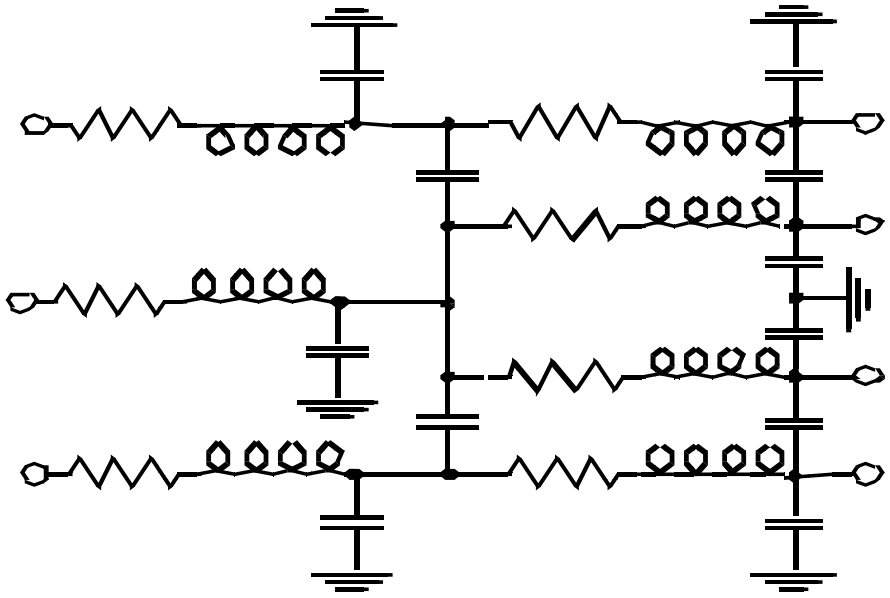
EECS 151/251A
Spring 2018
Digital Design and Integrated Circuits

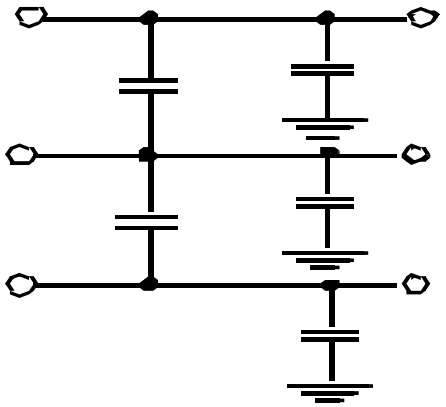
Instructors:
Nick Weaver & John Wawrzynek

Lecture 12

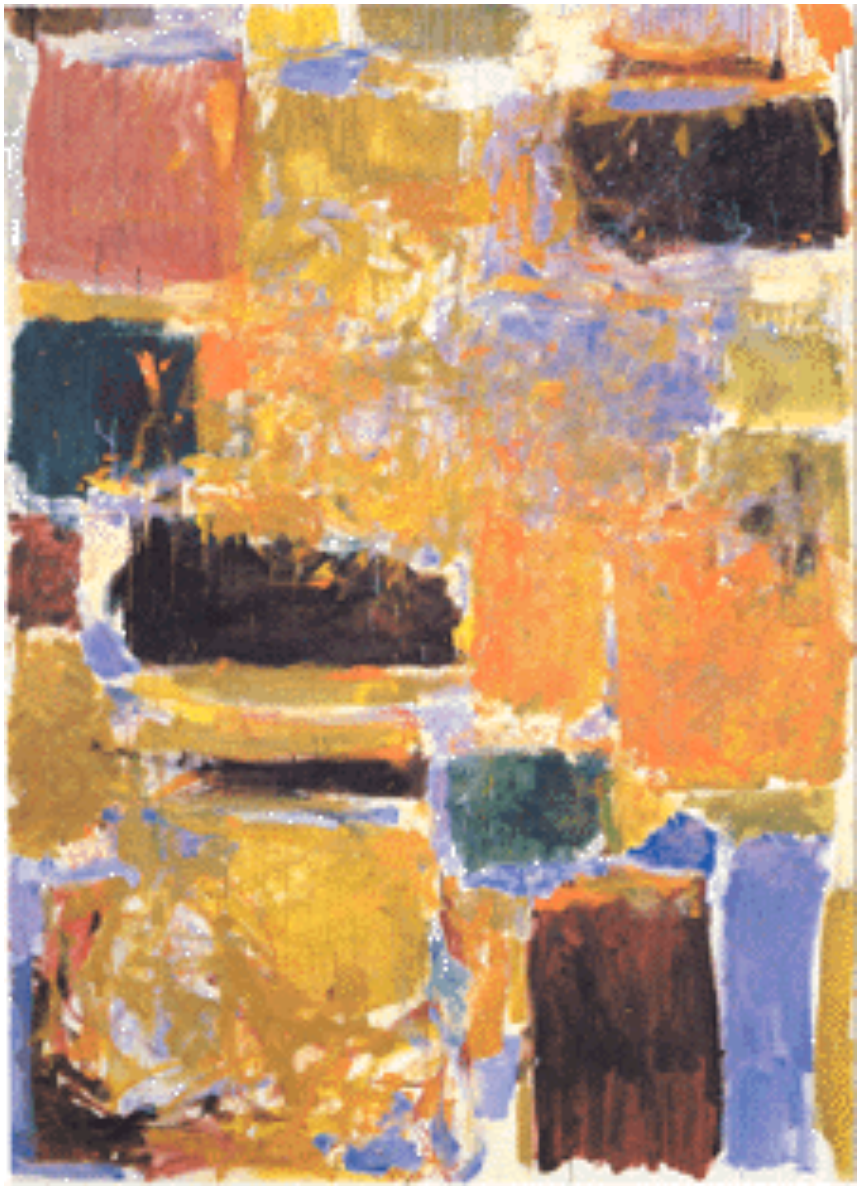
Wire Models



All-inclusive model

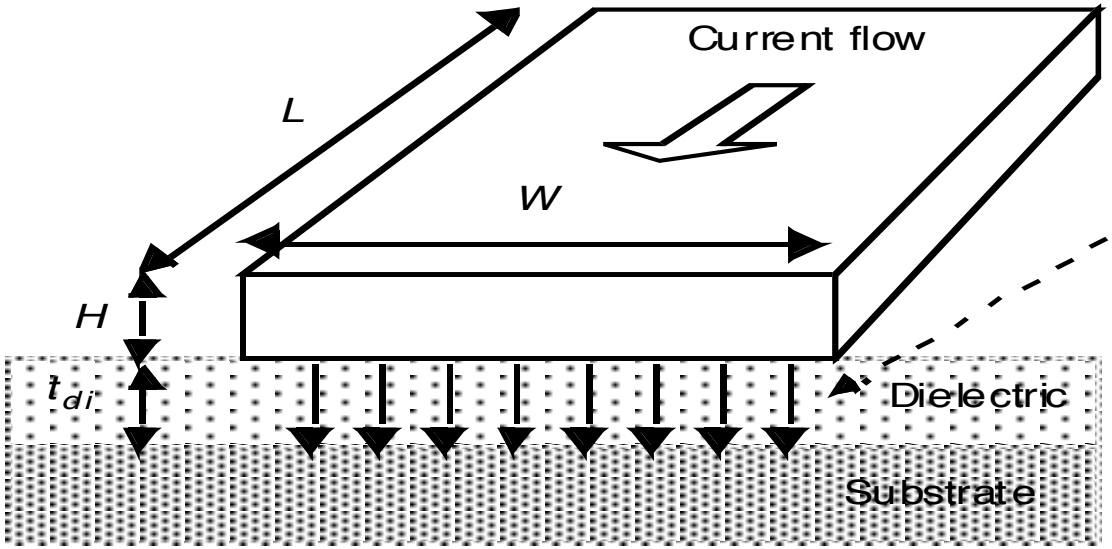


Capacitance-only



Capacitance

Capacitance: The Parallel Plate Model



$$c_{int} = \frac{\varepsilon_{di}}{t_{di}} WL$$

- Cap scaling (s – process scaling factor)
  - Local wires
    - W, L, tdi all decrease (~s)
    - Cap decreases linearly with feature size (~s)
  - Global wires
    - W, tdi decrease (~s), L constant
    - Cap ~constant
- Permittivity is a property of the dielectric: $\varepsilon_{di} = \kappa\varepsilon_0$

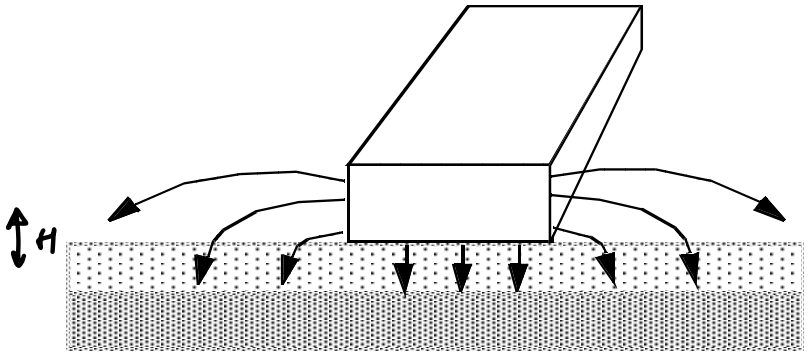
Permittivity/Dialectric Constant

Material	κ (factor of of ε_0)
Free space	1
Aerogels	~1.5
Polyimides (organic)	3-4
Silicon dioxide	3.9
Glass-epoxy (PC board)	5
Silicon Nitride (Si_3N_4)	7.5
Alumina (package)	9.5
Silicon	11.7

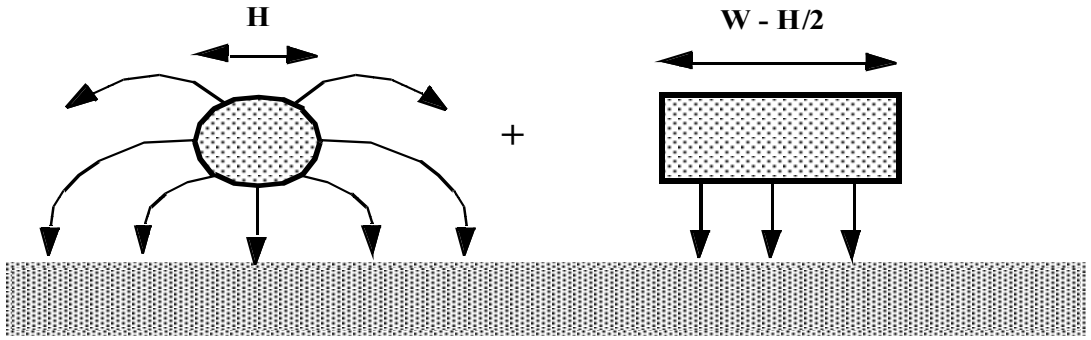
- Low-k dielectrics used sub-130nm
  - Carbon-doped oxide
  - Fabs also looking at air-gaps

Fringing Capacitance

$$c_{wire} = c_{pp} + c_{fringe} = \frac{w\varepsilon_{di}}{t_{di}} + \frac{2\pi\varepsilon_{di}}{\log(t_{di}/H)}$$



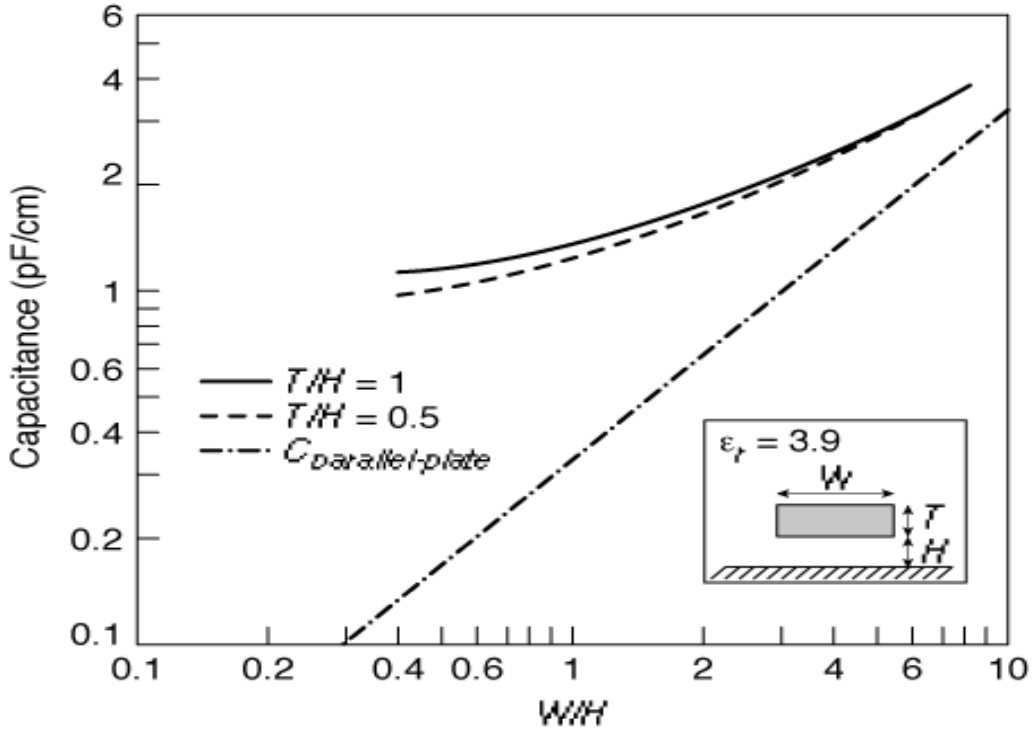
(a)



(b)

- Fringe cap per unit length ~const (good rule of thumb 0.2fF/um)

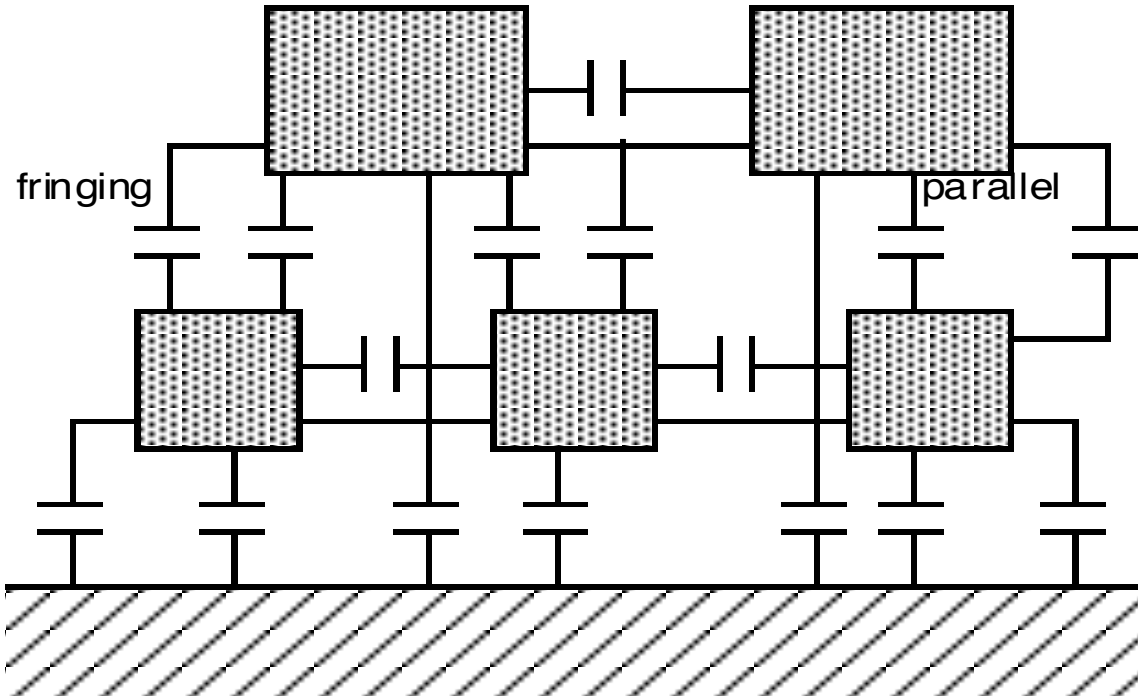
Fringing versus Parallel Plate



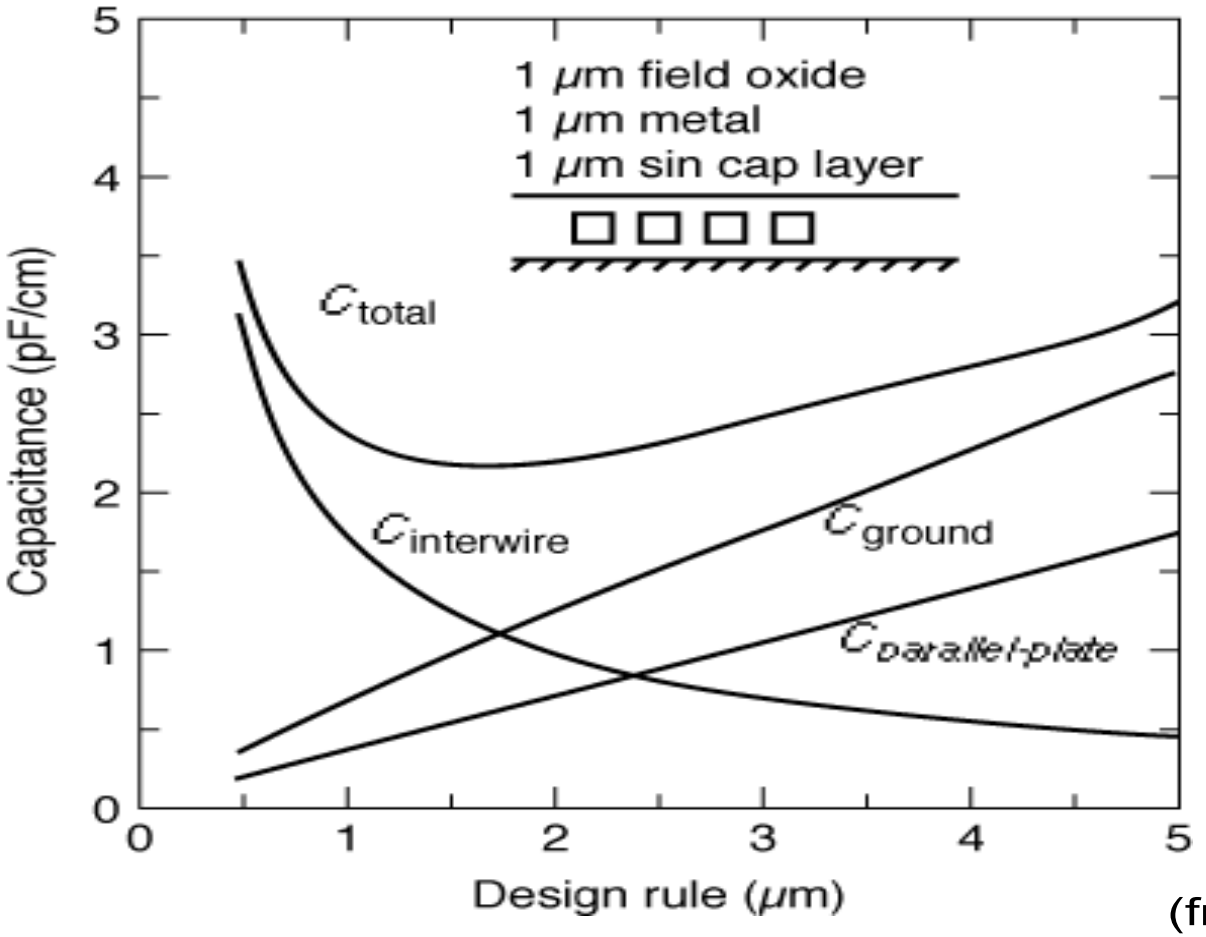
(from [Bakoglu89])

- Narrow, tall wires in modern processes
  - Trying to keep resistance from increasing
  - Comes at the expense of fringe cap

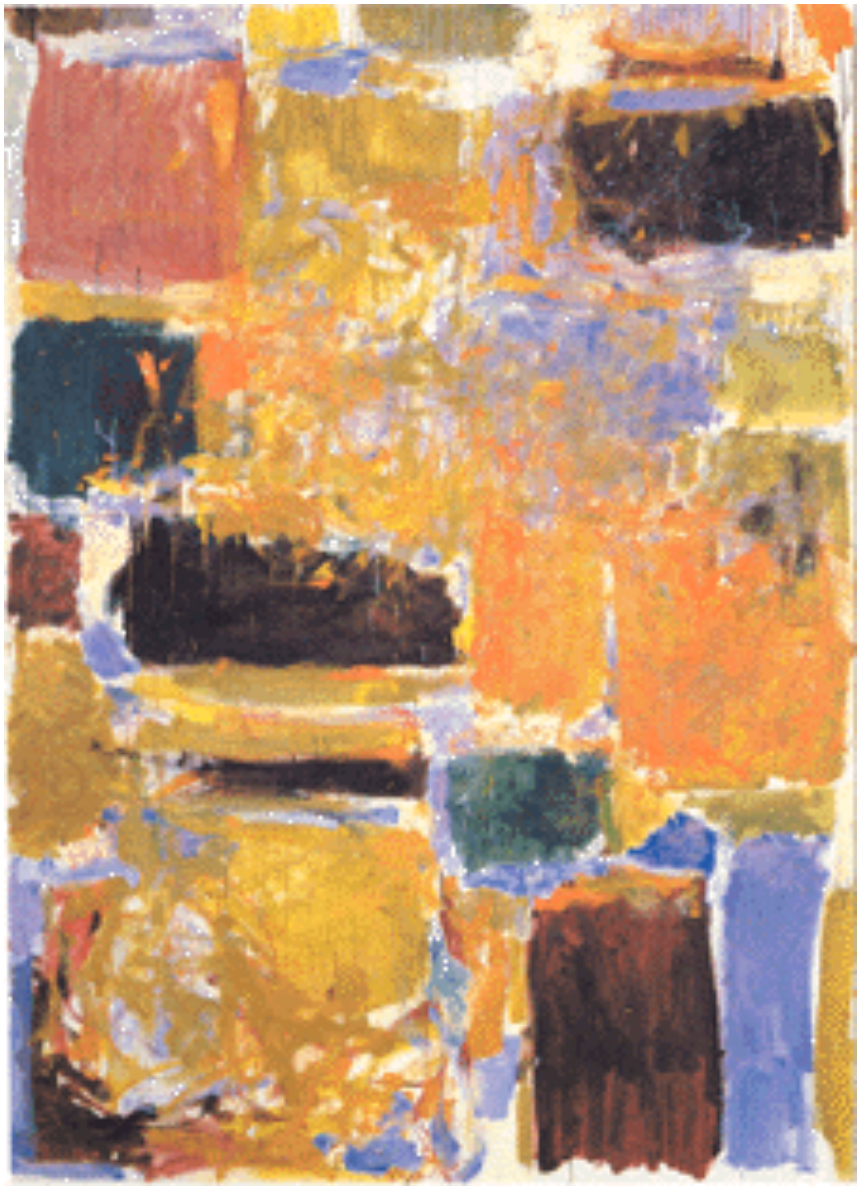
Interwire Capacitance



Impact of Interwire Capacitance

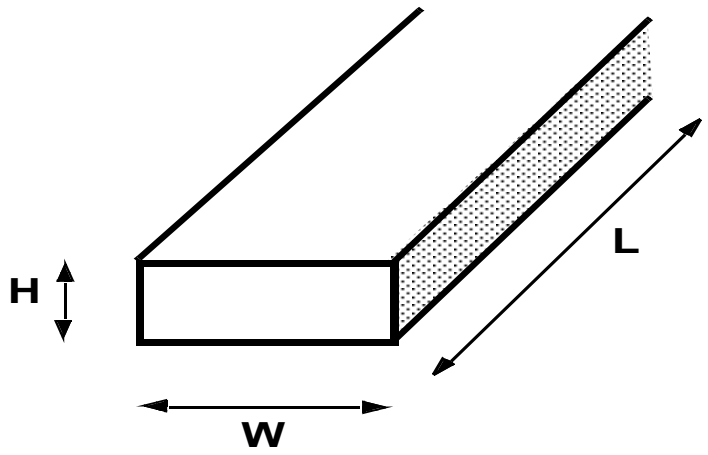


(from [Bakoglu89])



Resistance

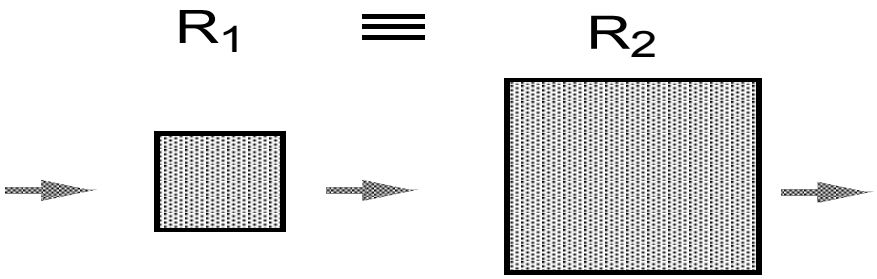
Wire Resistance



$$R = \rho \frac{L}{H W}$$

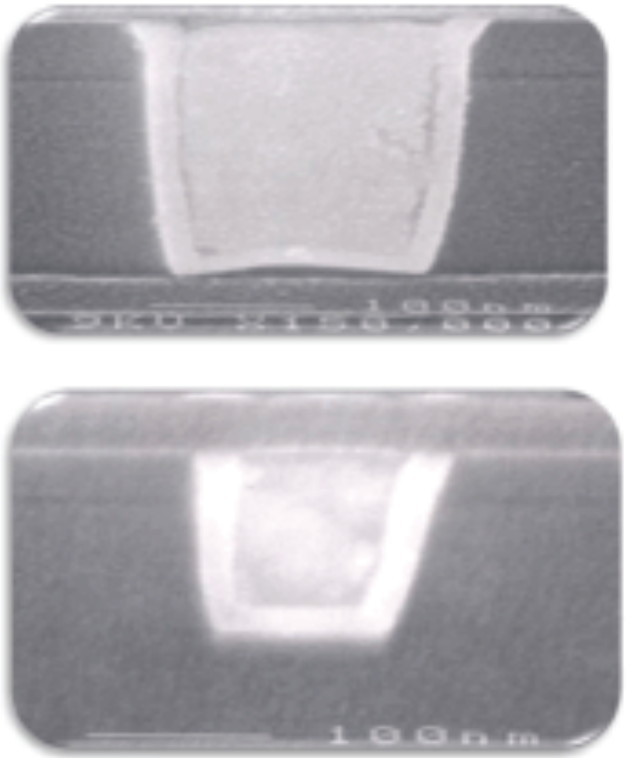
Sheet Resistance
R_o

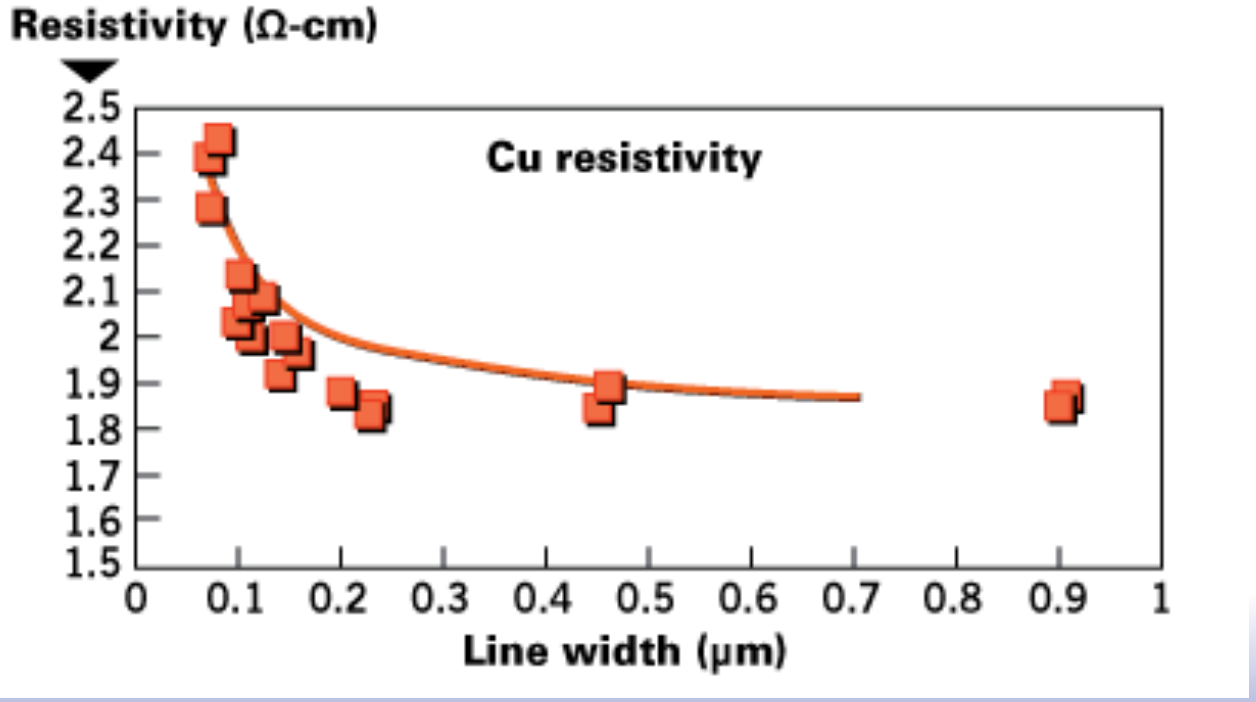
Units Ω-m



Interconnect Resistance

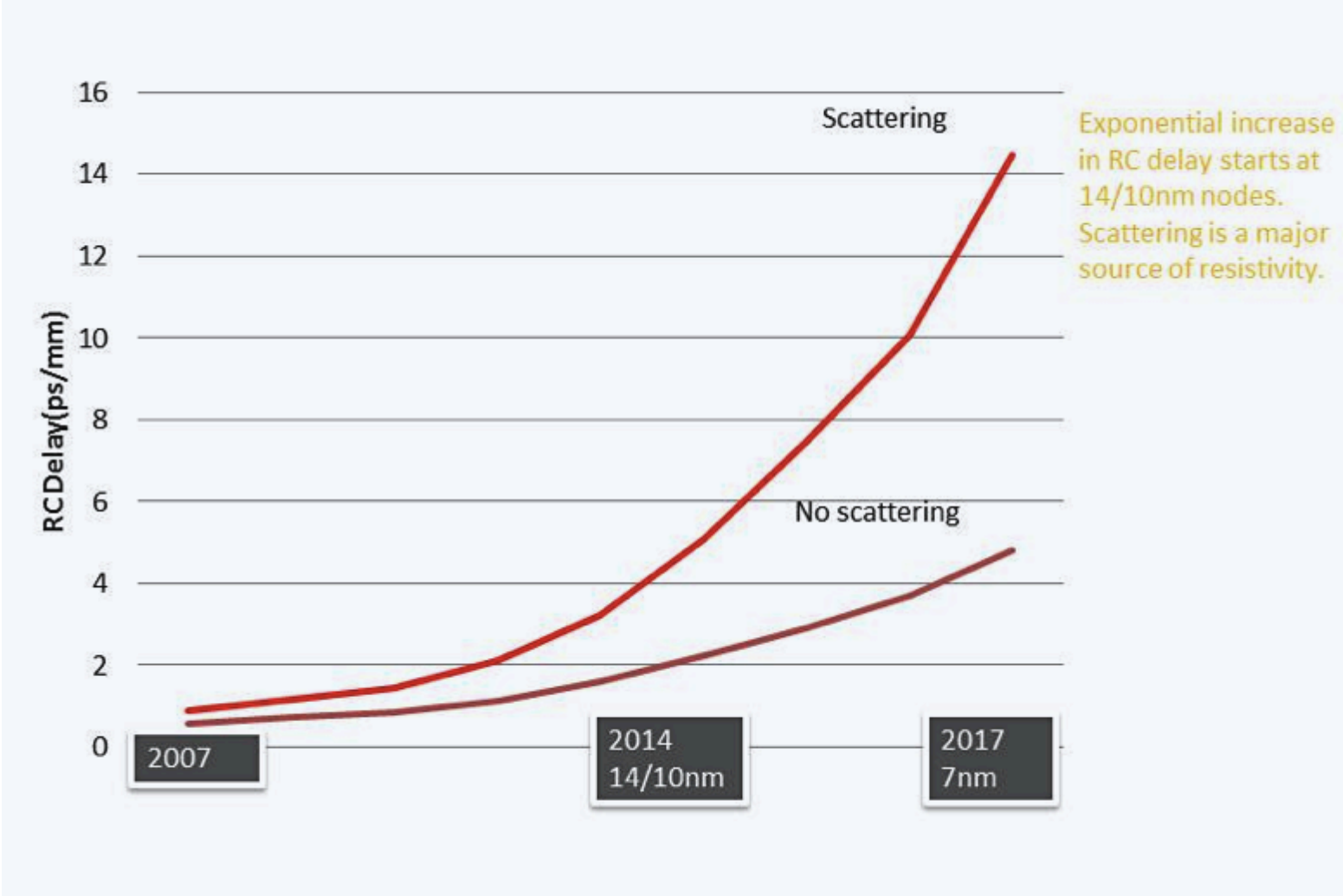
Material	ρ (Ω-m)
Silver (Ag)	1.6×10^{-8}
Copper (Cu)	1.7×10^{-8}
Gold (Au)	2.2×10^{-8}
Aluminum (Al)	2.7×10^{-8}
Tungsten (W)	5.5×10^{-8}



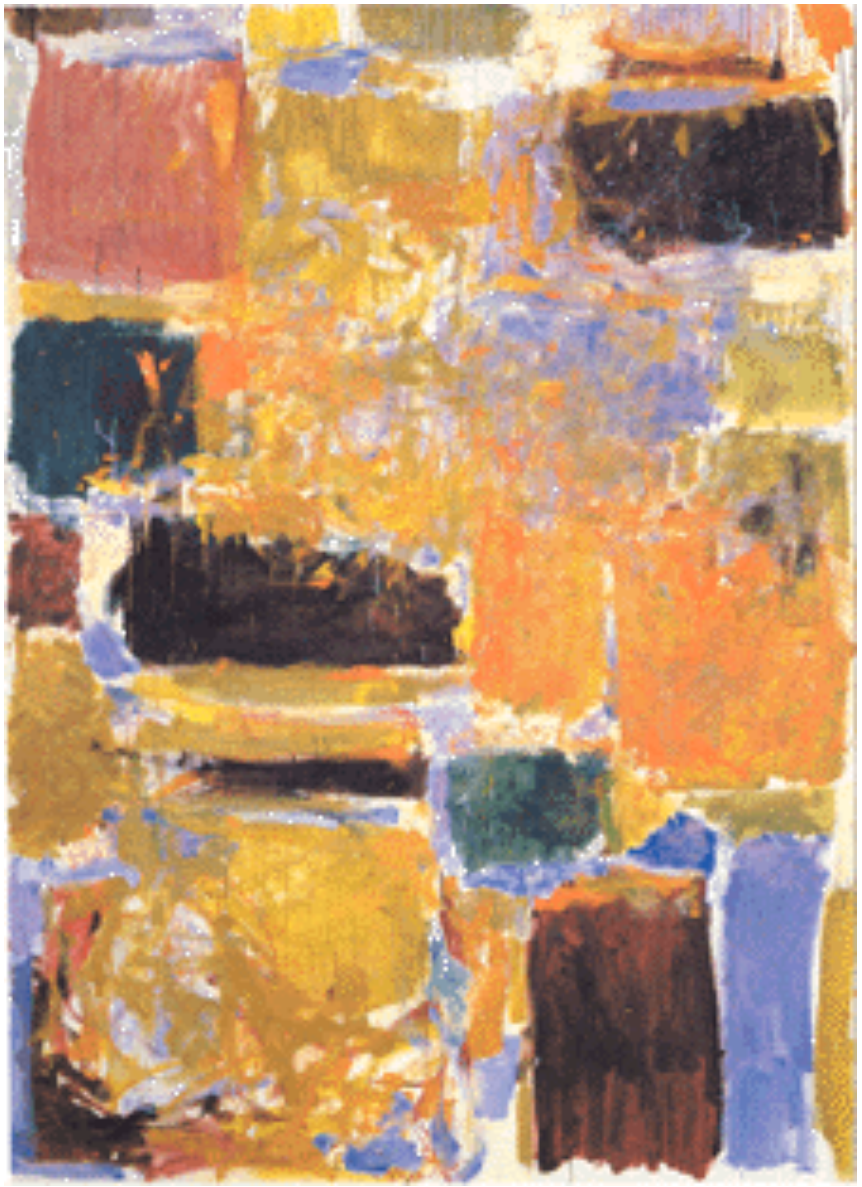


Source: S. Naffziger, AMD, VLSI 2011

Impact on Delay

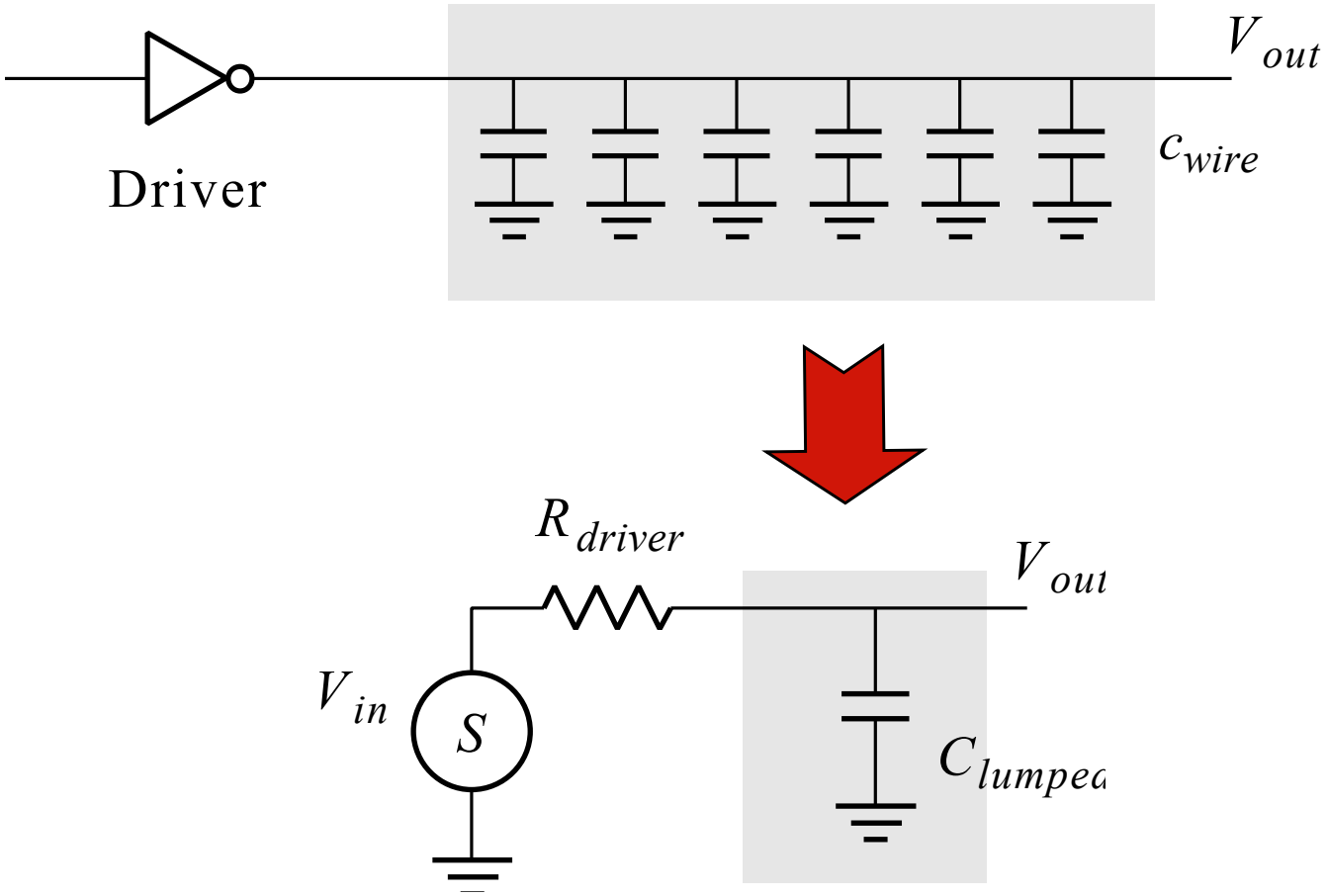


Source: Applied Materials

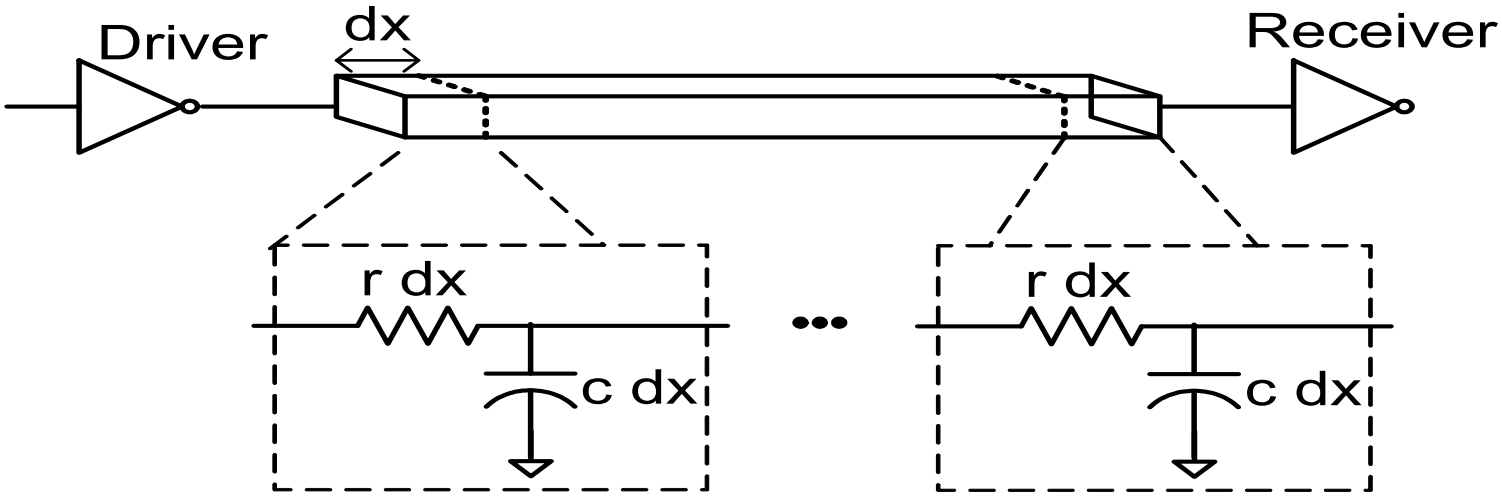


Interconnect Modeling

The Lumped Model

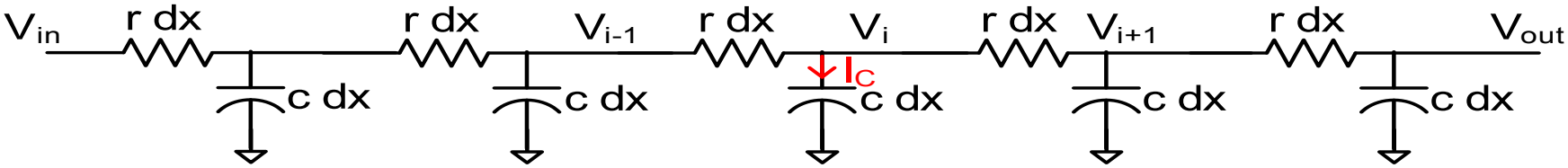


The Distributed RC-line



- Analysis method:
  - Break the wire up into segments of length dx
  - Each segment resistance (r dx)
  - Capacitance (c dx)

The Distributed RC-line

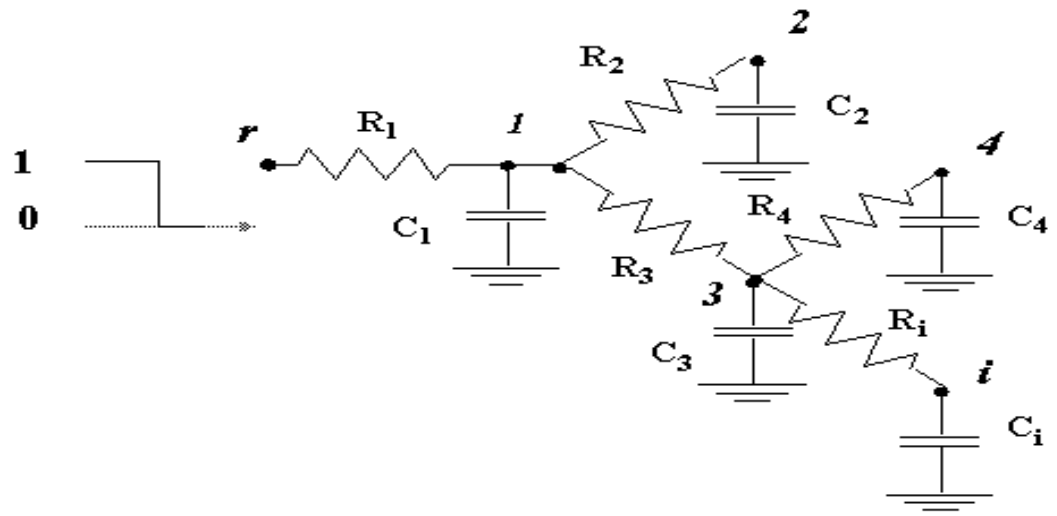


$$I_C = c\Delta L\frac{\partial V}{\partial t} = \frac{(V_{i-1} - V_i) - (V_i - V_{i+1})}{r\Delta L} \longrightarrow \boxed{rc\frac{\partial V}{\partial t} = \frac{\partial^2 V}{\partial x^2}}$$

The diffusion equation

- Idea: Lets approximate things somewhat...
  - But its a reasonably good approximation
- Time delay to fill each capacitor:
  - Resistance to that capacitor & that capacitor's capacitance
- Total time: Time to fill all capacitors
- Its a ***conservative*** and ***easy to calculate*** delay estimate
  - Will overestimate delays

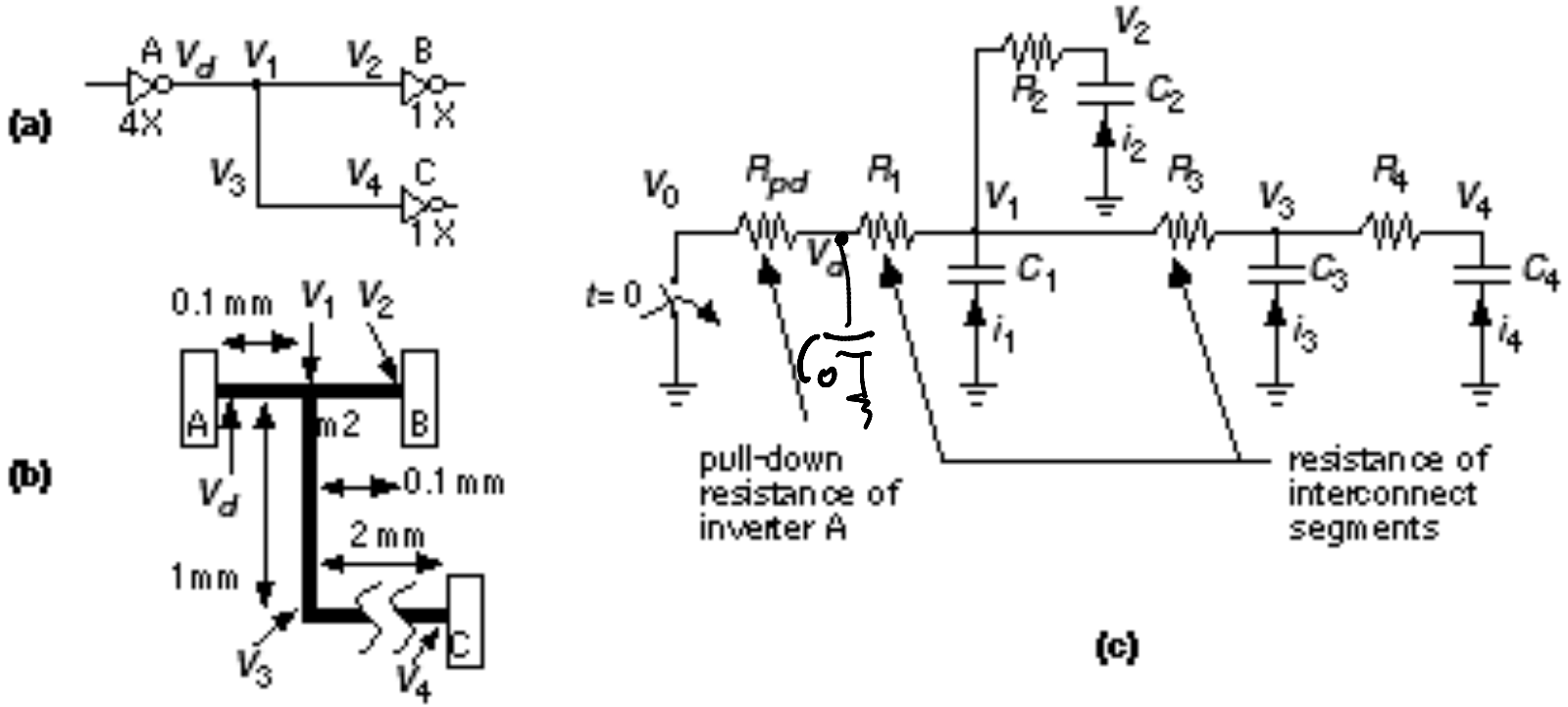
Delay Model of RC Networks:



$$\tau_{Di} = \sum_{k=1}^{N} C_k R_{ik}$$

$$R_{ik} = \sum R_j \Rightarrow (R_j \in [path(s \rightarrow i) \cap path(s \rightarrow k)])$$

Elmore Delay Example

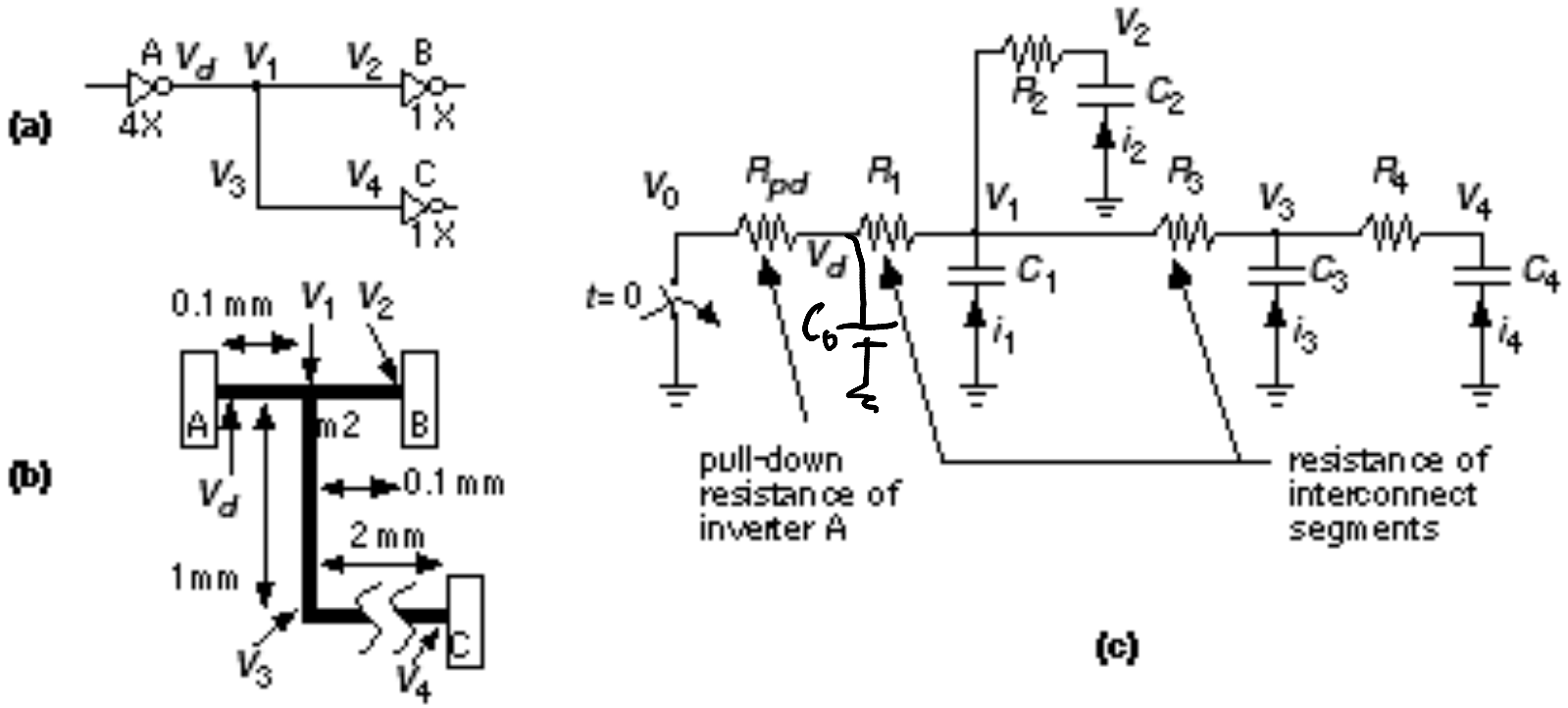


$$\tau_{D4} = \sum_{k=0}^{4} C_k R_{4k} = C_0 R_{pd} +$$
$$C_1(R_{pd} + R_1) +$$
$$C_2(Rpd + R_1) +$$
$$C_3(Rpd + R_1 + R_3) +$$
$$C_4(Rpd + R_1 + R_3 + R_4) =$$
$$R_{pd}(C_0 + C_1 + C_2 + C_3 + C_4) + R_1(C_1 + C_2 + C_3 + C_4)$$
$$+ R_3(C_3 + C_4) + R_4 C_4$$

Elmore Delay Example

- Assume:
  - m2 *r* is 50 mΩ/□
  - m2 *c* (for a minimum-width line) is 0.2fF/um
  - m2 minimum width is 90nm
  - 4x inverter R_{pd} = 500Ω
  - 1X inverter input capacitance is 20fF (a standard load)

Elmore Delay Example



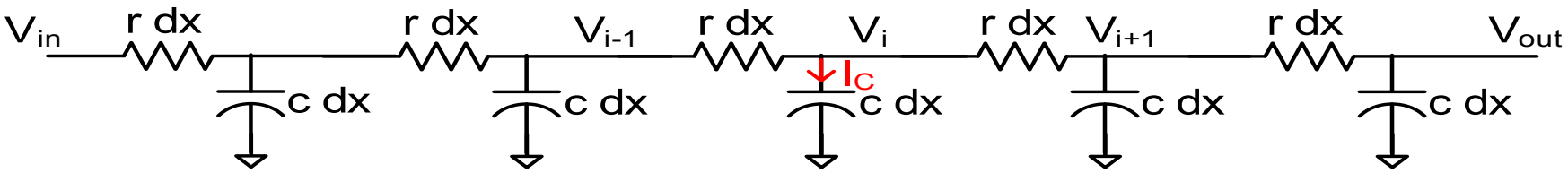
$$R_{pd} = 500\Omega, R_1 = R_2 = \frac{0.1mm}{90nm} 50m\Omega = 6\Omega, R_3 = 56\Omega,\ R_4 = 112\Omega$$

$$C_0 = 4 * 20fF = 80fF, C_1 = \frac{0.1mm\ 0.2fF}{um} = 20fF$$

$$C_2 = C_1 + 20fF = 40fF, C_3 = 200fF, C_4 = 420fF$$

$$\tau_{D4} = 380ps$$

Back to Wire Delay



Wire Model

Model the wire with N equal-length segments:

$$\tau_{DN} = \left(\frac{L}{N}\right)^2 (rc + 2rc + \ldots + Nrc) = (rcL^2)\frac{N(N+1)}{2N^2} = RC\frac{N+1}{2N}$$

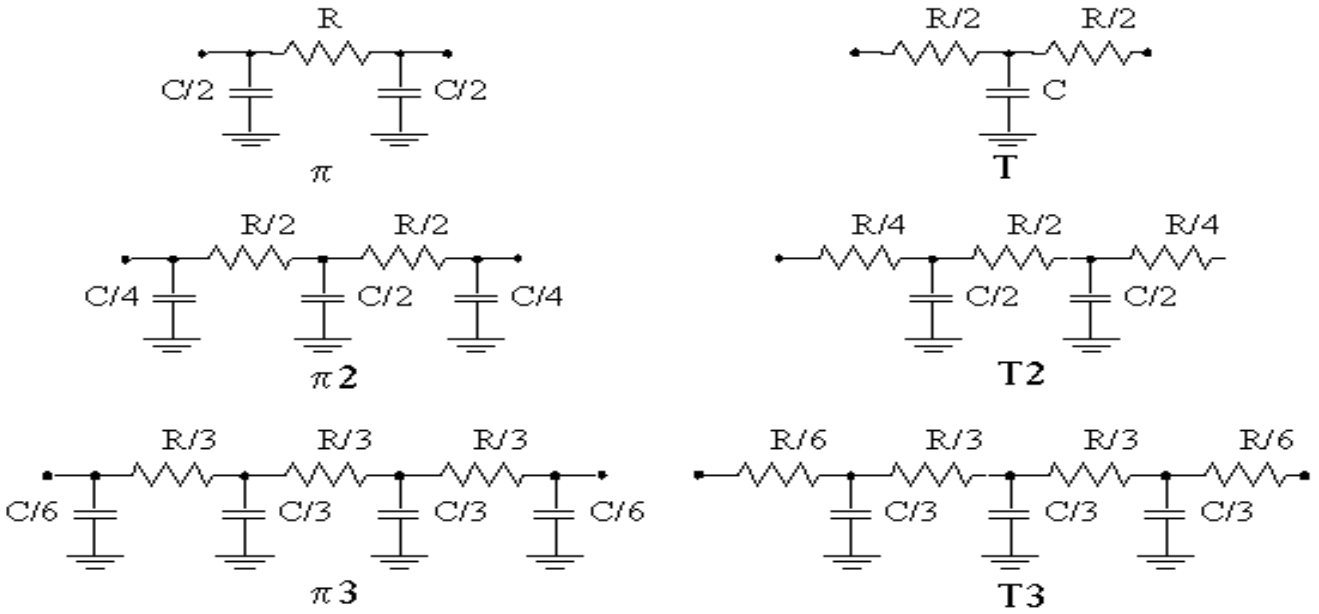
For large values of N:

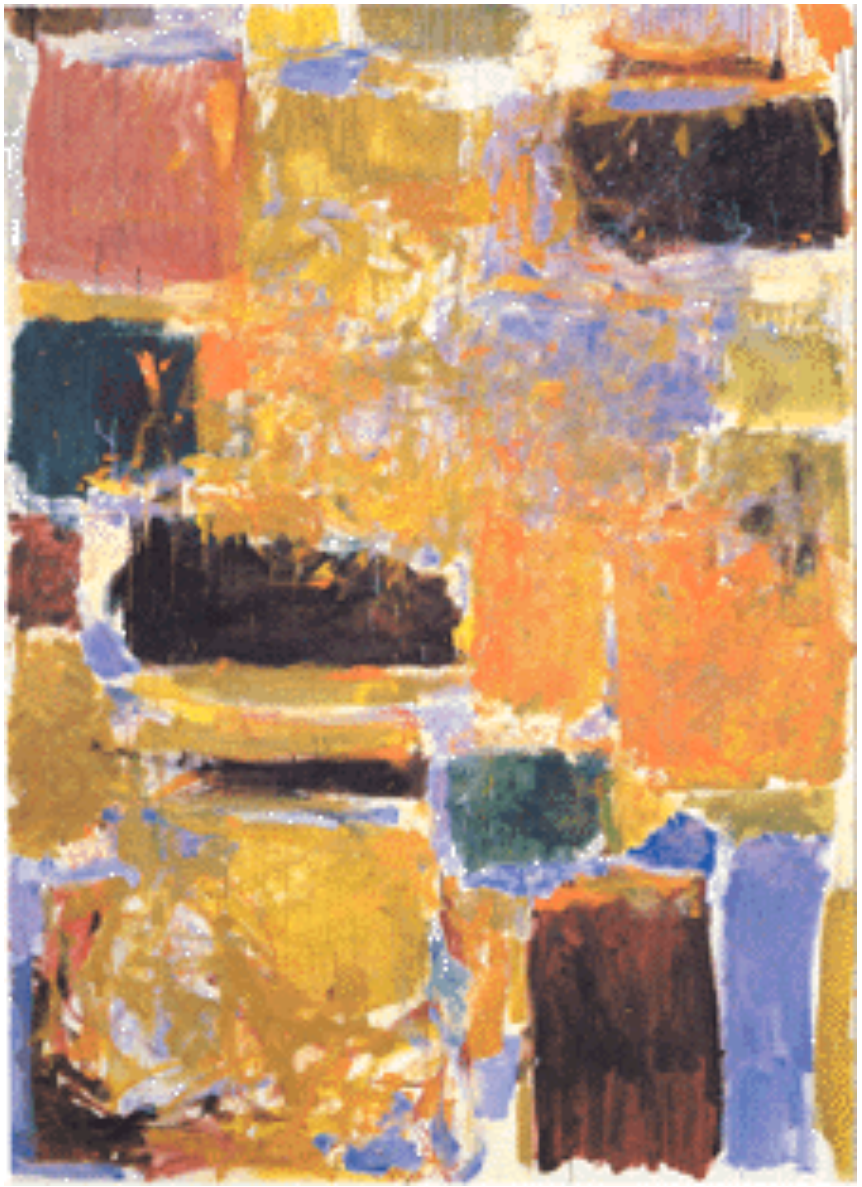
$$\tau_{DN} = \frac{RC}{2} = \frac{rcL^2}{2}$$

RC-Models

Voltage Range	Lumped RC-network	Distributed RC-network
0→50% (t_p)	0.69 RC	0.38 RC
0→63% (τ)	RC	0.5 RC
10%→90% (t_r)	2.2 RC	0.9 RC

Step Response of Lumped and Distributed RC Networks: Points of Interest.





Characteristic Impedence

When You Get To A Board...

- We have to fully treat the wires as transmission lines
  - Which means inductance also comes into play
- Modeled with "Characteristic impedance"
  - Measured in Ωs as well
- Propagation delay also matters even more
  - Since we are going longer distances
  - ~150 pS/inch
- But we don't do the math ourselves...

Why Impedance Matters...

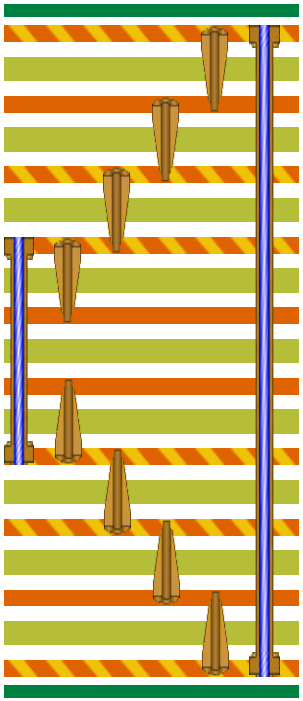
- Low impedance transmits energy better
- When you have an impedance mismatch...
  - Energy gets reflected back...
  - Which means noise & less efficiency
- Devices will specify the trace impedance needed:
  - EG, WiFi Antenna trace: 50 Ω
  - DDR3 DRAM on Zynq:
    40 Ω single, 80 Ω differential pair
- Fatter wires -> lower impedance
- Thinner boards -> lower impedance
  - It ups the capacitance but that lowers the impedance because it lowers the inductance/capacitance ratio

The Board "Stack-Up"

- Boards are produced in layers just like chips
  - Signal layers with wires
  - Plane layers with large areas of conductor for power or ground
  - Vias to connect between them
- In generating the stack-up can also find the characteristic impedance for traces

An Example Stack-Up

All MicroVias shown as  are STACKED MICROVIAS



		Finished Copper Weight	Finished Thickness (inches)
	SOLDER MASK		0.0005
L-1	TOP SIGNAL	1 Oz	0.0014
	DIELECTRIC		0.0035
L-2	PLANE	0.5 Oz	0.0007
	DIELECTRIC		0.0035
L-3	SIGNAL	0.5 Oz	0.0007
	DIELECTRIC		0.0035
L-4	SIGNAL	0.5 Oz	0.0007
	DIELECTRIC		0.0035
L-5	PLANE	0.5 Oz	0.0007
	DIELECTRIC		0.0100
L-6	PLANE	0.5 Oz	0.0007
	DIELECTRIC		0.0035
L-7	SIGNAL	0.5 Oz	0.0007
	DIELECTRIC		0.0035
L-8	SIGNAL	0.5 Oz	0.0007
	DIELECTRIC		0.0035
L-9	PLANE	0.5 Oz	0.0007
	DIELECTRIC		0.0035
L-10	BOTTOM SIGNAL	1 Oz	0.0014
	SOLDER MASK		0.0005
	Total Thickness	**0.0474 (inches)**	

Customer Saved Impedance Results

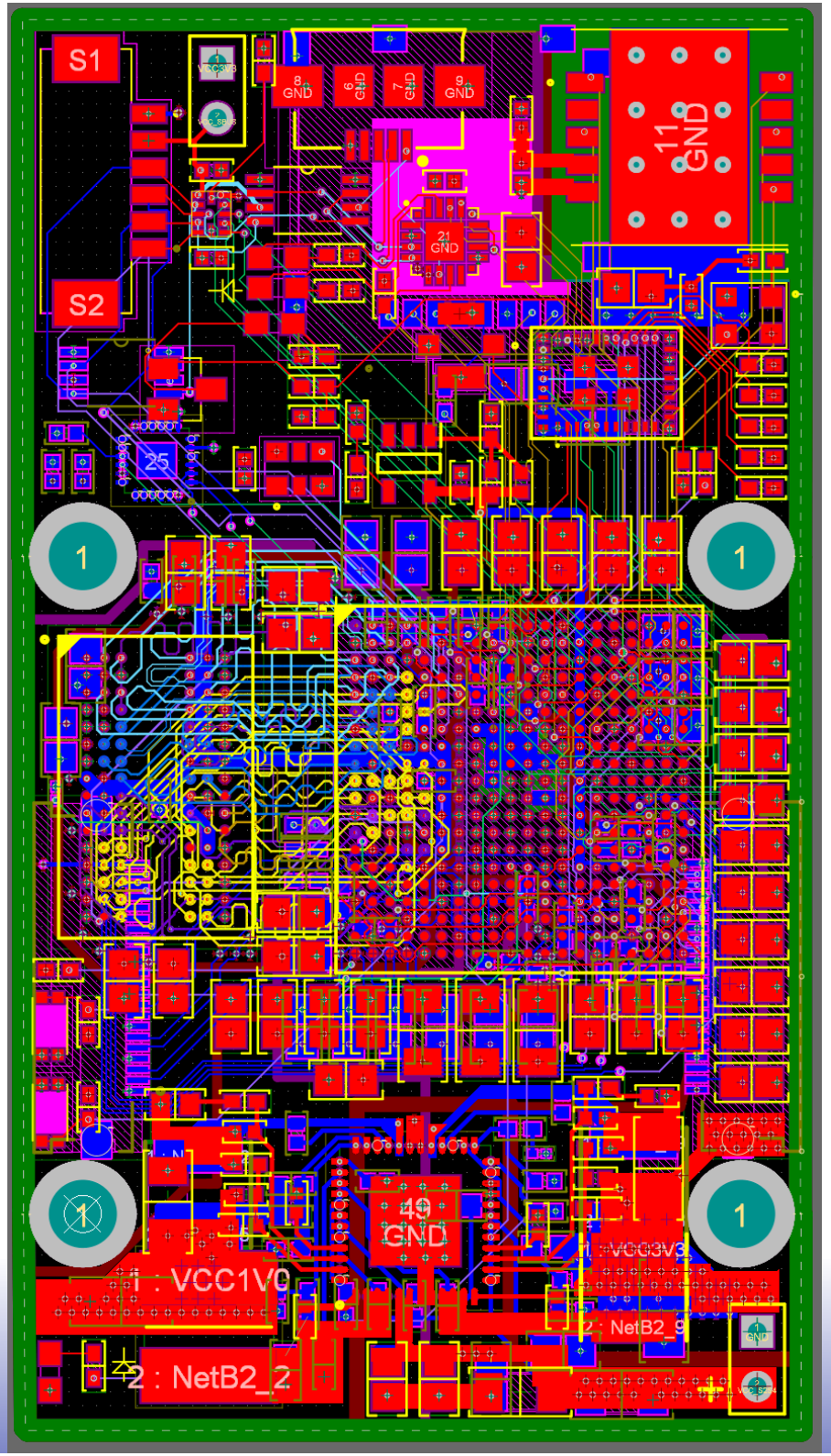
Layer	Impedance Model	Impedance (ohms)	Trace Width (mils)	Space (mils)
Layer 1	Soldermask Coated Microstrip Single-ended	51.7	5	--
Layer 1	Soldermask Coated Microstrip Differential Pair	97.82	5	10
Layer 3	Stripline Single-ended	39.32	6	--
Layer 3	Stripline Differential Pair	82.67	4	4

Board Return Paths...

- On a board, it isn't just transmitting a signal...
  - You also need a return path for the current for the ground plane
- In general, the return path should follow the trace
  - So if you have a break in the ground plane, the signal should route around the break too
- "Bypass capacitors": Read the data sheets
  - Act as temporary power reservoirs which can support specific frequency responses

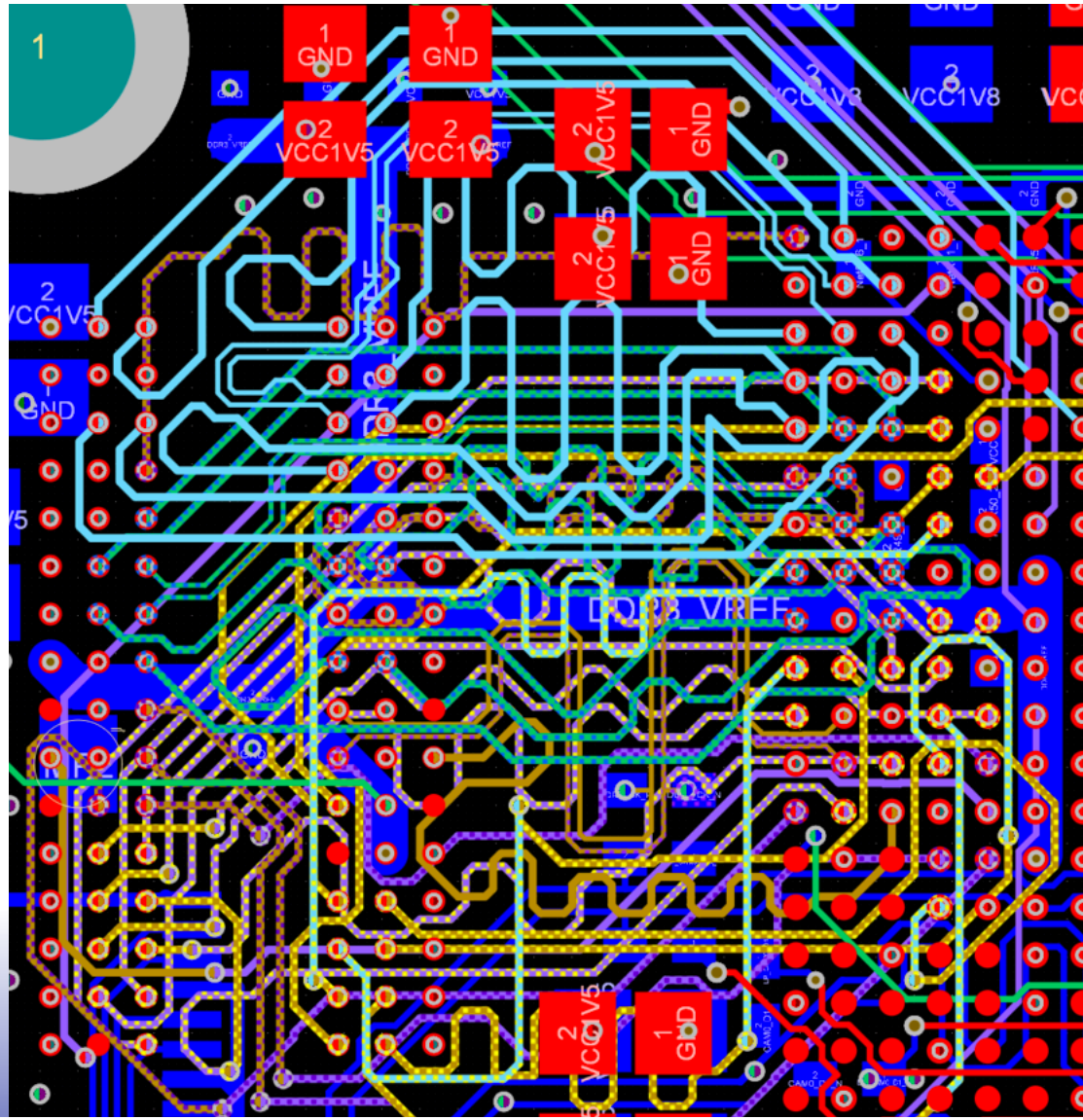
Seeing Some Of This In Action...

- Board With Routing...
- Areas of focus:
  - DDR: Controlled Impedance & length matching
  - Power & Ground planes: Low impedance, Lots of Bypassing



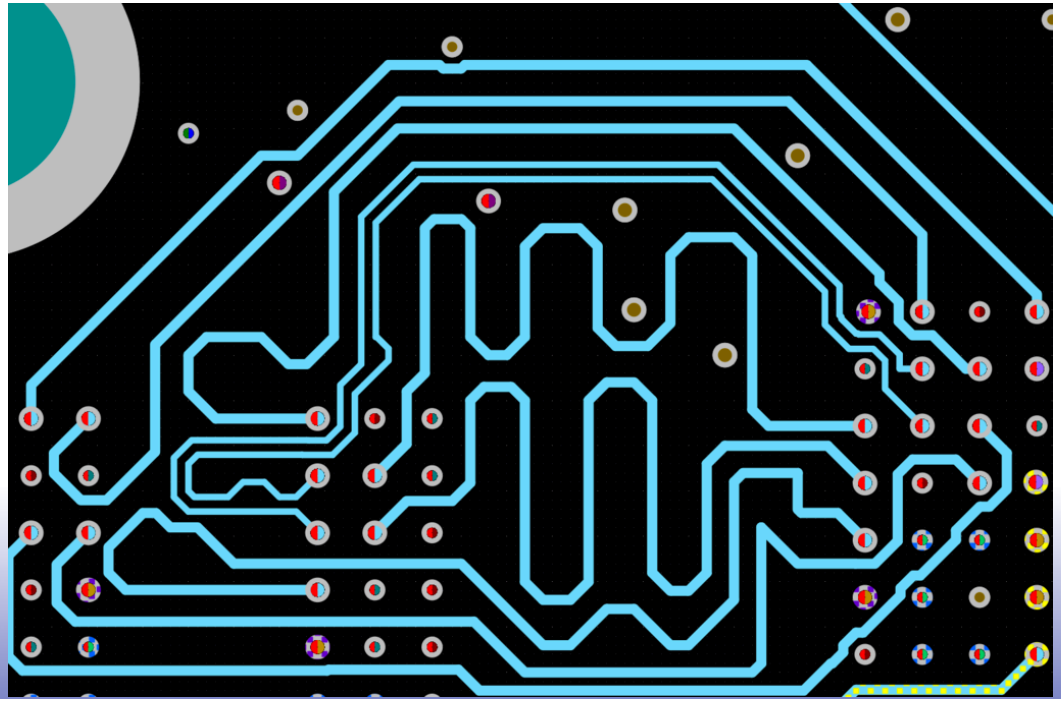
DDR Constraints

- 40Ω signal, 80Ω differential
  - Obtained from Xilinx datasheet
- 3 major groups of signals
  - 2 data banks: 8 data lines, differential data strobe
  - 1 group of control & address lines



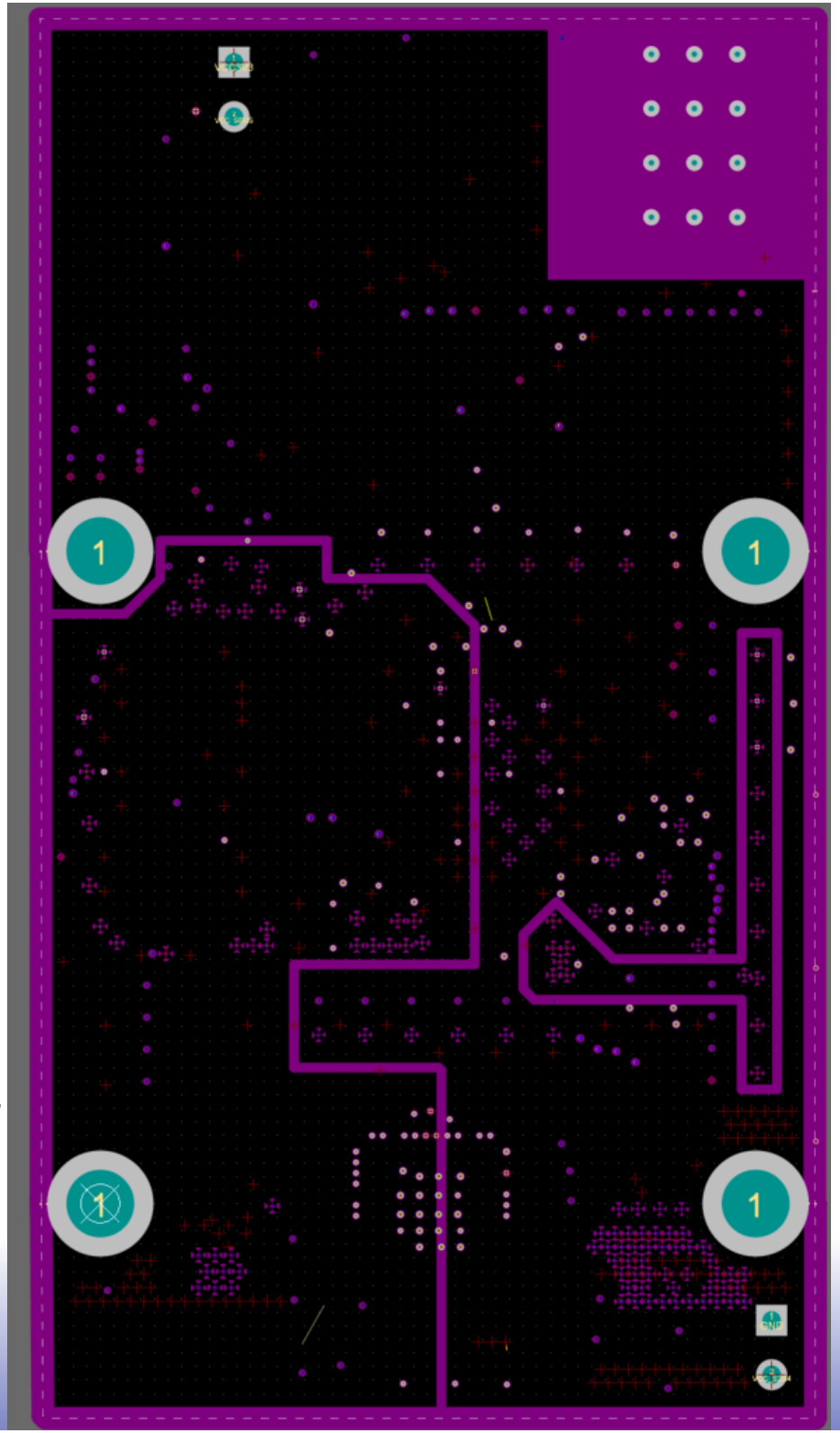
This Routing of DDR...

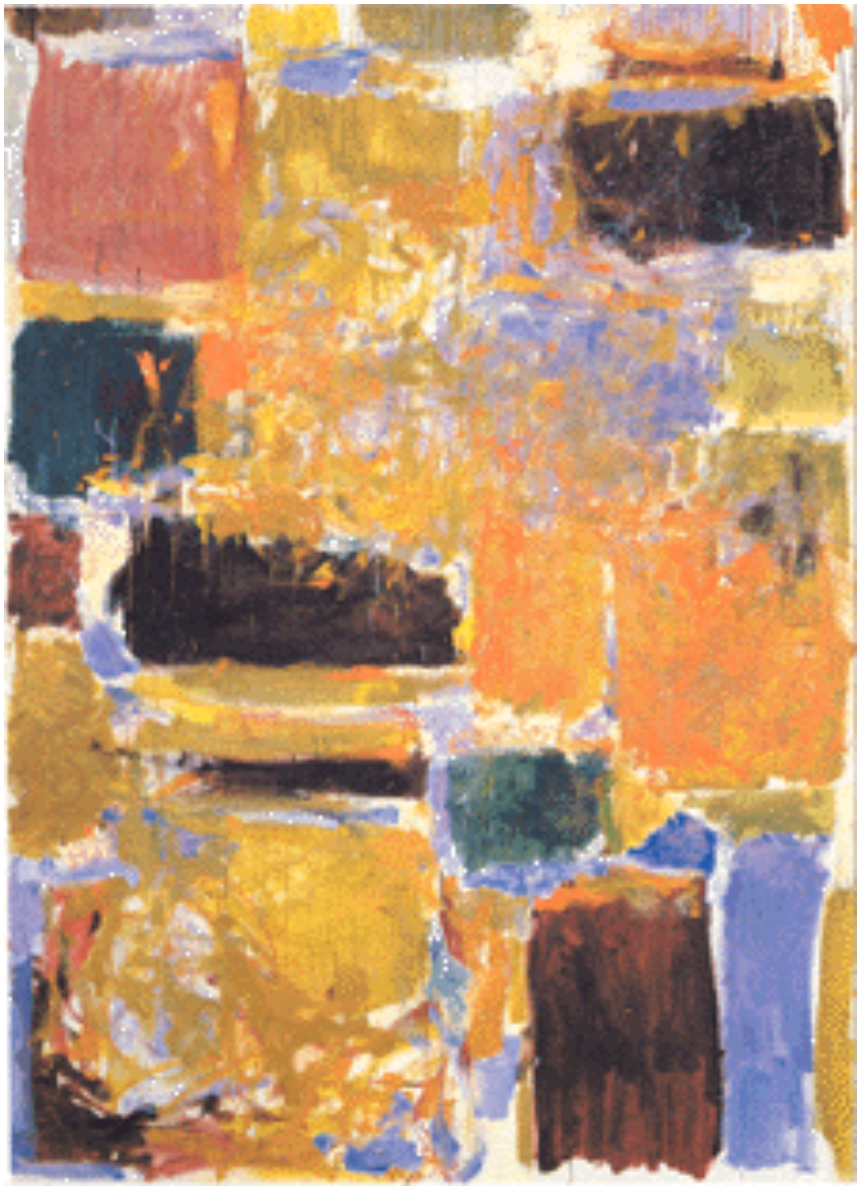
- Data-sheet's very low impedance requirement needs fat wires & thin layers
  - Limits routing to the internal layers: Can't "escape" the BGA on the top
- Length matching a semi-manual process
  - Target of .025"
    - Based on micron's data sheet



Ground and Power

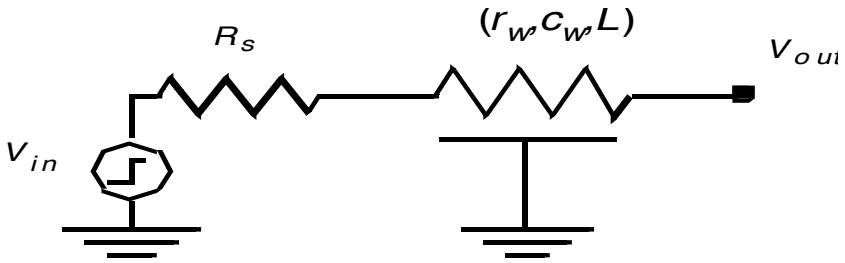
- Ground: 2 unified planes
  - Only exception a bit of isolation for the WiFi chip
- Power: L1: 1.0/3.3/1.5
- L2: 1.5/1.8/3.3
  - And a crap-ton® of bypass capacitors





Gates and Wires

Driving an RC-line



$$\tau_D = R_s C_w + \frac{R_w C_w}{2} = R_s C_w + 0.5 r_w c_w L^2$$

$$t_p = 0.69(R_s C_w + R_w C_w / 2)$$

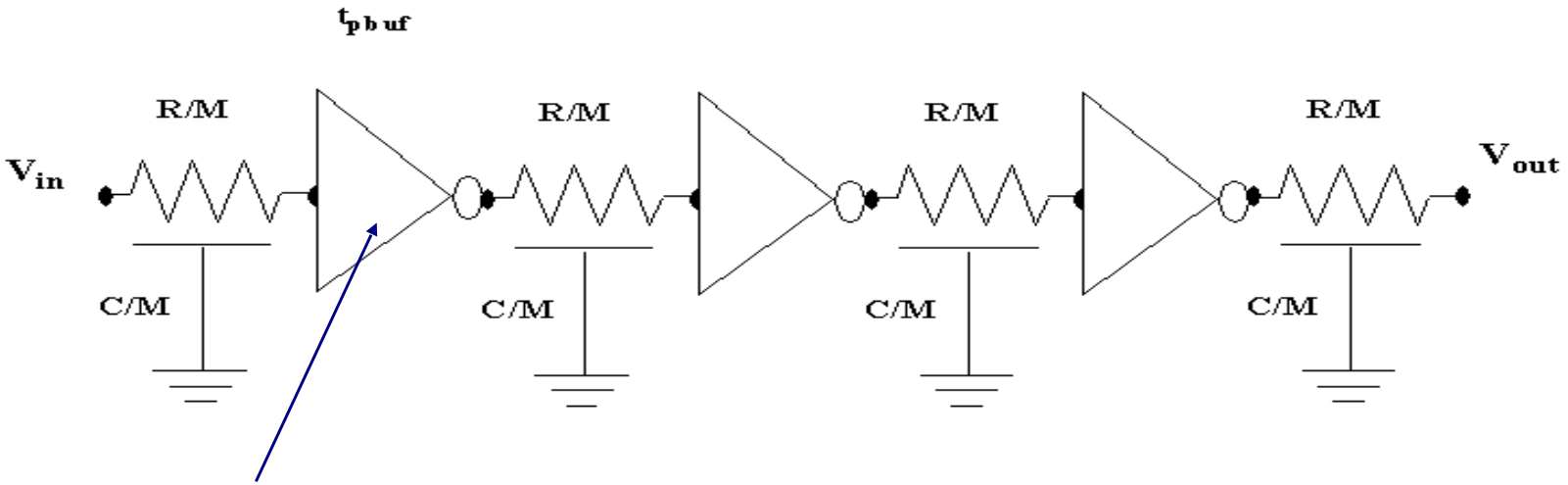
The Global Wire Problem

$$Td = 0.69(0.5R_wC_w + R_NC_{in} + R_NC_w + R_wC_{in})$$

Challenges

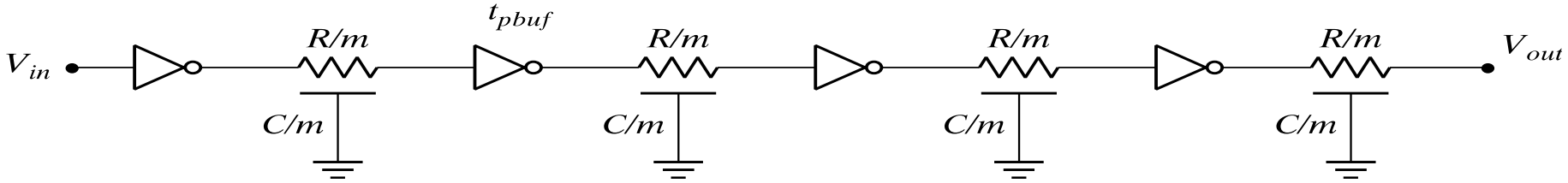
- No further improvements to be expected after the introduction of Copper (superconducting, optical?)
- Design solutions
  - Use of fat wires
  - Efficient chip floorplanning
  - Insert repeaters

Reducing RC-delay Using Repeaters



Repeater

Repeaters



$$t_p = 0.69m\left(\frac{R_N}{W}(W\gamma C_{in} + \frac{cL}{m} + WC_{in}) + \frac{rL}{m}(WC_{in} + 0.5\frac{cL}{m})\right)$$

$$m_{opt} = L\sqrt{\frac{rc}{2R_N C_{in}(\gamma + 1)}} = \sqrt{\frac{t_{pwire(unbuffered)}}{t_{p1}}}$$

$$W_{opt} = \sqrt{\frac{R_N c}{rC_{in}}}$$

Repeater Insertion

$$m_{opt} = L\sqrt{\frac{rc}{2R_N C_{in}(\gamma+1)}} = \sqrt{\frac{t_{pwire(unbuffered)}}{t_{p1}}}$$

$$W_{opt} = \sqrt{\frac{R_N c}{r C_{in}}}$$

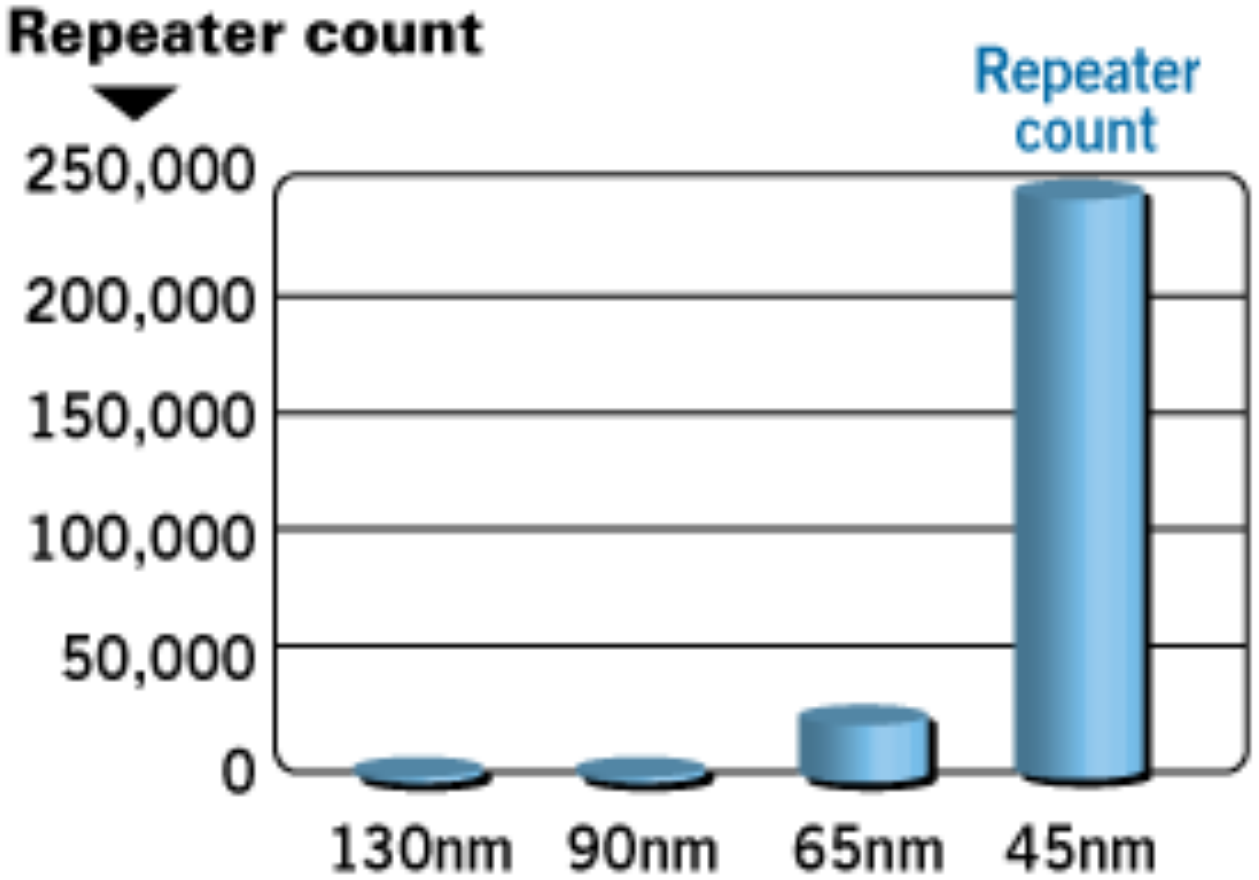
For a given technology and a given interconnect layer, there exists an optimal length of the wire segments between repeaters. The delay of these wire segments is independent of the routing layer!

$$L_{crit} = \frac{L}{m_{opt}} = \sqrt{\frac{2t_{p1}}{0.69rc}}$$

$$t_{p,crit} = \frac{t_{p,\min}}{m_{opt}} = 2\left(1+\sqrt{\frac{2}{(1+\gamma)}}\right)t_{p1}$$

From Elmore example: rc = 0.1fs/um², t_{p1} = 55ps, L_{crit} = 1262um (rule of thumb ~ 0.5-1mm)

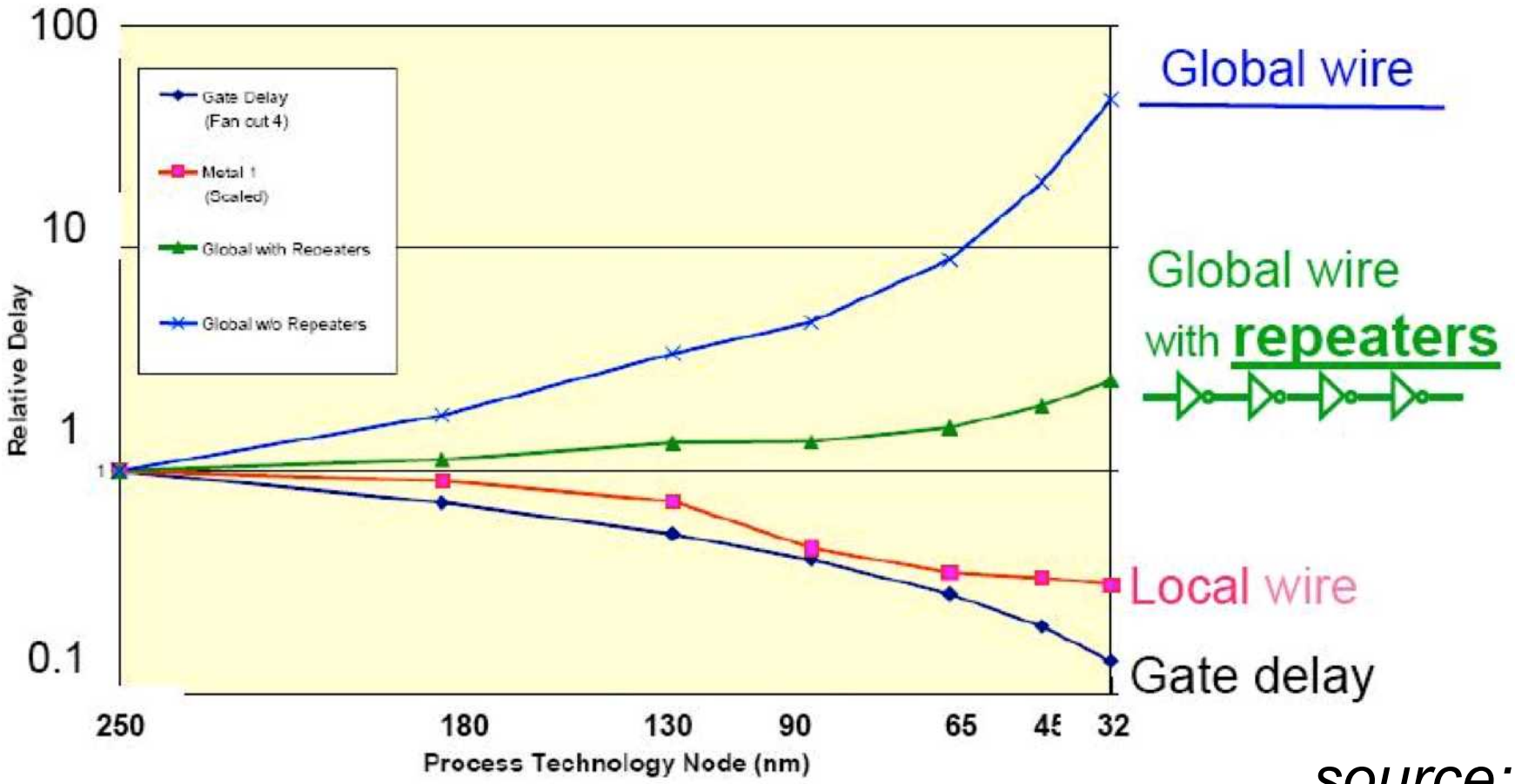
Importance of Repeaters



Source: IBM POWER processors,
R. Puri et al SRC Interconnect Forum 2006

- In modern designs the number of repeaters increases dramatically

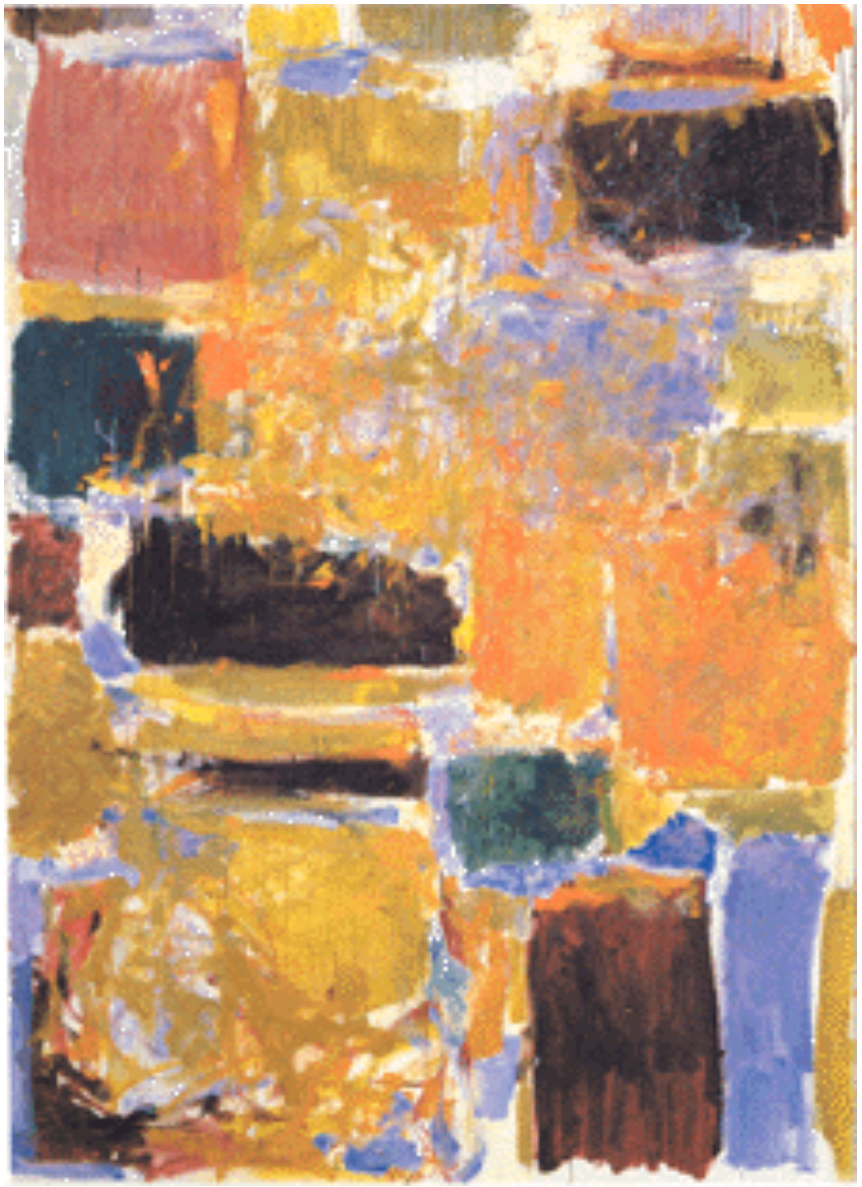
Wire and Gate Delay Scaling



source: ITRS

Delay for Metal 1 and Global Wiring versus Feature Size

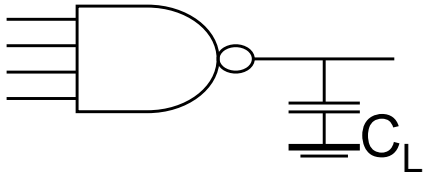
- Gate delay gets better, wire delay gets worse

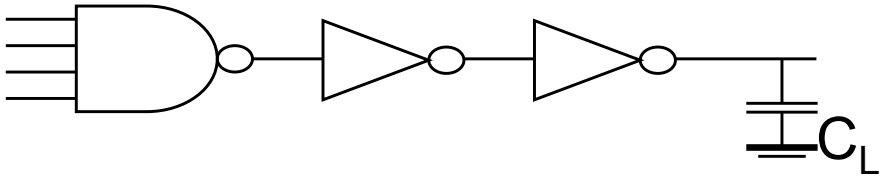


Logical Effort

Question #1

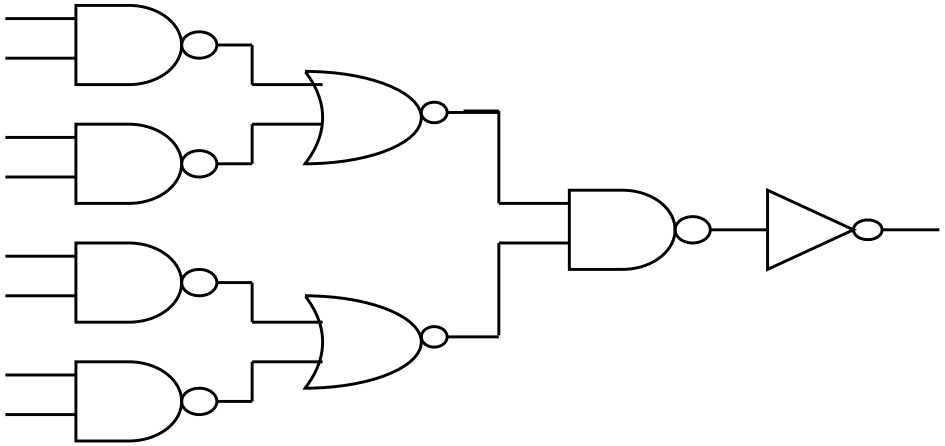
- How to best combine logic and drive for a big capacitive load?

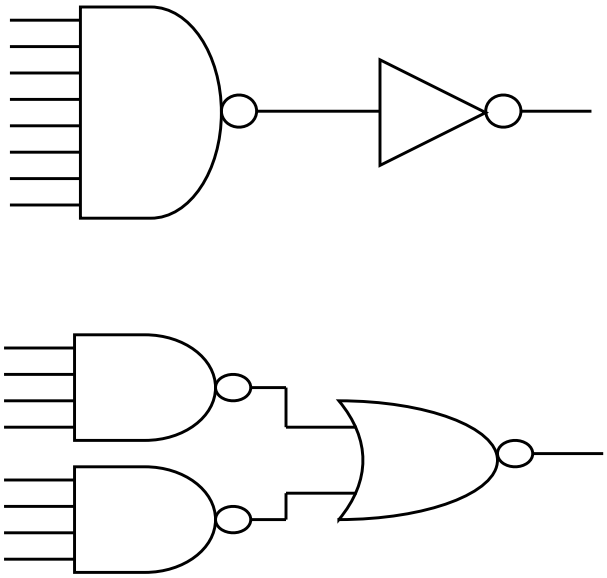




Question #2

- All of these are “decoders”
  - Which one is “best”?





Method to answer both of these questions

- Extension of buffer sizing problem
- Logical effort

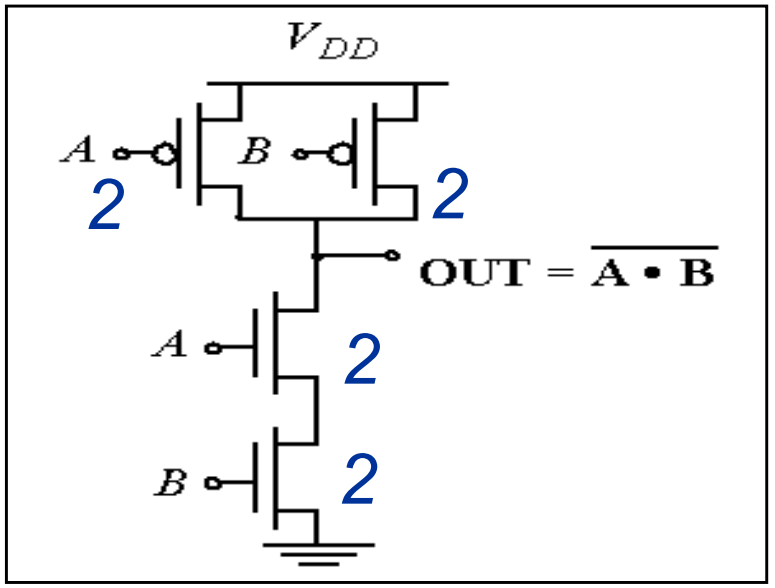
Complex Gate Sizing

Complex Gate Sizing: NAND-2 Example

$$C_{gnand} = 4C_G = (4/3)\, C_{ginv}$$

$$C_{dnand} = 6C_D = 6\gamma C_G = 2\gamma C_{ginv}$$

$$f = C_L/C_{gnand} = (3/4)\, C_L/C_{ginv}$$



$$t_{pNAND} = kR_N(C_{dnand} + C_L)$$

$$= kR_N(2\gamma C_{ginv} + C_L)$$

$$= kR_N C_{ginv}\,(2\gamma + C_L/C_{ginv})$$

$$= t_{inv}\,(2\gamma + (4/3)f)$$

Logical Effort

- Defines ease of gate to drive external capacitance
- Inverter has the smallest logical effort and intrinsic delay of all static CMOS gates
- Logical effort LE is defined as:
  - $(R_{eq,gate}C_{in,gate})/(R_{eq,inv}C_{in,inv})$
  - Easiest way to calculate (usually):
    - Size gate to ***deliver same current*** as an inverter, take ratio of ***gate input capacitance*** to inverter capacitance
- LE increases with gate complexity

Logical Effort

$$t_{pgate} = t_{inv}\,(p + LEf)$$

Measure everything in units of t_{inv} (divide by t_{inv}):

p – intrinsic delay - gate parameter ≠ f(W)
LE – logical effort – gate parameter ≠ f(W)
f – electrical fanout = C_L/C_{in} = f (W)

Normalize everything to an inverter:
LE_{inv} =1, p_{inv} = γ

Delay of a Logic Gate

Gate delay:

$$Delay = EF + p \quad \text{(measured in units of } t_{inv}\text{)}$$

effective fanout — intrinsic delay

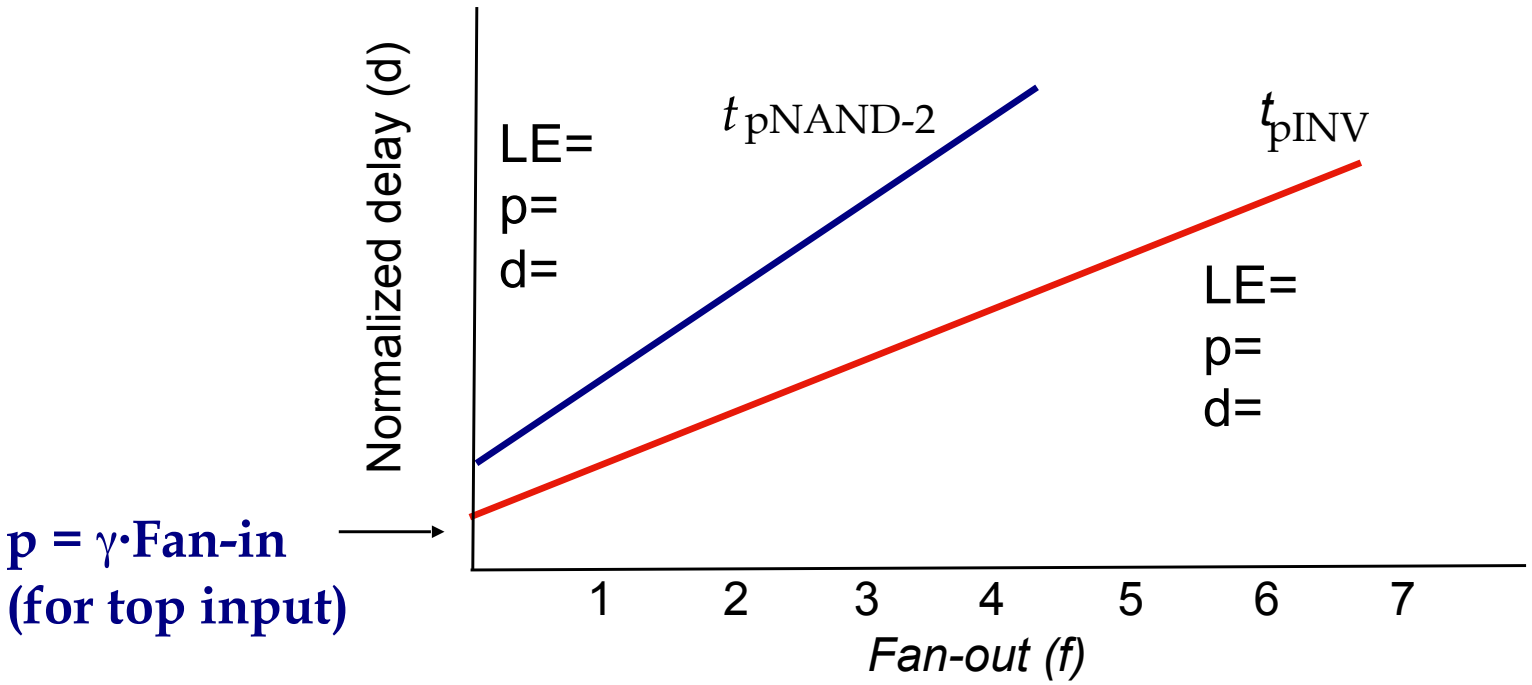
Effective fanout:

$$EF = LE\ f$$

logical effort — electrical fanout $= C_L/C_{in}$

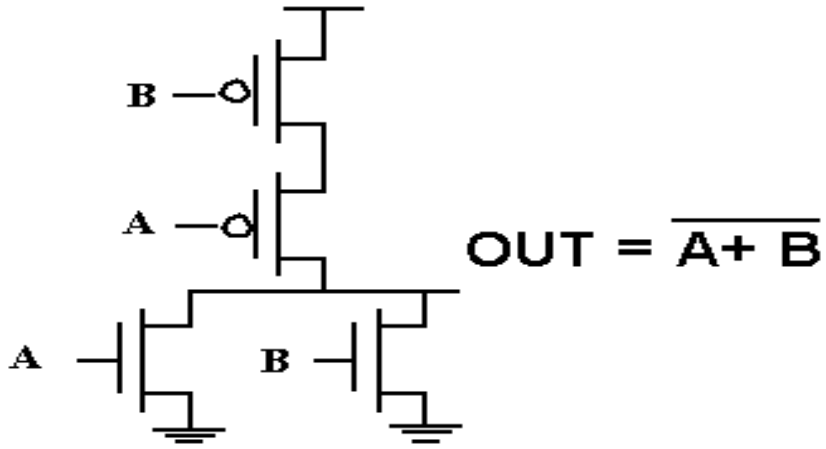
Logical effort is a function of topology, independent of sizing
Effective fanout is a function of load/gate size

Logical Effort of Gates



Delay Of NOR-2 Gate

1. Size for same resistance as inverter
2. LE = ratio of input cap of gate versus inverter



Intrinsic capacitance (C_{dnor}) =

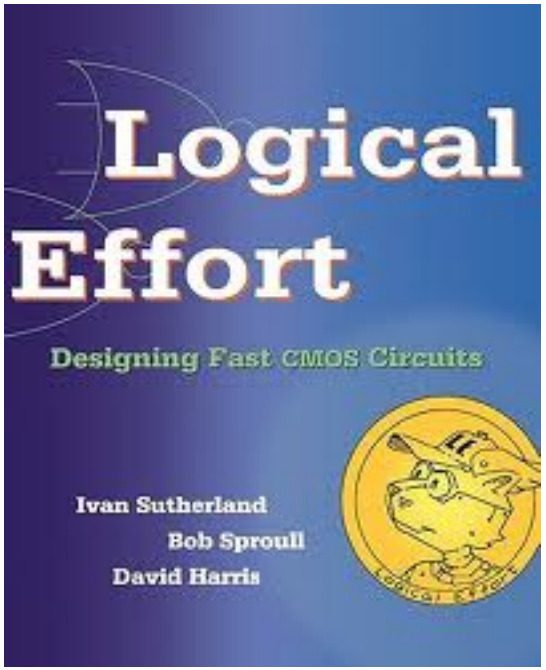
t_{pint} (NOR) =

Question

Any logic function can be implemented using NOR gates only or NAND gates only!

Which of the two approaches is preferable in CMOS (from a performance perspective)?

Logical Effort



Gate Type	Number of Inputs			
	1	2	3	n
Inverter	1			
NAND		4/3	5/3	(n + 2)/3
NOR		5/3	7/3	(2n + 1)/3
Multiplexer		2	2	2
XOR		4	12	

[From Sutherland, Sproull, Harris]