CAL POLY
SAN LUIS OBISPO

EE 531: ADVANCED VLSI DESIGN

Floorplanning

Nishith N. Chakraborty

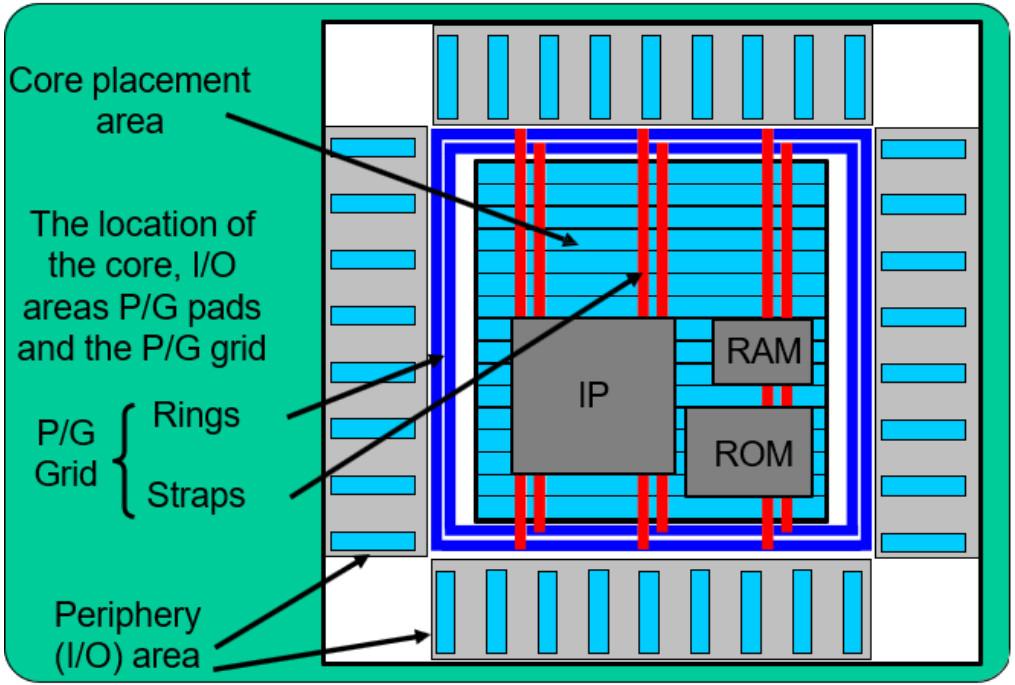
February, 2025

FLOORPLANNING GOALS AND OBJECTIVES

- Floorplanning is a mapping between the logical description (the netlist) and the physical description (the floorplan).
- **Goals of floorplanning:**
  - Arrange the blocks on a chip.
  - Decide the location of the I/O pads.
  - Decide the location and number of the power pads.
  - Decide the type of power distribution.
  - Decide the location and type of clock distribution.
- **Objectives of floorplanning:**
  - Minimize the chip area
  - Minimize delay
  - Minimize routing congestion

FULL CHIP DESIGN OVERVIEW

- Chip size
- Number of Gates
- Number of Metal layers
- Interface to the outside
- Hard IPs/Macros
- Power Delivery
- Multiple Voltages
- Clocking Scheme
- Flat or Hierarchical?

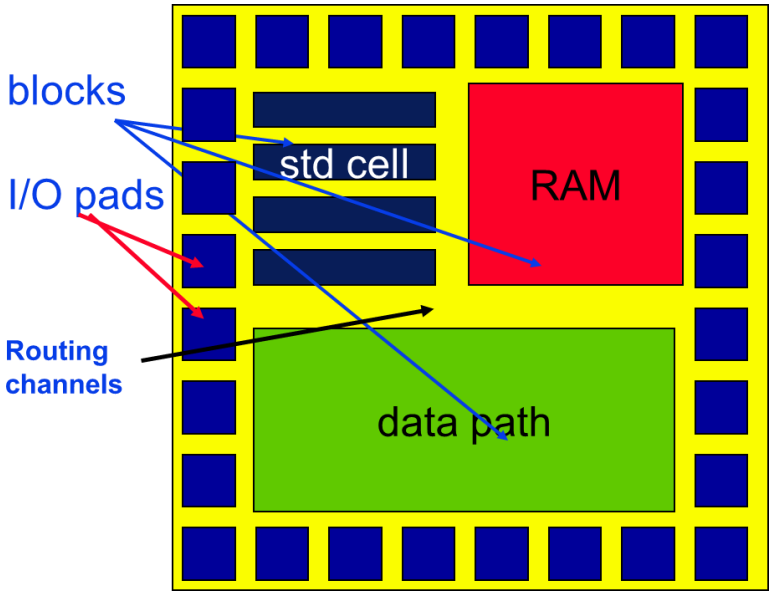


FLOORPLANNING INPUTS AND OUTPUTS

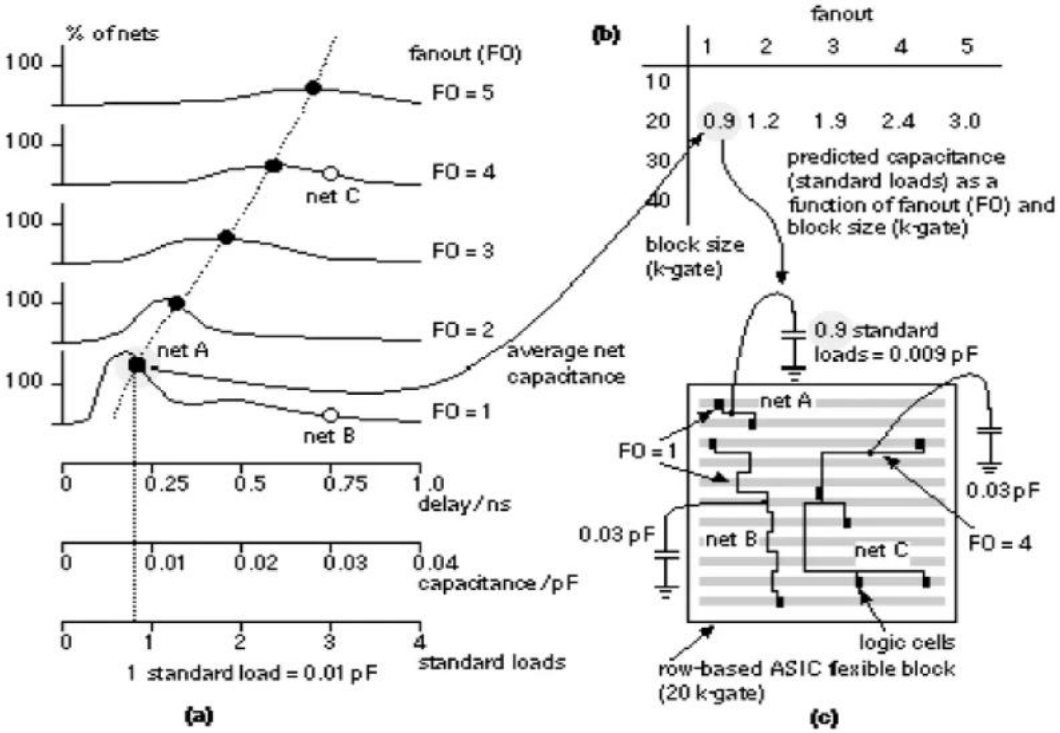
- **Inputs**
  - Design netlist (required)
  - Area requirements (required)
  - Power requirements (required)
  - Timing constraints (required)
  - Physical partitioning information (required)
  - Die size vs. performance vs. schedule trade-off (required)
  - I/O placement (optional)
  - Macro placement information (optional)

FLOORPLANNING INPUTS AND OUTPUTS

- **Outputs**
  - Die/block area
  - I/Os placed
  - Macros placed
  - Power grid designed
  - Power pre-routing
  - Standard cell placement areas
- Design ready for standard cell placement



FLOORPLANNING: DELAY MEASUREMENT

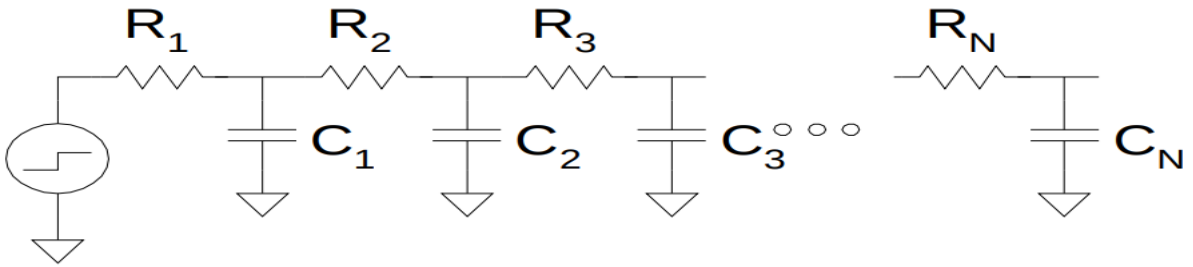


ELMORE DELAY

- Elmore delay of RC ladder

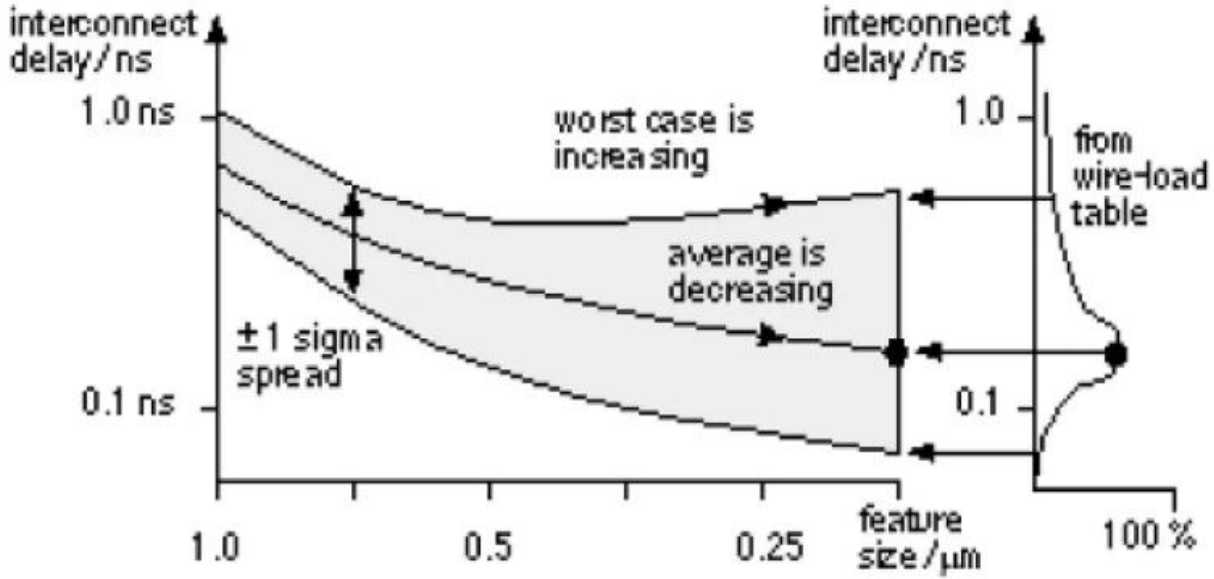
$$t_{pd} \approx \sum_{\text{nodes } i} R_{i\text{- to- source}} C_i$$

$$= R_1 C_1 + (R_1 + R_2) C_2 + \ldots + (R_1 + R_2 + \ldots + R_N) C_N$$

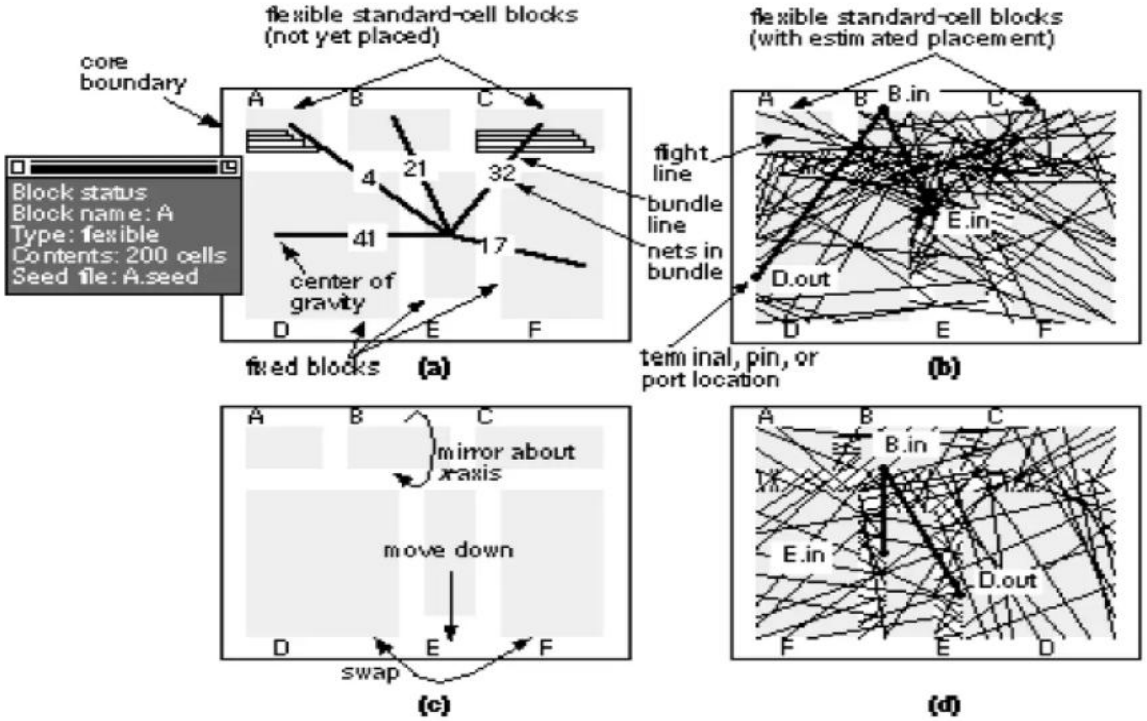


FLOORPLANNING: WORST CASE DELAY

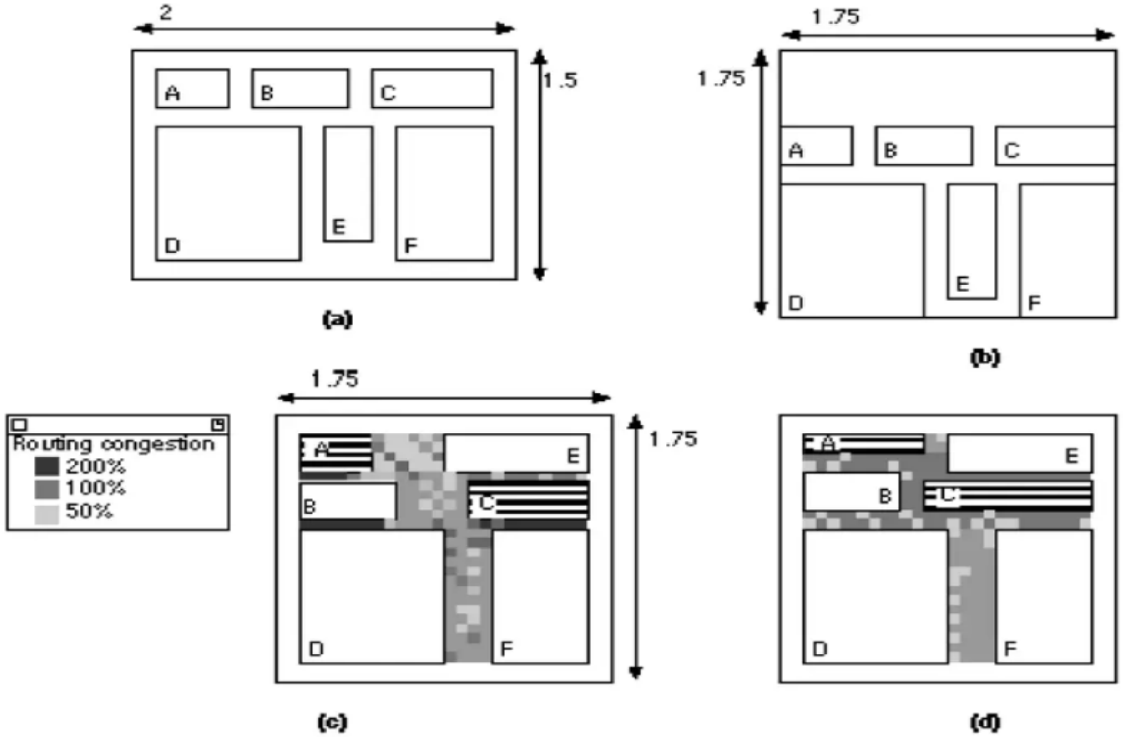
Worst-case interconnect delay. As we scale circuits, but avoid scaling the chip size, the worst-case interconnect delay increases.



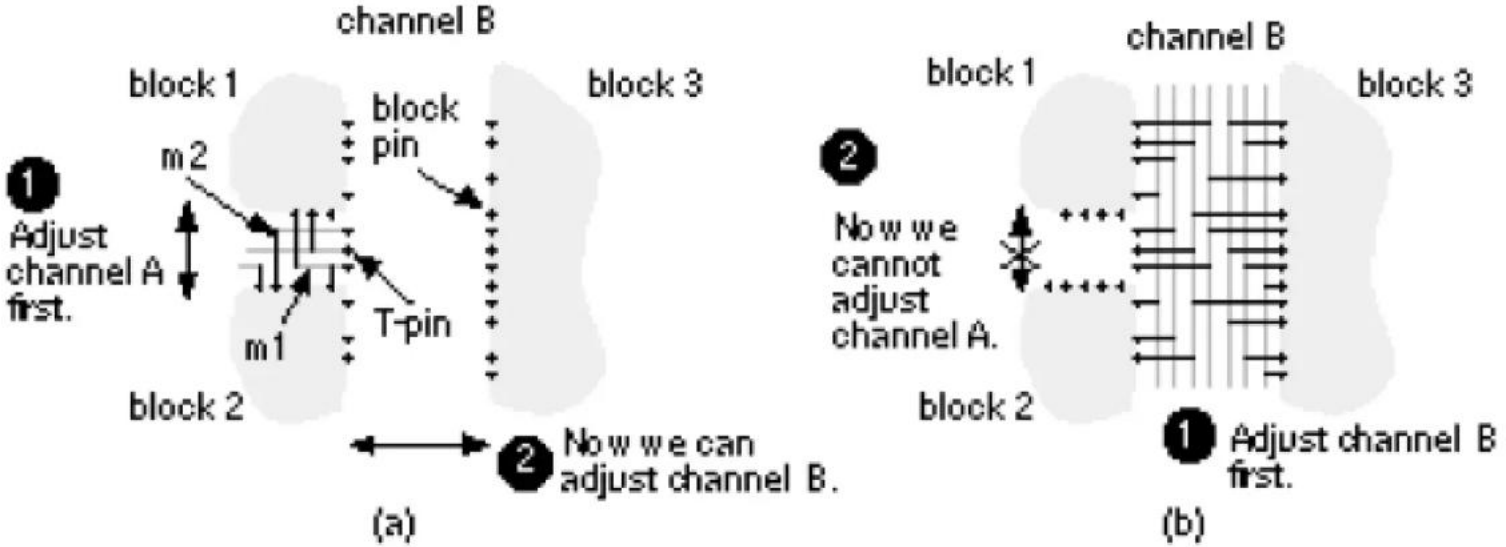
FLOORPLANNING TOOLS: OPTIMIZATION



FLOORPLANNING TOOLS: CONGESTION ANALYSIS



FLOORPLANNING TOOLS: CHANNEL DEFINITION

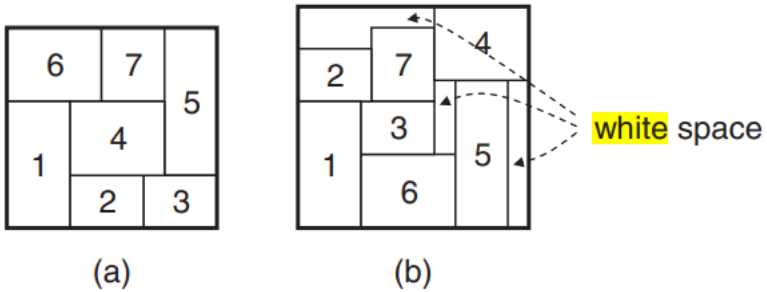


IMPORTANCE

- These decisions impact not only the functionality of the circuit, but also its performance.
- Mismanagement of channel allocation can lead to the following issues:
  - **Increased wiring congestion**: If one channel gets blocked or overused, it can force wires into longer paths, which increase the delay can may cause timing violations.
  - **Suboptimal area usage**: Poor routing can increase the area required for interconnects, leading to a larger chip size.
  - **Signal integrity issues**: Routing conflicts may also result in overlapping and congested wires which can lead to signal interference and other electrical problems.

FLOORPLANNING: WHITE SPACE MINIMIZATION

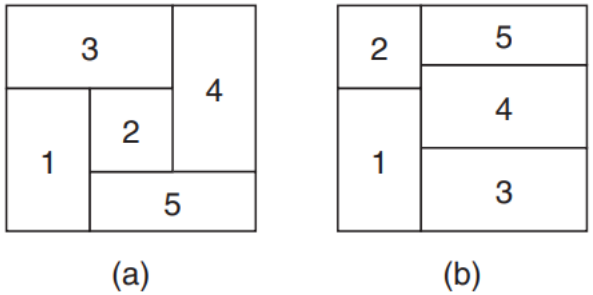
- The space in the floorplan bounding rectangle uncovered by any module is called white space or dead space



(a) an optimal floorplan in terms of area. (b) a non-optimal floorplan.

FLOORPLANNING MODELS

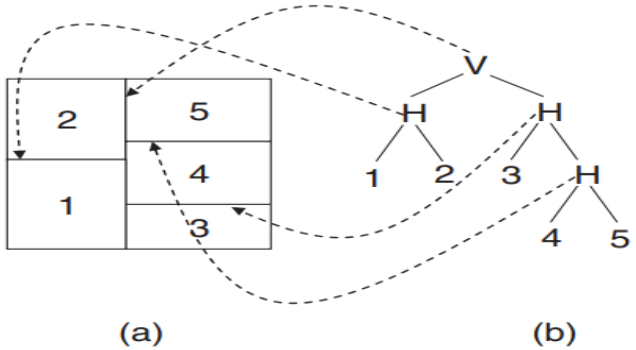
- We can classify floorplans into two categories for discussions:
  - Slicing floorplans: A slicing floorplan can be obtained by repetitively cutting the floorplan horizontally or vertically
  - Non-slicing floorplans: A non-slicing floorplan cannot be obtained by repetitively cutting the floorplan horizontally or vertically



Examples of (a) a non-slicing floorplan. (b) a slicing floorplan.

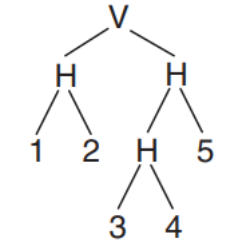
SLICING FLOORPLANS

- On the basis of the slicing property of a slicing floorplan, we can use a binary tree to represent a slicing floorplan [Otten 1982].
- A slicing tree is a binary tree with modules at the leaves and cut types at the internal nodes.
- There are two cut types, H and V.
- The H cut divides the floorplan horizontally, and the left (right) child represents the bottom (top) sub-floorplan.
- Similarly, the V cut divides the floorplan vertically, and the left (right) child represents the left (right) sub-floorplan.

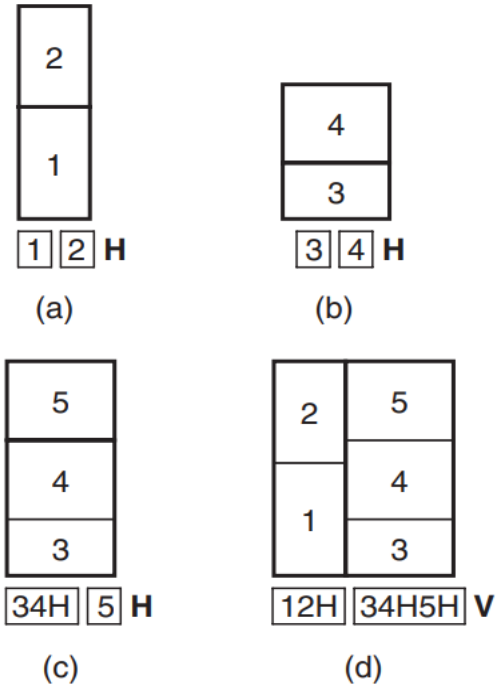


An example of (a) a slicing floorplan. (b) a slicing tree modeling.

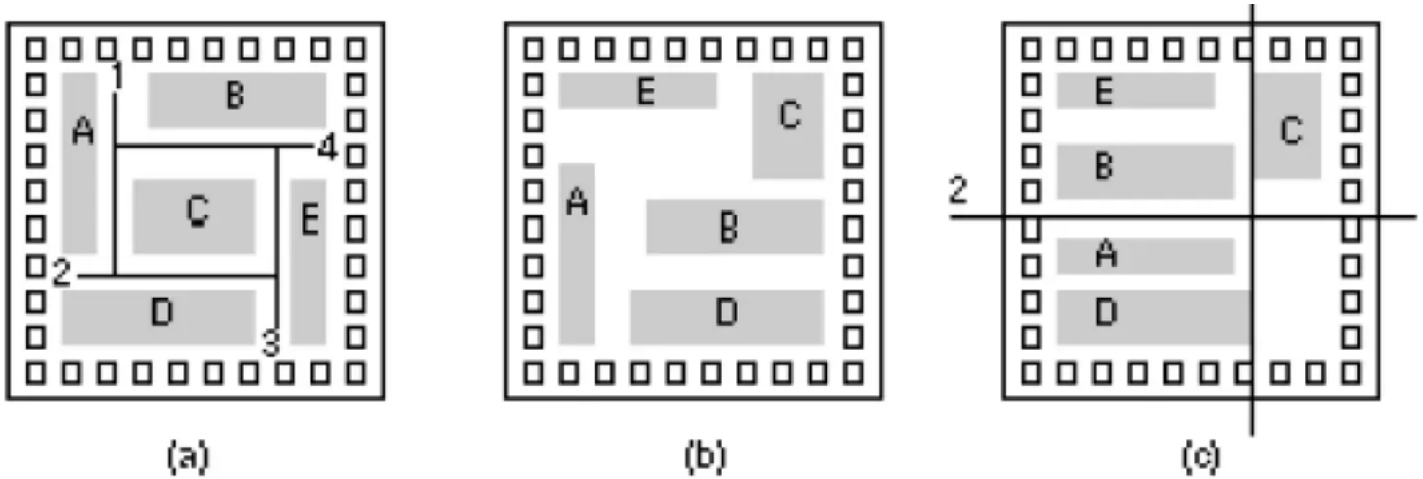
BINARY TREES AND POLISH EXPRESSIONS



E = 12H34H5HV

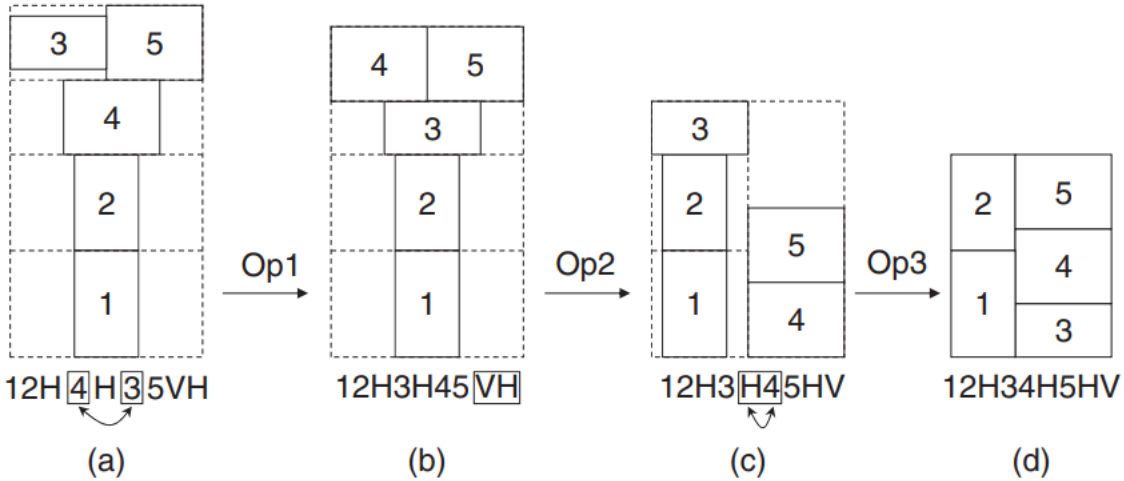


NON-SLICING FLOORPLANS



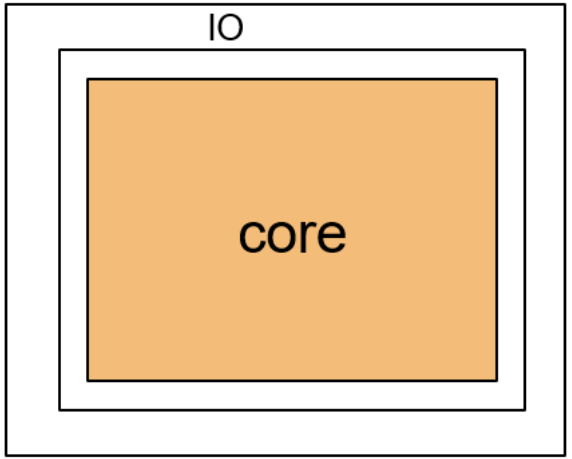
- Non-slicing blocks often lead to routing congestion
- Solution:
  - Move the blocks until we have a slicing floorplan
  - Use of L-shaped channels

NON-SLICING FLOORPLANS OPTIMIZATION

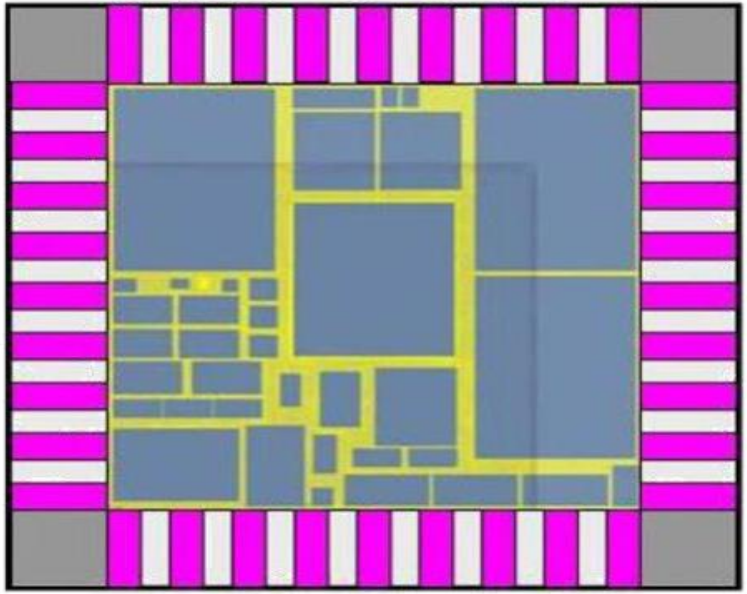


IO RING

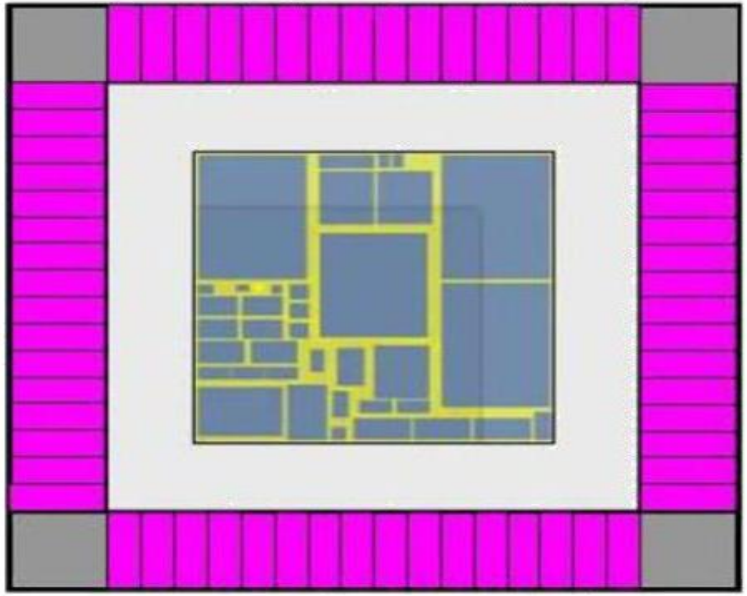
- **The pinout is often decided by front-end designers, with input from physical design and packaging engineers.**
  - I/Os do not tend to scale with Moore's Law and therefore, they are very expensive (in terms of area).
  - I/Os are not only needed for connecting signals to the outside world, but also to provide power to the chip.
  - Therefore, I/O planning is a critical and very central stage in Floorplanning the chip.



HOW DO WE CHOOSE OUR CHIP SIZE?



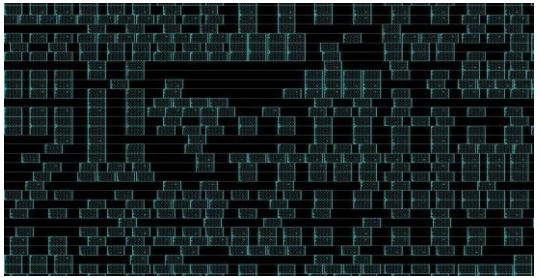
"Core Limited"



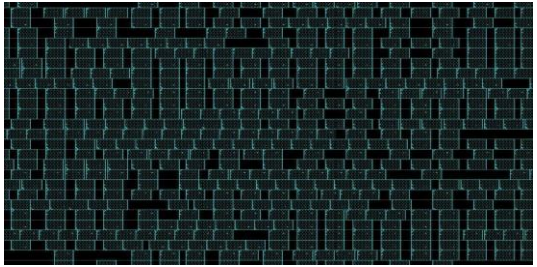
"Pad Limited"

UTILIZATION

- Utilization refers to the percentage of core area that is taken up by standard cells.
  - A typical starting utilization might be 70%
  - This can vary a lot depending on the design
- High utilization can make it difficult to close a design
  - Routing congestion
  - Negative impact during optimization legalization stages.
- Local congestion
  - Can occur with pin-dense cells like multiplexers, so utilization is not completely sufficient for determining die size.
  - Run a quick trial route to check for routing congestion
  - Refine the synthesis or increase routing resources



Low standard-cell utilization

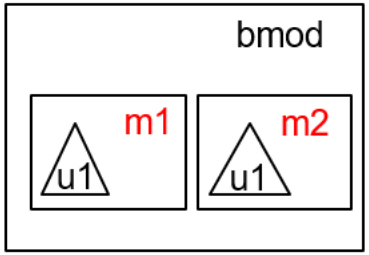


High standard-cell utilization

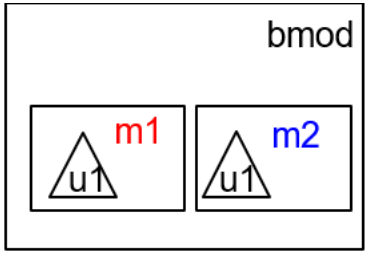
UNIQUIFYING THE NETLIST

- When moving to the physical domain, the netlist must be unique
  - A unique netlist, means that each sub-module is only referenced once.
  - In the example, the non-unique netlist cannot optimize instance m1/u1 without changing instance m2/u1

```
module amod ();
 BUFFD1 u1 ();
endmodule

module bmod ();
 amod m1 ;
amod  m2  ;
endmodule
```



Non-unique



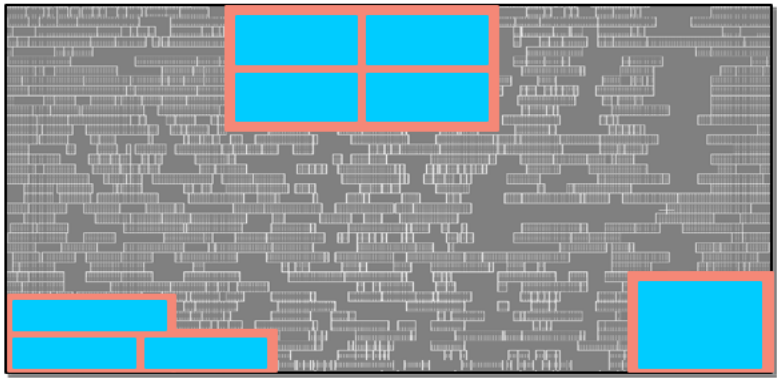
Unique

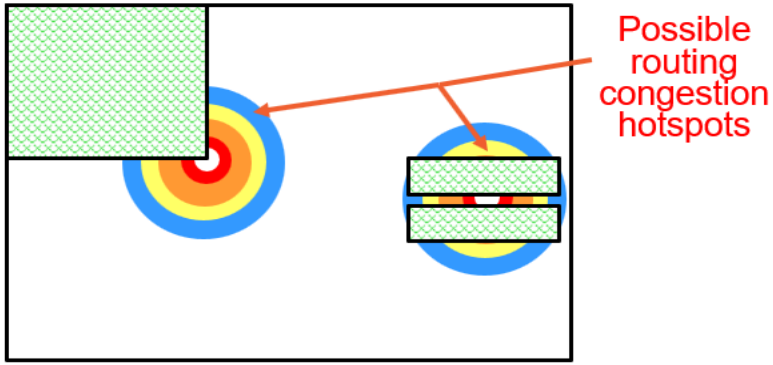
```
module amod1 ();
BUFFD1 u1 ();
endmodule

module amod2 ();
BUFFD1 u1 ();
endmodule

module bmod ();
amod1 m1 ;
amod2 m2 ;
endmodule
```

  - A synthesized netlist must be *uniquified* before placement can begin. This can be done either by the synthesizer or during design import.

HARD MACRO PLACEMENT

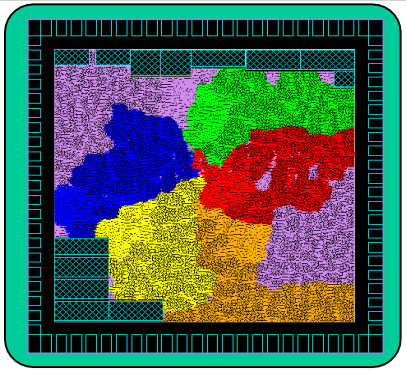
- When placing large macros we must consider impacts on routing, timing and power.
- Usually push them to the sides of the floorplan.
  - Placement algorithms generally perform better with a single large rectangular placement area.
  - For wire-bond place power hungry macros away from the chip center.
- After placing hard macros, mark them as FIXED.

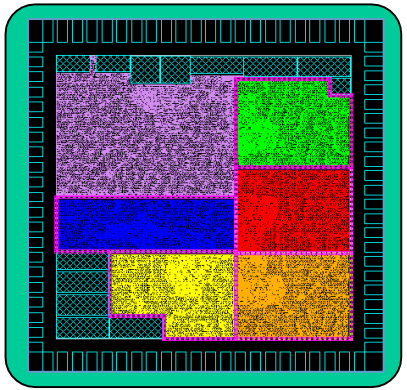




PLACEMENT REGIONS

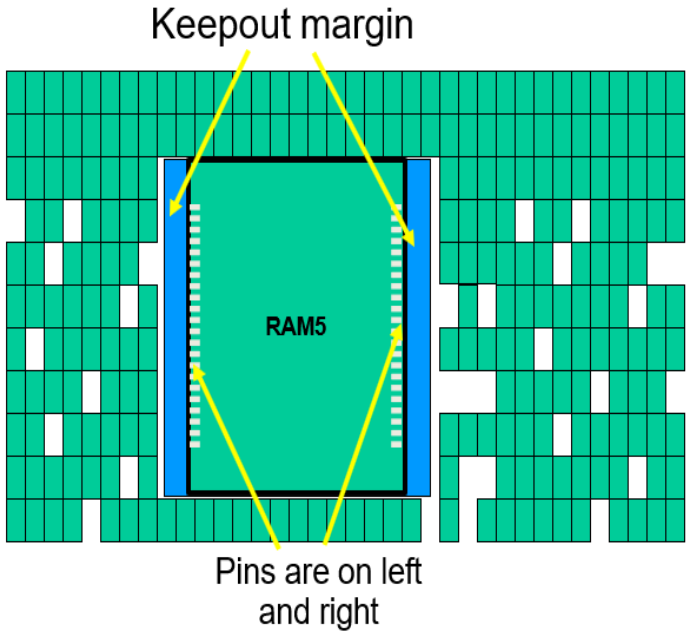
- Sometimes, we want to “help” the tool put certain logic in certain regions or cluster them together.
- Place and Route tools define several types of placement regions:
  - Soft guide – try to cluster these cells together without a defined area.
  - Guide – try to place the cells in the defined area.
  - Region – must place the cells in the defined area, but other cells may also be placed there.
  - Fence – must place the cells in the defined area and keep out all other cells.





PLACEMENT BLOCKAGES AND HALOS

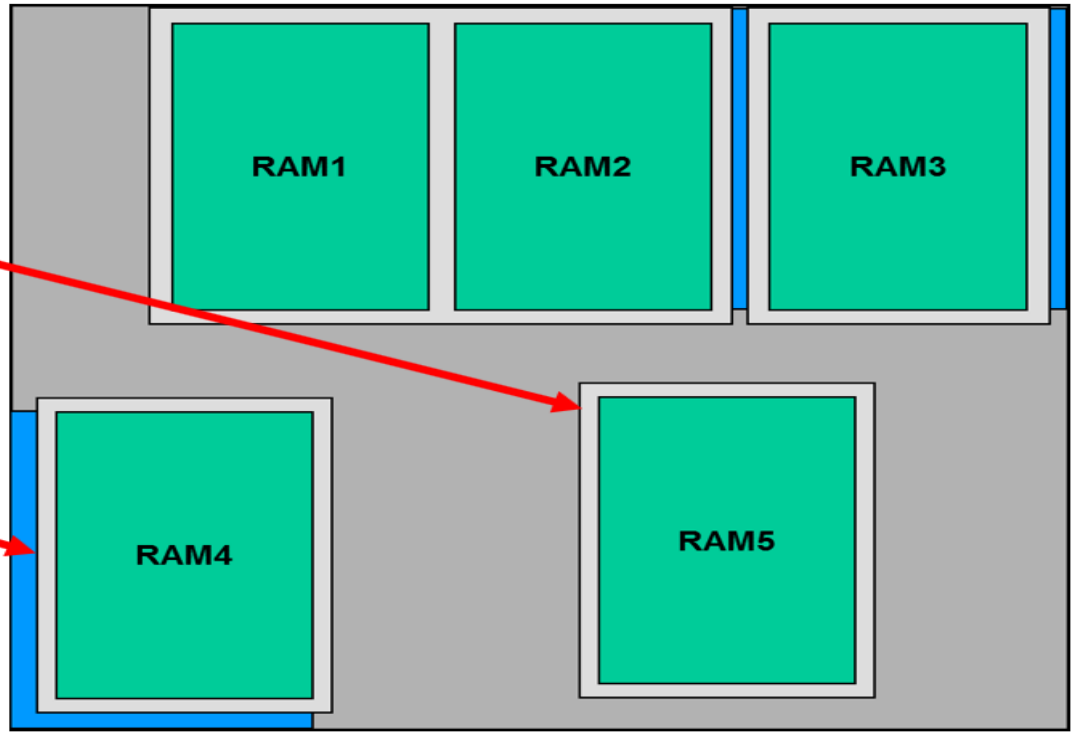
- Placement blockage halos are areas that the tools should not place any cells.
- These, too, have several types:
  - Hard Blockage – no cells can be placed inside.
  - Soft Blockage – cannot be used during placement, but may be used during optimization.
  - Partial Blockage – an area with lower utilization.
  - Halo (padding) – an area outside a macro that should be kept clear of standard cells.



PLACEMENT BLOCKAGES AND HALOS

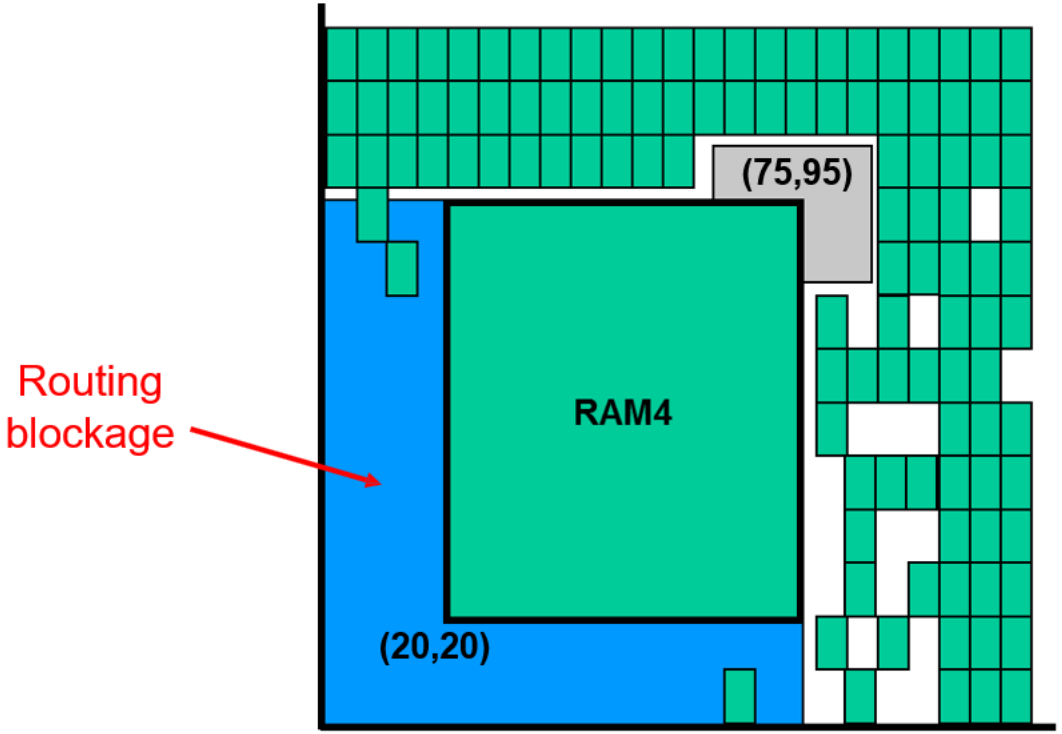
Hard blockage always created on all four sides

Soft blockage created only for the channels between the macros or between the macro and the core boundary

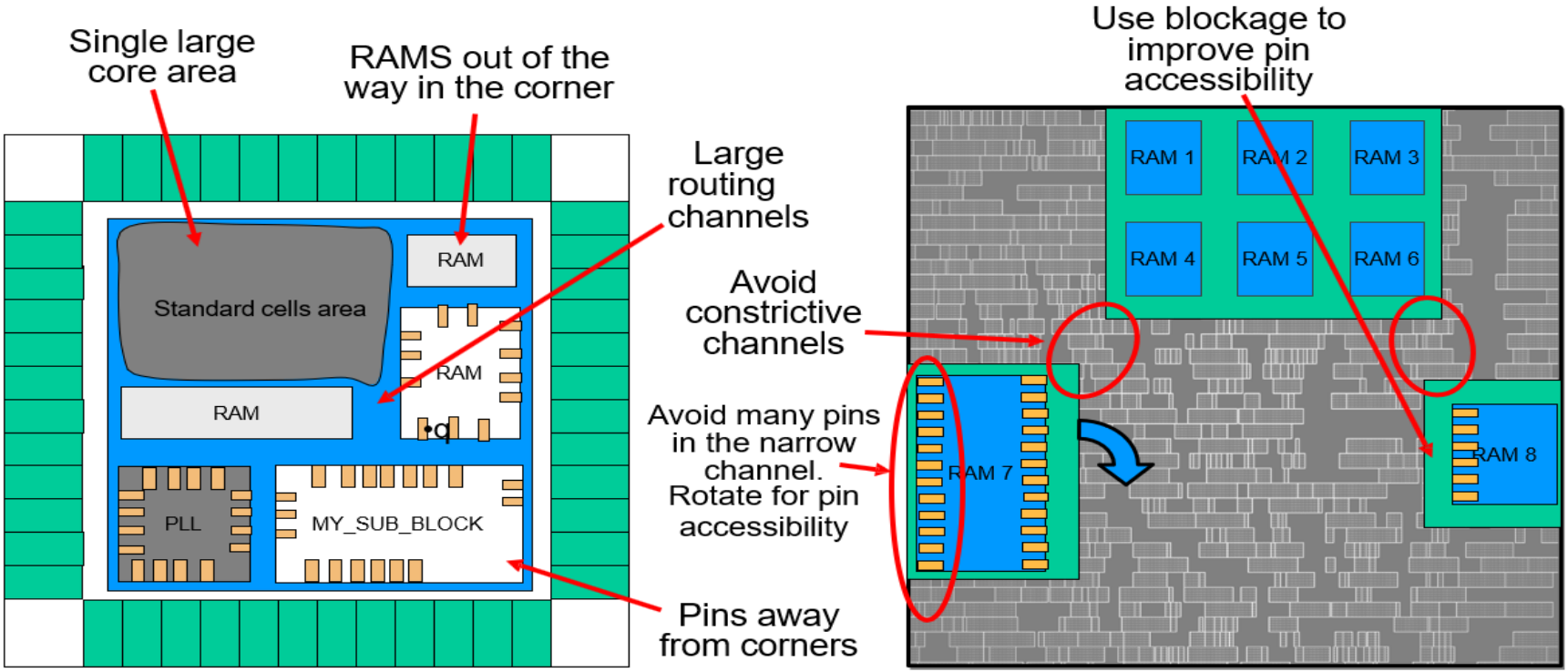


ROUTING BLOCKAGE

- Similar to placement blockage, routing blockage can be defined.
  - Blockage is defined for a given layer.



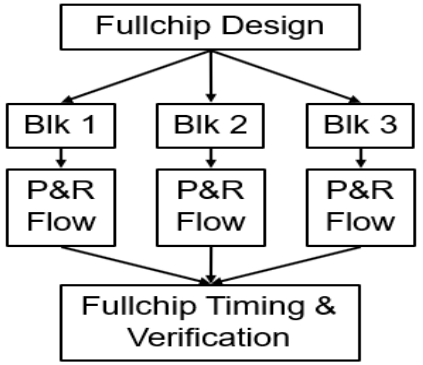
GUIDELINES FOR A GOOD FLOORPLAN

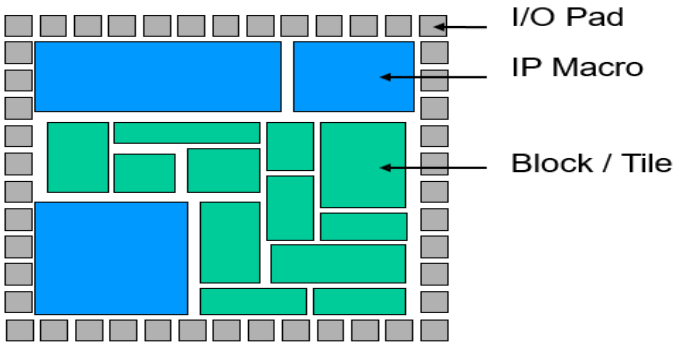


Hierarchical Design Concepts

FLAT VS. HIERARCHICAL DESIGN

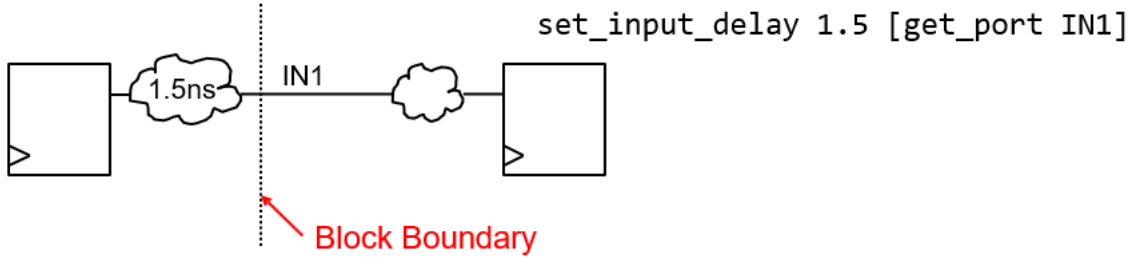
- If the design is too big, partition it into hierarchies
- **Advantages:**
  - Faster runtime, less memory needed for EDA tools
  - Faster ECO turn-around time
  - Ability to do design re-use
- **Disadvantages:**
  - Much more difficult for full-chip timing closure (ILMs)
  - More intensive design planning needed:
    - Feedthrough generation
    - Repeater insertion
    - Timing constraint budgeting
    - etc.



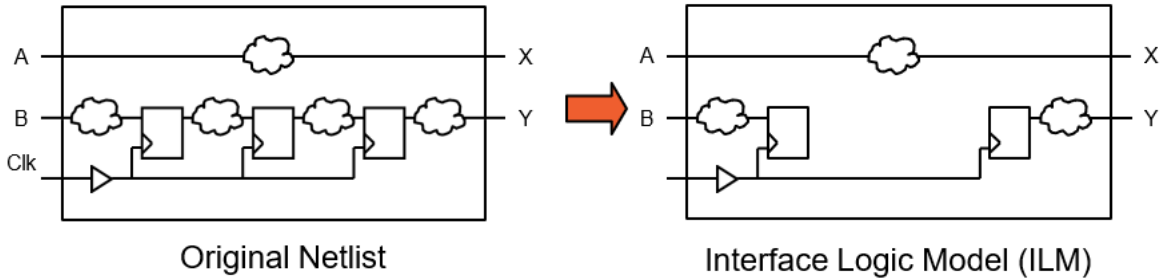


HIERARCHICAL DESIGN – TIME BUDGETING

- Chip level constraints must be mapped correctly to block level constraints as I/O constraints

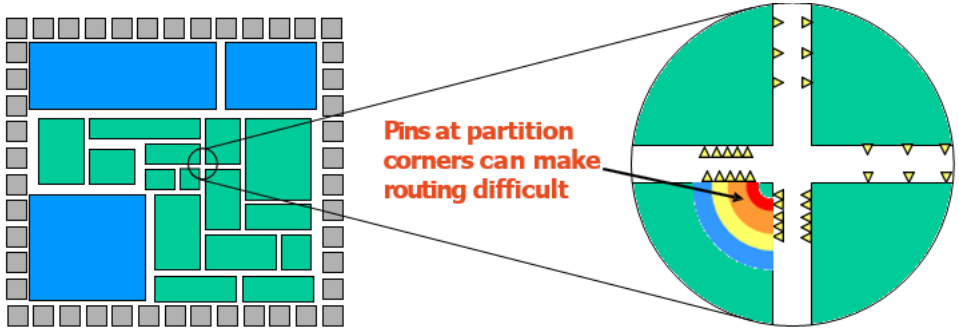


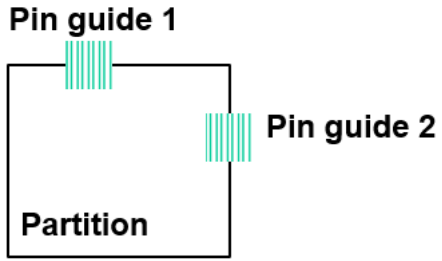
- Interface Logic Models (ILMs) help simplify and speed-up design



HIERARCHICAL DESIGN – PIN ASSIGNMENT

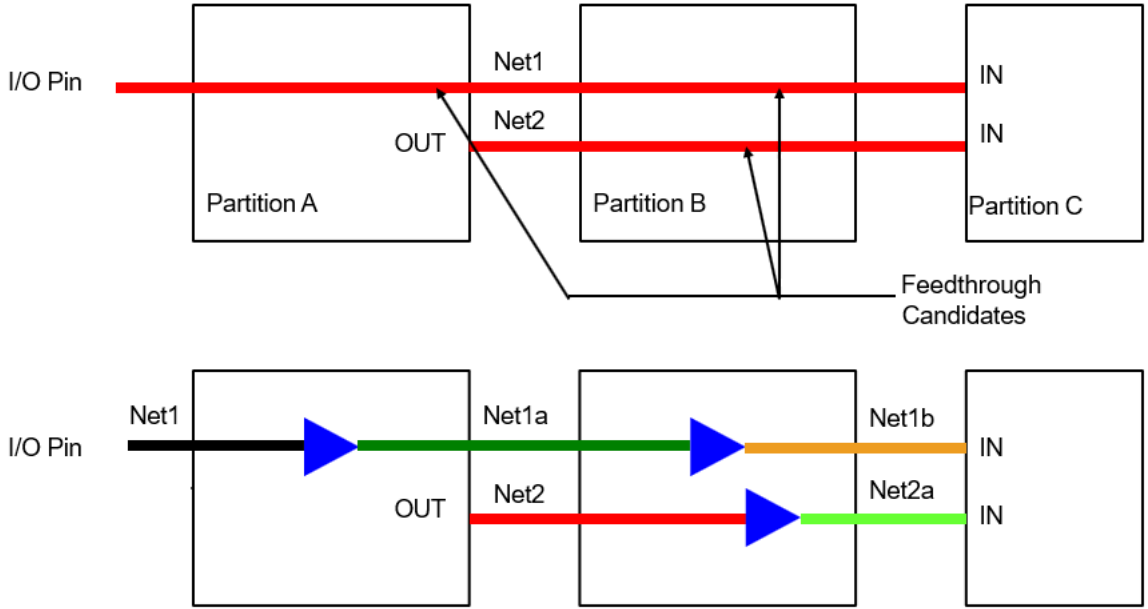
- Pin constraints include parameters, such as:
  - Layers, spacing, size, overlap
  - Net groups, pin guides
- Pins can be assigned:
  - Placement-based (flightlines)
  - Route-based (trial route, boundary crossings).
- Pin guides
  - Can be used to influence automatic pin placement of particular net groups





HIERARCHICAL DESIGN – FEEDTHROUGH

- For channel-less designs or designs with limited channel resources



Power Planning

POWER PLANNING

Dynamic Power

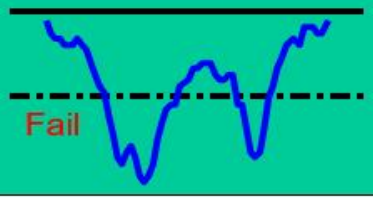
Static Power
(Leakage Power)

Floorplan
+
Design of the grid

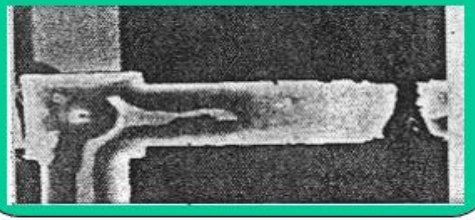
Average or Instantaneous Power problem

Power density problem in the Long run

IR-Drop / Voltage Droop

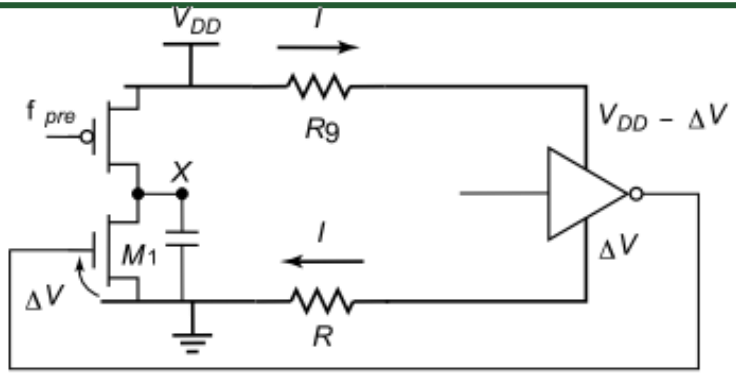


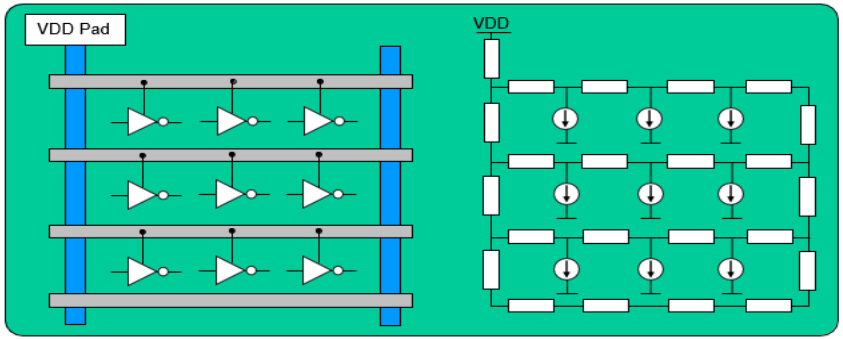
Electromigration (EM)

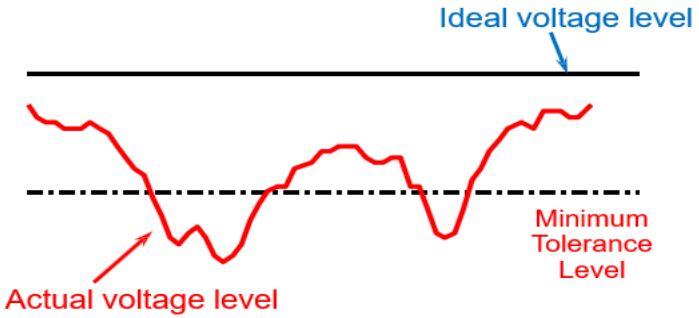


IR DROP

- The drop in supply voltage over the length of the supply line
  - A resistance matrix of the power grid is constructed
  - The average current of each gate is considered
  - The matrix is solved for the current at each node, to determine the IR-drop.





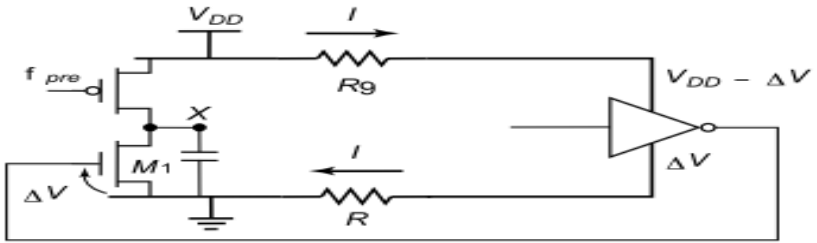


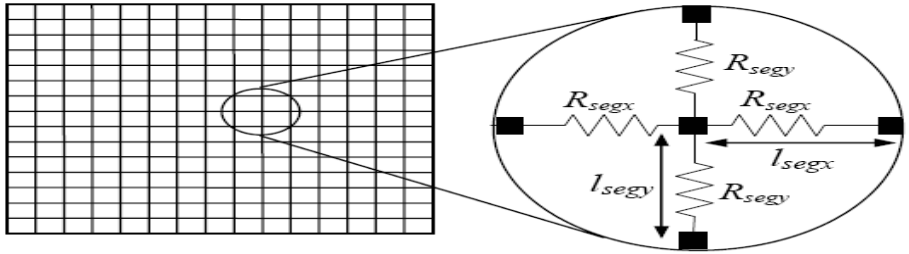
ELECTROMIGRATION (EM)

- Electromigration refers to the gradual displacement of the metal atoms of a conductor as a result of the current flowing through that conductor.
  - Transfer of electron momentum
- Can result in catastrophic failure do to either
  - Open : void on a single wire
  - Short : bridging between to wires
- Even without open or short, EM can cause performance degradation
  - Increase/decrease in wire RC

POWER DISTRIBUTION

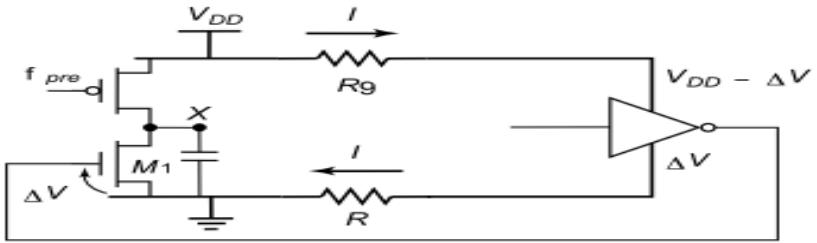
- Power Distribution Network functions
  - Carry current from pads to transistors on chip
  - Maintain stable voltage with low noise
  - Provide average and peak power demands
  - Provide current return paths for signals
  - Avoid electromigration & self-heating wearout
  - Consume little chip area and wire
  - Easy to lay out





POWER DISTRIBUTION

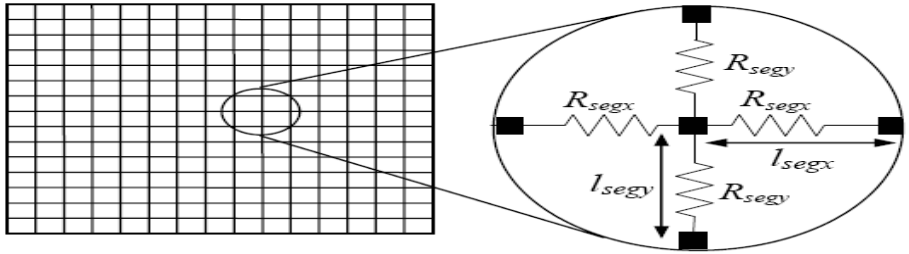
- Power Distribution Network functions
  - Carry current from pads to transistors on chip
  - Maintain stable voltage with low noise
  - Provide average and peak power demands
  - Provide current return paths for signals
  - Avoid electromigration & self-heating wearout
  - Consume little chip area and wire
  - Easy to lay out



More (Wider) Power Lines:
- Less Static (IR) drop
- Less Dynamic (*dI*/*dt*) drop
- Less Electromigration

BUT

More (Wider) Power Lines:
- Fewer (signal) routing resources (i.e., higher congestion)



POWER DISTRIBUTION CHALLENGE

- Assume we have a 1mm long power rail in M1.
  - Square resistance is given to be 0.1 ohm/square
  - If we make a 100nm wide rail, what is the resistance of the wire?

$$R = R_{\square} L/W = 0.1\,\Omega/\square \cdot \frac{10^{-3}\,m}{100 \cdot 10^{-9}\,m} = 1000\Omega$$

  - Now, given a max current of 1mA/1um, due to Electromigration, what is the IR drop when conducting such a current through this wire?

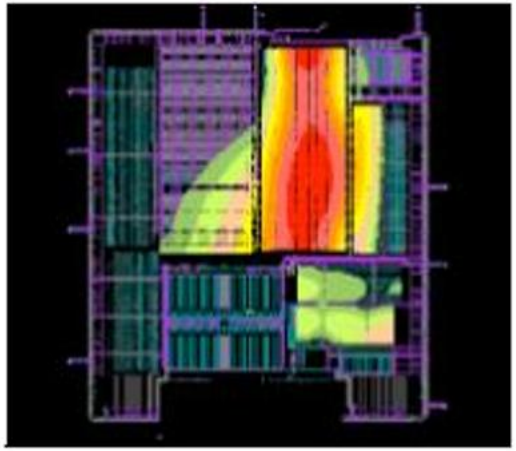
$$I_{\max} = \frac{1\text{mA}}{1\mu\text{m}} \cdot 100\text{nm} = 0.1\text{mA}$$

$$IR_{drop} = I_{\max} \cdot R_{wire} = 10^{-4} \cdot 10^{3} = 100\text{mV}$$

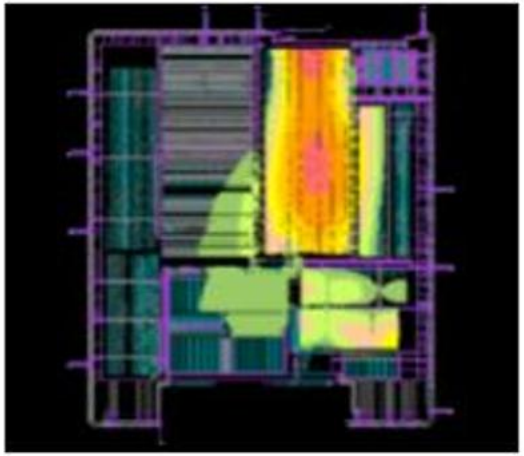
- So what do we do?
  - Make the power rails as wide and as thick as possible!

HOT SPOTS

- We generally map the IR drop of a chip using a color map to highlight “hot spots”, where the IR drop is bad.



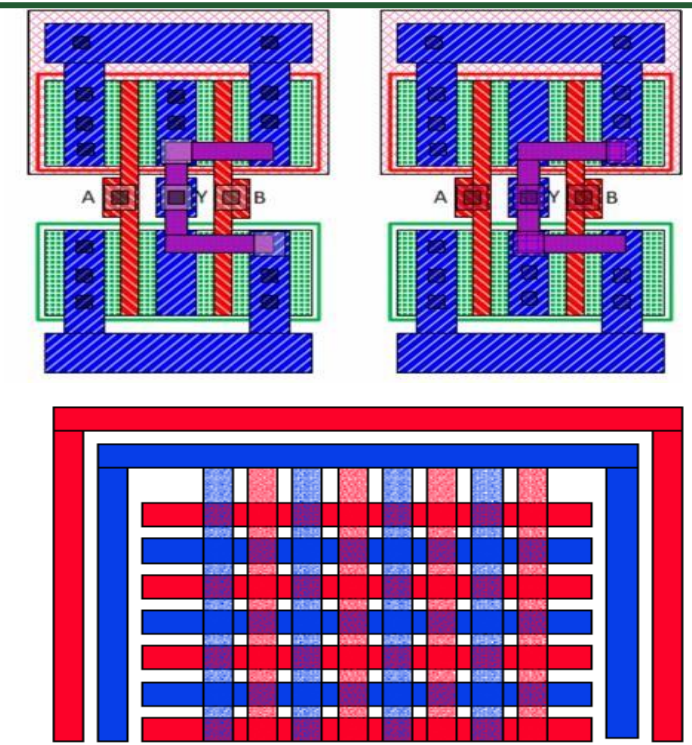
Initial IR Drop Mapping



Source:
Cadence

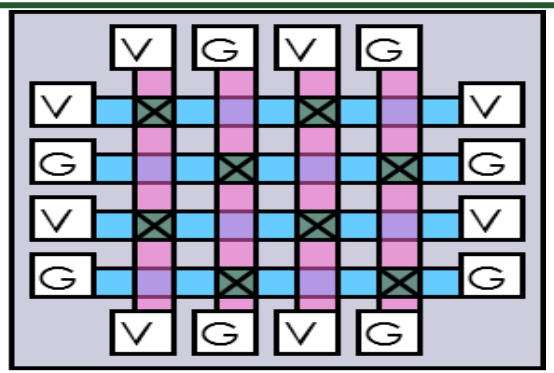
POWER AND GROUND ROUTING

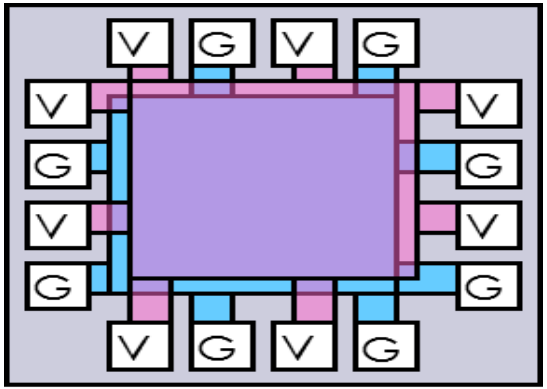
- Each standard cell or macro has power and ground signals, i.e., VDD (power) and GND (ground)
  - They need to be connected as well
- Power/Ground mesh will allow multiple paths from P/G sources to destinations
  - Less series resistance
  - Hierarchical power and ground meshes from upper metal layers to lower metal layers
  - Multiple vias between layers
- You can imagine that they are HUGE NETWORKS!
  - In general, P/G routings are pretty regular
  - P/G routing resources are usually reserved



STANDARD APPROACHES TO POWER ROUTING

- Power grid
  - Interconnected vertical and horizontal power bars.
  - Common on most high-performance designs.
  - Often well over half of total metal on upper thicker layers used for VDD/GND.
- Dedicated VDD/GND planes.
  - Very expensive.
  - Only used on Alpha 21264, Dropped on subsequent Alphas.
  - Simplified circuit analysis.
- Some thoughts/trends:
  - P/G I/O pad co-optimization with classic physical design
  - Decoupling capacitors to reduce P/G related voltage drop
  - Multiple voltage/frequency islands make the P/G problem and clock distributions more challenging.

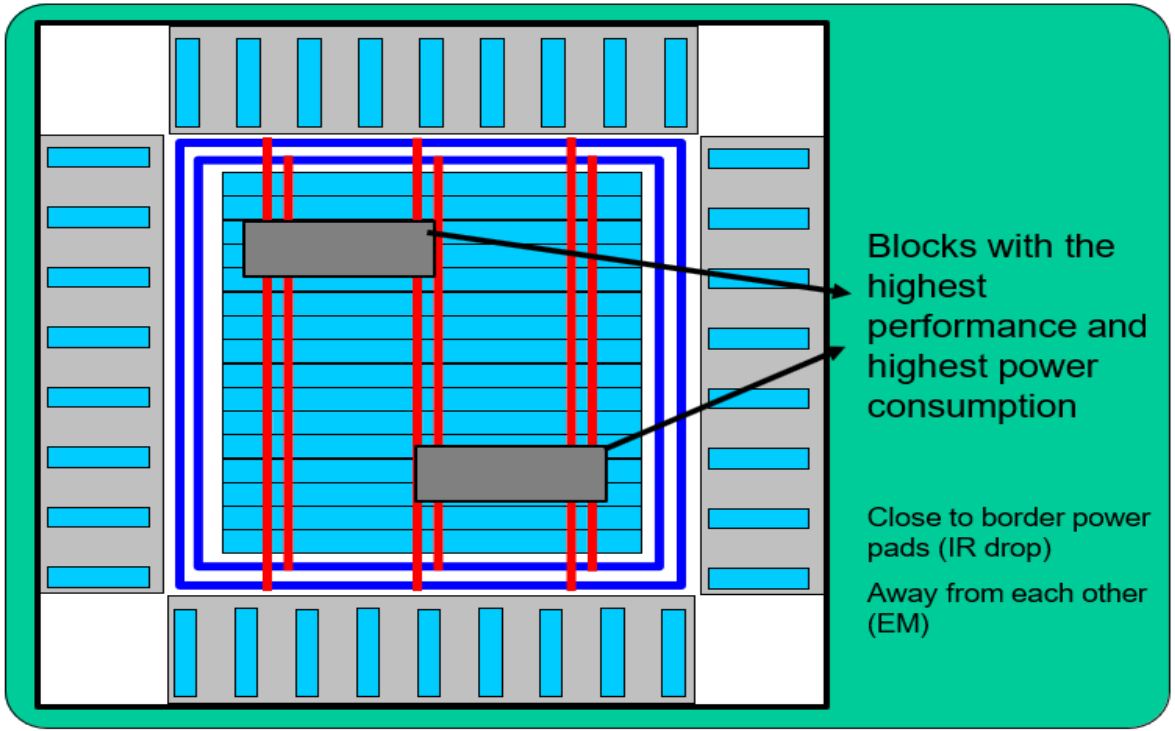




POWER GRID CREATION

- Tradeoff IR drop and EM versus routing resources
  - Require power budget
    - Initial power estimation
    - Average current, max current density
- Need to determine
  - General grid structure (gating or multi-voltage?)
  - Number and location of power pads (per voltage)
  - Metal layers to be used
  - Width and spacing of straps
  - Via stacks versus available routing tracks
  - Rings / no rings
  - Hierarchical block shielding
- Run initial power network analysis to confirm design

POWER GRID CREATION – MACRO PLACEMENT



CLOCK PLANNING

- Clocking scheme is planned at this stage
- We will talk about the schemes in the upcoming lectures.

Thank you!