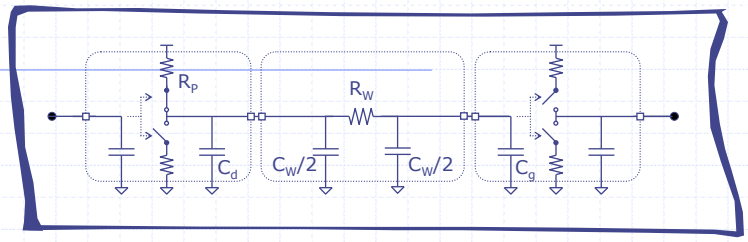
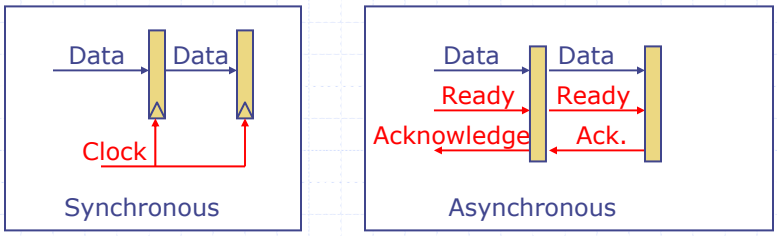
Physical Design – 2: Clock and Power



Arvind
Computer Science & Artificial Intelligence Lab
Massachusetts Institute of Technology

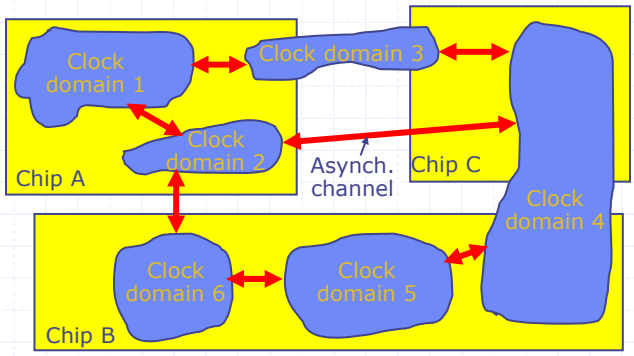
Digital System Need Timing Conventions …

- about when a receiver can sample an incoming data value
  - synchronous systems use a common clock
  - asynchronous systems encode "data ready" signals alongside, or encoded within, data signals
- for when it's safe to send another value
  - synchronous systems, on next clock edge (after hold time)
  - asynchronous systems, acknowledge signal from receiver



Clock Domains

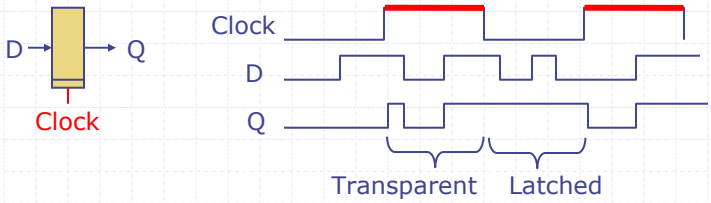
Most large ASICs, and systems built with these ASICs, have several synchronous clock domains connected by asynchronous communication channels



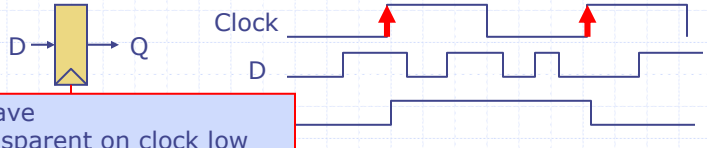
We'll focus on a single synchronous clock domain in this class

Clocked Storage Elements

- Transparent Latch, Level Sensitive
  - data passes through when clock is high, latched when low

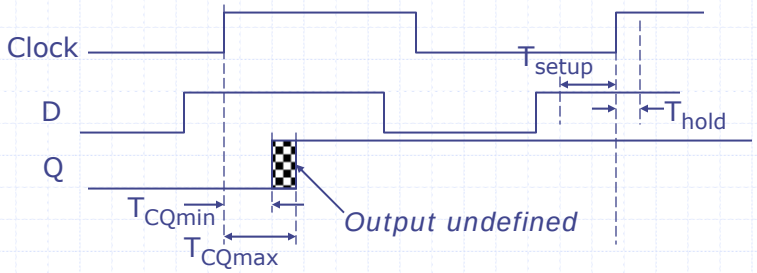


- D-Type Register or Flip-Flop, Edge-Triggered
  - data captured on rising edge of clock, held for rest of cycle



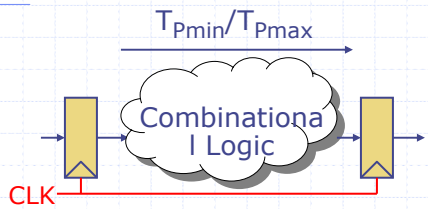
Can also have
- latch transparent on clock low
- negative-edge triggered flip-flop

Flip-Flop Timing Parameters



- T_{CQmin}/T_{CQmax}
  - propagation of D→Q at clock edge
- T_{setup}/T_{hold}
  - define window around rising clock edge during which data must be steady to be sampled correctly
  - either setup or hold time can be negative

Edge-Triggered Timing Constraints



Single clock with edge-triggered registers

common in stdcell ASICs

- Slow path timing constraint

$$T_{cycle} \geq T_{CQmax} + T_{Pmax} + T_{setup}$$

  - can always work around slow path by using slower clock
- Fast path timing constraint

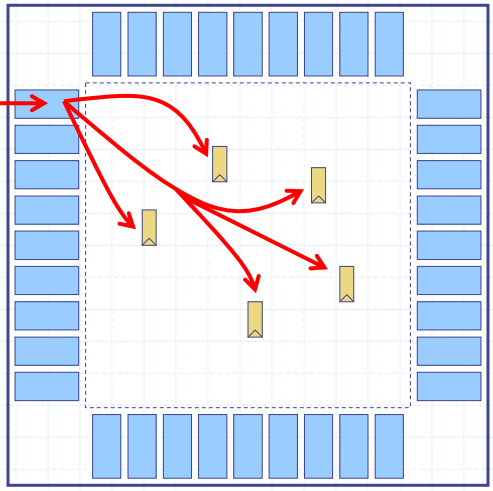
$$T_{CQmin} + T_{Pmin} \geq T_{hold}$$

  - bad fast path cannot be fixed without redesign!
  - might have to add delay into paths to satisfy hold time

Clock Distribution

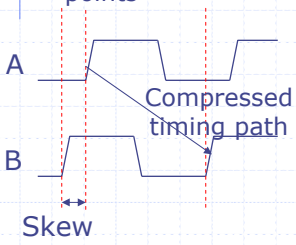
Clock

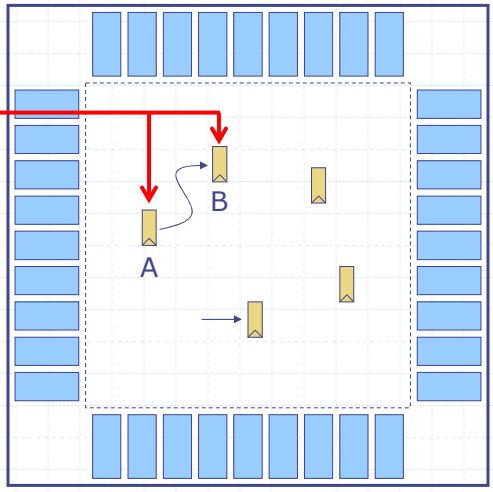
Cannot really distribute clock instantaneously with a perfectly regular period



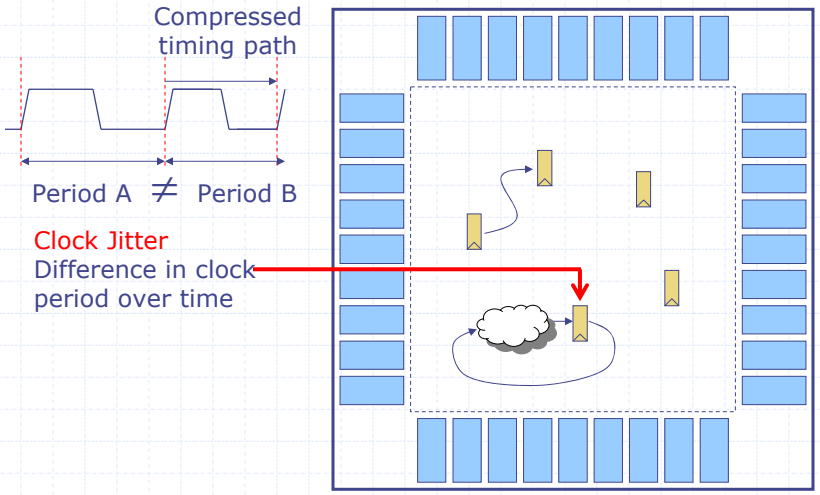
Clock Skew: Spatial Clock Variation

Clock Skew
Difference in clock arrival time at two spatially distinct points

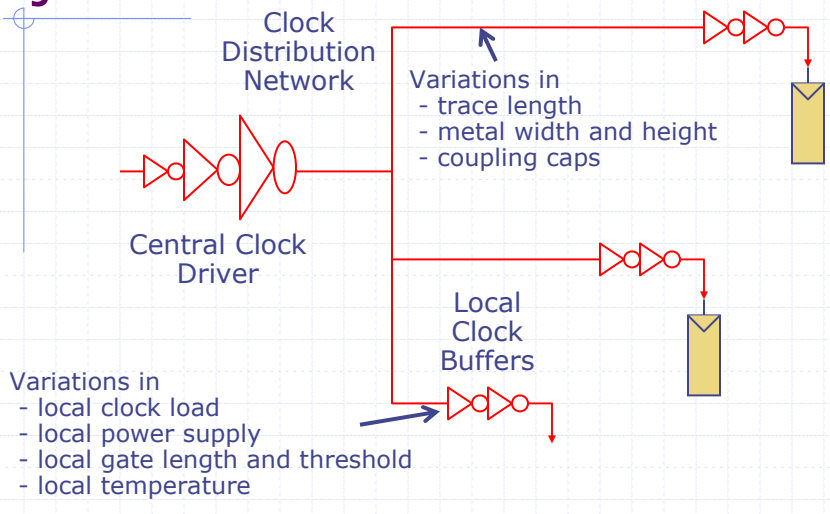




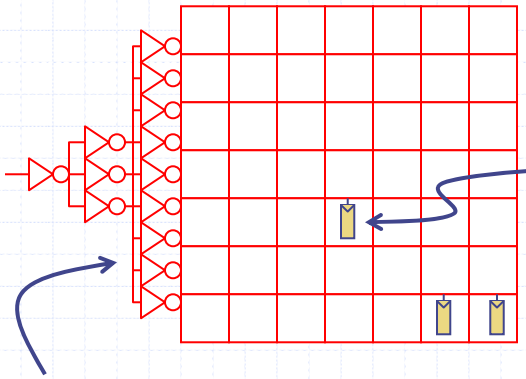
Clock Jitter: Temporal Clock Variation



How do clock skew and jitter arise?



Clock Distribution with Clock Grids



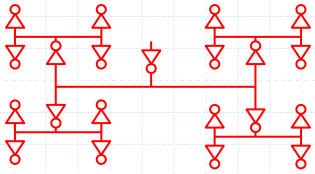
Grid feeds flops directly, no local buffers

Low skew but high power

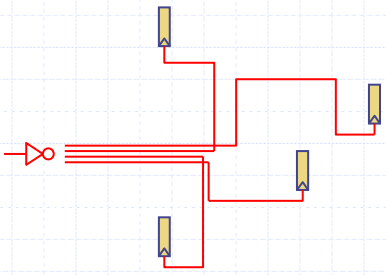
Clock driver tree spans height of chip
Internal levels shorted together

Clock Distribution with Clock Trees

H-Tree



RC-Tree

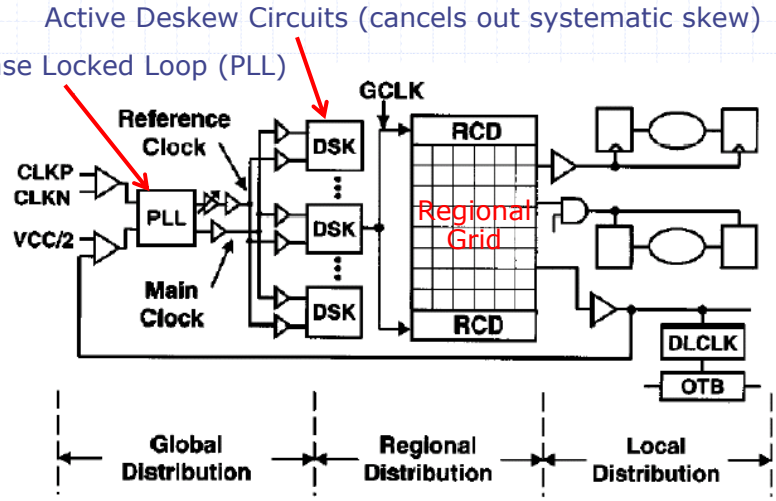


Recursive pattern to distribute signals uniformly with equal delay over area

Each branch is individually routed to balance RC delay

Clock trees have more skew but less power

Clock Distribution Example:
Active deskewing in Intel Itanium



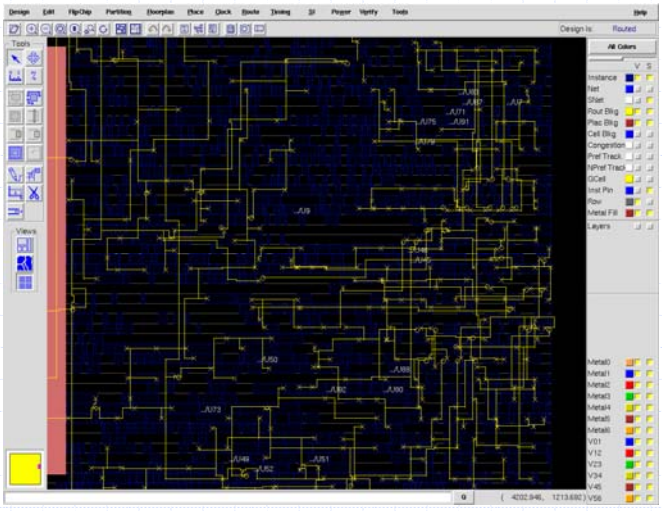
Reducing Clock Distribution Problems

- Use latch-based design
  - Time borrowing helps reduce impact of clock uncertainty
  - Timing analysis is more difficult
  - Rarely used in fully synthesized ASICs, but sometimes in datapaths of otherwise synthesized ASICs
- Make logical partitioning match physical partitioning
  - Limits global communication where skew is usually the worst
  - Helps break distribution problem into smaller subproblems
- Use globally asynchronous, locally synchronous design
  - Divides design into synchronous regions which communicate through asynchronous channels
  - Requires overhead for inter-domain communication
- Use asynchronous design
  - Avoids clocks all together
  - Incurs its own forms of control overhead

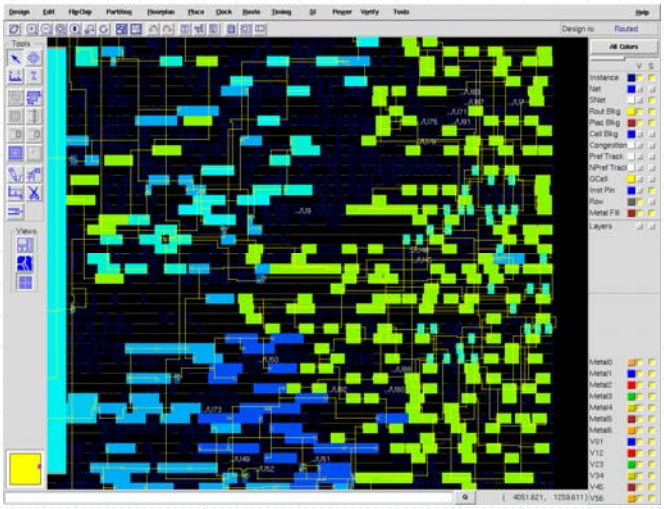
Clock Tree Synthesis for ASICs

- Modern back-end tools include clock tree synthesis
  - Creates balanced RC-trees
  - Uses special clock buffer standard cells
  - Can add clock shielding
  - Can exploit useful clock skew
- Automatic clock tree generation still results in significantly worse clock uncertainties as compare to hand-crafted custom clock trees
  - Modern high-performance processors have clock distribution with <10ps skew at 250ps cycle-time (4GHz)

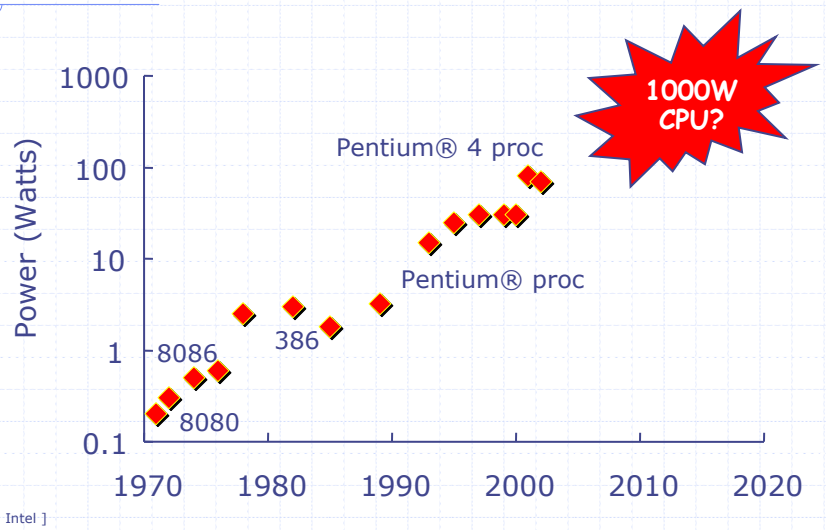
Clock tree synthesis using commercial tools: *an example*



Clock tree synthesis using commercial tools: *an example*



Power has been increasing rapidly



[Source: Intel]

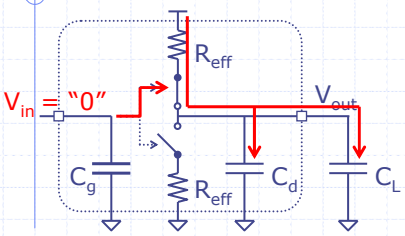
Power Dissipation Problems

- Power dissipation is limiting factor in many systems
  - Battery weight and life for portable devices
  - Packaging and cooling costs for tethered systems
  - Case temperature for laptop/wearable computers
  - Fan noise for media hubs

Cellphone
- 3 Watt total power limit – any more and customers complain
- Battery life/size/weight are strong product differentiators

Internet data center
- ~8,000 servers
- ~2 MegaWatts
- 25% of operational cost is electricity for supplying power and air-conditioning to remove heat

RC model of an Invertor can also be used to understand the energy consumption



Dynamic power

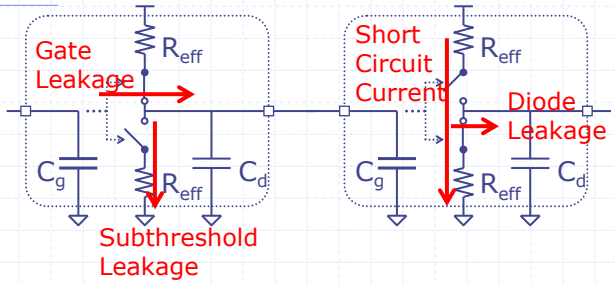
$$E_{0\to1} = \int_0^T P(t)\,dt = V_{DD}\int_0^T I(t)\,dt = V_{DD}\int_0^T \frac{dQ}{dt}dt$$

$$= V_{DD}\int_0^T C\frac{dV}{dt}dt = V_{DD}\int_0^{V_{DD}} (C_d + C_L)dV_{out}$$

$$= (C_d + C_L)V_{DD}^2 = CV_{DD}^2$$

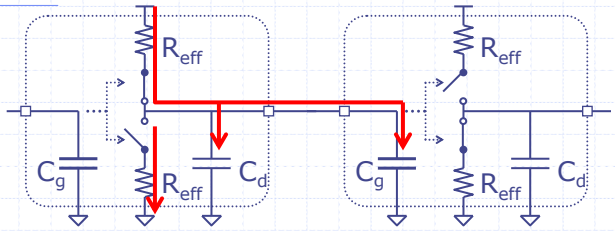
- During 0→1 transition, energy CV_{DD}^2 removed from power supply
- After transition, 1/2 CV_{DD}^2 stored in capacitor, the other 1/2 CV_{DD}^2 was dissipated as heat in pullup resistance
- The 1/2 CV_{DD}^2 energy stored in capacitor is dissipated in the pulldown resistance on next 1→0 transition

Other types of power consumption



Short Circuit Current	Fast edges keep to <10% of cap charging current
Subthreshold Leakage	Approaching 10-40% of active power
Diode Leakage	Usually negligible
Gate Leakage	Was negligible, increasing due to thin gate oxides

Dynamic and Static power



Dynamic Power

Switching power used to charge up load capacitance

$P_{dynamic} = \alpha \; f \; (1/2) \; C \; V_{DD}^2$

Activity Factor (transitions/cycle)

Clock Frequency

Static Power

Subthreshold leakage power when transistor is "off"

$P_{static} = V_{DD} \; I_{off}$

Reducing Dynamic Power (1)

$$P_{dynamic} = \alpha \; f \; (1/2) \; C \; V_{DD}^2$$

Reduce Activity

- Clock gating so clock node of inactive logic doesn't switch
- Data gating so data nodes of inactive logic doesn't switch
- Bus encodings to minimize transitions
- Balance logic paths to avoid glitches during settling

Reduce Frequency

- Doesn't save energy, just reduces rate at which it is consumed
- Lower power means less heat dissipation but must run longer

Reducing Dynamic Power (2)

$$P_{dynamic} = \alpha \; f \; (1/2) \; C \; V_{DD}^2$$

Reduce Switched Capacitance

- Careful transistor sizing (small transistors off critical path)
- Tighter layout (good floorplanning)
- Segmented bus/mux structures

Reduce Supply Voltage

- Need to lower frequency as well – quadratic+ power savings
- Can lower statically for cells off critical path
- Can lower dynamically for just-in-time computation

Reducing Static Power

$$P_{static} = V_{DD}\, I_{OFF}$$

Reduce Supply Voltage

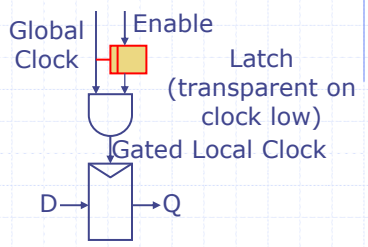
- In addition to dynamic power reduction, reducing Vdd can help reduce static power

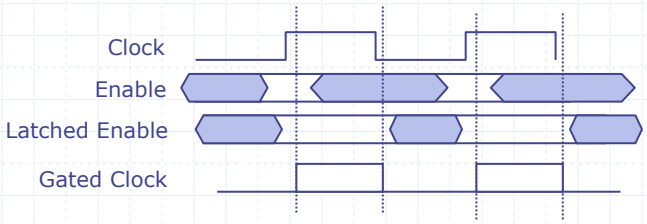
Reduce Off Current

- Increase length of transistors off critical path
- Use high-Vt cells off critical path (extra Vt increases fab costs)
- Use stacked devices (complex gates)
- Use power gating (i.e. switch off power supply with large transistor)

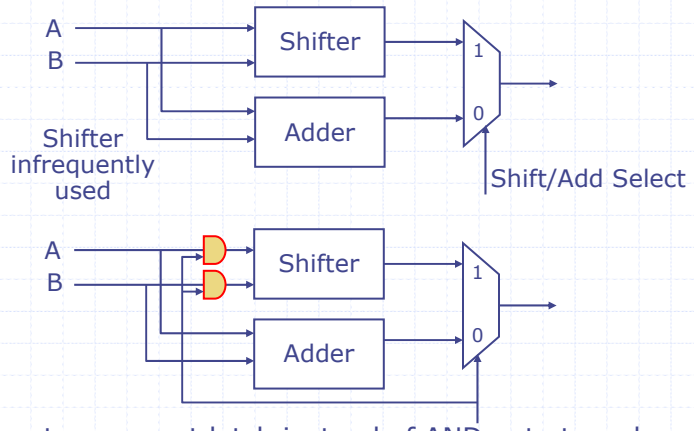
Clock gating

- Don't clock flip-flop if not needed
- Avoids transitioning downstream logic
- Enable adds control logic complexity
- Pentium-4 has hundreds of gated clock domains



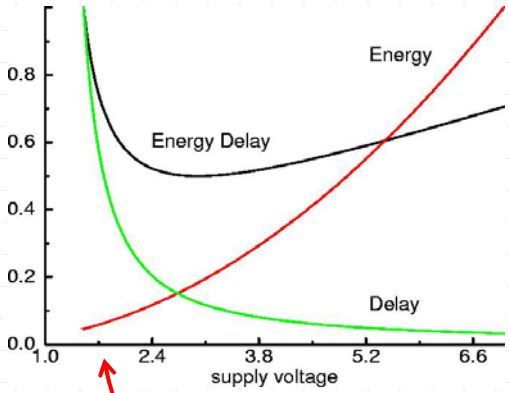


Data gating



Could use transparent latch instead of AND gate to reduce number of transitions, but would be bigger and slower.

Voltage Scaling to trade Energy for Delay



Both static and dynamic voltage scaling is possible

Delay rises sharply as supply voltage approaches Vt

[Source: Horowitz]

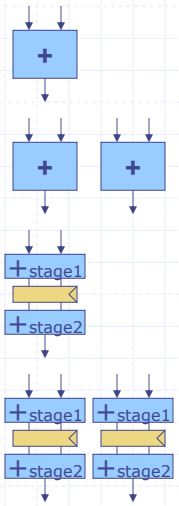
Parallelism Reduces Energy

8-bit adder/compare
- 40MHz at 5V, area = 530 $k\mu^2$
- Base power P_{ref}

Two parallel interleaved adder/cmp units
- 20MHz at 2.9V, area = 1,800 $k\mu^2$ (3.4x)
- Power = 0.36 P_{ref}

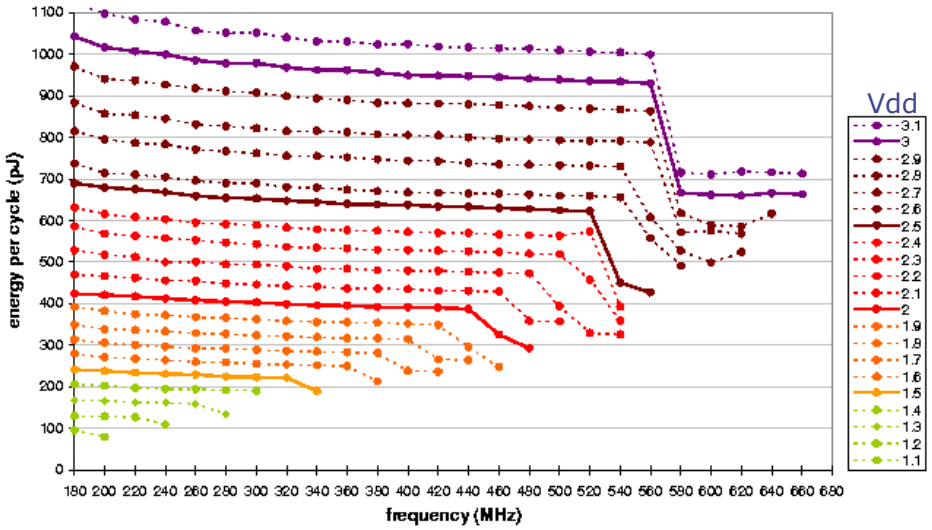
One pipelined adder/cmp unit
- 40MHz at 2.9V, area = 690 $k\mu^2$ (1.3x)
- Power = 0.39 P_{ref}

Pipelined and parallel
- 20MHz at 2.0V, area = 1,961 $k\mu^2$ (3.7x)
- Power = 0.2 P_{ref}



Chandrakasan et al, IEEE JSSC 27(4), April 1992

Voltage Scaling Example



[STC1 32-bit RISC Processor + SRAM in TSMC 180nm ASIC process]

Reducing Power in ASIC Designs (1)

- Minimize activity
  - Automatic clock gating is possible if tools can infer gating from HDL
  - Partition designs so minimal number of components activated to perform each operation
- Use lowest voltage and slowest frequency necessary to reach target performance
  - Use pipelined and parallel architectures if possible

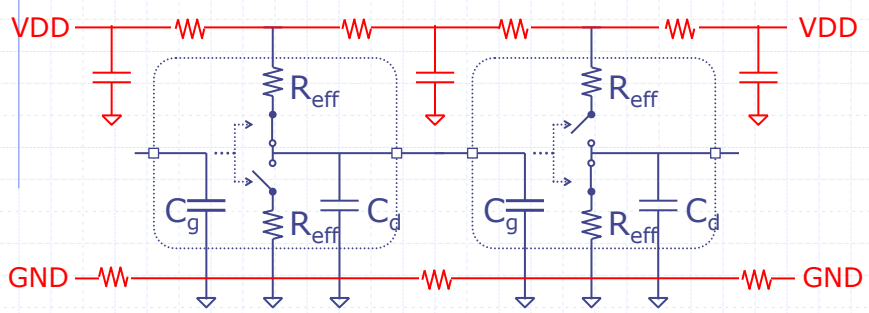
Reducing Power in ASIC Designs (2)

- Reducing switched capacitance
  - Design efficient RTL! Biggest savings come from picking better hardware algorithms to reduce power and area
  - Floorplan units to reduce length of power-hungry global wires
- Optimizing for static power
  - Reduce amount of logic required for function, multiplex units
  - Partition design such that components can be power-gated or have independent voltage supplies
  - Modern standard cell libraries include low-power cells, high-VT cells, and low-VT cells – tools can automatically replace non-critical cells to optimize for static power

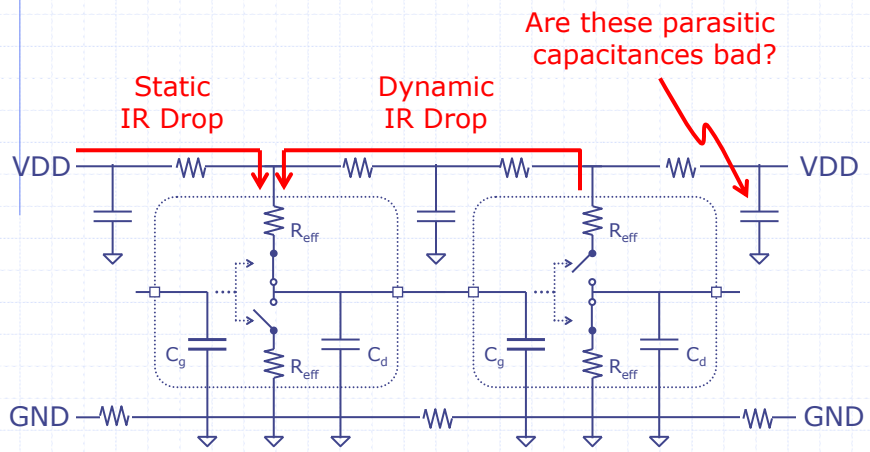
Power Distribution

Power Distribution

Possible IR drop across power network

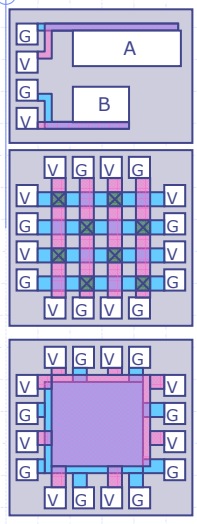


IR drop can be static or dynamic



Power Distribution-Custom Approach:

Carefully tailor power network



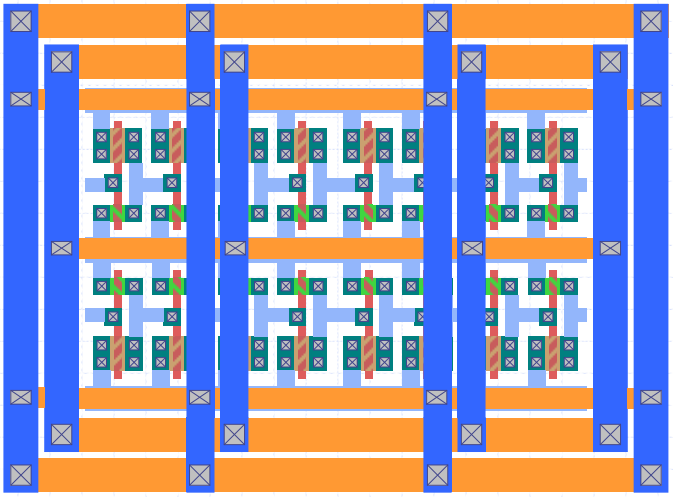
Routed power distribution on two stacked layers of metal (one for VDD, one for GND). OK for low-cost, low-power designs with few layers of metal.

Power Grid. Interconnected vertical and horizontal power bars. Common on most high-performance designs. Often well over half of total metal on upper thicker layers used for VDD/GND.

Dedicated VDD/GND planes. Very expensive. Only used on Alpha 21264. Simplified circuit analysis. Dropped on subsequent Alphas.

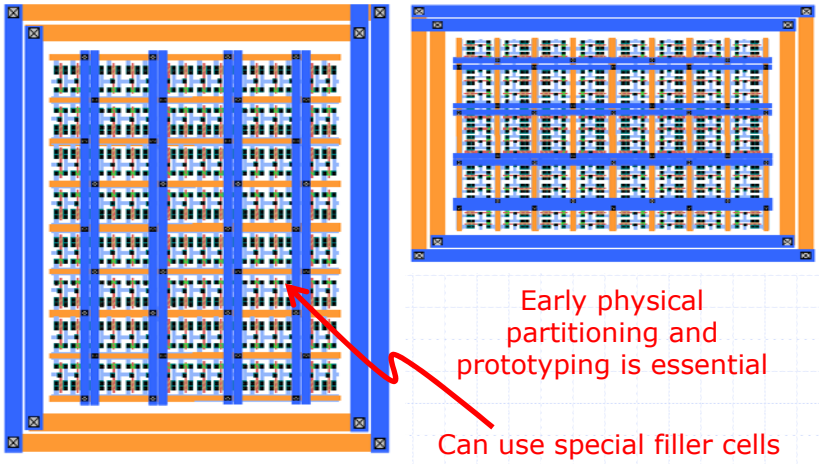
Power Distribution-ASIC Approach:

Strapping & rings for standard cells



Power Distribution- ASIC Approach:

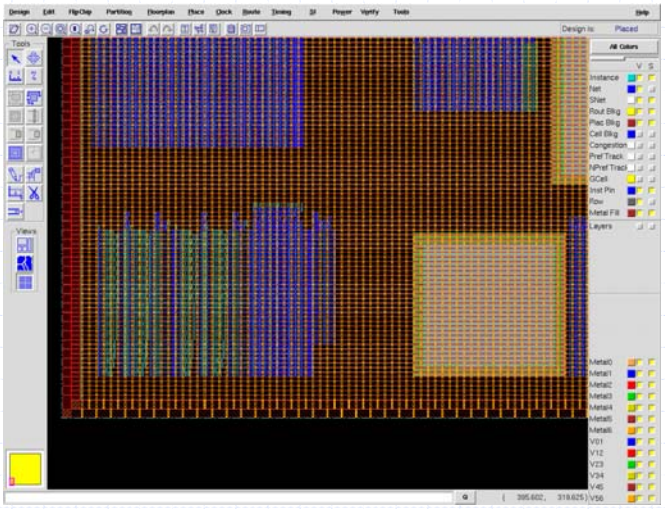
Power rings partition the power problem



Early physical partitioning and prototyping is essential

Can use special filler cells to help add decoupling cap

Example: Power distribution network using commercial tools



Example: Power distribution network using commercial tools