VLSI for Architects

Krste Asanović

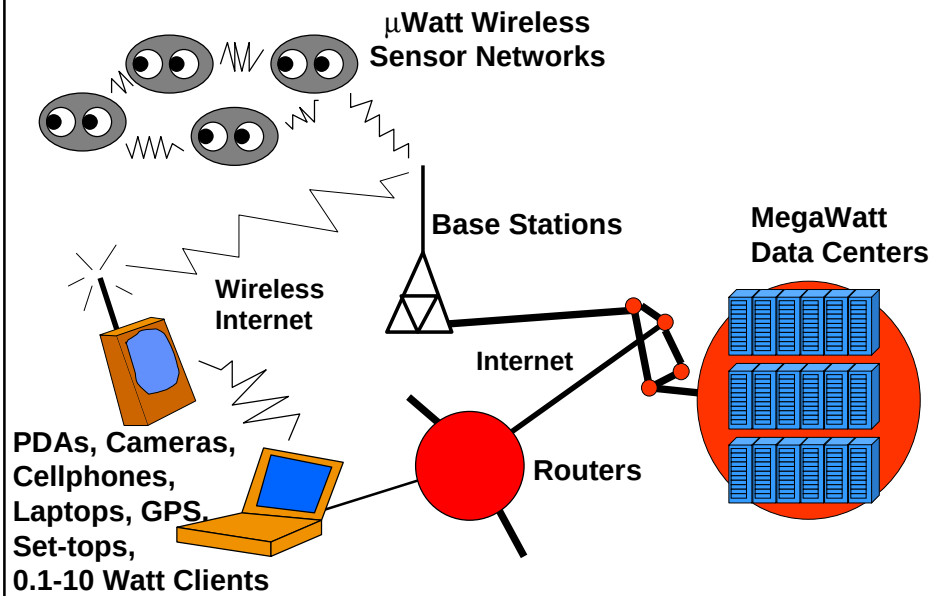
Computer Architecture Group

MIT Laboratory for Computer Science

krste@lcs.mit.edu

http://www.cag.lcs.mit.edu/6.893-f2000/

Future Computing Infrastructure



Semiconductor Trends

- **Non-Recurring Engineering (NRE) costs are increasing rapidly for new designs**
  - **>$1M for masks to spin a new design**
  - **Engineers cost ~$200K/year (salary+benefits+overhead)**
  - **Pentium Pro design verification took around 350 engineer years or ~$70M**

=> Tremendous economies of scale

(Can't sell <1,000,000 parts for <$100 each)

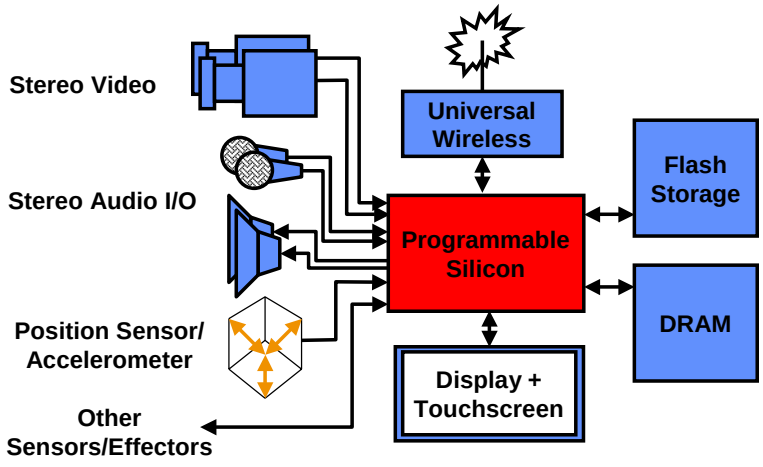
- **CMOS following Moore's Law until (at least) 2011-2014**
  - **ITRS'99* roadmap 2011, 50nm technology**
    - **64 Gb DRAMs (8 GB/chip)**
    - **7 billion transistor CPUs**
    - **10 GHz clocks (100 ps cycle time)**

=> Smallest viable chips have huge capacity

(~10 million transistors/mm^2,
10 million transistors per person per day)

[*International Technology Roadmap for Semiconductors]

Programmable Silicon Replaces Custom Hardware



Programmable silicon replaces ASICs, or collections of DSPs, microprocessors and glue logic

Benchmarks & Metrics

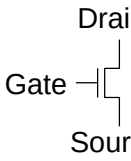
- Application space wider than desktop processors
  - Benchmark as many applications as possible
  - Include apps done with special hardware now (graphics, audio, crypto)
  - Whole system measures
  - Real-time important
- Primary metrics
  - Cost (related to die area but also whole system cost)
  - Execution Time (latency and throughput, average and worst-case)
  - Energy (also peak power and peak switching current)
- Compare against best possible solution for each application
  - How much worse than application-specific circuitry?
  - Moore's law perhaps makes area the most forgiving dimension
    - try to keep energy and delay competitive, possibly at expense of area

VLSI for Architects

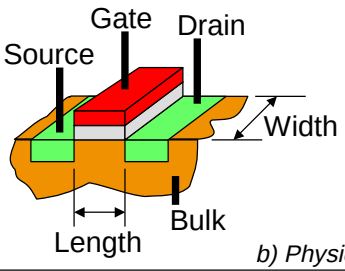
Two types of question architects ask:

- How will this change affect area/delay/energy in current technology?
- How will this design scale to future technologies?
- For next 10-15 years, *the* technology is CMOS

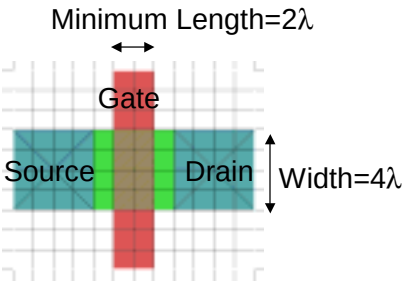
Transistors



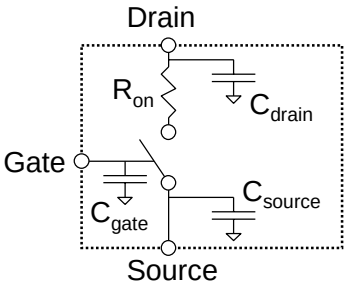
a) Circuit Symbol



b) Physical Realization

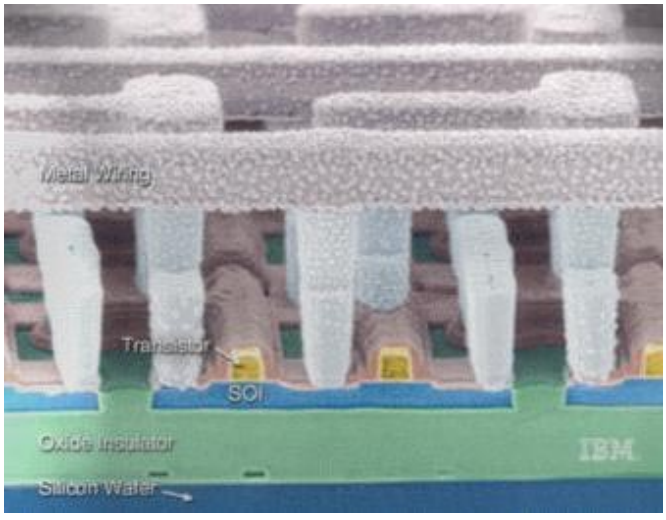


c) Layout View



d) Simple RC Model

Transistors

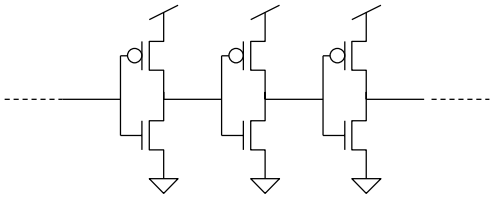


IBM SOI Technology

Method of Logical Effort
(Sutherland and Sproul)

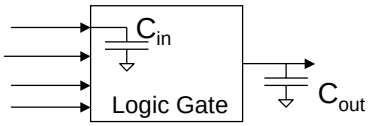
- Easy way to estimate delays in CMOS process
- Indicates correct number of logic stages to use and transistor sizes
- Characterize process speed with single delay parameter: τ

τ, delay of inverter driving same-sized inverter (no parasitics)



τ in range 10-15ps for 0.18μm processes

Gate Delay Components

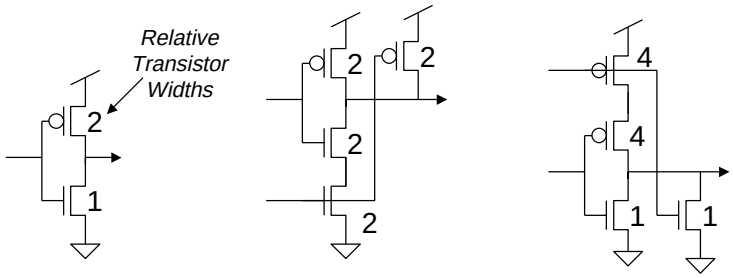


- Split delay of logic gate into three components

  Delay = Logical Effort x Electrical Effort + Parasitic Delay
- Logical Effort
  - Complexity of logic function (Invert, NAND, NOR, etc)
  - Define inverter has logical effort = 1
  - Depends only on topology not transistor sizing
- Electrical Effort
  - Ratio of output capacitance to input capacitance C_{out}/C_{in}
- Parasitic Delay
  - Intrinsic self-loading of gate
  - Independent of transistor sizes and output load

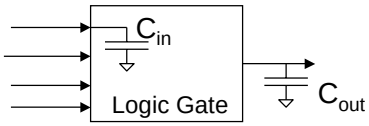
Logical Effort for Simple Gates

- Define Logical Effort of Inverter = 1
- For other gates, size to give same current drive as inverter
- Logical Effort is ratio of logic gate's input cap. to inverter's input cap.



Inverter	NAND	NOR
Input Cap = 3 units	Input Cap = 4 units	Input Cap = 5 units
L.E.=1 (definition)	L.E.=4/3	L.E.=5/3

Electrical Effort

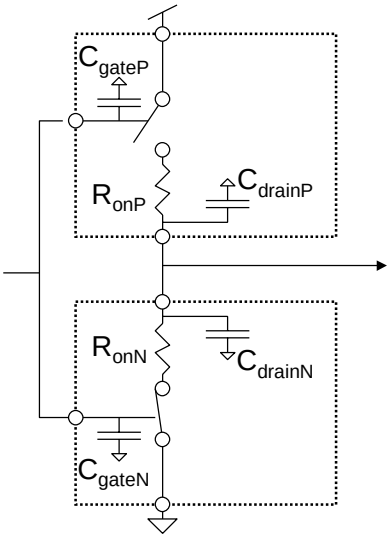


- Ratio of output load capacitance over input capacitance:

$$E.E. = C_{out}/C_{in}$$

- Usually, transistors have minimum length
- Input and output capacitances can be measured in units of transistor gate widths

Parasitic Delay



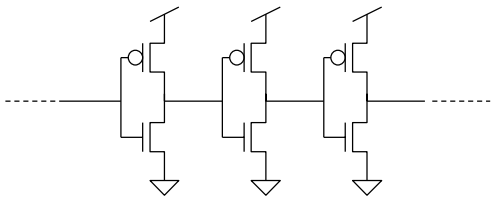
- Main cause is drain capacitances
- These scale with transistor width so P.D. independent of transistor sizes
- Useful approximation:

$$C_{gate} \sim= C_{drain}$$

- For inverter:

Parasitic Delay ~= 1.0 τ

Inverter Chain Delay



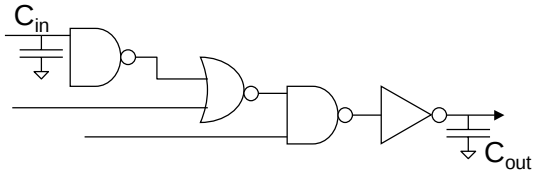
- For each stage:

Delay = Logical Effort x Electrical Effort + Parasitic Delay

= 1.0 *(definition)* x 1.0 *(in = out)* + 1.0 *(drain caps)*

= 2.0 units

Optimizing Circuit Paths

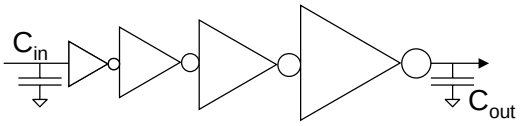


- Path logical effort, $G = \prod g_i$ (g_i = L.E. stage i)
- Path electrical effort, $H = C_{out}/C_{in}$ (h_i = E.E. stage i)
- Parasitic delay, $P = \sum p_i$ (p_i = P.D. stage i)
- Path effort, $F = GH$
- Minimum delay when each of N stages has equal effort

$$\text{Min. } D = NF^{1/N} + P$$

$$\text{i.e. } g_i h_i = F^{1/N}$$

Optimal Number of Stages



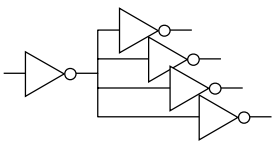
- Minimum delay when:

  stage effort = logical effort x electrical effort ~= 3.4-3.8

  - Some derivations have *e* = 2.718.. as best stage effort – this ignores parasitics
  - Broad optimum, stage efforts of 2.4-6.0 within 15-20% of minimum

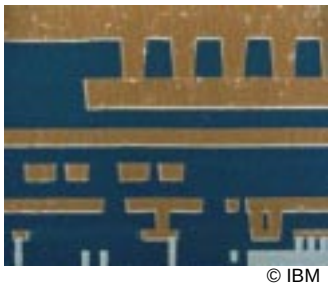
- Fan-out-of-four (FO4) is convenient design size *(~5τ)*

FO4 delay: Delay of inverter driving four copies of itself



Wires


© IBM

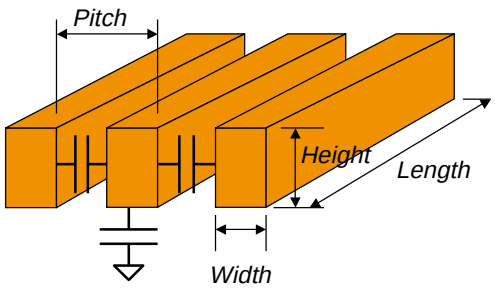

© IBM


© IBM

IBM CMOS7
process

6 layers of
copper wiring

Wires



- **Resistance fixed by (length*resistivity) / (height*width)**
  - **bulk aluminum 2.8 μΩ-cm, bulk copper 1.7 μΩ-cm**
- **Capacitance depends on geometry of surrounding wires and relative permittivity, ε_r,of dielectric**
  - **silicon dioxide ε_r = 3.9, new low-k dielectrics in range 1.2-3.1**

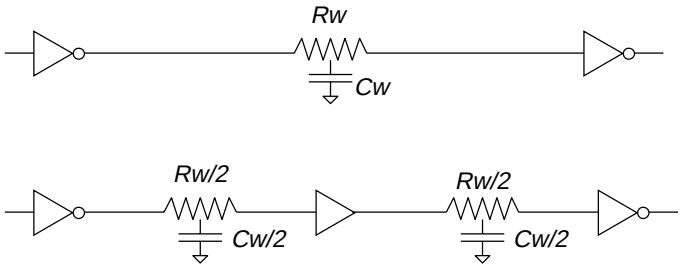
Current Interconnect Densities

- Intel Pentium-III, 0.18μm, 6 aluminum layers, SiOF dielectric (ε_r = 3.1)

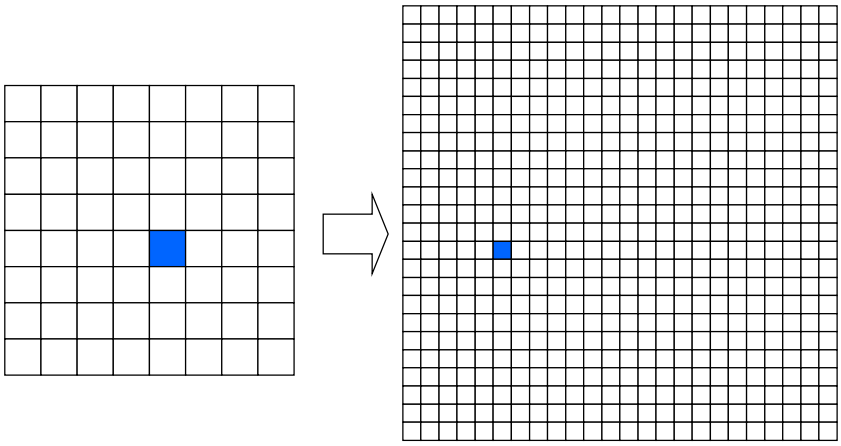
Metal Layer	Pitch (μm)	Aspect Ratio (Height/Width)
M1	0.50	1.9
M2	0.64	2.2
M3	0.64	2.2
M4	1.08	2.0
M5	1.60	2.0
M6	1.76	2.0

Wire Delays

- Resistance, R, increases per unit length
  - in 0.25μm CMOS, ~1000λ thin M1 wire = minimum inverter resistance
- Capacitance, C, increases per unit length
  - in 0.25μm CMOS, ~1000λ thin M1 wire = minimum inverter capacitance
- Wire delay increases as RC, quadratic in length
  - in 0.25μm, ~1000λ thin M1 wire = 30ps (~τ)
- Inserting repeaters makes delay linear with length



Scaling



- Scale linear dimensions by factor S (around 0.7 / generation)
- Chip size also increases

Scaling Slides from Horowitz DAC'2000 Talk

http://www.dac.com/37slides/05_2.ppt

(link on class web page)