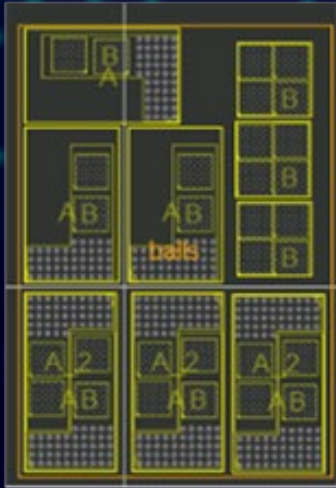


Incorporating Hierarchical Construction for Advanced IC Packaging



Chris Cone, Ed Hudson – Siemens EDA, EBS Division

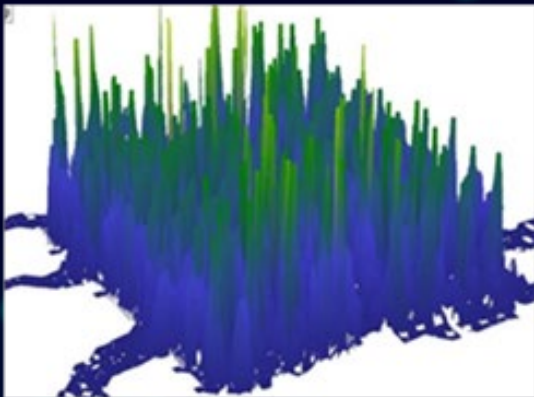
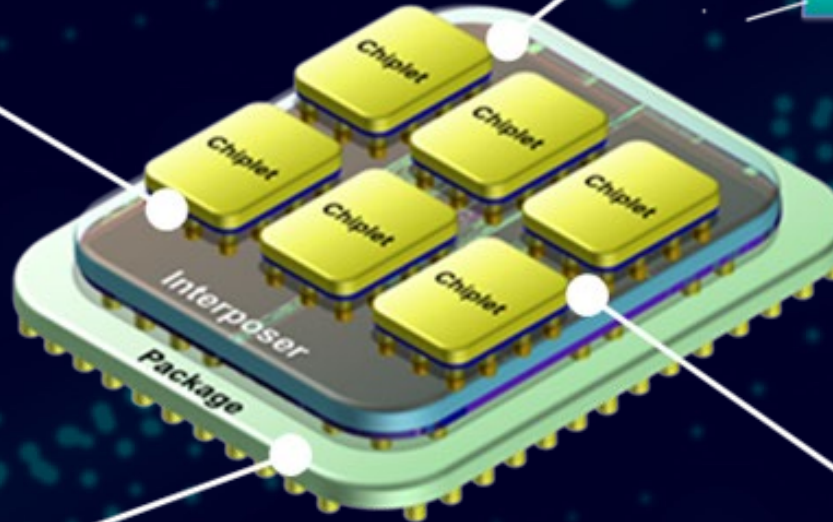
Hierarchy in IC Package Design



- LEF cell library definition
- DEF represents designs with multiple levels of hierarchy



- Technology constraint driven route planning
- Netclass partitioning



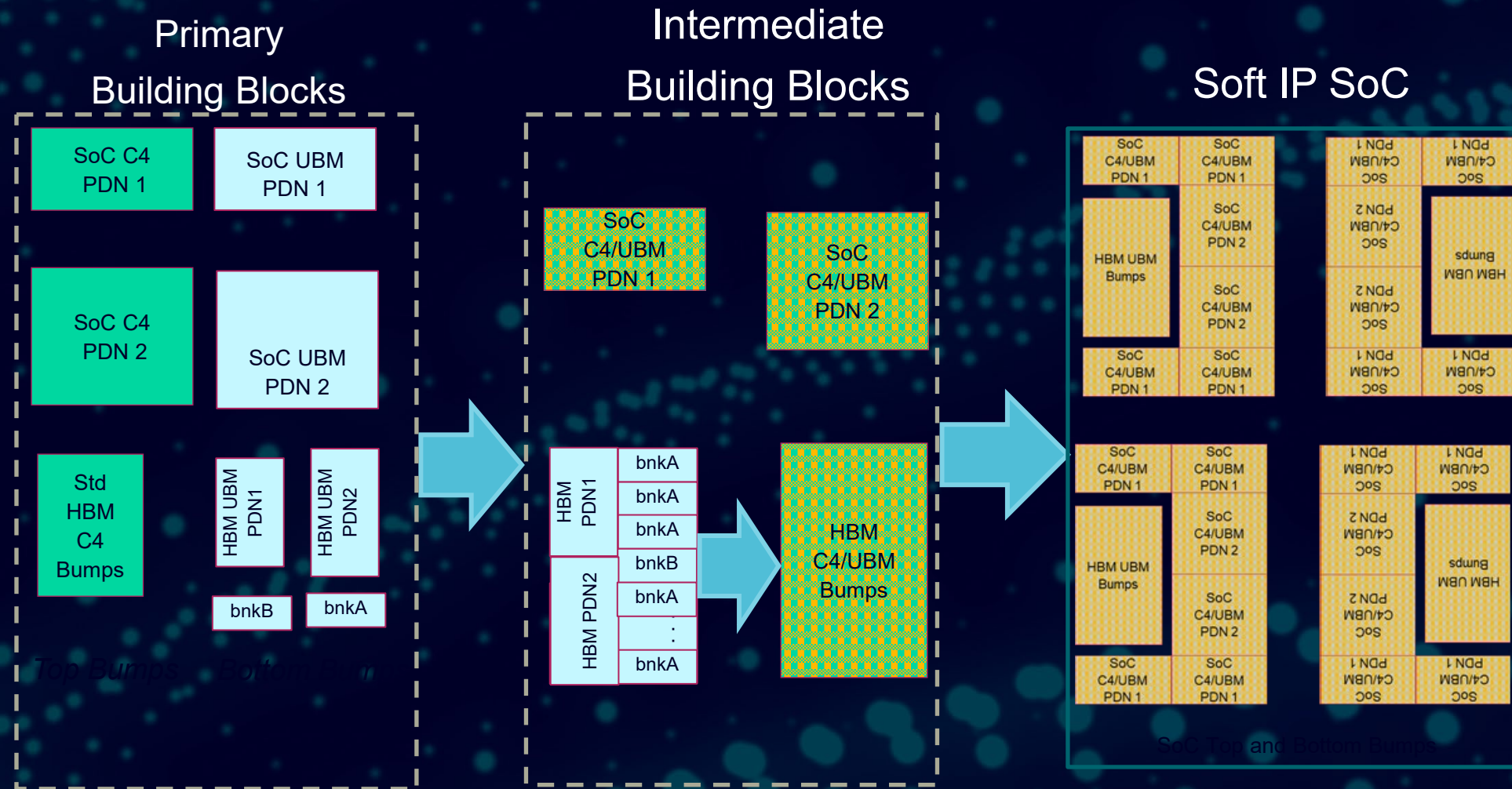
- Material based models for stress and thermal
- Hierarchical circuit models in SI/PI

```
ASIC U1 (
  .TX_N(U1_U2_N),
  .TX_P(U1_U2_P),
  .RX_N(RX_N),
  .RX_P(RX_P),
  .PILSYNC_IN(U2_U1_SYNC),
  .PILSYNC_OUT(U1_U2_SYNC),
  .CLK(sysCLK),
  .RESET(sysRESET)
)

// Instantiate U2 ASIC
ASIC U2 (
  .TX_N(TX_N),
  .TX_P(TX_P),
  .RX_N(U1_U2_N),
  .RX_P(U1_U2_P),
  .PILSYNC_IN(U1_U2_SYNC),
  .PILSYNC_OUT(U2_U1_SYNC),
  .CLK(sysCLK),
  .RESET(sysRESET)
)
```

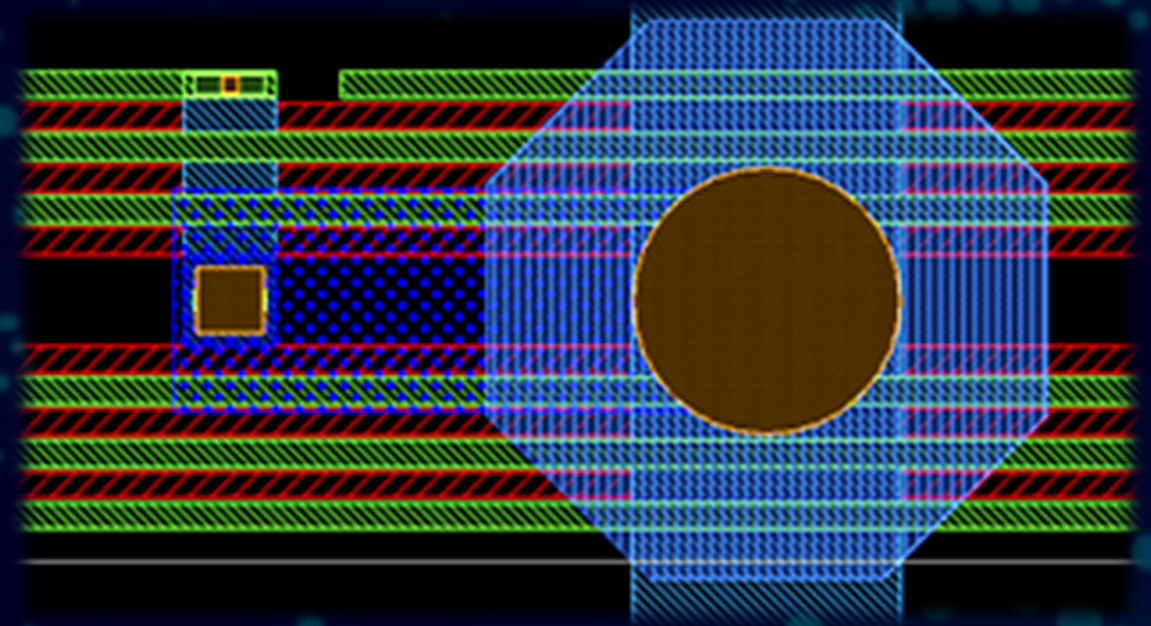
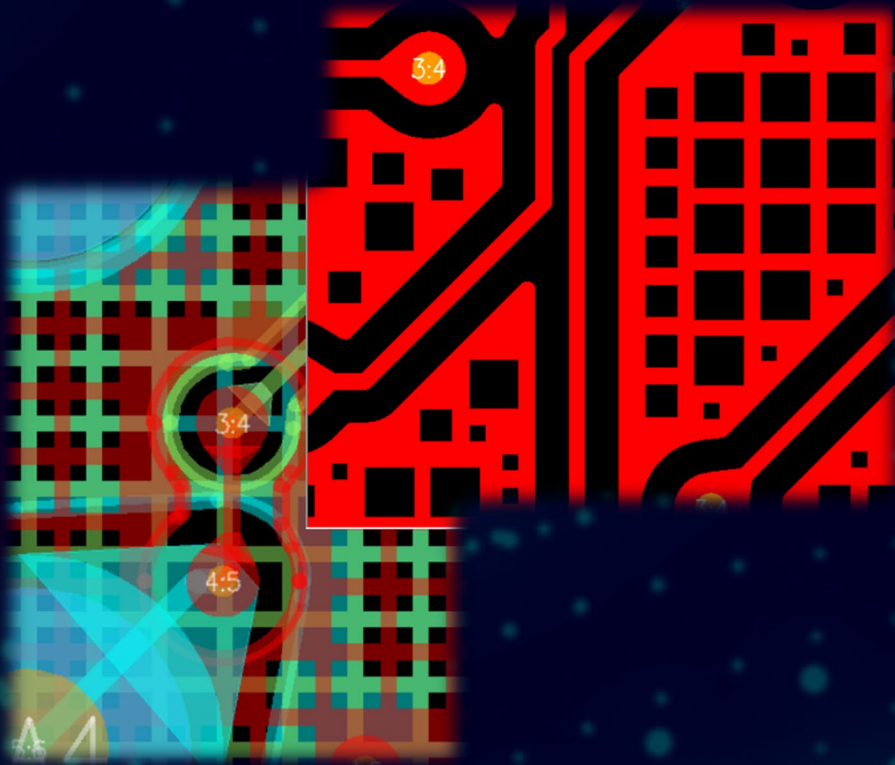
- Hierarchical block design creation
- RTL synthesis to standard cell library

Hierarchy with Building Blocks



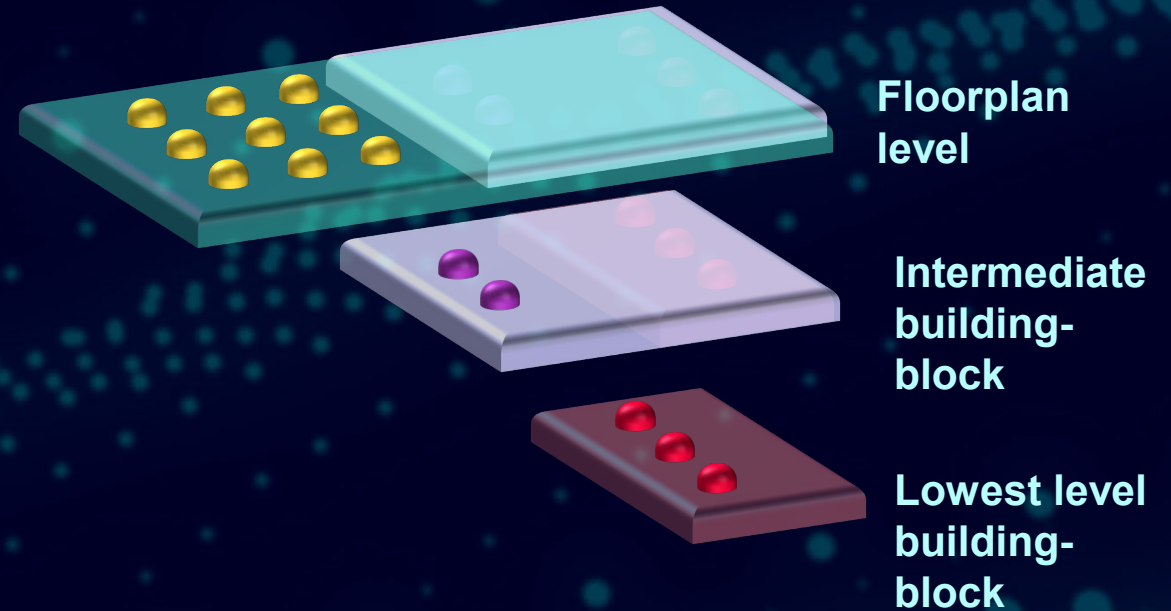
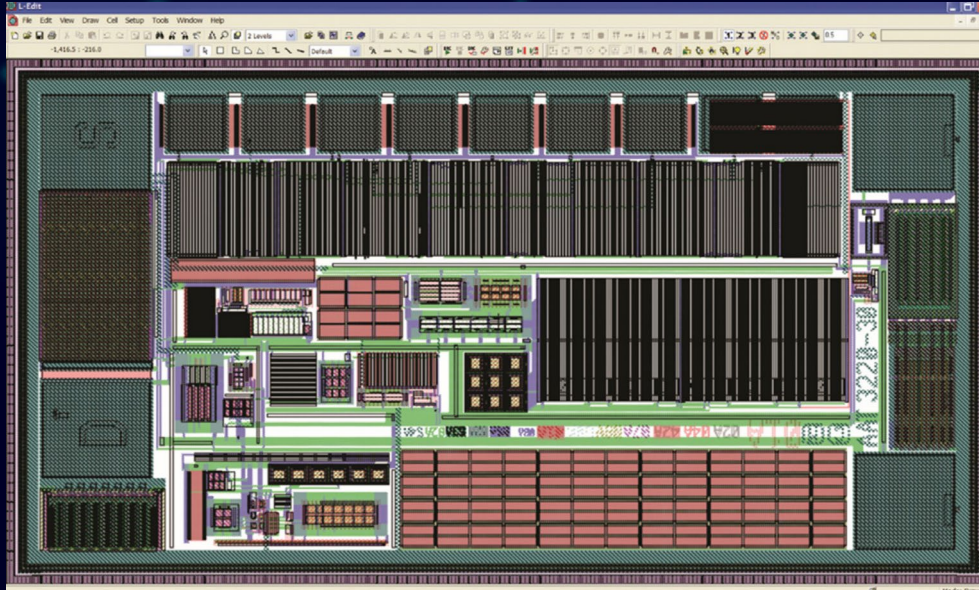
Hierarchy with Parameterization

Dynamic planes for power/ground networks –
parameterized by constraints and plane class



Technology driven bumps and vias with
constraint driven (parameterized) trace routing

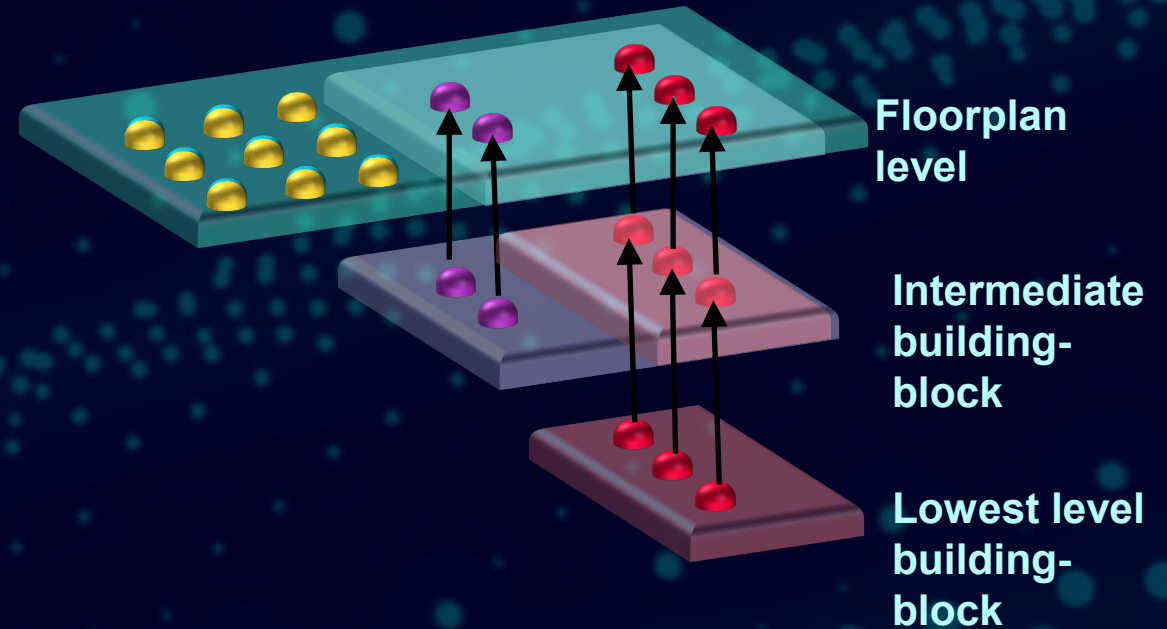
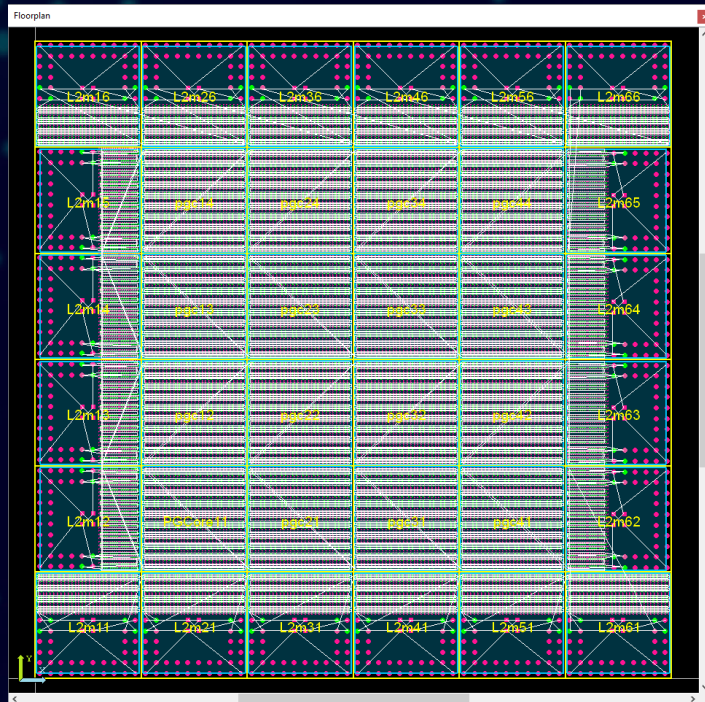
Connectivity in IC Design



Key Attribute

Connectivity remains in hierarchical context unless promoted by the tool

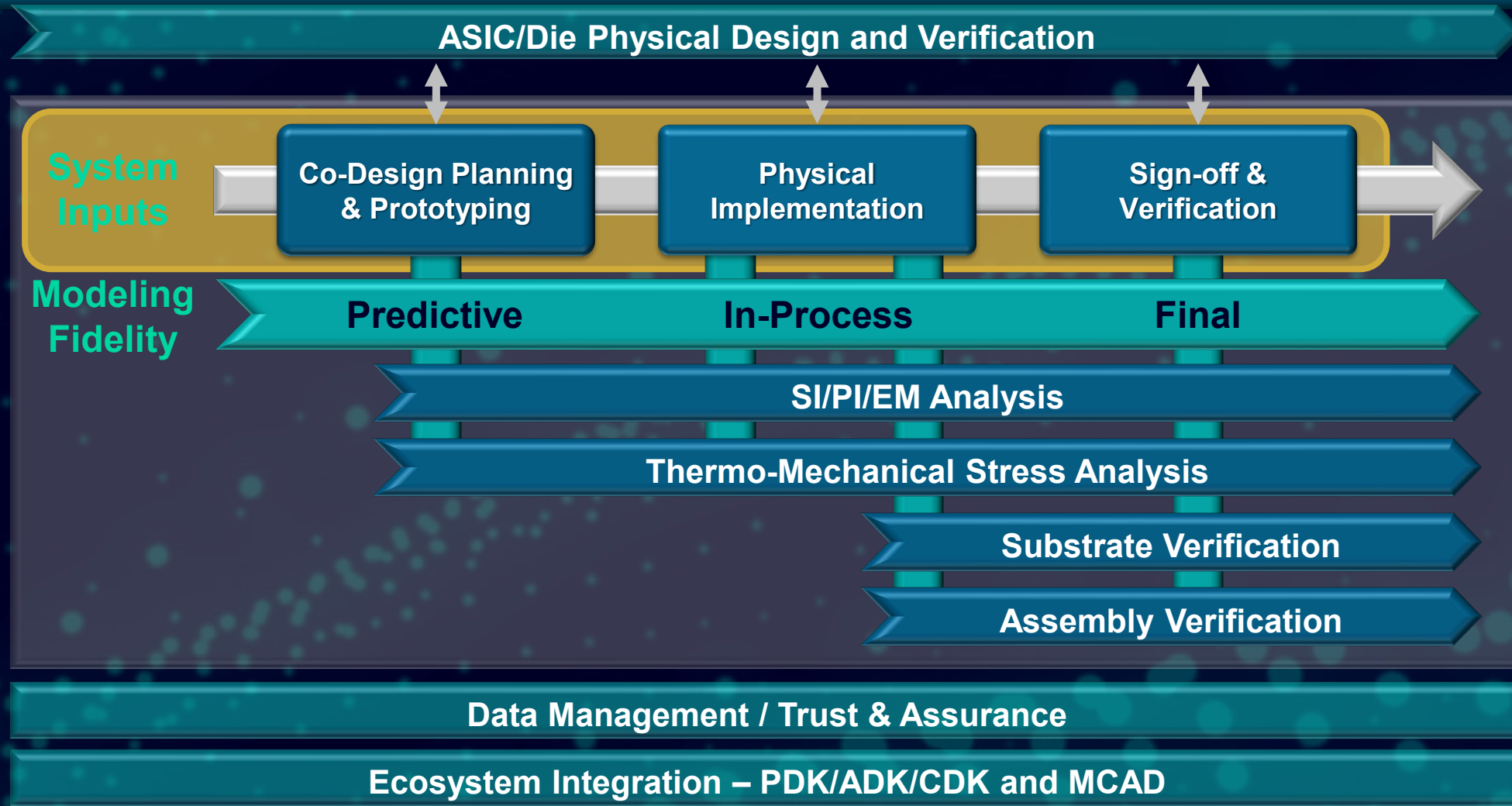
Connectivity in IC Package Design



Key Attribute

All defined connectivity promoted to the floorplan level

Chiplet Integration Workflow



Chiplet Integration Planning & Prototyping

Design Planning Cockpit

Eliminate iterations & drive better decisions

ML driven
System Technology
Co-Optimization

Predictive Analysis

Balance PPA & Cost

Eliminate problematic
spreadsheets

Generic Mem/IO
Interface

ASIC IP
Chiplet CDK
IC/Interposer PDK

Mechanical Integration

NX

Signal & Power Integrity

HyperLynx, mPower
Calibre xACT

STA

3rd Party

Thermal Analysis/Stress

Simcenter

DFT

Tessent Multi die

Package Assembly Verification

Calibre 3DSTACK

Substrate Verification

Calibre nmDRC

Physical Design

Xpedition
Package Designer
Xpedition PCB
Aprisa P&R
3rd Party P&R

Logic Data

Verilog
KYN
CSV/TXT
Schematic

Silicon Data

LEF/DEF, GDS
CSV/TXT

PCB/PKG Data

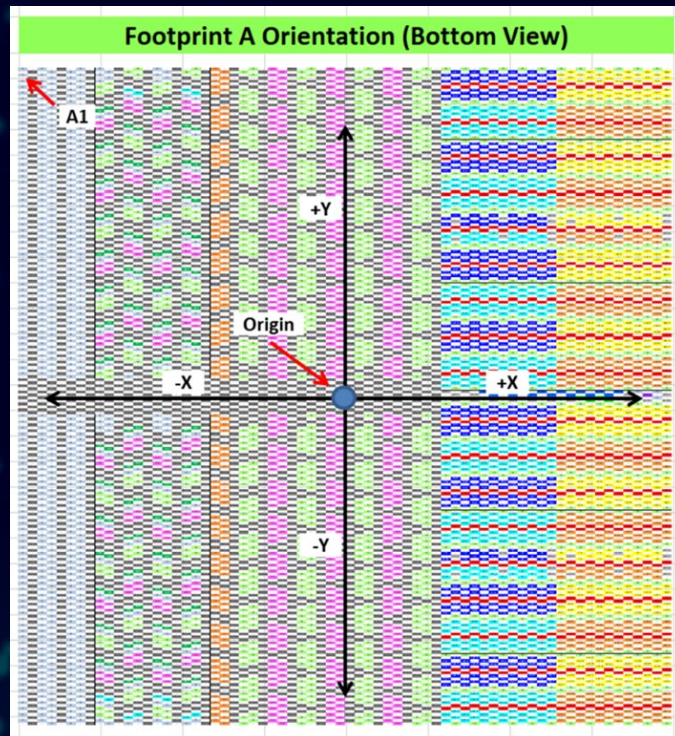
ODB++, AIF
CSV/TXT
Die/BGA.txt

ADK/PDK

Foundry
OSAT
Substrate Supplier

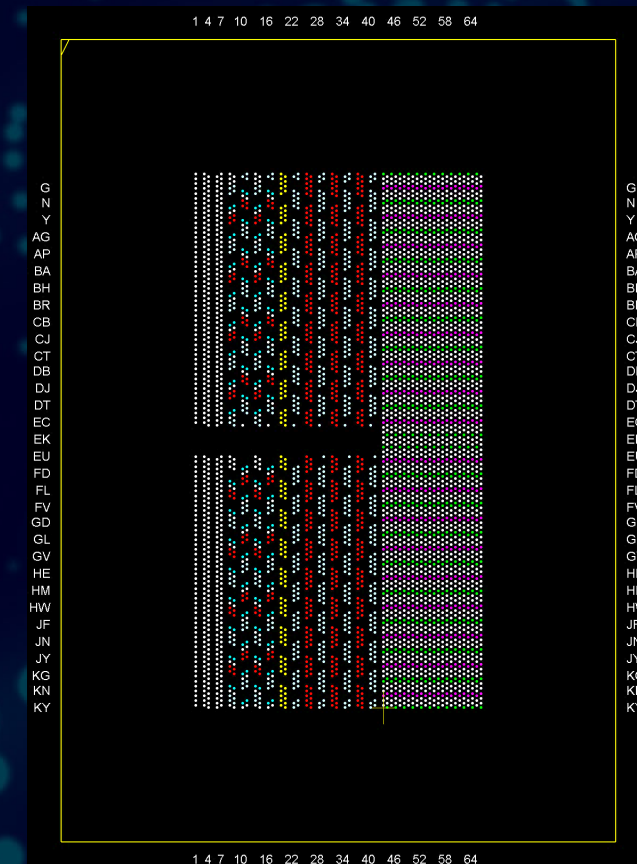
Importing Building Blocks for Floorplanning

Bump map interface specification for fixed die – typically represented in csv format



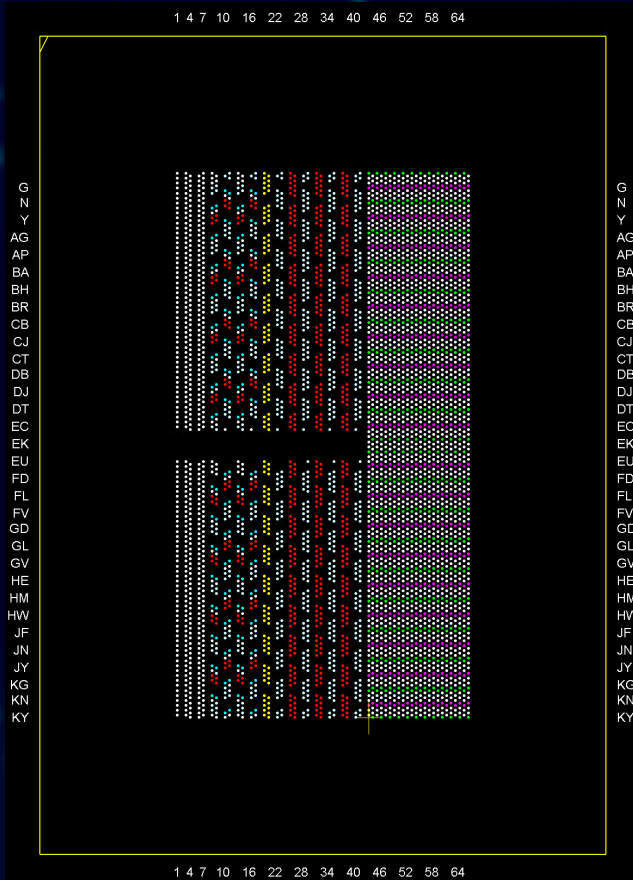
Bump map of the JESD235B HBM Ballout A highlighting repeating die to die signal channel support regions in the 4097-bit wide interface.

Interface imported into floorplanning tool as a graphical part

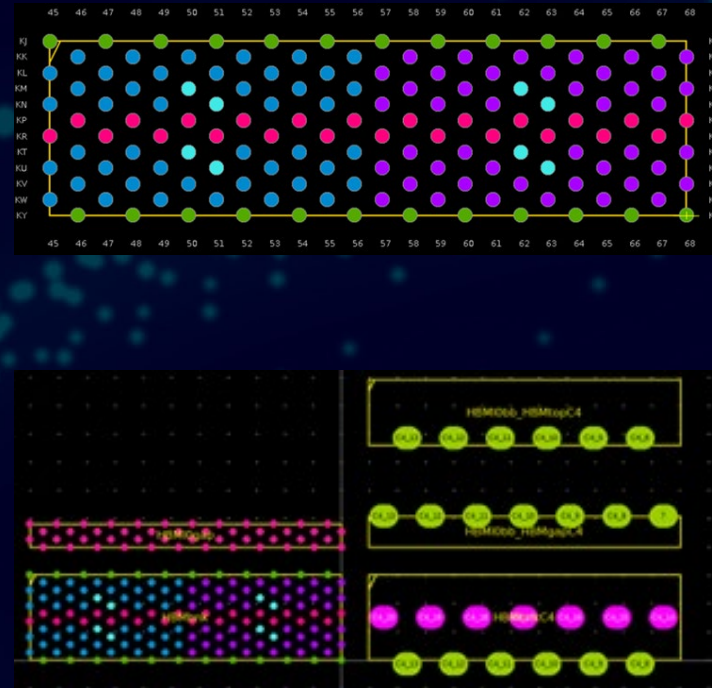


Applying Building Blocks to HBM

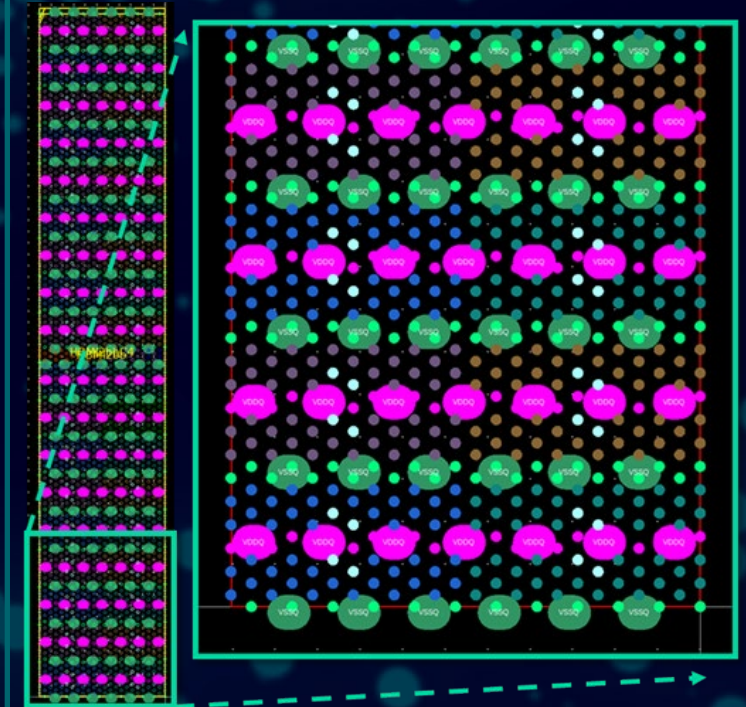
Imported part from specification



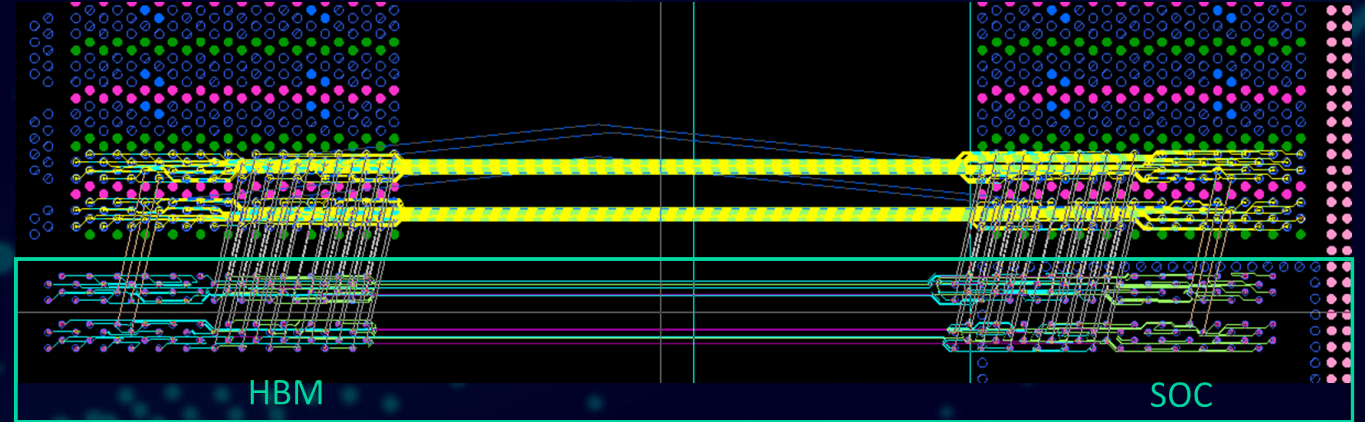
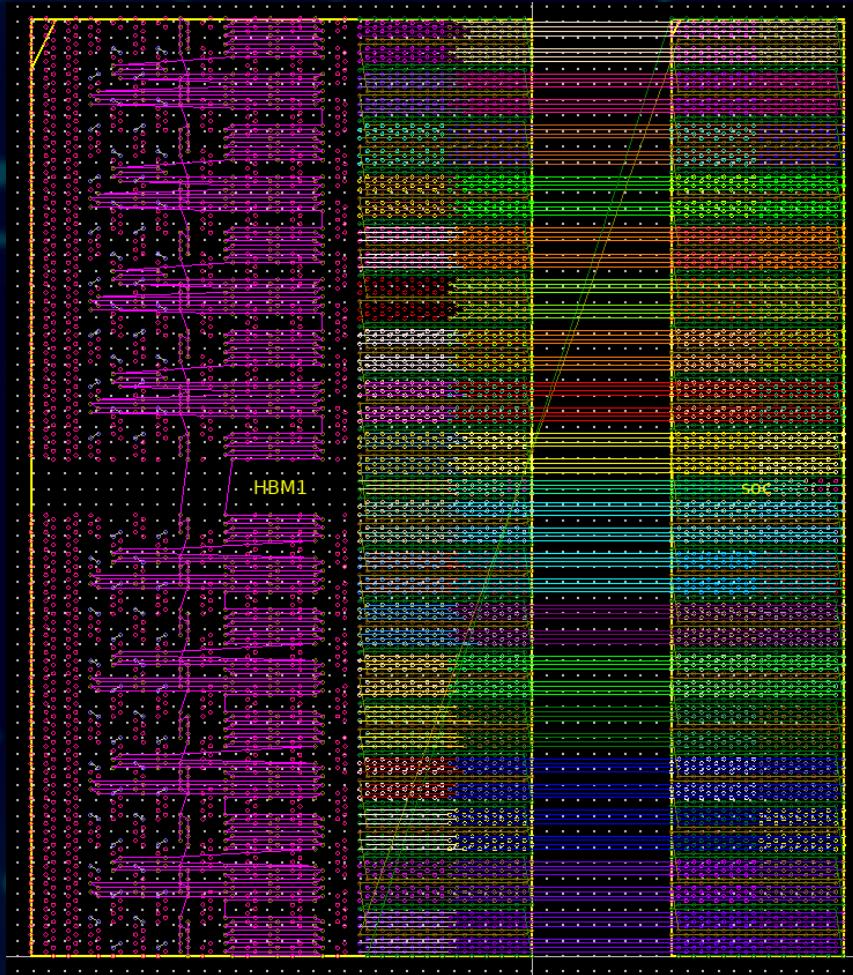
Disaggregate into building blocks with BGA bump primary blocks



Reassembled design with BGA bumps



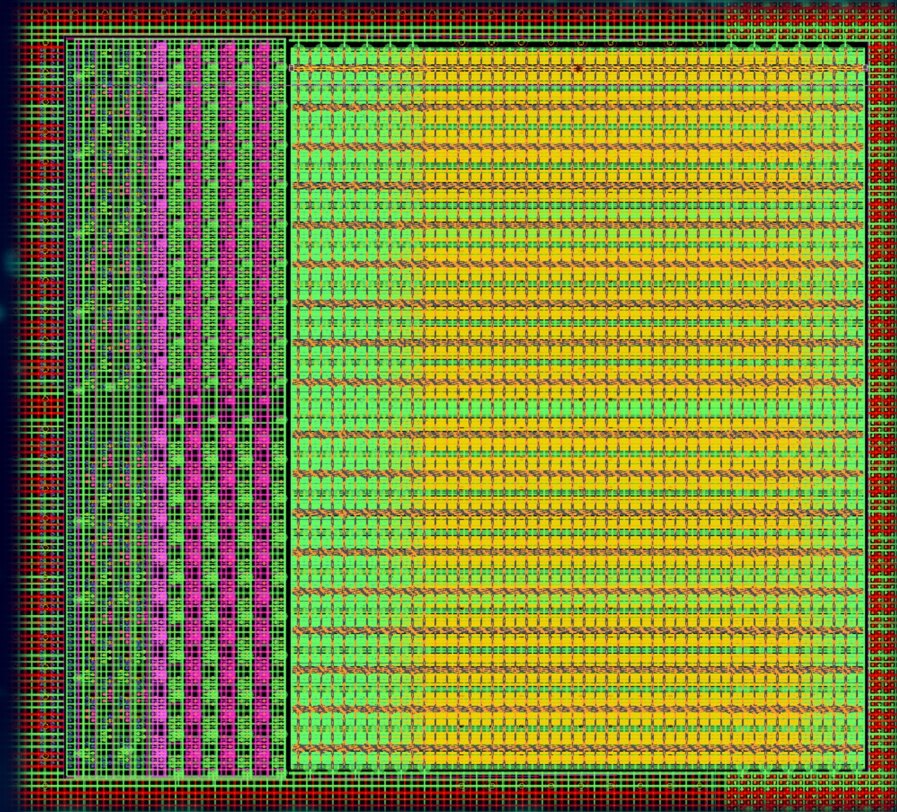
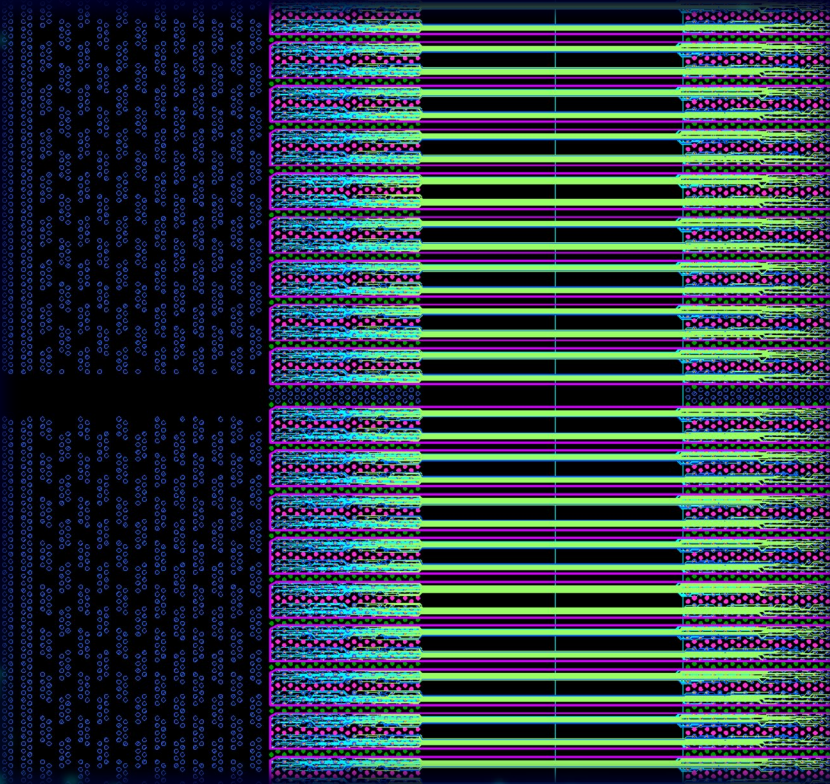
Managed Die to Die Interface



- Partition channels and signal groups for faster layout implementation and analysis
- Routing and SI/PI completed on a single bank of HBM routes and preserved into a parent block (Physical Reuse) circuit
- Hierarchical placements of parent block adapt to connectivity

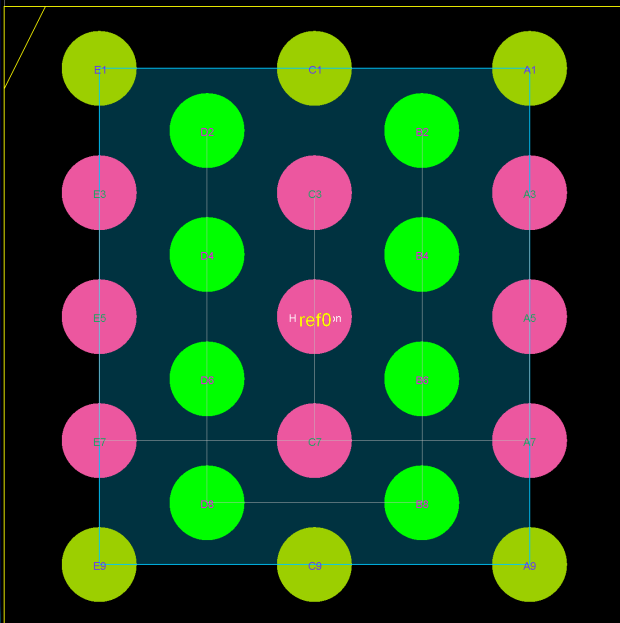
Managed Die to Die Interface

Apply repeatable, reliable physical layout strategy to managed interface

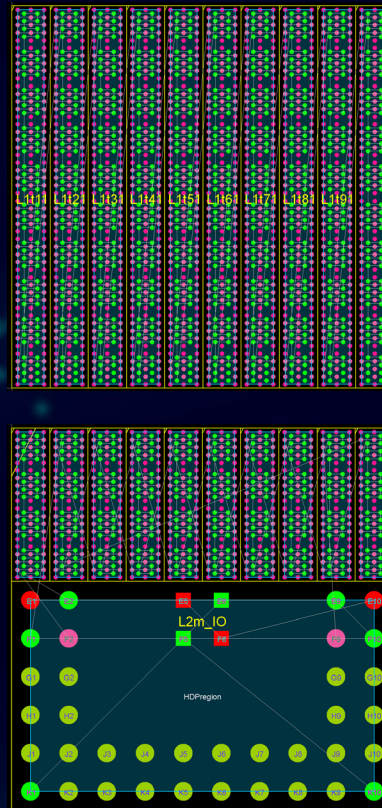


Arrayed Building Blocks for BGA

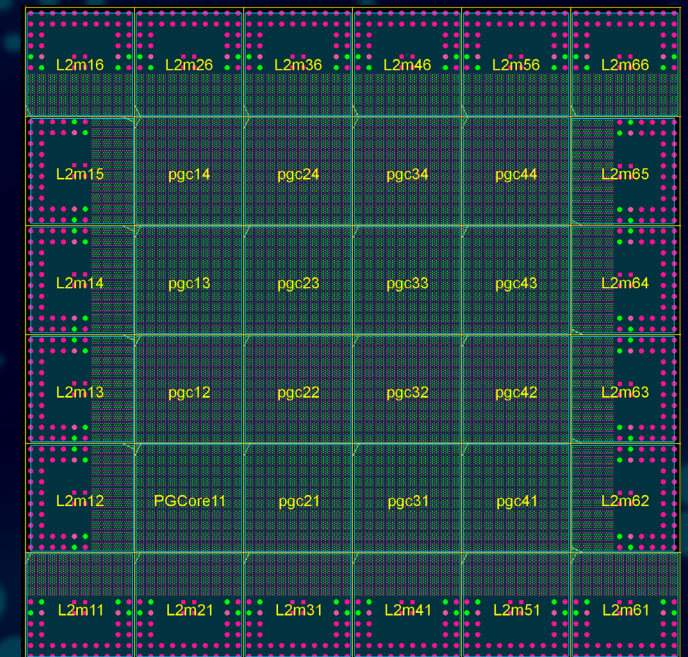
Imported part from specification



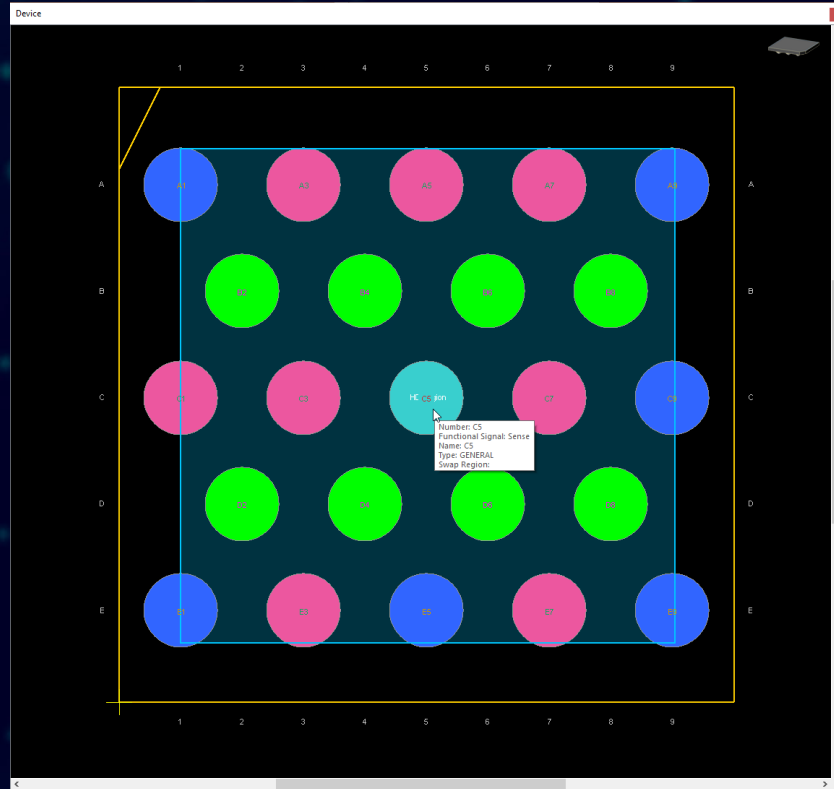
Partition into building blocks with back-side bump primary blocks



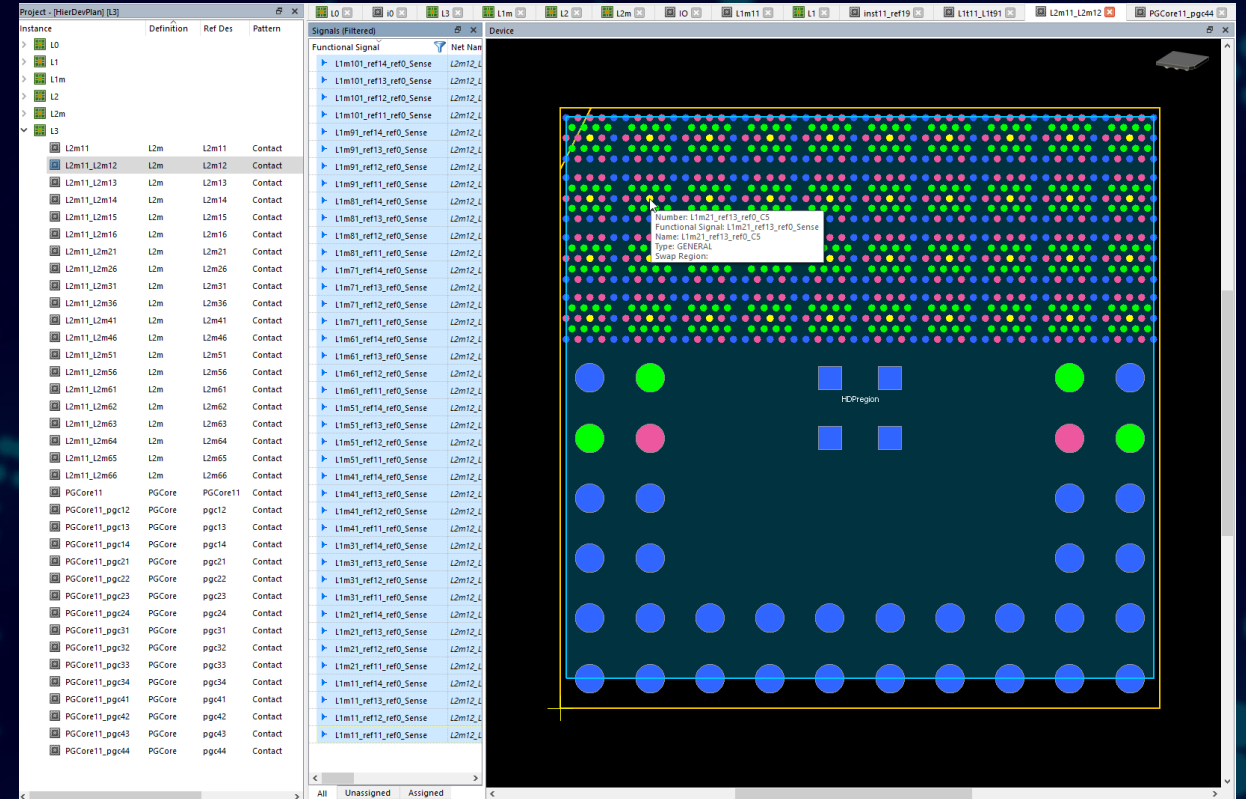
Reassembled design with back-side bumps



Fast Design Iterations



Update bumps and connectivity in the L0 (level 0) block



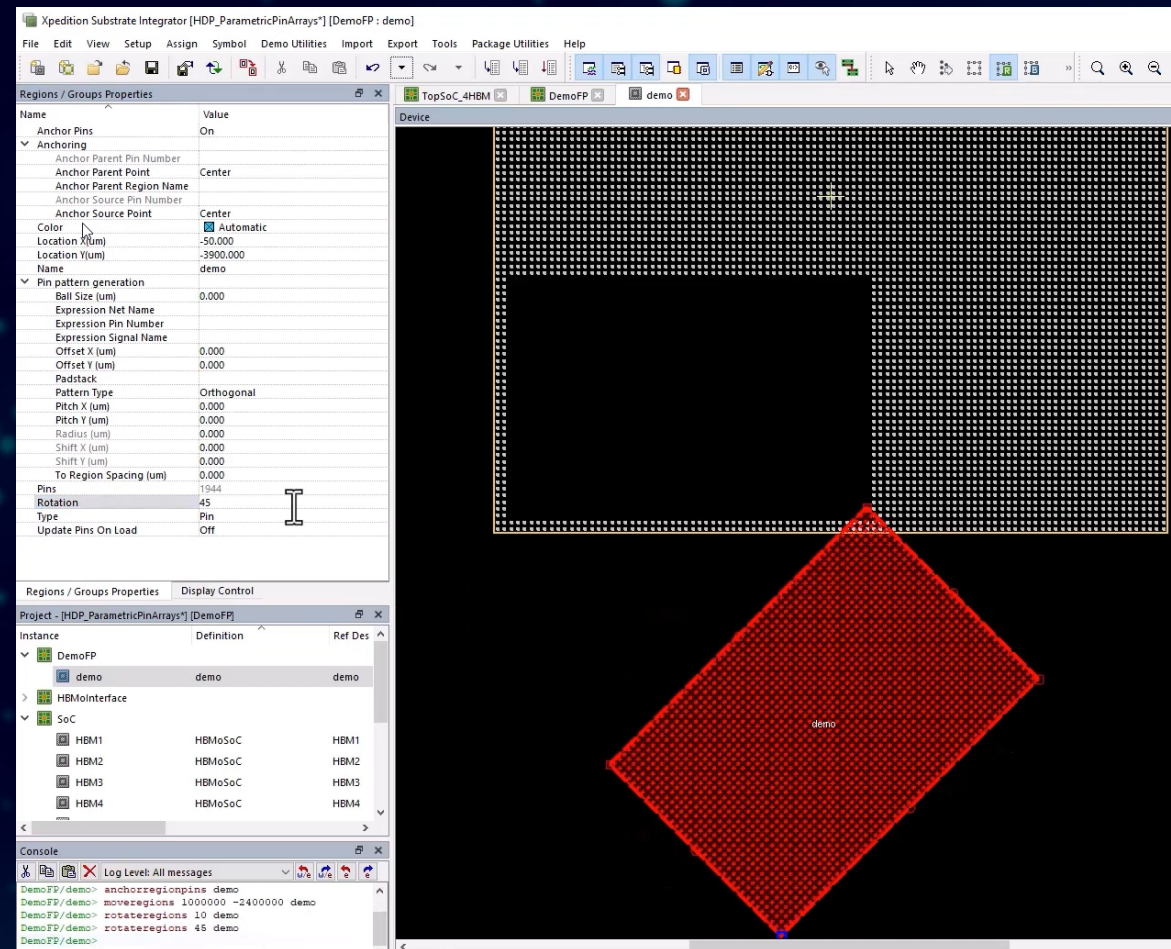
Apply automation to update hierarchy and create unique signal interface at top using pin regions

“Smart” Pin Regions

Enables rapid hierarchical prototyping of complex ASICs and chiplets by using “smart regions”

- Smart Regions are parametric regions that can auto synthesize arrays of pins
- Region anchoring provides automatic region move when a region is resized
- Complete pin regions along with pin-numbering, signal assignments and net connections are automatically generated

VERY rapid design of complex bump arrays and implement design changes in seconds vs. days to weeks



Floorplan without Bumps

Regions / Groups Properties

Name	Value
Anchor Pins	On
▼ Anchoring	
Anchor Parent Pin Number	
Anchor Parent Point	Center
Anchor Parent Region Name	
Anchor Source Pin Number	
Anchor Source Point	Center
Color	Yellow
Location X(um)	7488.888
Location Y(um)	19400.000
Name	h1hslO
▼ Pin pattern generation	
Ball Size (um)	80.000
Expression Net Name	@FunctionalSignal
Expression Pin Number	@AlphaNumericXY
Expression Signal Name	PR_/^/[Q4..BM9]\\$/ PL_/^/[CP4..EB9]\\$/
Offset X (um)	-1.000
Offset Y (um)	0.000
Padstack	
Pattern Type	Staggered
Pitch X (um)	150.000
Pitch Y (um)	150.000
Radius (um)	0.000
Shift X (um)	0.000
Shift Y (um)	0.000
To Region Spacing (um)	100.000
Pins	0
Rotation	0
Type	Pin
Update Pins On Load	Off

Device

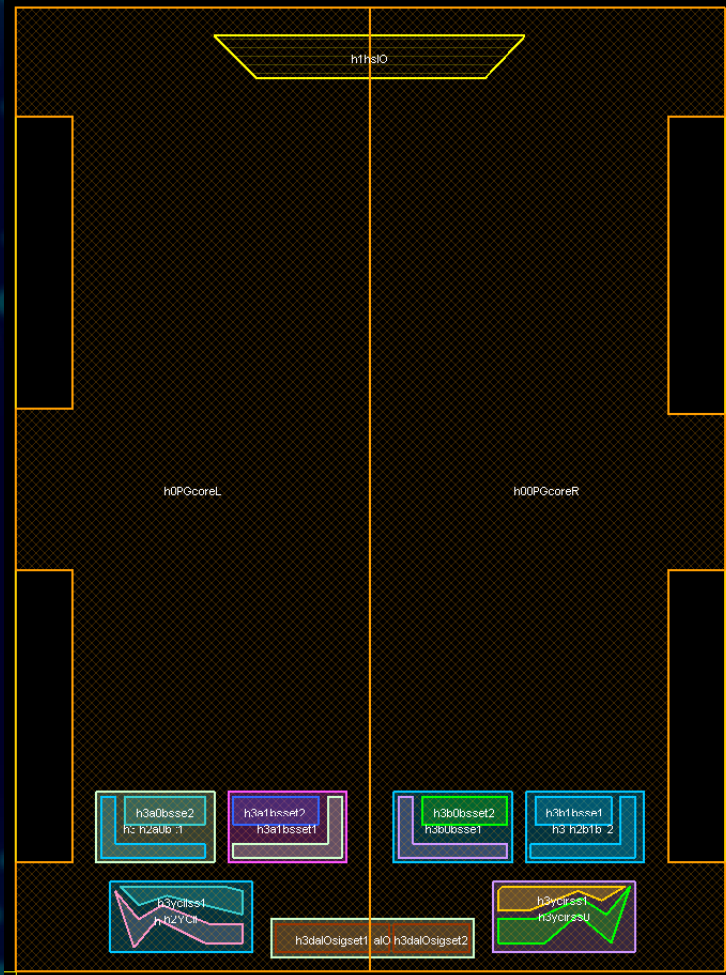
The central window shows a floorplan of a device with two main cores, h0PGcoreL and h0PGcoreR. Various peripheral blocks are placed around the cores, including h2a0b, h2a1b, h2b0b, h2b1b, h2dalO, h2YCII, h2YClr, h3a0bsse2, h3a0bsse1, h3a1bsse1, h3a1bsse2, h3b0bsse1, h3b0bsse2, h3b1bsse1, h3b1bsse2, h3dalOsigset1, h3dalOsigset2, h3ycilss0, and h3ycilss1. The floorplan is outlined in yellow, and the cores are labeled h0PGcoreL and h0PGcoreR.

Regions / Groups

Group name/pin number	Type	Including
h0PGcoreL	Pin	
h0PGcoreR	Pin	
h1hslO	Pin	
h2a0b	Pin	
h2a1b	Pin	
h2b0b	Pin	
h2b1b	Pin	
h2dalO	Pin	
h2YCII	Pin	
h2YClr	Pin	
h3a0bsse2	Pin	
h3a0bsse1	Pin	
h3a1bsse1	Pin	
h3a1bsse2	Pin	
h3b0bsse1	Pin	
h3b0bsse2	Pin	
h3b1bsse1	Pin	
h3b1bsse2	Pin	
h3dalOsigset1	Pin	
h3dalOsigset2	Pin	
h3ycilss0	Pin	
h3ycilss1	Pin	

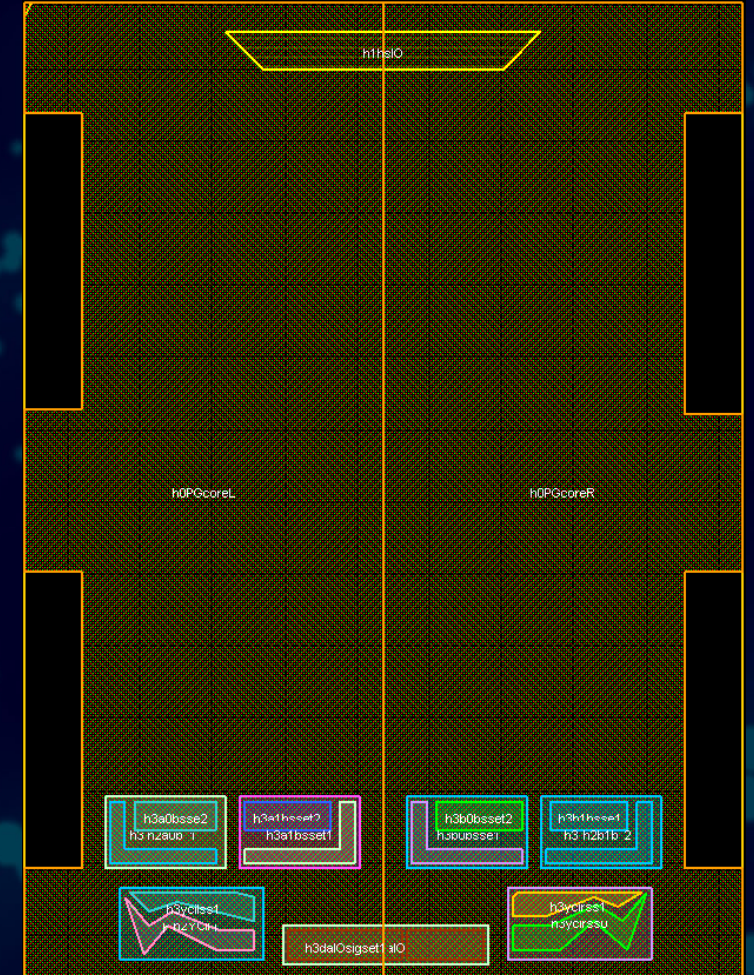
Streamline tool memory usage while floor-planning with the information that is relevant to the human brain

Staged Bump Generation – Step 1

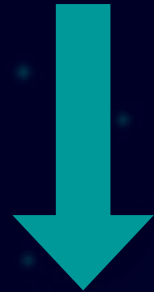
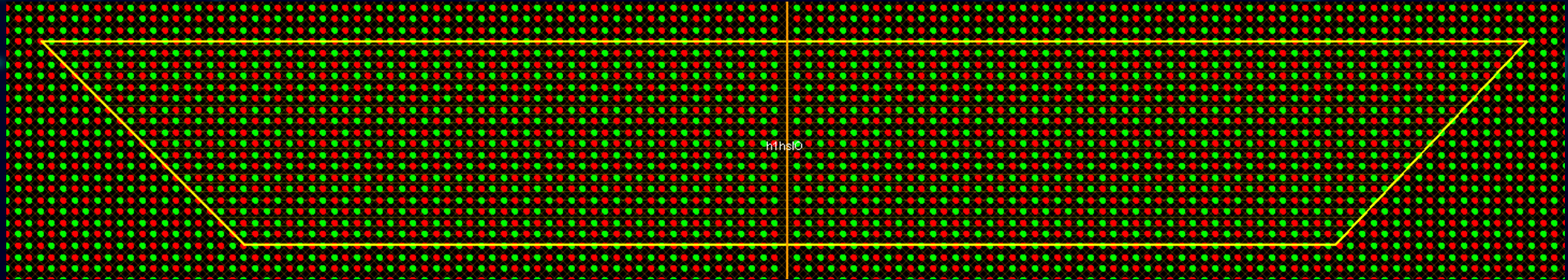


Flood full chip with power/ground bumps

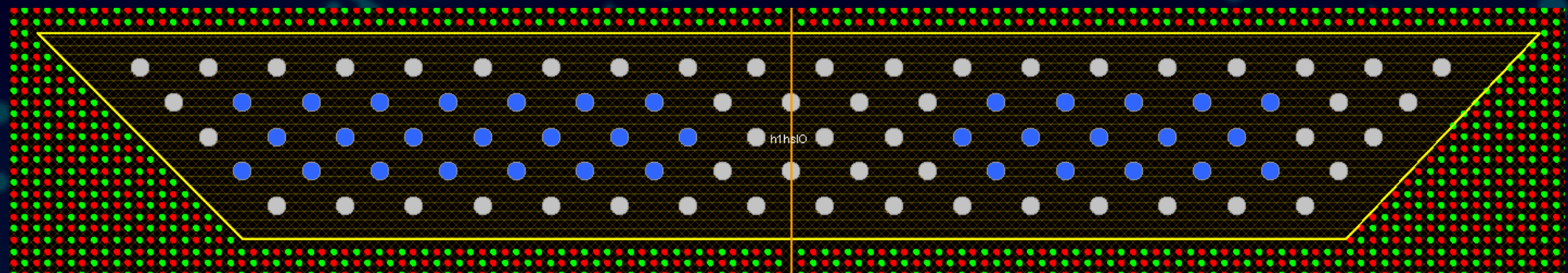
- Regenerate “h0PGcore” pin regions
- Apply expression-based pin numbering
- Auto Assign pins to checkerboard VDD/VSS



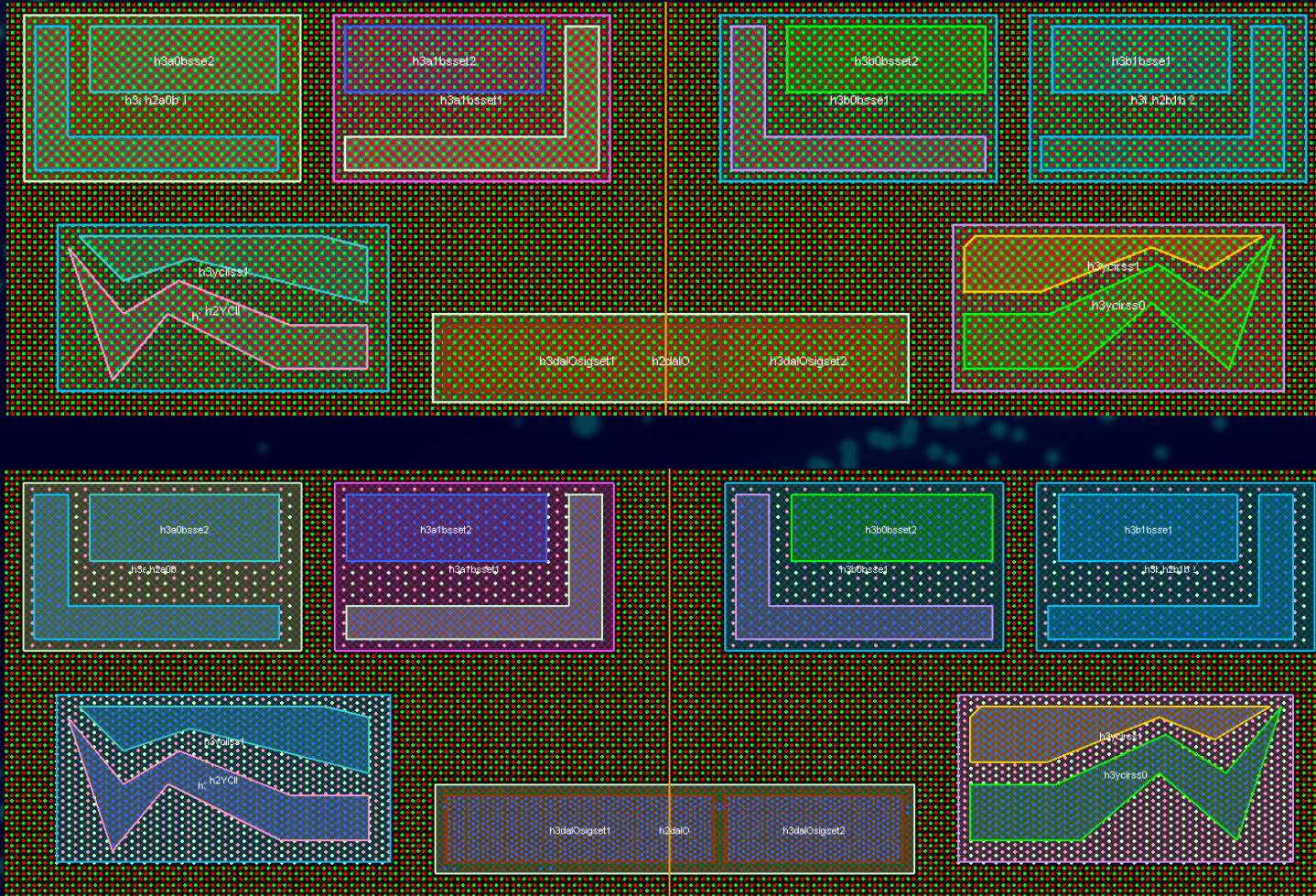
Staged Bump Generation – Step 2



- Setup high speed interface pin region with staggered pin pattern, low impedance bump, 3x pitch and adequate “To Region Spacing”
- Regenerate pins, pin numbers and signals for high-speed interface



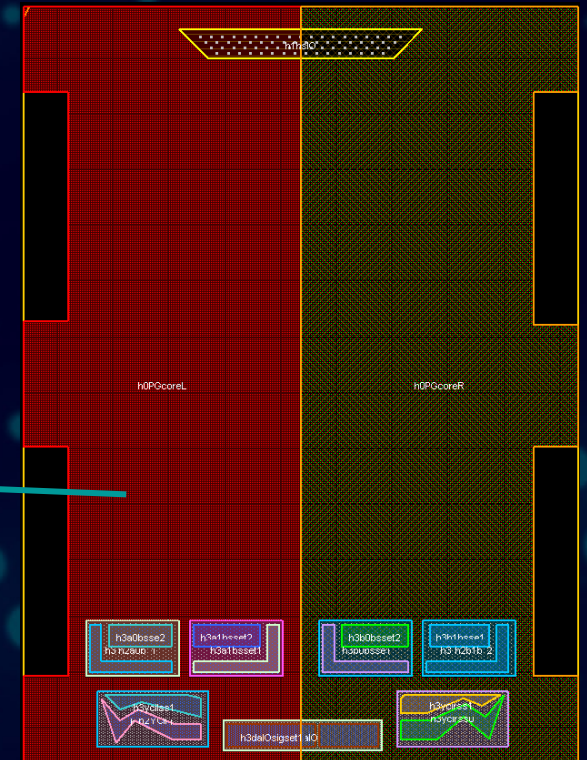
Staged Bump Generation – Step 3



- Setup high density interface pin regions with hexagonal pin pattern with various pin pattern radius
- Define sub-regions to generate signal sets
- Regenerate pins, pin numbers and signals high speed interface

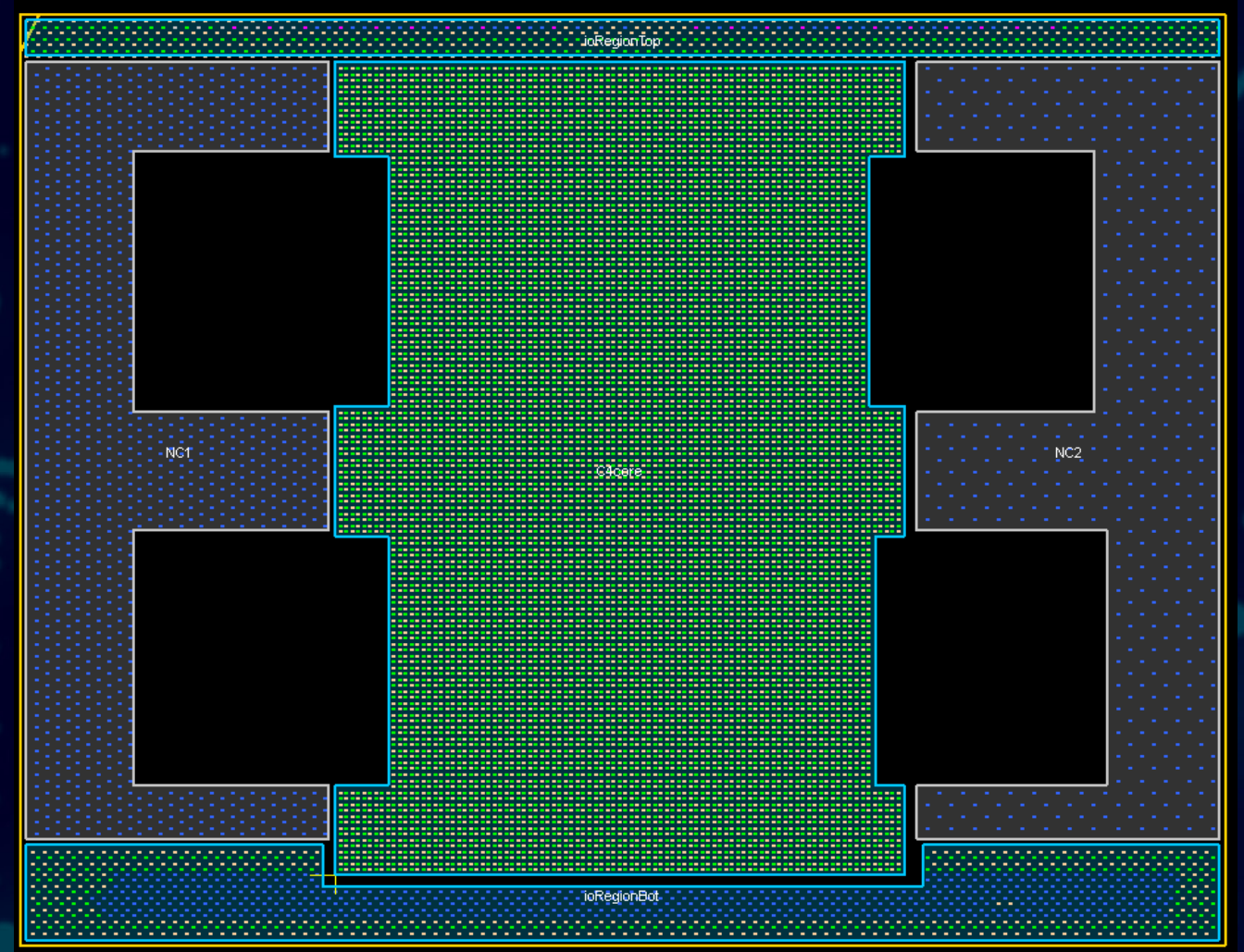
- The screenshot shows the 'Regions / Groups Properties' window in Altium Designer. The 'Anchoring' section is expanded, showing various properties. A red box highlights the 'To Region Spacing (um)' property, which is set to 50.000. A red arrow points from this property to the 'D0_BUMP_POWER_BUMP' region in the 'Regions / Groups' list on the left.

Name	Value
Anchor Pins	On
▼ Anchoring	
Anchor Parent Pin Number	
Anchor Parent Point	Center
Anchor Parent Region Name	
Anchor Source Pin Number	
Anchor Source Point	Center
Color	255, 158, 0
	3750.000
	10200.000
	h0PGcoreL
	30.000
	@AlphaNumericXY
	0.000
	0.000
	D0_BUMP_POWER_BUMP
	Orthogonal
	50.000
	50.000
	0.000
	0.000
	0.000
	50.000
	50745
	0
	Pin
	Off
	Shift Y (um)
	50.000
	To Region Spacing (um)
	50745
	0
	Pin
	Off

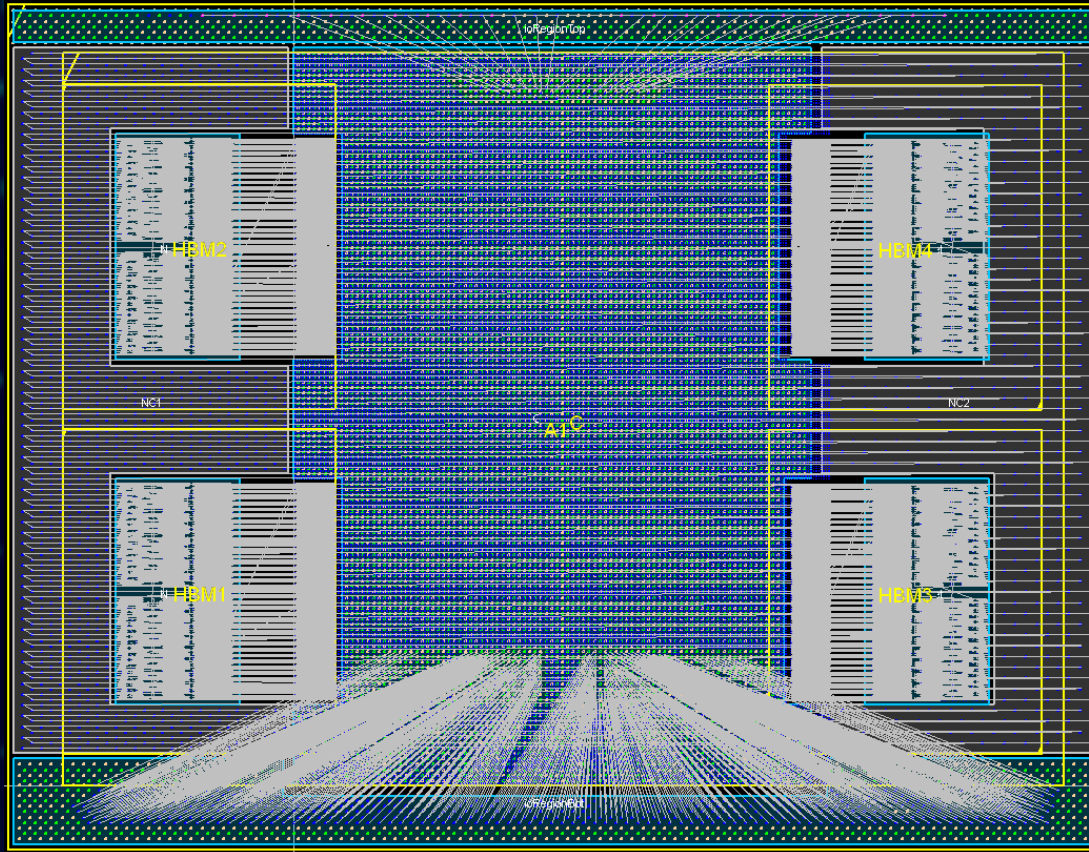


Define BGA Bumps

- Comprised entirely of parameterized pin regions
 - Geometric outline of SoC with notches for HBM building blocks
 - No Connect bumps to meet assembly rules on edges
 - Nested IO regions at bottom and top embedded with VDD and VSS



Assemble Full Design



- Placement of generated Soft IP SoC with hard HBM die to die building blocks
- Placement of generated BGA bump building block
- Placement of 4 HBM parts defined by imported CSV specification
- BGA bump signal assignments optimized with unraveling

Simply update region boundaries and regenerate pins and connectivity to accommodate design iterations

Summary

Two new methods introduced to Incorporate Hierarchy in your package design

- Hierarchical Building Blocks
- Parameterized “Smart” Pin Regions

Intuitively generate a complete silicon interposer package floorplan with imported building blocks and “Smart” pin regions

- Leverage your managed die-to-die interface blocks
- Define power/ground and signal bumps with a set of geometrical shapes
- Bypass need for the tool to render bumps during early design

“Smart” Pin regions with interactive graphical editing enable fast and intuitive design updates

Thank You

