Advanced LLE Aware Timing Analysis / Optimization

Tusharkant Mishra Damodaran Trikkadeeri Ranjith VR
Santosh Varanasi Nikhil Nalla

Agenda

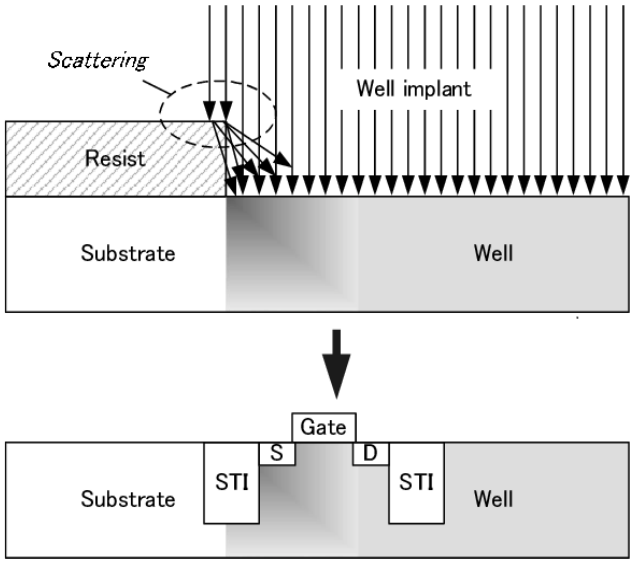
- **LLE Background**
- **LLE Effects on GAA Devices**
- **LLE Impact on Timing**
- **Primetime Based LLE Aware Timing**
- **LLE Aware Timing Optimization**
- **Results**
- **Supplement Topics**
- **Conclusion**

Background

- **Layout Dependent Effects**

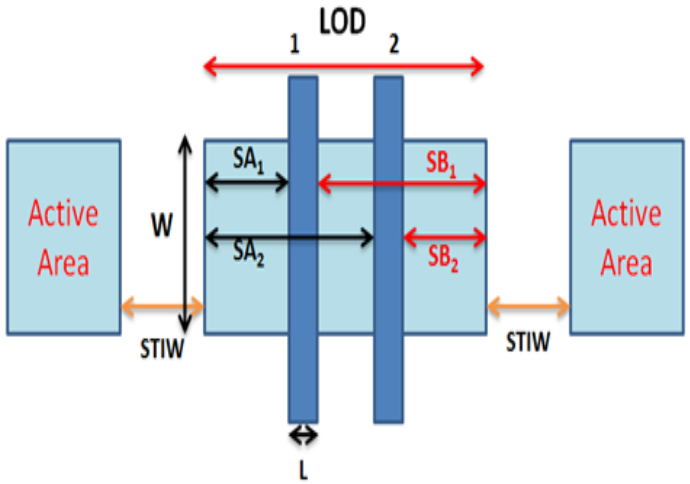
WPE

(Well Proximity Effect)



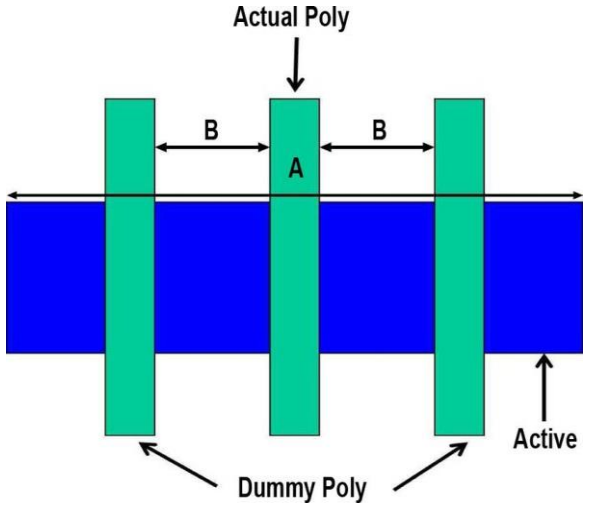
LOD

(Length Of Diffusion)



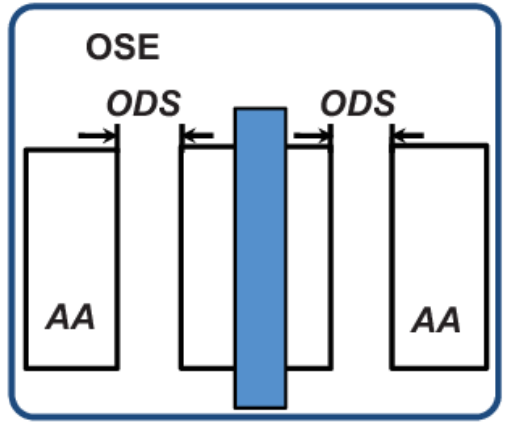
PSE

(Poly spacing effect)



OSE

(OD-to-OD spacing)



(sources: Internet)

Overview

- **GAA (Gate All Around) Device**
  - MBCFET (Multi Bridge Channel) Gate-All-Around (GAA) from Samsung Foundry (Fig.1). MBCFET can support multiple nanosheet width. Here we are presenting GAA device with four types of nanosheet (**NS1/NS2/NS3/NS4**)

    NS4 Width > NS3 Width > NS2 Width > NS1 Width
- **LLE (Local Layout Effect)**
  - Neighbor nanosheet devices adversely effect the victim cell delay causing setup/hold timing degradation .
- **Tapered-RX is dominant LLE in GAA devices** *RX = **R**egrowth o**X**ide = Active Area*
  - **T-shape** : taller NS gets slower by shorter neighbor (Fig.2)
  - **U-shape** : shorter NS gets faster by taller neighbor (Fig.3)
  - LLE impact is getting bigger (>10%) in GAA Devices
  - Bigger impact as lower the voltage and higher the Vt class
- **Library characterization relies on fixed overlay patterns**
  - Problem: overlay vs. actual context
  - Conventional min (for FF) and max (for SS) overlays used for library characterization
  - **min-overlay (NS3) @FF** :
    - data path fastest delay, pessimistic but no hold timing risk and setup timing trivial @ FF -
  - **max-overlay(NS1) @SS :**
    - data path slowest delay, hold timing risk and setup timing could be overly pessimistic

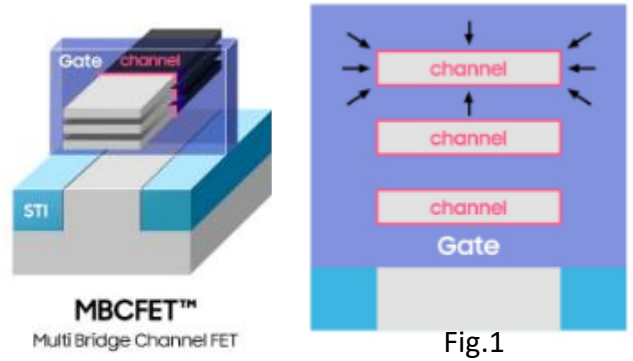


Fig.1

Tapered T-shape RX LLE – GDS vs Si

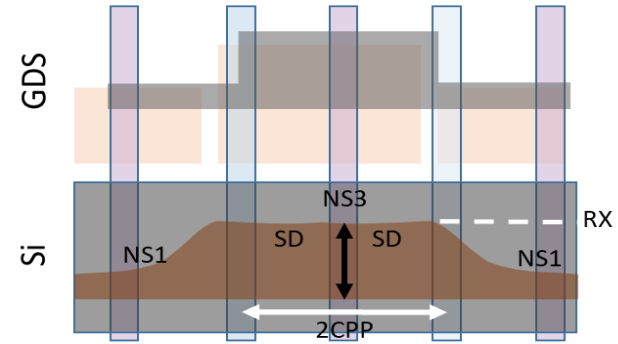


Fig.2

Tapered U-shape RX LLE – LLE

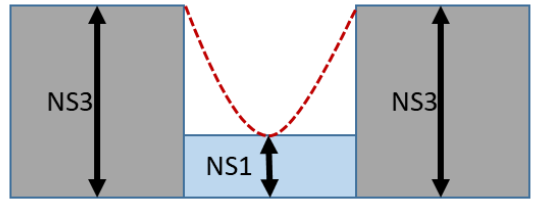
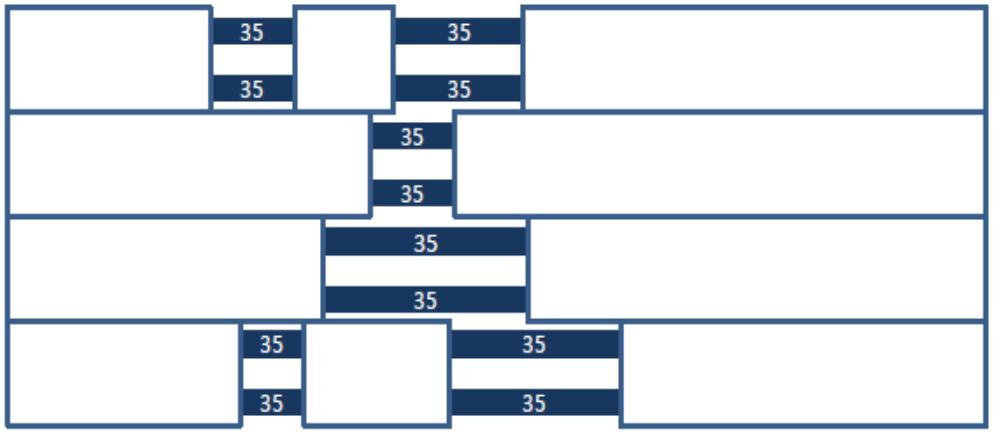


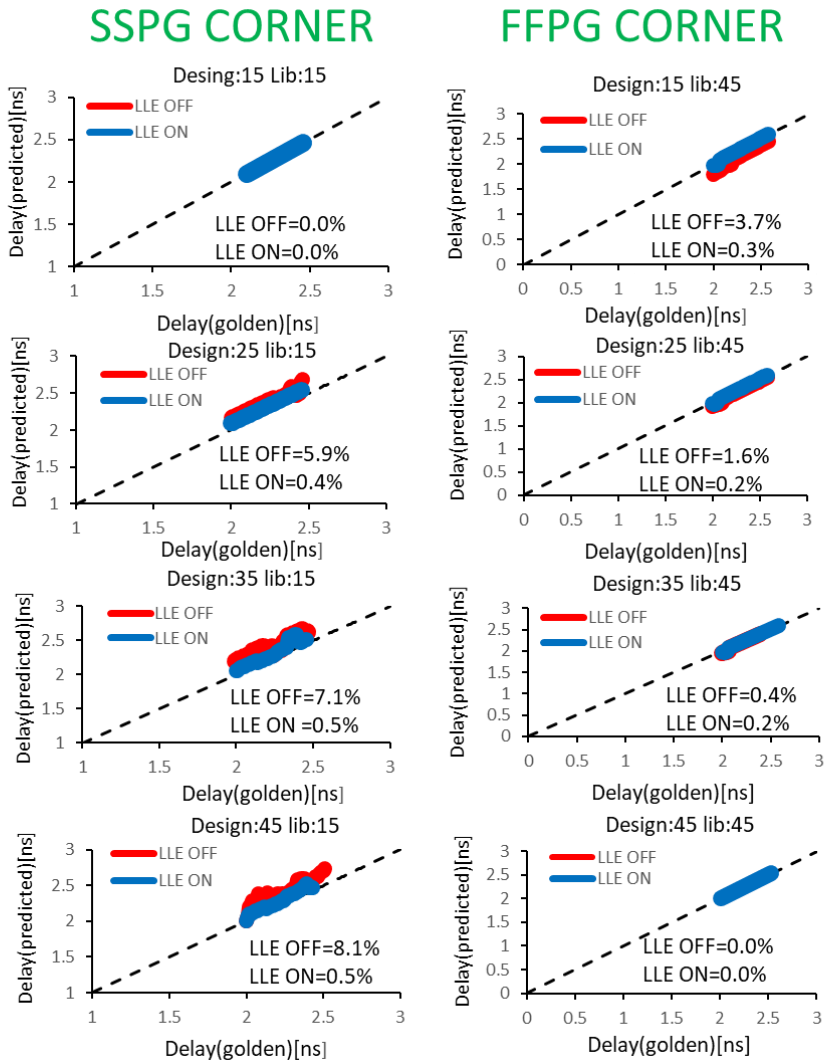
Fig.3

Design level Gap between LLE off and on

- **Validation : STA-SPICE Correlation**
  - Design with various NS width cells
  - Between all instances, have enough space for filler cell
  - Same NS filler cells are inserted in this space
  - Check STA-SPICE correlation by all NS cases
  - Shows better correlation with LLE aware Flow

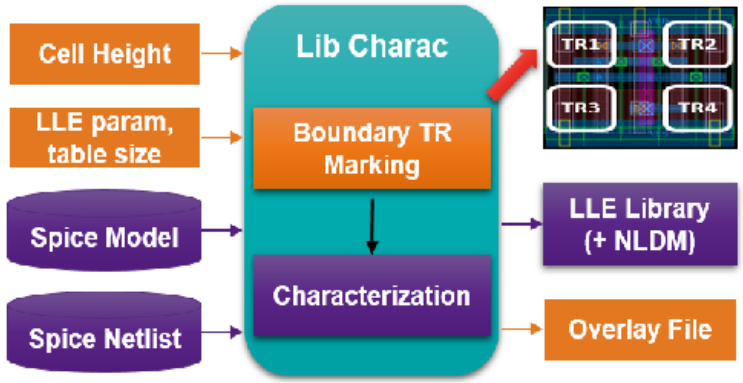
Layout (after filler insertion. e.g. 35nm filler cells)





LLE Sensitivity Library Characterization

- Sensitivity Characterization
  - LLE impact is modelled for
    - Boundary transistors
    - timing (delay/slew), leakage power
    - Perturbation parameters (Vth, u0)
  - Needed information
    - Cell height for boundary transistor recognition
    - Perturbation values for LLE parameters
    - Index points (table size)



Sample Liberty Format

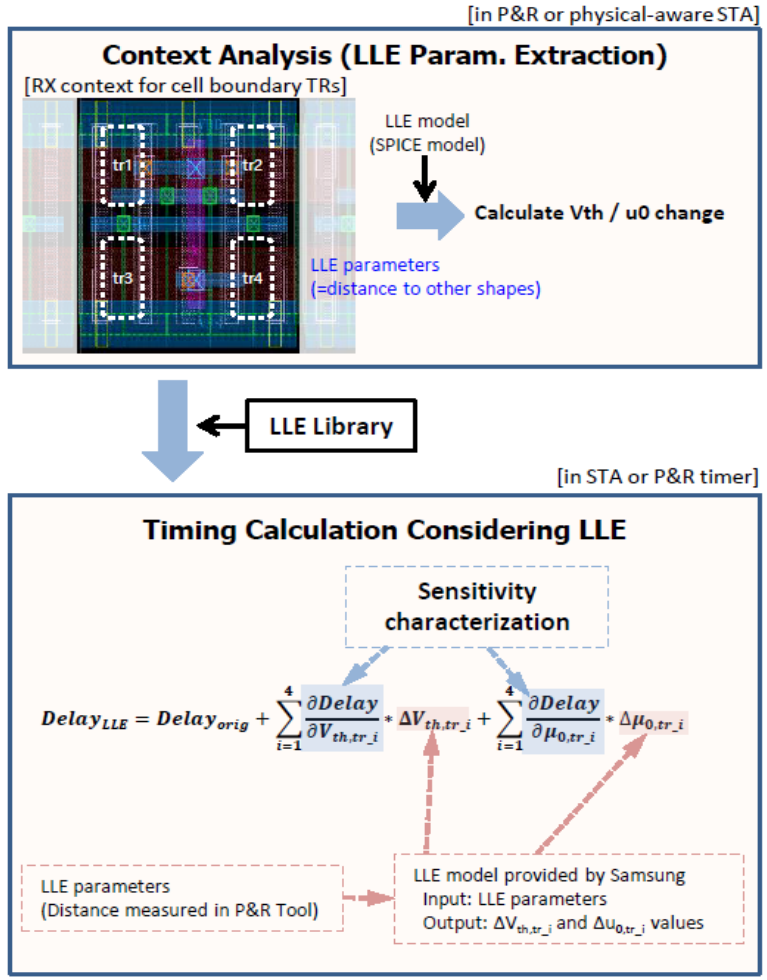
```
cell(test_cell) {
leakage_power() {
   related_pg_pin : "VDD" ;
   when : "!A&!B&!CI&!CO&!S" ;
   value : "2.91394e-05" ;      ## origin leakage value
   ...
   lle_delta_leakage_power(lle_lkg_delta_2) {
     related_params : "vth" ;   ## param definition > Vth or u0
     related_devices : "tr1" ;   ## boundary tr definition > tr1, tr2, tr3, tr4 ....
     index_1("-0.015, 0.015"); ## perturbation value of parameter
     values("2.87158e-09, -2.56085e-09");  ## sensitivity (delta value)
   }
   ...
}
 ...
 pin(Y)
   timing() {
   ...
     cell_fall(8x8) {
       index_1("...");
       index_2("...");
       values("..."); ;      ## origin timing table
     }
     ...
     lle_delta_cell_fall(lle_1x3x3) {
       related_params : "vth" ;   ## param definition > Vth or u0
       related_devces : "tr1" ;   ## boundary tr definition > tr1, tr2, tr3, tr4 ....
       index_1("0.015");          ## perturbation value of parameter
       index_2("0.0011535, 0.102186, 0.809411");  ## index from origin index
       index_3("8.5511e-05, 0.00230265, 0.0355597");
       values("-1.51201e-06, -2.65675e-07, 6.45165e-09",\
              "-1.87921e-07, -3.77126e-07, -1.41947e-07",\
              "1.25844e-06, -1.54662e-05, -1.19289e-05");  ## sensitivity (delta value
     }
...
     }
}
```

Advanced LLE Aware Timing Analysis

- **As LLE effect cannot be accurately characterized into library so LLE aware STA is required to accurately model LLE effect**

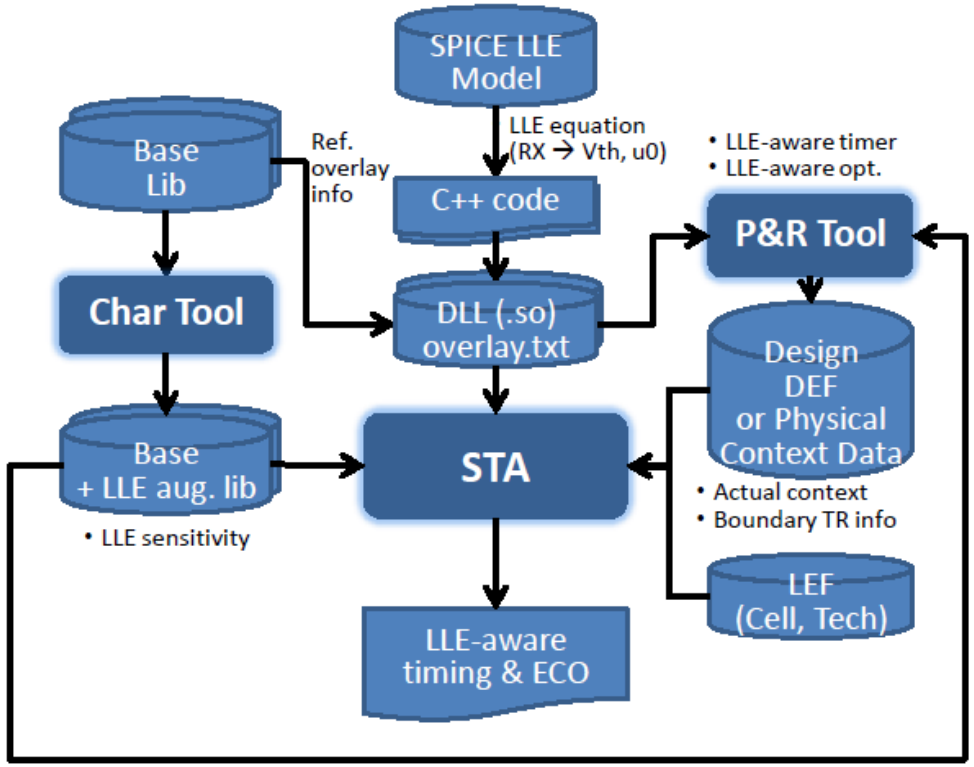
- **LLE impact is modelled into Vth and mobility (μ0) SPICE parameter shift per transistor (SPICE LLE Model)**

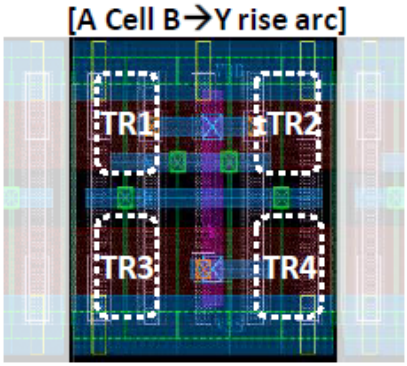
- **LLE Sensitivity Library**
  - Timing impact per cell boundary TR's parameter shift is characterized for each timing arc
- **Advanced LLE-aware STA**
  - **STA tool estimates actual timing impact based on the RX context from design using LLE augmented library**
  - Actual Context
    - Vth and μ0 shift per boundary device
    - Adjust timing arc delay /transition by the parameter shift (char overlay vs. actual)



LLE-Aware Design Methodology

- **Flow and Required Collaterals**
  - LLE augmented libraries
    - Timing arc sensitivity per victim transistor impact
  - LLE model and overlay info:
    - DLL(Dynamic Link Library) compiled for LLE impact equation
    - Char. overlay (as a shift baseline)
  - DEF and LEF
    - For context info and boundary TR mapping





Physical Context

LLE Equation

Param shift over char overlay:
TR1: Vth +5mV, u0 -3%
TR2: Vth +10mV, u0 -5%
TR3: Vth -8mV, u0 +4%
TR4: Vth +9mV, u0 +1%

charac vs. actual context

Sensitivity Lib

B → Y rise:
Δdelay
Δtransition

param shift effect on timing arc

LLE-Aware Delay & Transition

LLE Aware Timing Optimization

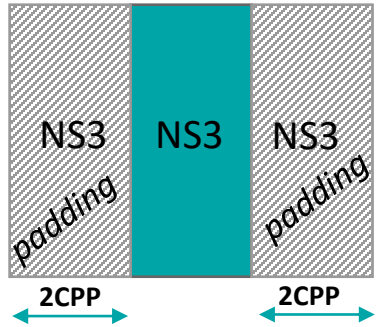
- **Clock Path Optimization**

PROBLEM

- Design context can't be determined at CTS stage
- Changing clock cell based on design context can have huge impact

SOLUTION

- Flat RX overlay for Clock cells (Char. Context = Actual Context)
- 2CPP padding to eliminate discrepancy between char. context vs actual context



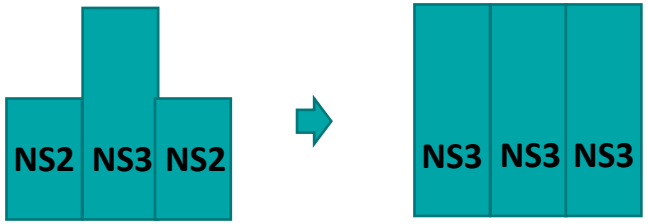
- **Data Path Nanosheet Swap**

PROBLEM

- Neighbor nano sheet difference can impact the delays of cells on critical path
- Adjacent filler cells with mismatch nanosheet can affect delays of cells on critical path

SOLUTION

- Create flat RX profile for critical cells on data path by swapping neighbor mismatch nano sheet cells (margin aware swap)
- On base filled DB swap the mismatch nano sheet fillers adjacent to critical data path cells



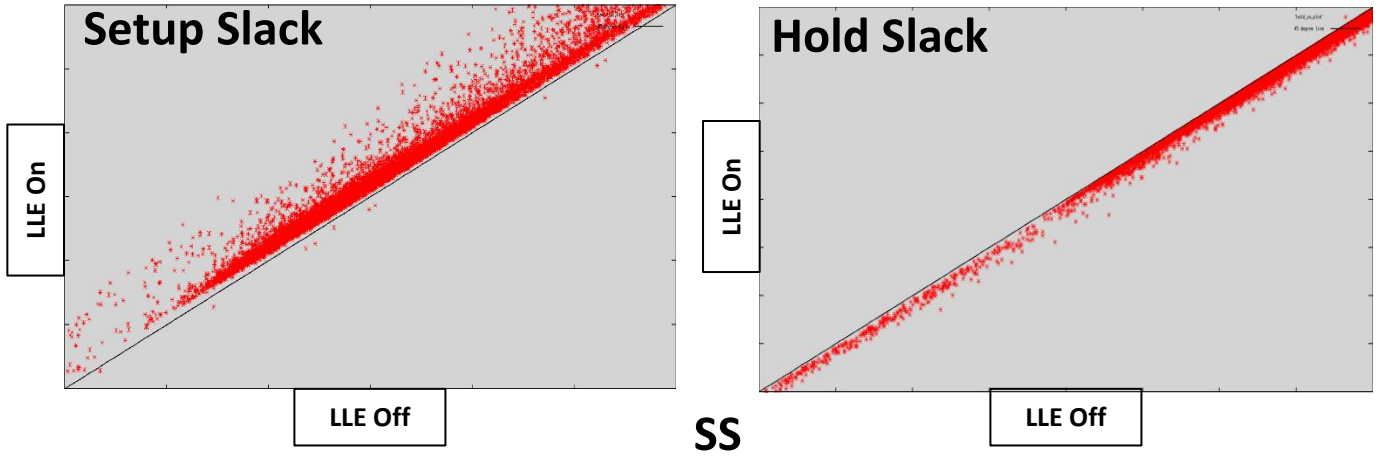
Setup Optimization

Hold Optimization

Results

LLE Off (Char Overlay library) Vs LLE On (Actual Context)

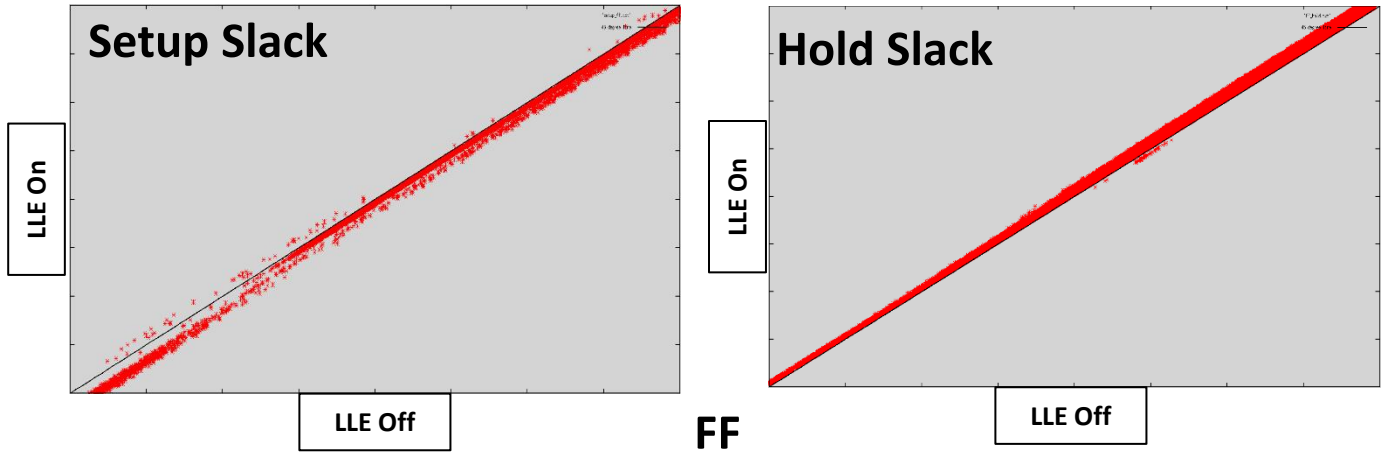
* Based on 3nm SOC data



SS

SS LLE off setup is overly pessimistic than SS LLE on setup

SS LLE off hold optimistic than SS LLE on hold

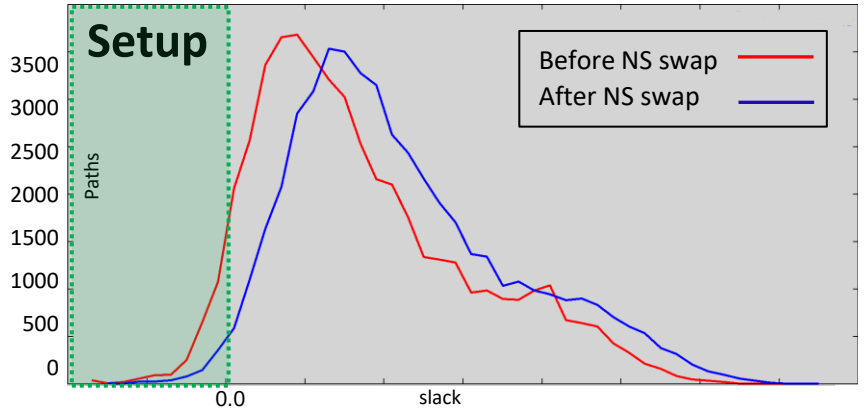


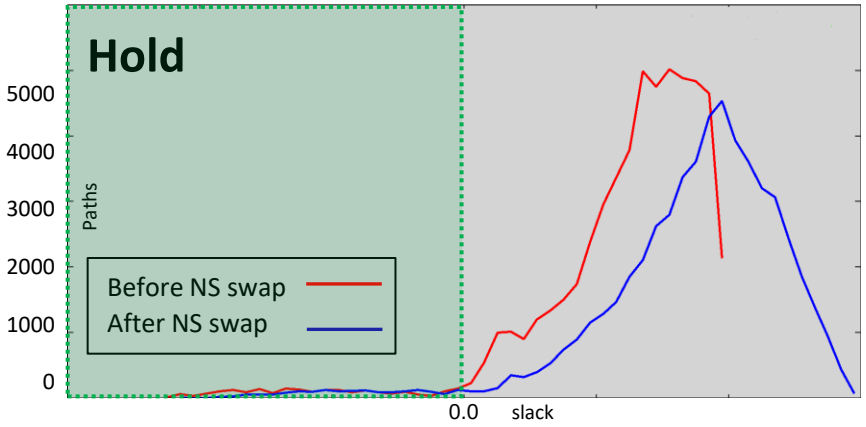
FF

FF LLE off hold pessimistic than FF LLE on hold

Slack profile with datapath nanosheet swap

* Based on 3nm SOC data

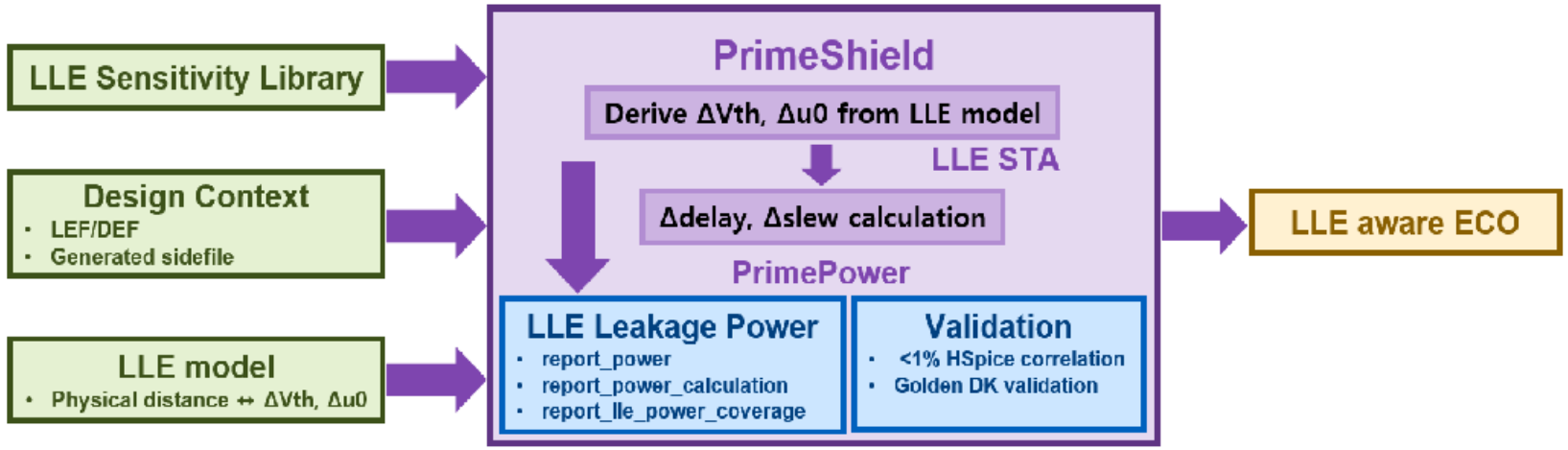




LLE Aware Power Analysis

- Overall Flow and Required Collaterals
  - Similar design flow and input collaterals as LLE STA

- LLE Leakage power calculation
  - Leakage varies linearly with mobility (u0) and quadratically with threshold voltage (Vth) shift
  - For FFPG (SSPG) corner, leakage generally *decreases* (*increases)* when LLE is ON
  - PrimePower leakage power accuracy within 1% of HSpice simulation and Golden DK results
  - New “physDB lite mode” flow for full chip LLE signoff support



PrimeTime Reference Scripts

- **For LLE-aware STA, following steps should be added**
  - **(1) Enable LLE (2) DLL and overlay information (3) Physical layout information**

PrimeTime (LEF/DEF)

```
# Enable LLE
set ps_enable analysis true
source enable_lle.tbc

# LLE model(DLL) and overlay information
enable_lle -lle_model_path DLL_SNPS_${process_corner}/lle.so \
 -overlay_sidefile OVERLAY_${process_corner}/lib_lle_overlay.txt
set timing_lle_enable_sensitivity_based_calculation false

set link_path flk.db_ccs_tn_lvf
read_verilog top.v
link_design top
read_sdc top.sdc
read_parasitics top.spef.gz

# Physical layout information
set lef_files [glob LEF/*.lef]
set_eco_options \
   -physical_lib_path $lef_files \
   -physical_design_path top.def \
   -physical_tech_lib_path top.tech.lef \
   -log_file ./lefdef.log
check_eco

update_timing -full
```

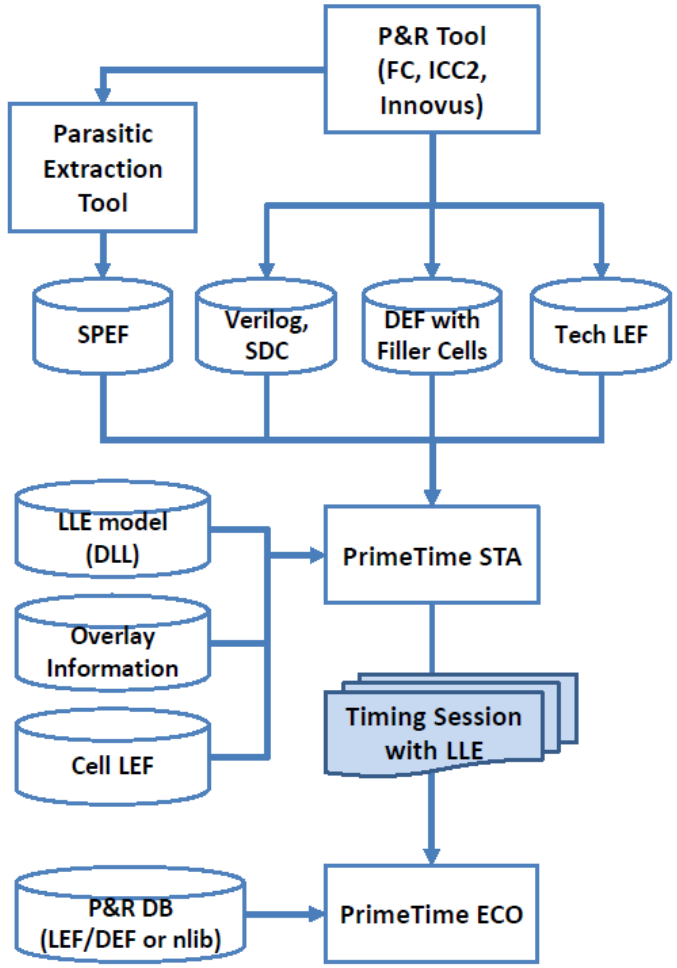
PrimeTime (nlib)

```
# Enable LLE
set ps_enable analysis true
source enable_lle.tbc

# LLE model(DLL) and overlay information
enable_lle -lle_model_path DLL_SNPS_${process_corner}/lle.so \
 -overlay_sidefile OVERLAY_${process_corner}/lib_lle_overlay.txt
set timing_lle_enable_sensitivity_based_calculation false

set link_path flk.db_ccs_tn_lvf
read_verilog top.v
link_design top
read_sdc top.sdc
read_parasitics top.spef.gz

# Physical layout information
set_eco_options \
   -physical_enable_clock_data \
   -physical_enable_all_vias \
   -physical_icc2_nlib top_chip_finish.nlib \
   -physical_icc2_blocks top \
   -log_file ./nlib.log
check_eco

update_timing -full
```

PrimeTime ECO Flow

- **Inputs**
  - STA session (LLE aware)
- **LLE-aware ECO**
  - If LLE setting is set in STA, don't need to set in ECO
  - During ECO iteration, tool considers LLE timing
  - Tool calculates LLE timing with existing filler cell
- **For LLE-aware PrimeTime ECO , following steps should be added**
  - Filler cell invisible setting
  - Advanced legalizer enable (nlib base)

```
# Filler cell invisible
set eco_allow_filler_cells_as_open_sites false
set eco_physical_make_fillers_invisible true

# Advanced legalizer enable
fix_eco_timing -legalize -type setup .....
fix_eco_timing -legalize -type hold .....
```



LLE Aware Place And Route

- For LLE-aware optimization, LLE-aware analysis is can be also turned on in P&R
- P&R engine takes LLE effect in timing calculation, so that it could select critical paths / resize cells / legalize with more accurate timing data.

LLE-aware timing calculation

(*) Empty spaces are assumed to be filled following filler app option

Optimization with LLE-aware timing

Summary

- Local layout effects are becoming pronounced with GAA nanosheet transistors
- Conventional overlay methods of library characterization based on min/max overlay results in overly pessimistic analysis for setup and optimistic analysis for hold
- Advance LLE aware STA analysis and optimizations are essential for accurate timing convergence on designs with GAA transistors
- LLE aware optimization (clock cell padding , neighbor data path cell nanosheet swap , filler cell nano sheet swap) will help to reduce LLE impact on timing QOR

Acknowledgement

- We, Samsung Foundry Design Methodology Team thankful to AE team **Santosh/Nikhil** for enabling LLE aware timing Signoff flow


THANK YOU
Our Technology, Your Innovation™