



JUNE 23-27, 2024

MOSCONE WEST CENTER  
SAN FRANCISCO, CA, USA



# Evaluating power, performance and area for standard cell libraries from different IP providers

Aravind Radhakrishnan Nair (Infineon Technologies)

Ajay Kumar, Austin Shirley, Lars Kishchuk (Siemens EDA)



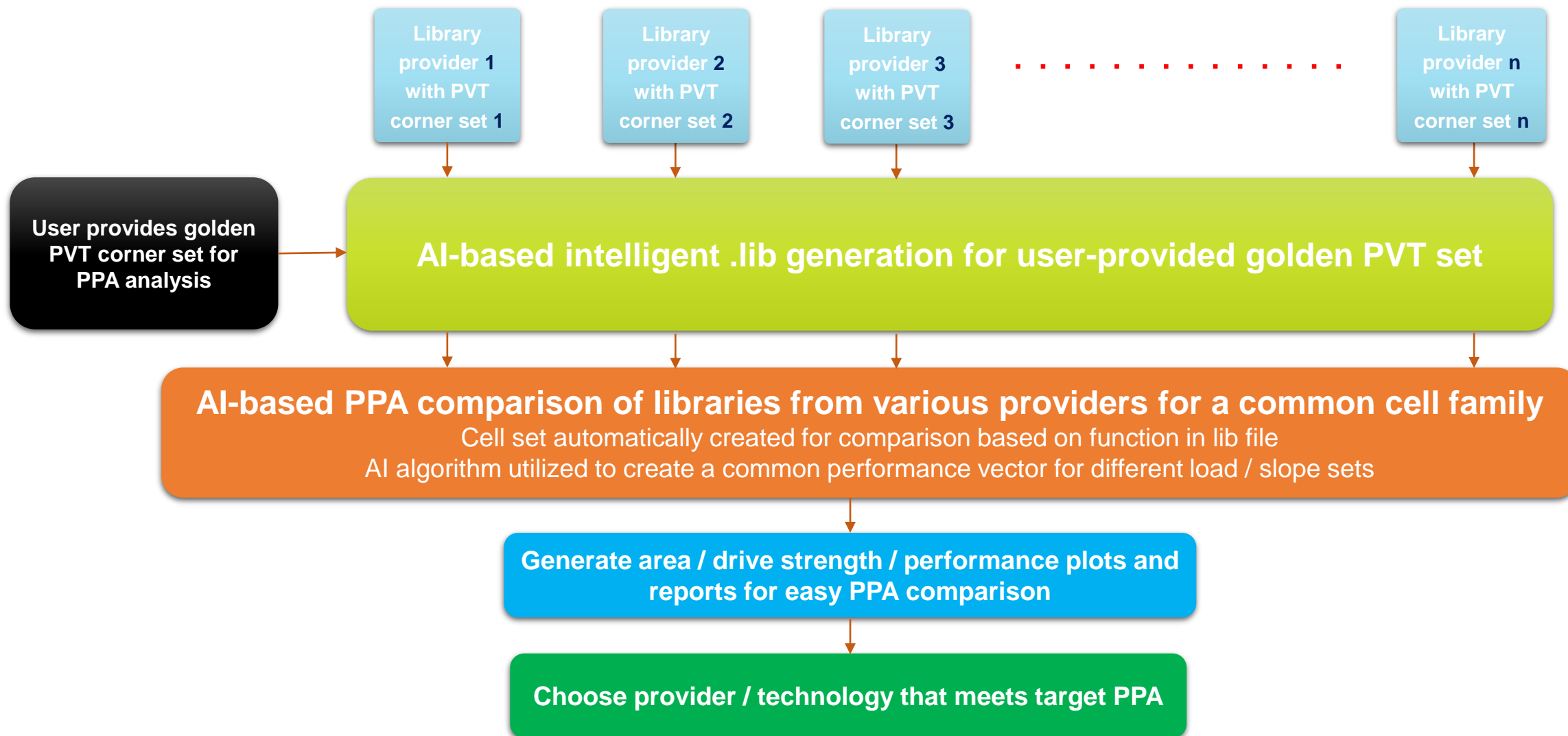
# Motivation

## ***Fast and efficient way to choose a library technology/providers for an optimized PPA***

- Many library providers, many library selections for various library technologies available in market today
- Choosing a right IP provider and library technology to meet PPA requirements is challenging and time consuming
- A typical standard cell library can have thousands of unique cells. Formatting of cells, pins, and data in .libs differ significantly between technologies, providers and variants
- Typical analysis methods are runtime intensive and inaccurate
  - Existing PPA analysis: run a reference design through Synthesis and STA
  - Requires expensive production tools, expertise, and compute time
  - Results are dependent on reference design selection and may be inaccurate
- Incorrect .lib selection is expensive
  - Introduces PPA bottlenecks in chip design and implementation
  - Results in sub-optimal final design metrics and delayed schedules
  - Early detection and analysis is the only way to solve these design issues

# Main Idea

## *Automated PPA assessment methodology*



# Additional content

## Challenges and Proposed Solution

### Existing Challenges

- Library files from each provider are different
- Different PVT corner set
  - Different cell set
  - Different cell naming styles
  - Different pin names
  - Different load/slope used in characterization

Apples-to-apples comparison not possible using .lib files from different providers / technologies

Plotting and analysis rely on difficult to maintain in-house solutions

### Proposed Methodology

Use AI to create .lib files for a common corner set with available provider .libs as training set

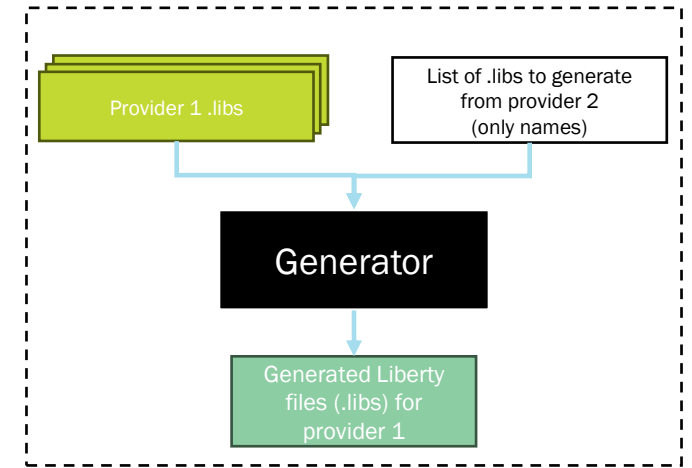
Intelligently map cells/pins/arcs/when conditions in the lib files based on functionality

Generate timing & power interpolated data for load & slopes using AI for apples-to-apples comparison

Create PPA plots and reports for analysis using built in tools

# Creating common PVT corner set using AI

- PVT corners from available .libs are often inconsistent across provider libraries. In most cases, few or none of the .libs match as required.
- Alignment of PVTs is crucial in producing an accurate and meaningful comparisons.
- For this process, alignment is performed using Solido Generator. This tool is an established AI-based tool for creating new .libs at new PVTs. Its steps are as follows:
  1. Create a list of targeted common PVTs.
  2. Seed Generator models using available anchor PVTs. Generator builds a new model based on observed trends in input data. This model is unique for the provided corners.
  3. Produce .libs with Generator and verify results for outliers, as done for any other normally produced .lib
- The final .lib files are used as normal .lib files for comparison



*Repeat process for provider n and combine all .libs for analysis*

# Alignment of .lib Data

- Given aligned PVT corners, pairs of .libs need further processing to complete alignment of *cells*, *pins*, *arcs*, and *data tables*.
- Alignment of items by defining explicit mappings "one-by-one" is not feasible for a modern library with potentially 1000s of cells and millions of values.
- To enable this flow, tool-based automatic alignment is crucial. Its steps include:
  1. Analyze cell names to classify cells into families and drive strengths
  2. Analyze cell functionality and pin counts to match common cells
  3. Among common functions, align matching pins
  4. Apply pin alignment to redefine timing and power arcs
  5. For matching timing and power tables, perform 2D interpolation to tables to match capacitance and transition times to calculate a value for comparison
- Many heuristics for automatic alignment are used, including analysis of ff and latch structures, as well as through learning pin usage in arcs within the cell.
- These methods are generic and are seeded only by input data provided. They work on standard cells, complex IO, analog blocks, or memory cells.

# Power, Performance, and Area Fundamentals

**Performance** is an indication of the speed at which a cell performs, i.e., the signal propagation time through a cell

- Lower values = better (faster)

**Power** summary helps to understand the power efficiency of cells within a library, i.e., the total energy consumed during the signal transition of a cell

- Lower values = better (less energy consumed)

**Area** summary helps to understand the sum of the occupied area across similar cells

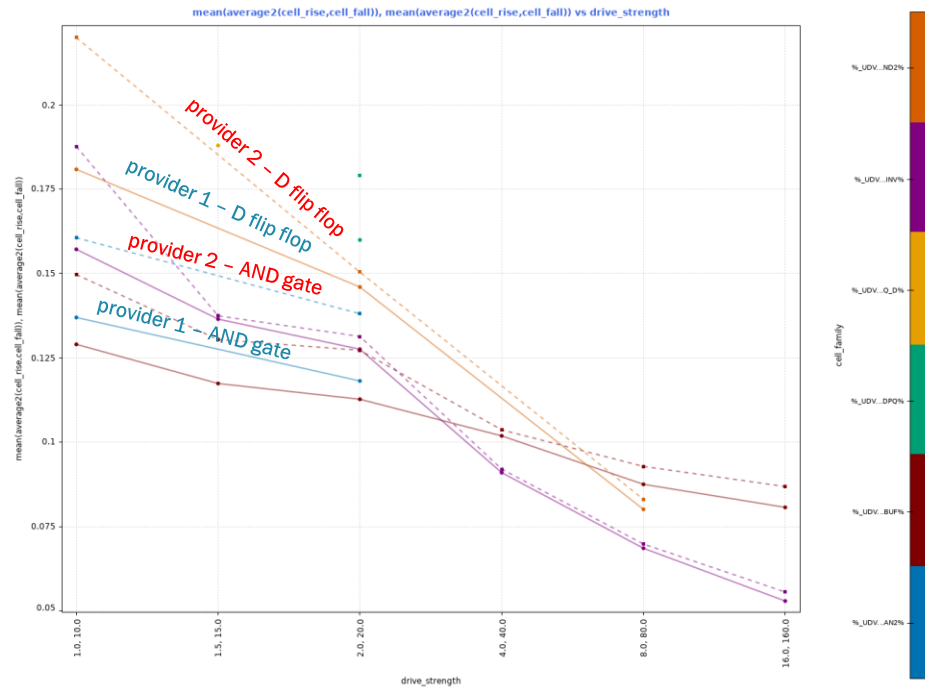
- Lower values = better (less space in use)

Drive strengths in a library allow for improved insight on cell selection prior to design

- More drive strength data = better breadth of PPA = more optimized cell selection

# PPA Plots

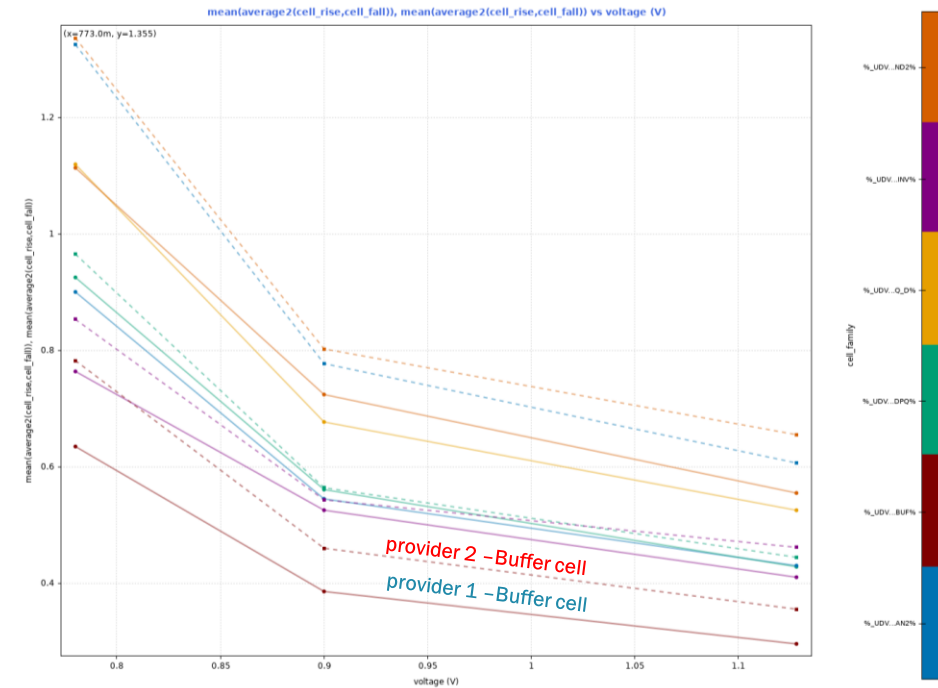
## Performance vs. Drive Strength



Across drive strengths, **provider 1** is consistently “faster” (performs better) than **provider 2** for all cell families. Looking at some specific examples:

- For drive strengths < 2, AND and D flip-flop perform considerably “faster” for **provider 1** than **provider 2**
- For drive strengths > 4, most cells converge, but with **provider 1** still outperforming **provider 2** slightly

## Performance vs. Voltage

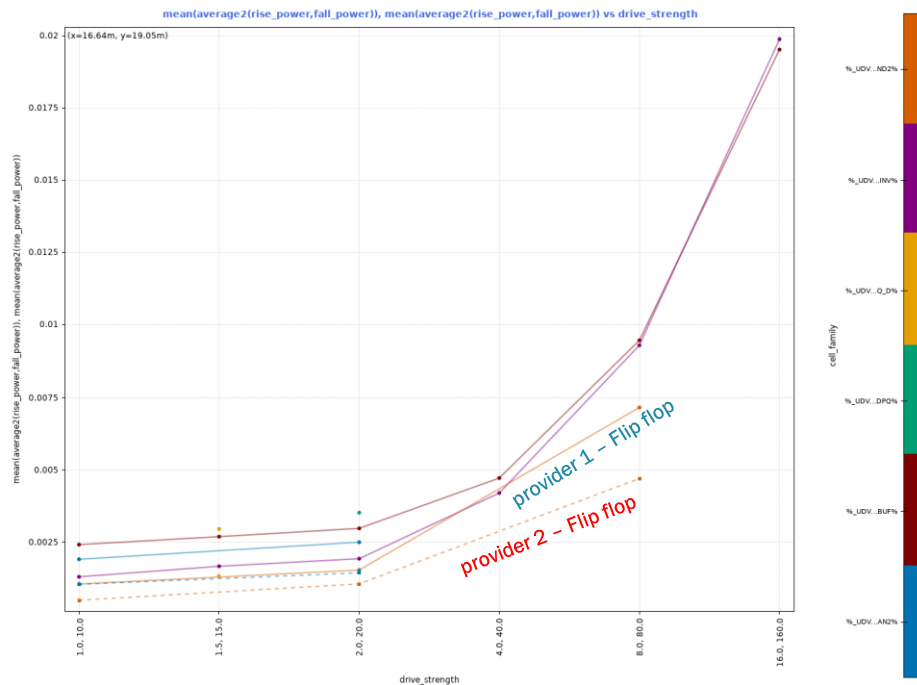


Across voltages, **provider 1** is consistently “faster” (performs better) than **provider 2** for all cell families. Looking at some specific examples:

- The buffer cells show a consistent performance gap of ~0.2ns between **provider 1** and **provider 2**

# PPA Plots

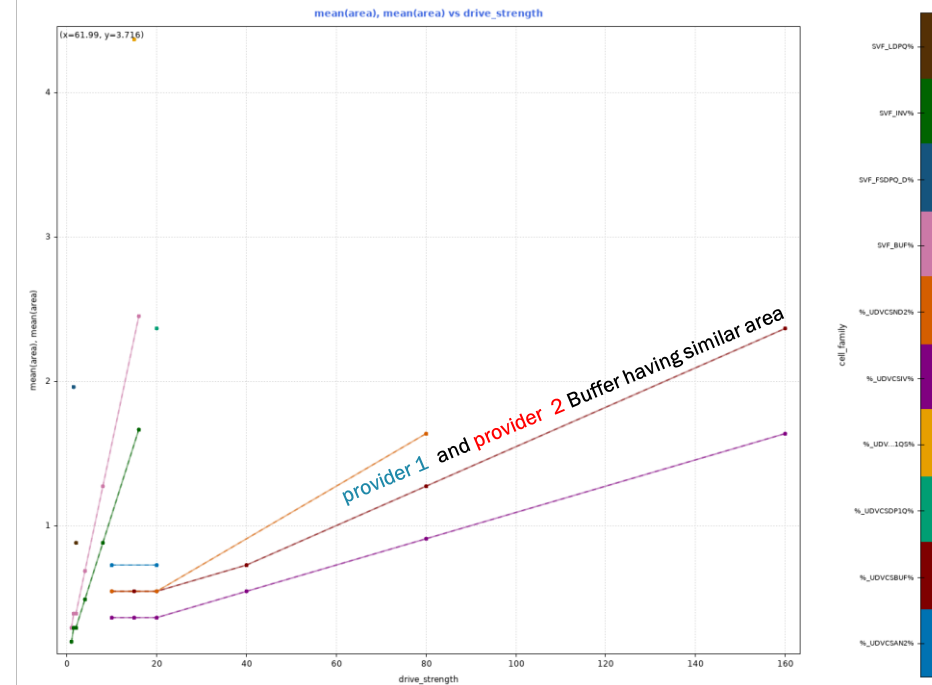
## Power vs. Drive Strength



Across drive strengths, there is sparse Power information for **provider 2**. Looking specifically at the denoted Flip Flop families:

- It can be observed that **provider 2** FF's are more power efficient than **provider 1** FF's

## Area vs. Drive Strength



Across drive strengths, Area is consistently the same across both **provider 1** and **provider 2** cell families.

- This is made apparent by the complete overlap of area trendlines

# Summary

- **Conventional methodology:**

- Conventional method of PPA analysis requires extensive setup and machine intensive steps
- Process is time consuming, can take several months depending on the number of library providers/technologies to be evaluated
- Existing methods require machine-intensive computing to generate STA based results for each library provider/technology

- **Proposed methodology:**

- Uses AI together with Solido Library Profiler to reduce decision-making time from several months to few days
- Does not require expensive production tools/expertise, or intensive compute resources
- Also provides effective method to narrow down the library provider and technology by creating plots/reports for different PPA targets
- Enables faster time to market by reducing the initial library/technology evaluation time

# Future Work

- **Automation**

- Replay of projects with new data to generate plots and reports directly
- Custom tabular reports incorporating user-defined metrics in batch mode

- **Cell Selection**

- Smart selection of cells and weights based on design profile and/or best representative cell families
- Selection of cells based on functionality to identify the representative ones for analysis

- **Reporting**

- Save plot points as tables in reports for relevant templates
- Summary reporting page listing unmatched and matched items with corresponding matching rules