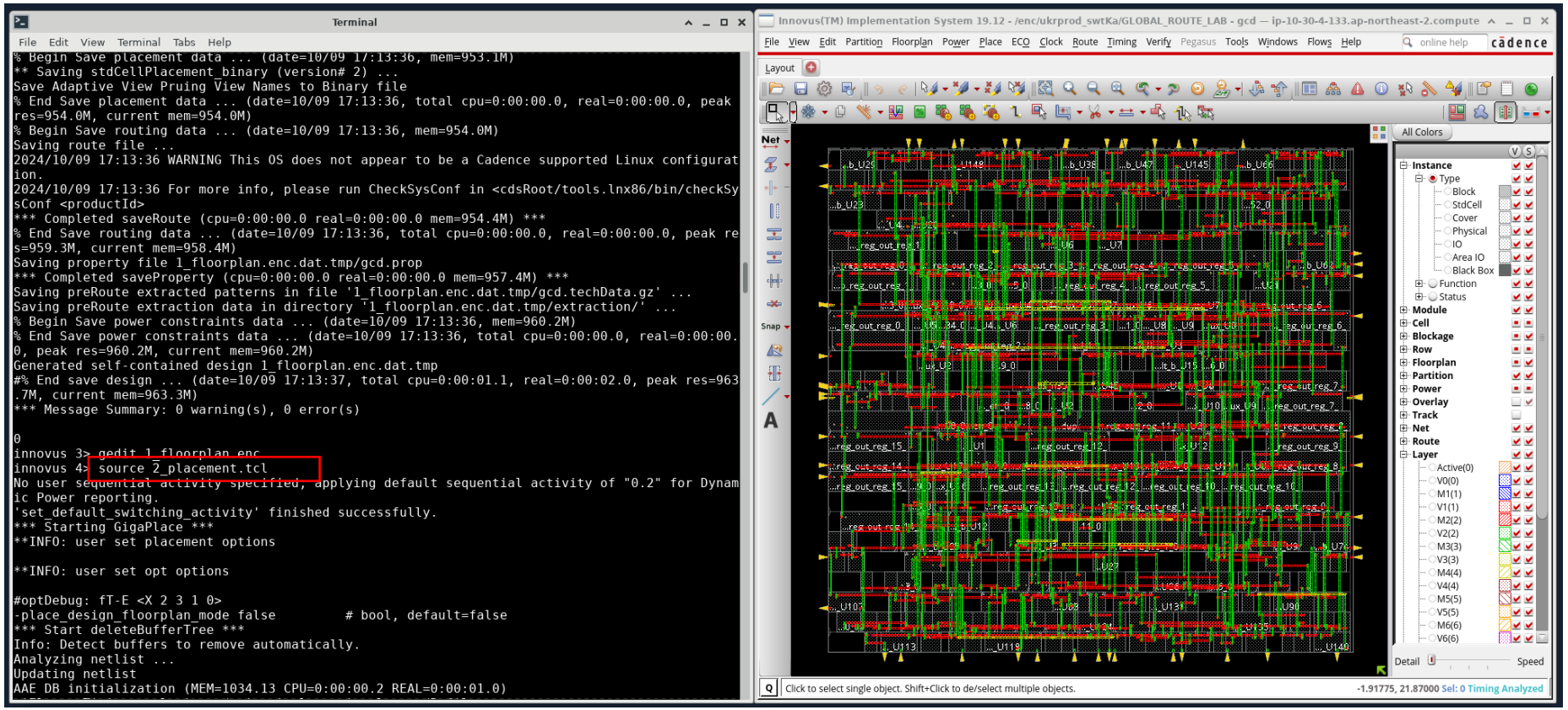
[LAB1] Run Floorplan

```
gedit 1_floorplan.tcl
```



[LAB1] Run Placement

```
source 2_placement.tcl
```

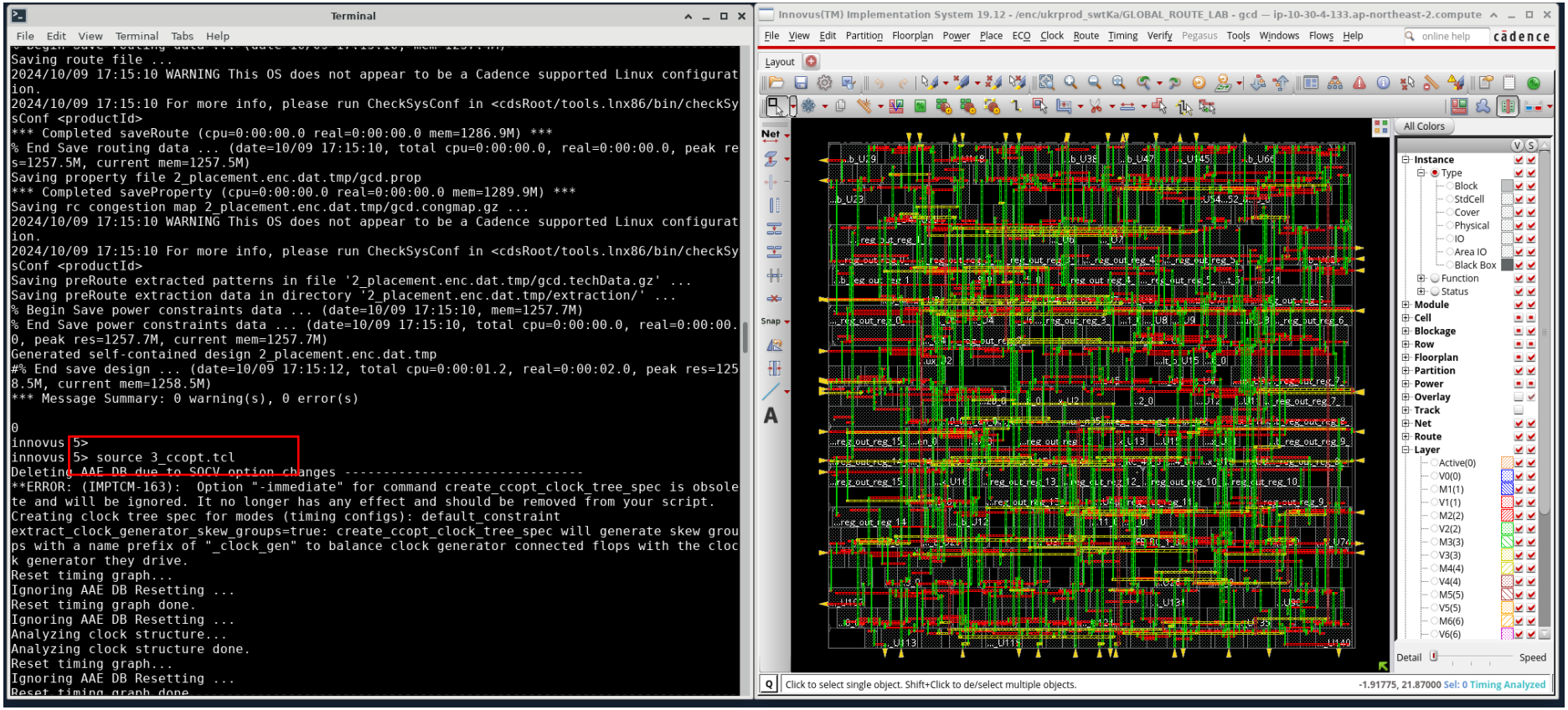


Place instances(cells)

placement w/ trial route

[LAB1] Run CTS

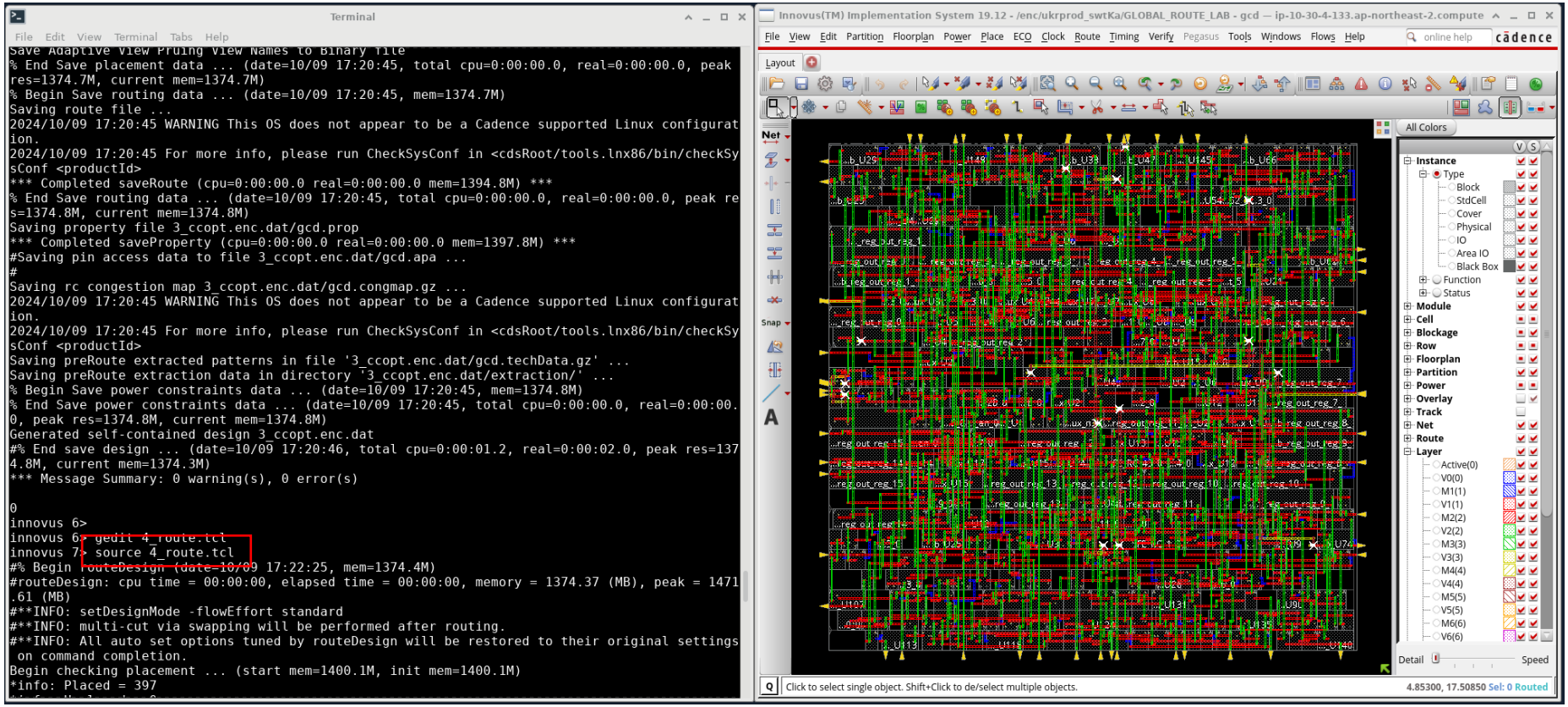
```
source 3_ccopt.tcl
```



CTS + Optimization

[LAB1] Run Route

```
source 4_route.tcl
```

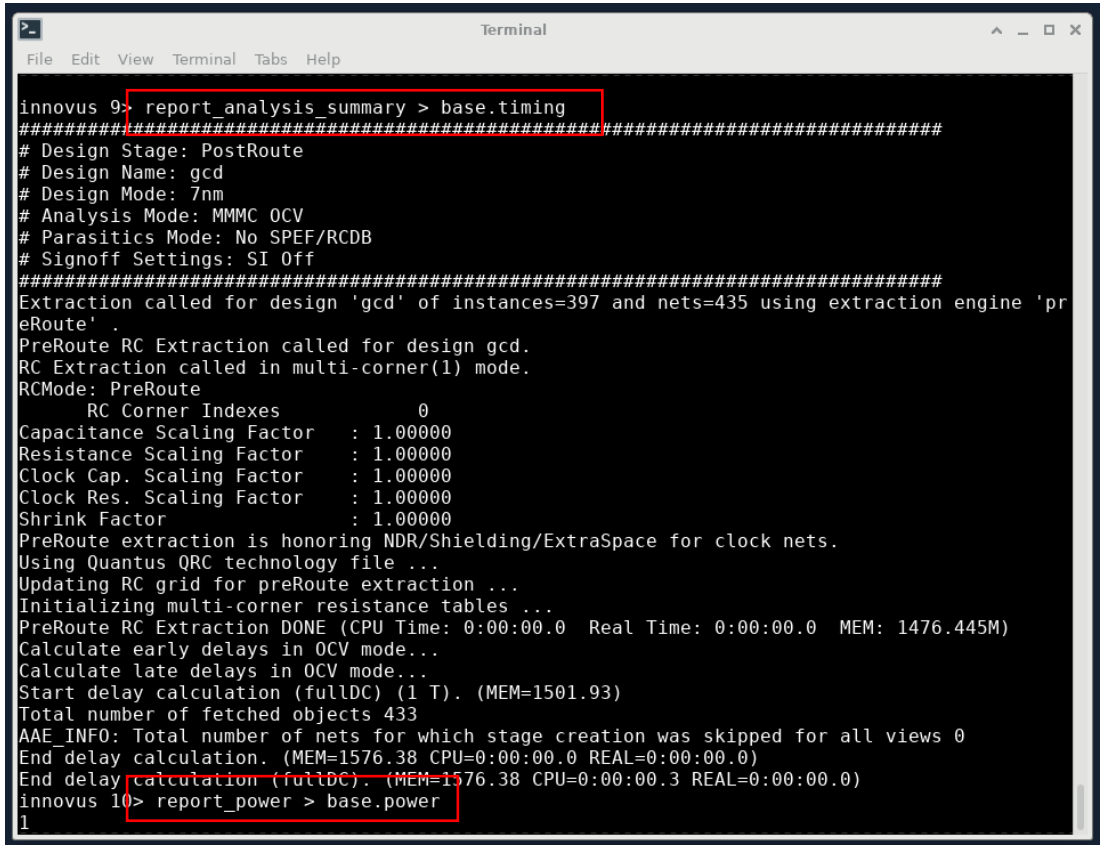


Global + Detail Route

[LAB1] Report

기본 Flow 결과 Report 저장

```
report_analysis_summary > base.timing
report_power > base.power
```



Report 확인

```
gedit base.timing
gedit base.power
```

[LAB1] Objective - Random Sensitivity Analysis

Random exploration can be used to identify **potential improvements over default flow**

- **Ex) Random-ordered Cell Placement → Placement metric change**

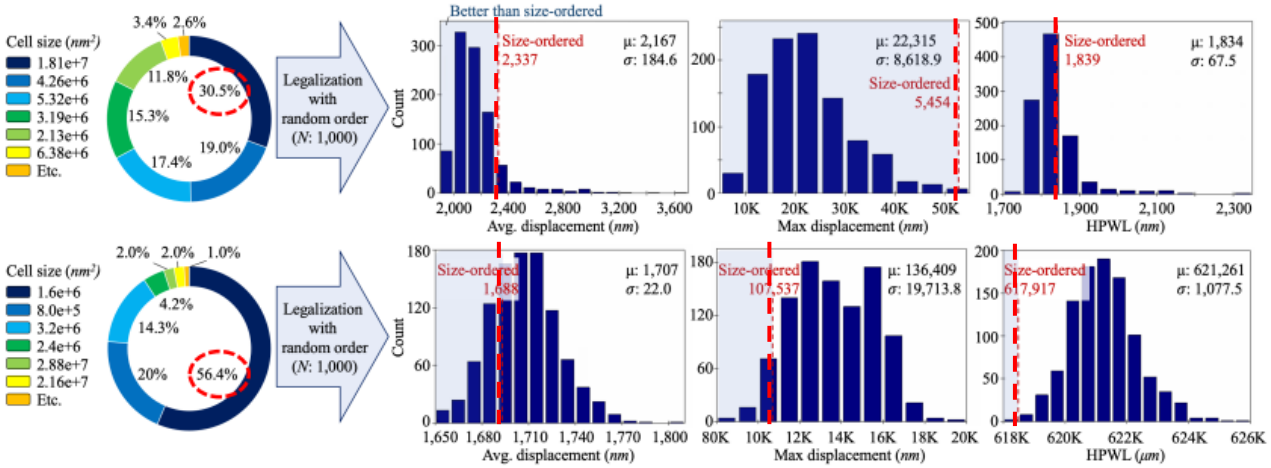
S. -Y. Lee, et al., "RL-Legalizer: Reinforcement Learning-based Cell Priority Optimization in Mixed-Height Standard Cell Legalization," 2023 DATE

(a) usb_phy

OpenCores design implemented with 75% util. and 1.0 aspect ratio in 45-nm NanGate technology.

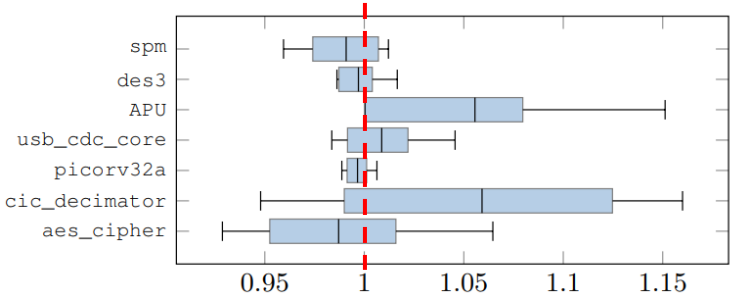
(b) pci_bridge32_b_md3

ICCAD-2017 CAD contest benchmark.



- **Ex) Random Steiner Point Disturbance → Timing metric change**

S. Liu, Z. et al., "Concurrent Sign-off Timing Optimization via Deep Steiner Points Refinement," 2023 DAC



Distribution of sign-off TNS ratio of the updated solution with random Steiner point disturbance to the original one

[LAB1] Adjustable parameters in Routing (Innovus)

```
setAttribute -net <net_name> -avoid_detour {true|false} -weight <integer> …
```

Attaches attributes to nets and subnets. Attaching the attributes allows the NanoRoute routing commands to route the nets following specific requirements.

Parameters

`-avoid_detour {true | false}`

Avoids detours of roughly more than a few gcell grids on the specified nets. This attribute affects global routing only. (Default: false)
Note: Cadence recommends that you use caution with this attribute, as it adds congestion to the design.

`-weight <integer>`

Specifies a relative weight for routing nets. In each switch box, the NanoRoute router routes nets with the highest weight first, then the next highest weight, and so on. Specify a value higher than 2 to ensure a net is routed before other nets. (Default: **2**)

[LAB1] Run Random Sensitivity Analysis

Random Sensitivity Analysis 실행 전 innovus 종료하여 shell로 이동

```
exit
```

Check run script(python)

```
gedit 5_random_sensitivity_analysis.py
```

Open | 5_random_sensitivity_analysis.py ~/GLOBAL_ROUTE_LAB | Save

```
#!/usr/bin/env python3
import subprocess

N_RUNS = 5

def run_script():
    try:
        result = subprocess.run('innovus -files 5_random_run.tcl', shell=True)
        if result.stderr:
            print("Error:")
            print(result.stderr)
    except Exception as e:
        print(f"An error occurred: {e}")

for _ in range(N_RUNS):
    run_script()
```

5_random_run.tcl 스크립트를 5회 반복 수행

Python 3 | Tab Width: 8 | Ln 1, Col 1 | INS

[LAB1] Run Random Sensitivity Analysis

```
# Check 5_random_run.tcl
```

```
gedit 5_random_run.tcl
```

Open | 5_random_run.tcl ~/GLOBAL_ROUTE_LAB | Save

```
source 3_ccopt.enc

set timestamp [clock format [clock seconds] -format "%y%m%d_%H%M%S"]
puts $timestamp

set avoid_detour_prob [expr {(int(rand() * 5) + 1) * 0.1}]  ;# 0.1에서 0.5 사이

set dataList {}

foreach_in_collection net [get_nets] {
    set weight [expr {int(rand() * 4) + 2}] ; # 2에서 5 사이의 값
    set avoid_detour [expr {rand() < ${avoid_detour_prob} ? "true" : "false"}]
    set net_name [get_property $net name]
    setAttribute -net $net_name -weight $weight -avoid_detour $avoid_detour

    lappend dataList [list $net_name $weight $avoid_detour]
}

routeDesign

# REPORTS
set fileId [open "random_${timestamp}_${avoid_detour_prob}.nets.csv" "w"]
puts $fileId "Net,Weight,Avoid_Detour"
foreach row $dataList {
    puts $fileId [join $row ","]
}
close $fileId

report_analysis_summary > "random_${timestamp}_${avoid_detour_prob}.timing"
report_power > "random_${timestamp}_${avoid_detour_prob}.power"

exit
```

Tcl | Tab Width: 8 | Ln 1, Col 1 | INS

3_ccopt.enc Design을 불러온 후
net별 weight, avoid_detour 랜덤 할당

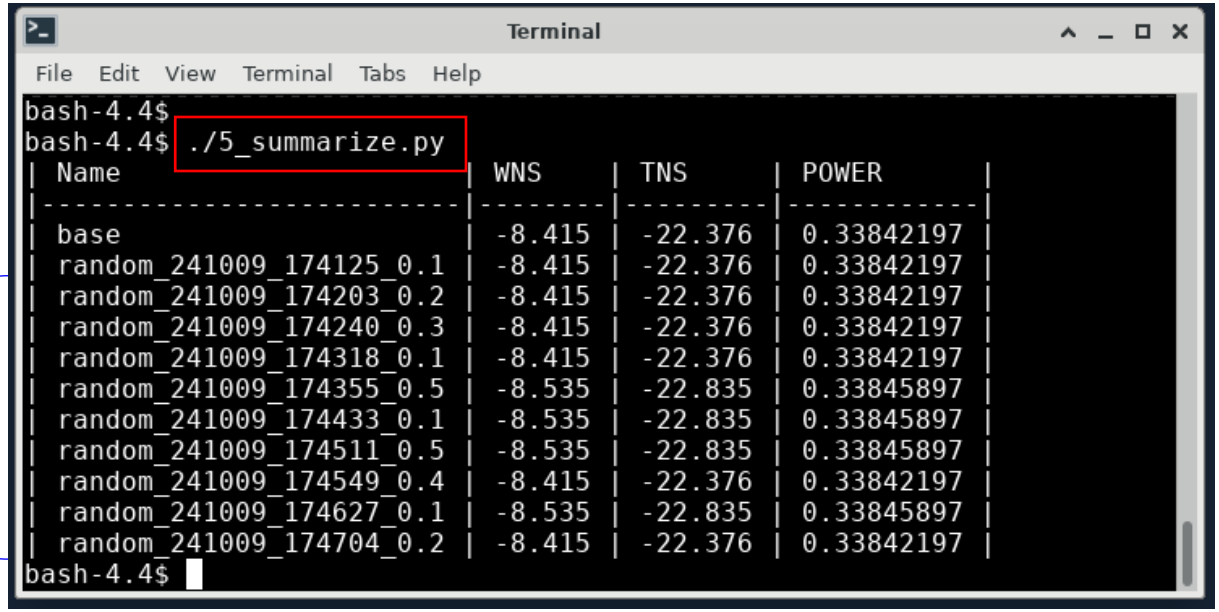
결과 비교를 위한 Report 후 Exit

[LAB1] Run Random Sensitivity Analysis

```
# Run script 실행 / 약 5분 소요(N_RUNS=5 기준)
./5_random_sensitivity_analysis.py

# 결과 조회 (*.timing, *.power 파싱)
./5_summarize.py
```

기본 Flow

Random x10



Toy design net 수가 적어(약 400개) 큰 효과 없음

[LAB1] Results for AES_CIPHER_TOP

#Nets of Design: 약 10000개

Name	WNS	TNS	POWER
base	**-9.208**	**-339.913**	**4.62745333**
random_241009_055859_0.5	-9.684	-374.335	4.62808571
random_241009_060453_0.5	-9.812	**-333.828**	4.62767426
random_241009_061049_0.4	-10.168	-373.995	4.62823325
random_241009_061644_0.4	-10.022	-365.142	4.62761984
random_241009_062236_0.4	-9.605	-347.652	4.62797298
random_241009_062832_0.2	-11.032	-363.802	4.62793837
random_241009_063425_0.3	-10.568	-359.964	4.62790291
random_241009_064018_0.4	-9.236	-363.022	4.62795988
random_241009_064607_0.3	-11.273	-368.496	4.62805885
random_241009_065204_0.4	-11.753	-370.290	4.62816721
random_241009_093218_0.5	-9.793	-349.804	4.62796977
random_241009_093814_0.1	-12.990	-428.342	4.62795213
random_241009_094407_0.3	-11.646	-372.209	**4.62724831**
random_241009_094958_0.1	-9.773	-356.929	**4.62731261**
random_241009_095549_0.3	-9.290	-369.146	4.62773923
random_241009_100143_0.5	-10.732	-374.225	4.62784993
random_241009_100733_0.5	-11.444	-414.964	4.62819967
random_241009_101322_0.3	**-8.441**	**-337.672**	4.62749583
random_241009_101909_0.4	-9.937	**-335.503**	**4.62738434**
random_241009_102501_0.5	-11.339	-367.085	4.62801802

개선 가능성 확인

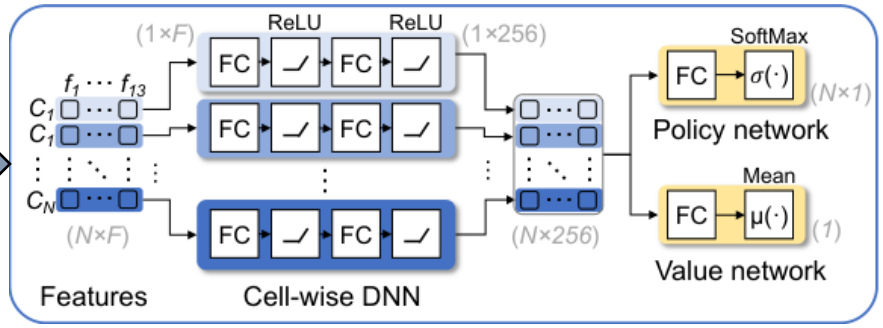
[LAB2] Feature Extraction

Extensive feature extraction is crucial for optimization / machine learning.

- **Ex) RL-based placement priority optimization**

S. -Y. Lee, et al., "RL-Legalizer: Reinforcement Learning-based Cell Priority Optimization in Mixed-Height Standard Cell Legalization," 2023 DATE

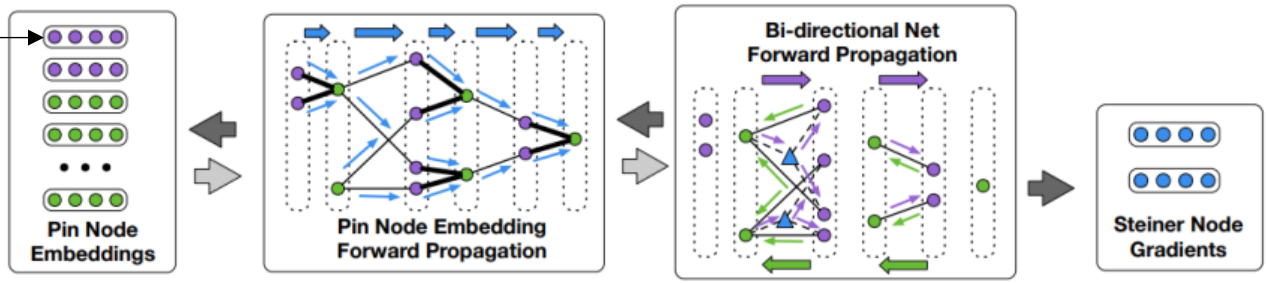
7 cell features
(1) X- & **(2)** Y-coordinate of the *i*-th cell c_i
(3) Width & **(4)** Height of c_i
(5) Number of nets connected to c_i
(6) Number of overlapped cells to c_i
(7) Avg. distance of the 2 nearest obstacles from c_i



- **Ex) Timing Optimization via Deep Steiner Points Refinement**

S. Liu, Z. et al., "Concurrent Sign-off Timing Optimization via Deep Steiner Points Refinement," 2023 DAC

Type	Name
Features	is primary I/O pin or not is fanin or fanout distance to the 4 die area boundaries pin capacitance
Tasks	net delay to root pin arrival time slew is timing endpoint or not required arrival time for endpoints



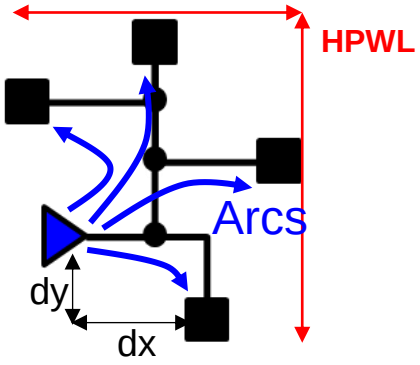
[LAB2] Feature Extraction

```
gedit 6_feature_extraction.tcl
```

6_feature_extraction.tcl
~/GLOBAL_ROUTE_LAB

```tcl
foreach_in_collection net [get_nets] {
    set net_name [get_property $net name]
    set pins [get_pins -of_objects $net]
    set n_pins [sizeof_collection $pins]

    puts "Net: $net_name"
    puts "  Pins: $n_pins"

    set x_coordinates [get_property $pins x_coordinate]
    set y_coordinates [get_property $pins y_coordinate]
    set min_x [lindex [lsort -real $x_coordinates] 0]
    set max_x [lindex [lsort -real $x_coordinates] end]
    set min_y [lindex [lsort -real $y_coordinates] 0]
    set max_y [lindex [lsort -real $y_coordinates] end]
    set hpwl [expr {($max_x - $min_x) + ($max_y - $min_y)}]
    puts "  HPWL: $hpwl"

    set arcs [get_arcs -of_objects $net]
    foreach_in_collection arc $arcs {
        set source_pin [get_property $arc source_pin]
        set sink_pin [get_property $arc sink_pin]
        set source_pin_name [get_property $source_pin full_name]
        set sink_pin_name [get_property $sink_pin full_name]
        puts "  Arc: $source_pin_name -> $sink_pin_name"
        set is_clock [get_property $sink_pin is_clock]
        puts "    is_clock: $is_clock"
        set delay_max_fall [get_property $arc delay_max_fall]
        set delay_max_rise [get_property $arc delay_max_rise]
        if { $delay_max_fall > $delay_max_rise } {
            puts "    delay: $delay_max_fall"
        } else {
            puts "    delay: $delay_max_rise"
        }

        set slack_max_fall [get_property $sink_pin slack_max_fall]
        set slack_max_rise [get_property $sink_pin slack_max_rise]
        if { $slack_max_fall < $slack_max_rise } {
            puts "    slack: $slack_max_fall"
        } else {
            puts "    slack: $slack_max_rise"
        }

        set source_coordinate_x [get_property $source_pin x_coordinate]
        set source_coordinate_y [get_property $source_pin y_coordinate]
        set sink_coordinate_x [get_property $sink_pin x_coordinate]
        set sink_coordinate_y [get_property $sink_pin y_coordinate]
        set diff_x [expr {abs($source_coordinate_x - $sink_coordinate_x)}]
        set diff_y [expr {abs($source_coordinate_y - $sink_coordinate_y)}]
        puts "    diff_x: $diff_x"
        puts "    diff_y: $diff_y"
    }
}
```

Tcl Tab Width: 8 Ln 52, Col 2 INS



features.txt
~/GLOBAL_ROUTE_LAB

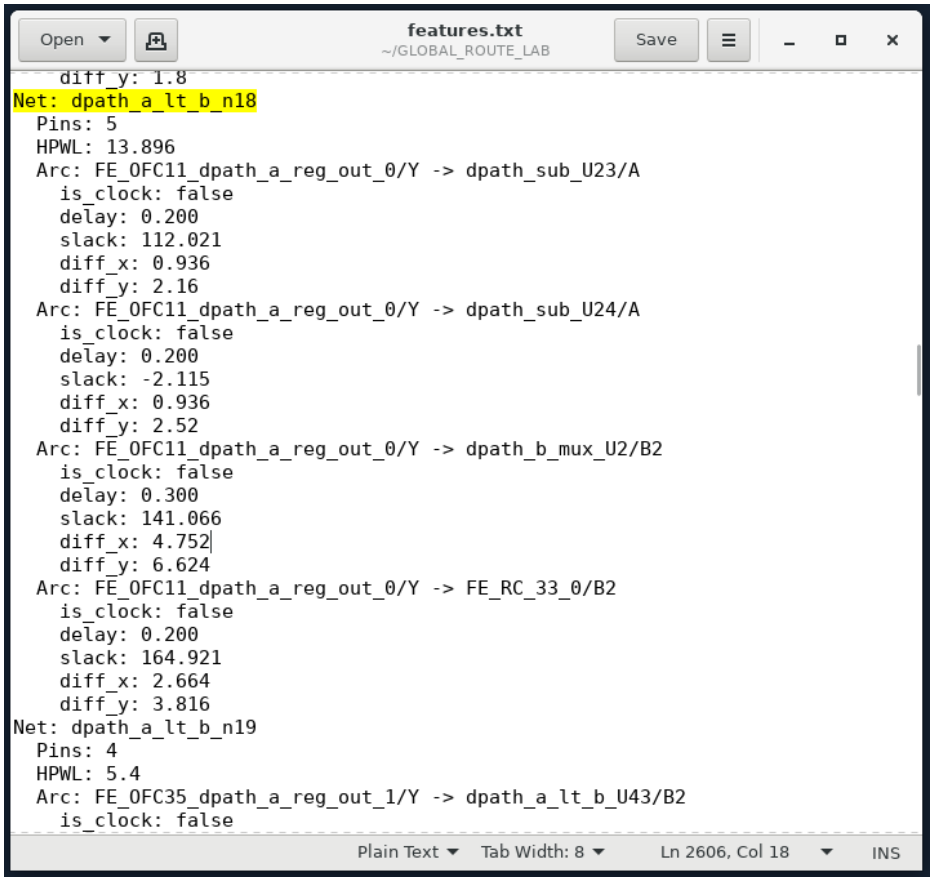
```
    diff_y: 1.8
Net: dpath_a_lt_b_n18
  Pins: 5
  HPWL: 13.896
  Arc: FE_OFC11_dpath_a_reg_out_0/Y -> dpath_sub_U23/A
    is_clock: false
    delay: 0.200
    slack: 112.021
    diff_x: 0.936
    diff_y: 2.16
  Arc: FE_OFC11_dpath_a_reg_out_0/Y -> dpath_sub_U24/A
    is_clock: false
    delay: 0.200
    slack: -2.115
    diff_x: 0.936
    diff_y: 2.52
  Arc: FE_OFC11_dpath_a_reg_out_0/Y -> dpath_b_mux_U2/B2
    is_clock: false
    delay: 0.300
    slack: 141.066
    diff_x: 4.752
    diff_y: 6.624
  Arc: FE_OFC11_dpath_a_reg_out_0/Y -> FE_RC_33_0/B2
    is_clock: false
    delay: 0.200
    slack: 164.921
    diff_x: 2.664
    diff_y: 3.816
Net: dpath_a_lt_b_n19
  Pins: 4
  HPWL: 5.4
  Arc: FE_OFC35_dpath_a_reg_out_1/Y -> dpath_a_lt_b_U43/B2
    is_clock: false
```

Plain Text Tab Width: 8 Ln 2606, Col 18 INS

[LAB2] Feature Extraction

```
innovus
```

```
source 4_route.enc
source 6_feature_extraction.tcl > features.txt
gedit features.txt
```



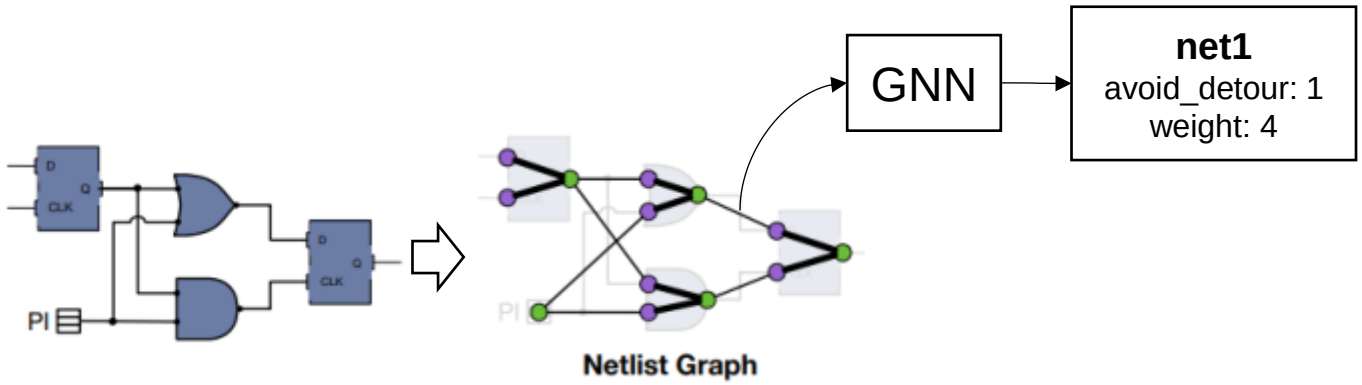
[LAB2] Feature Extraction

생각해볼 거리

→ Addtional features

→ Rule-based algorithms

→ ML-model designs



Thank you