ECO in the Age of AI: Advancing Signoff Convergence with PrimeClosure™

Presenter : Tee, Luck Kuan
Co-Authors : Walimbe, Atul S & Kewale, Dhananjay V
(Intel Graphics CAD Team)

Objective

Share the signoff ECO
(Engineering Change Order)
convergence results (timing, power and runtime)
observed with
the next-generation ECO solution,
PrimeClosure™ (PC) as compared to PTECO™

Agenda

- Post-Route/ECO Optimization
- Introduction of Key PrimeClosure™ Technologies
  1. SMSA: **S**ingle-Machine **M**ulti-**S**cenario **A**nalysis
  2. Clock-Surgery
  3. AI-driven ECO
  4. Hierachical ECO

- Conventional Signoff ECO Convergence Flow
- Comparison Results
  1. PTECO™ vs PrimeClosure™ (Timing, Routing, Power, and Runtime)
  2. With Clock-Surgery Enabled
  3. AI-driven ECO (Leakage Power Saving)
- Hierarchical ECO Flow at Section/Full-Chip Level (w/ Hyperscale)

- Future Technologies & Enhancements
- Summary

Intra-Block Design Closure

Subsystem/Top-Level: Inter-Block/Section Design Closure

Post-Route/ECO Optimization (PTECO™ & Tweaker™)

snug

Essential and critical last-mile optimization to meet PPA & tight project schedule

i. Achieve optimal power.
ii. Address timing and logical DRC (max-cap & max-trans).
iii. Meet final signoff goals (>20 vs ~6 dominated PVTs in P&R).
iv. Reconverge the design after intercepting late functional ECOs, newer design collaterals/PDK, etc.

		PTECO	Tweaker
1	Data Preparation Efforts	Low	Medium
2	Physical Aware (PA)	Yes	Yes+ (Lesser cells displacement)
3	Timing Correlation	100%	~95%
4	Machine Memory	Medium	Low
5	Distributed Machines Feature	Yes	Yes- (Can be split/partition based-on viols)
6	Hierarchical ECO Convergence (Section/Full-chip)	Yes (Hyper-Scale PTECO)	Yes+ (Physical Aware & lesser memory)
7	Debugging GUI	Yes	Yes+
8	Multi Corners/Scenarios Support	Yes	Yes+ (Single Machine & Capable of handing more corners, 100+)

Few Questions?

1. Are you encountering challenges with ECO design convergence/closure?

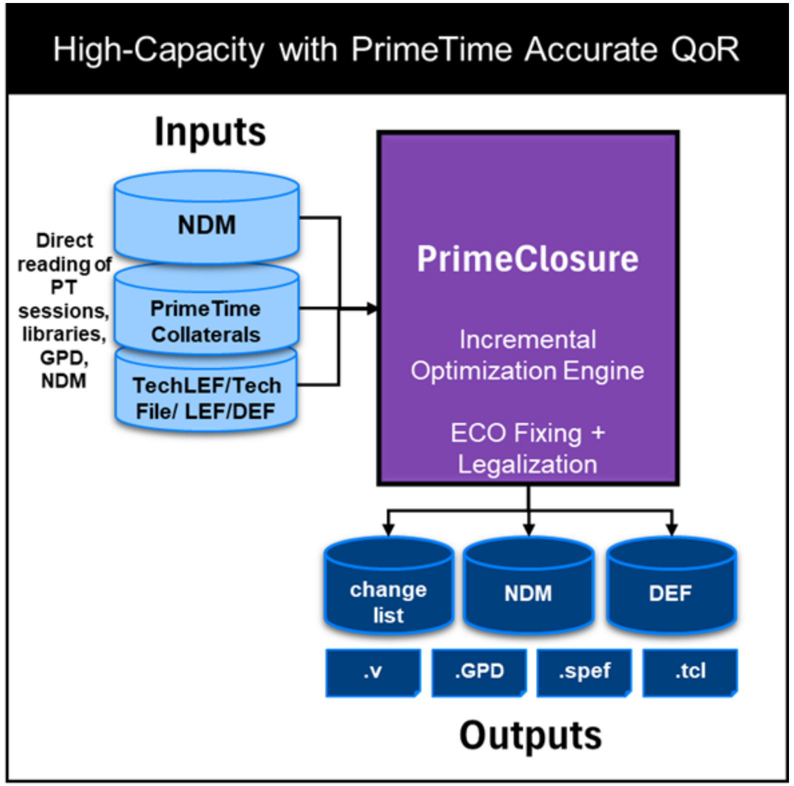
2. Are you grappling between PTECO™ and Tweaker™ ECO while also supporting both tools?

PrimeClosure™: Next-Generation ECO (Integrated the strengths of PTECO™ and Tweaker™ ECO + New Technologies)

Introduction of Key PrimeClosure™ Technologies

1. SMSA: **S**ingle-**M**achine **M**ulti-**S**cenario **A**nalysis

- **Integrated** ECO and STA (Static Timing Analysis) in one cockpit to reduce iterations and to produce better QoR (Quality of Results) by utilizing the full set of PrimeClosure ECO fixing technologies.

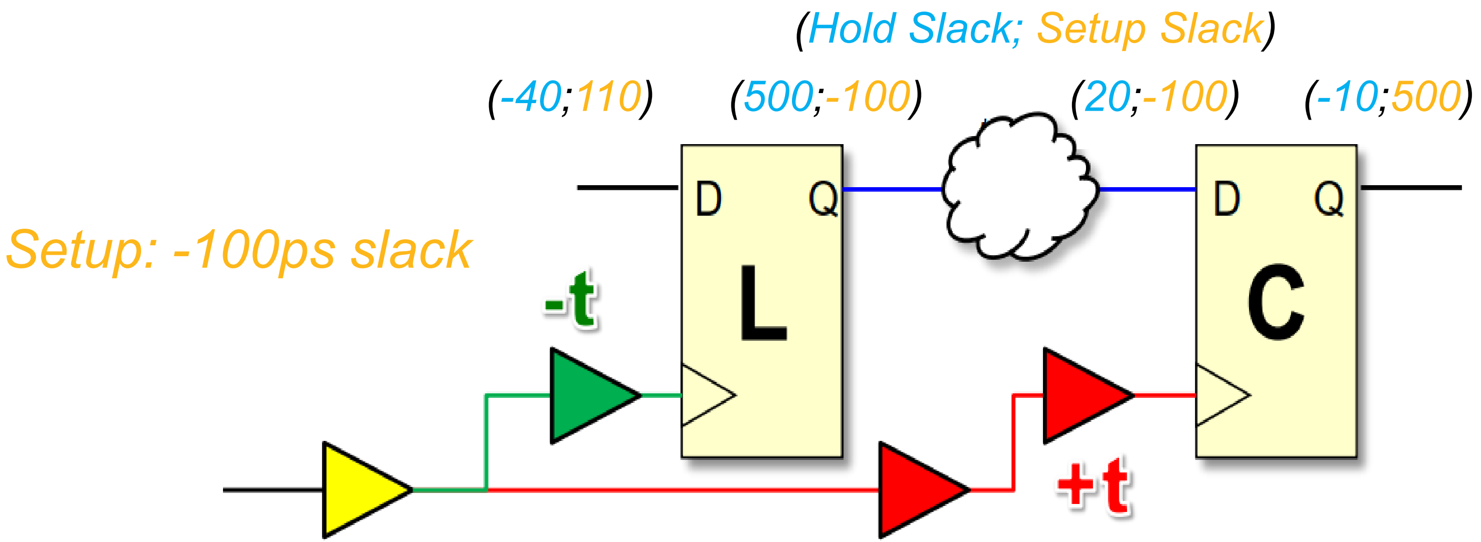


- **PrimeTime Accurate QoR** through signoff PT-delay calculation and an automated what-if co-optimization flow.
- **Native Cells Legalizer** to reduce ECO iterations by improving pre and post implementation correlation, as well as minimizing design perturbation.
- **High-Capacity and Less Compute Resources** - Capable of optimizing 100+ scenarios with fewer machines, and subsystem/top-level.

Introduction of Key PrimeClosure™ Technologies

2. Clock-Surgery

- Address setup timing speed-paths with useful clock skew.



Fix at the starting point (Green: Pull-in):
Decrease clock delay by upsizing and/or bypass
Launch Flop, D(Setup) = 110 > 100
Launch Flop, Q(Hold) = 500 > 100

Fix at the endpoint (Red: Push-out):
Increase clock delay by downsizing and/or inserting
Capture Flop, Q(Setup) = 500 > 100
Capture Flop, D(Hold) = 20 < 100

New Options/Enhancements

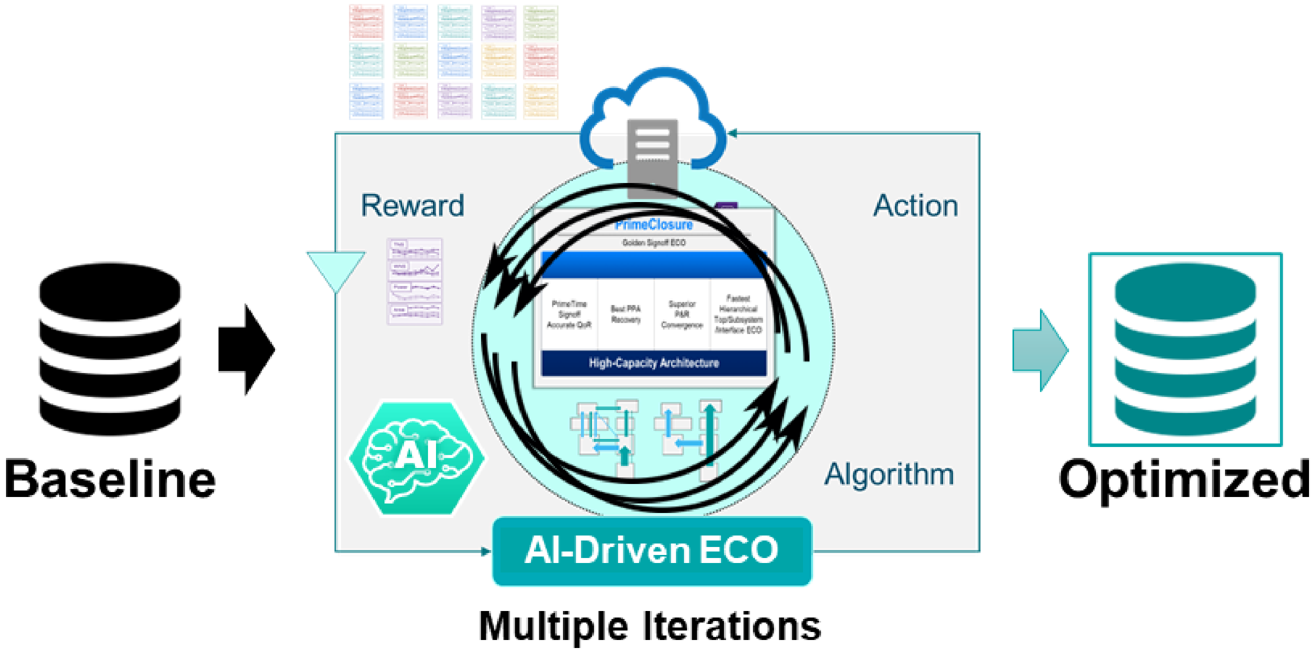
- Restrict the maximum clock level (*-max_level_in_clock 5*)
- Exclude/Ignore IO related endpoints (*-ignore_boundary_flops 1*)

Introduction of Key PrimeClosure™ Technologies

3. AI-driven ECO – Early Version

- Integrated DSO.ai (AI-driven Design Space Optimization) technology, which allows users to provide different ECO parameter values (built-in Permutons) to achieve better PPA/design convergence results.

AI-driven ECO for Last-Mile Closure



Original Timing Fixing Settings

```
set slk_auto_sizing_max_shift_distance 4
set slk_fix_hold_watch_driving_pin_setup_slack false
set slk_fix_hold_watch_driving_pin_hold_slack false
set slk_fix_hold_watch_driving_pin_slack false
```

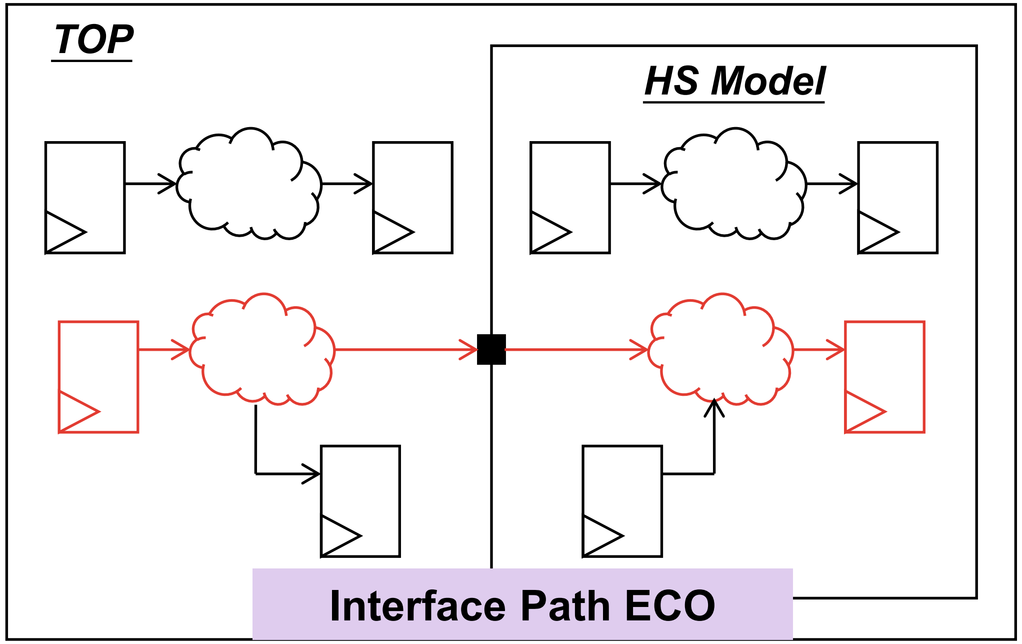
Built-In Permutons

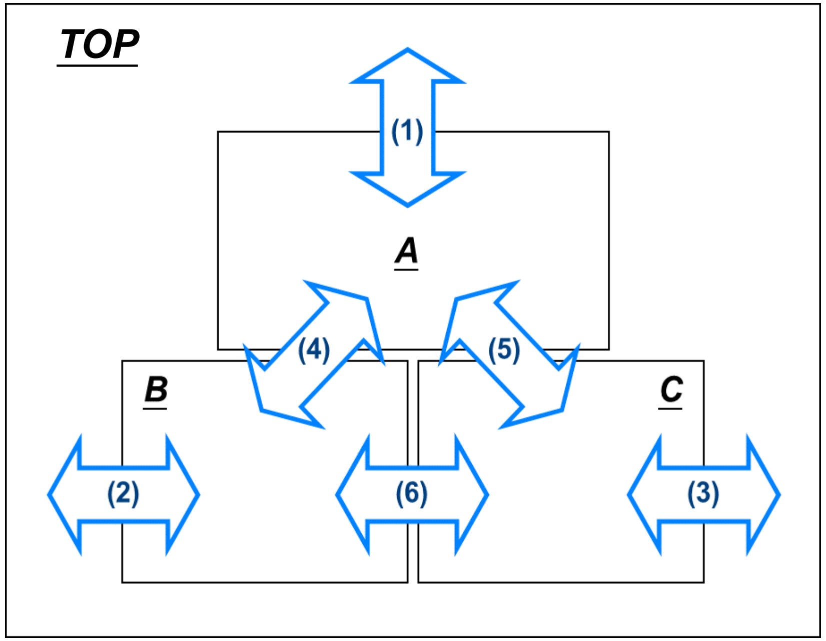
```
set slk_auto_sizing_max_shift_distance [0, 4, 8]
set slk_fix_hold_watch_driving_pin_setup_slack ---
set slk_fix_hold_watch_driving_pin_hold_slack ---
set slk_fix_hold_watch_driving_pin_slack ---
```

Introduction of Key PrimeClosure™ Technologies

4. Hierarchical ECO

- Primarily focusing on interface paths only to accelerate inter-block closure for Section-level designs and inter-section closure for Full-chip designs, respectively.
- Support both Hyperscale (HS) and Flat Approaches.





Post-Route/ECO Optimization at Block Level

(Intra-Block Design Closure)

Data Exchange from PrimeTime™ to PrimeClosure™

PrimeTime™
- STA (save_session)
- write_eco_session

↓

PrimeClosure™
- read_eco_session
- ECO

```
# Generate sessions for PT and PC
save_session ./pt_session_scen1

write_eco_session -include {smsa_data}  \
  -smsa_data_type {setup hold max_transition max_capacitance  \
      max_fanout power drc_max_transition drc_max_capacitance pin_slew}  \
  -link_session $sh_launch_dir/pt_session_scen1  \
  -smsa_data_format binary \
  -smsa_pba_mode exhaustive <pre_eco_session_scen1>
```

Reuse/Link the existing STA session to save disk-space

Design 1 (~1.3M Inst)	PT Session	PC Session	PC Session (-link_session)
PVT 1	6	7.9	1.9
PVT 2	6	7.9	1.9
PVT 3	7.3	8.8	1.5
…	…	…	…
PVT 10	5.9	7.3	1.4
PVT 11	5.9	7.3	1.4
PVT 12	5.9	7.3	1.4
Total	73.9	92.4	18.5
		125%	25%

Comparison of PT and PC Save Session File Sizes (GB)

PrimeClosure™ Recipe

pc_script.tcl

```
set multi_scenario_working_directory PC_DMSA; set_host_options -num_processes 12
set_technology –node <N>

# Additional settings from set_eco_options can be included here
read_physical_data  -tech $tech_lef  -lef $lef_files \
                    -physical_icc2_lib design.ndm  -physical_icc2_blocks pre_eco

read_eco_session pre_eco_session
start_eco -mode smsa
fix_eco_drc
fix_eco_timing -type setup
# fix_clock_setup_endpoint; fix_eco_timing -type setup
fix_eco_timing -type hold

fix_eco_power
start_eco -mode dmsa; report_global_timing;
start_eco -mode smsa
fix_eco_drc -type max_transition -verbose -methods {recovery}
fix_eco_timing -type setup -methods {recovery}
ecotclout -icc2 <eco_changes.tcl>
report_eco_summary -summary

start_eco –mode dmsa
# Assess Timing with standard PrimeTime reports within same shell
report_global_timing; report_constraint; report_timing
```

```
pc_shell –f pc_script.tcl –output_log ./logs/run.log
```

- Configure Machine Resources for STA
- Set Technology Node
- Read Physical Data
- Read pre-ECO sessions
- ECO Fixing (LDRC, Timing, Clock-Surgery, and/or Power Recovery)
- Changelist
- PrimeTime What-if

An Example of PrimeClosure™ Log File

report_eco_summary -summary

```
Group : all_group            Original                   Current
                             setup        hold          setup        hold
---------------------------------------------------------------------------
       Critical Path Slack:  -0.26        -0.02         -0.08        -0.00
      Total Negative Slack:  -1719.99     -27.67        -194.12      -0.00
     No. of Violating Paths: 121578       10947         10254        13
  TNS of Violating Endpoints: -82.92      -7.66         -4.36        -0.00
  No. of Violating Endpoints: 13231       2613          216          6
---------------------------------------------------------------------------
```

Setup

Hold

```
#------------------------------------------------------------------
# autofix detail log summary:
#------------------------------------------------------------------
# B004 Blocked by Timing Window (Setup)(1)
# B010 Improved slack < min improved slack(75)
# B012 Blocked by sizing non-STD cell(8)
# B025 Blocked by Timing Window (Setup) (Input Pin)(5)
# B051 Blocked by don't use setting(1787)
# B073 Blocked by twf clock pin(34)
# B086 Blocked by fix setup/cons without sizing down(1718)
# B126 Blocked by auto-sizing area ratio(1767)
#------------------------------------------------------------------
```

*Blocking Codes for Unresolvable **Setup** Viols*

```
#------------------------------------------------------------------
# autofix detail log summary:
#------------------------------------------------------------------
# B003 Blocked by no setup margin (twf)(1)
# B004 Blocked by Timing Window (Setup)(3)
# B006 Blocked by Driving Timing Window (Setup)(1)
#------------------------------------------------------------------
```

*Blocking Codes for Unresolvable **Hold** Viols*

An Example of ECO Implementation Log File

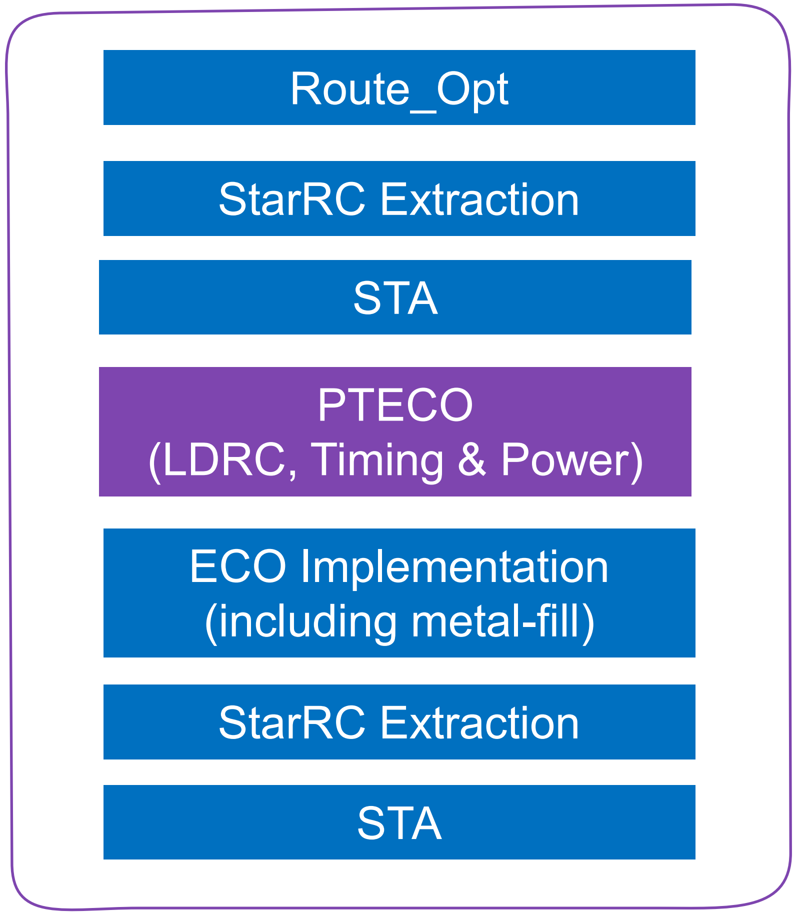
source <eco_changes.tcl>

```
ECO: === Summary of dropped ECOs due to target pre-check          ===
ECO: === Enabled by PrimeClosure variable (eco_tcl_pre_check_target)  ===
ECO:                          accept         drop        total
ECO: ---------------------------------------------------------------
ECO:     insertion              2000            0         2000
ECO:     deletion                 23            0           23
ECO:     dummy_load                0            0            0
ECO:     sizing               193872            0       193872
ECO:     by_pass                   0            0            0
ECO:     pin_swap                  0            0            0
ECO:     other_eco                 0            0            0
```

legalize_placement -post_route -incremental

```
        Number of cells moved: 2180 (0.16%)   Orientation changes only: 211
    Average cell displacement: 0.0003 um (AW: 0.0003 um = 0.0021 rh)
        Max cell displacement: 1.9433 um = 13.5902 rh
Number of large displacements: 215
```

Conventional Signoff ECO Convergence Flow



Highly leveraging the fully automated signoff ECO convergence flow to address design violations and minimize manual ECO fixing efforts in accelerating design closure.

fix_eco_drc
fix_eco_timing -type setup
fix_eco_timing -type hold
fix_eco_power

Goal: Evaluate potential to accelerate design closure further with PrimeClosure™

Comparison Results

PTECO™ vs PrimeClosure™ (SMSA & U-2022.12-SP5-2)

Timing (Intra-Block Only + post-ECO)

Block	Advanced Node	Total Setup TNS (RouteOpt)	Total Setup TNS (PTECO)	Total Setup TNS (PC)	Total Hold Viols (PTECO)	Total Hold Viols (PC)
Design 1	N	-25.4	-8.4	-0.36	15	43
Design 2	N	-156.9	-7.83	-8.9	17	94
Design 3	N	-28.3	-0.37	-0.24	114	217
Average Diff				**43%**		**-208**
Design 4	N+1	-74.79	-1.06	-0.33	78	25
Design 5	N+1	-583.58	-20.28	-3.24	715	409
Design 6	N+1	-1992	-56.26	-35.25	1288	790
Average Diff				**50%**		**857**

Routing

Block	Total DRC (PTECO)	Total DRC (PC)	Total Short (PTECO)	Total Short (PC)
Design 1	227	213	17	17
Design 2	1086	852	17	20
Design 3	380	399	27	20
Average Diff		**229**		**-3**
Design 4	52	50	11	12
Design 5	80	97	19	23
Design 6	4960	4752	2289	2161
Average Diff		**193**		**123**

Power

Block	Total PWR (PTECO)	Total PWR (PC)	Leakage PWR (PTECO)	Leakage PWR (PC)
Design 1	31.85	31.77	4.98	4.92
Design 2	39.15	39.02	6.79	6.64
Design 3	6.43	6.41	4.22	4.2
Average Diff		**0.30%**		**1.44%**
Design 4	2.63	2.61	2.3	2.28
Design 5	30.13	30.34	12.36	12.5
Design 6	51.59	52.13	15.79	16.3
Average Diff		**-0.87%**		**-2.07%**

Total Setup TNS is significantly reduced by **>40%**

Comparable

Leakage power is improved by **~1.5%** for high route_opt DB quality on the more mature advanced process node N

Comparison Results

PTECO™ vs PrimeClosure™ (Power)

Design 1	route_opt	PTECO	PC	Diff
ULVT%	4.38	4.42	4.25	-0.17
ULVTLL%	6.65	6.66	6.54	-0.12
LVT%	30.01	29.98	27.84	-2.14
LVTLL%	58.93	58.90	61.33	2.43
PTPX				
Total Power [mw]	31.774	31.846	31.766	0.25%
Dyanamic Power [mw]	26.838	26.865	26.844	0.08%
Leakage Power [mw]	4.936	4.981	4.922	1.18%
Timing				
R2R Setup TNS	-25.4	-8.41	-0.36	96%
Number of R2R Setup Viols	3194	556	52	504
Number of R2R Hold Viols	2485	15	43	-28

⬇ *High-Leakage Cell Usage*

⬆ *Low-Leakage Cell Usage*

PrimeClosure™ achieves better power recovery than PTECO™

Comparison Results

PTECO™ vs PrimeClosure™ (Runtime)

Block	Advanced Node	Number of Instances (M)	PTECO (HH:MM)	PC (HH:MM)
Design 1	N	~1	5:53	4:50
Design 2	N	~1.35	9:30	7:29
Design 3	N	~1.3	9:30	7:58
Design 3	N+1	~0.33	1:54	2:05
Design 4	N+1	~1.44	8:17	6:55
Design 5	N+1	~1.39	8:35	7:31
			Average	15.60%

Logical DRC, Timing & Power Recovery

Faster Runtime

Multiple projects in progress to further speed-up runtime

Comparison Results

With Clock-Surgery Enabled

Timing (Intra-Block Only + post-ECO)

Block	Advanced Node	# of New Clock Cells	Total Setup TNS (Before)	Total Setup TNS (After)	Total Hold Viols (Before)	Total Hold Viols (After)
Design 2	N	28	-8.9	-7.7	94	348
Design 5	N+1	1	-3.24	-3.02	409	1108
Average Diff				12%		-953

Routing

Total DRC (Before)	Total DRC (After)	Total Short (Before)	Total Short (After)
852	867	20	17
97	99	23	24
	-17		2

Power

Total PWR (Before)	Total PWR (After)	Leakage PWR (Before)	Leakage PWR (After)
39.02	39.08	6.64	6.72
30.34	30.45	12.5	12.6
	-0.25%		-0.94%

Total Setup TNS is further improved by **~12%**

Comparable

Power is only increased slightly

An additional ECO loop may be required to address hold degradation at lower buckets, primarily caused by pre- and post-ECO miscorrelation.

Comparison Results

AI-driven ECO (Leakage Power Saving)

Testcase : ~0.33M instances and leakage power dominated design
Configuration : Default built-in power Permutons

Design 4	PTECO	PC	PC (AI-driven)	Diff (vs PTECO)
LVT%	5.63	5.86	5.31	-0.32
SVT%	67.31	64.65	62.45	-4.86
HVT%	27.06	29.50	32.24	5.18
PTPX				
Total Power [mw]	2.612	2.510	2.419	7%
Dyanamic Power [mw]	0.330	0.330	0.330	0%
Leakage Power [mw]	2.281	2.180	2.088	8%
	i	4%	4% (ii)	
Timing				
R2R Setup TNS	-1.06	-0.32	-0.32	70%
Number of R2R Setup Viols	8	6	7	-1
Number of R2R Hold Viols	78	89	87	9

⬇ *High-Leakage Cell Usage*

⬆ *Low-Leakage Cell Usage*

Total ~8% reduction in leakage power without affecting timing:-

i. ~4% leakage power saving has been achieved with PrimeClosure (vs PTECO).

ii. Another ~4% leakage power reduction is observed with PrimeClosure AI-driven ECO enabled.

A fully automated flow for further advancing PPA

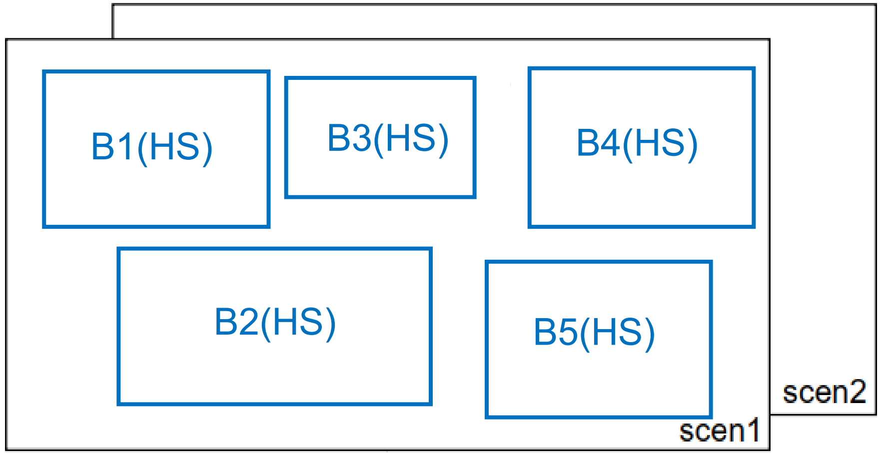
Hierarchical ECO Optimization Flow at Section/Full-chip Level

(Inter-Block or Inter-Section Design Closure)

Hierarchical ECO Flow at Section/Full-Chip Level (w/ Hyperscale)

snug

Section STA (w/ HS)



Only focus on inter-block/section optimization to reduce runtime and memory usage

write_eco_session

```
# Start pt_shell -m for ECO (non DMSA)
read_verilog; link;
# Each scenario includes HS block models:
# set_hier_config -block Block1 -path $model_block1
# set_hier_config -block Block4 -path $model_block4

update_timing

write_eco_session -include {pt_session smsa_data} \
-smsa_data_types {setup hold max_transition
max_capacitance pin_slew drc_max_transition
drc_max_capacitance } -smsa_data_format binary

-smsa_netlist_data {<path to block1.v> <path to
block2.v> <path to block3.v> {<path to block4.v> {<path
to block5.v> {<path to Top.v> } \
-smsa_spef_data {<path to block1.spef.gz> <path to
block2.spef.gz> <path to block3.spef.gz> {<path to
block4.spef.gz> {<path to block5.spef.gz> {<path to
Top.spef.gz> } \
$sh_launch_dir/section_eco_sess_binary_scen1
```

pt_shell

scen2

Path to full Verilog file

Path to full Spef file

Hierarchical ECO Flow at Section/Full-Chip Level (Recipe)

pc_script.tcl

```
set multi_scenario_working_directory PC_DMSA; set_host_options -num_processes 12
set_technology –node <N>

# Additional settings from set_eco_options can be included here
read_physical_data  -tech $tech_lef  -lef $lef_files

# Voltage areas
read_physical_data -def {Top.def} -physical_constraint_file {Topva.tcl}
read_physical_data -def {B1.def} -physical_constraint_file {B1va.tcl}
read_physical_data -def {B2.def} -physical_constraint_file {B2va.tcl}
…

read_eco_session section_pre_eco_scen1 -scenario_name scen1 # Section level sessions
read_eco_session section_pre_eco_scen2 -scenario_name scen2 # Section level sessions
start_eco -mode smsa -smsa_data_type {setup hold max_transition pin_slew
    drc_max_transition drc_max_capacitance} \
    -scope_design_list {SECTION B1 B2} -scope_path_type {interblock} \
    -scope_drc_type {interface_wires|interblock}

# Enable technology specific settings
fix_eco_drc
fix_eco_timing -type setup
fix_eco_timing -type hold

# report_eco_summary or other SMSA reporting commands
ecotclout -icc2 <eco_changes.tcl>
```

Optional for Hyperscale

- Configure Machine Resources for STA
- Enable technology specific settings
- Provide the standard LEF, DEF and VA (Voltage Area) inputs for physical information for TOP and blocks
- Read PT pre-ECO sessions
- Conduct ECO operations
- Write out hierarchical changes for implementation.

Hierarchical ECO Flow at Section Level (Comparison Results)

Section Hyperscale Testcase

- Consists of 16 blocks
- ~14M instances (~30% involving inter-block logics)

	PTECO	PC
Runtime	4:37:57	6:31:35
Physical-Aware	none	open_site + none
Total Violating Endpoints	Non-Unique	Unique

PTECO

	Total Violations			
	Before PTECO		After PTECO	
Scenario	Setup	Hold	Setup	Hold
PVT1_vmid_min	0	861	0	2
PVT2_vmid_min	0	523	0	3
PVT3_vhigh_min	0	6104	0	55
PVT4_vlow_min	0	356	0	0
PVT5_vlow_max	8	0	0	0
PVT6_vmid_min	0	1907	0	1
PVT7_vmid_max	121	0	0	0
PVT8_vlow_max	10	0	0	0
PVT9_vmid_max	835	0	25	0
PVT10_vmid_max	194	0	0	0
PVT11_vhigh_max	1504	0	235	0
PVT12_vhigh_max	1426	0	366	0
Total	**4098**	**9751**	**626**	**61**
Fix-Rate			85%	99%

PrimeClosure

```
Group : all_group              Original                  Current
                               setup       hold          setup       hold
-----------------------------------------------------------------------------
       Critical Path Slack:    -0.05       -0.21         -0.04       -0.08
       Total Negative Slack:   -155.68     -278.73       -33.10      -11.17
       No. of Violating Paths: 16497       29025         3076        674
  TNS of Violating Endpoints:  -20.60      -58.98        -3.08       -1.82
   No. of Violating Endpoints: 1879        8345          284         164
-----------------------------------------------------------------------------
```

Hold

Setup

Key takeaway: Based on pre-ECO what-if analysis, PrimeClosure demonstrates >20% better setup timing than PTECO in terms of violations.

Future Technologies & Enhancements

- Production release of new & advanced 'AI-driven ECO' technology.
- Clock-Surgery with splitting/cloning technique (Maintain the same level of clock tree during timing and/or clock max-transition/capacitance fixing).
- Improving Ease-of-Use and Debuggability, e.g. eliminating DEFs and LEFs input collaterals when NDM is provided, generating bottleneck analysis reports including blocking code for each unfixable endpoint, etc.

Summary

- PrimeClosure™ (The Next-Generation ECO Solution) is demonstrating better timing results than the conventional PTECO™ signoff ECO convergence flow, by significantly reducing setup TNS by >40%.

- These results are achieved without impacting routing. Additionally, also observed >1.5% leakage power saving & ~15% runtime improvement.

- The new 'Clock-Surgery' & ‘AI-driven ECO’ features are capable of further improving setup TNS by >10% & leakage power by another ~4%, respectively.

- In conclusion, PrimeClosure™ has the potential to reduce ECO efforts and iterations at advanced nodes for high-speed and high-density designs, while also accelerating design closure and time-to-market.

THANK YOU

YOUR
INNOVATION
YOUR
COMMUNITY

Introduction of Key PrimeClosure™ Technologies

Debug GUI and Manual ECO

- Violations analysis through different views, design analysis through different maps
- Provides clear block messages for unfixable violations
- Versatile and interactive with easy flow control with manual ECOs
- Summarizes the trend of blocking info (unfixable reasons) at the end of each autofix log.
- Let's user to know the specific reason and corresponding variables to set