

# Topics



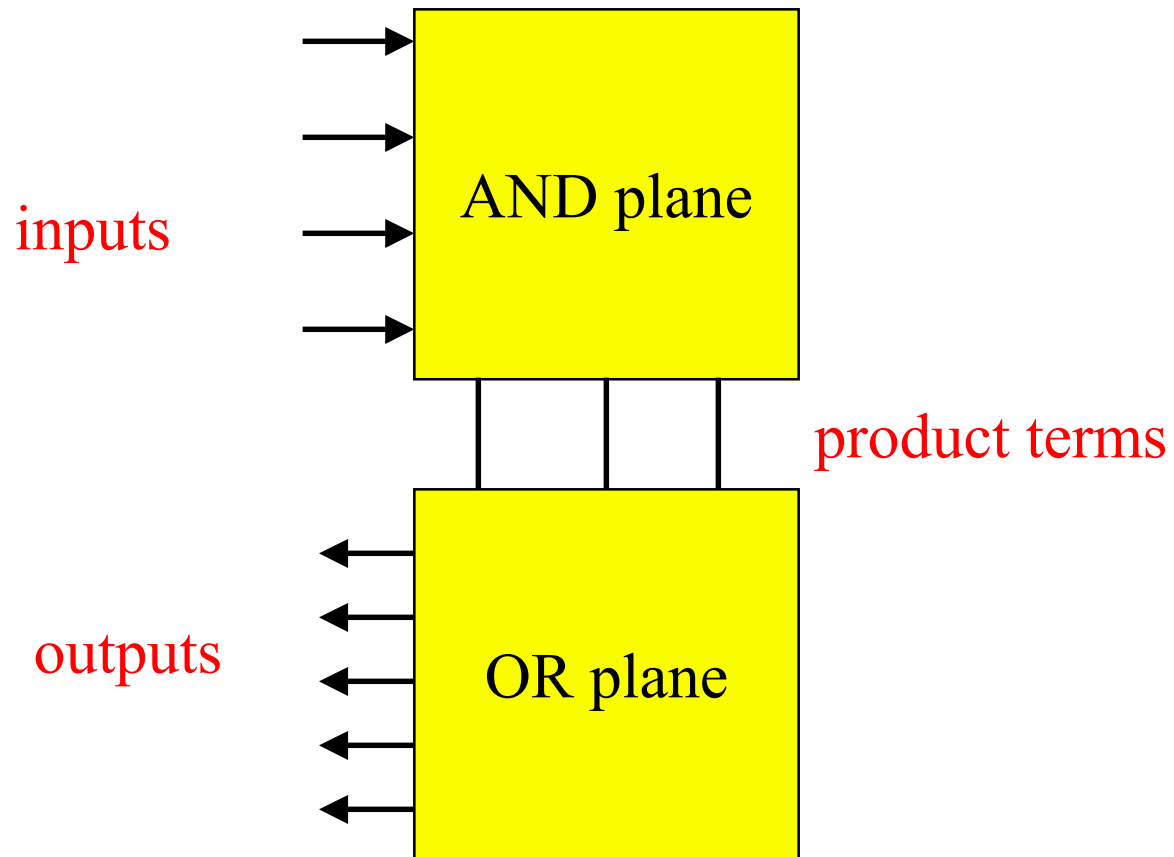
- PLAs.
- Memories:
  - ROM;
  - SRAM;
  - DRAM.
- Datapaths.
- Floor Planning

# Programmable logic array (PLA)



- Used to implement specialized logic functions.
- A PLA decodes only some addresses (input values); a ROM decodes all addresses.
- PLA not as common in CMOS as in nMOS, but is used for some logic functions.

# PLA organization

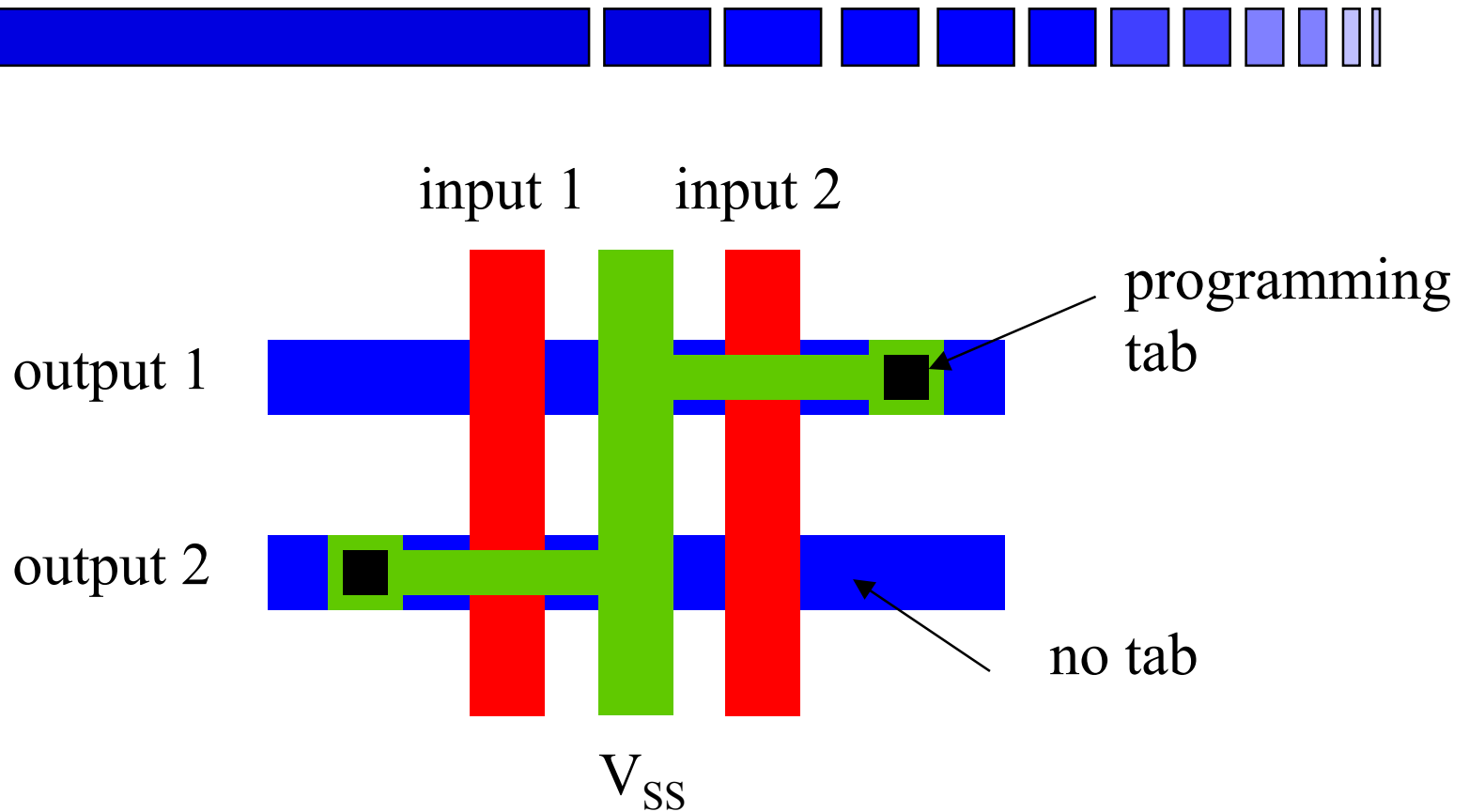


# PLA structure

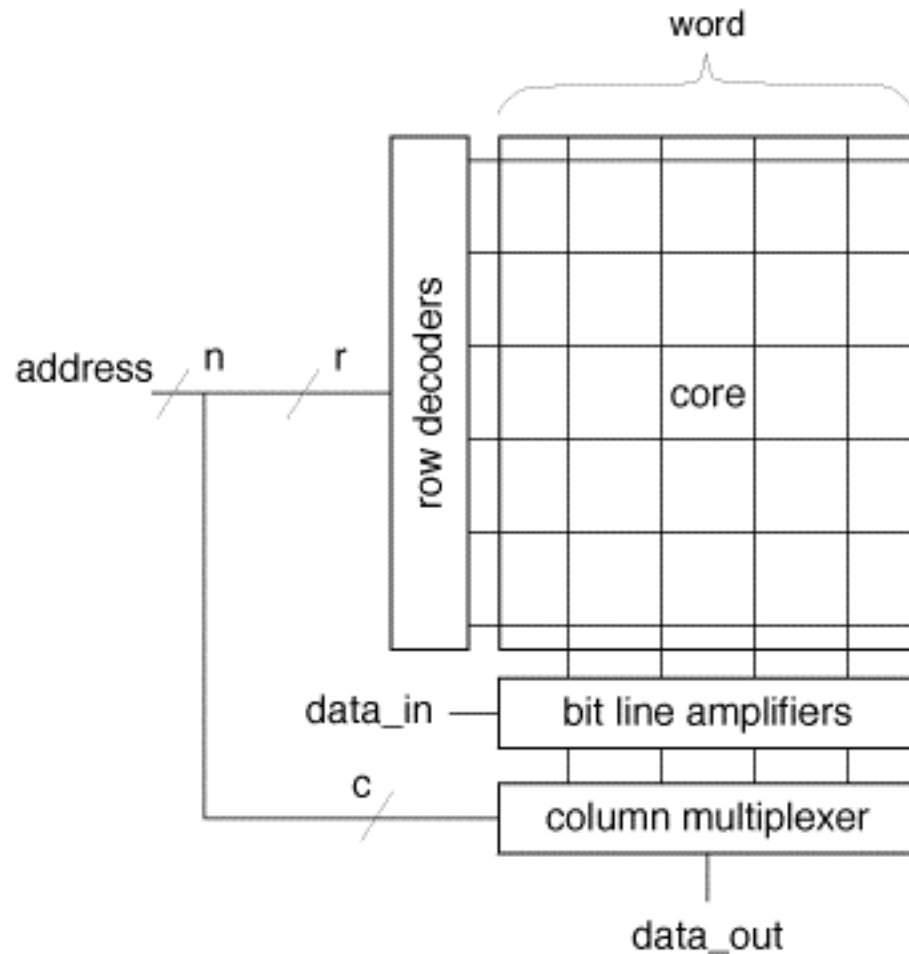


- AND plane, OR plane, inverters together form complete two-level logic functions.
- Both AND and OR planes are implemented as NOR circuits.
- Pulldown transistors form programming/personality of PLA. Transistors may be referred to as programming tabs.

# PLA AND/OR cell



# High-density memory architecture



# Memory operation



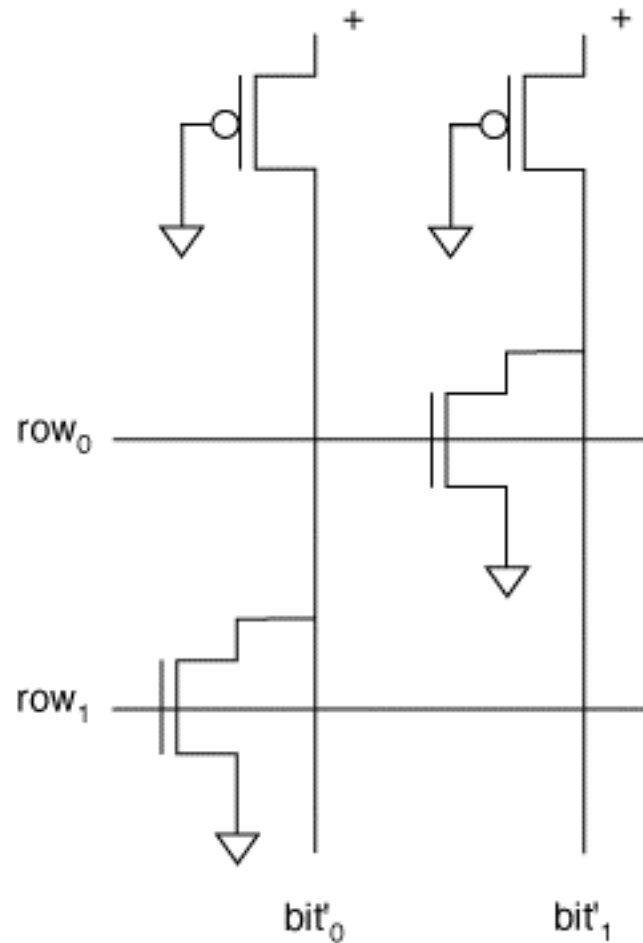
- Address is divided into row, column.
  - Row may contain full word or more than one word.
- Selected row drives/senses bit lines in columns.
- Amplifiers/drivers read/write bit lines.

# Read-only memory (ROM)



- ROM core is organized as NOR gates—pulldown transistors of NOR determine programming.
- Erasable ROMs require special processing that is not typically available.
- ROMs on digital ICs are generally mask-programmed—placement of pulldowns determines ROM contents.

# ROM core circuit

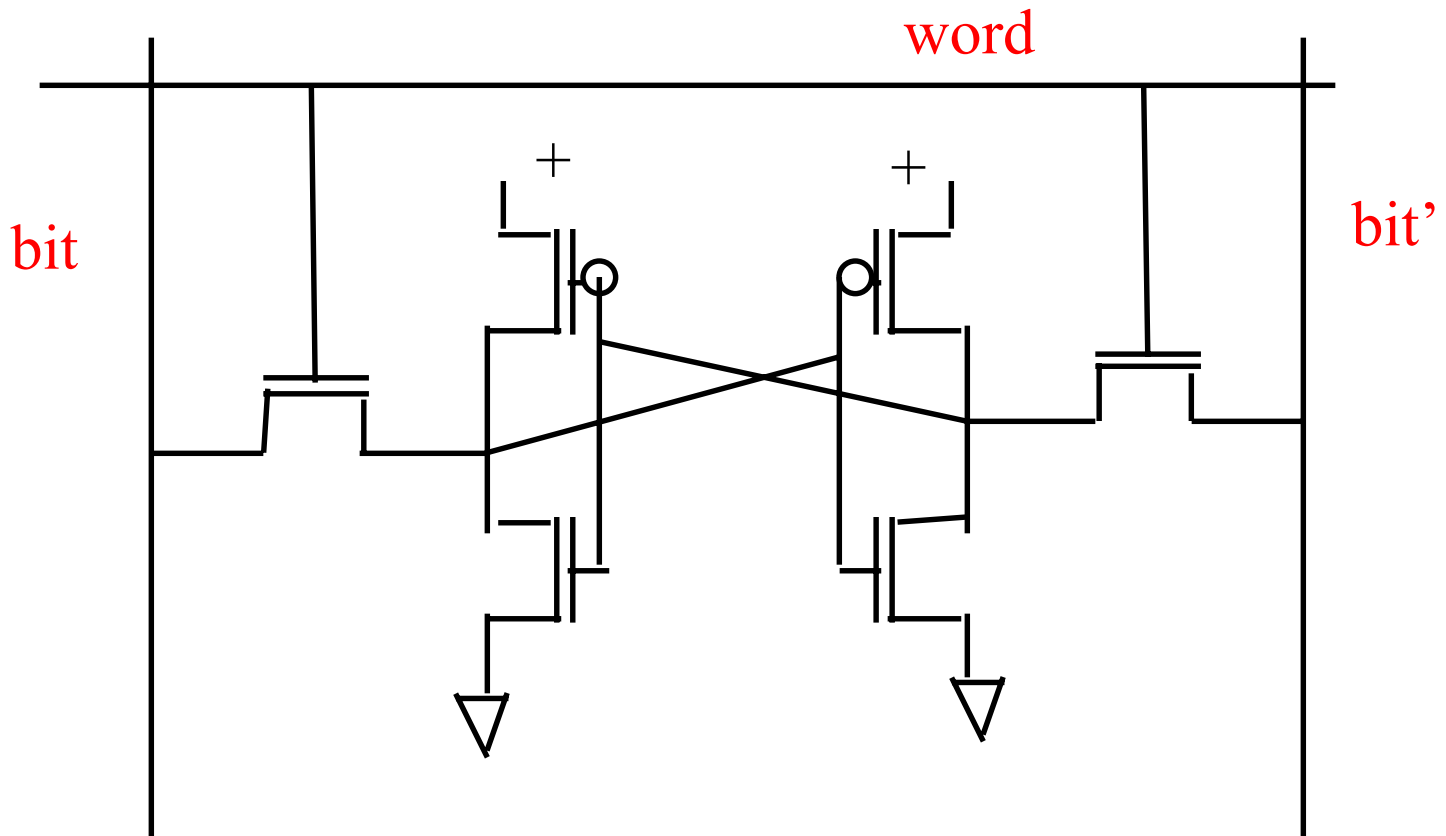


# Static RAM (SRAM)



- Core cell uses six-transistor circuit to store value.
- Value is stored symmetrically—both true and complement are stored on cross-coupled transistors.
- SRAM retains value as long as power is applied to circuit.

# SRAM core cell

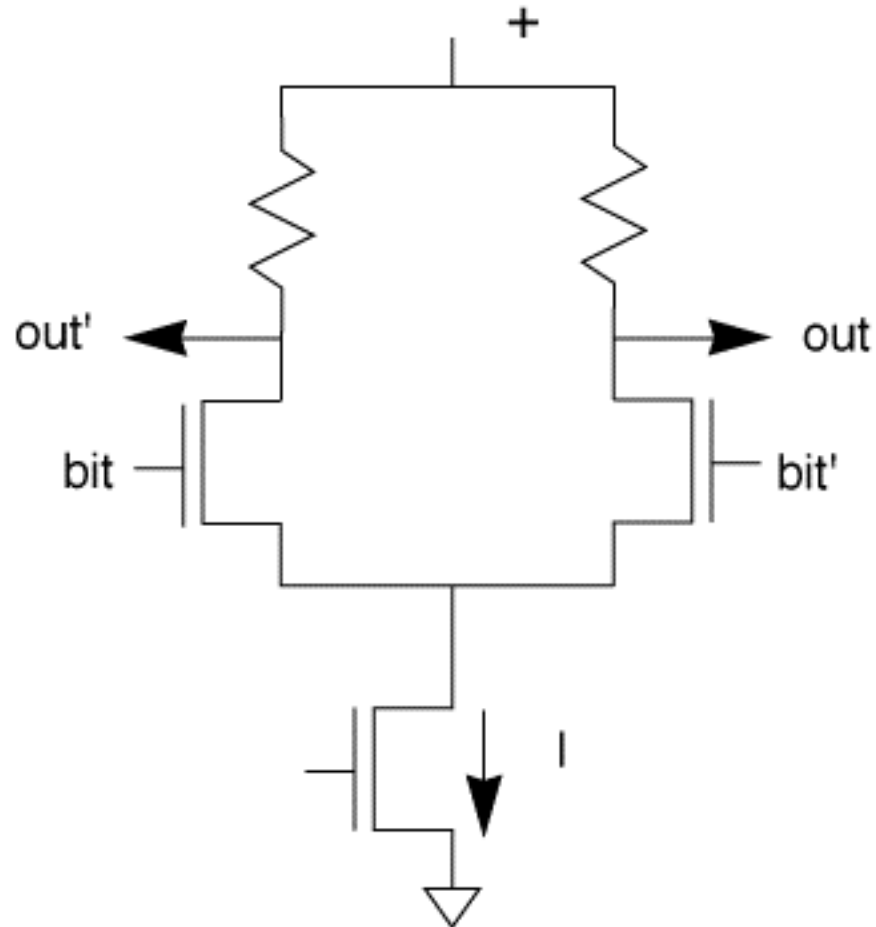


# SRAM core operation



- **Read:**
  - precharge bit and bit' high;
  - set select line high from row decoder;
  - one bit line will be pulled down.
- **Write:**
  - set bit/bit' to desired (complementary) values;
  - set select line high;
  - drive on bit lines will flip state if necessary.

# SRAM sense amp

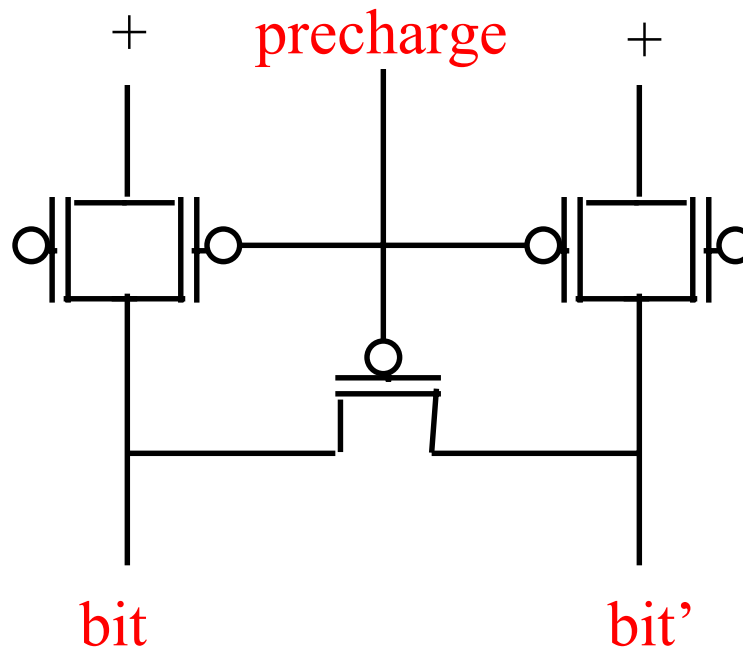


# Sense amp operation



- Differential pair—takes advantage of complementarity of bit lines.
- When one bit line goes low, that arm of diff pair reduces its current, causing compensating increase in current in other arm.
- Sense amp can be cross-coupled to increase speed.

# SRAM precharge circuit

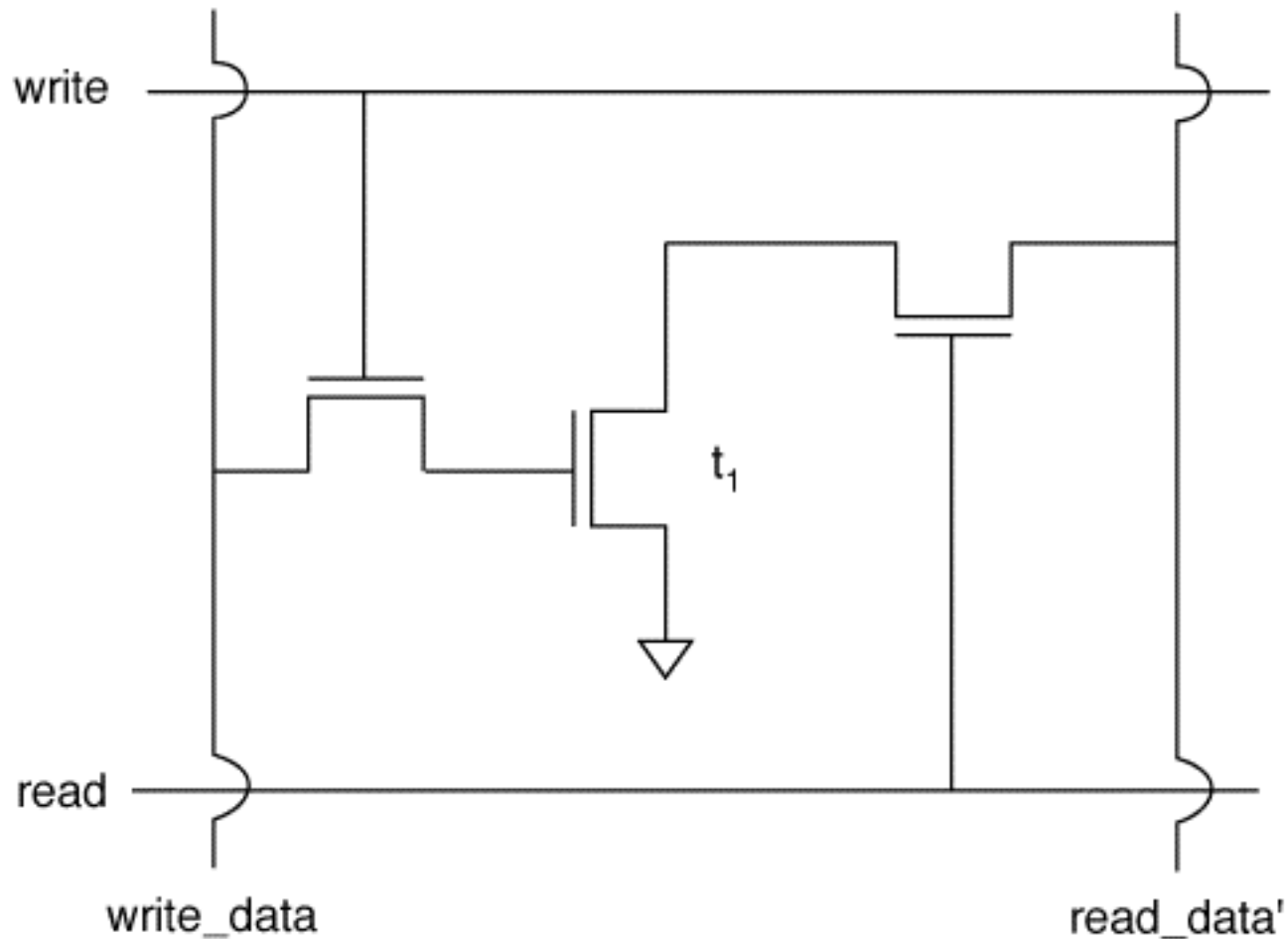


# 3-transistor dynamic RAM (DRAM)



- First form of DRAM—modern commercial DRAMs use one-transistor cell.
- 3-transistor cell can easily be made with a digital process.
- Dynamic RAM loses value due to charge leakage—must be refreshed.

# 3-T DRAM core cell



# 3-T DRAM operation



Value is stored on gate capacitance of  $t_1$ .

- **Read:**
  - read = 1, write = 0, read\_data' is precharged;
  - $t_1$  will pull down read\_data' if 1 is stored.
- **Write:**
  - read = 0, write = 1, write\_data = value;
  - guard transistor writes value onto gate capacitance.

# Data paths

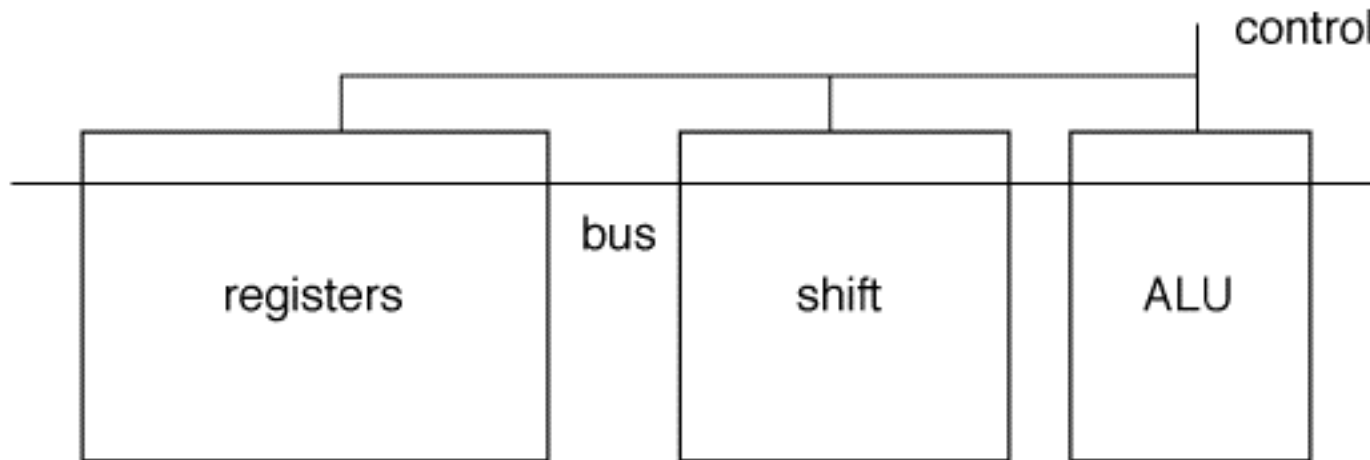


- A data path is a logical and a physical structure:
  - bitwise logical organization;
  - bitwise physical design.
- Datapath often has ALU, registers, some other function units.
- Data is generally passed via busses.

# Typical data path structure



Slice includes one bit of function units,  
connected by busses:

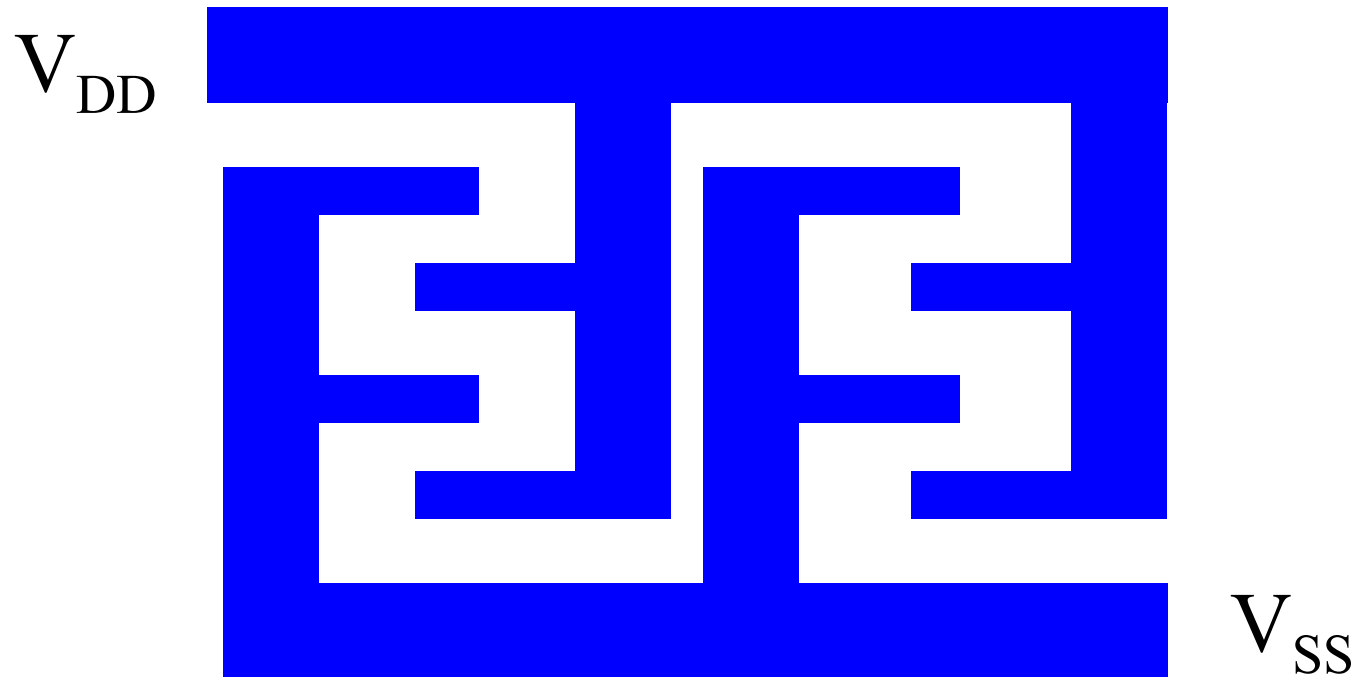


# Power distribution



- Must size wires to be able to handle current—requires designing topology of  $V_{DD}/V_{SS}$  networks.
- Want to keep power network in metal—requires designing planar wiring.

# Interdigitated power and ground lines



# Power tree design



- Each branch must be able to supply required current to all of its subsidiary branches:

$$I_x = \sum_{b \in x} I_b$$

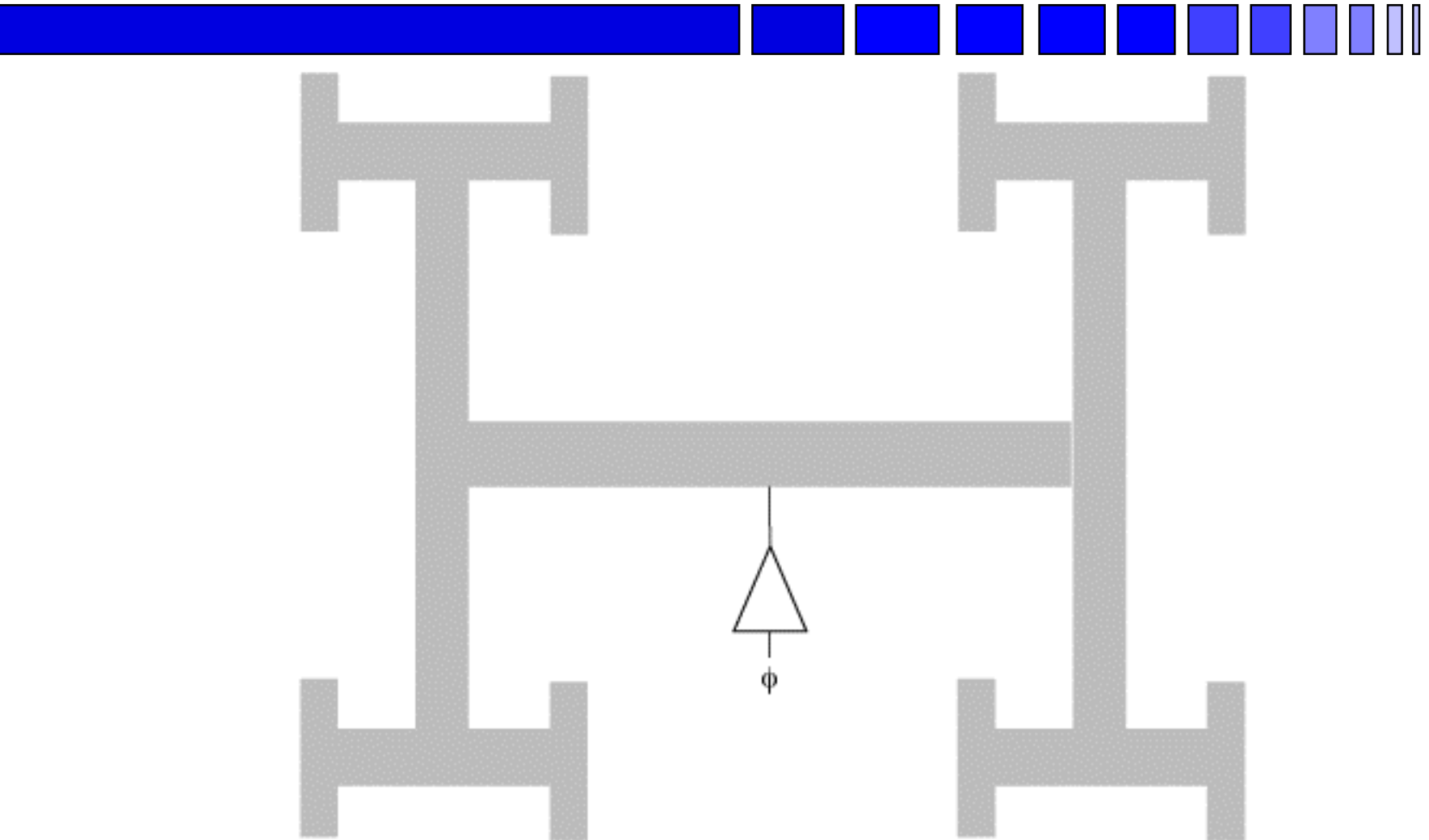
- Trees are interdigitated to supply both sides of power supply.

# Clock distribution



- Goals:
  - deliver clock to all memory elements with acceptable skew;
  - deliver clock edges with acceptable sharpness.
- Clocking network design is one of the greatest challenges in the design of a large chip.

# H-tree

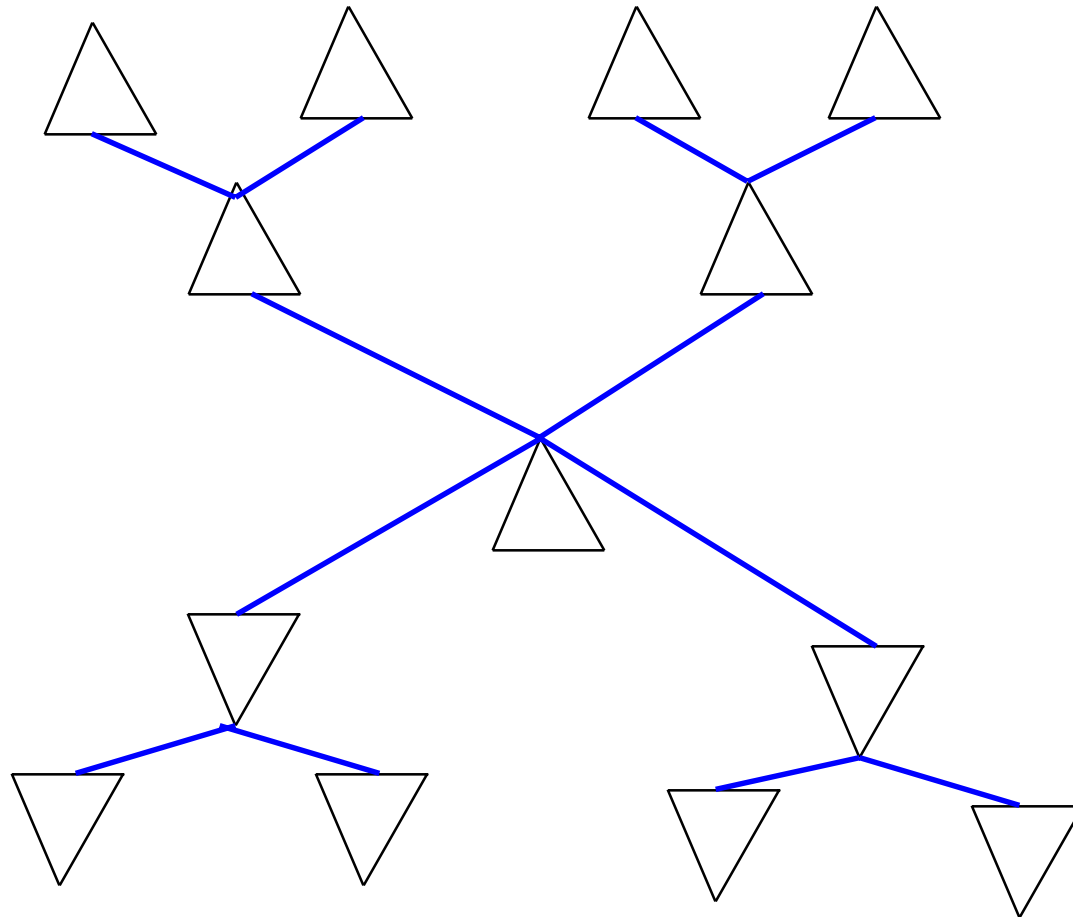


# Clock distribution tree



- Clocks are generally distributed via wiring trees.
- Want to use low-resistance interconnect to minimize delay.
- Use multiple drivers to distribute driver requirements—use optimal sizing principles to design buffers.
- Clock lines can create significant crosstalk.

# Clock distribution tree example



# Floorplanning tips



- **Develop a wiring plan.** Think about how layers will be used to distribute important wires.
- **Sweep small components into larger blocks.** A floorplan with a single NAND gate in the middle will be hard to work with.
- **Design wiring that looks simple.** If it looks complicated, it is complicated.

# Floorplanning tips, cont'd.



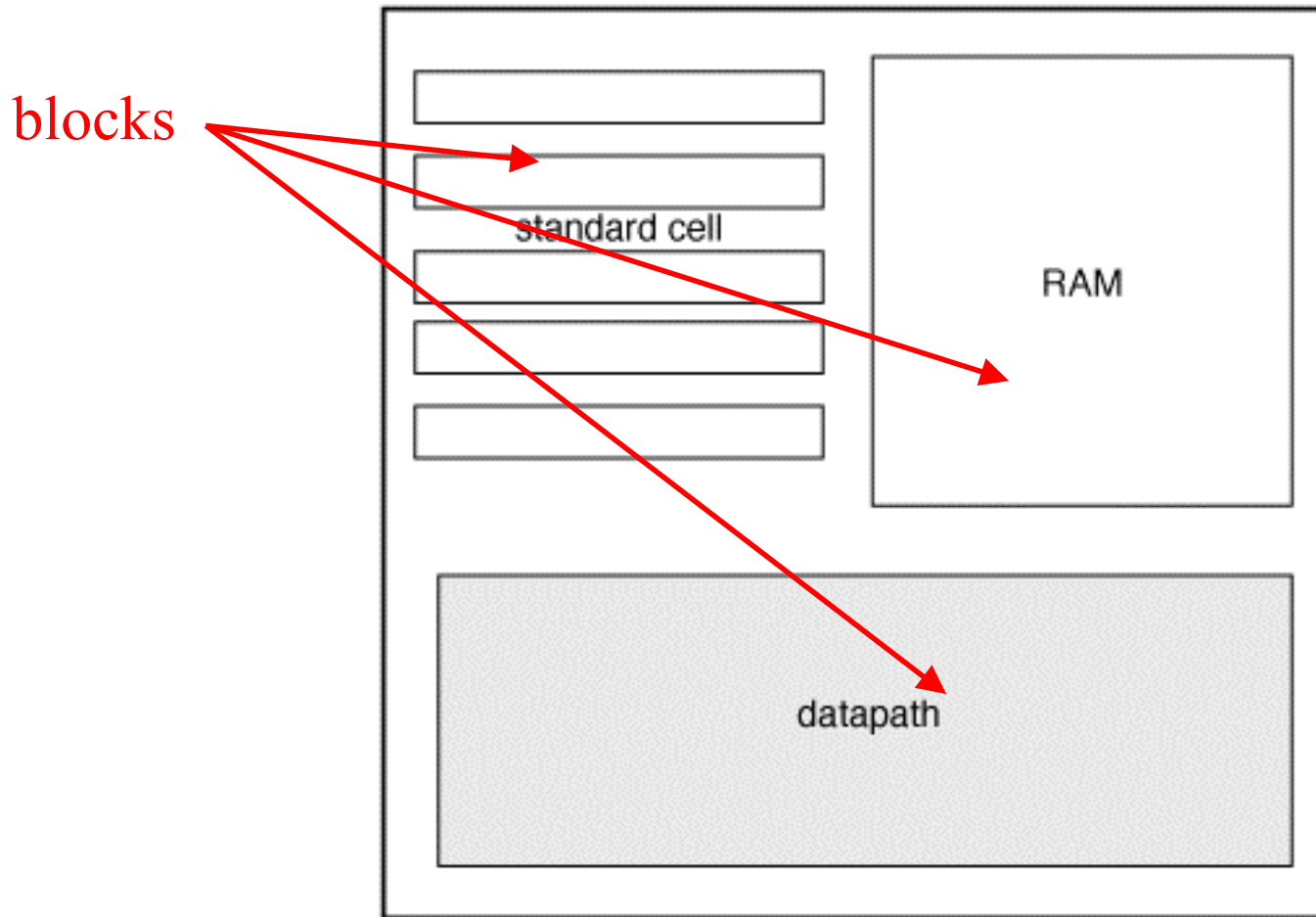
- **Design planar wiring.** Planarity is the essence of simplicity. It isn't always possible, but do it where feasible (and where it doesn't introduce unacceptable delay).
- **Draw separate wiring plans for power and clocking.** These are important design tasks which should be tackled early.

# Floorplanning strategies



- Floorplanning must take into account blocks of varying function, size, shape.
- Must design:
  - space allocation;
  - signal routing;
  - power supply routing;
  - clock distribution.

# Bricks-and-mortar floorplan



# Purposes of floorplanning



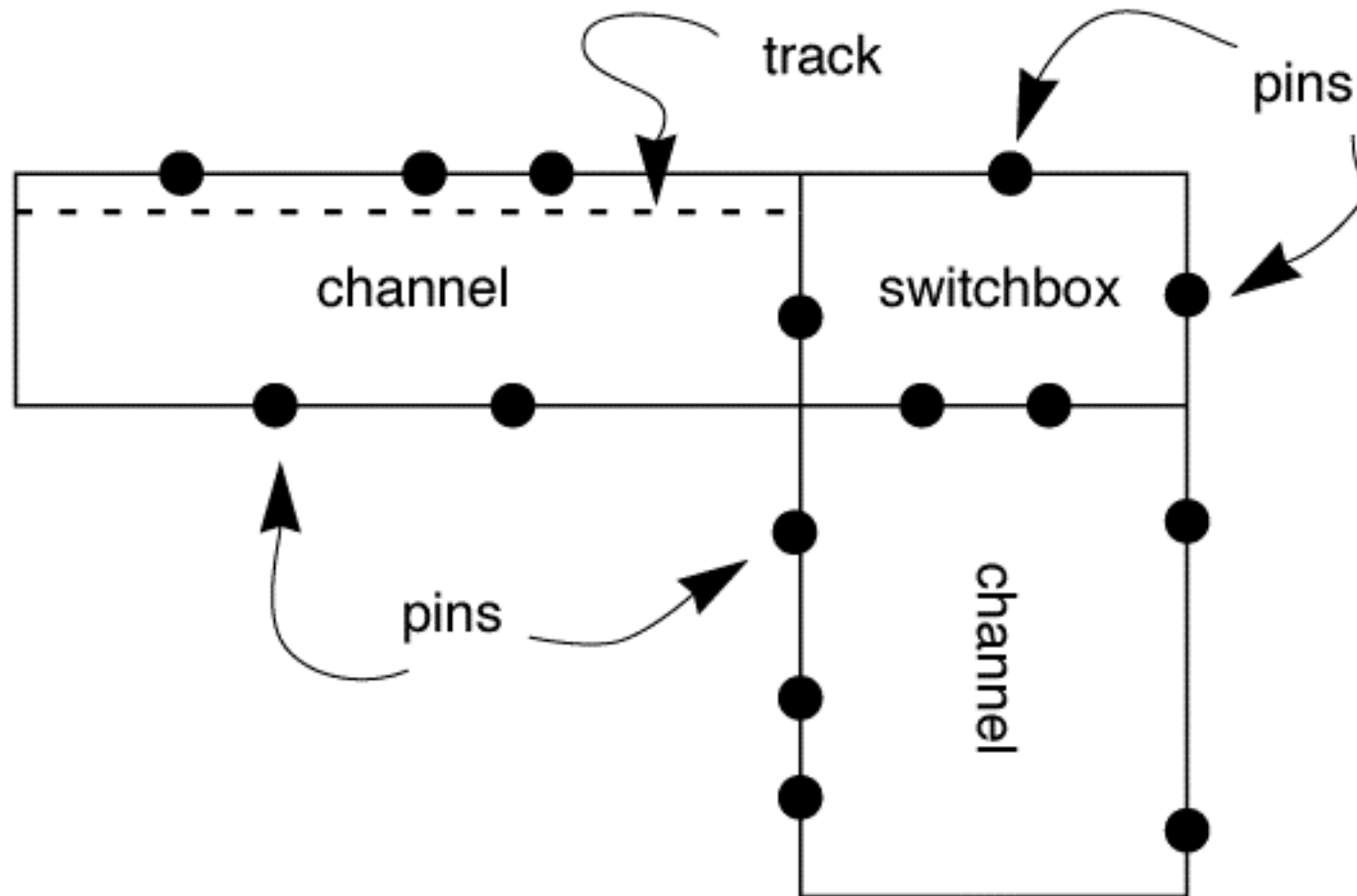
- **Early in design:**
  - Prepare a floorplan to budget area, wire area/delay. Tradeoffs between blocks can be negotiated.
- **Late in design:**
  - Make sure the pieces fit together as planned.
  - Implement the global layout.

# Types of routing



- Channel routing:
  - channel may grow in one dimension to accommodate wires;
  - pins generally on only two sides.
- Switchbox routing:
  - cannot grow in any dimension;
  - pins are on all four sides, fixing dimensions of the box.

# Channels and switchboxes



# Block placement



- Blocks have:
  - area;
  - aspect ratio.
- Blocks may be placed at different rotations and reflections.
- Uniform size blocks are easier to interchange.

# Blocks and wiring

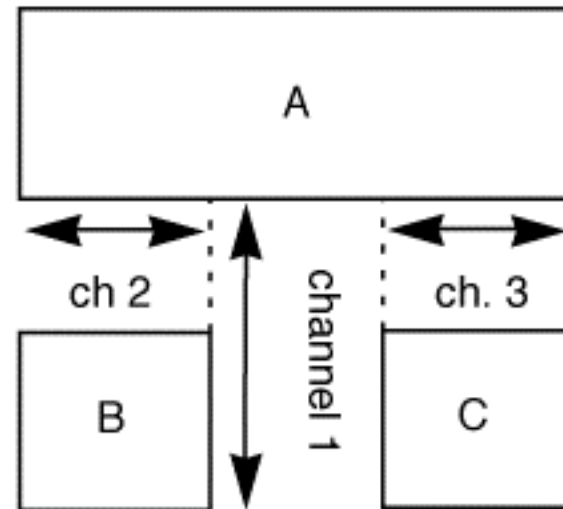
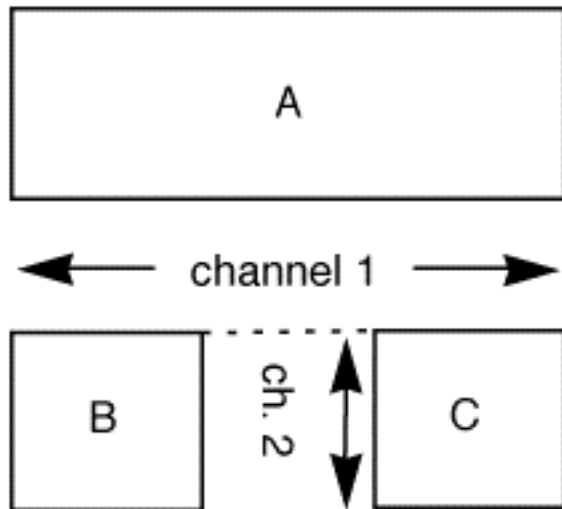


- Cannot ignore wiring during block placement—large wiring areas may force rearrangement of blocks.
- Wiring plan must consider area and delay of critical signals.
- Blocks divide wiring area into routing channels.

# Channel definition



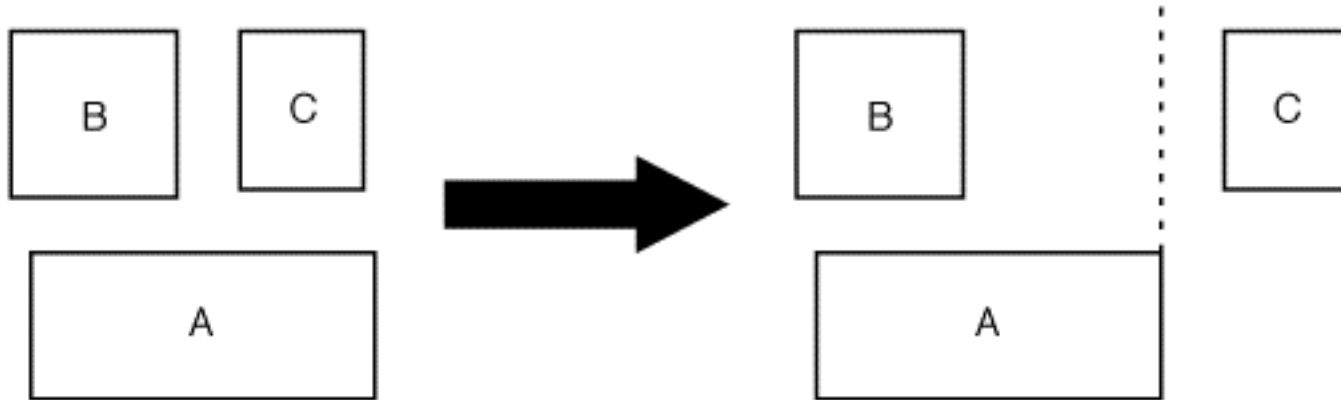
- Channels end at block boundaries.
- Several alternate channel definitions are possible:



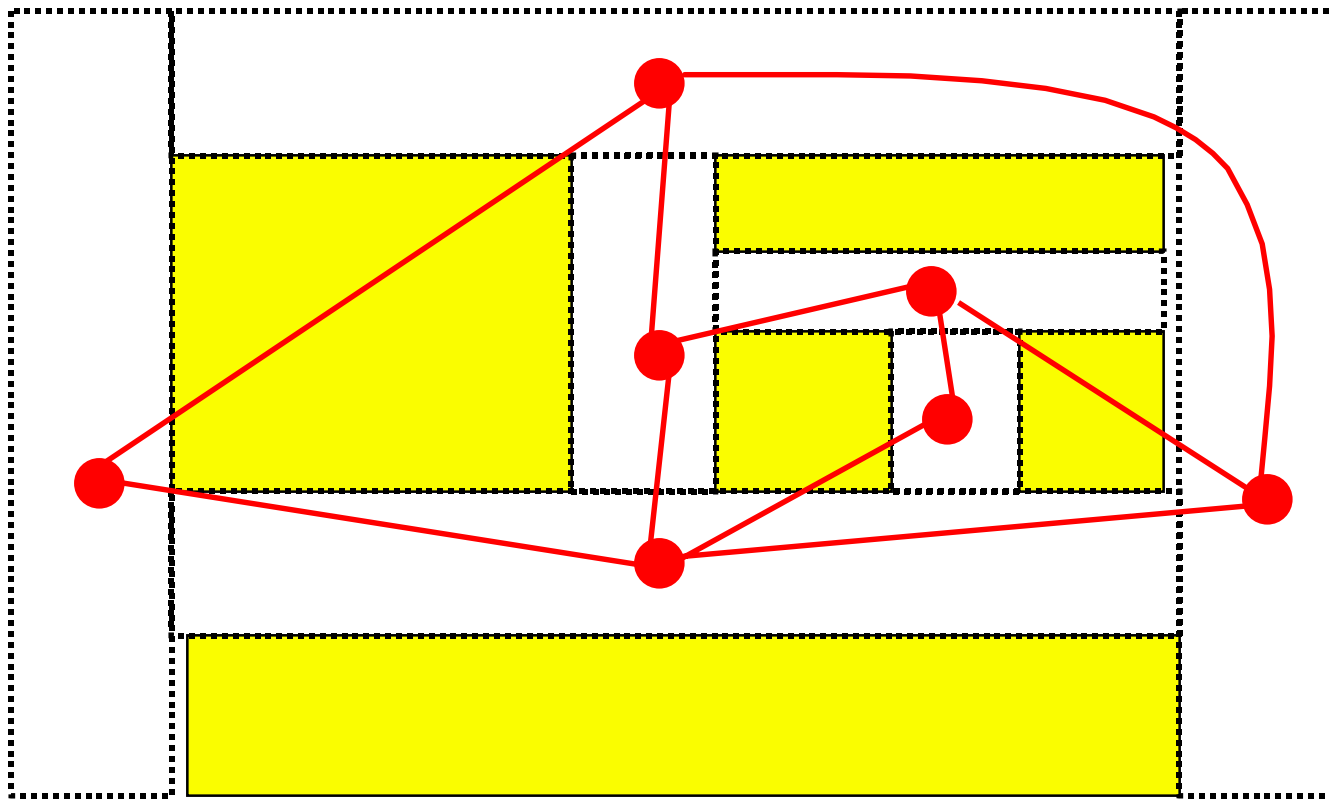
# Channel definition changes with block spacing



Changing spacing changes relationship  
between block edges:



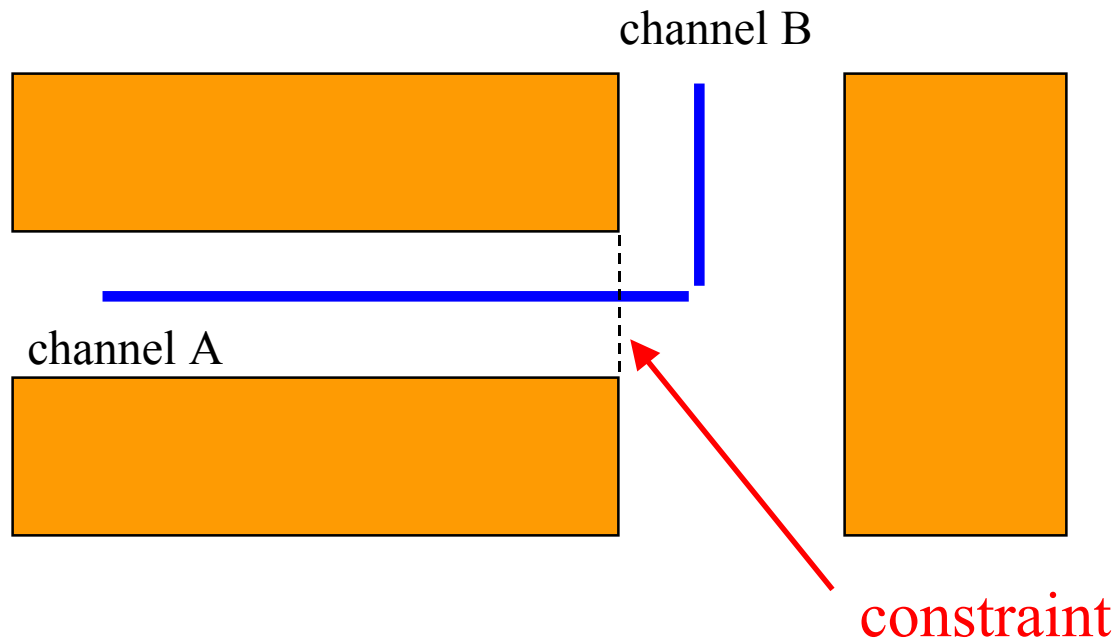
# Channel graph



# Channels must be routed in order



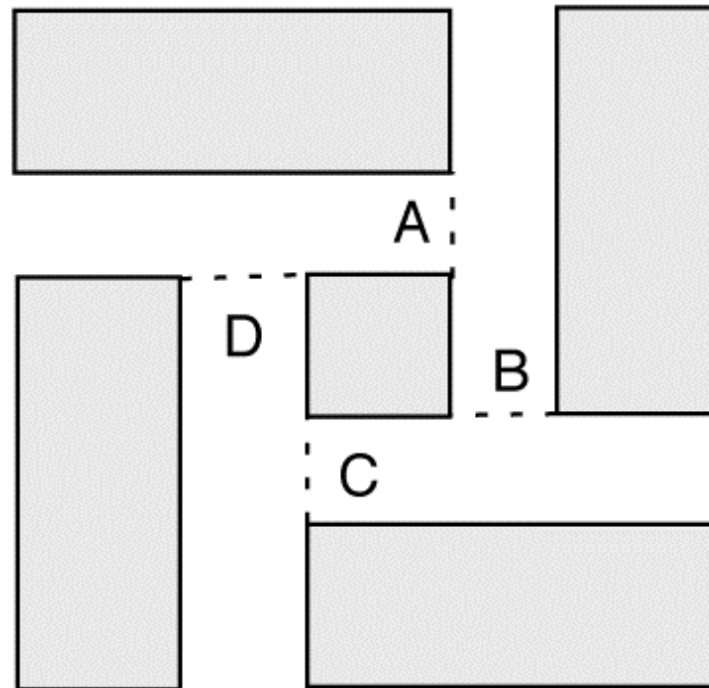
Wire out of end of one channel creates pin on side of next channel:



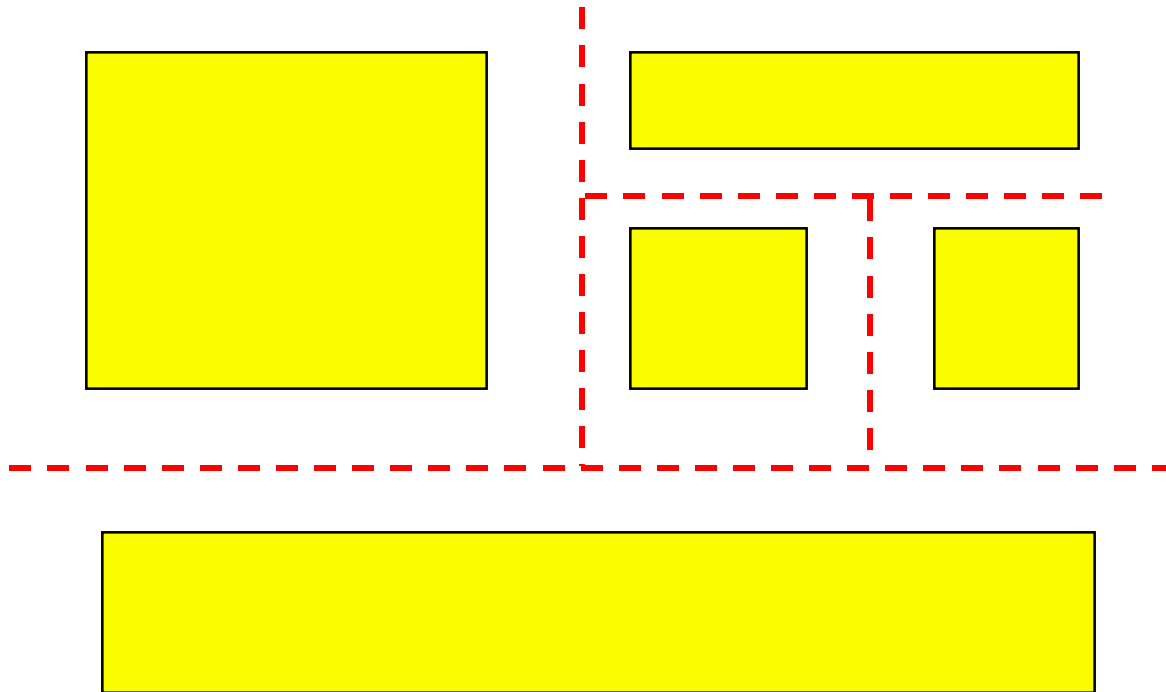
# Windmills



Can create an unroutable combination of channels with circular constraints:



# Slicable floorplan



# Slicability property



- A slicable floorplan can be recursively cut in two without cutting any blocks.
- A slicable floorplan is guaranteed to have no windmills, therefore guaranteed to have a feasible order of routing for the channels.
- Slicability is a desirable property for floorplans.

# Global routing



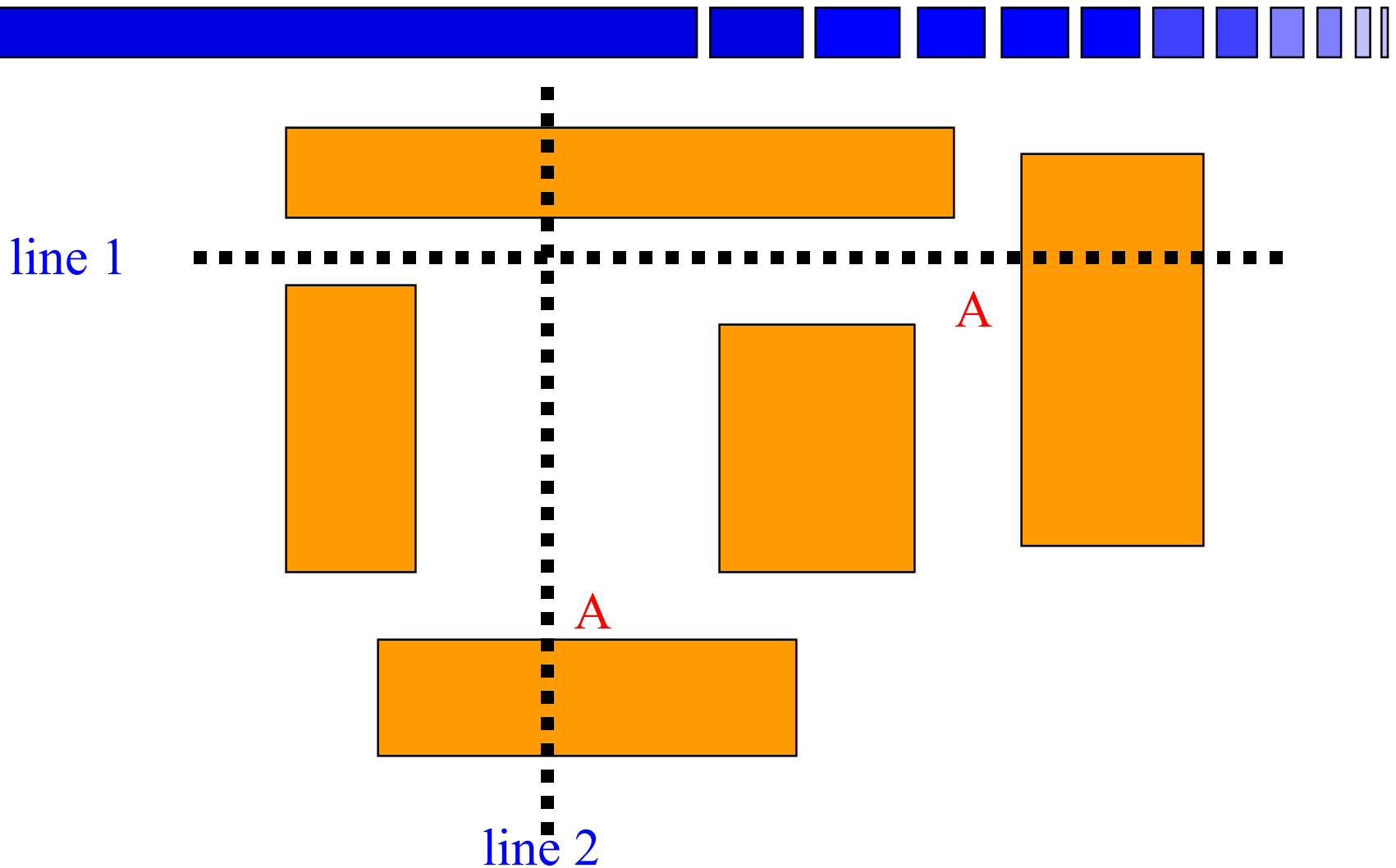
- Goal: assign wires to paths through channels.
- Don't worry about exact routing of wires within channel.
- Can estimate channel height from global routing using congestion.

# Line probe routing



- Heuristic method for finding a short route.
- Works with arbitrary combination of obstacles.
- Does not explore all possible paths—not optimal.

# Line probe example



# Channel utilization



- n Want to keep all channels about equally full to minimize wasted area.
- n Important to route time-critical signals first.
- n Shortest path may not be best for global wiring.
- n In general, may need to rip-up wires and reroute to improve the global routing.