ICV DRC Scalability on Samsung Designs

800+ Core

KYEUNGKEUN CHOE, Staff Engineer
Samsung Foundry

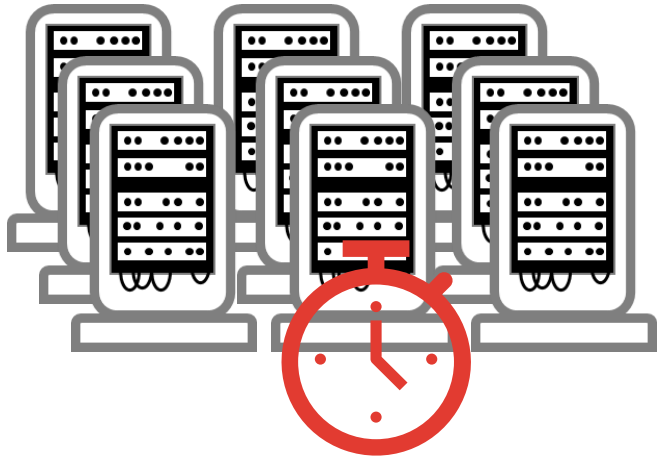
Index

01 Abstract

SAMSUNG SYNOPSYS®

Improving linearity of engine performance for Over-night (One day) Physical Verification

ICV Distributed Processing Capability until 800+ Core on Samsung's Advanced Nodes



- Samsung in collaboration with Synopsys and its distributed processing capabilities helped bring down runtimes from over 3 Days to 13 Hours.
- Optimized runsets and specific engine enhancements achieved the goal to keep linearity of performance over 800 cores.

02 Motivations and Evaluation details

- Run time saturation with multiple cores
  - In general, DRC runs on SoC designs uses 200 cores due to some bottle necks of run time performance.
- Methods for reducing overall time : Split runs into multi-options are not enough
  - Modules along design levels (BEOL / FEOL)
  - Connectivity runs (Antenna / ESD / HV)
  - Density after fill insertion
  - Selective checking with specific design rules
- Main goal for collaboration is to keep scalability of engine performance
- Specifications of project
  - Test items

ID	Design	Process	Area (mm^2)	DB Size
1	**Chip_1**	4nm	283	44 GB
2	**Chip_2**	4nm	115	21 GB
3	**Chip_3**	3nm	161	82GB

  - ICV Version : 2023.12*
  - Machine : Intel(R) Xeon(R) CPU @ 3.40GHz (Memory 1TB), Intel(R) Xeon(R) CPU @ 3.50GHz (Memory: 750 GB)

Improvements

Engine Enhancement

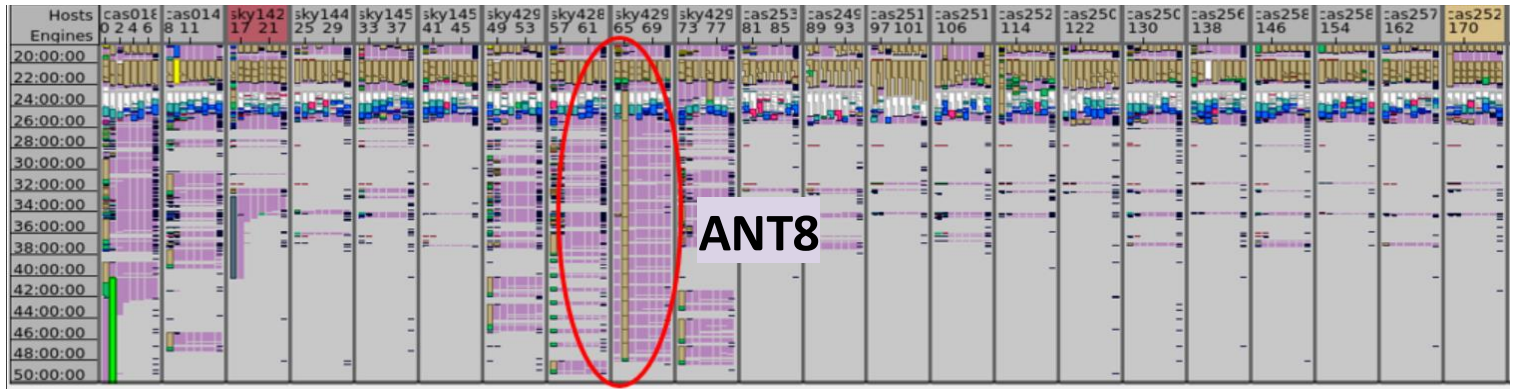
a. Re-Scheduling executions

b. Tool performance enhancement

04-A Improvements

Engine Enhancements

- Scheduling
  - Due to the machine resource limitation, the connect function in engine does NOT parallelize ANT8 design rule check.

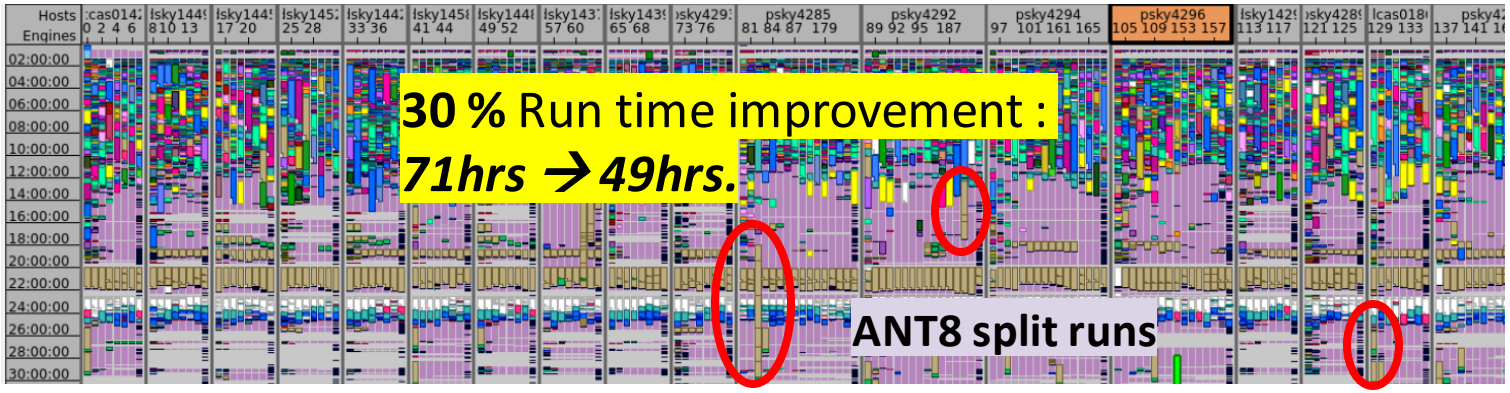


Idle machine

Mem. usage

cmd. execution

  - By updated engine, all connection buildings & checks of antenna are able to run in parallel



04-B Improvements

Engine Enhancements

- Hierarchy optimization
  - hierarchy optimization commands are executed on a single machine
    - the total number of CPUs does not affect the runtime.
    - Negative effect on scalability if it takes long time
  - Optimization of storing cell properties save most of hierarchy optimization time

Idle machine
Mem. usage
cmd. execution
cmd. Hier. Optm.

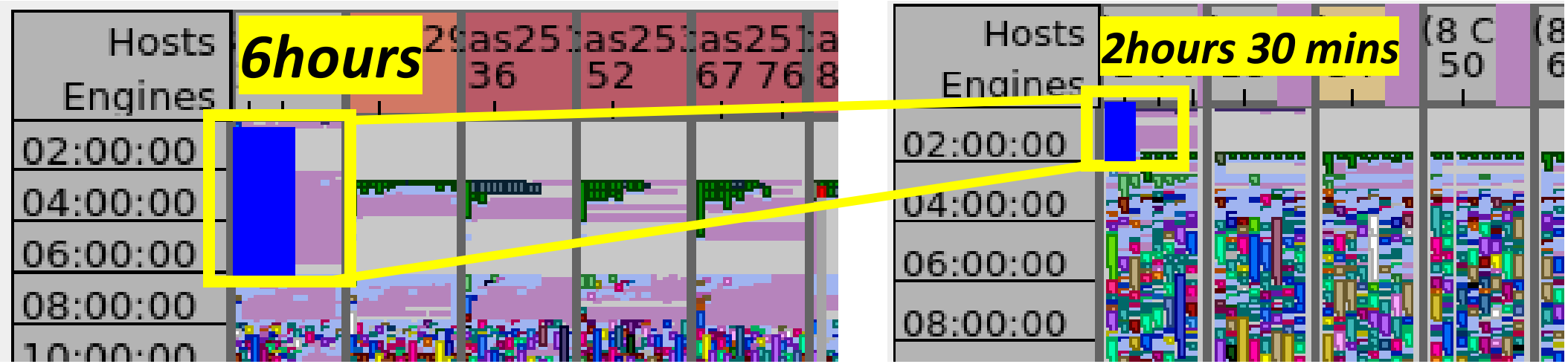


Fig. MRI comparison between two engines along Chip2 run time

04-C Improvements

Engine Enhancements

- High voltage (HV) command
  - “property_to_net()” command optimization : pull_down property enables low memory usage and improve run time cost

Idle machine

Mem. usage

cmd. execution

Fig. MRI of original run with problem (CHIP_1)

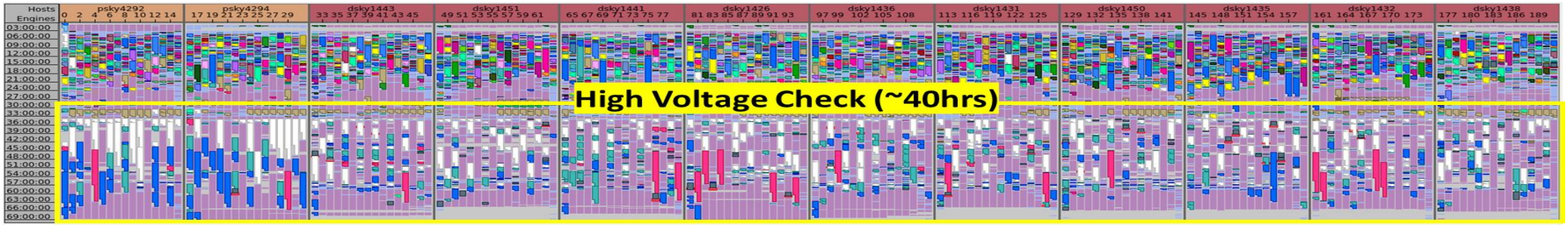
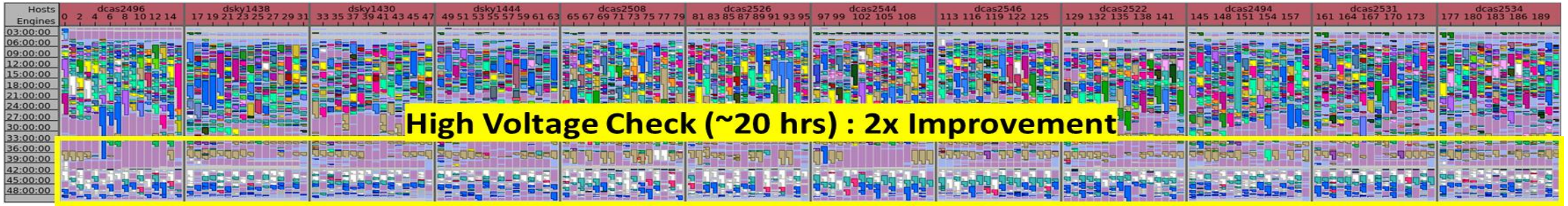


Fig. MRI of fixed run (CHIP_1)



04-D Improvements

Engine Enhancements

- Dimensional command : external*, size_outside*
  - With engine improvement, commands takes ***27x*** lesser memory.

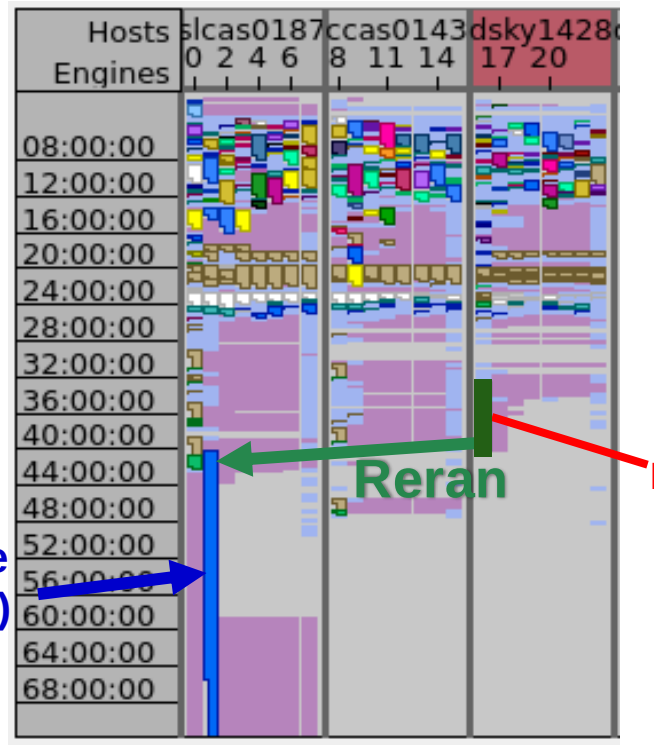
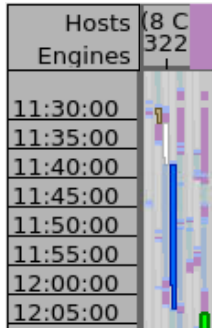


Fig. MRI of original run with problem (CHIP_1)

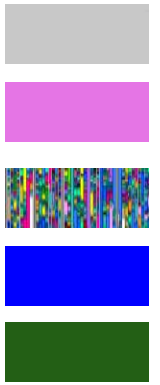


Runtime improved by ~60x

Runtime : 24 mins

Memory usage : 28 GB

Fig. MRI of original run with fix (CHIP_1)



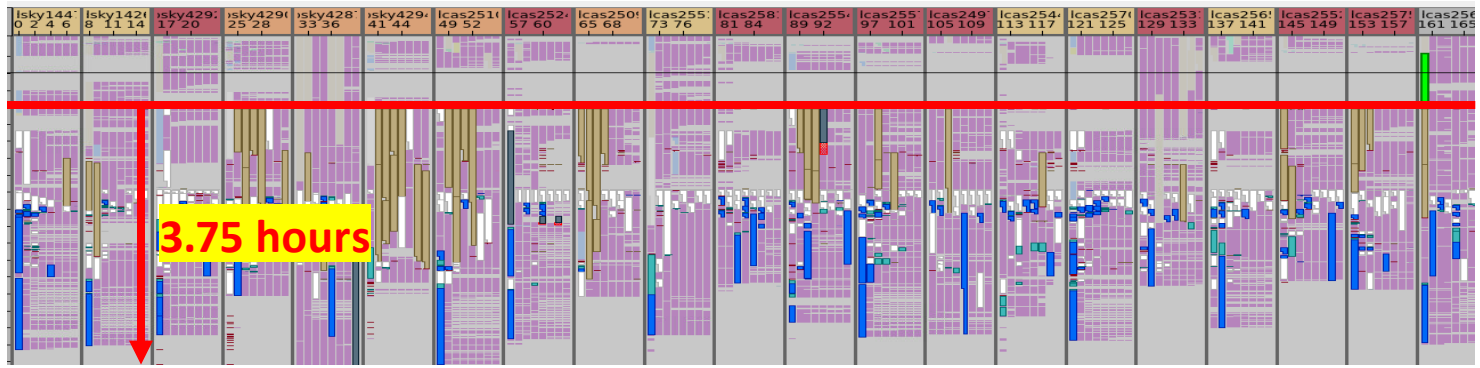
04-D Improvements

Engine Enhancements

- Connect command : engine optimization on duplicated connect commands

Fig. MRI of original run with problem (CHIP_3)

Fig. MRI of original run with fix (CHIP_3)



Idle machine

Mem. usage

cmd. execution

cmd. CONNECT

cmd. deferred

Parallel execution of connect commands causes memory issues and commands are deferred.

Duplicated connect commands are removed, and runtime improved by **30%.**

Improvements

Runset optimizations

a. Re-Scheduling executions

b. Tool performance enhancement

05-A Improvements

Runset Optimizations

- drc_features* (specialized function) command optimization
  - Length check
    - drc_features_edge is replaced to not_rectangles

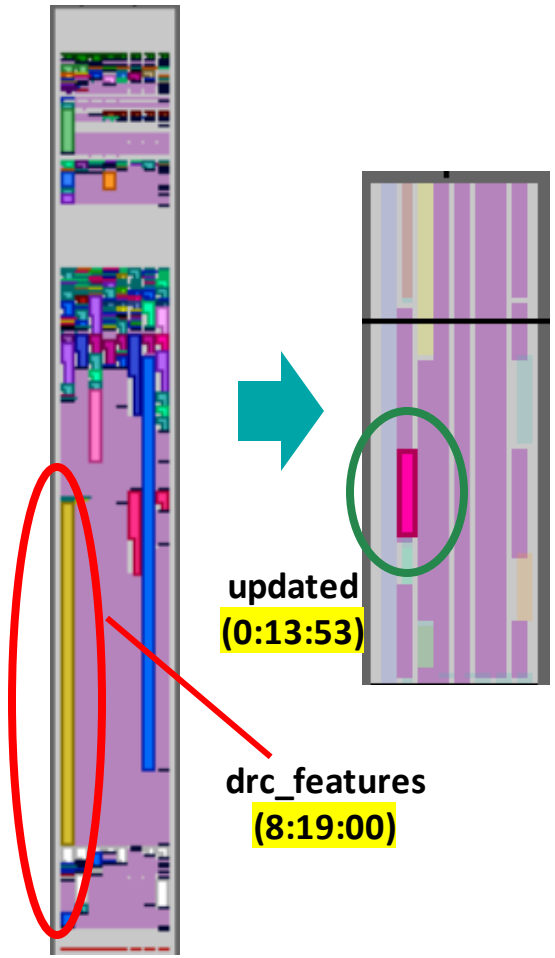
```
LONGER_EDGE = drc_features_edge( primary_layer = EDG, secondary_layers = {
"layer2" => POLYGON }, output_from_layer = EDG, drc_function = func1 );
length_edge( LONGER_EDGE, < 0.072 );
```

```
not_rectangles( sLAY1, orientation = ORTHOGONAL, sides = { >= 0.072 } );
```

  - Oriented check
    - drc_features is alternated to aspect_ratio

```
drc_features( primary_layer = LAY1, secondary_layers = { "layer2" => EDG_1, "layer3"
=> EDG_2}, output_from_layer = LAY1, drc_function = func2);
```

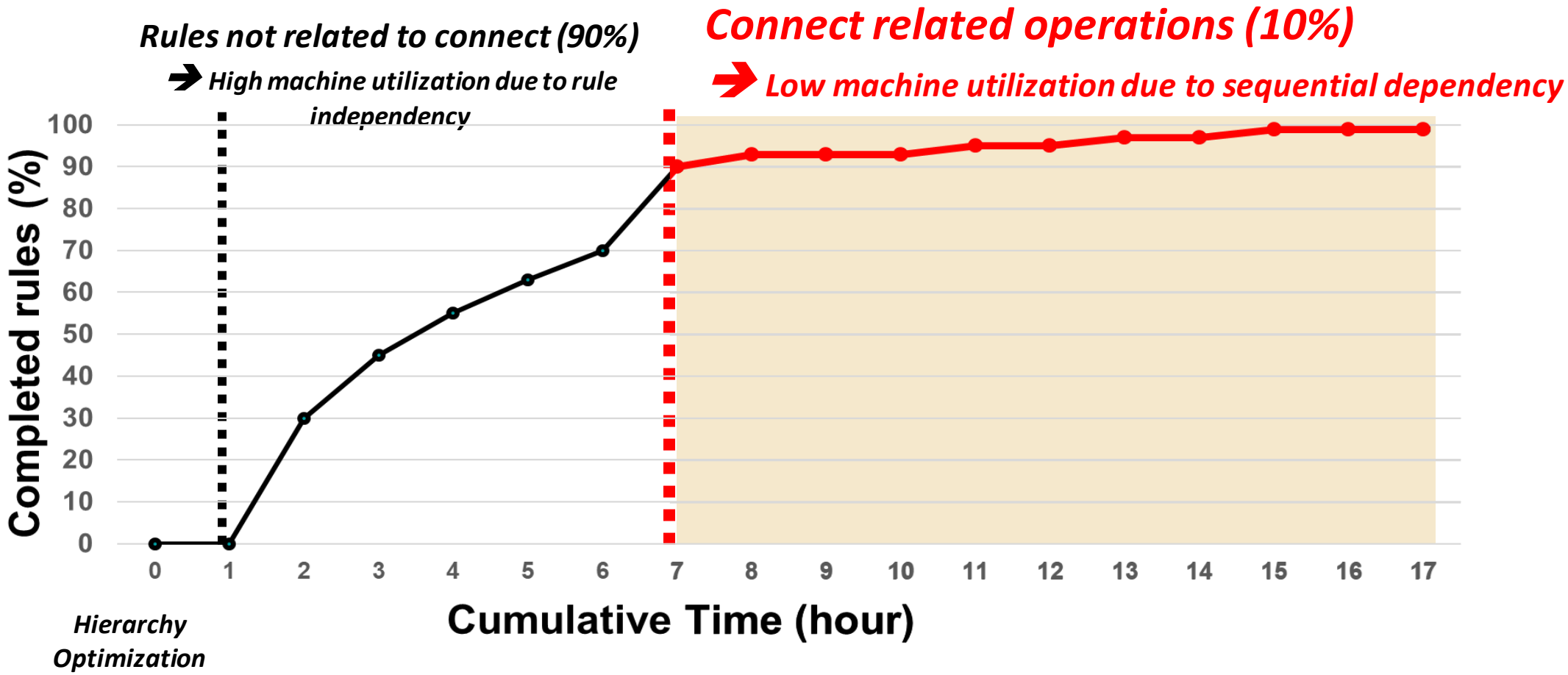
```
aspect_ratio( LAY1, ratio = > 1, orientation = ORTHOGONAL, direction = X_BY_Y);
```



05-B Improvements

Runset Optimizations

- Portion of connect related design rule during whole run time



05-B Improvements

Runset Optimizations

- Split Main Connect as parallel

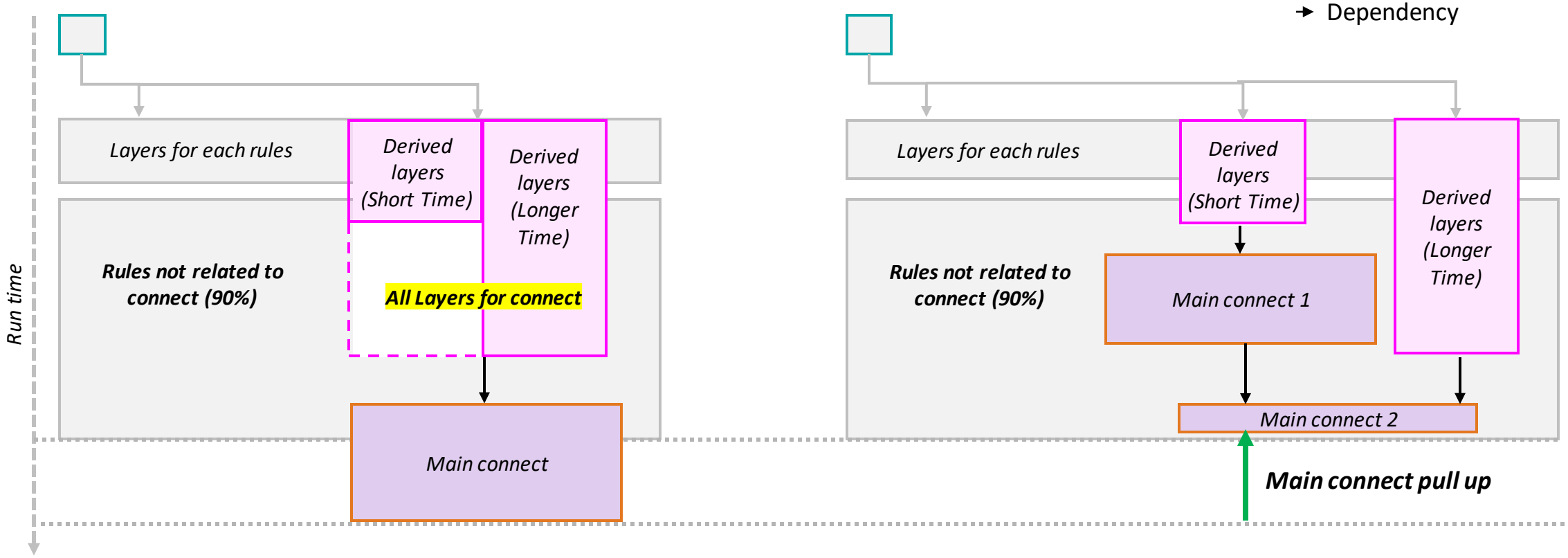


Fig. Original connect

Fig. Connect Split

05-B Improvements

Runset Optimizations

- Redundant Connect Removal
  - Cumulative runtime is improved by 28%

Connect
Rules related to connect
Dependency

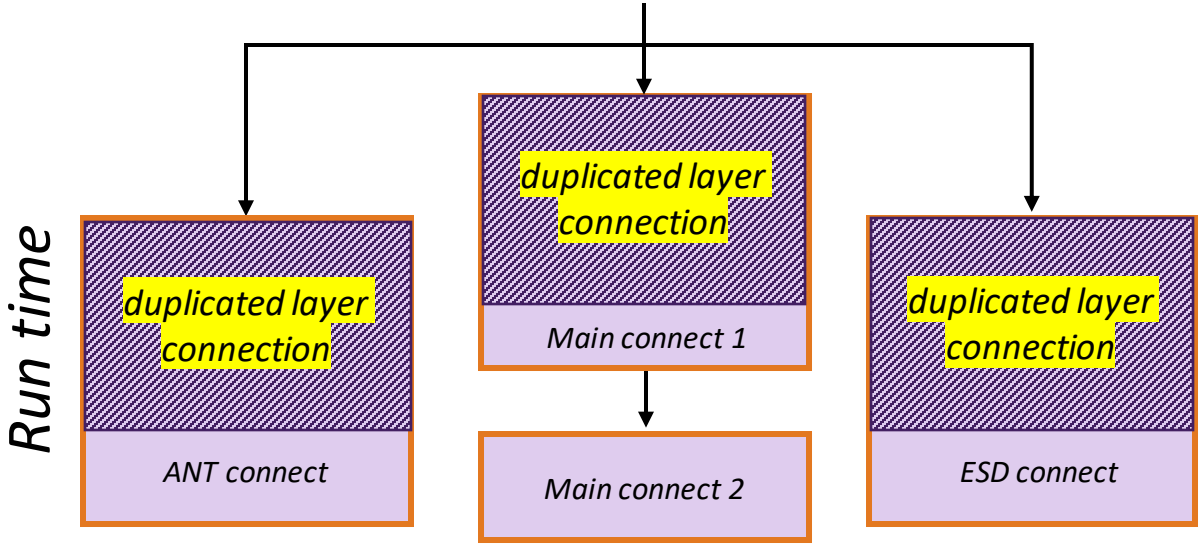


Fig. Original connect

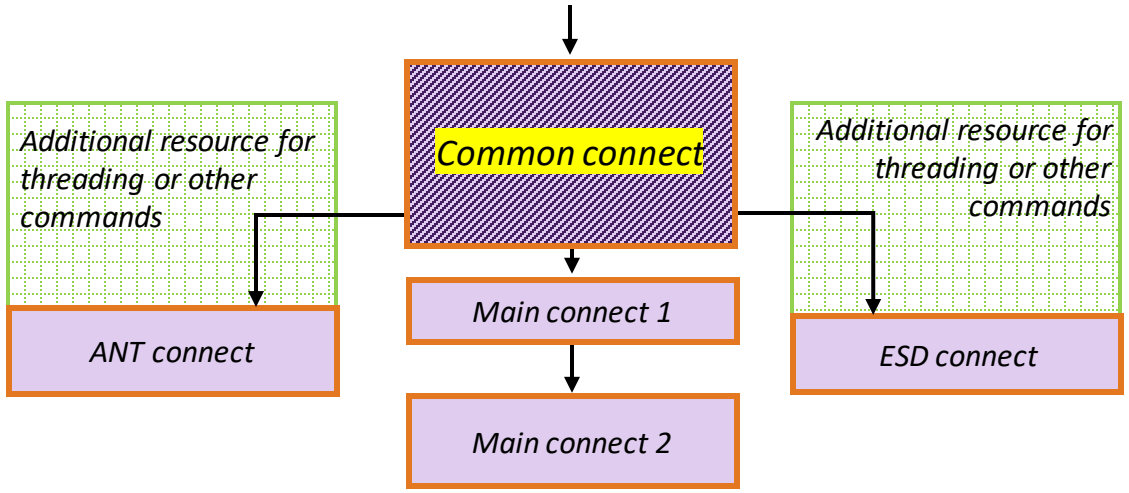


Fig. Merged connect

05-B Improvements

Runset Optimizations

- Detaching series processing of Connect

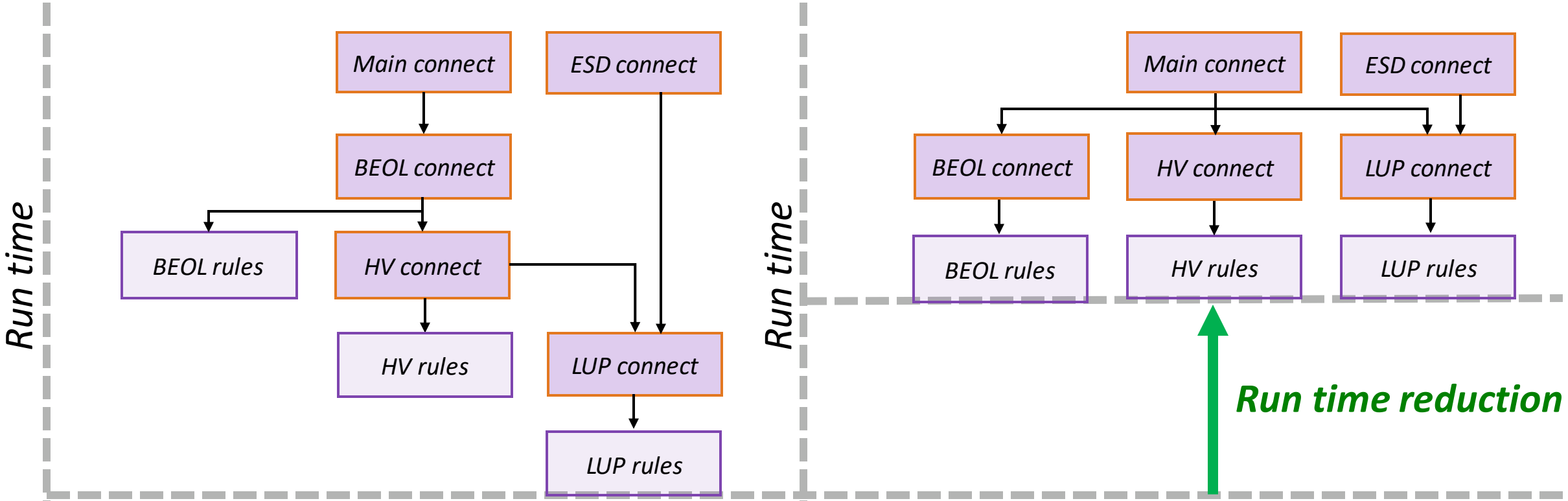


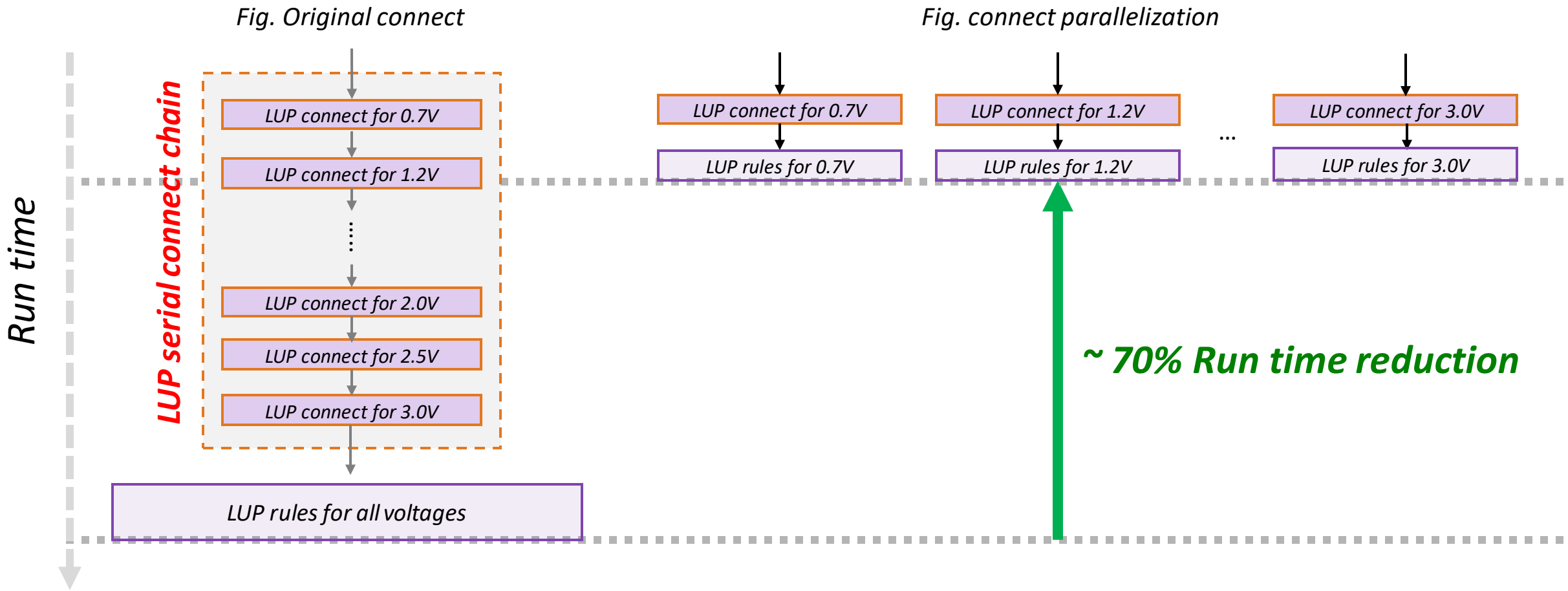
Fig. Original connect (connects in serial)

Fig. Break dependency of Connect

05-B Improvements

Runset Optimizations

- LUP Connect Parallelization



05-B Improvements

Runset Optimizations

- Result of Connection dependency update – Run time from 17:18 >> 12:48 (~600 Core)
  - Connect commands are merged and executed earlier than original DRC
  - Idle machines are mostly occupied with updated DRC
  - Broken red line indicates time point of 90% of Design rules are finished

Fig. MRI of Chip_1 Run with original DRC

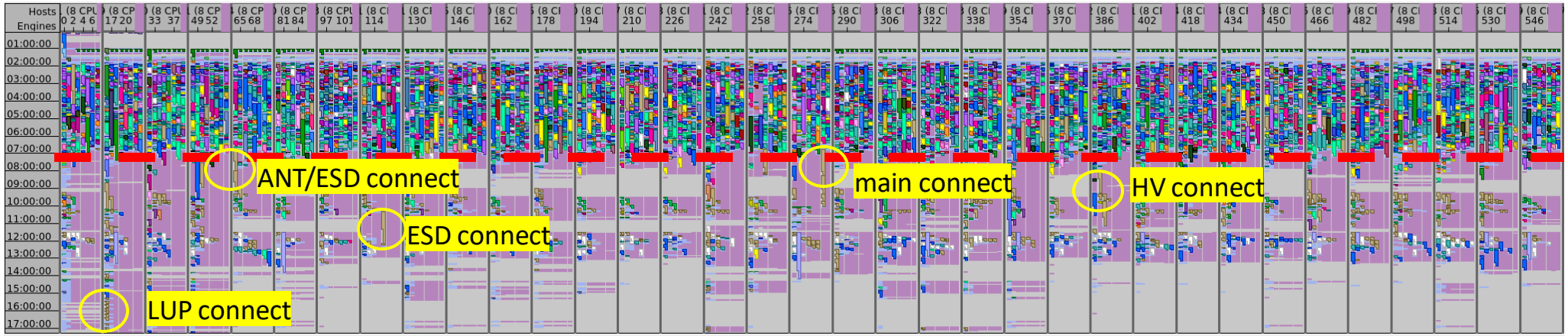
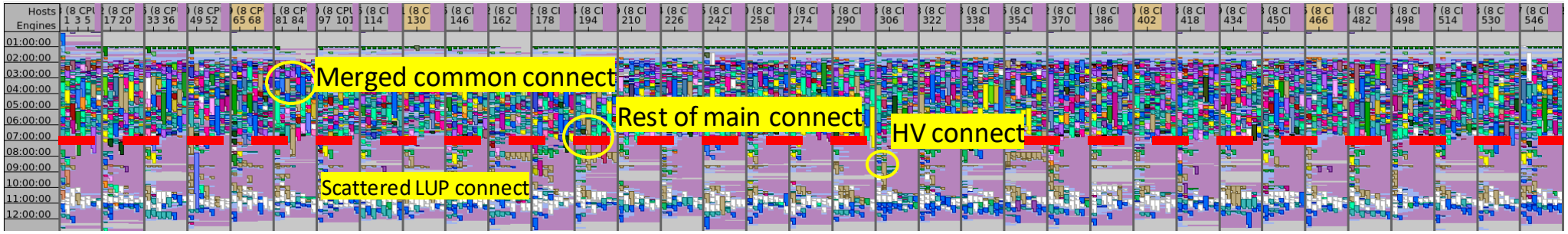


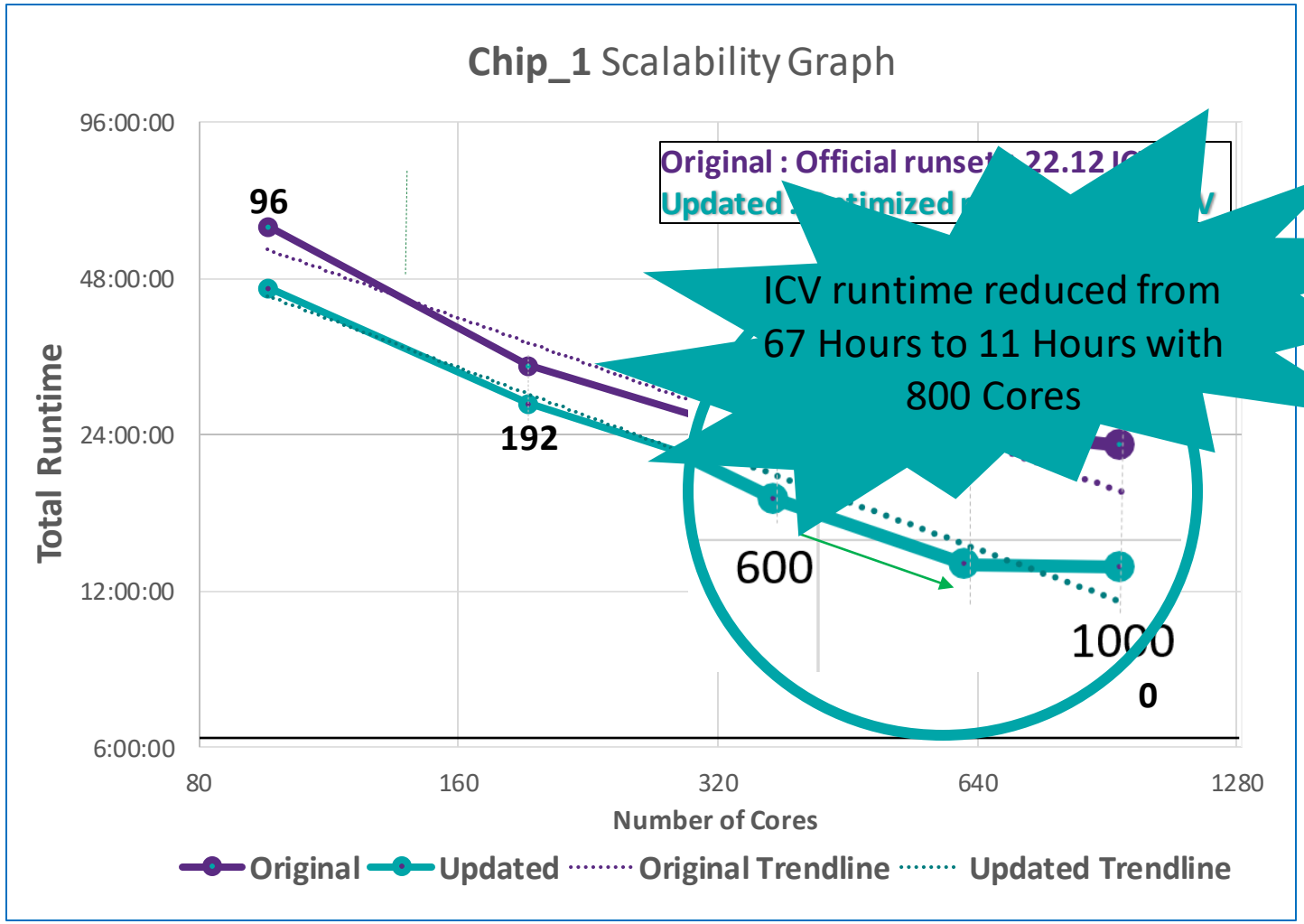
Fig. MRI of Chip_1 Run with Optimized DRC



Results

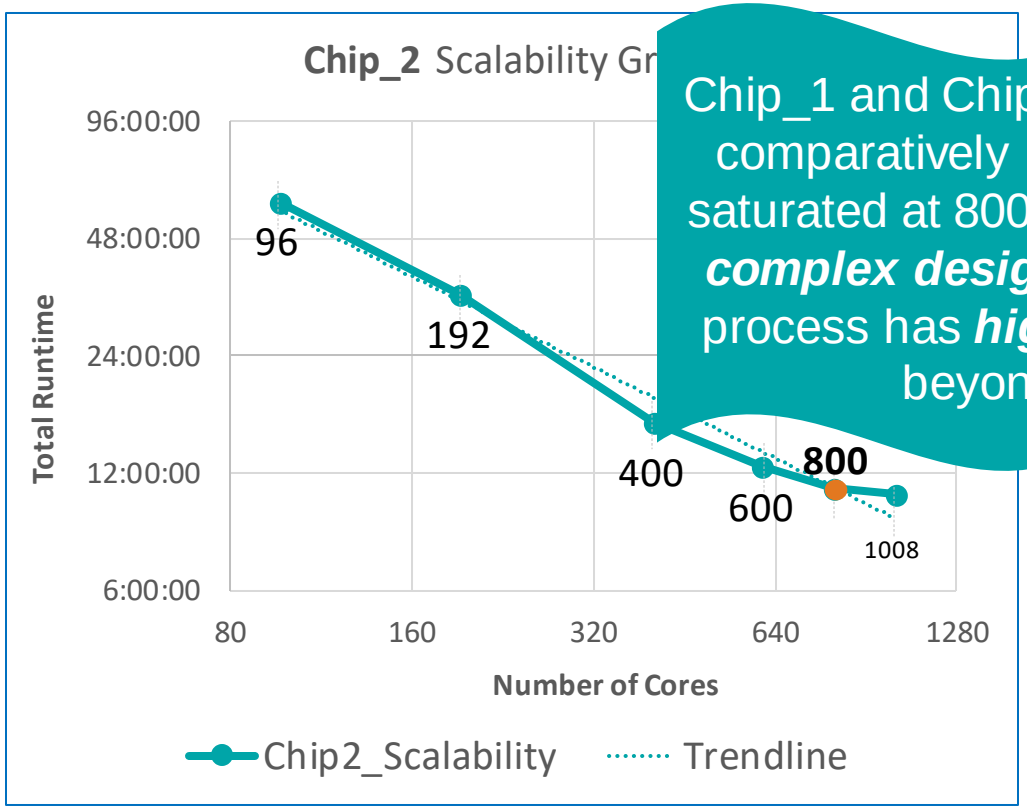
Review and summary

06 Results

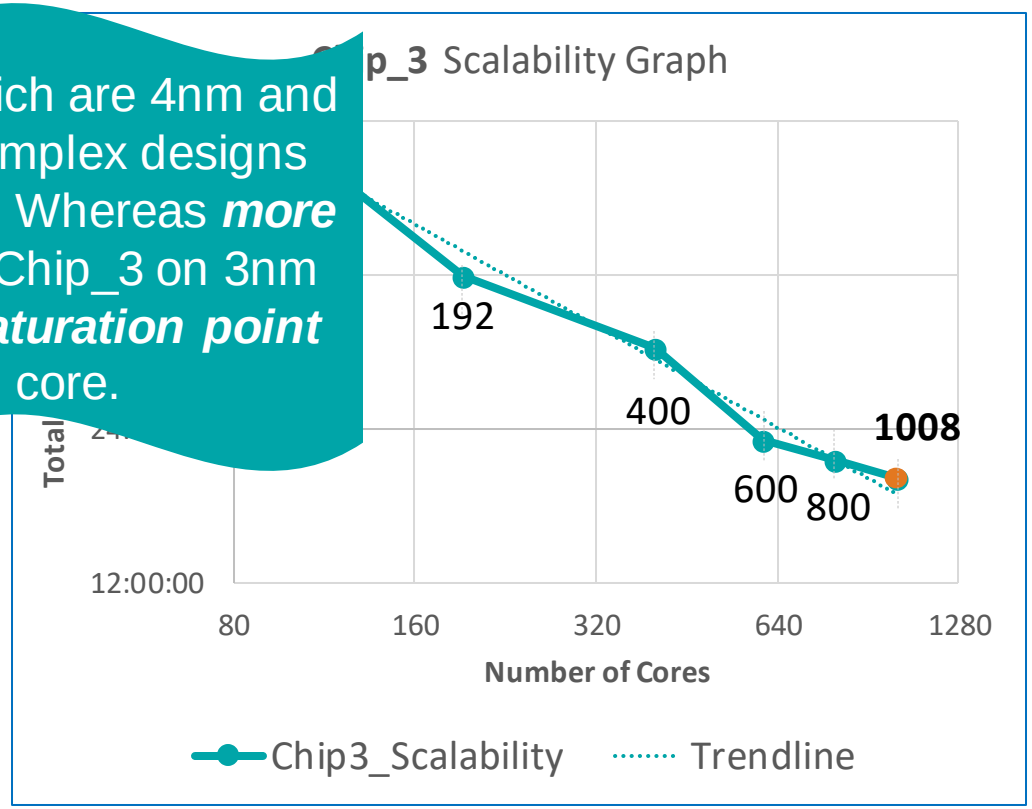


- As the graph shows in Original run(purple line), runtime decrease started to lag from 600 cores
- Improvement of building connection by multiple threading, evidently green line has better linearity along 600 to 800 cores rather than Original one.
- Overnight run target time achieved with ICV engine and run-set optimization.
- In this project, ICV achieved run time reduction linearity until 800 Cores without saturation on 4nm Large size Design.

06 Results



Chip_1 and Chip_2 which are 4nm and comparatively less complex designs saturated at 800 cores. Whereas ***more complex design*** like Chip_3 on 3nm process has ***higher saturation point*** beyond 1000 core.



- Based on all updates, chip2 run saturates at 800 Cores.
- 62 hrs total run time with 96 cores was reduced to 11hrs with 800 cores.

- Chip_3 run time keep a linearity of reduction until 1,008 cores.
- About **70 hours** runtime reductions with 1008 cores is achieved on Chip_3 compare to 96 cores. (95 > 13 hours)

07 Future work

- Scalability beyond 1000 cores for large designs, for better turnaround time (TAT).
- Extend scalability enhancement project to other PDK components (LVS, FILL…)
- Apply optimized point of run-set on the ICV engine for run time performance
- New command application on DRC: **DP connect**
  - Utilize CPU and Memory resource on multiple hosts for long pole Connect commands
  - Currently this is available only in LVS and planned to be part of DRC in 2024.09.

SAMSUNG snug

THANK YOU

YOUR
INNOVATION
YOUR
COMMUNITY