ESE534: Computer Organization

Day 18: March 26, 2012
Interconnect 5: Meshes (and MoT)

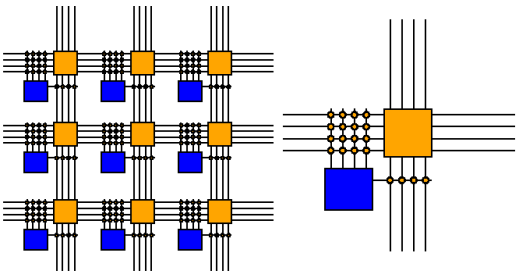
Previously

- Saw
  - need to exploit locality/structure in interconnect
  - a mesh might be useful
  - Rent's Rule as a way to characterize structure

Today

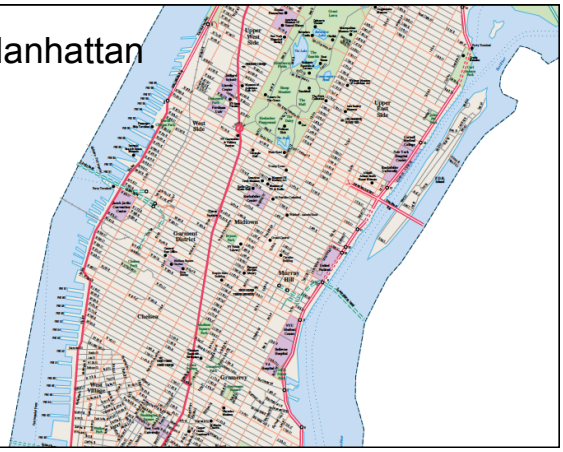
- Mesh:
  - Channel width bounds
  - Linear population
  - Switch requirements
  - Routability
  - Segmentation
  - Clusters
  - Directional Wires
- Mesh-of-Trees (time permitting)

Mesh

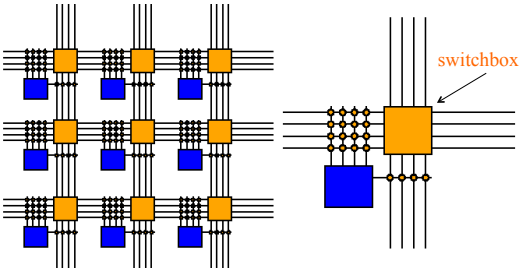


Street Analogy

Manhattan

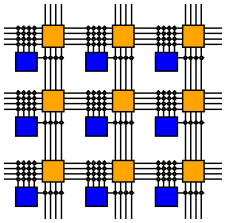


Mesh



Mesh

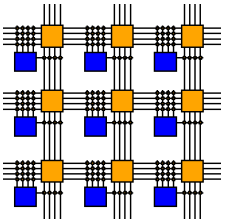
- Strengths?



Mesh Channels

- Lower Bound on w?
- Bisection Bandwidth
  - BW ∝ N^p
  - channels in bisection $=N^{0.5}$

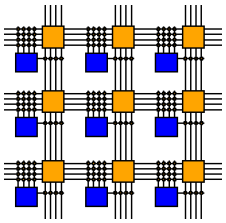
$$W \propto N^p / \sqrt{N} = N^{(p-0.5)}$$



Channel width grows with N.

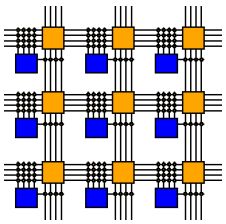
Straight-forward Switching Requirements

- Total Switches?
- Switching Delay?



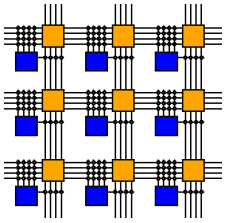
Switch Delay

- Switching Delay:
  - Manhattan distance
    - $|X_i-X_j|+|Y_i-Y_j|$
  - $2\sqrt{(N_{subarray})}$
    - worst case:
    - $N_{subarray} = N$



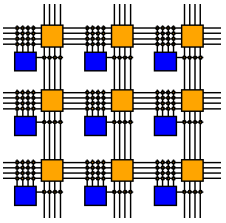
Total Switches

- Switches per switchbox:
  - $4\times(3w\times w)/2 = 6w^2$
  - Bidirectional switches
    - (N→W same as W→N)
    - double count



Total Switches

- Switches per switchbox:
  - $6w^2$
- Switches into network:
  - (K+1) w
- Switches per PE:
  - $6w^2$ +(K+1) w
  - $w = cN^{p-0.5}$
  - Total ∝ $w^2 \propto N^{2p-1}$
- Total Switches: N×(Sw/PE) ∝ N^{2p}



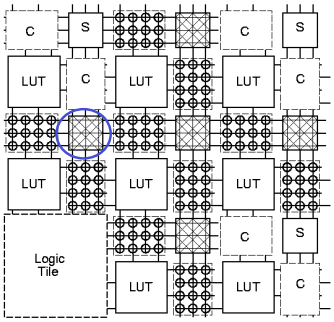
Routability?

- Asking if you can route in a given channel width is:
  - NP-complete
- Contrast with Beneŝ, Beneš-crossover tree….

Linear Population Switchbox

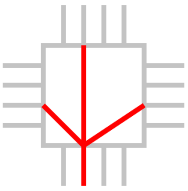
Traditional Mesh Population: Linear

- Switchbox contains only a linear number of switches in channel width



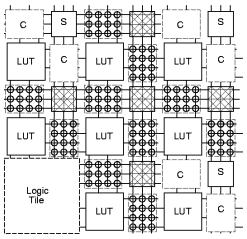
Linear Mesh Switchbox

- Each entering channel connect to:
  - One channel on each remaining side (3)
  - 4 sides
  - W wires
  - Bidirectional switches
    - (N→W same as W→N)
    - double count
  - 3×4×W/2=6W switches
    - vs. $6w^2$ for full population



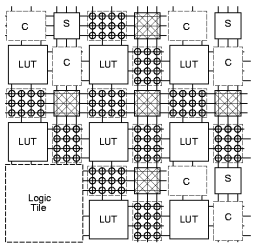
Total Switches

- Switches per switchbox:
  - 6w
- Switches into network:
  - (K+1) w
- Switches per PE:
  - 6w +(K+1) w
  - $w = cN^{p-0.5}$
  - Total $\propto N^{p-0.5}$
- Total Switches: $N \times (Sw/PE) \propto N^{p+0.5} > N$



Total Switches (linear population)

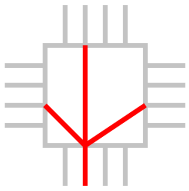
- Total Switches
  - $\propto N^{p+0.5}$
  - $N < N^{p+0.5} < N^{2p}$
- Switches grow faster than nodes
- Wires grow faster than switches



Checking Constants (Preclass 3)

When do linear population designs become wire dominated?

- Wire pitch = 4 F
- switch area = 625 F^2
- wire area: $(4w)^2$
- switch area: 6×625 w
- Crossover?

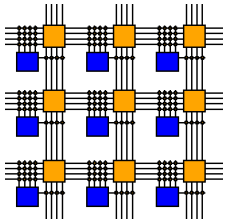


Checking Constants: Full Population

Does full population really use all the wire physical tracks?

- Wire pitch = 4F
- switch area = 625 F^2
- wire area: $(4w)^2$
- switch area: 6×625 w^2
- effective wire pitch: 60F

~15 times pitch

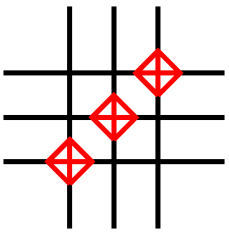


Practical

- Full population is **always** switch dominated
  - doesn't really use all the potential physical tracks
  - …even with only two metal layers
- Just showed:
  - would take 15× Mapping Ratio for linear population to take same area as full population (once crossover to wire dominated)
- Can afford to not use some wires perfectly
  - to reduce switches (area)

Diamond Switch

- Typical linear switchbox pattern:
  - Used by Xilinx



Mapping Ratio?

- How bad is it?
- How much wider do channels have to be?

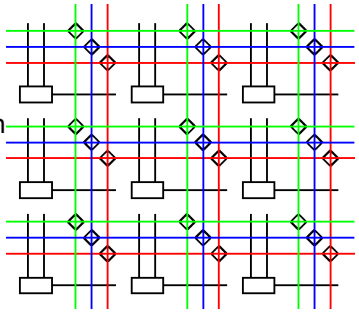
- Mapping Ratio:
  - detail channel width required / global ch width

Mapping Ratio

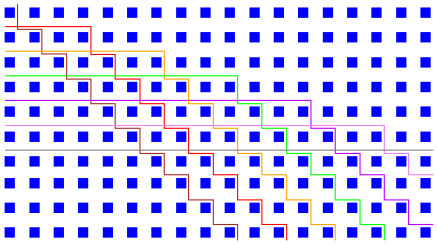
- Empirical:
  - Seems plausibly, constant in practice
- Theory/provable:
  - There is no Constant Mapping Ratio
    - At least detail/global
  - can be arbitrarily large!

Domain Structure

- Once enter network (choose color) can only switch within domain



Detail Routing as Coloring



- Global Route channel width = 2
- Detail Route channel width = N
  - Can make arbitrarily large difference

Routability

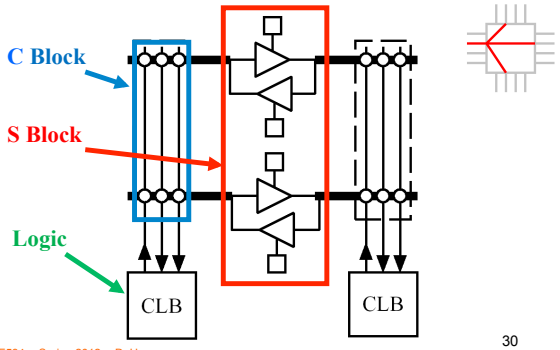
- Domain Routing is NP-Complete
  - can reduce coloring problem to domain selection
    - *i.e.* map adjacent nodes to same channel
    - Previous example shows basic shape
  - (another reason routers are slow)

Routing

- Lack of detail/global mapping ratio
  - Says detail can be arbitrarily worse than global
  - Doesn't necessarily say domain routing is bad
    - Maybe can avoid this effect by changing global route path?
  - Says global not necessarily **predict** detail
  - Argument against decomposing mesh routing into global phase and detail phase
    - Modern FPGA routers do not
    - VLSI routers and earliest FPGA routers did

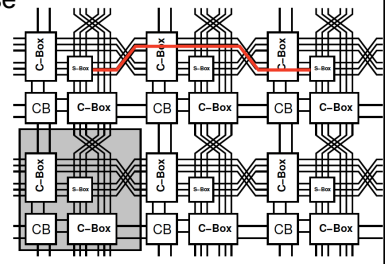
Buffering and Segmentation

Buffered Bidirectional Wires



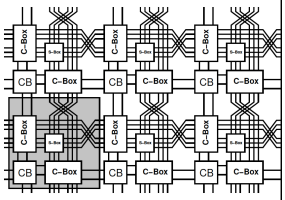
Segmentation

- To improve speed (decrease delay)
- Allow wires to bypass switchboxes
- Maybe save switches?
- Certainly cost more wire tracks



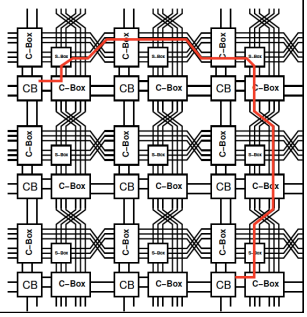
Segmentation

- Segment of Length L_{seg}
  - 6 switches per switchbox visited
  - Only enters a switchbox every L_{seg}
  - SW/sbox/track of length Lseg = $6/L_{seg}$



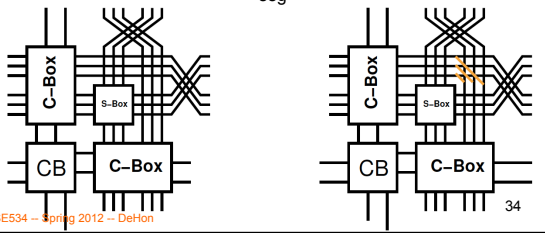
Segmentation

- Reduces switches on path $\sqrt{N/L_{seg}}$
- May get fragmentation
- Another cause of unusable wires

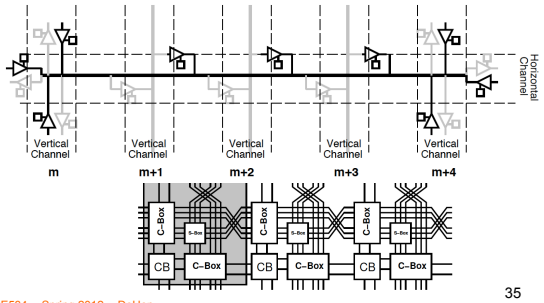


Segmentation: Corner Turn Option

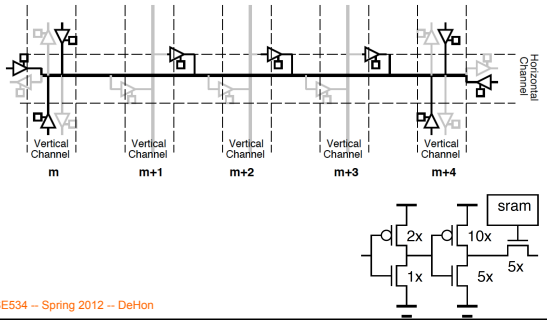
- Can you corner turn in the middle of a segment?
- If can, need one more switch
- SW/sbox/track = $5/L_{seg}$ + 1



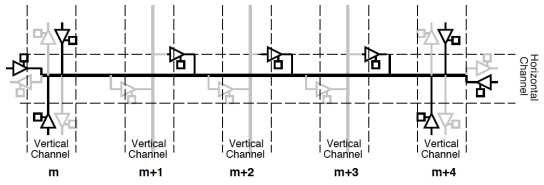
Buffered Switch Composition



Buffered Switch Composition

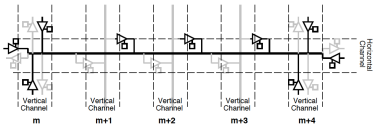


Delay of Segment



$$T_{seg} = T_{sw} + \left(L_{seg}\right)^2 \times R_{seg} \times C_{seg}$$

Segment R and C



$$T_{seg} = T_{sw} + \left(L_{seg}\right)^2 \times R_{seg} \times C_{seg}$$

- What contributes to?
- R_{seg}?
- C_{seg}?

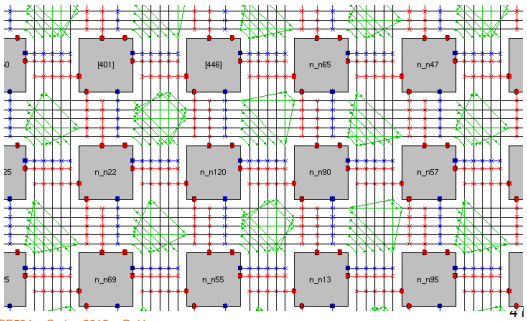
Preclass 4

- Fillin Tseg table together.

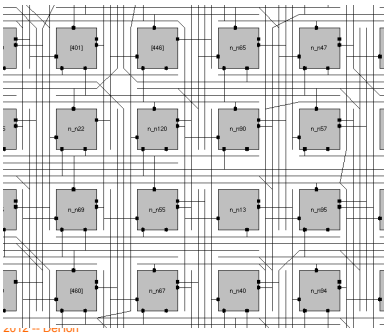
Preclass 4

- What L_{seg} minimizes delay for:
- Distance=1?
- Distance=2?
- Distance=6?
- Distance=10?
- Distance=20?

VPR L_{seg}=4 Pix



VPR L_{seg}=4 Route



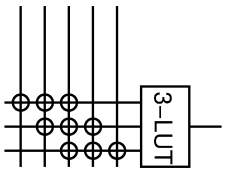
Effect of Segment Length?

- Experiment with on HW9

Connection Boxes

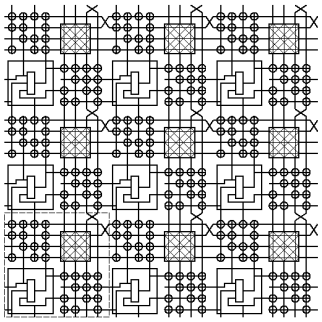
C-Box Depopulation

- Not necessary for every input to connect to every channel
- Saw last time:
  - $K \times (N-K+1)$ switches
- Maybe use fewer?

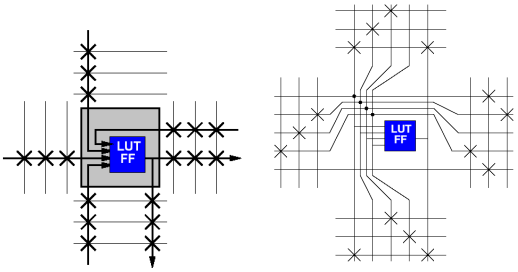


IO Population

- Toronto Model
  - Fc fraction of tracks which an input connects to
- IOs spread over 4 sides
- Maybe show up on multiple
  - Shown here: 2



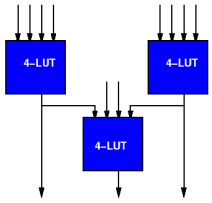
IO Population



Clustering

Leaves Not LUTs

- Recall cascaded LUTs
- Often group collection of LUTs into a Logic Block



Logic Block

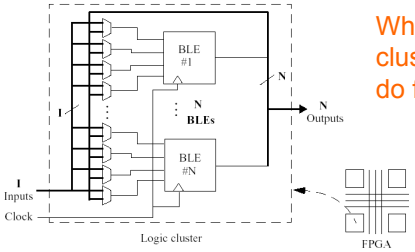
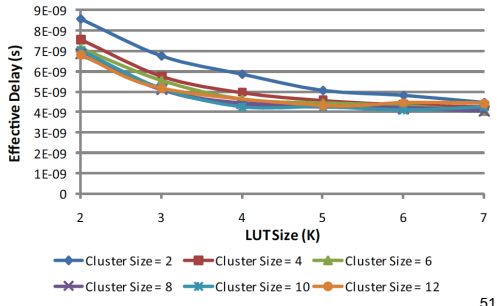


Figure 3: Logic cluster structure.

What does clustering do for delay?

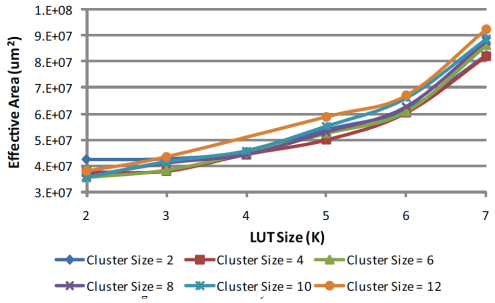
[Betz+Rose/IEEE D&T 1998]

Delay versus Cluster Size



[Lu et al., FPGA 2009]

Area versus Cluster Size



[Lu et al., FPGA 2009]

Review: Mesh Design Parameters

- Cluster Size
  - Internal organization
- LB IO (Fc, sides)
- Switchbox Population and Topology
- Segment length distribution
  - and staggering
- Switch rebuffering

Directional Drive

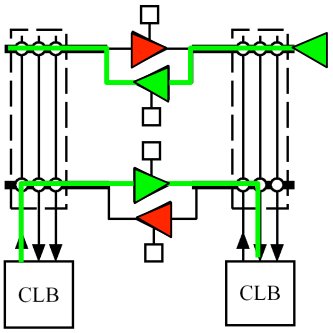
Slides and Study
from Guy Lemieux
(Paper FPT2004 – Tutorial FPT 2009)

Bidirectional Wires

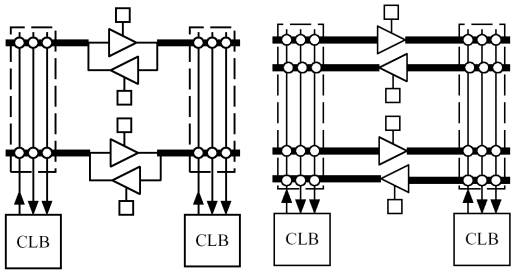
Problem

Half of tristate buffers **left unused**

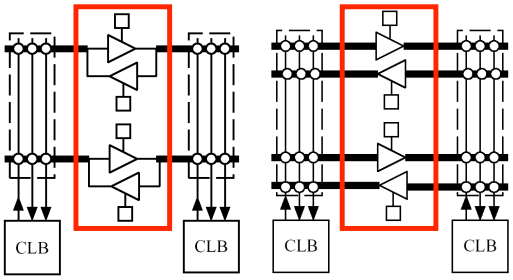
Buffers + input muxes **dominate** interconnect area



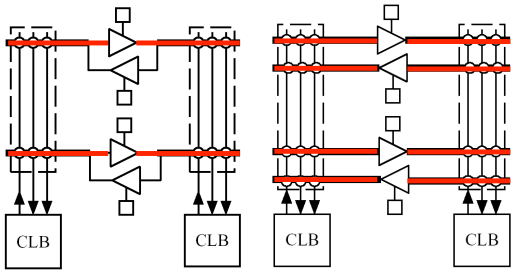
Bidirectional vs Directional



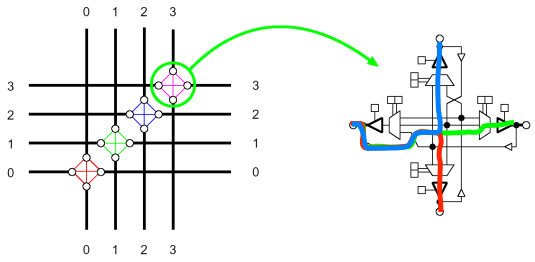
Bidirectional vs Directional



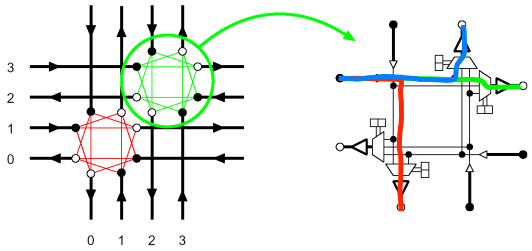
Bidirectional vs Directional



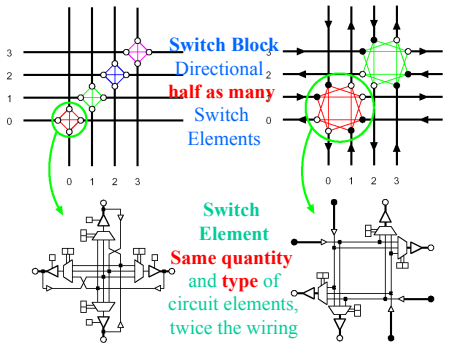
Bidirectional Switch Block



Directional Switch Block

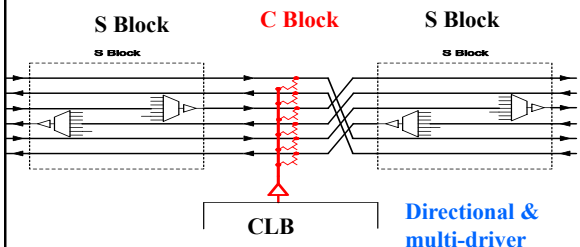


Bidirectional vs Directional



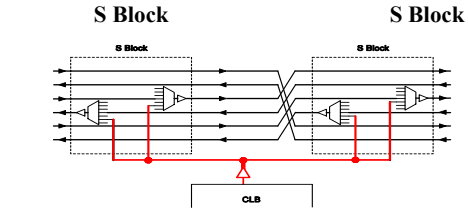
Multi-driver Wiring

Logic outputs use tristate buffers (C Block)



Single-driver Wiring

Logic outputs use muxes (S Block)



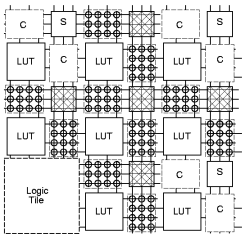
Directional, Single-driver Benefits

- Average improvements
  - 0% channel width (most surprising?)
  - 9% delay
  - 14% tile length of physical layout
  - 25% transistor count
  - 32% area-delay product
  - 37% wiring capacitance
- Any reason to use bidirectional?

MoT

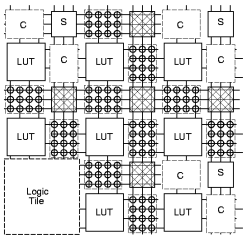
Recall: Mesh Switches

- Switches per switchbox:
  - $6w/L_{seg}$
- Switches into network:
  - $(K+1)\ w$
- Switches per PE:
  - $6w/L_{seg} + Fc \times (K+1)\ w$
  - $w = cN^{p-0.5}$
  - Total $\propto N^{p-0.5}$
- Total Switches: $N*(Sw/PE) \propto N^{p+0.5} > N$



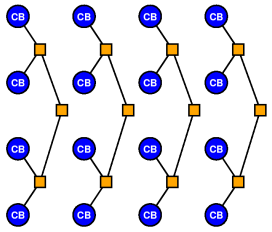
Recall: Mesh Switches

- Switches per PE:
  - $6w/L_{seg} + Fc \times (K+1)\ w$
  - $w = cN^{p-0.5}$
  - Total $\propto N^{p-0.5}$
- Not change for
  - Any constant Fc
  - Any constant L_{seg}



Mesh of Trees

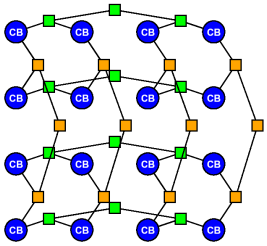
- Hierarchical Mesh
- Build Tree in each column



[Leighton/FOCS 1981]

Mesh of Trees

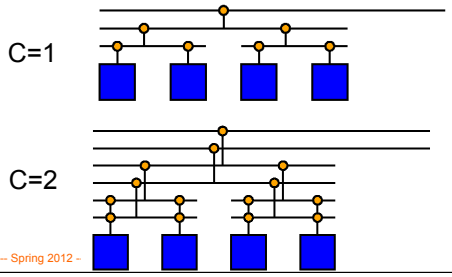
- Hierarchical Mesh
- Build Tree in each column
- …and each row



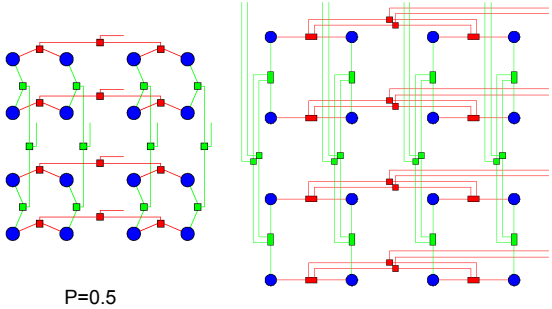
[Leighton/FOCS 1981]

MoT Parameterization

- Support C with additional trees
  - (like BFT)



MoT Parameterization: P

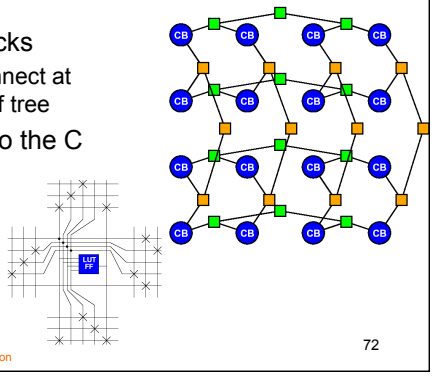


P=0.5

P=0.75

Mesh of Trees

- Logic Blocks
  - Only connect at leaves of tree
- Connect to the C trees
  - Per side
  - 4C total
- C<W

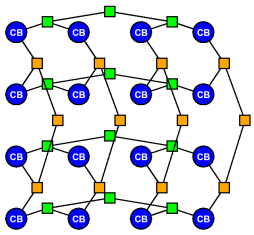


Switches

- Total Tree switches
  - 2 C √N× (switches/ tree)
- Sw/Tree:

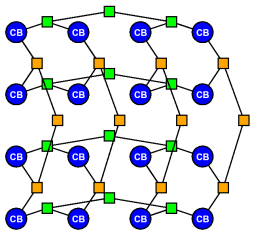
$$\left(\frac{\sqrt{N}}{2}\right)\times\left(\frac{1}{1-2^{p-1.5}}\right)$$

$$TreeSwitches = \left(\frac{C\times N}{1-2^{p-1.5}}\right) = O(N)$$



Switches

- Only connect to leaves of tree
- Leaf switches: C×(K+1)
- Total switches
  - Leaf + Tree
  - O(N)
  - Compare Mesh $O(N^{p+0.5})$



Empirical Results

- **Benchmark:** Toronto 20
- Compare to L_{seg}=1, L_{seg}=4
  - CLMA ~ 8K LUTs
    - Mesh(L_{seg}=4): w=14 → 122 switches/LB
    - MoT(p=0.67,arity=2): C=4 → 89 switches/LB
  - Benchmark wide: 10% less
    - CLMA largest
    - Asymptotic advantage

[Rubin,DeHon/FPGA2003]

MoT Parameters

- C, P
- Arity
- Staggering
- Upper-Level Corner Turns
- Leaf IO Population/topology

Overall 26% fewer than mesh

TABLE V
TOTAL SWITCHES VERSUS ARITY AND RENT EXPONENT (p)

arity	2		3		4			5	8
p	0.67	0.75	0.65	0.81	0.625	0.67	0.75	0.60	0.67
alu4	86	101	88	94	95	88	74	91	90
apex2	106	98	88	95	109	87	91	105	103
apex4	110	129	108	99	113	89	95	123	104
bigkey	62	72	52	71	62	51	54	60	58
clma	103	96	86	99	106	85	93	100	103
des	65	79	69	69	63	71	60	58	61
diffeq	88	77	71	72	64	71	75	61	76
dsip	62	72	70	71	62	68	54	60	58
elliptic	82	93	85	91	91	84	71	88	86
ex1010	107	102	84	90	93	88	98	101	106
ex5p	113	106	108	99	114	90	96	123	107
frisc	103	94	85	91	91	85	89	103	101
misex3	108	100	89	97	95	87	93	106	88
pdc	128	128	118	112	124	124	117	144	136
s298	84	73	70	71	62	69	72	74	73
s38417	84	75	70	77	61	69	76	72	74
s38584.1	84	100	70	77	77	69	76	72	74
seq	107	98	104	93	93	86	91	105	104
spla	123	117	101	91	106	101	106	117	114
tseng	90	80	72	74	65	72	77	62	76
max	128	129	118	112	124	124	117	144	136
sum	1895	1890	1688	1733	1746	1634	1658	1825	1792

[Rubin&DeHon/TRVLSI2004]

Admin

- HW9 out
- Reading for Wednesday on web

Big Ideas [MSB Ideas]

- Mesh natural 2D topology
  - Channels grow as $\Omega(N^{p-0.5})$
  - Wiring grows as $\Omega(N^{2p})$
  - Linear Population:
    - Switches grow as $\Omega(N^{p+0.5})$
      - Worse than shown for hierarchical
    - Unbounded global→detail mapping ratio
    - Detail routing NP-complete
    - But, seems to work well in practice…

Big Ideas [MSB-1 Ideas]

- Segmented/bypass routes
  - can reduce switching delay
  - costs more wires (fragmentation of wires)
- Hierarchy structure allows to save switches
  - O(N) vs. $\Omega(N^{p+0.5})$