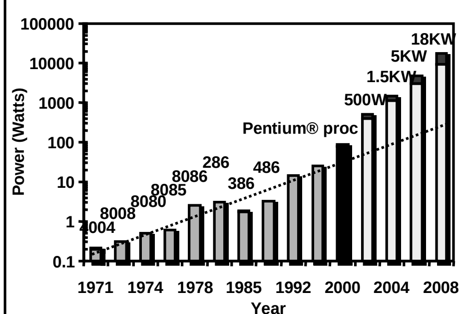
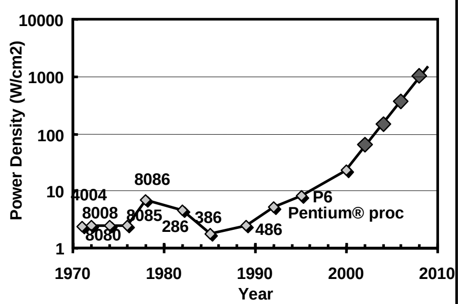
L18: Power Dissipation in Digital Systems

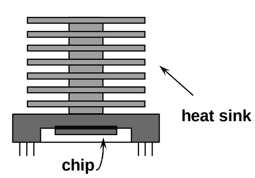
Problem #1: Power Dissipation/Heat



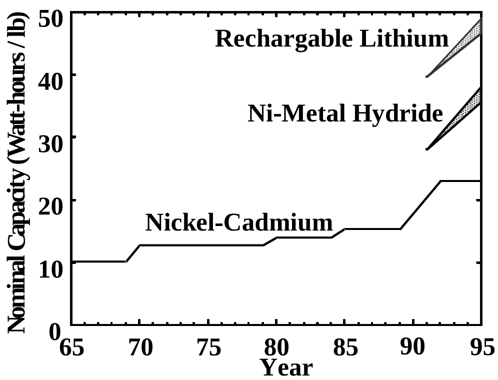


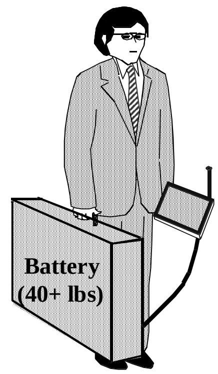
Courtesy Intel (S. Borkar)

How do you cool these chips??



Problem #2: Energy Consumption

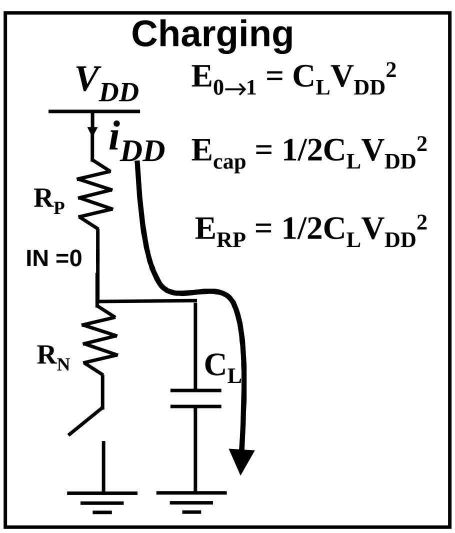


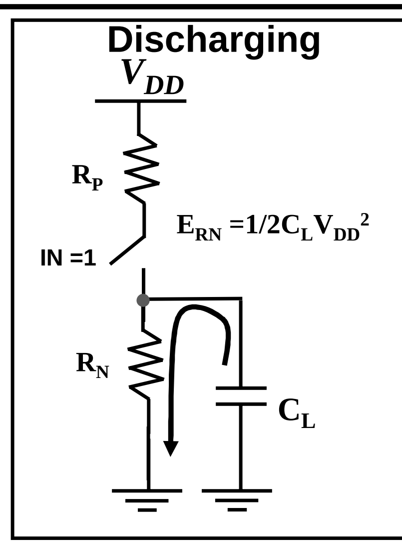


(from Jon Eager, Gates Inc. , S. Watanabe, Sony Inc.)

No Moore's law for batteries…
Today: Understand where power goes and ways to manage it

Dynamic Energy Dissipation

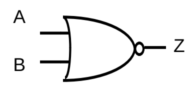




$$P = C_L V_{DD}^2 f_{clk}$$

The Transition Activity Factor $\alpha_{0\to1}$

Current Input	Next Input	Output Transition
00	00	1 –> 1
00	01	1 –> 1
00	10	1 –> 1
00	11	1 –> 0
01	00	1 –> 1
01	01	1 –> 1
01	10	1 –> 1
01	11	1 –> 0
10	00	1 –> 1
10	01	1 –> 1
10	10	1 –> 1
10	11	1 –> 0
11	00	0 –> 1
11	01	0 –> 1
11	10	0 –> 1
11	11	0 –> 0



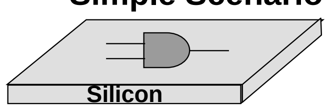
Assume inputs (A,B) arrive at *f* and are uniformly distributed
What is the average power dissipation?

$$\alpha_{0\to1} = \_\_\_\_\_\_$$

$$P = \alpha_{0\to1} C_L V_{DD}^2 f$$

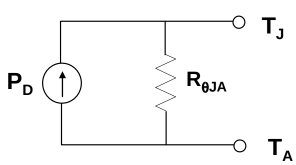
Junction (Silicon) Temperature

Simple Scenario



$$T_j - T_a = R_{\theta JA} P_D$$

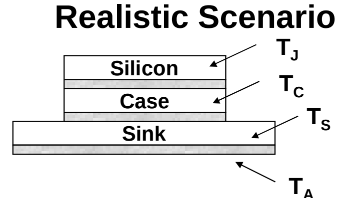
$R_{\theta JA}$ is the thermal resistance between silicon and Ambient

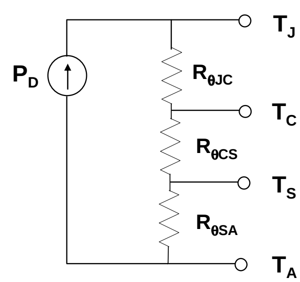


$$T_j = T_a + R_{\theta JA} P_D$$

Make this as low as possible

Realistic Scenario

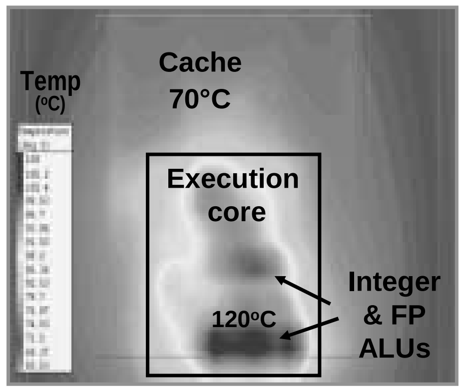




$$R_{\theta CA} = R_{\theta CS} + R_{\theta SA}$$

is minimized by facilitating heat transfer (bolt case to extended metal surface – heat sink)

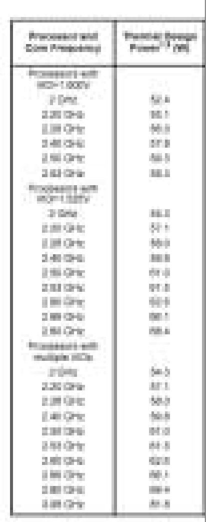
Large Thermal Gradients

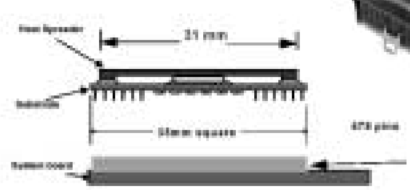


Courtesy of Intel (Ram Krishnamurthy)

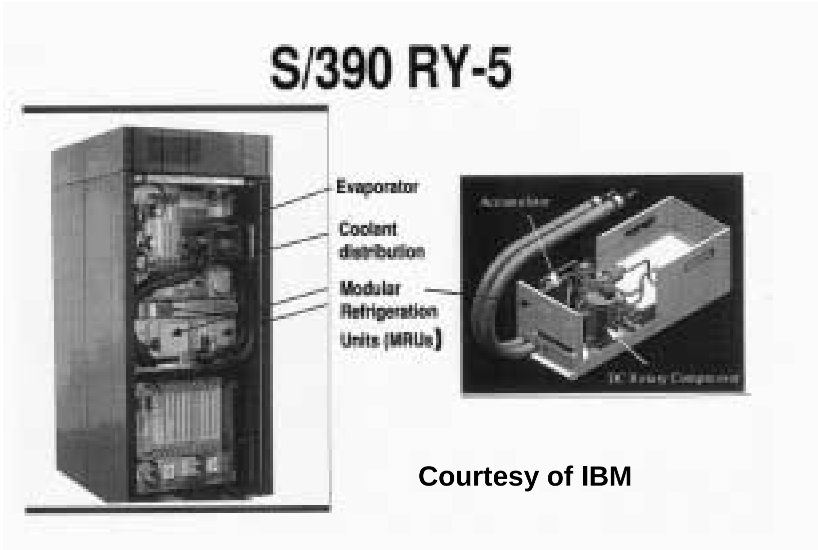
Intel Pentium 4 Thermal Guidelines

- Pentium 4 @ 3.06 GHz dissipates 81.8W!
- Maximum T_C = 69 °C
- R_{CA} < 0.23 °C/W for 50 C ambient
- Typical chips dissipate 0.5-1W (cheap packages without forced air cooling)





Refrigerator Inside!



Courtesy of IBM

Power Reduction Strategies

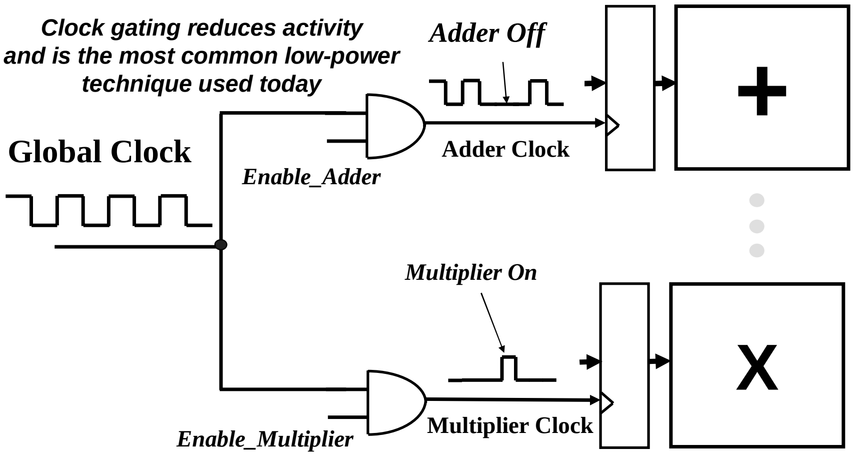
$$P = \alpha_{0\to1} C_L V_{DD}^2 f$$

- Reduce Transition Activity or Switching Events
- Reduce Capacitance (e.g., keep wires short)
- Reduce Power Supply Voltage
- Frequency is typically fixed by the application, though this can be adjusted to control power

Optimize at all levels of design hierarchy

Clock Gating is a Good Idea!

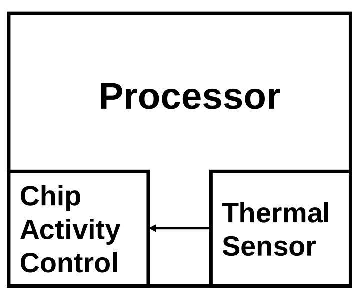
Clock gating reduces activity and is the most common low-power technique used today



100's of different clocks in a microprocessor

Clock Gating Reduces Energy, does it reduce Power?

Does your GHz Processor run at a GHz?

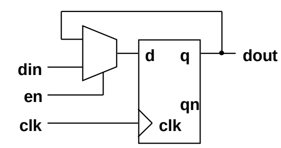


- Note that there is a difference between average and peak power
- On-chip thermal sensor (diode based), measures the silicon temperature
- If the silicon junction gets too hot (say 125 °C), then the activity is reduced (e.g., reduce clock rate or use clock gating)

Use of Thermal Feedback

ASIC Support for Clock Gating

- **Objective**
  - Reduce/inhibit unnecessary clocking
  - Registers need not be clocked if the data input hasn't changed
- **Data Gating**



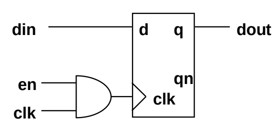
- Conventional Code

```
If rising_edge (clk) then
    q <= din;
End if;
```

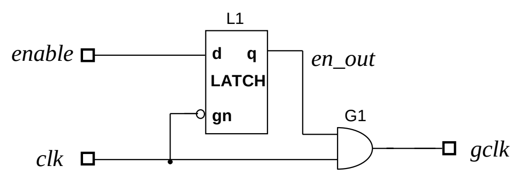
- Low Power Code

```
// only transfer din to q when enable is true
gclk <= enable and clk; // gate the clock
if rising_edge (gclk) then
    q <= din;
end if;
```

- **Clock Gating**



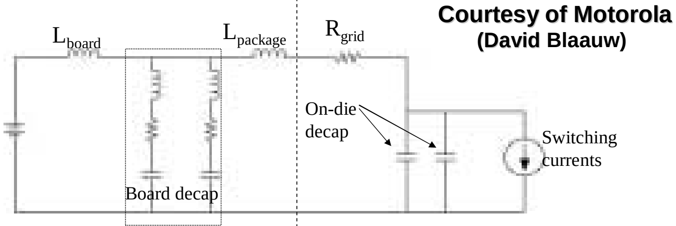
- Can also add a Latch to Prevent Clock Glitching

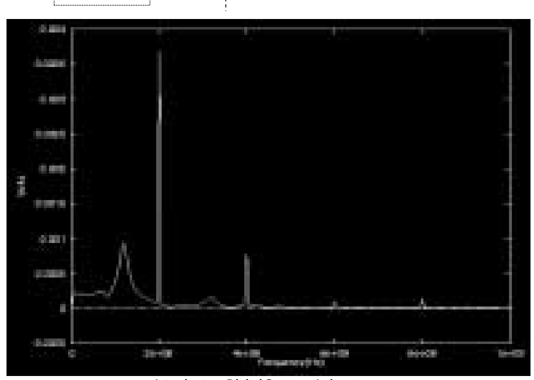


Courtesy of Sequence Design (J. Frenkil)

Power Supply Parasitics

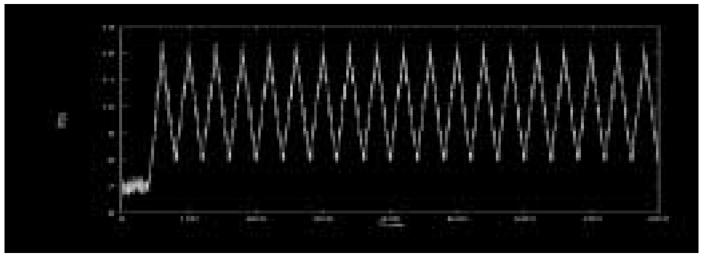
Courtesy of Motorola (David Blaauw)

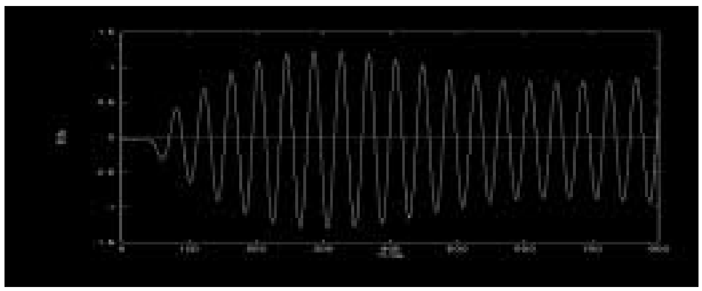




200Mhz Design

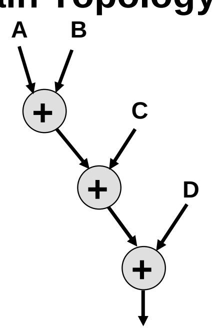
Can write a Virus to Activate Power Supply Resonance!





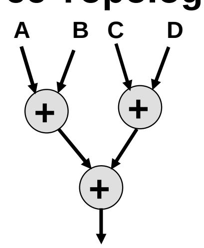
Glitching Transitions

Chain Topology



(((A+B) + C)+D)

Tree Topology

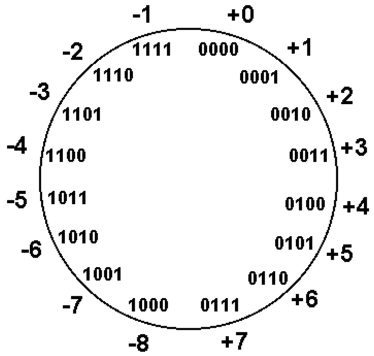


(A+B) + (C+D)

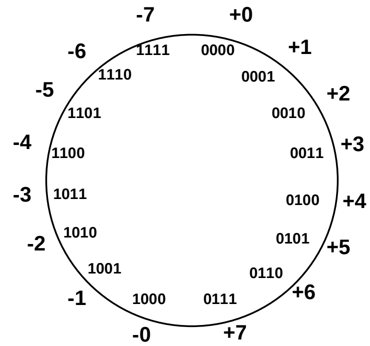
- Balancing paths reduces glitching transitions
- For 4 inputs, 50% less transitions using a tree approach
- Structures such as multipliers have lot of glitching transitions
- Keeping logic depths short (e.g., pipelining) reduces glitching

Number Representation: Two's Complement vs. Sign Magnitude

Two's complement

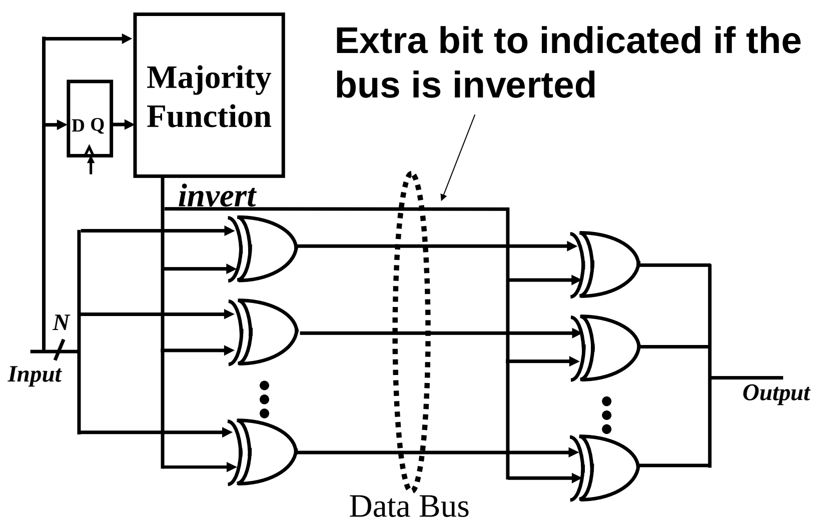


Sign-Magnitude



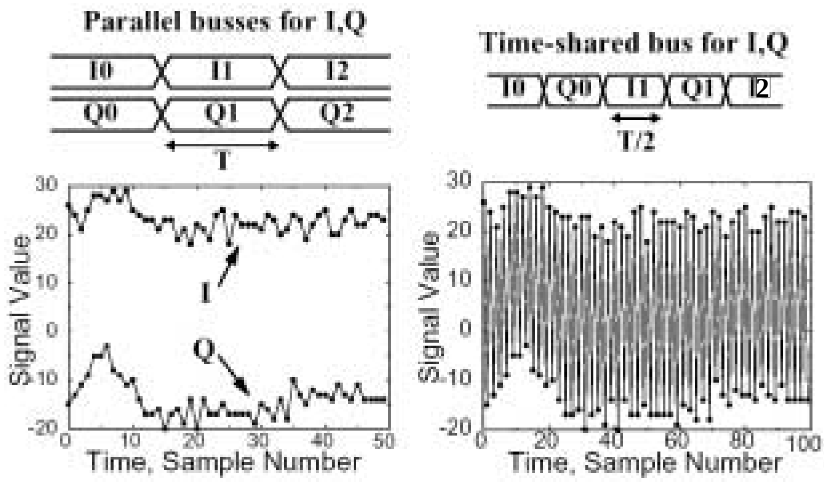
Consider a 16 bit bus where inputs toggles between +1 and –1 (i.e., a small noise input) Which representation is more energy efficient?

Bus Coding to Reduce Activity



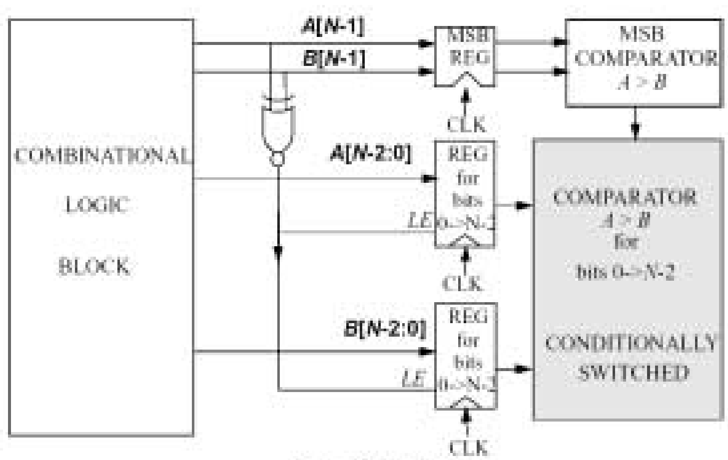
[Stan94]

Time Sharing is a Bad Idea



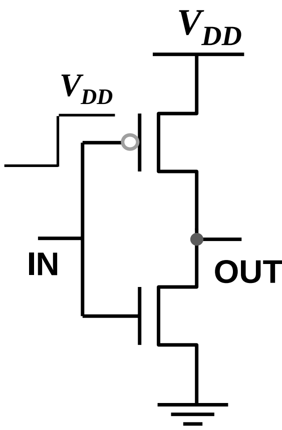
Time Sharing Increases Switching Activity

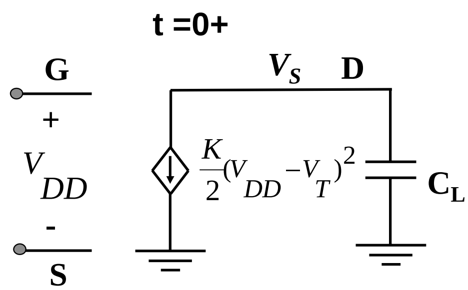
Pre-Computation



from [Alidina94]
(*1994 International Workshop on Low-power Design*)

Reduce Supply Voltage : But is it Free?

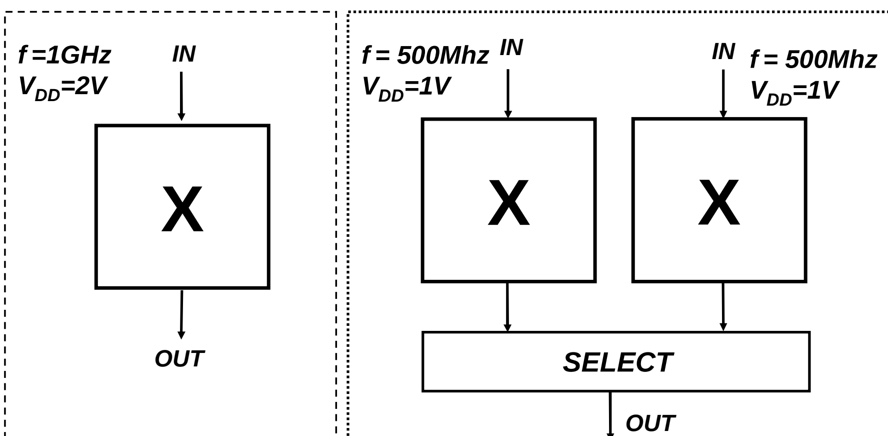




$$Delay = \frac{C_L \cdot \Delta V}{i_D} = \frac{C_L \cdot \frac{V_{DD}}{2}}{\frac{k}{2}(V_{DD} - V_T)^2} \propto \frac{V_{DD}}{(V_{DD} - V_T)^2} \approx \frac{1}{V_{DD}}$$

V_{DD} from 2V to 1V, energy ↓ by x4, delay ↑ x2

Transistors Are Free…
(What do you do with a Billion Transistors?)



$P_{serial} = C_{mult}\, 2^2\, f$

$P_{parallel} = (2C_{mult}\, 1^2\, f /2) = P_{serial}/4$

Trade Area for Low Power

Dynamic Voltage Scaling

Fixed Power Supply

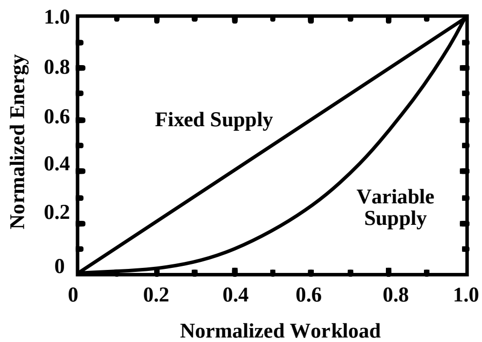
ACTIVE	IDLE

$E_{FIXED} = ½ \, C \, V_{DD}^2$

Variable Power Supply

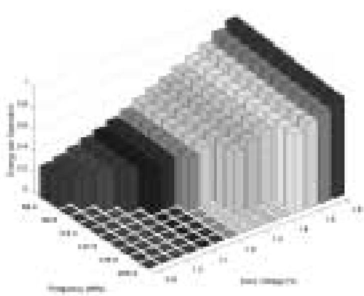
ACTIVE

$E_{VARIABLE} = ½ \, C \, (V_{DD}/2)^2 = E_{FIXED} / 4$

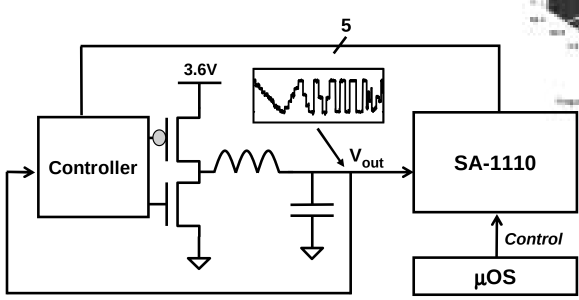


[Gutnik97]

DVS on a Processor

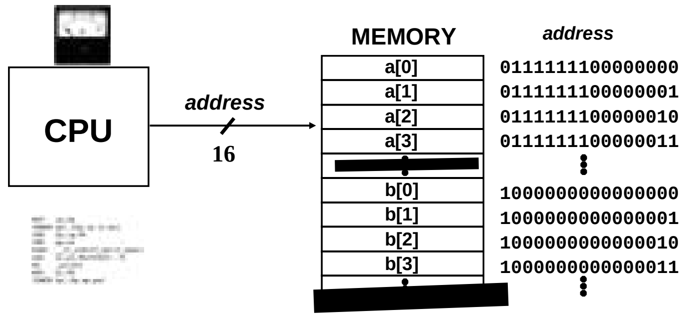


Digitally adjustable DC-DC converter powers SA-1110 core



µOS selects appropriate clock frequency based on workload and latency constraints

Not just a 6-1 Issue: “Cool” Software ???




```
float a [256], b[256];
float pi= 3.14;

for (i = 0; i < 255; i++) {
   a[i] = sin(pi * i /256);
   b[i] = cos(pi * i /256);
}
```

512(8)+2+4+8+16+32+64+128+256
= 4607 bit transitions

```
float a [256], b[256];
float pi= 3.14;

for (i = 0; i < 255; i++) {a[i] = sin(pi * i /256);}
for (i = 0; i < 255; i++) {b[i] = cos(pi * i /256);}
```

2(8)+2(2+4+8+16+32+64+128+256)
= 1030 transitions

Energy Scavenging

MEMS Generator

**Jose Mur Miranda/
Jeff Lang**

Vibration-to-Electric Conversion

~ 10µW

Power Harvesting Shoes

**Joe Paradiso
(Media Lab)**

After 3 6 steps, it provides 3 mA for 0.5 sec

~10mW