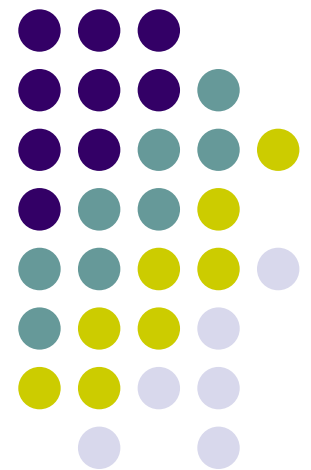


CS159 Introduction to Computational Complexity

The VLSI Model I



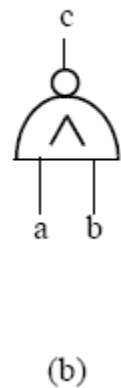
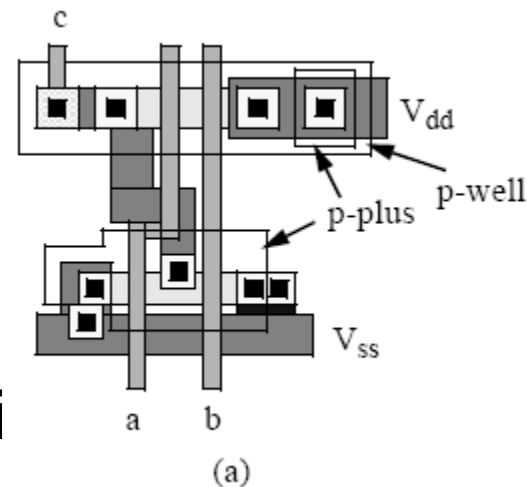
The VLSI Revolution



- **Time Line:**

- 1947 - transistor invented
- 1958-9 - integrated circuit
- 1970's - small CPU on chip
- 2000's - $> 10^8$ transistors/chip

- Gates constructed of overlays of rectangular sections of materials.





The VLSI Model

- **Architectural Model:**

- Chips realize FSMs
- Wires are rectilinear.
- Wires have bounded width and separation λ .
- Gates can be binary or non-binary.
- Gates/memory cells occupy area proportional to λ^2 .
- I/O pads also occupy area proportional to λ^2 .
- Wires on at most $v \geq 1$ levels. Gates on one level.
- Time for signal to travel wire of length l is constant.
- (Time for *diffusion model* is proportional to l^2 !)



The VLSI Model

- **Performance Measures:**

- Chip area A - manufacturing cost increases with A
- Number of steps T - reflects computational cost



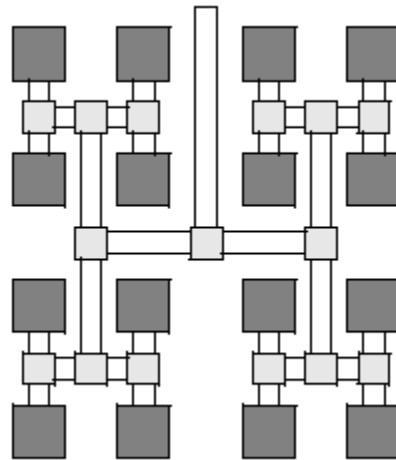
The VLSI Model

- **Functional Model:** Finite functions computed $f : B^n \rightarrow B^m$ where B is any set, usually $\{0,1\}$.
- **Algorithmic Model:** Limitations on I/O
 - Inputs provided at times and places on the chip that are data-independent.
 - Each input is supplied once (*semilective alg.*) or provided multiple times (*multilective alg.*).
 - Outputs are produced once at times and places on the chip that are data-independent.



Chip Layout

- Trees are very important. The H-tree uses area well. Leaves & interior nodes are square



- Let H-tree have 4^k leaves.



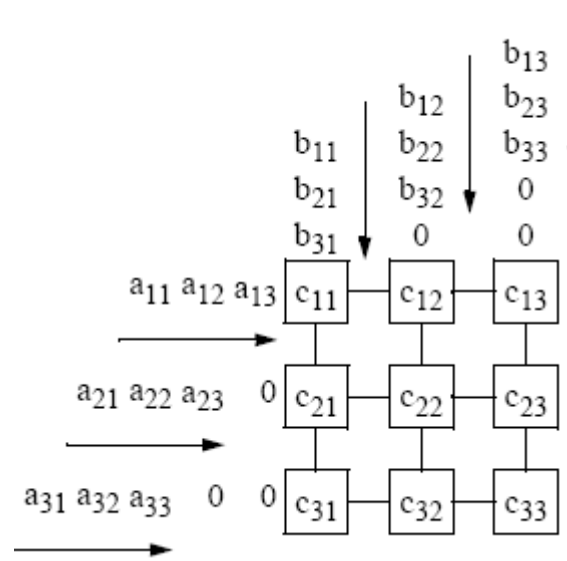
H-Tree Layout

- Let H-tree leaves have area b and interior nodes have area c .
- Let $S(k)$ = length of H-tree side with 4^k leaves
 - $S(1) = 2b + 1$. H-tree has 4 leaves
 - $S(k) = 2S(k-1) + c = (b+c)2^k - c$
- Area of H-tree with $n = 4^k$ leaves, $A(n) = S^2(k)$
 - $A(n) \leq n(b+c)^2$



VLSI Architectures

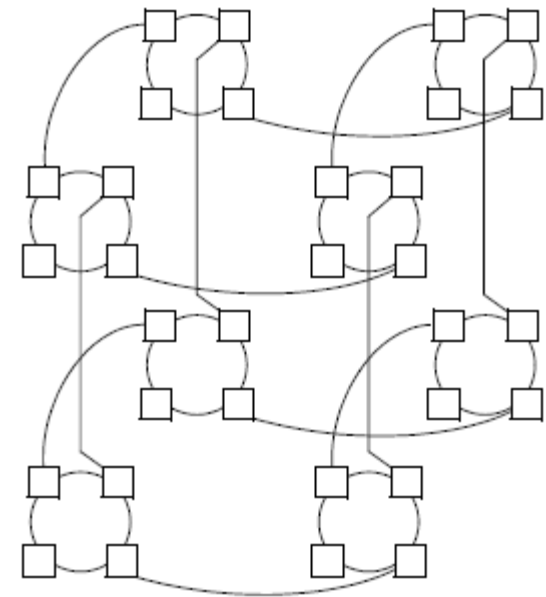
- Tree-based computations
 - Matrix-vector multiplication
 - Prefix computations
- Meshes (see textbook for other examples)
 - Matrix multiplication
 - Bubble sort
- Systolic array



CCC Network



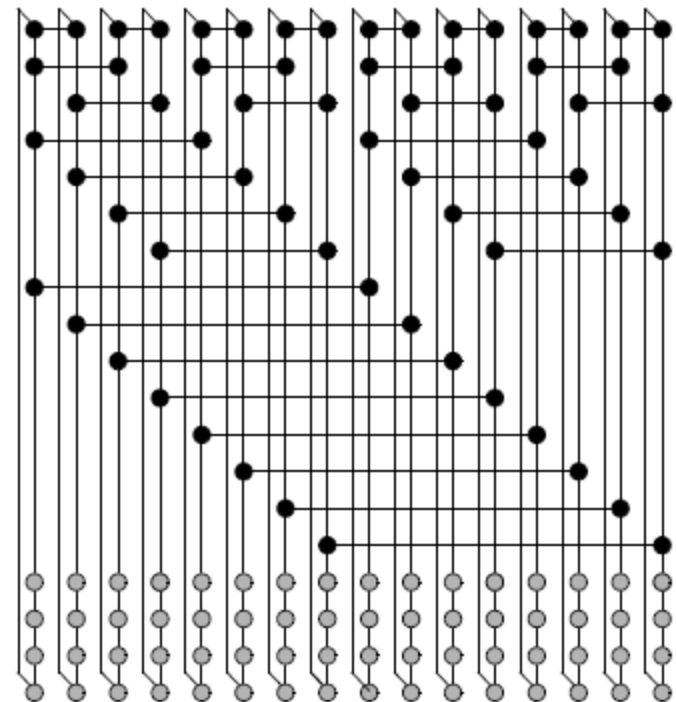
- The (k, d) -CCC network (it has 2^d cycles, 2^k vertices per cycle) shown for $k = 2$, $d = 3$.
- The j th vertex on each cycle connected to j th vertex on another cycle.





CCC Network

- Layout of the (k, d) -CCC network with 2^d cycles and 2^k vertices per cycle shown at right with $k = 3$, $d = 4$.

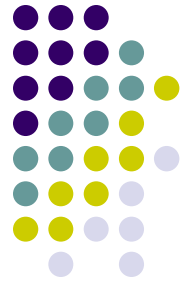




CCC Layout

- The CCC network has $n = 2^{d+k}$ vertices, $2^k \geq d$, and 2^d columns. Index columns from left by binary tuples starting with $(000\dots 0)$. First (k th) dimension connections are made using one (2^{k-1}) row(s). Then, $R = 1 + 2 + \dots + 2^{d-1} = 2^d - 1$ rows are used for connections. $2^k - d$ more rows are used for the $2^k - d$ extra processors on each cycle. Thus, $R \leq 2^d + 2^k - d$ and area $A = R2^d \leq 2^d(2^d + 2^k - d)$. $T = O(d + 2^k)$ steps for normal algorithm to move data across d dimensions and within cycle of length 2^k . When $2^k \approx d$, following theorem holds.

CCC Layout



Theorem Every fully normal algorithm for a n -processor hypercube can be implemented on a VLSI CCC network with area A and using time T satisfying the following bound when $\Omega(\log n) \leq T = O(\sqrt{n})$.

$$AT^2 = O(n^2).$$