Logic and SRAM Library Generation and Analysis for Digital Design Enablement

Sravanth Mucharla, Shruti Rakheja (Google - gChips)
Ray Valencia, Shane Toma (Siemens EDA)

SPONSORED BY CEDA IEEE Council on Electronic Design Automation sigda acm

Sravanth Mucharla

Google, Physical Design Engineer

Shruti Rakheja

Google, Staff Physical Design Engineer

Motivation and Main Idea

SPONSORED BY CEDA® IEEE Council on Electronic Design Automation SIGDA acm

Motivation

Liberty (.lib) files are the universally accepted format for representing digital circuits early in chip design

- They form the initial building blocks in implementation and sign off cycles, such as in Mobile SoC design

Mobile SoCs have many unique challenges:

→ Annual re-design of tape outs are necessary to implement newest cutting edge foundry processes

- PDK evolution is rapid, sometimes a few weeks before tape-out!

→ Wide range of voltage operation to push the boundaries of performance and power efficiency

- Require large sets of PVT libraries to meet design closure
- Library characterization is expensive - cost and compute intensive, impacting design cycle times

Main Idea

The work in this presentation details proposed flows for comparing critical parameters in the liberty, alongside a methodology for producing liberty files quickly and flexibly

This study describes 3 separate flows;

1. **Liberty comparison flow**, for use in std cell library comparison to compare attributes present in the liberty file and identify erroneous data

2. **Library generation flow**, for liberty generation using machine learning technologies with the intent to reduce up-front characterization time for accelerated turnaround

3. **Library data parser flow**, to enable flow users to write custom checks for customized use cases. Extracts parameter wise average data for each cell present in the library files. Ideal for understanding key behaviors of cells.

Liberty comparison flow

SPONSORED BY CEDA® IEEE Council on Electronic Design Automation SIGDA acm

Liberty comparison flow

- Std cell liberty comparison flow is used to compare all the attributes present in the liberty file with the help of the *Solido Analytics tool*
- This data is useful to understand the trend in new lib (or DK) with respect to PPA impacting parameters like:
  - Speed, Variation, Input pin capacitance, Internal and leakage power, Output transition, … etc.
- This data is available for all the cells in liberty file and across all input slew and load range.
  - Significant advantage over RO spice simulations
- Designers can make informed decisions on impact due to new timing models which would otherwise be known only after one round of design implementation such as:
  - Assessing the impact on perf critical compute cores on signoff frequency target.
  - Analyze the impact on internal and leakage power.
  - Rank cell-architecture/cells with high-leakage - provide guidance to design implementation.
  - Choice of clock tree cells which have best internal power and delay trade-off.

Liberty comparison flow

Primary and secondary liberty files

Primary Data Set
PVT corner 1
PVT corner 2
.
.
.
PVT corner n

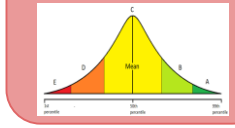
Corner Mapping by tool

Secondary Data Set
PVT corner 1
PVT corner 2
.
.
.
PVT corner n

Compare matching .libs

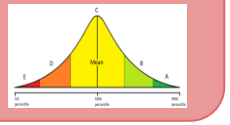
Compare the PVT mapped Liberty files across all the parameters for all the cells

Corner 1 compare distributions



Distribution Reports

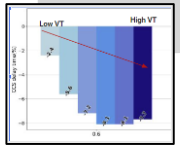
Corner n compare distributions



Summarize and plot the comparison Data

Plot graphs capturing design parameter variation across PVT and Vt

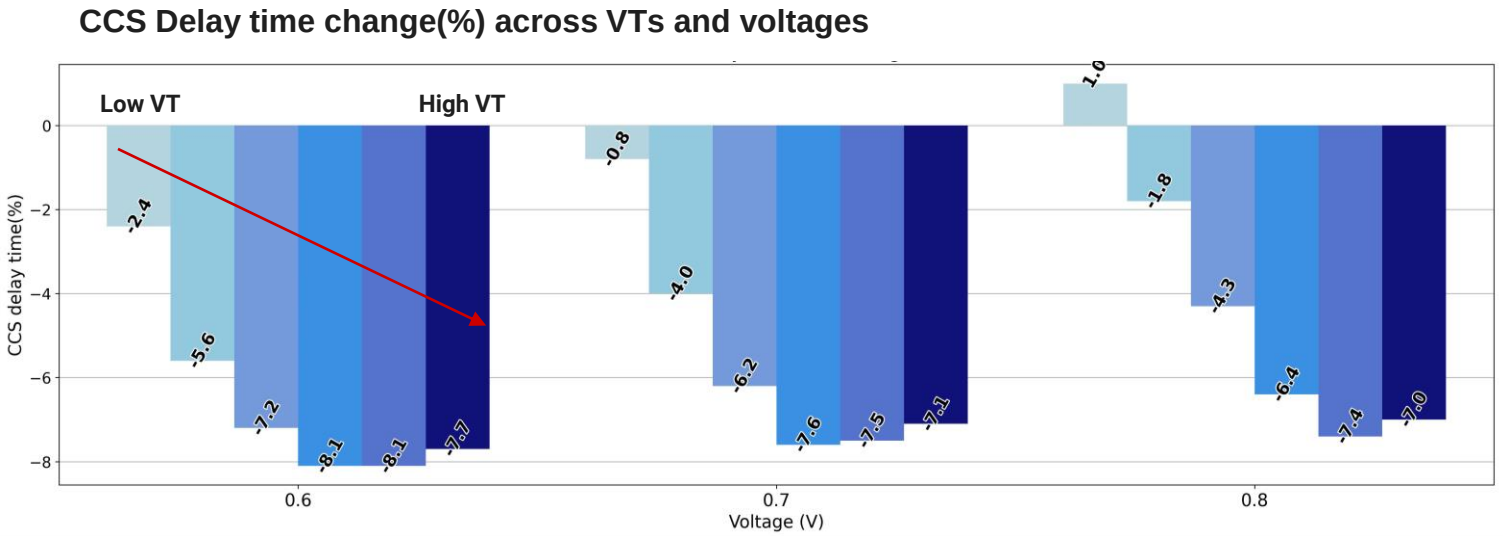
Parameter change Bar Plot



Liberty comparison flow output

PPA: library level comparison

- Plot shown here is an output of lib comparison flow
- Provides additional layer of information for methodology over technology entitlement
- Critical parameters: delay, delay-variation, internal power, input capacitance … etc.



Key takeaways from above plot

- Speed up in general across all VT devices, higher VT has higher speed up
- Speed up decreases as we increase the voltage.

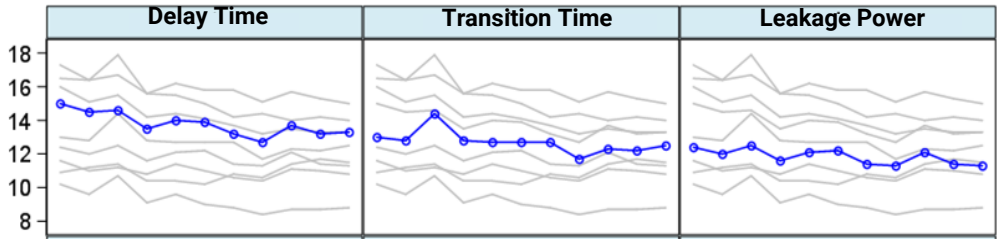
Library Generation

SPONSORED BY CEDA® IEEE Council on Electronic Design Automation sigda acm

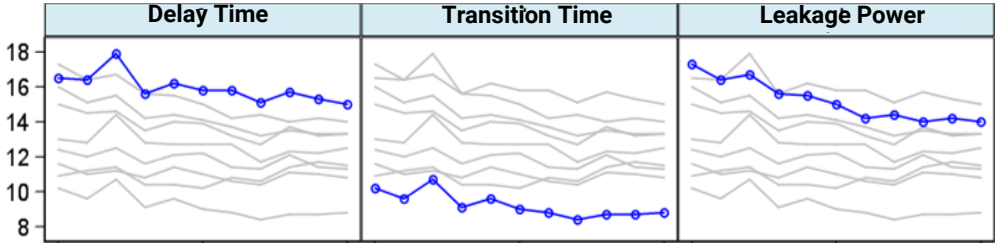
Liberty generation

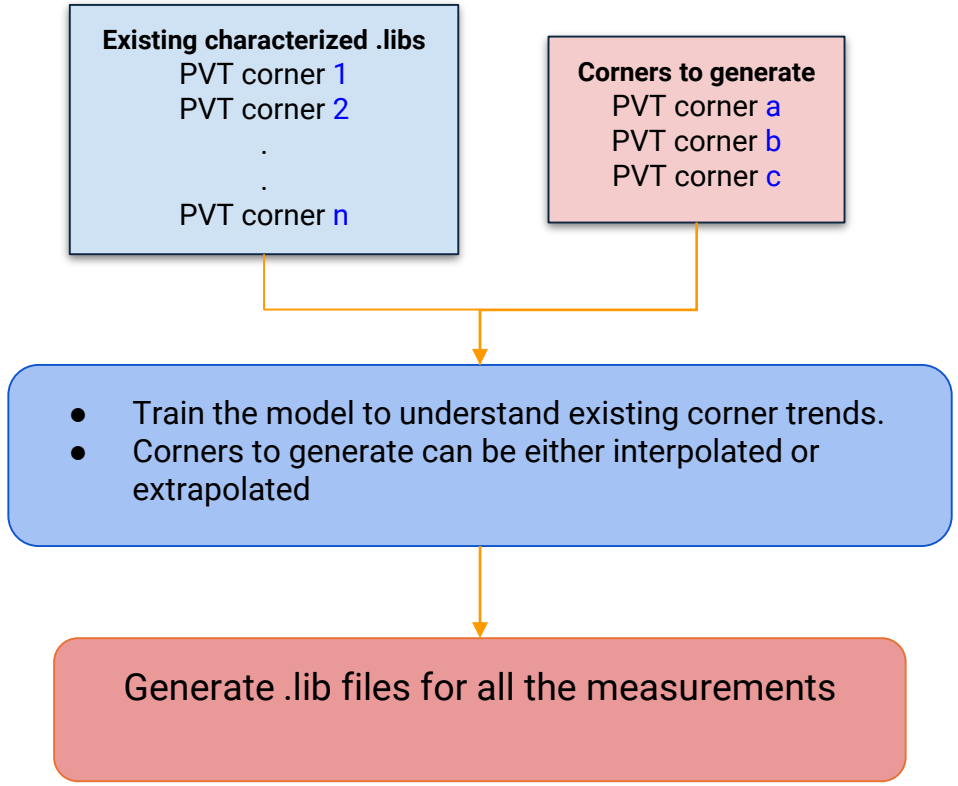
- *Solido Generator solution* uses machine learning to accelerate characterization of standard cells, memory, and I/O.
- Significantly reduces up-front characterization time and turnaround to generate new requests.

Data Interpolation



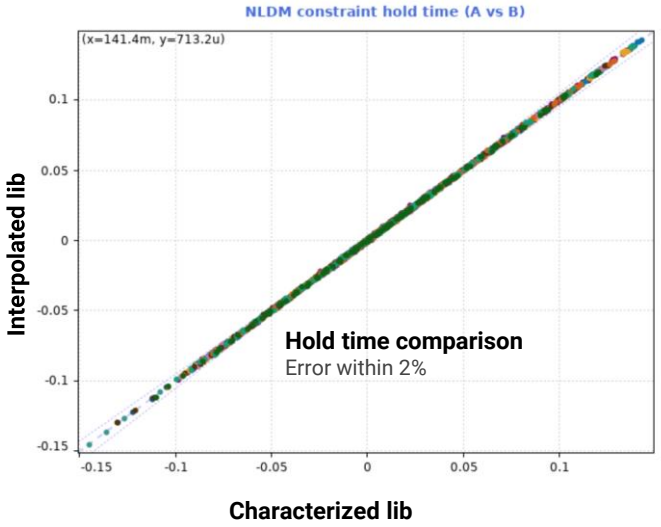
Data Extrapolation (less accurate)

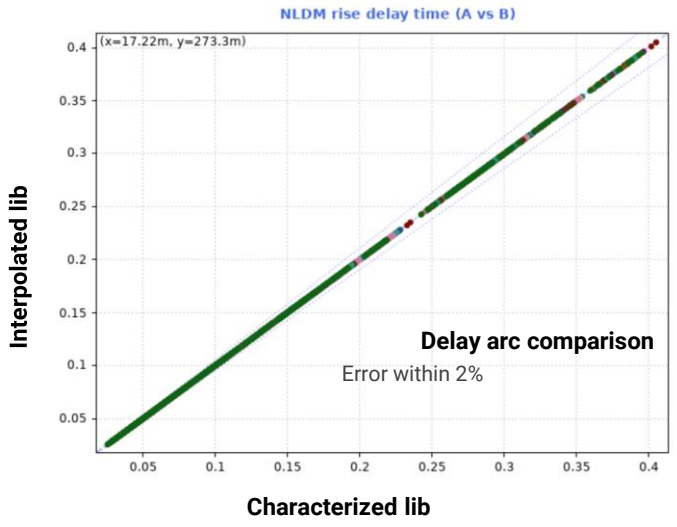




Liberty generation results

- **Intermediate liberty generation** (interpolation)
  - Std cell libs for intermediate temp corner generated and validated against characterized libs
  - Delta b/w characterized and generated lib is within 1-2% for voltages > 0.65V. At lower voltages the delta is within 3-4%
  - These libs were used in implementation in the absence characterized libs to mitigate the schedule impact
- **Extreme corner lib generation** (extrapolation)
  - Generated libs for performance critical high voltage corner, overall correlation with spice within 2-3%
  - Schedule time Saving!





Method	Accuracy	Run time
Lib Characterization	0-1 %	10-15 hours
Lib Generation	1-2 % (3-4% lower VDD)	1-2 hours

Library data Parser

SPONSORED BY

Liberty data parser

- Getting to know how different cells behave within a Liberty File is an important feature for understanding the behavior of cells
- The lib parser API script using *Solido Analytics Validate* allows users to extract measurement data from library file.
- For the below matrix we can get average of number across index ranges of interest. The output data has each line representing parameter values for a cell, arc and when condition.

Mean cell rise

```
cell_rise (delay_template_8x8) {

                                Index 1 range (3,7)
index1 ("0.1", "0.2", "0.3", "0.4", "0.5", "0.6", "0.7", "0.8");
index2 ("1.0", "1.5", "2.0", "2.5", "3.0", "3.5", "4.0", "4.5", "5.0");
                        Index 2 range (2,5)
values (\
      "0.01", "0.03", "0.06", "0.1", "0.13", "0.16", "0.19", "0.22" /
      "0.25", "0.29", "0.32", "0.35", "0.38", "0.41", "0.44", "0.48" /
      "0.51", "0.54", "0.57", "0.6", "0.63", "0.67", "0.7", "0.73" /
      "0.76", "0.79", "0.83", "0.86", "0.89", "0.92", "0.95", "0.98" /
      "1.02", "1.05", "1.08", "1.11", "1.14", "1.17", "1.21", "1.24" /
      "1.27", "1.3", "1.33", "1.37", "1.4", "1.43", "1.46", "1.49" /
      "1.52", "1.56", "1.59", "1.62", "1.65", "1.68", "1.71", "1.75" /
      "1.78", "1.81", "1.84", "1.87", "1.9", "1.94", "1.97", "2.0"/

);
}
```

Liberty data parser results

Timing data results example

Corner	cell_name	arc	when	area	Input pin capacitance	Mean cell rise	Mean rise transition	Mean internal power rise	Mean leakage power
Library 1	INVD1	I>Z	DEFAULT	0.02	0.05	0.02	0.025	1.2	0.15
Library 2	ND2D2	A1>ZN	!A2	0.13	0.08	0.05	0.03	2.3	0.52

Constraint data results example

Corner	cell_name	arc	when	Timing type	Mean rise constraint	Mean ocv sigma rise constraint
Library 1	SDFD4	CP>D	!SE	setup_rising	0.028	0.0039
Library 2	MB6SDFD2	CP>D1	!CD&!SE	hold_rising	0.043	0.0053

Summary

SPONSORED BY CEDA® IEEE Council on Electronic Design Automation sigda acm

Summary

- ✓ The Liberty comparison flow offers an early and accurate assessment across PDK/technology changes.
  - Predicting the impact of changes
  - Making informed decisions
- ✓ Library generator solution provides Liberty files with reasonable accuracy
  - Significantly Improves run time
- ✓ Library data parser is a valuable tool for understanding the behavior of cells
  - Identifying the root cause of anomalies
  - Essential resource for design analysis and debugging

AI

Security

Systems

EDA

Design

62 THE CHIPS TO SYSTEMS CONFERENCE

SPONSORED BY CEDA® IEEE Council on Electronic Design Automation

sigda acm