

CS184a: Computer Architecture (Structure and Organization)

Day 13: February 4, 2005
Interconnect 1: Requirements



Last Time

- Saw various compute blocks
- To exploit structure in typical designs we need programmable interconnect
- All reasonable, scalable structures:
 - small to moderate sized logic blocks
 - connected via programmable interconnect
- been saying delay across programmable interconnect is a big factor

Today

- Interconnect Design Space
- Dominance of Interconnect
- Interconnect Delay
- Simple things
 - and why they don't work

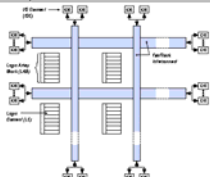
Dominant Area

$$A_{bit_elm} = A_{fixed} + \frac{N_{SW}(N_p, w, p)}{interconnect} \cdot A_{SW} + \left(\frac{c}{w}\right) \cdot n_{bits} \cdot A_{mem_cell} + \frac{d \cdot A_{mem_cell}}{retiming\ memory}$$

Function	Area (λ^2)
LUT MUX + ff	20K (generous, closer to 10K)
Programming Memory	80K (240K typical unencoded)
Interconnect	700K (for $N_p = 2048$)

Dominant Time

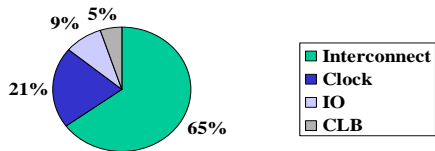
Design	Path	Total Delay	LUT Delay	Inter. %
Altera 10K130V-2	LUT-local-LUT	2.5 ns	2.1 ns	16%
	LUT-row-local-LUT	6.6 ns	2.1 ns	68%
	LUT-column-local-LUT	11.1 ns	2.1 ns	81%
	LUT-row-column-local-LUT	15.6 ns	2.1 ns	87%
	LUT-row-fanout-local-LUT (fanout)	28 ns	2.1 ns	90%



Dominant Time

Design	Path	Total Delay	LUT Delay	Inter. %
DPGA	LUT-LUT (in subarray)	3.5 ns	1.5 ns	60%
	LUT-xbar-LUT	7 ns	1.5 ns	80%
HSRA	LUT-LUT	8 ns	<2 ns	25%
	LUT-cascade	4 ns/4	<2 ns	0%
	LUT-4tree-LUT	8 ns	<2 ns	80%
	LUT-8tree-LUT	12 ns	<2 ns	83%
	LUT-16tree-LUT	16 ns	<2 ns	88%
LUT-64tree-LUT	20 ns	<2 ns	90%	

Dominant Power



XC4003A data from Eric Kusse (UCB MS 1997)

Caltech CS184 Winter2005 -- DeHon

7

For Spatial Architectures

- Interconnect dominant
 - area
 - power
 - time
- ...so need to understand in order to optimize architectures

Caltech CS184 Winter2005 -- DeHon

8

Interconnect

- **Problem**

- Thousands of independent (bit) operators producing results
 - true of FPGAs today
 - ...true for *LIW, multi-uP, etc. in future
- Each taking as inputs the results of other (bit) processing elements
- Interconnect is **late bound**
 - don't know until after fabrication

Caltech CS184 Winter2005 -- DeHon

9

Design Issues

- **Flexibility** -- route "anything"
 - (w/in reason?)
- **Area** -- wires, switches
- **Delay** -- switches in path, stubs, wire length
- **Power** -- switch, wire capacitance
- **Routability** -- computational difficulty finding routes

Caltech CS184 Winter2005 -- DeHon

10

Delay

Caltech CS184 Winter2005 -- DeHon

11

Wiring Delay

- Delay on wire of length L_{seg} :

$$T_{\text{seg}} = T_{\text{gate}} + 0.4 RC$$

- $C = L_{\text{seg}} \times C_{\text{sq}}$

- $R = L_{\text{seg}} \times R_{\text{sq}}$

$$T_{\text{seg}} = T_{\text{gate}} + 0.4 C_{\text{sq}} \times R_{\text{sq}} \times L_{\text{seg}}^2$$

Caltech CS184 Winter2005 -- DeHon

12

Wire Numbers

- $R_{sq} = 0.17 \Omega/\text{sq}$.
 - from ITRS:Interconnect
 - Conductor effective resistance
 - A/R (aspect ratio)
- $C_{sq} = 7 \times 10^{-18} \text{F}/\text{sq}$.
- $R_{sq} \times C_{sq} \approx 10^{-18} \text{s}$
- $T_{gate} = 30 \text{ps}$
- Chip: 7mm side, 70nm sq. (45nm process)
 - 10^5 squares across chip

Caltech CS184 Winter2005 -- DeHon

13

Wiring Delay

- Wire Delay

$$T_{seg} = T_{gate} + 0.4 C_{sq} \times R_{sq} \times L_{seg}^2$$

$$T_{seg} = 30\text{ps} + 0.4 \cdot 10^{-18} \text{s} \times 10^{10}$$

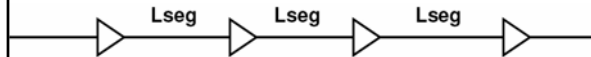
$$T_{seg} = 30\text{ps} + 4\text{ns} \approx 4\text{ns}$$

Caltech CS184 Winter2005 -- DeHon

14

Buffer Wire

- Buffer every L_{seg}
 - $T_{cross} = (L_{cross}/L_{seg}) T_{seg}$
- $$T_{cross} = (L_{cross}/L_{seg}) (T_{gate} + 0.4 C_{sq} \times R_{sq} \times L_{seg}^2)$$
- $$= (L_{cross}) (T_{gate}/L_{seg} + 0.4 C_{sq} \times R_{sq} \times L_{seg})$$



Caltech CS184 Winter2005 -- DeHon

15

Opt. Buffer Wire

- $T_{cross} = (L_{cross}) (T_{gate}/L_{seg} + 0.4 C_{sq} \times R_{sq} \times L_{seg})$
- Minimize:
 - Take $d(T_{cross})/d(L_{seg}) = 0$
 - $0 = (L_{cross}) (-T_{gate}/L_{seg}^2 + 0.4 C_{sq} \times R_{sq})$
 - $T_{gate} = 0.4 C_{sq} \times R_{sq} L_{seg}^2$

Caltech CS184 Winter2005 -- DeHon

16

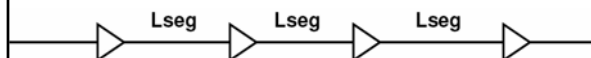
Optimization Point

- Optimized:

$$T_{cross} = (L_{cross}/L_{seg}) (T_{gate} + 0.4 C_{sq} \times R_{sq} \times L_{seg}^2)$$

$$T_{gate} = 0.4 C_{sq} \times R_{sq} L_{seg}^2$$

- Says: equalize gate and wire delay



Caltech CS184 Winter2005 -- DeHon

17

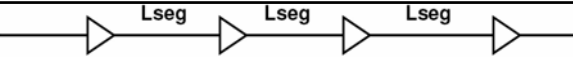
Optimal Segment Length

- $T_{gate} = 0.4 C_{sq} \times R_{sq} L_{seg}^2$
- $L_{seg} = \text{Sqrt}(T_{gate}/0.4 C_{sq} \times R_{sq})$
- $L_{seg} = \text{Sqrt}(30 \cdot 10^{-12} \text{s}/0.4 \cdot 10^{-18} \text{s})$
- $L_{seg} \approx \text{Sqrt}(10^8) \approx 10^4 \text{sq}$.

Caltech CS184 Winter2005 -- DeHon

18

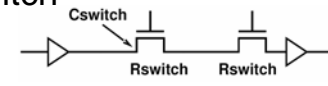
Buffered Delay



- Chip: 7mm side, 70nm sq. (45nm process)
 - 10^5 squares across chip
- $L_{\text{seg}} \approx 10^4$ sq.
- 10 segments:
 - Each of delay $2 T_{\text{gate}}$
 - $T_{\text{cross}} = 20 \times 30\text{ps} = 600\text{ps}$
 - **Compare: 4ns**

Caltech CS184 Winter2005 -- DeHon 19

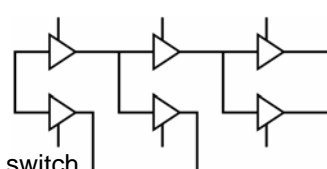
Unbuffered Switch



- $R \sim 600\Omega$ (width ~ 20)
 - About 3600 squares? [$0.17 \Omega/\text{sq.}$]
- $C \sim 5 \times 10^{-16}\text{F}$
 - About 100 squares?
- Not lumped $\sim 2x$ worse
- Together contribute roughly 1200 squares
- Maybe 8 per rebuffer?
 - ...assumes large switch and no wire...

Caltech CS184 Winter2005 -- DeHon 20

Buffered Switch

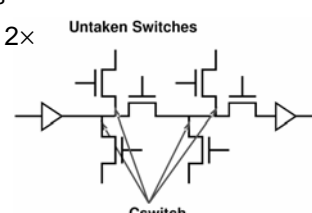


- Pay T_{gate} at each switch
- Slows down relative to
 - Optimally buffered wire
 - Unbuffered switch
- ...when placed too often

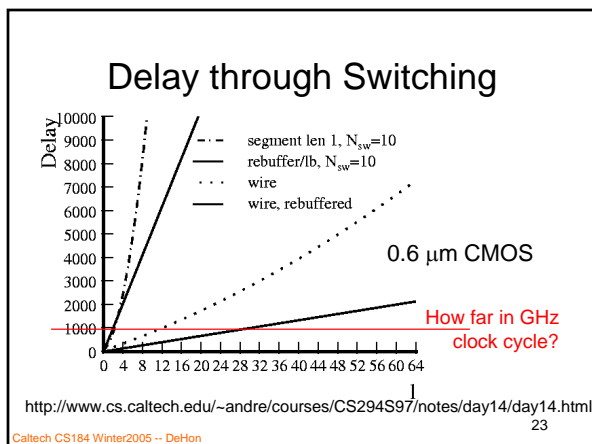
Caltech CS184 Winter2005 -- DeHon 21

Stub Capacitance

- Every untaken switch touching line
- $C \sim 2.5 \times 10^{-16}\text{F}$
 - About 50 squares
- ...and lumped so $2x$



Caltech CS184 Winter2005 -- DeHon 22

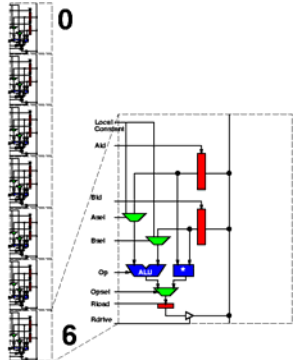


First Attempts

Caltech CS184 Winter2005 -- DeHon 24

(1) Shared Bus

- Familiar case
- Use single interconnect resource
- Reuse in Time
- Consequence?



Caltech CS184 Winter2005 -- DeHon

Shared Bus

- Consider operation: $y = Ax^2 + Bx + C$
 - 3 mpys
 - 2 adds
 - ~5 values need to be routed from producer to consumer
- Performance lower bound if have design w/:
 - m multipliers
 - u madd units
 - a adders
 - i simultaneous interconnection busses

Caltech CS184 Winter2005 -- DeHon

26

Resource Bounded Scheduling

- Scheduling in general NP-hard
 - (find optimum)
 - can approximate in $O(E)$ time

Caltech CS184 Winter2005 -- DeHon

27

Lower Bound: Critical Path

- ASAP schedule ignoring resource constraints
 - (look at length of remaining critical path)
- Certainly cannot finish any faster than that

Caltech CS184 Winter2005 -- DeHon

28

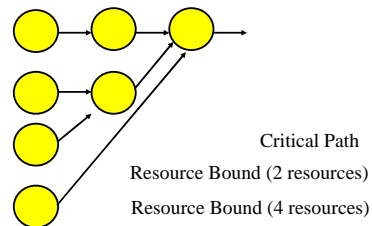
Lower Bound: Resource Capacity

- Sum up all capacity required per resource
- Divide by total resource (for type)
- Lower bound on remaining schedule time
 - (best can do is pack all use densely)

Caltech CS184 Winter2005 -- DeHon

29

Example



Caltech CS184 Winter2005 -- DeHon

30

Example 2

RB = $8/2=4$
 LB = 5
 best delay = 6

Caltech CS184 Winter2005 -- DeHon 31

Shared Bus

- Consider operation: $y = Ax^2 + Bx + C$
 - 3 mpys
 - 2 adds
 - ~5 values need to be routed from producer to consumer
- Performance lower bound if have design w/:
 - m multipliers
 - u madd units
 - a adders
 - i simultaneous interconnection busses

Caltech CS184 Winter2005 -- DeHon 32

Viewpoint

- Interconnect is a resource
- Bottleneck for design can be in availability of **any** resource
- Lower Bound on Delay: Logical Resource / Physical Resources
- May be worse
 - Dependencies (critical path bound)
 - ability to use resource

Caltech CS184 Winter2005 -- DeHon 33

Shared Bus

- Flexibility (+)
 - routes everything (given enough time)
 - can be trick to schedule use optimally
- Area (++)
 - kn switches
 - O(n)
- Delay (Power) (--):
 - wire length O(kn)
 - parasitic stubs: kn+n
 - series switch: 1
 - O(kn)
 - **sequentialize I/B**

Caltech CS184 Winter2005 -- DeHon

Term: Bisection Bandwidth

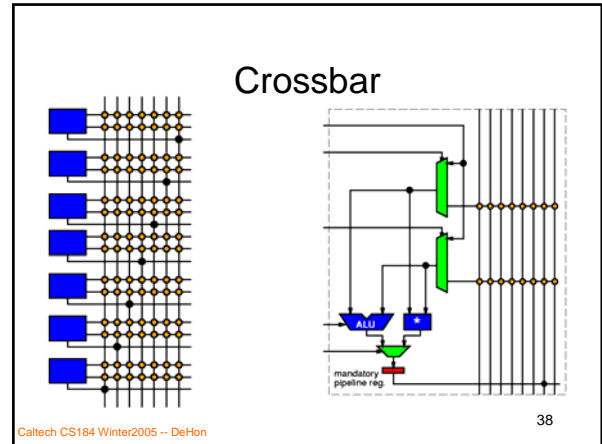
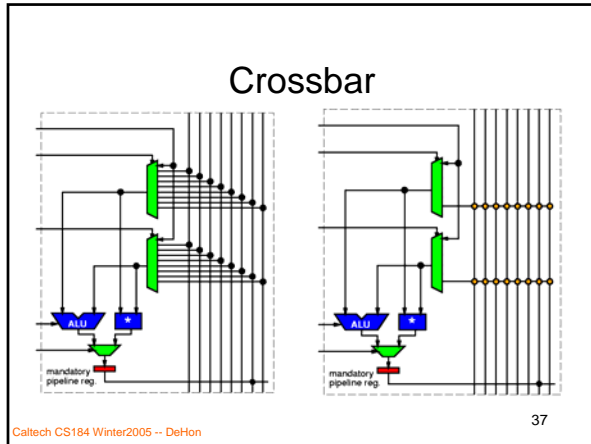
- Partition design into two equal size halves
- Minimize wires (nets) with ends in both halves
- Number of wires crossing is **bisection**

Caltech CS184 Winter2005 -- DeHon 35

(2) Crossbar

- Avoid bottleneck
- Every output gets its own interconnect channel

Caltech CS184 Winter2005 -- DeHon



Crossbar

- Flexibility (++)
 - routes everything (guaranteed)
- Delay (Power) (-)
 - wire length $O(kn)$
 - parasitic stubs: $kn+n$
 - series switch: 1
 - $O(kn)$
- Area (-)
 - Bisection bandwidth n
 - kn^2 switches
 - $O(n^2)$

Caltech CS184 Winter2005 -- DeHon 39

Crossbar

- Better than exponential
- **Too expensive**
 - Switch Area = $k*n^2*2.5K\lambda^2$
 - Switch Area/LUT = $k*n*2.5K\lambda^2$
 - $n=1024, k=4 \rightarrow 10M \lambda^2$
- What can we do?

Caltech CS184 Winter2005 -- DeHon 40

Avoiding Crossbar Costs

- Typical architecture trick:
 - exploit expected problem **structure**
- We have **freedom** in operator placement
- Designs have spatial locality
- \rightarrow place connected components "close" together
 - don't need full interconnect?

Caltech CS184 Winter2005 -- DeHon 41

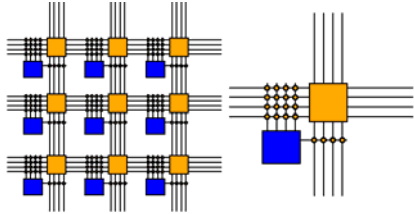
Exploit Locality

- Wires expensive
- Local interconnect cheap
- 1D versions
- What does this do to
 - Switches?
 - Delay?
- (quantify on hwrk)

Caltech CS184 Winter2005 -- DeHon 42

Exploit Locality

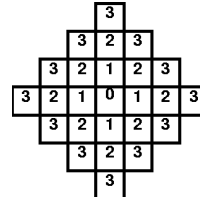
- Wires expensive
- Local interconnect cheap
- Use 2D to make more things closer
- Mesh?



Caltech CS184 Winter2005 -- DeHon

Mesh Analysis

- Can we place everything close?



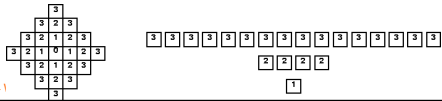
44

Caltech CS184 Winter2005 -- DeHon

Mesh "Closeness"

- Try placing "everything" close

Manhattan Distance	Places	Transitive Fanin
1	4	4
2	8	16
3	12	64
i	i	i
n	$4n$	4^n

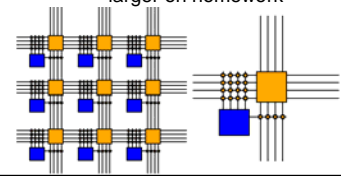


Caltech CS184

45

Mesh Analysis

- Flexibility - ?
 - Ok w/ large w
- Delay (Power)
 - Series switches
 - $1 \rightarrow \sqrt{n}$
 - Wire length
 - $w \rightarrow w\sqrt{n}$
 - Stubs
 - $O(w) \rightarrow O(w\sqrt{n})$
- Area
 - Bisection BW -- $w\sqrt{n}$
 - Switches -- $O(nw)$
 - $O(w^2n)$
 - larger on homework



Caltech CS184 Winter2005 -- DeHon

Mesh

- Plausible
- ...but What's w
- ...and how does it grow?

Caltech CS184 Winter2005 -- DeHon

47

Big Ideas [MSB Ideas]

- Interconnect Dominant
 - power, delay, area
- Can be bottleneck for designs
- Can't afford full crossbar
- Need to exploit locality
- Can't have everything close

Caltech CS184 Winter2005 -- DeHon

48