UCLA

Abuttable Analog Cell Library and Automatic AMS Layout

Tianjia Zhou[1], Cheng Chang[1], Li Huang[2], Jingyun Gu[2], Zexin Ji[3], Xiangyang Liu[4], Hailang Liang[5], Zhanfei Chen[5], Ting-Jung Lin[5], Song Wang[4], Na Bai[4], Zhengping Li[4], **Lei He**[1]

[1]University of California, Los Angeles

UCLA

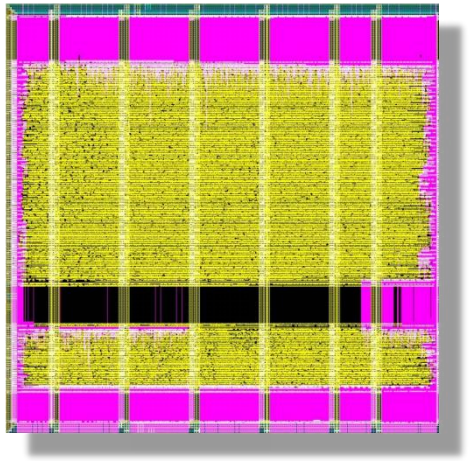
I. Background and Motivation

- Digital APR and standard cells
- Motivation
- Standard cells for analog circuits

Physical Design: Analog and Digital

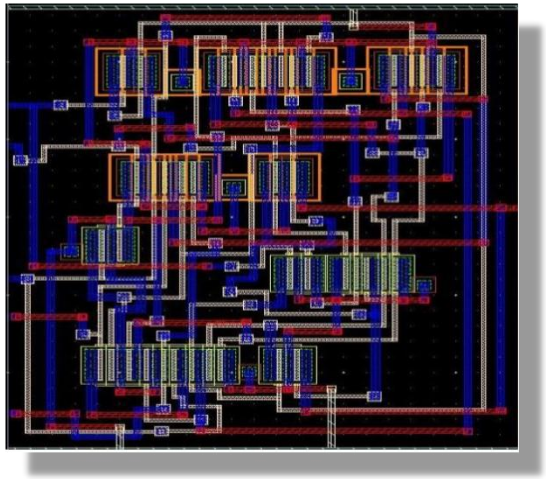
Digital:

- Fully automated
- Market proven APR tools (Innovus, ICC, NitroSoC....)
- Easy to adjust, modify, improve
- Large number of transistors, short time, low cost
- Algorithm-driven

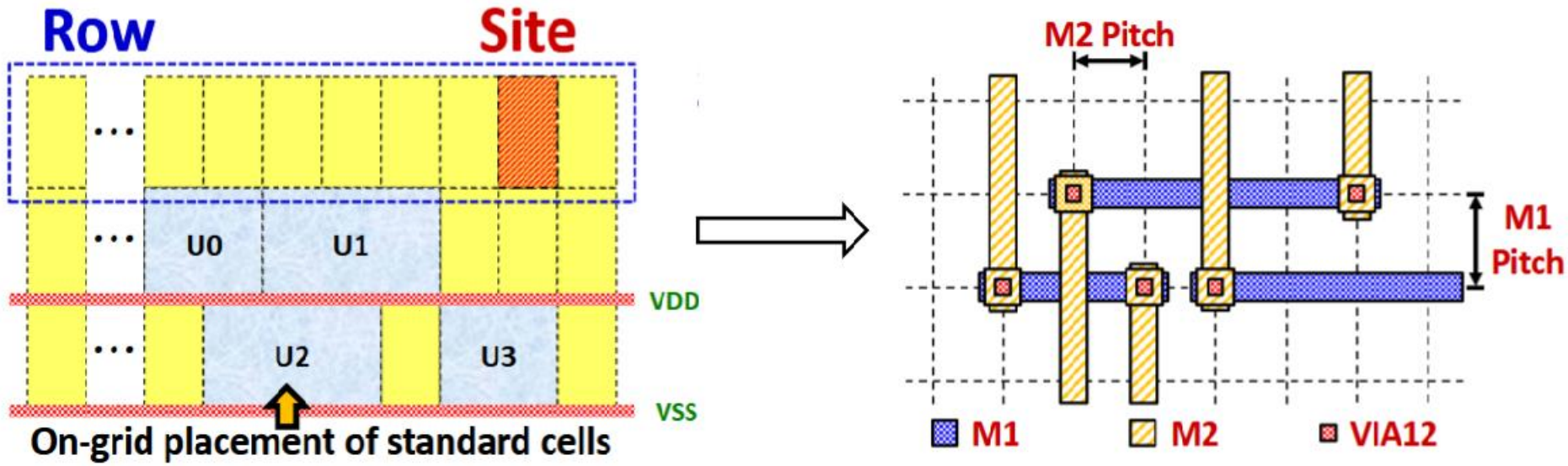


Analog:

- Mainly customly designed
- Difficult to make changes
- Small number of transistors, large amount of time and cost
- Experience-driven



Foundation of digital APR: Standard Cells



- Standard cells are “standard” in that:
  1. Uniform height, shared power/ground rails
  2. Regular width: N × Site
  3. Pins on uniformly-distributed tracks
  4. Uniform PnR granularity (Site & Pitch)
  5. **Abuttable** in any direction (No DRC/LVS violation)
- Compactly placed into a row-based array

Motivation

- Start PnR with analog “standard cells” instead of Pcells
- Acells: abuttable Pcells
- Amacro for abuttable macros.

- Abuttability + regular width and height  DRC clean
- Abuttability with standard cells, ideal for mixed signal circuits.

UCLA

II. Acells and Automatic AMS Layout

- Existing Work: Stem Cells
- Acell Improvements
- Automatic Acell generation
- Place and Route with Acells

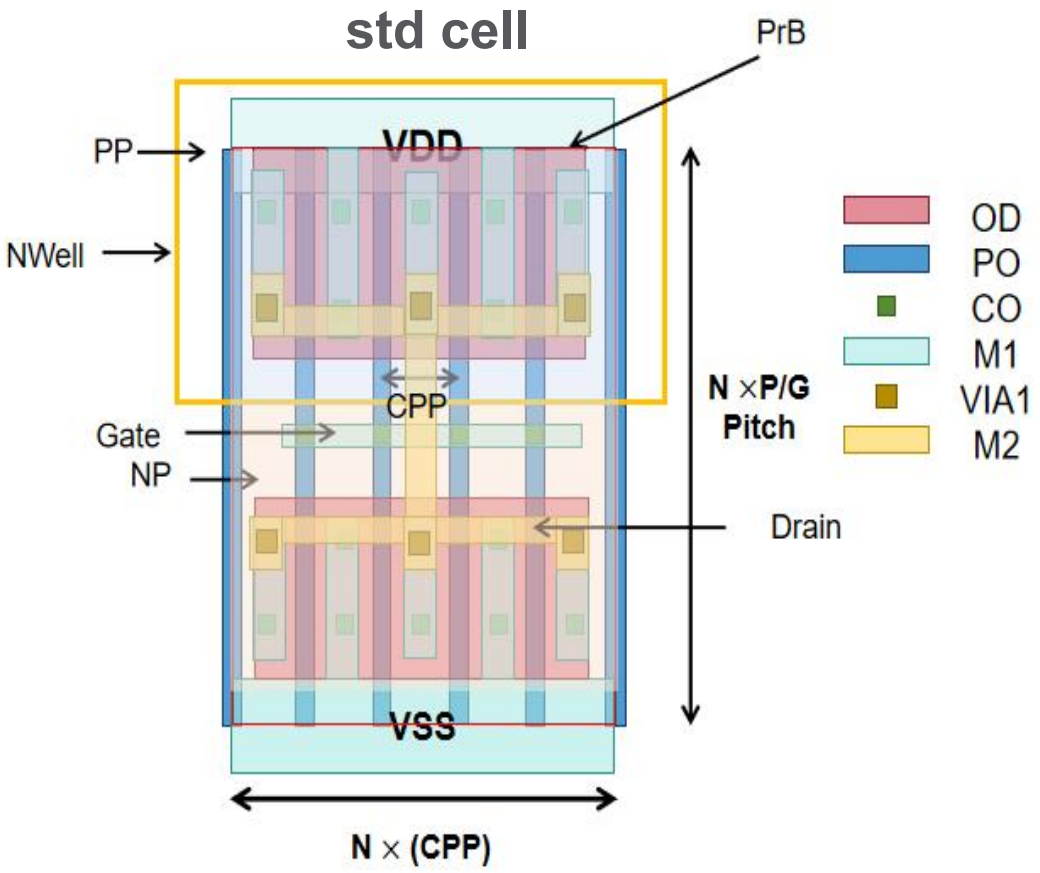
Existing Work: Stem Cell

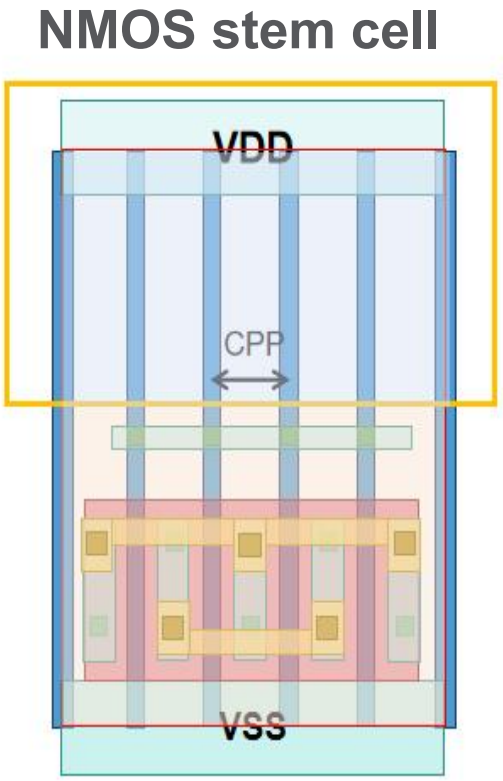
1. Take the simplest standard cell: inverter

2. Remove 1 transistor

3. Create Pins

[Wei & Murmann, TVLSI 2021]



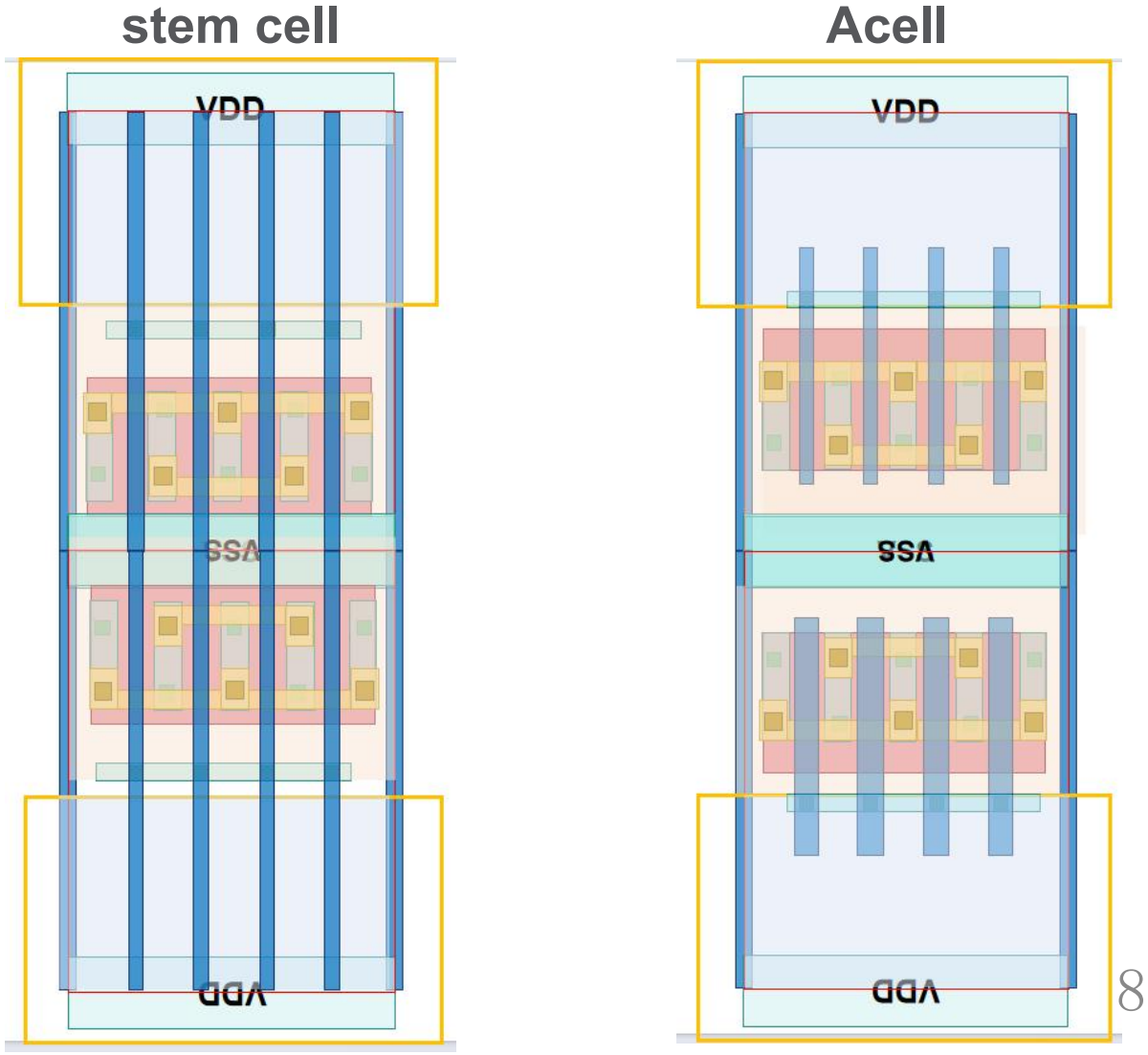


[Wei & Murmann, TVLSI 2021]

Acell Improvement: Variable Channel Length (L)

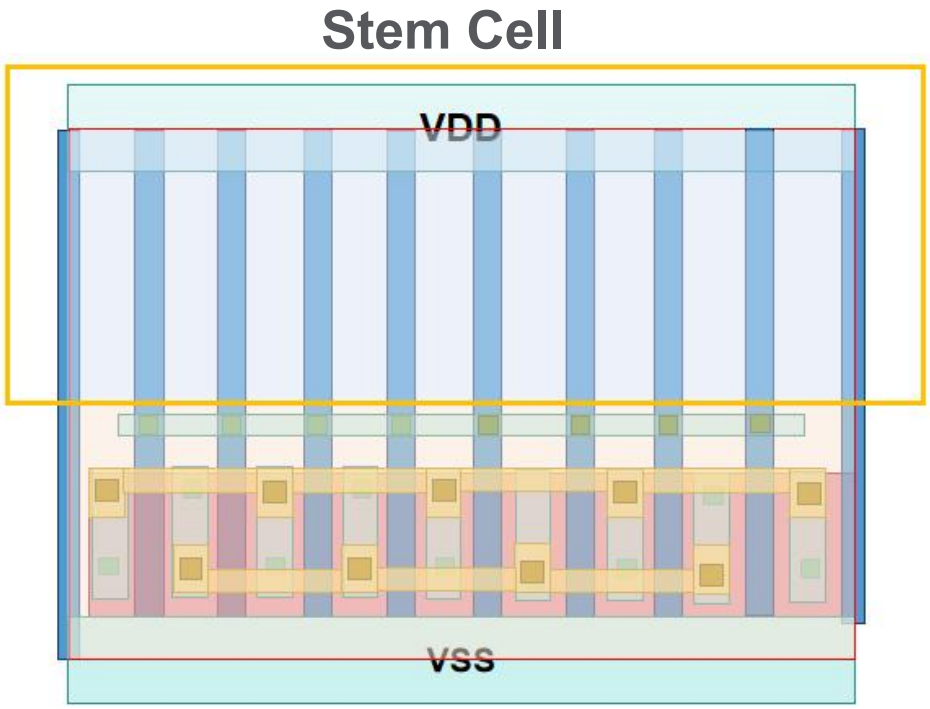
1. All transistors in Std and Stem Cells use the same L (usually minimum L)
    - polys align perfectly when abutted vertically
    - not feasible in analog cases

2. Shortend gate polys + active region moved towards center.
    - polys have enough spacing when abutted vertically
    - reduced gate resistance

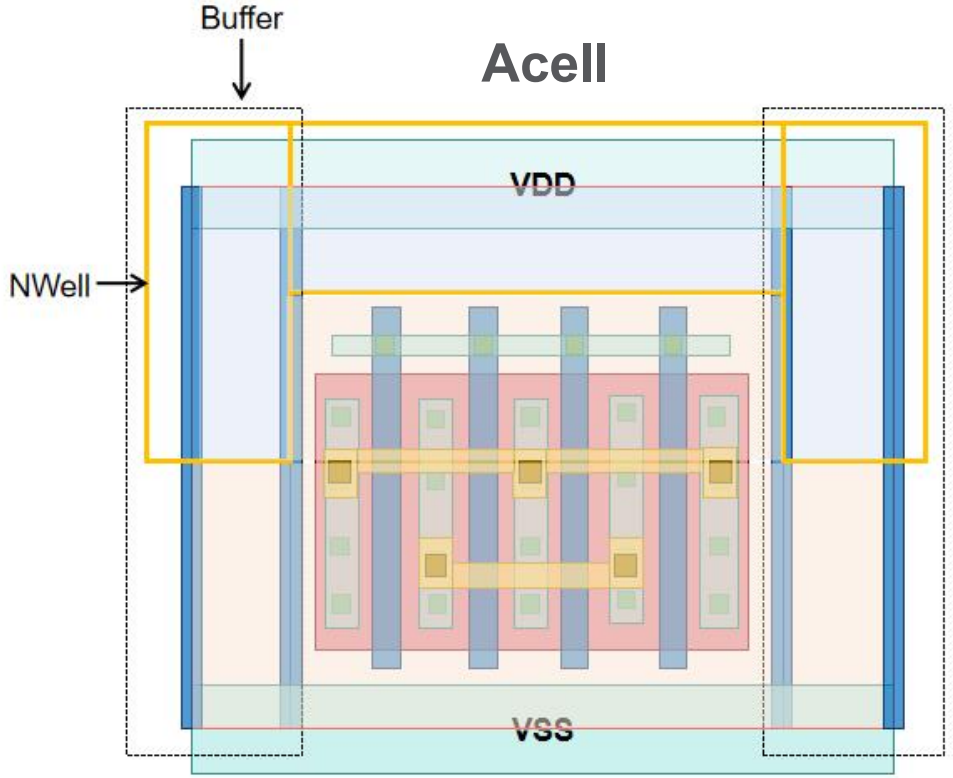


Acell Improvement: Area Reduction

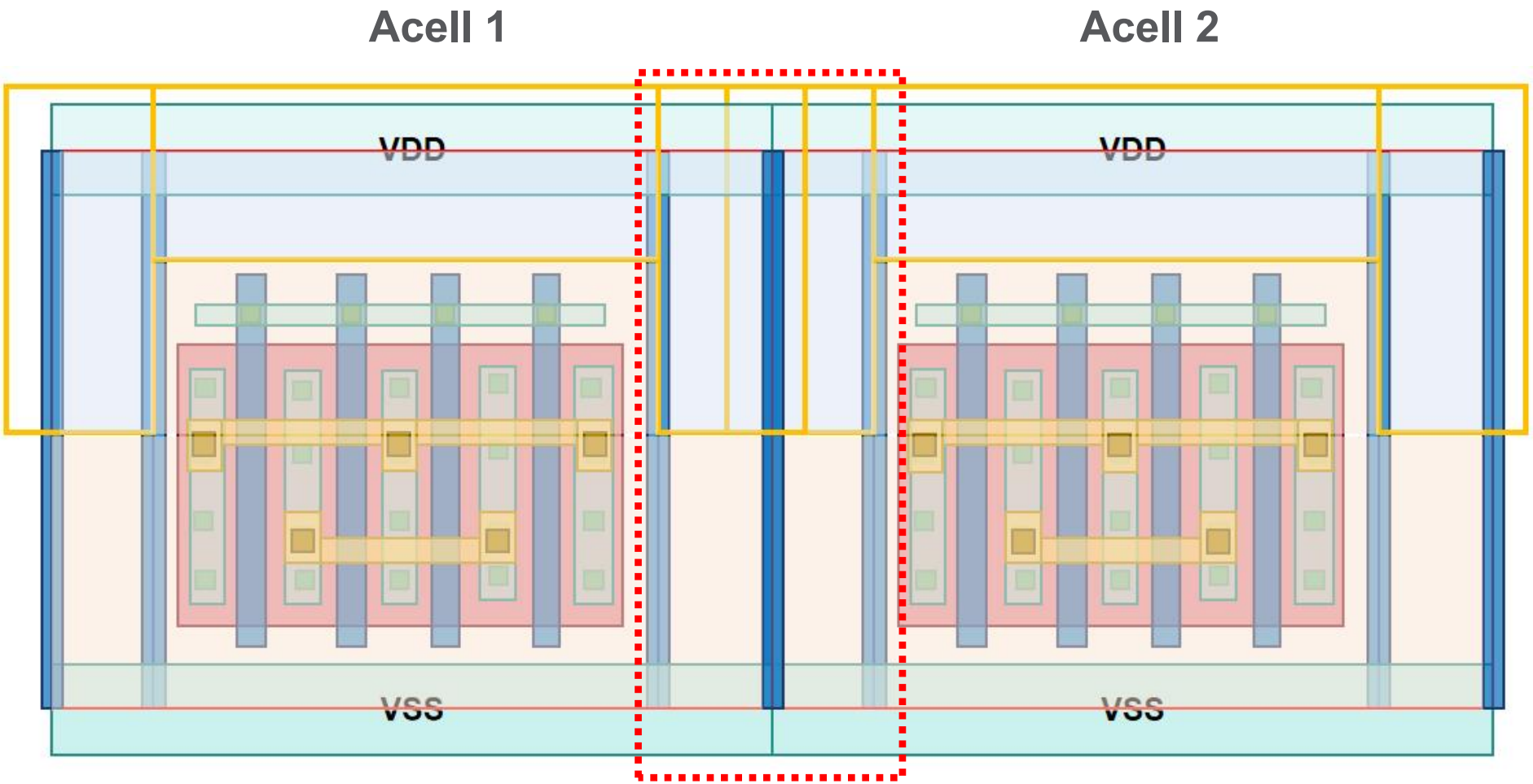
- Half of the area is wasted: low area efficiency
- Maxmimum allowed FW small: large NF needed for ultra wide device



- Rectilinearly reshaped N/P Well and N/P implanted region
- Buffers added **optionally** to maintain abuttability



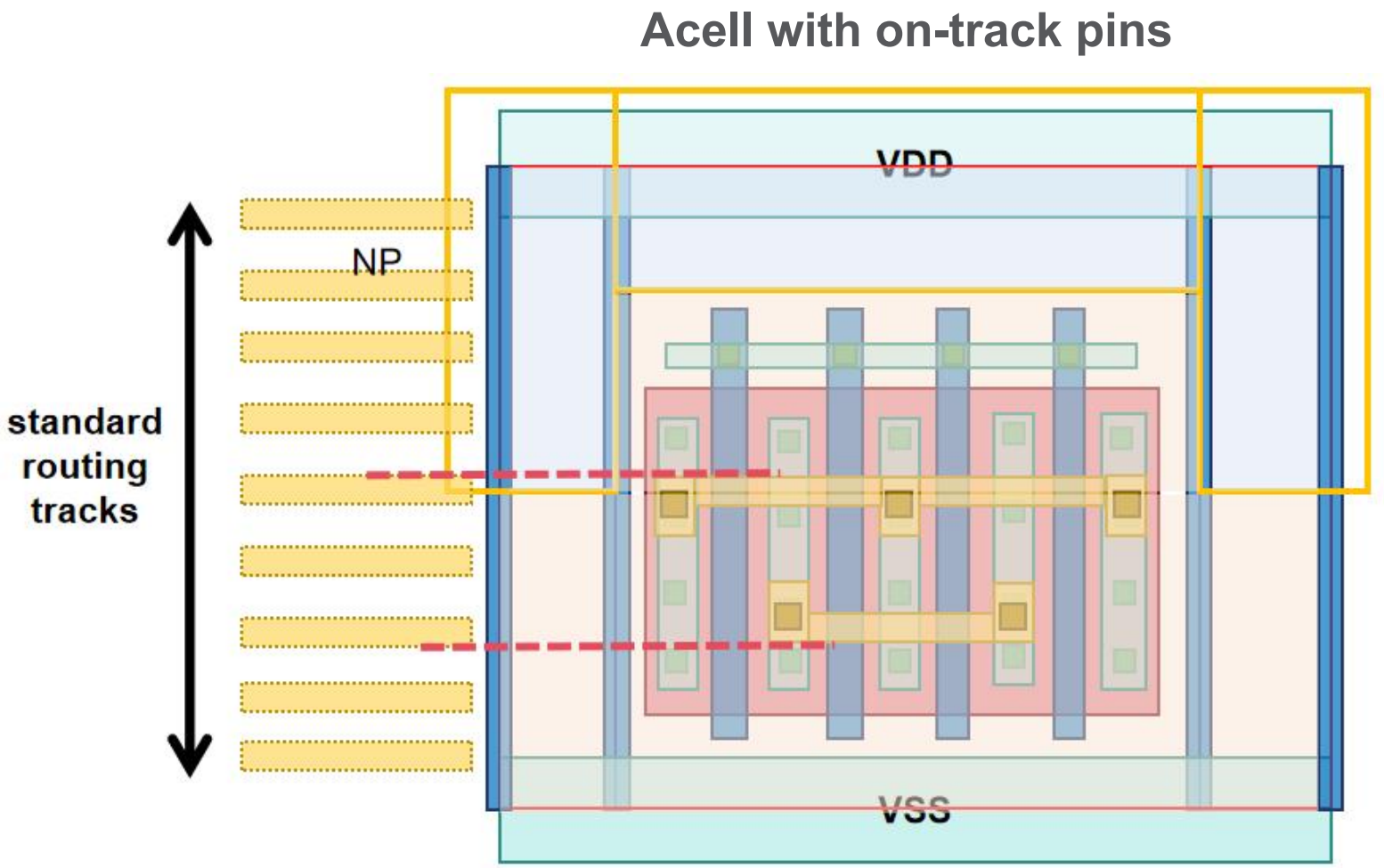
Abuttability of Acells with buffers



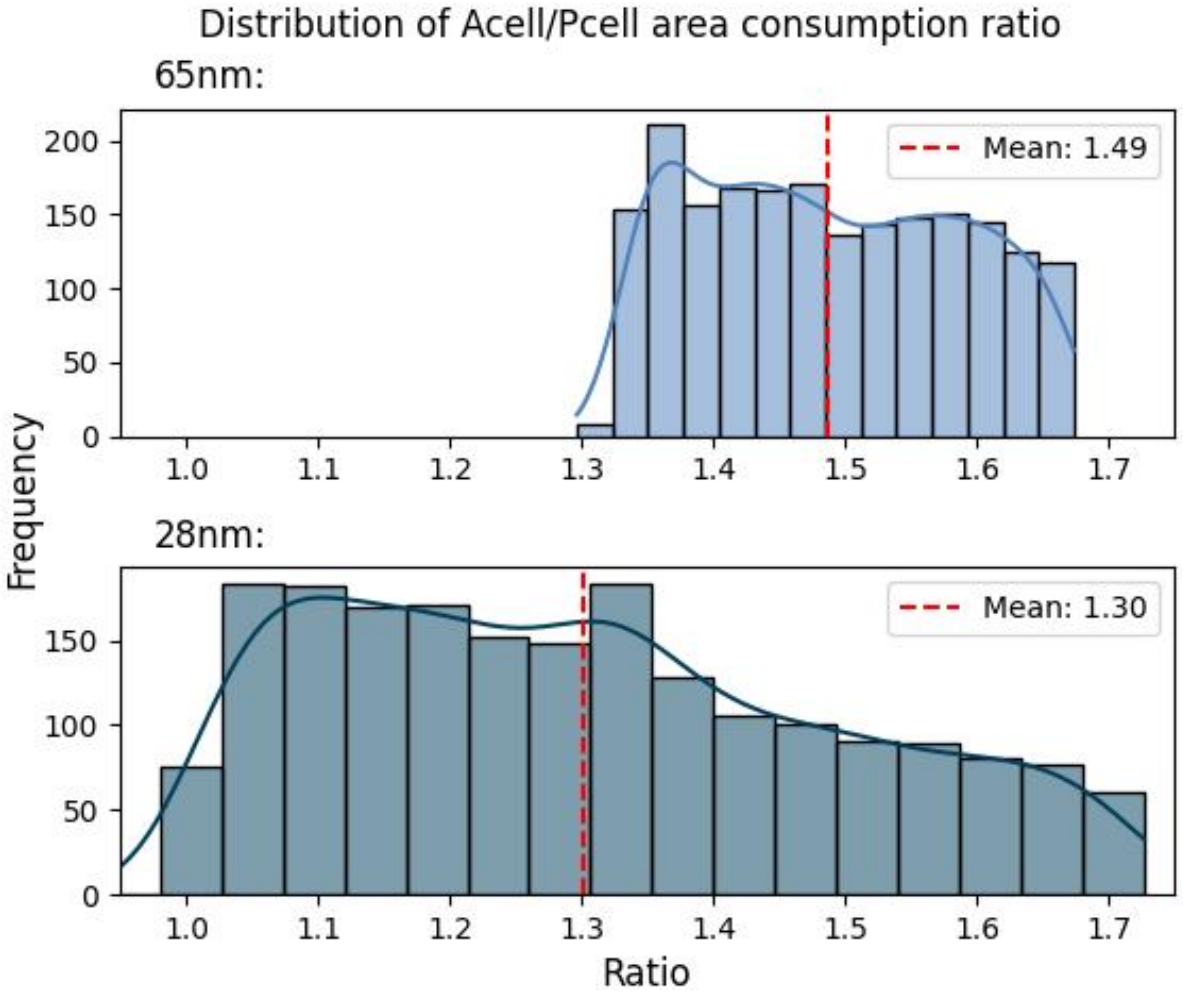
Can be removed and compressed

Acell: Pins on Track

- Gate, Source, Drain pins on standard routing tracks.
- Easier routing
- Easier for automatic cell generation



Acell Area Assessment



- Randomly generate 2000 parameters, compare area consumption
  - NF$\in [2,100]$
  - L$\in [30n, 1u]$
  - FW$\in [100n, 1u]$
- In 28nm 9-track tech, Acells cost ~1.3x area than directly using Pcell
  - Stem cells have a relative constant ratio of ~2
- Expected to be more area effecient when scaling down
- ~1 if FW is small and NF is large

Global Area and Cell Area

- Acells waste 30% area than Pcells ?
  - Wasted area mainly comes from “doubling NF to accomodate large FW”
  - Almost no area wasted if FW is near the maximum -> Can be optimized during schematic design

- Acell-based layouts have comparable area with Pcell-based:
  - Abuttability and regularity enables easy placement optimization
  - Dummy structures can be merged

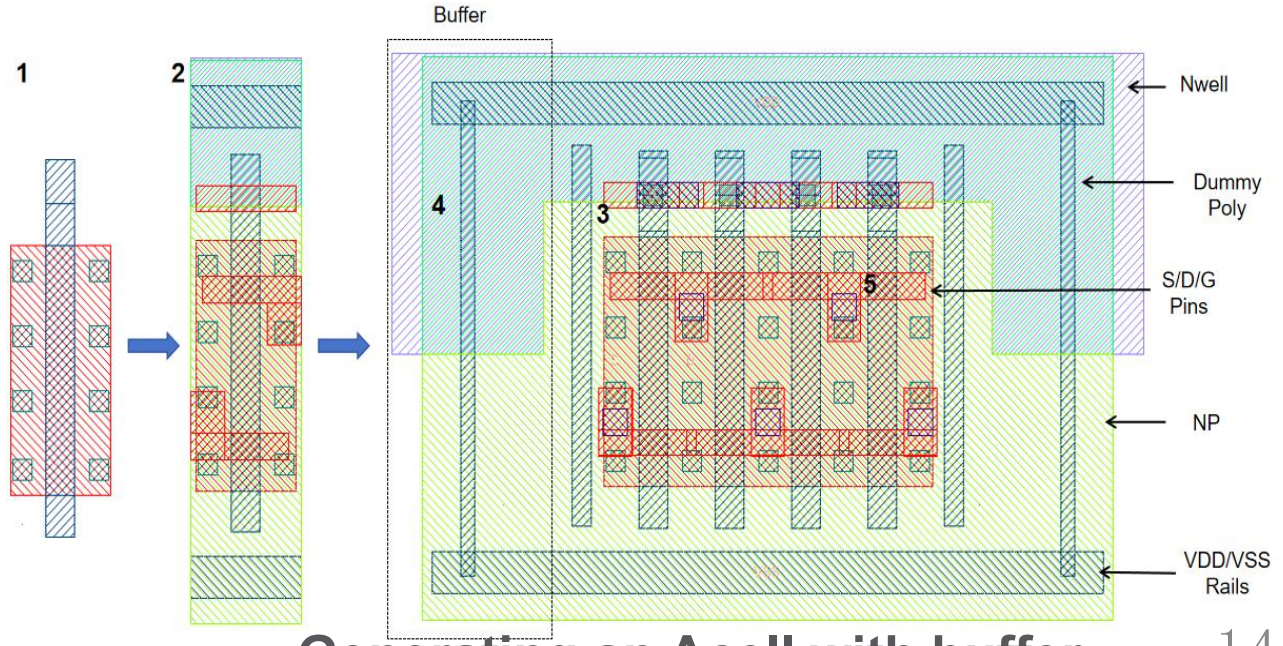
Acell Automatic Generation

- Continuous change in finger width (FW) and channel length (L) => Large number of Acells needed for one PDK
- Acell = Pcell + redundant complementary layers + packaging
  1. Start with a Pcell
  2. Add P/G rails, complementary layers, and metal shapes reserved for the pins
  3. Replicate for NF times
  4. Add side buffering structures (optional)
  5. Add Vias and label the pins.

- each cell 7~8 min (python+SKILL),

~10 sec (C++)

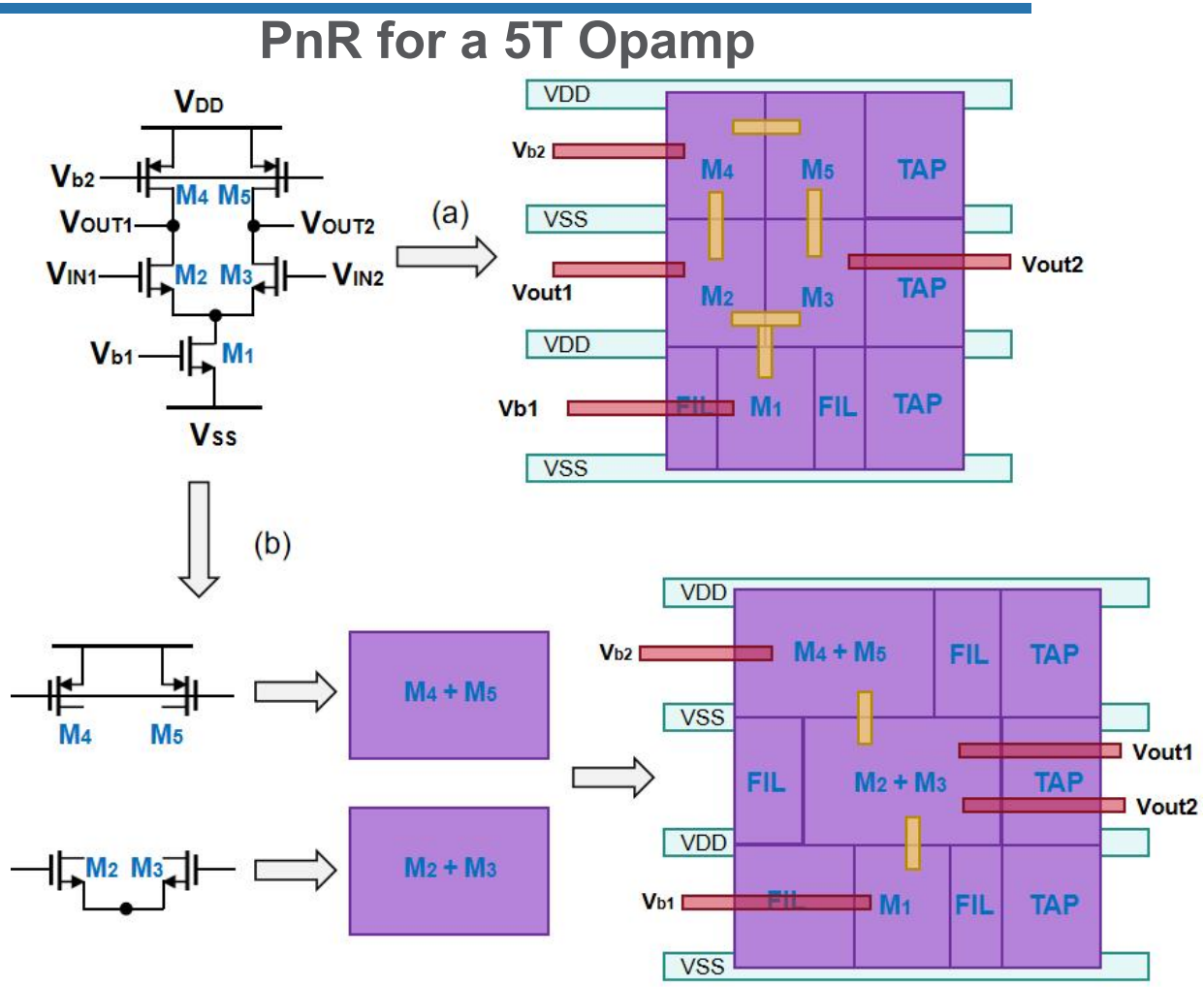
- Generate >10K cells for a PDK within hours



Generating an Acell with buffer

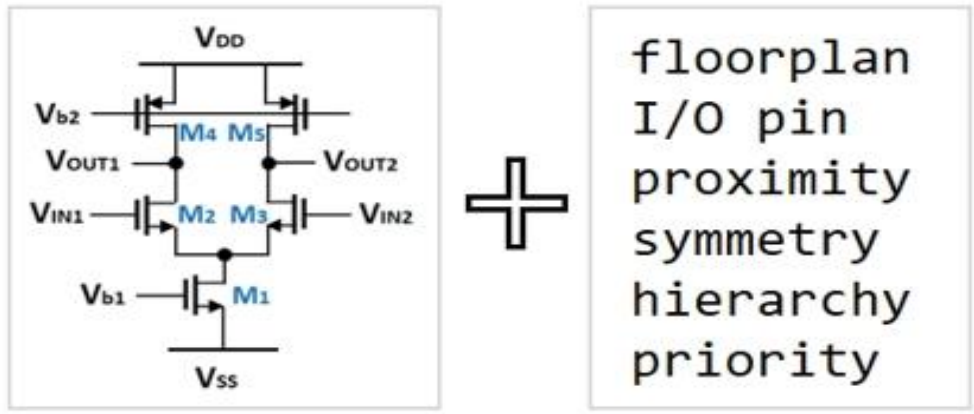
Placing and Routing with Acells

- Direct use of digital APR tools
- Connectivity described using gate-level verilog
- PnR engine place and route all the cells
- TAP and FILLER added automatically
- Mismatch/Symmetry can be respected through hierachy PnR



Place and Route Flow

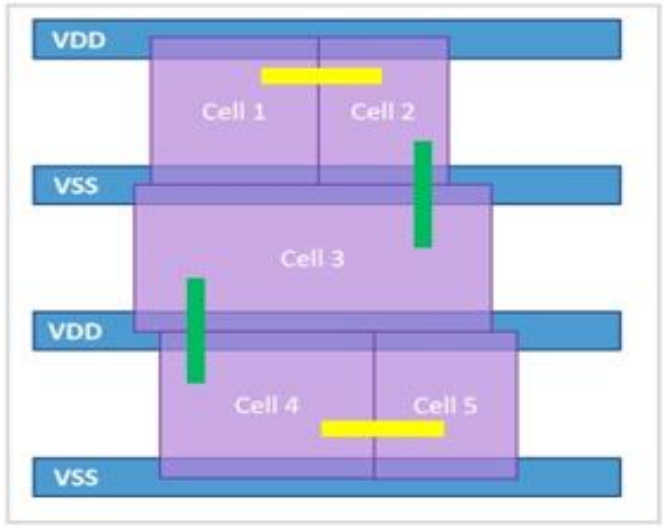
Acells



Schematic

floorplan
I/O pin
proximity
symmetry
hierarchy
priority

Constraints



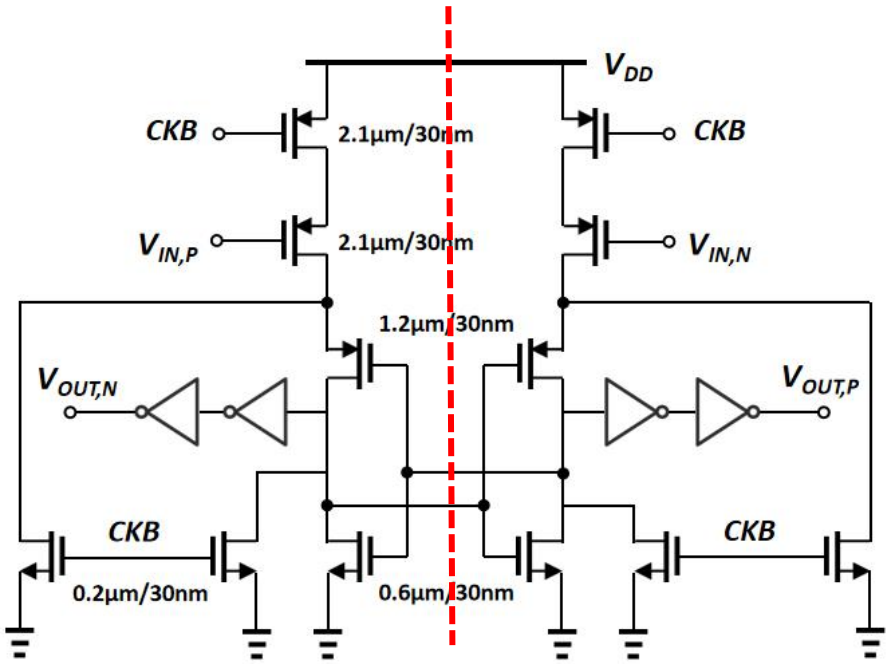
Auto P&R with digital-like P&R

III. Experiments and Discussions

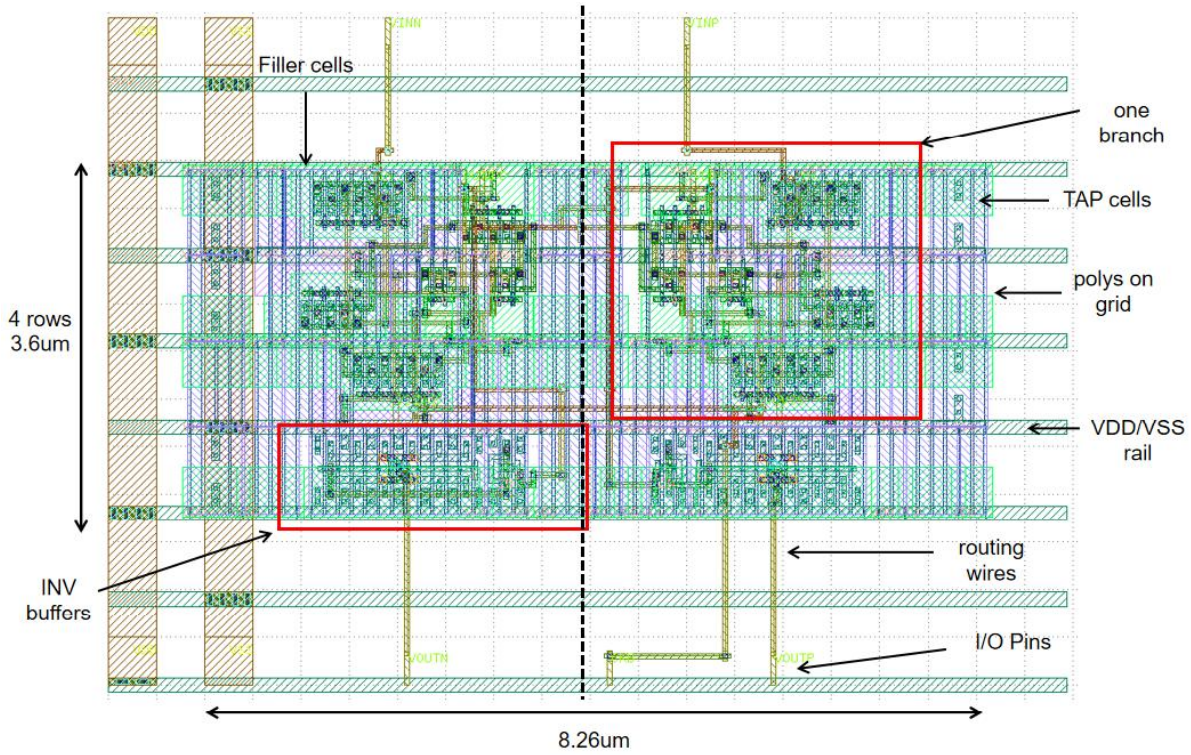
- Verified in 65nm and 28nm PDKs
- PnR used Cadence Innovus

Experiments: Comparator

schematic



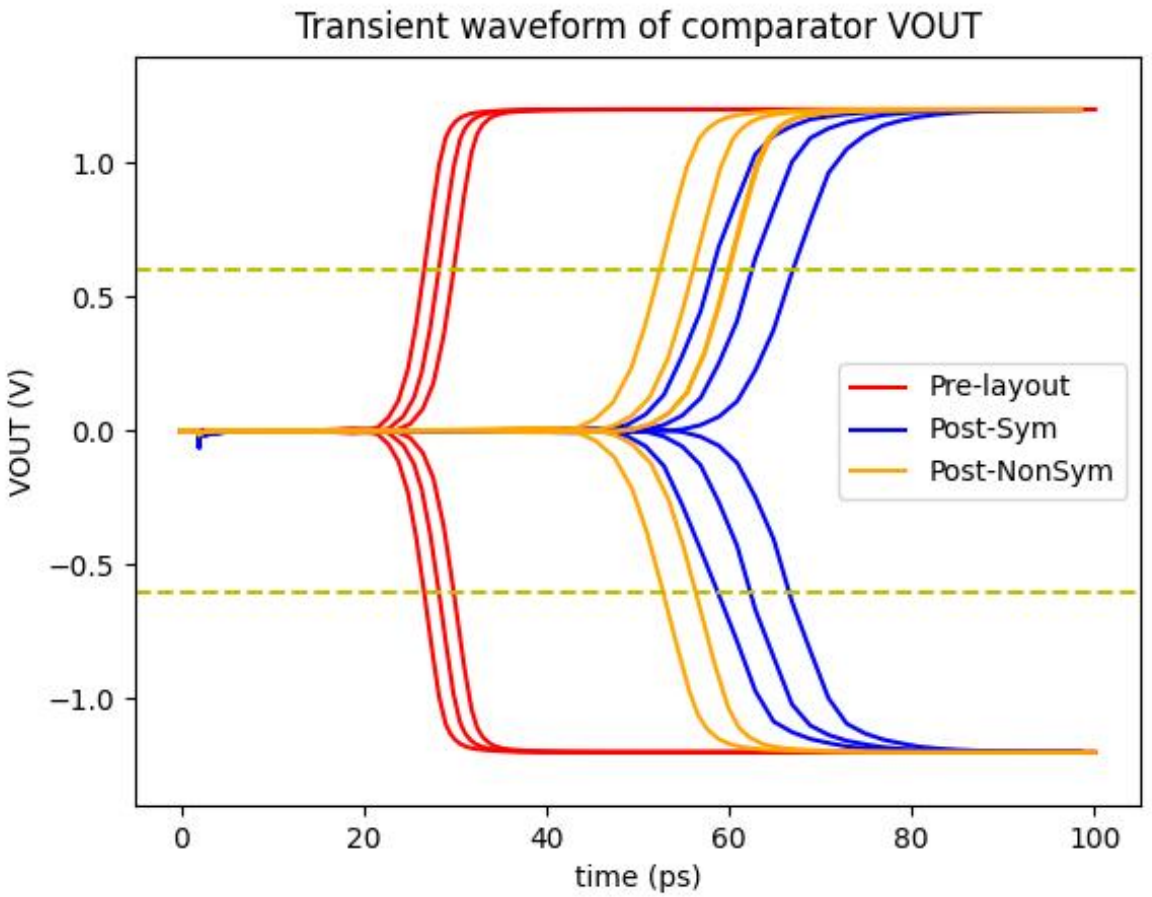
generated layout



- Two branches are symmetrical using hierarchical PnR
- Output nodes are set to have top PnR priority
- Digital cells and Acells are PnRed simultaneously

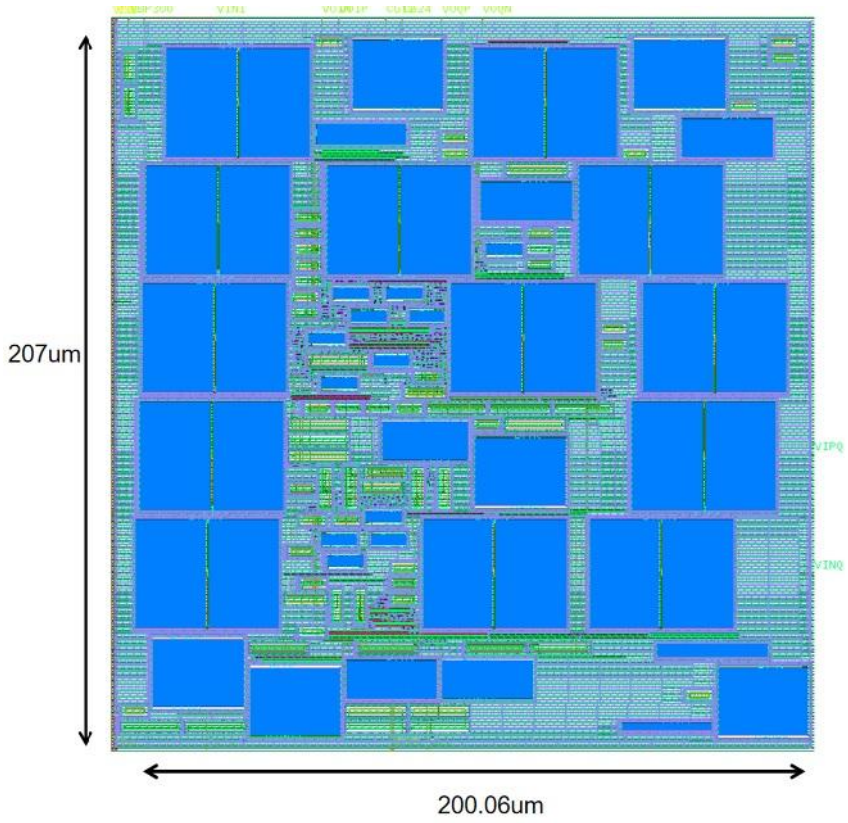
Symmetry-Speed Trade-off

- One-time PnR results in more compact layout ($5.88\mu m \times 3.6\mu m = 21.168\mu m^2$)
  - faster speed: delay ~ 50ps
  - larger offset: V_{offset} ~ -2.55mV
- Hierachical PnR results in more symmetric layout ($8.26\mu m \times 3.6\mu m = 29.736\mu m^2$)
  - slower speed: delay ~ 60ps
  - smaller offset: V_{offset} ~ 1.52mV



Experiment: PGA

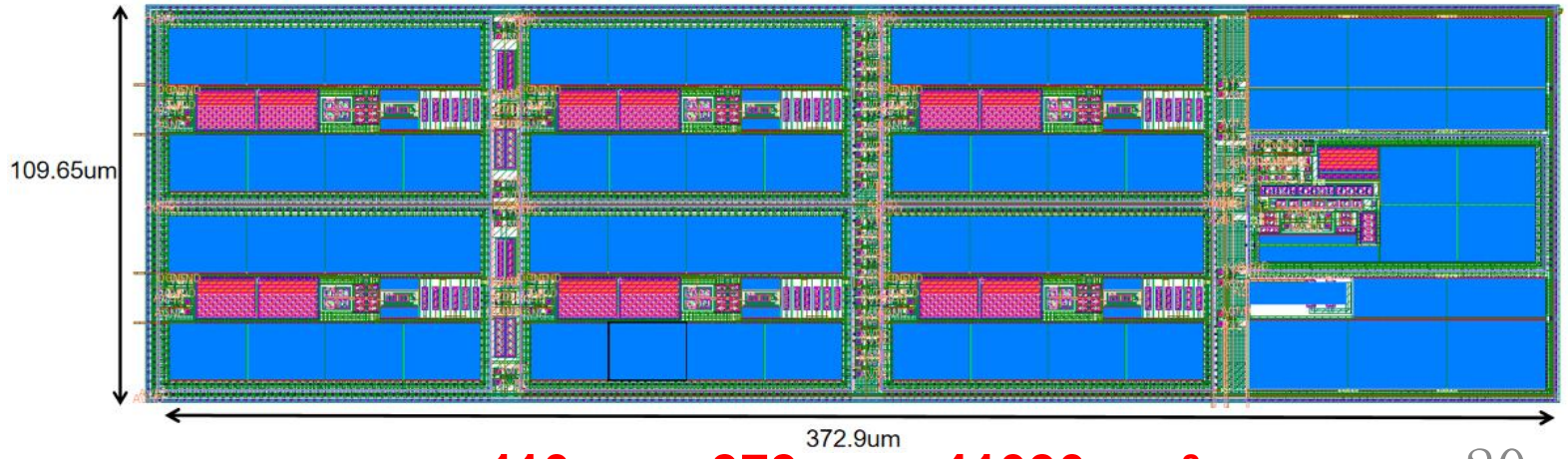
generated layout with Acells



207um x 200um = 41400 um²

- The large capacitor Acells are manually optimized after initial placement by Innovus
- Comparable total area with manual layout, but better aspect ratio

manual layout with Pcells

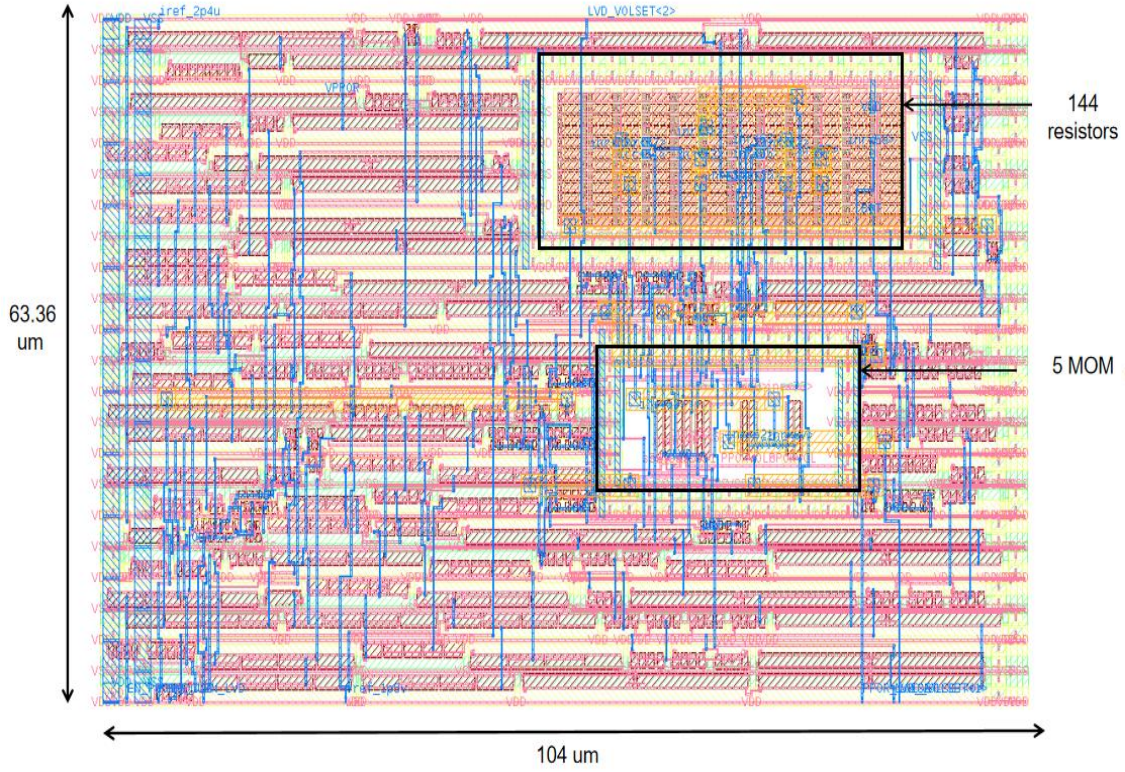


110um x 373um = 41030 um²

Experiment: POR (65nm industrial case)

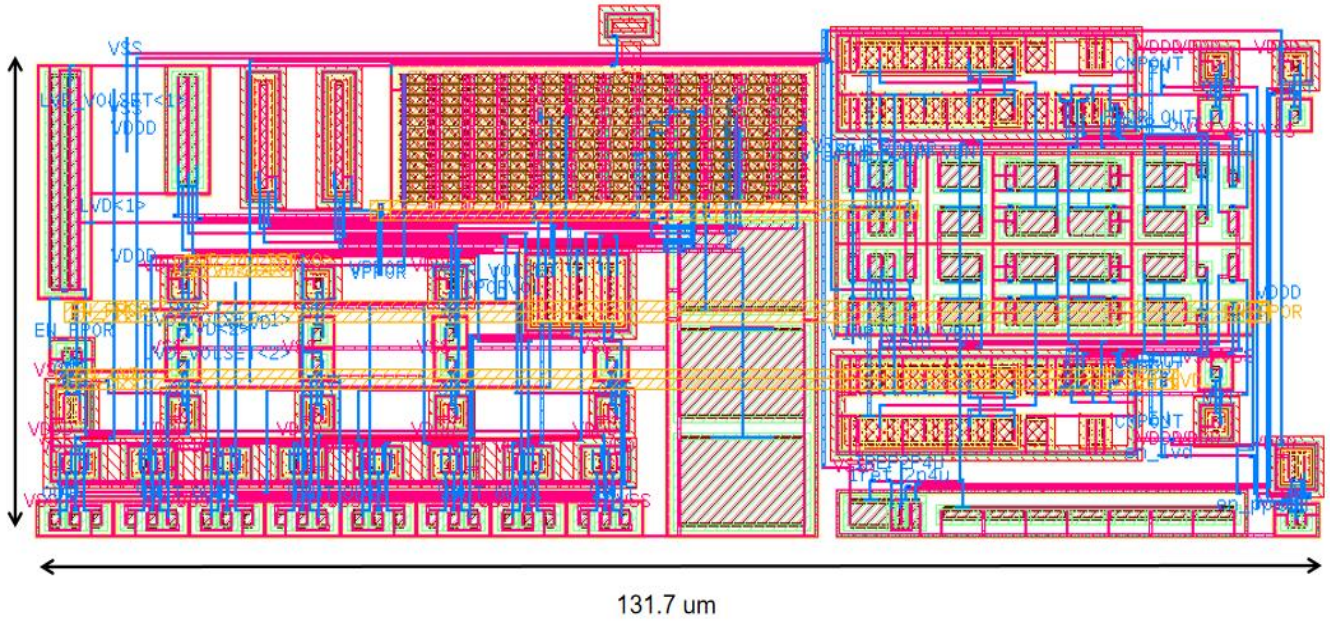
- Resistor Acell array is placed first and dummy structures merged

generated layout with Acells



63um x 104um = 6552 um²

manual layout with Pcells



49um x 132um = 6468 um²

Place and Route Summary

Table 1: Summary of Place and Route performances

Circuit	# of cells	PnR time (sec)	Total Routing Wire (μm)	Layout area (μm^2)
Comparator	20	69.9	77.75	29.808
Opamp	12	70.9	75.76	65.52
S/H	12+3	36.3	108.355	408.24
PLL	39	74.6	122.472	274.88
PGA	338+100	208	9566	41442
POR	866 + 200	132	5174	6589

- All PnR completed within minutes
- All layouts generated are DRC/LVS clean

Conclusions and Future Work

Conclusions

- Acells: abuttable Pcells, analog counterpart for digital std cells
- Acell generation is fully automated
- Full layouts have comparable area with manual layouts

Future Work

- Improved cell: almost no overhead area consumption at cell level
- Multi-row cell: improved performance on ultra-wide transistors

UCLA

Thank You

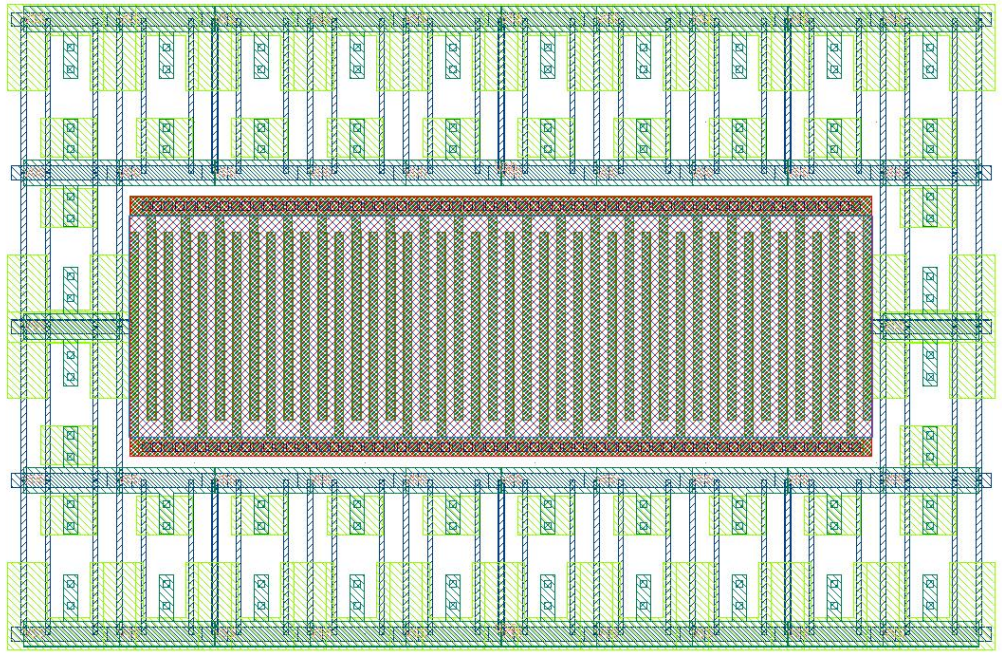
zhoutj@g.ucla.edu

Tianjia Zhou[1], Cheng Chang[1], Li Huang[2], Jingyun Gu[2], Zexin Ji[3], Xiangyang Liu[4], Hailang Liang[5], Zhanfei Chen[5], Ting-Jung Lin[5], Song Wang[4], Na Bai[4], Zhengping Li[4], **Lei He**[1]

[1]University of California, Los Angele

Acells for Passive Units

Acell for cfmom cap



- Example: CFMOM Capacitor Acell
- Pcell at the core
- Surrounded with dummy standard cells (e.g. Filler, Tap)
  - Legal place and route boundary
  - Abuttability
  - Isolation
- Dummy cells can be merged when abutting with other passive unit Acells

Manual layout and simulation results

- Comparable total area consumption with manual layout
- Affordable extra delay in signal generation