Lecture 15

6.205 Fall 2024

Power and related concepts

Last Lecture

Administrative I

- This is the last lecture of the semester
- Last Lab/Week 7 is due tonight.
- Make sure you have all labs complete (all checkoffs completed) by November 8.
  - Remember you need to have all labs completed to pass the class.

Administrative II

- Final Project Abstract Feedback was sent earlier in the week.
- Project Block Diagram Report is due on October 29th (Tuesday)
- We'll continue to have office hours
- Presentation Timeslots out in next 24 hours-ish for the week of November 4 through 8

Administrative lll

- If you need parts early in project, parts should be ordered. Email me (Joe). We want to make sure the cost is reasonable.

Power

While we're not really focusing on this in final projects, maybe think about this as another way to characterize your device's performance

Things To Talk About

- What is power?
- Why does it matter?
- Why do we use it?
- How do we make it?
- How to measure it?

All Computation Requires Power

- Power is related to energy
- All computation uses energy
- For a given computational technology…
  - The more computation you do, the more energy you use
  - The faster you do your computation, the more energy used per unit time, the more *power* your system uses
- We always want more computation and we want that computation faster, so we are constantly using more power…
- The implications of this can vary…

In Stationary Situations…

- (Such as desktop computer, server farms, stationary equipment connected to grid, etc…)

- The tendency to use more and more power means you'll:
  - Use more energy and therefore cost more to operate
  - May have to deal with waste heat disposal

In Mobile Situations

- Power is sourced locally and may be finite.

- The tendency to use more and more power means
  - your system might not last as long if on battery
  - Needs to harvest energy from the environment

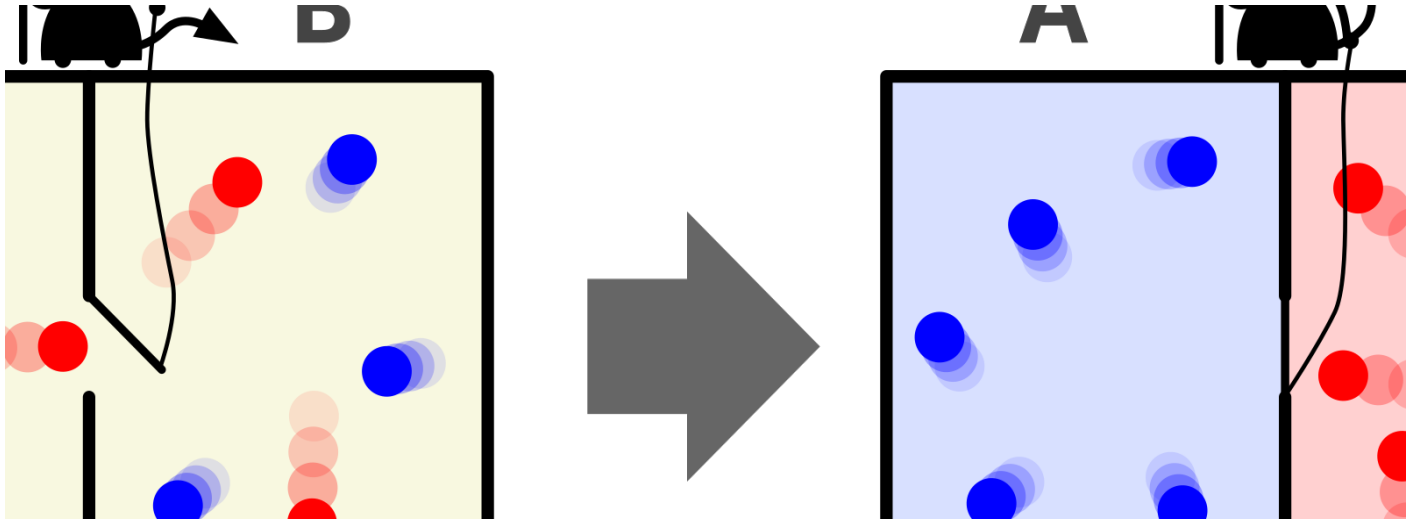
Maxwell's Demon

- Mid-nineteenth thought experiment that appears to violated the **Second Law of Thermodynamics**
- *The entropy of a closed system (meaning it doesn't exchange matter or energy with its surroundings) may never decrease*
- *In other words, order will never appear out of disorder…*
- *A closed system will never become less disordered*

https://en.wikipedia.org/wiki/Maxwell%27s_demon

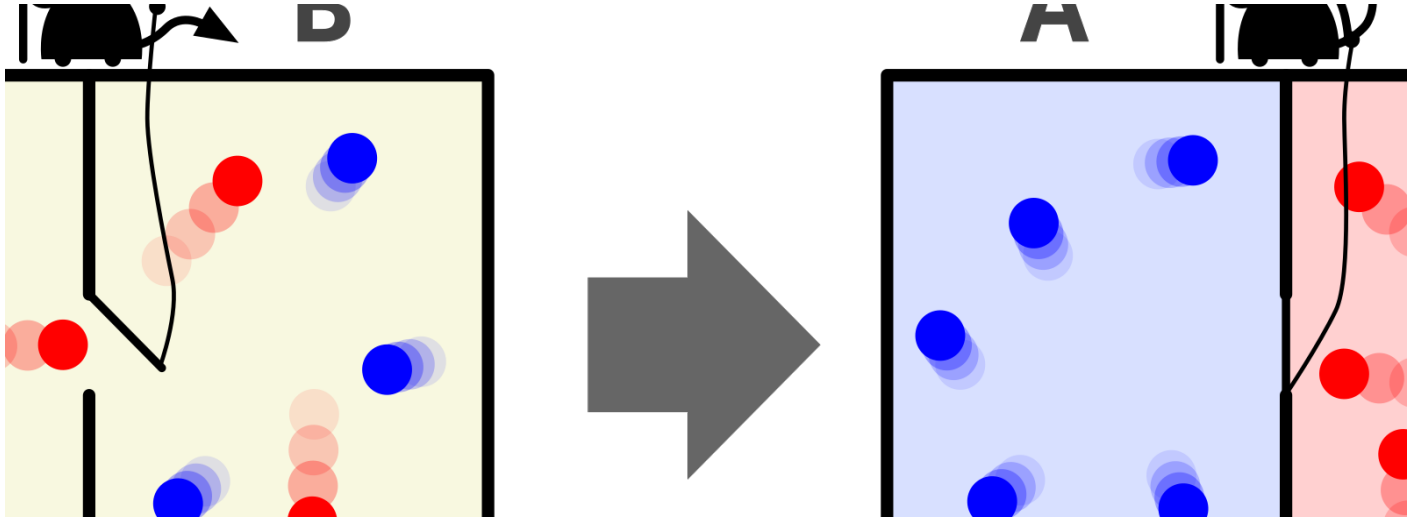
Maxwell's Demon

- Closed perfect box with two halves and door starts out with blue and red perfectly elastic balls bouncing around mixed up.
- Demon watches room and strategically opens/closes door such that over time you order balls in each half



https://en.wikipedia.org/wiki/Maxwell%27s_demon

Maxwell's Demon



- This is one of many ways to see that *information* must be related to energy

https://en.wikipedia.org/wiki/Maxwell%27s_demon

What is Information?

- Anything that provides an answer to a question of some kind
- Anything that can resolve uncertainty…
- And the more uncertainty you resolve, the more information you are providing

- Fundamental unit is the **bit**: one yes/no amount of information

Lower Limit on Computation

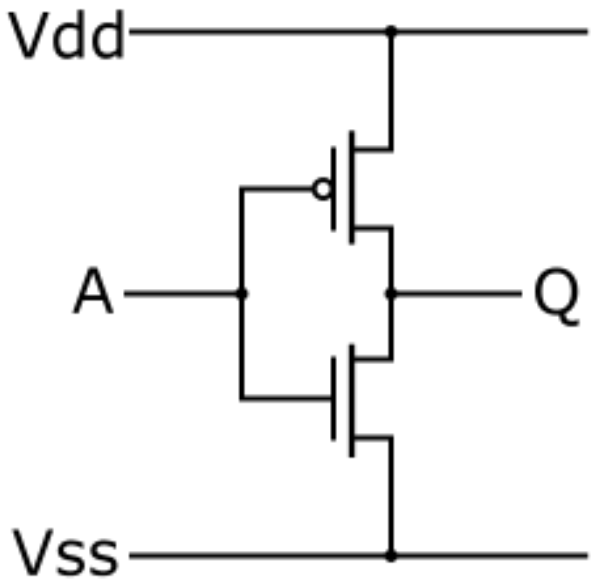
- Somewhat controversial…
- There is a lower limit: it takes about **$\mathbf{3\times10^{-21}}$ Joules** to flip a bit no matter what
  - Called Landauer Limit
  - Experimentally shown in 2012 (Berut et al., Nature 2012) and in more recent experiments
- Intel 22nm process takes approximately
  - **$\mathbf{100\times10^{-15}}$ Joules (estimate/approximation)**
- *Between those two numbers are the inefficiencies and limitations of circuits*
- Very relevant topic as we approach quantum computation

https://spectrum.ieee.org/computing/hardware/landauer-limit-demonstrated
https://en.wikipedia.org/wiki/Landauer%27s_principle

What actually uses the power in Modern Digital Circuits?

- All modern digital electronics mostly use CMOS architectures:
  - **C**omplementary **M**etal **O**xide **S**emiconductor (Field Effect Transistors)

CMOS NOT gate:

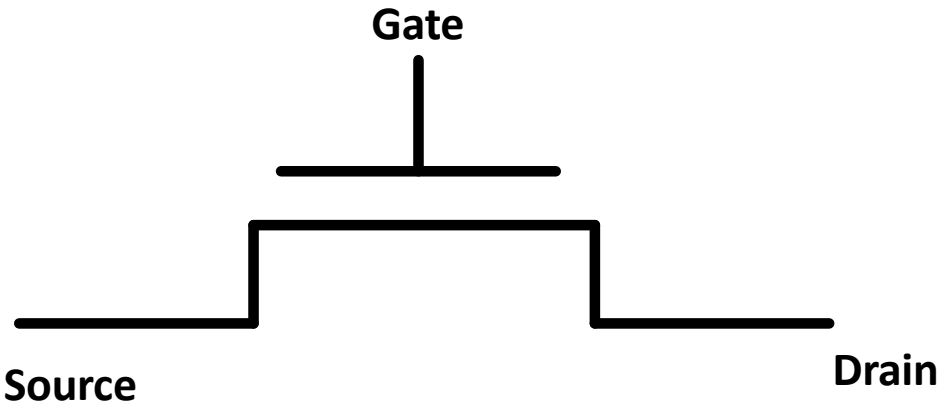


- Complementary nature means almost no current flowing (no power) at rest
- Power only* expended on the switch since we need to charge up any MOSFETs that we're driving with our output

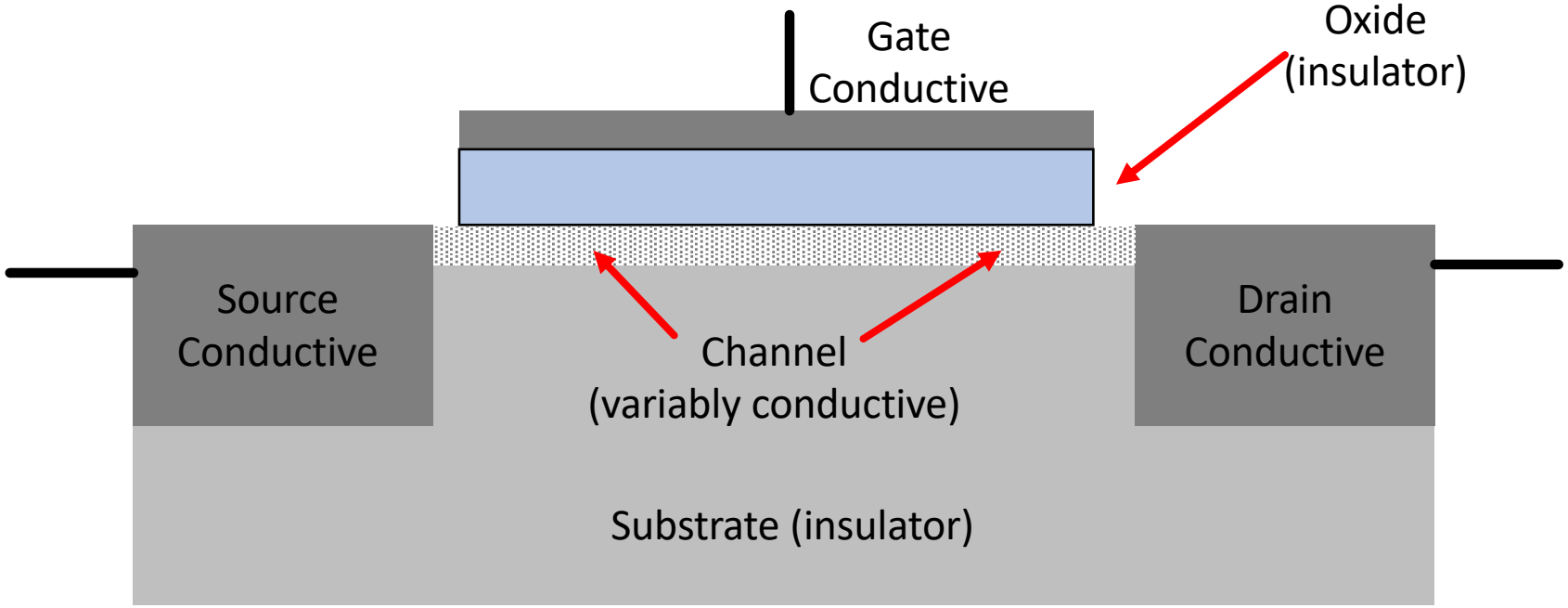
*ideally

A MOSFET

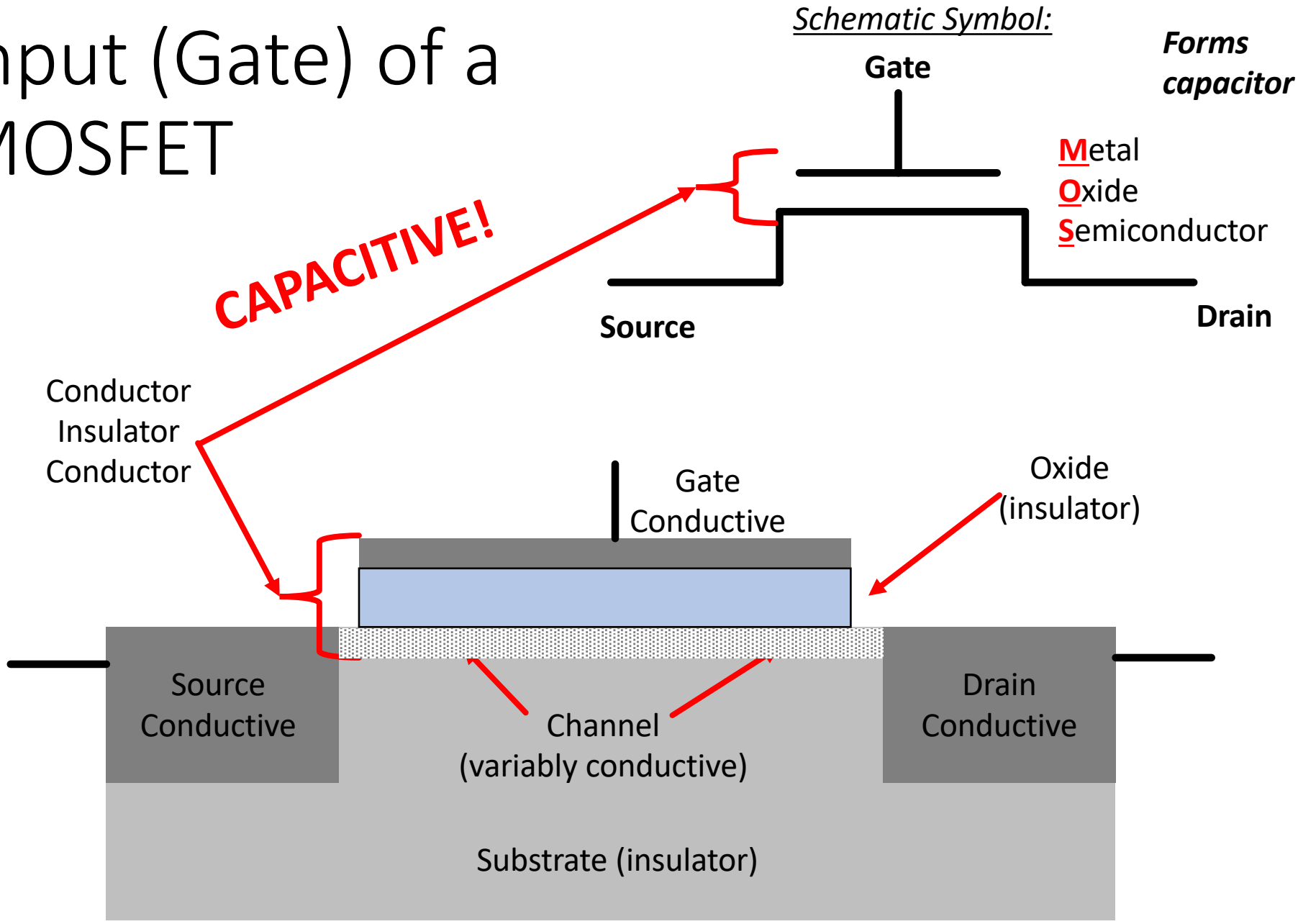
Schematic Symbol:



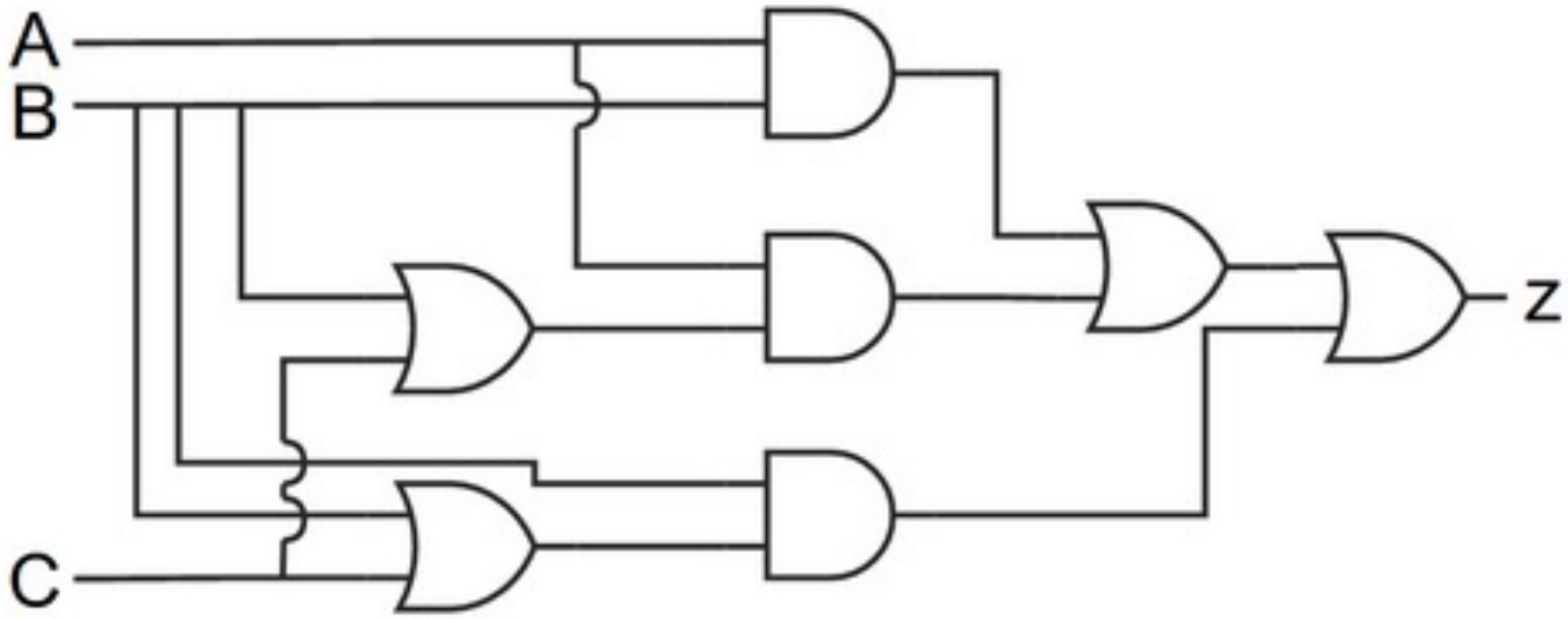
Physical Design:



Input (Gate) of a MOSFET

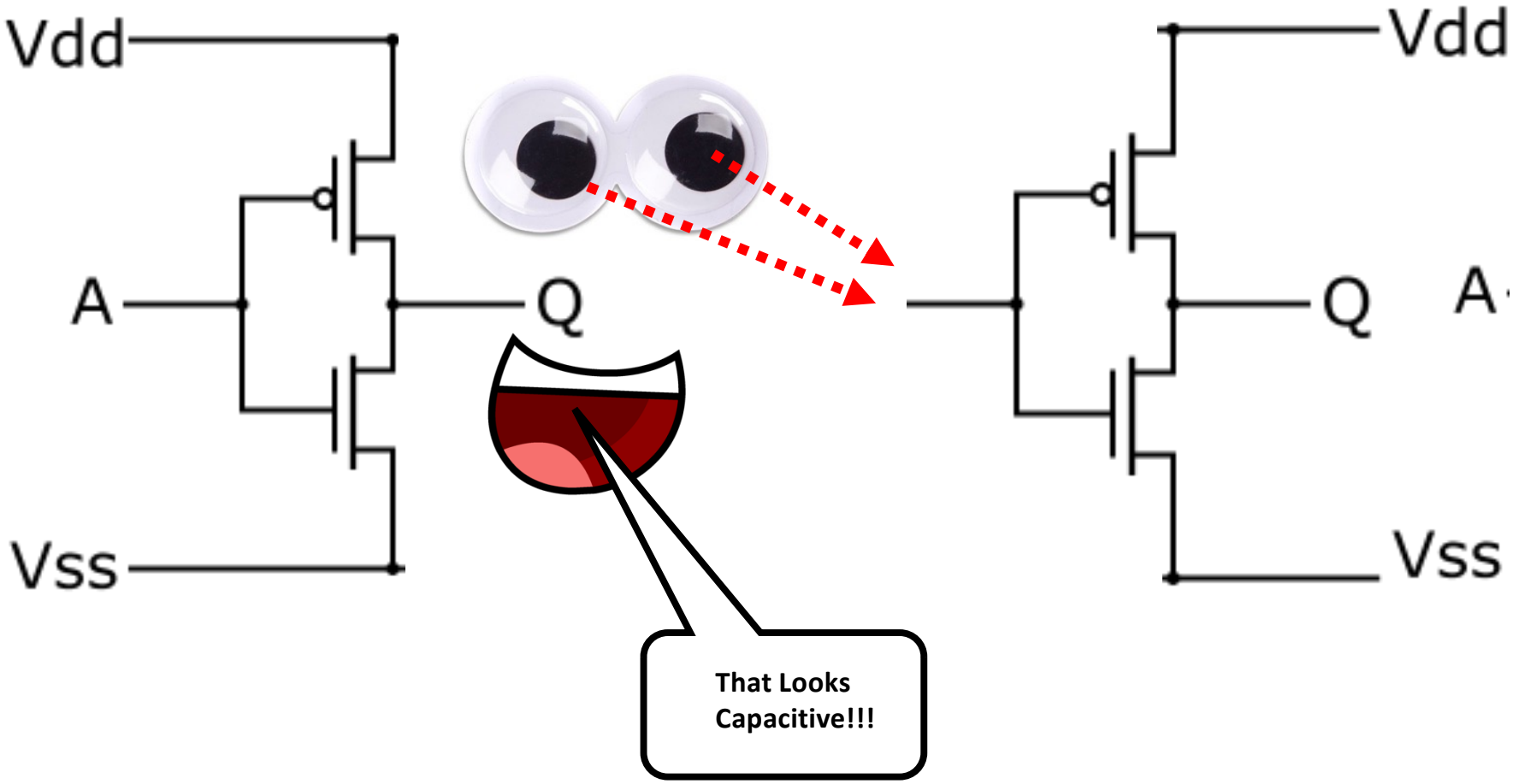


So when we start to build complicated digital circuits…



CMOS Circuit drives another CMOS Circuit…

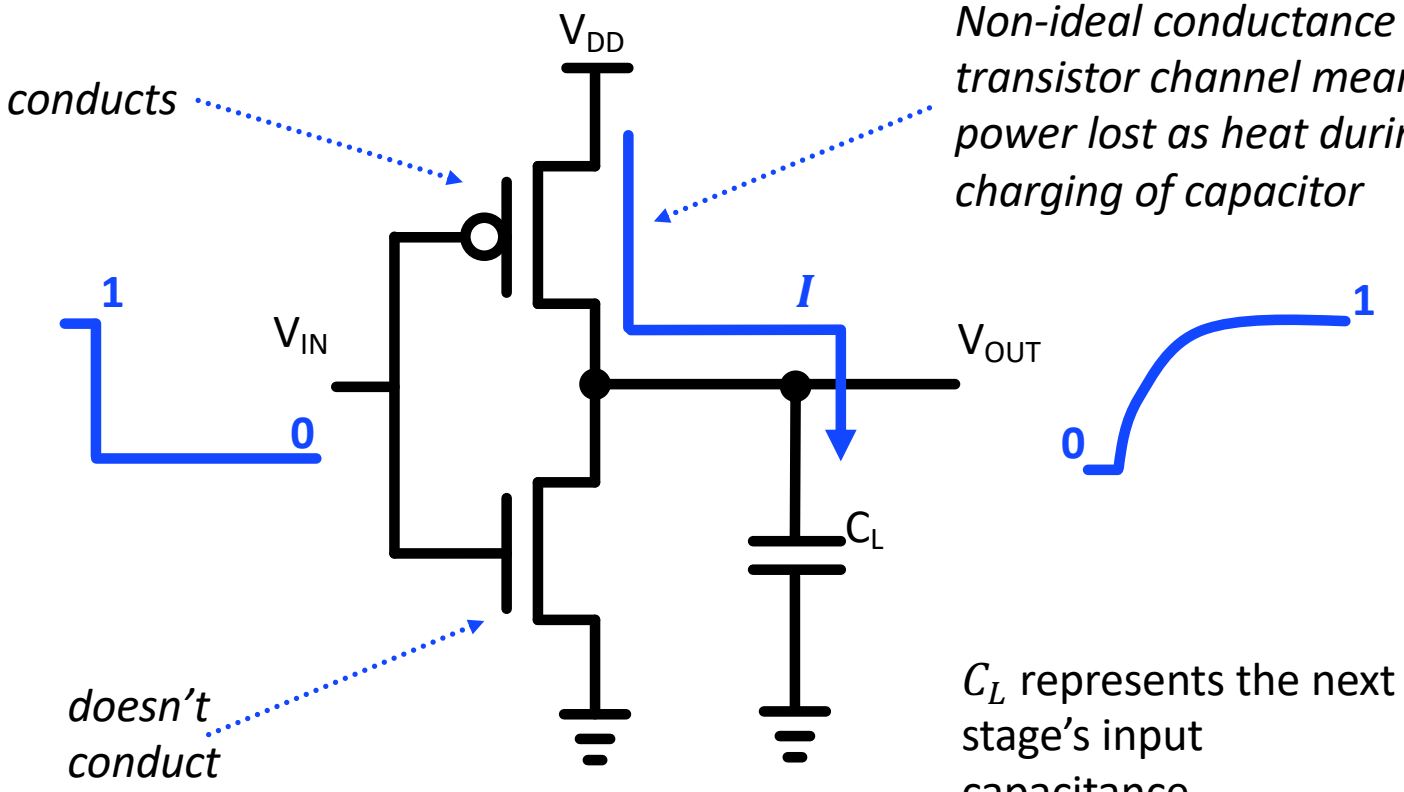
It is CMOS all the way down



Implications

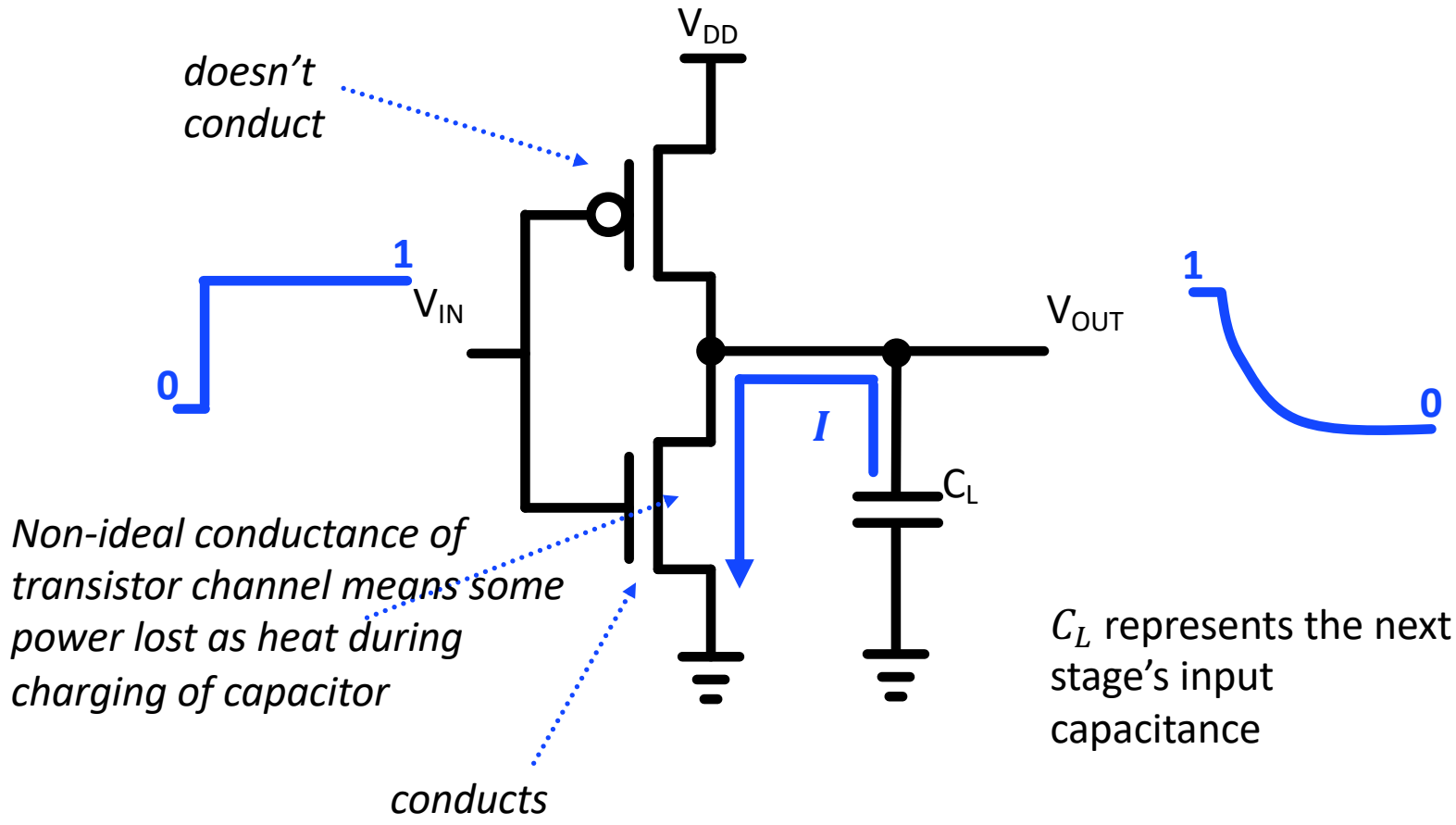
- If a circuit has to charge or discharge up a capacitive input that means…:
- There is an inherent speed limit from RC time constants that will limit the switching speed
- The fact that you have to move charge onto that capacitor means energy is needed…and that energy will never get recovered.

Takes energy to charge up capacitors (Dynamic Power Consumption)



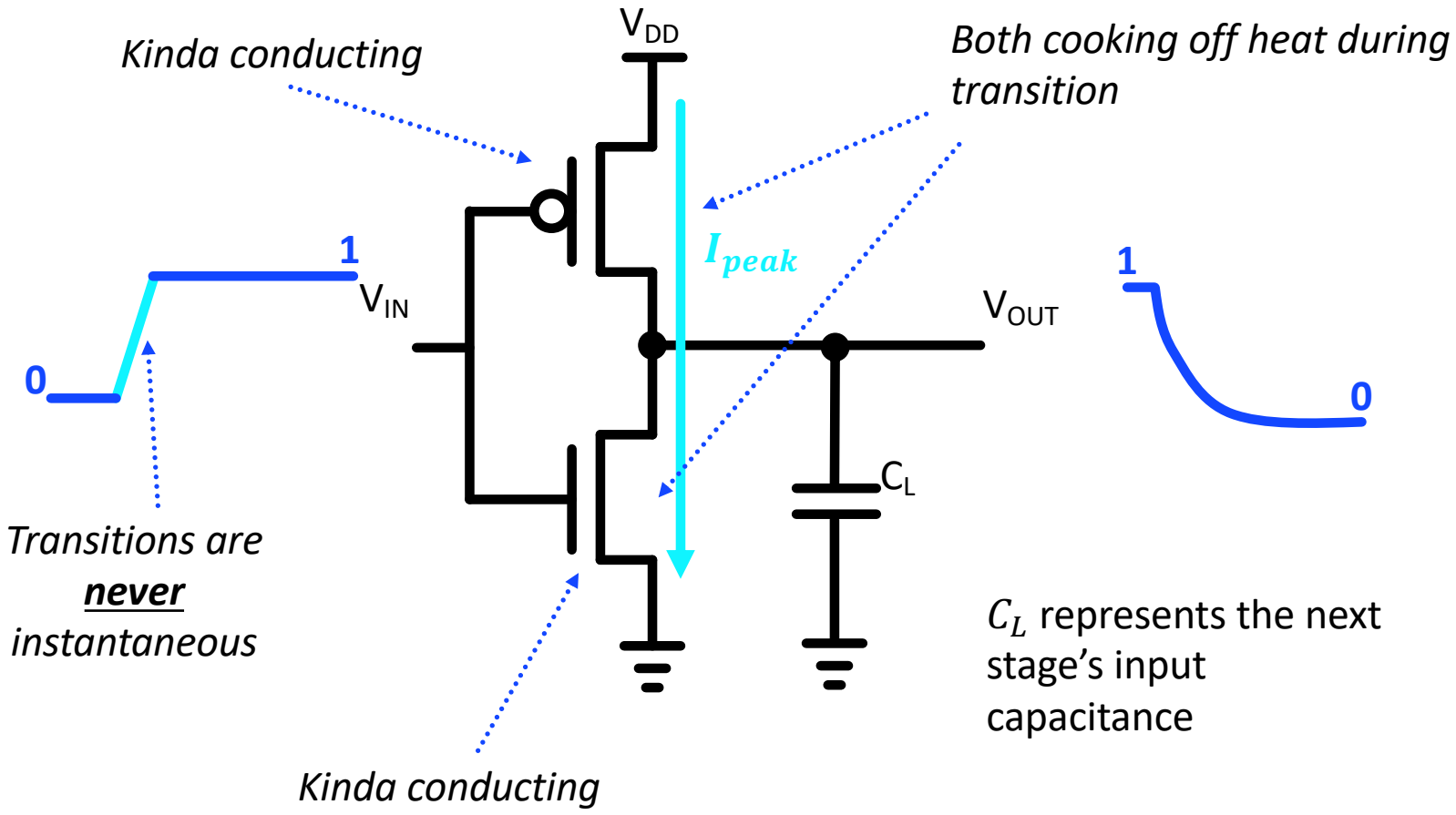
http://web.eecs.umich.edu/~twenisch/470_F07/lectures/21.pdf GO BLUE

Takes energy to charge *down* capacitors (Dynamic Power Consumption)



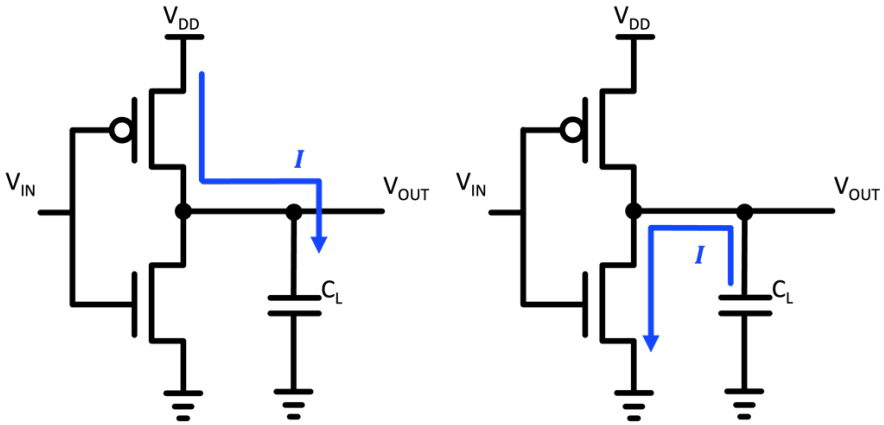
http://web.eecs.umich.edu/~twenisch/470_F07/lectures/21.pdf GO BLUE

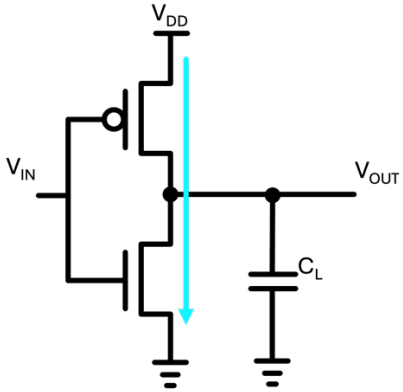
During Transition, there may be overlap of conductances



http://web.eecs.umich.edu/~twenisch/470_F07/lectures/21.pdf GO BLUE

Dynamic Power Consumption





Capacitive Dis/Charging

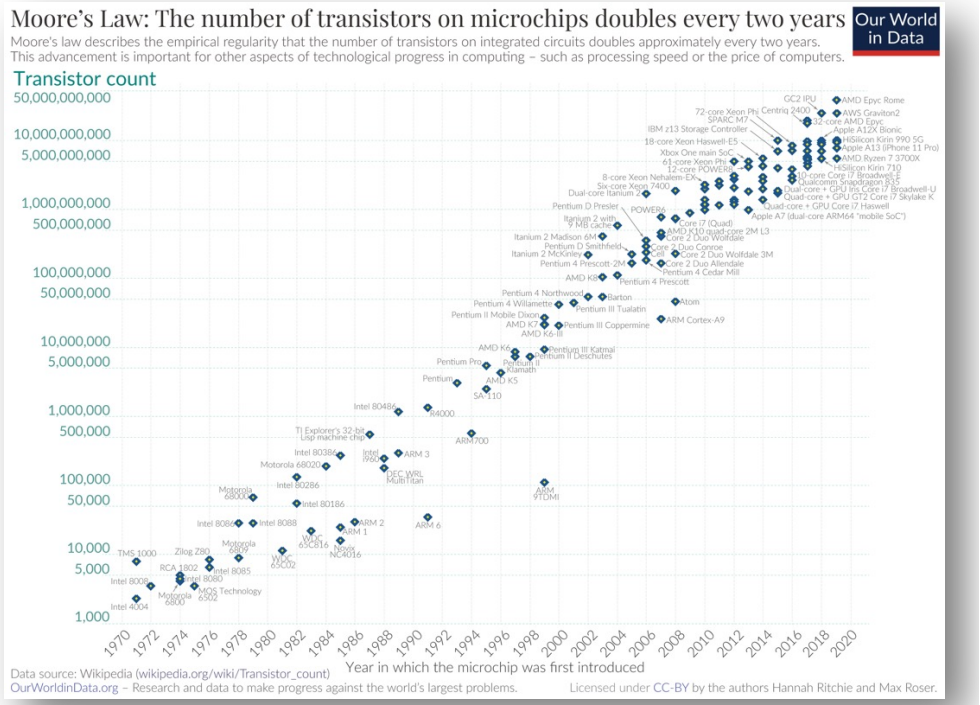
- Caused by need to store up finite charge
- $P \propto CV^2 f$
- C: capacitance of gate
- V: V_{DD} of system
- f: frequency of switching

Short Circuit

- $P \propto t_{sc} V I_{peak}$
- t_{sc}: in crossover
- V: V_{DD} of system
- I_{peak}: Max current at crossover
- Good news this is usually rather small compared to **capacitive**

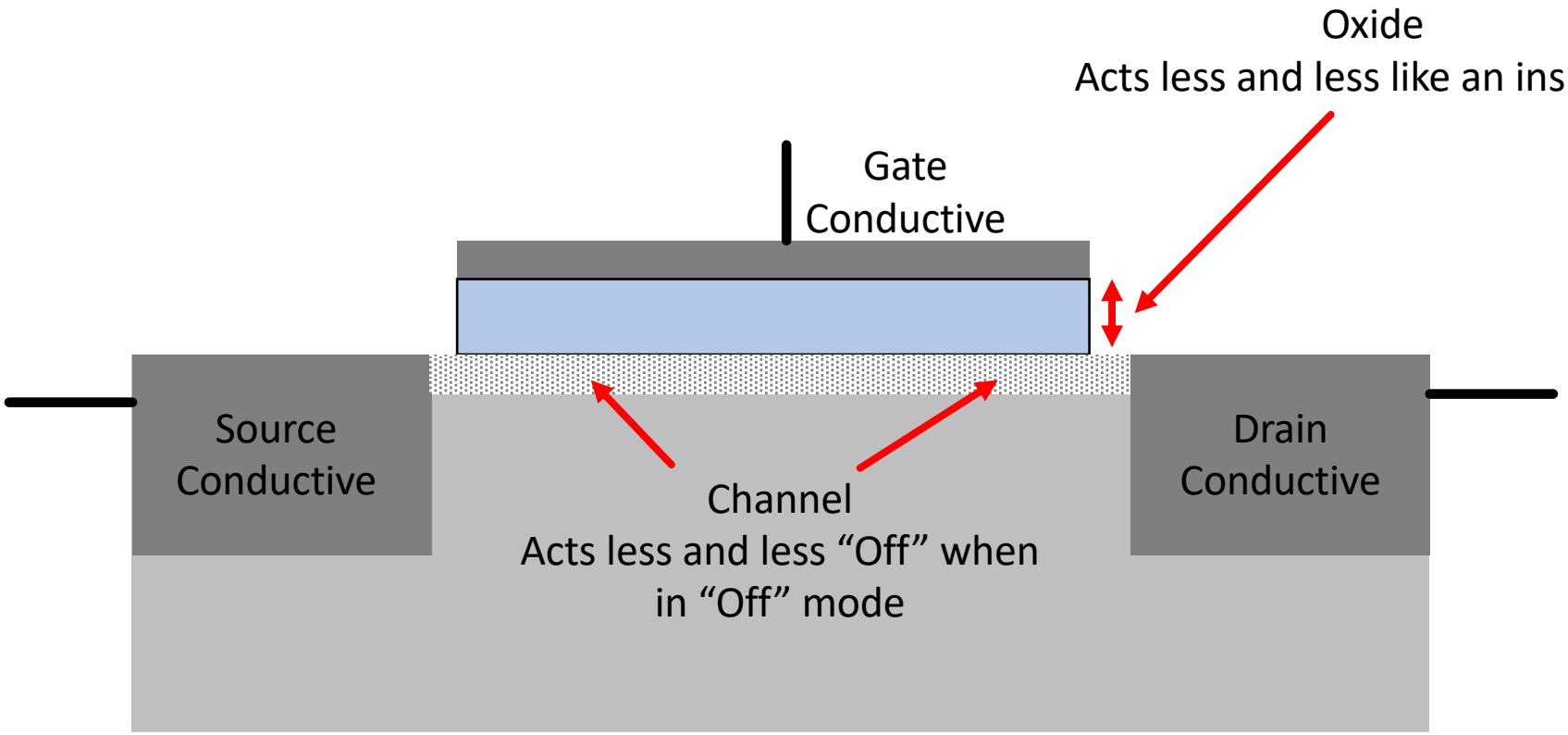
The Downward Scaling of Transistors

- Most gains in computation have arisen from making smaller transistors…
- Fit more on a chip.
- They use less power (less gate capacitance!!!)
- Can do more for less!

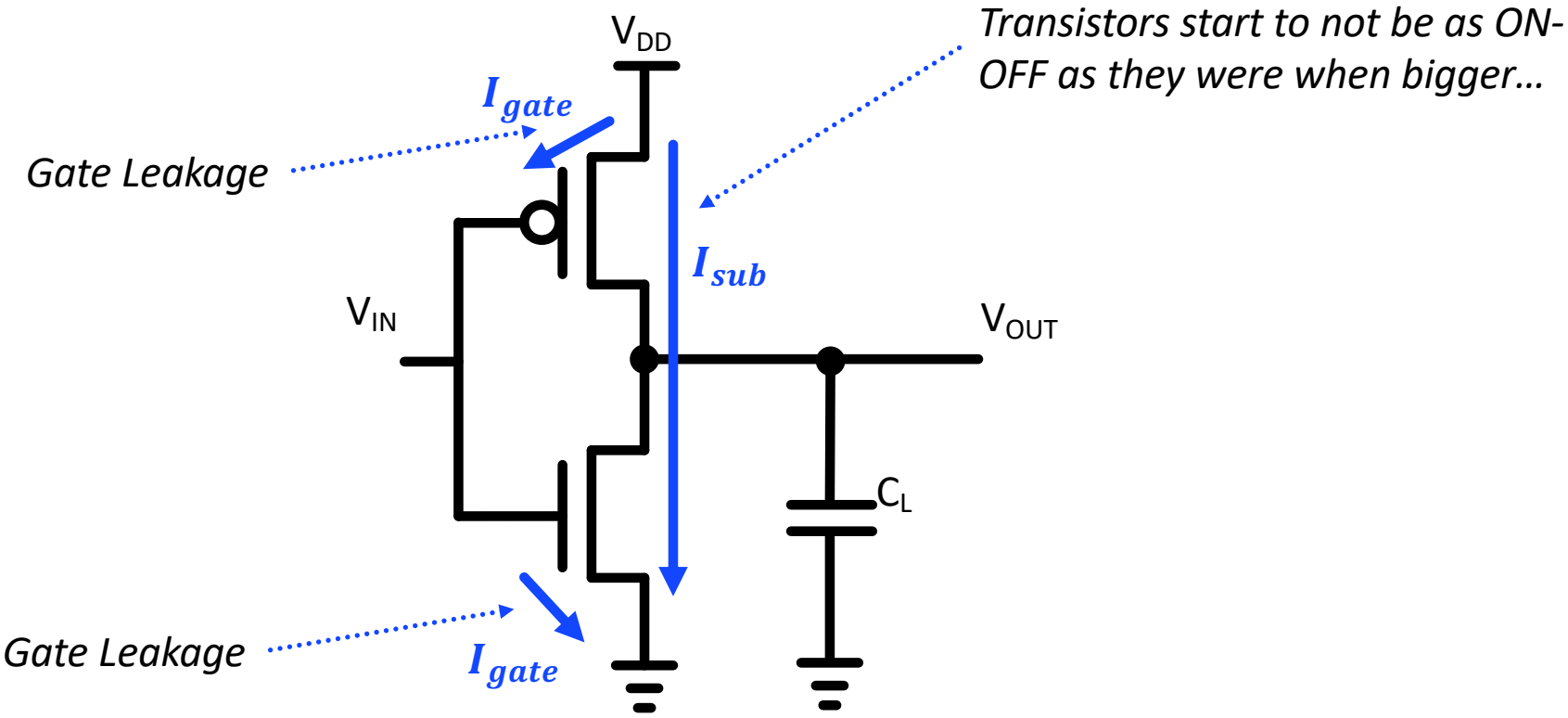


https://en.wikipedia.org/wiki/Moore%27s_law#/media/File:Moore's_Law_Transistor_Count_1970-2020.png

As MOSFETs get Very Small…



Static Power Consumption



http://web.eecs.umich.edu/~twenisch/470_F07/lectures/21.pdf GO BLUE

New Loss Mechanisms: Static Power

- **Gate Leakage:** The gates don't act like perfect insulators so you leak current (power) through them

- **Sub-threshold Leakage:** The transistors don't turn On and off as sharply, so there's more mushing and overlap and start to conduct significant amounts all the time (wasted leaks)

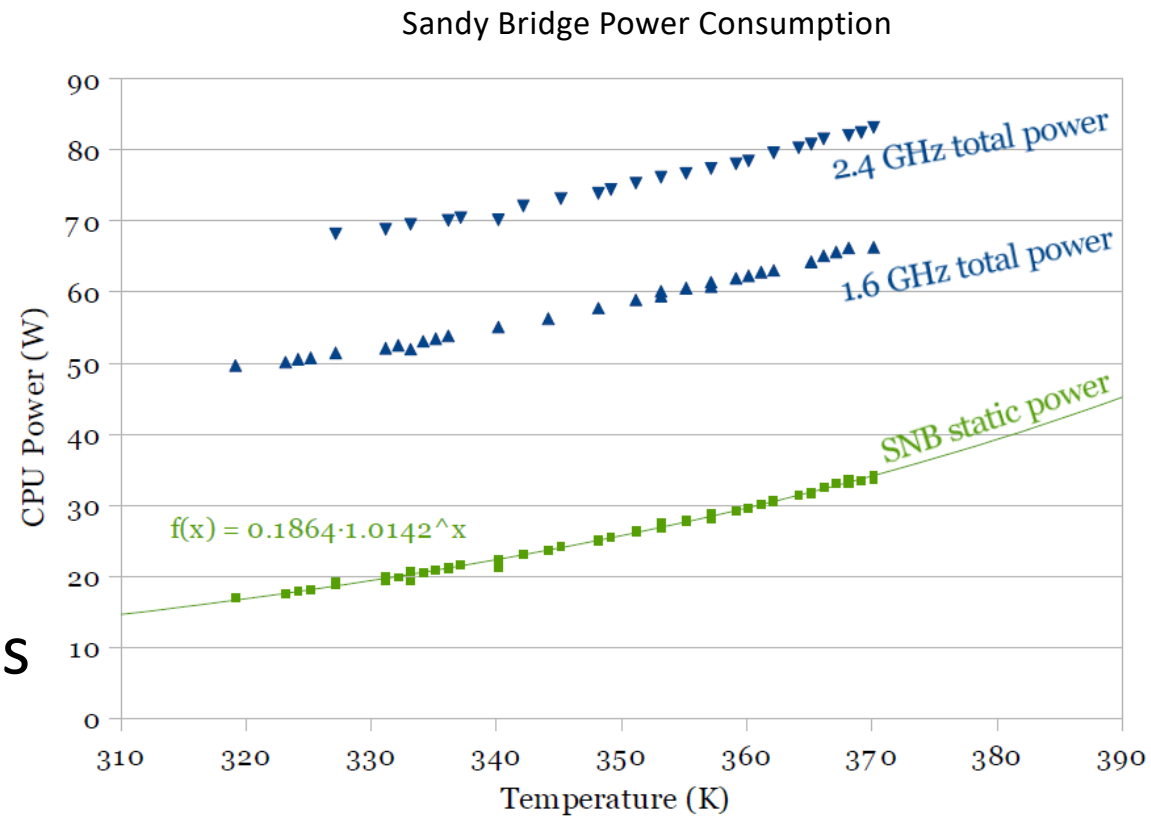
Static Power Losses

- As opposed to dynamic power losses which arise from the act of switching bits…
- Static losses take place just by being powered up and existing.
- So the total power consumed ends up being roughly explained by this equation:

$$\boldsymbol{P_{total} = P_{dynamic} + P_{static}}$$

Power Consumed $P_{total} = P_{dynamic} + P_{static}$

- In the CMOS era, historically static power has been smaller compared to dynamic power
- This has changed in recent years as things have gotten smaller!

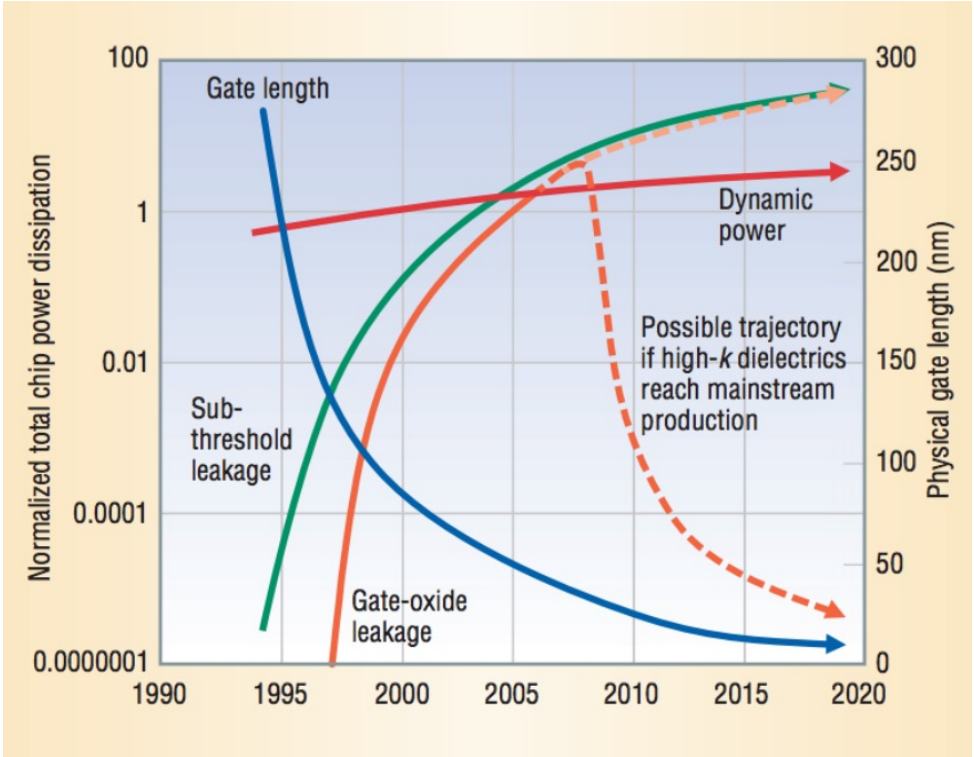


Shmoo plot

https://blog.stuffedcow.net/2012/10/intel32nm-22nm-core-i5-comparison/

Leakage Has Gotten So bad

- How bad is it?
- In some contexts, static loss starts to dominate dynamic loss
- This is a really big deal since the primary loss mechanism is beyond the control of implementation design, etc…

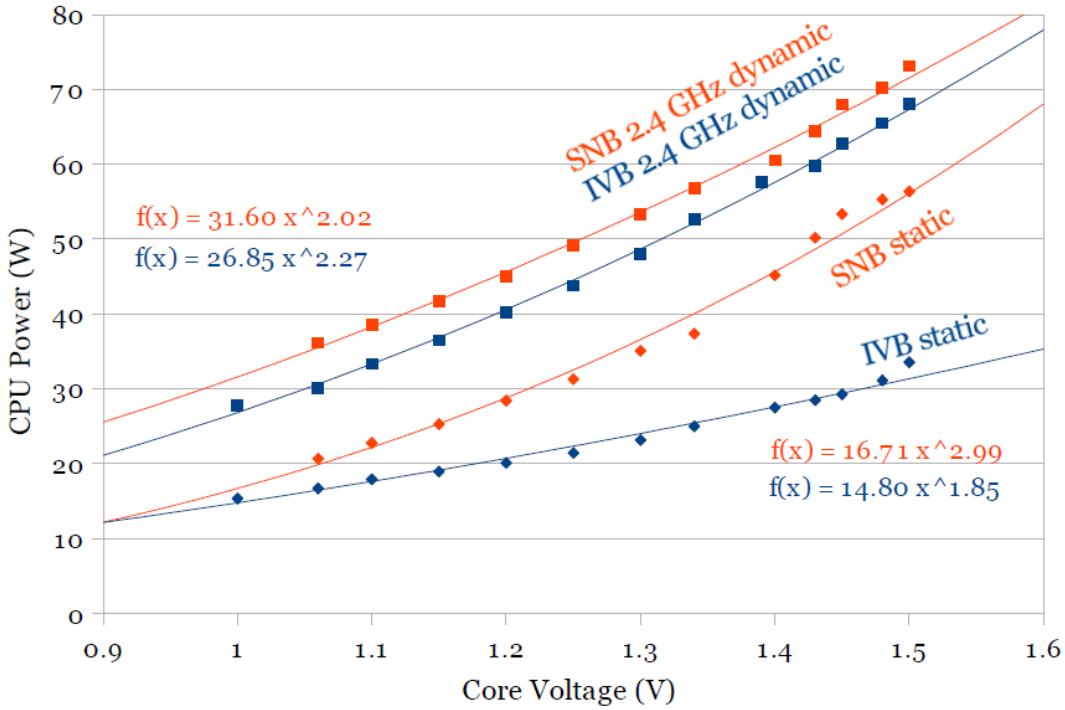


From 2011

Leakage Current: Moore's Law Meets Static Power
http://www.ruf.rice.edu/~mobile/elec518/readings/DevicesAndCircuits/kim03leakage.pdf

Sandy Bridge vs. Ivy Bridge (32nm vs. 22 nm core i5)

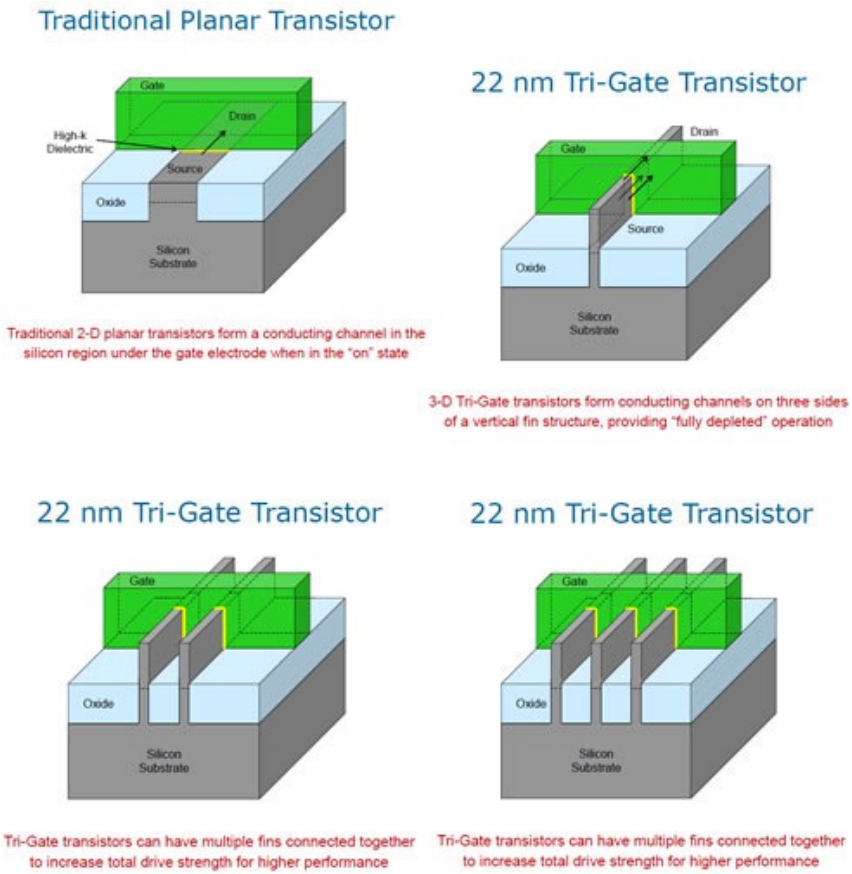
- Sandy Bridge was older model transistor
- Ivy Bridge was 3D transistor which greatly improved static loss (less leakage)



Shmoo plot

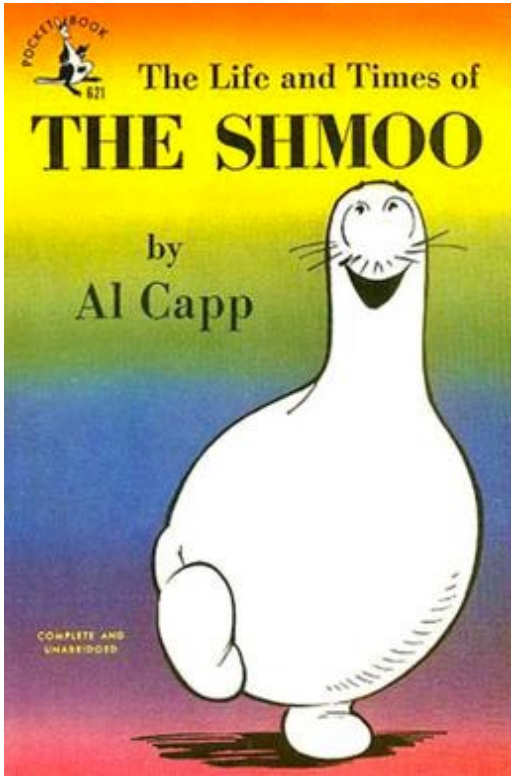
Trigate MOSFETs

- One of the first departures from planar semiconductor fabrication since we started doing it as humans in the early 1950s.
- Was in the pipeline since right around 2000, and finally started coming out in 2014
- Cuts static loss (sub-threshold loss in particular) by 50%



Aside: Shmoo Plot

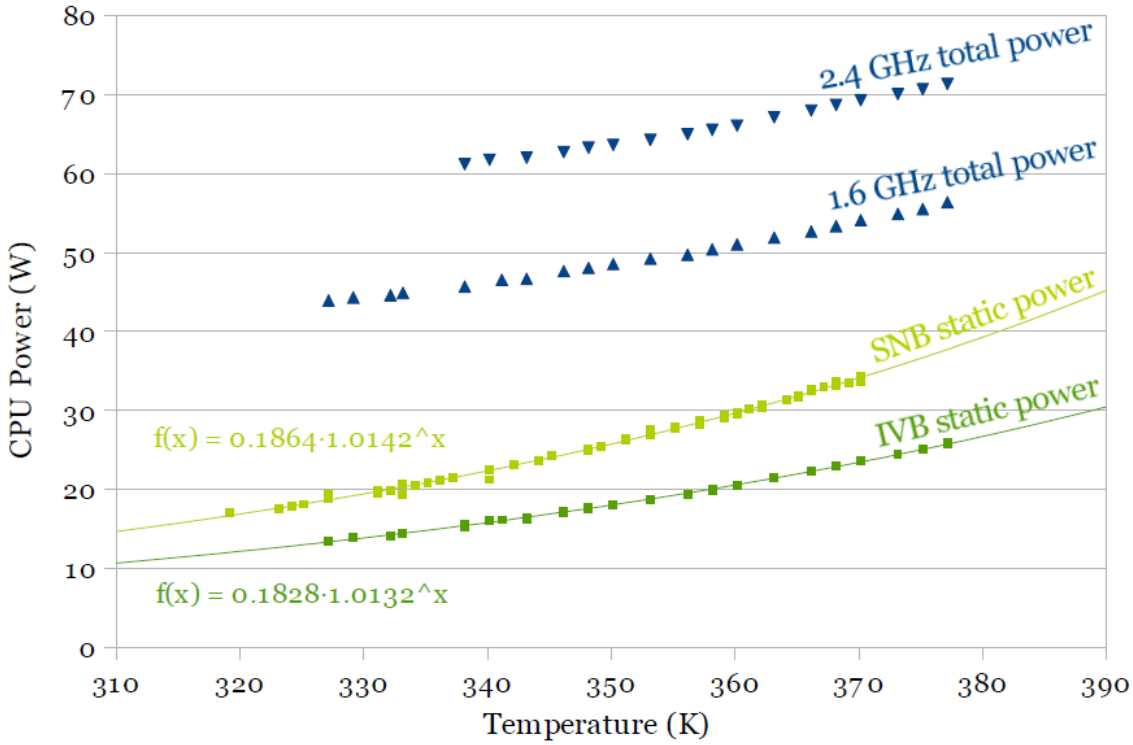
- Sometimes hear plots of various performance specs on semiconductors called “Shmoo” plots
- Called that because original plots looked like Shmoos, which were weird bowling-pin like creatures from Lil Abner,
- Anyways sometimes these comparison plots are called Shmoos



**Wikipedia finally explained this to me…pre-semiconductor, Shmoo plots for magnetic devices looked like Shmoos*

Sandy Bridge vs. Ivy Bridge (32nm vs. 22 nm core i5)

- Intel fell way behind schedule getting their 22nm tech into production, but its trigate devices in IVB, drastically cut down static power loss



http://blog.stuffedcow.net/2012/10/intel32nm-22nm-core-i5-comparison/

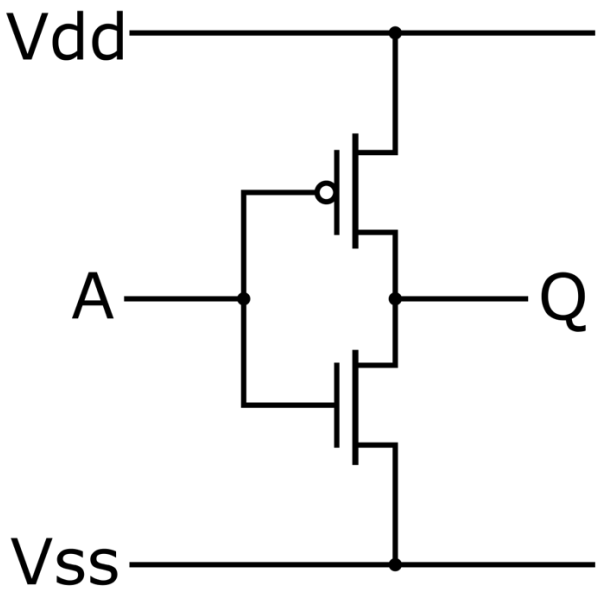
Summary: Digital Power Consumption

- P : total power consumed
- $\alpha_{0\to1}$: fraction of gates switching
- C : Capacitance of gates, busses, interconnects
- V : Operating voltage (V_{dd})
- f : frequency of operation
- I_{leak}: Leakage Current:
  - Sub-threshold leakage
  - Gate-Leakage

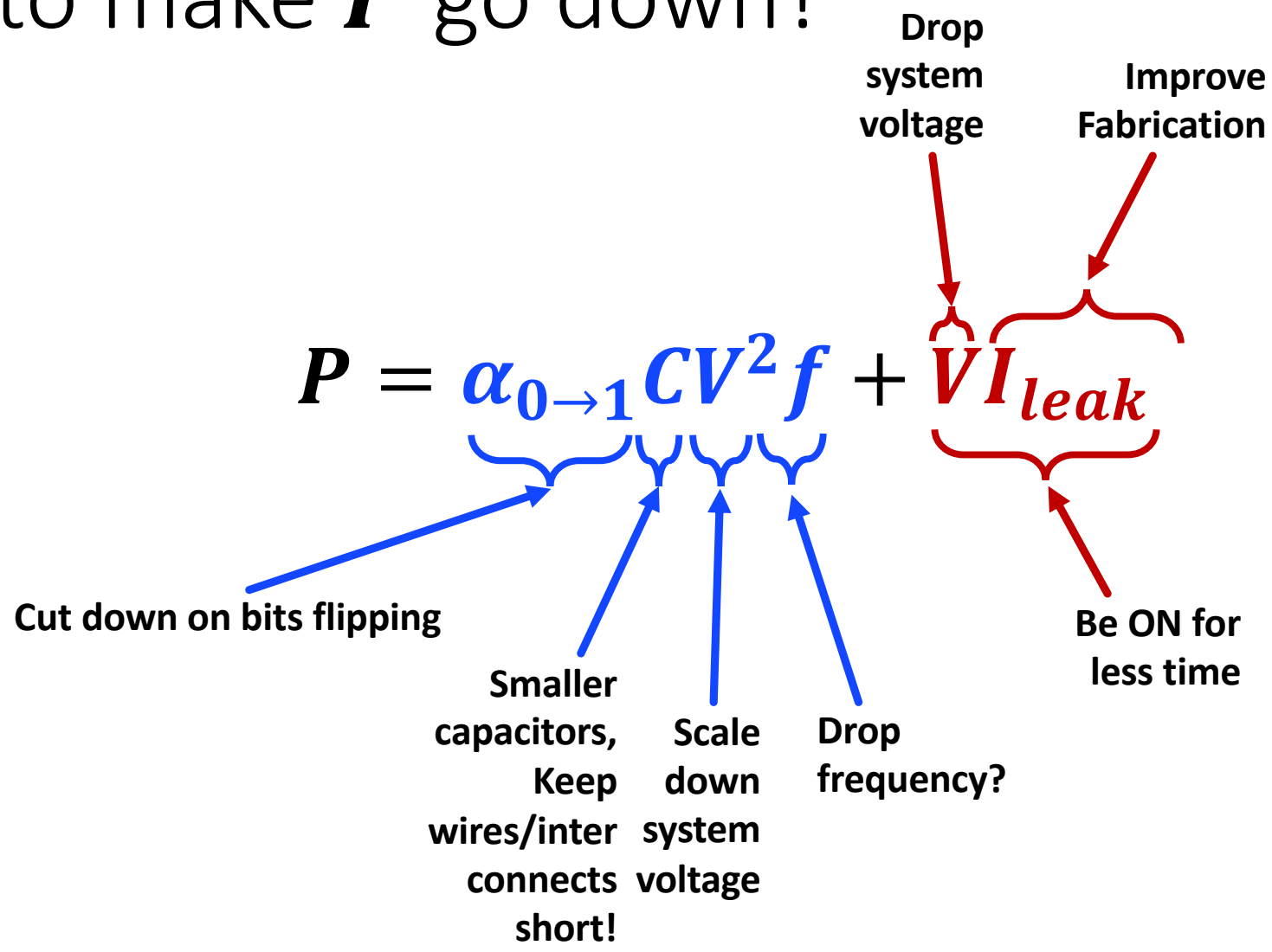
Dynamic power consumption

$$\boldsymbol{P = \alpha_{0\to1} C V^2 f + V I_{leak}}$$

Static power consumption



How to make $\boldsymbol{P}$ go down?



What do ***we (aka 6.205)*** have control over?

Dynamic power consumption

Static power consumption

$$P = \alpha_{0\to1} C V^2 f + V I_{leak}$$

- **Dynamic Power Usage** is more closely tied to how we use the system:
  - Design, data structures and representation, etc…
  - Clock
  - Etc…

- **Static Power Usage** is more closely tied to actual system fabrication and capabilities, but our usage of it can also factor in:
  - Temperature (heat sink it)
  - Turn on/off completely

Dynamic Power Reduction Strategies

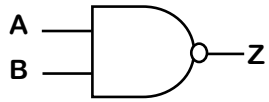
$$P = \alpha_{0->1} C_L V_{DD}^2 f$$

- Reduce Transition Activity or Switching Events
- Reduce Capacitance (e.g., keep wires short)
- Reduce Power Supply Voltage (not really option in 6.205 designs, but when designing from scratch yes)
- Frequency is sometimes fixed by the application, though this can be adjusted to control power as needed

Optimize at all levels of design hierarchy

The Transition Activity Factor $\alpha_{0->1}$

0 to 1 since that's when we put energy into the system/charge the capacitors



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 (not guaranteed at all)

What is the average power dissipation?

$$\alpha_{0->1} = 3/16$$

$$P = \alpha_{0->1} C_L V_{DD}^2 f$$

Pages 7 and 8 here of the notes: https://www.brown.edu/Departments/Engineering/Courses/engn2912/Lectures/LP-02-logic-power-est.pdf

Power Consumption Can Be Data Dependent!

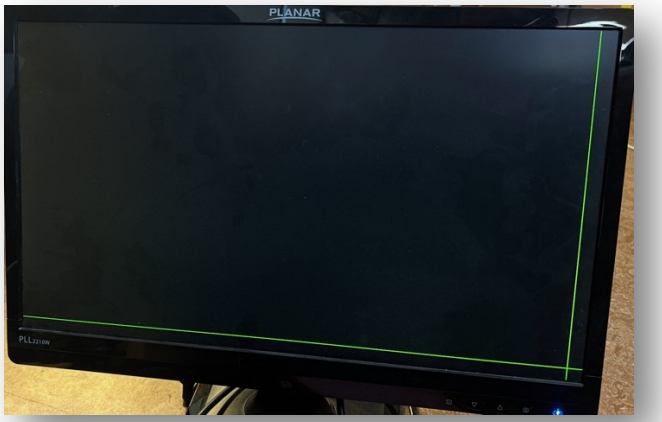
$$P = \alpha_{0\to1} C V^2 f$$

- We don't think about this but this is true.
- For a given system, it will consume more power when it is actually processing legitimate information than when it isn't.
- And this is not ignorable….

Exact Same Device, just change data

6.205 Camera Running and Exposed:

6.205 Camera Running and Covered:



*Achieve a ~30% drop in power consumed by camera and FPGA based off of the *type* of data transferred through the system*

Power Consumption Can Be Data Dependent!

$$P = \alpha_{0\to1} C V^2 f$$

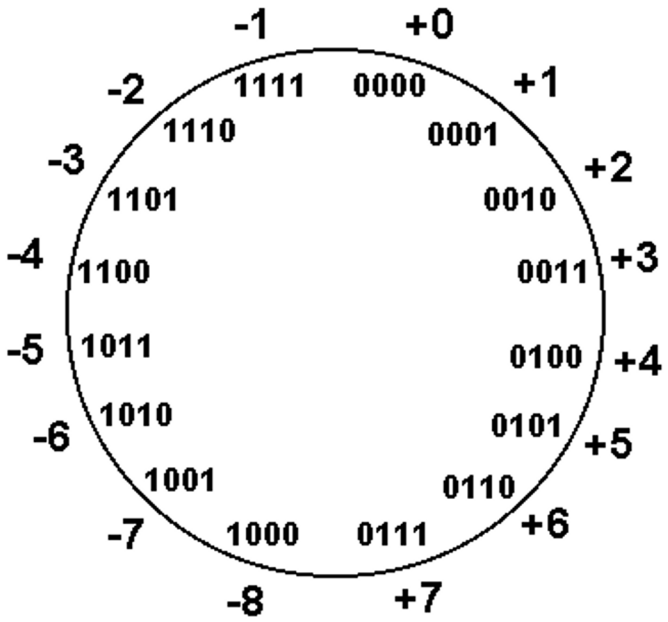
- We don't think about this at the programming level, but at the bit level it can really matter!

- Is your data encoded in a way such that lots of bits flip lots of the time? (lots of charge/discharge cycles!)

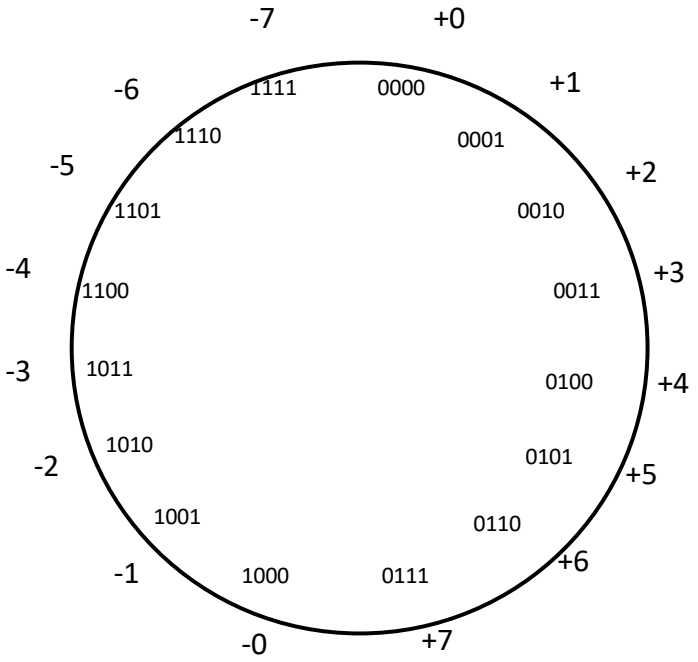
    - Are the common transitions using the fewest bit changes?

Number Representation: Two's Complement vs. Sign Magnitude

Two's complement



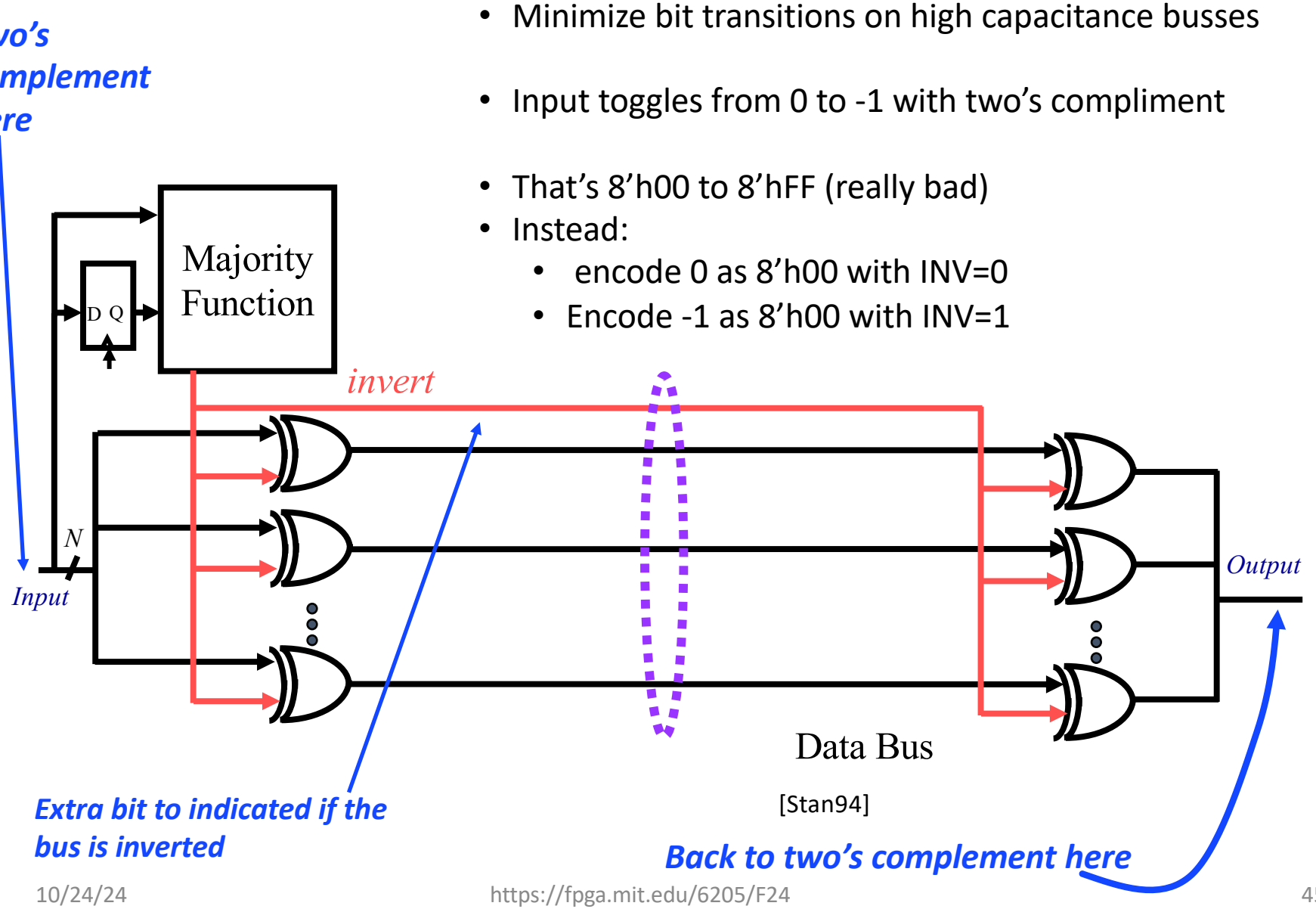
Sign-Magnitude



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

- Minimize bit transitions on high capacitance busses
- Input toggles from 0 to -1 with two's compliment
- That's 8'h00 to 8'hFF (really bad)
- Instead:
  - encode 0 as 8'h00 with INV=0
  - Encode -1 as 8'h00 with INV=1



[Stan94]

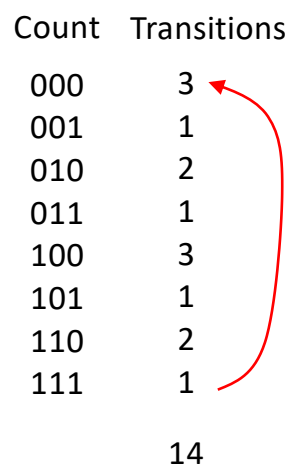
Counting in General

- Going up in “natural” format is beneficial for math and things, but isn’t necessarily the best way to encode data from a bit-flip perspective.

- There are other ways of counting that minimize changes.

Hamming Distance

- Counting to 8 in regular 3bit binary involves 14 total bit changes
- Counting to 8 in 3bit Gray involves 8 total bit changes(big savings)
- Reduce Hamming Distance between sequences…don't count up with states using regular binary…use a Gray code perhaps



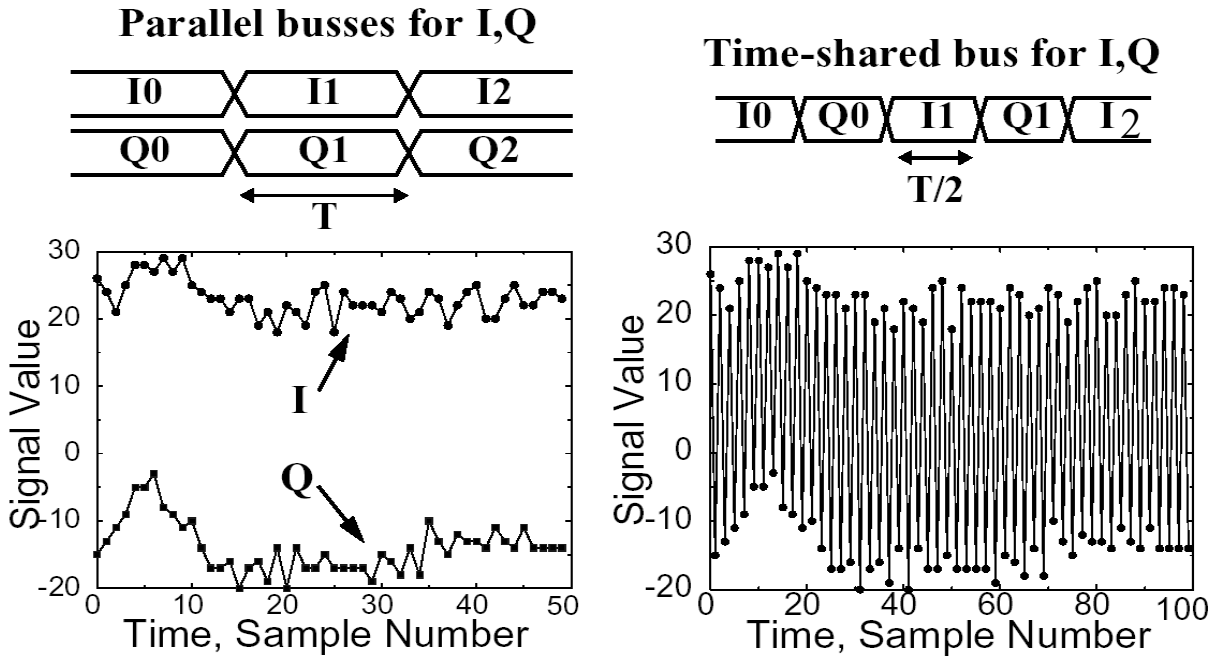
Gray code by bit width

3-bit	4-bit
000	0000
001	0001
011	0011
010	0010
110	0110
111	0111
101	0101
100	0100
2-bit	1100
00	1101
01	1111
11	1110
10	1010
1-bit	1011
0	1001
1	1000

Time Sharing is a Bad Idea (From a power perspective)

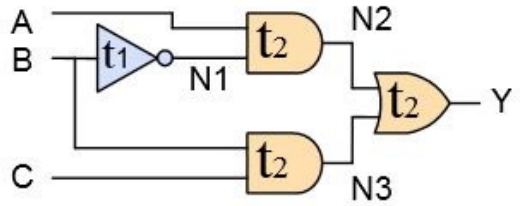
- If you have data sets that are expected to be very different in value, consider giving each their own bus

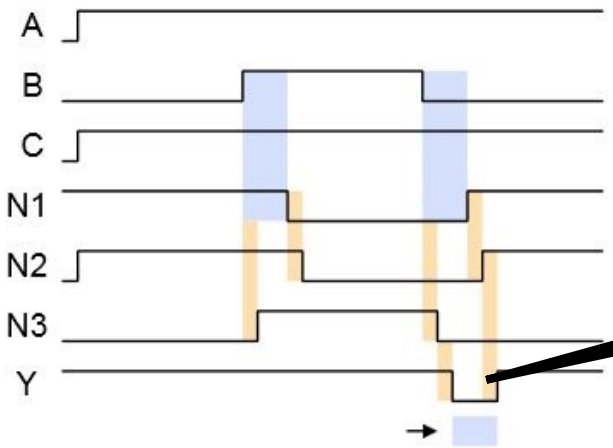
- Minimize the 0->1 transitions that will happen on any given one bus.



Glitching Transitions

Glitches are no longer an annoyance, but vile leeches sucking our vital life fluids (power) from our bodies (electrical devices)





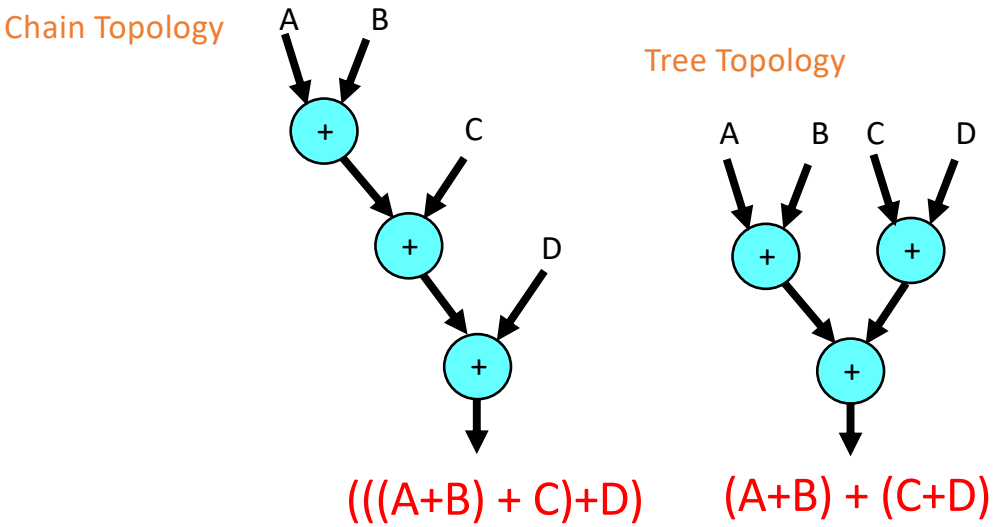
10 femtoJoules, please!

@22nm process: 10 fJ x 100 MHz = ***1 microWatt***

Glitching Transitions

Glitches are no longer an annoyance, but vile leeches sucking our vital life fluids (power) from our bodies (electrical devices)

- Balancing paths reduces glitching transitions
- Structures such as multipliers have lot of glitching transitions
- Keeping logic depths short (e.g., pipelining) reduces glitching



Software Issues

- Doesn't just matter in hardware…
- Consider two collections of code below…
- Which one is better?

```
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);
}
```

```
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);
}
```

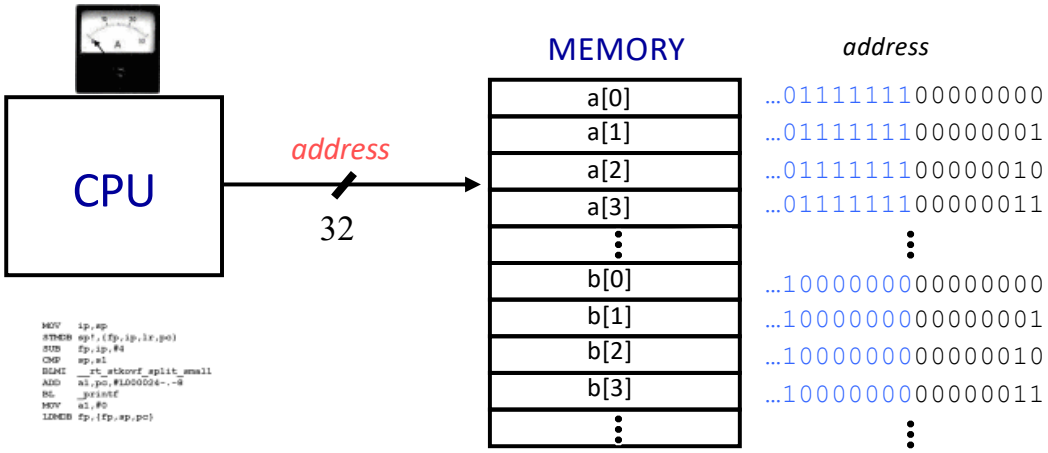
Software Issues

```
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);
}
```

```
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);
}
```

- It really depends on where these arrays are located in memory…
- If nearby or clean multiples/aligned in memory might not be too bad…

But like if arrays are far apart in memory you could have problems…




```
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);
}
```

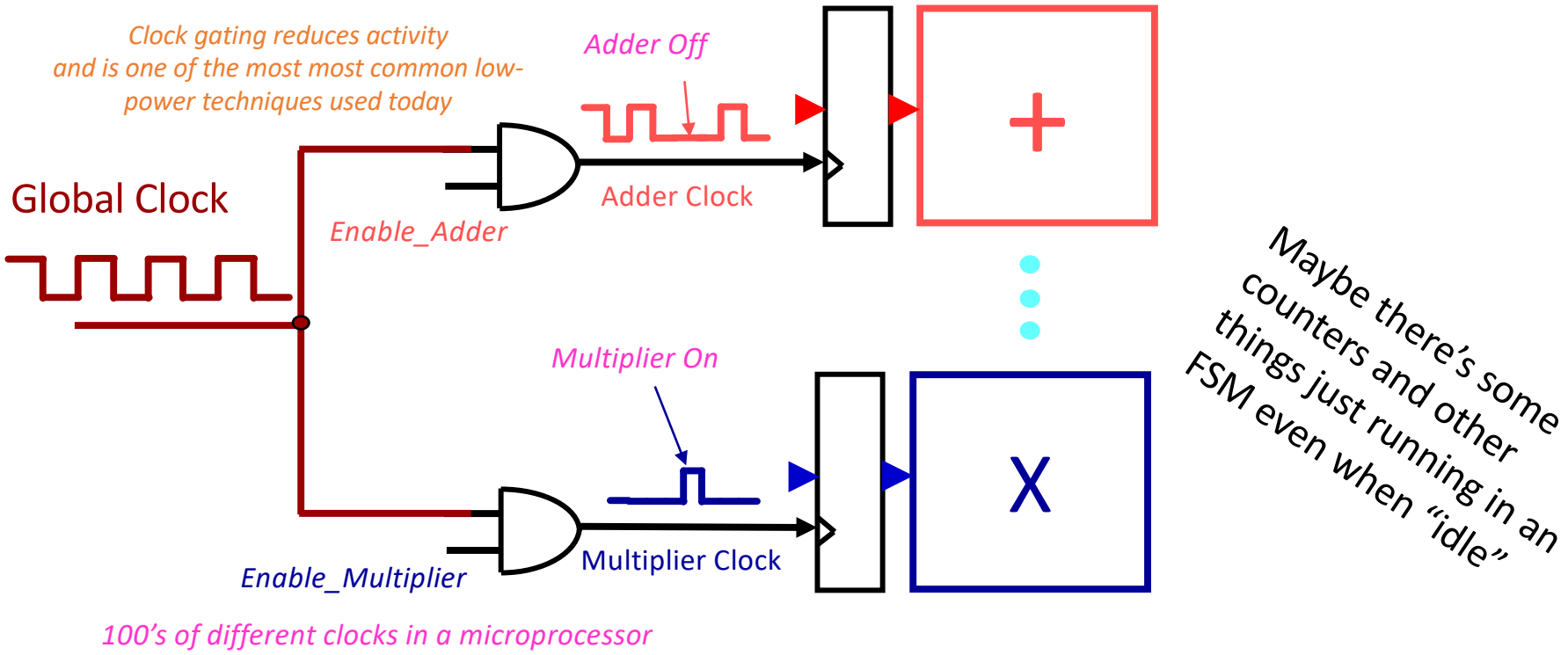
Address bus will undergo:
= 4607 bit transitions

```
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);
}
```

Address bus will undergo:
= 1030 transitions

In addition to possible issues with cache hits/misses, etc…

Clock Gating is another Potentially Good Idea!

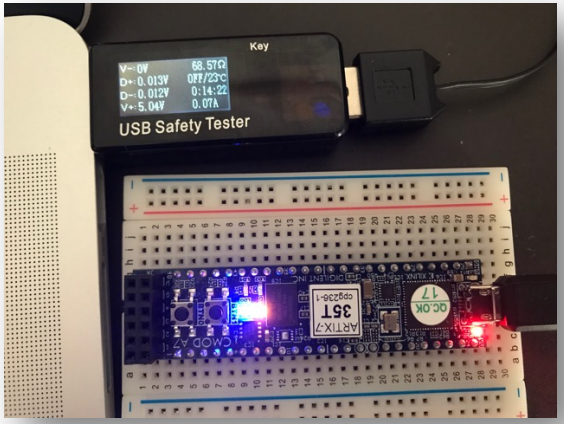


- Careful to keep combinatorial paths short…avoid clock skew!!!
- Clock-gating is widely used nowadays, particularly in ASICs

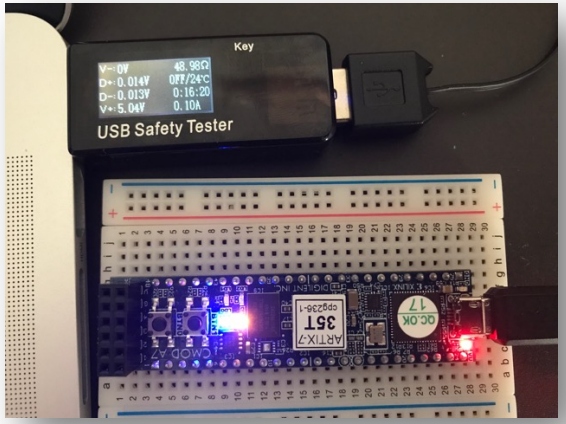
Clock Generation Uses Power!

- In very complex modern systems, clock management (amplifiers, phase locked loops, etc) add significant overhead to a system's power needs. In some recent systems, the clock distribution system can account for 30% of all power.
- In general with everything, if you don't need it, don't use it.
- Human eye can't tell difference between these two dimmers

Running Blue LED at 50% duty cycle at 100Hz
Based off of 12 MHz clock
Consuming 0.35 W

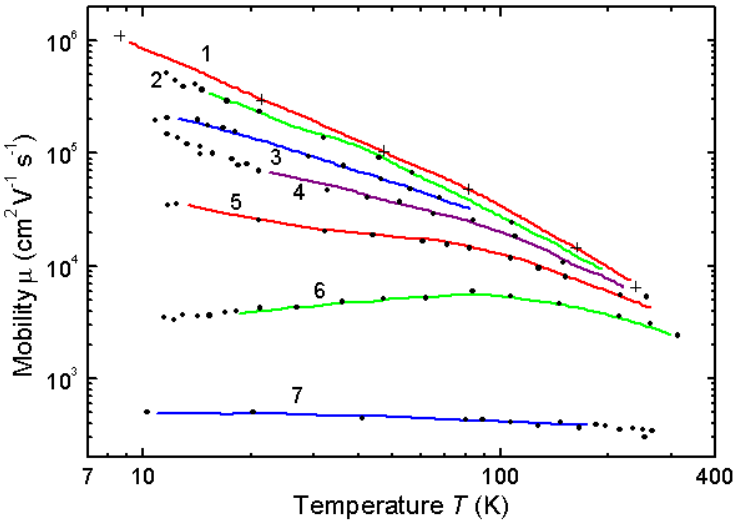


Running Blue LED at 50% duty cycle at 4000Hz
Based off of 480 MHz clock
Consuming 0.5W



Temperature

- While some input power gets used for information/computation, most is ultimately lost as heat
- As temperature rises, carrier (the electrons and holes) mobility will drop off quickly
- As mobility drops off, current delivered drops off, systems can't charge/discharge as quickly, we run into trouble
- **Static losses also get really bad at high temperatures**
- So you want to keep temperature down!

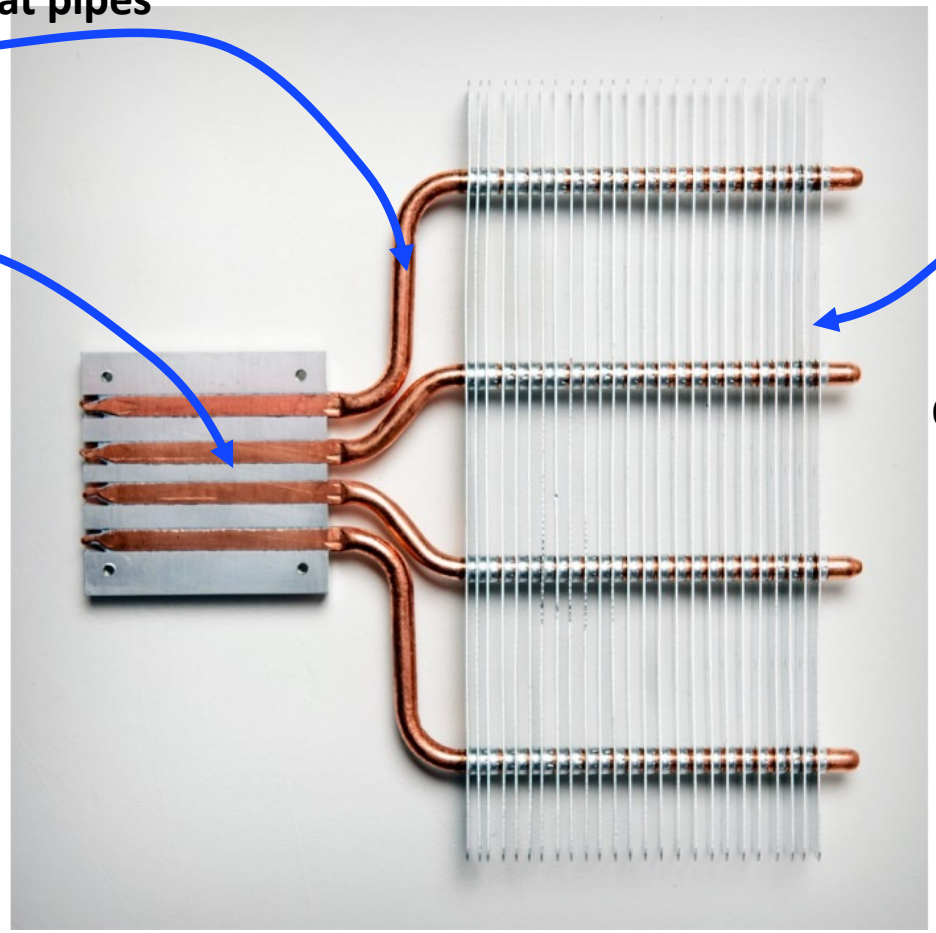


Not only does an inefficient design waste more energy, you now have to get rid of more heat, else your device will run less efficiently and waste even more energy.

It is a positive feedback process.

http://www.ioffe.ru/SVA/NSM/Semicond/Ge/electric.html

Anatomy of a Heat Sink



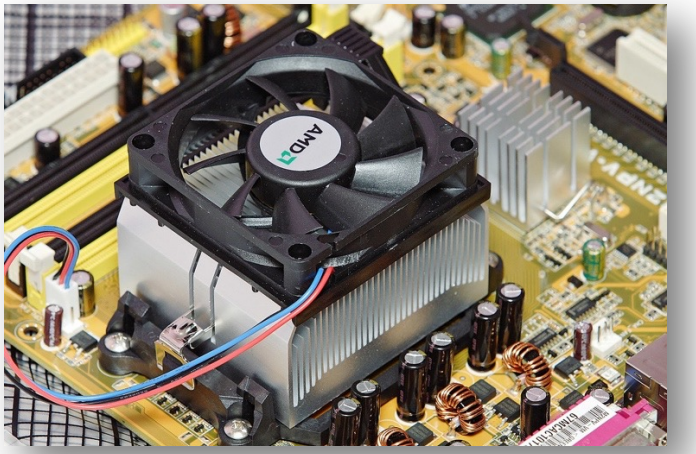
https://reefll.com/index.php?route=product/product&product_id=69

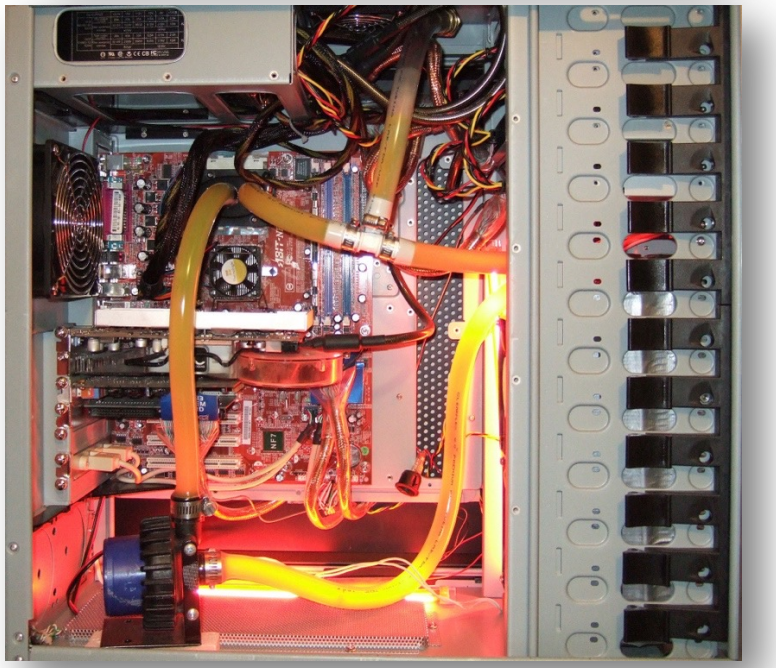
Heat Sink Metal

- Copper is the best but expensive.
- Aluminum is cheaper but not as good.
- Other metals not very good.

More than just Heat Sink

- Add fan to move air across heatsink to further improve cooling
- Thermal capacity of air is low…use water or oil or carcinogens to move heat away from computer more effectively

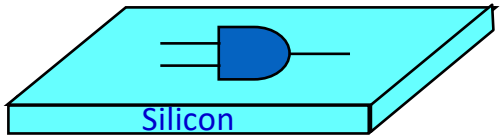




https://en.wikipedia.org/wiki/Computer_cooling

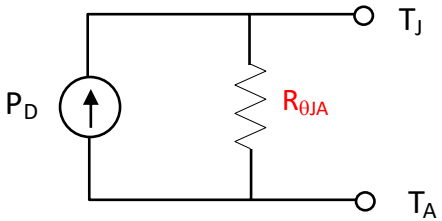
Junction (Silicon) Temperature

Simple Scenario



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

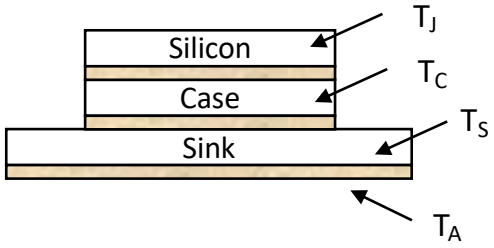
$R_{\theta JA}$ is the thermal resistance between silicon and Ambient

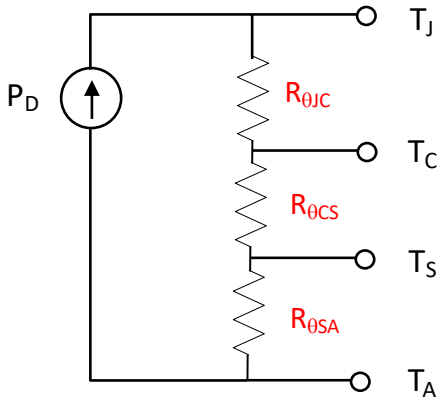


$T_j = T_a + R_{\theta JA} P_D$

Make this as low as possible

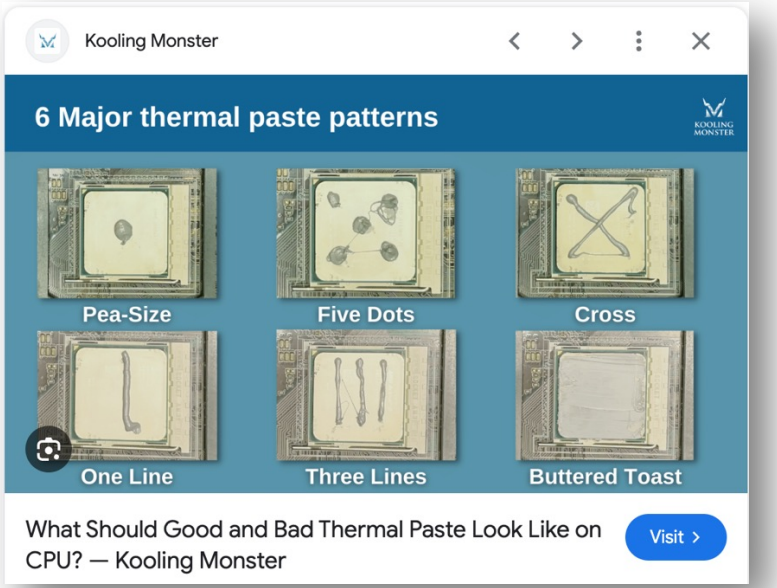
Realistic Scenario





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

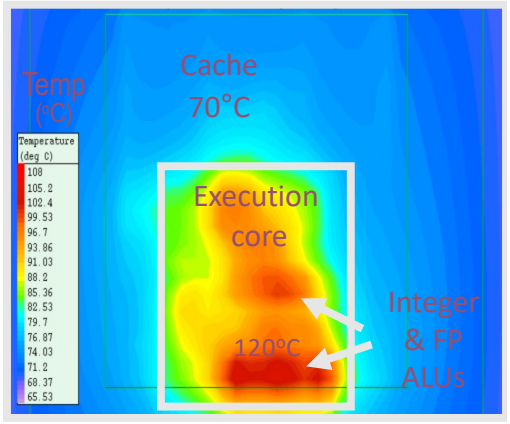
is minimized by facilitating heat transfer
(bolt case to extended metal surface – heat sink)



These are thermal model circuits
Not electrical circuits

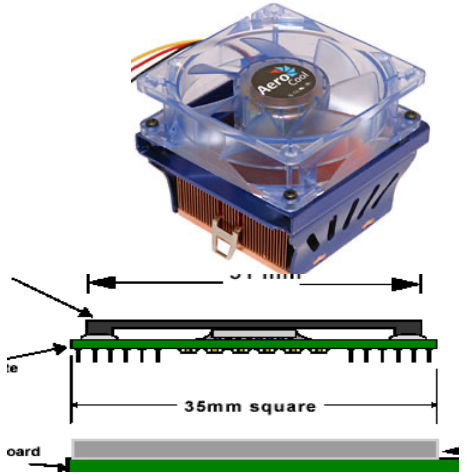
Intel Pentium 4 Thermal Guidelines

- Pentium 4 @ 3.06 GHz dissipates 81.8W! (i7 Haswell 3.2GHz 65W)
- 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)

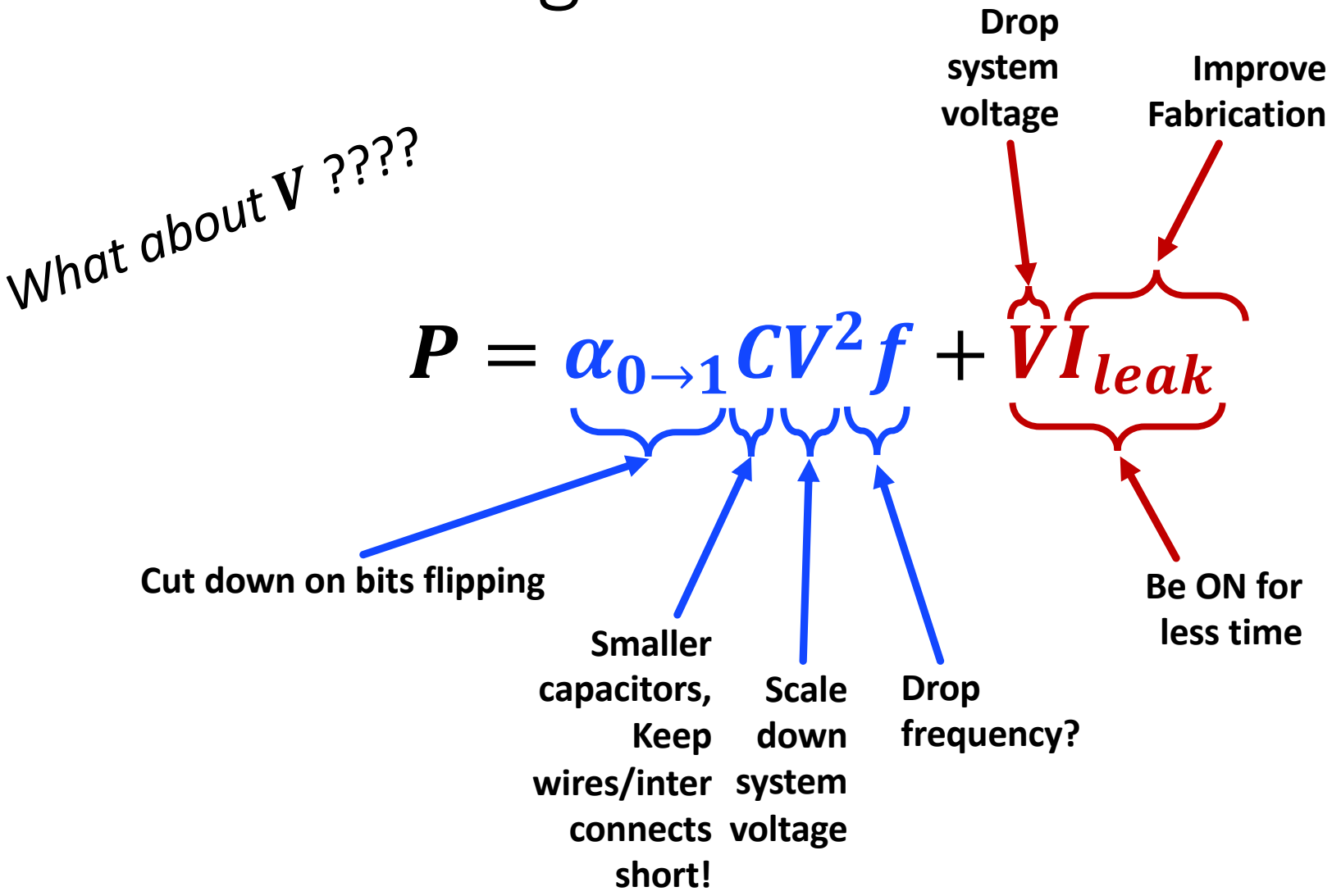


Courtesy of Intel
(Ram Krishnamurthy)

Our 6.205 Camera does a tremendous amount of processing (and also has a problematic internal regulator) so it gets very warm and needs a heat sink

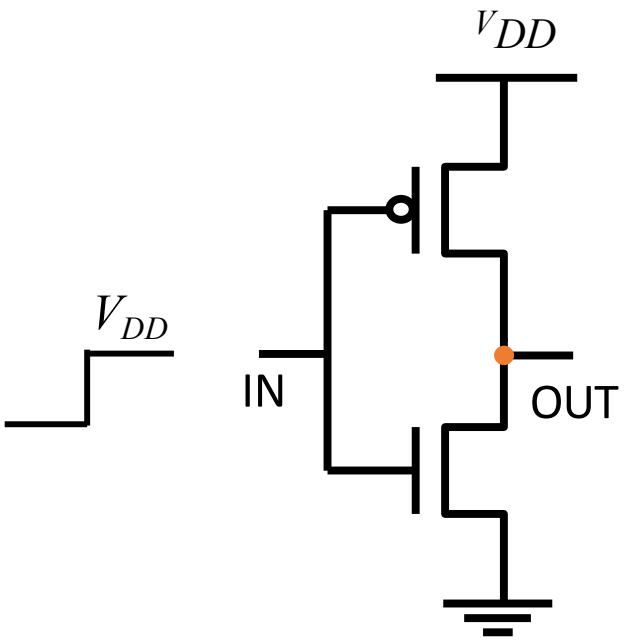


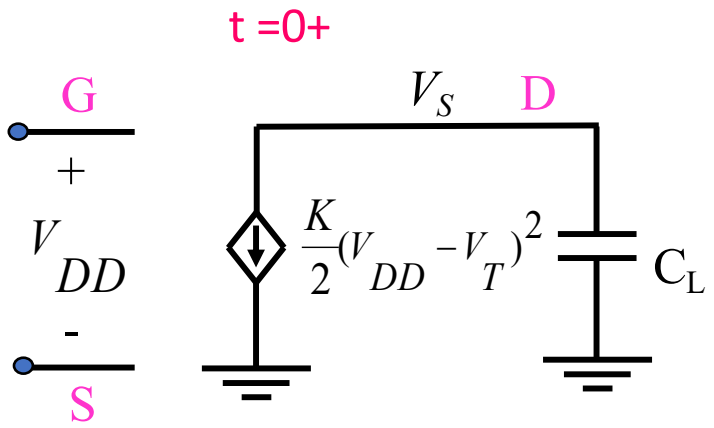
How to make $\boldsymbol{P}$ go down?



Reduce Supply Voltage: But is it Free?

$$\boldsymbol{P = \alpha_{0\to1} C V^2 f + V I_{leak}}$$



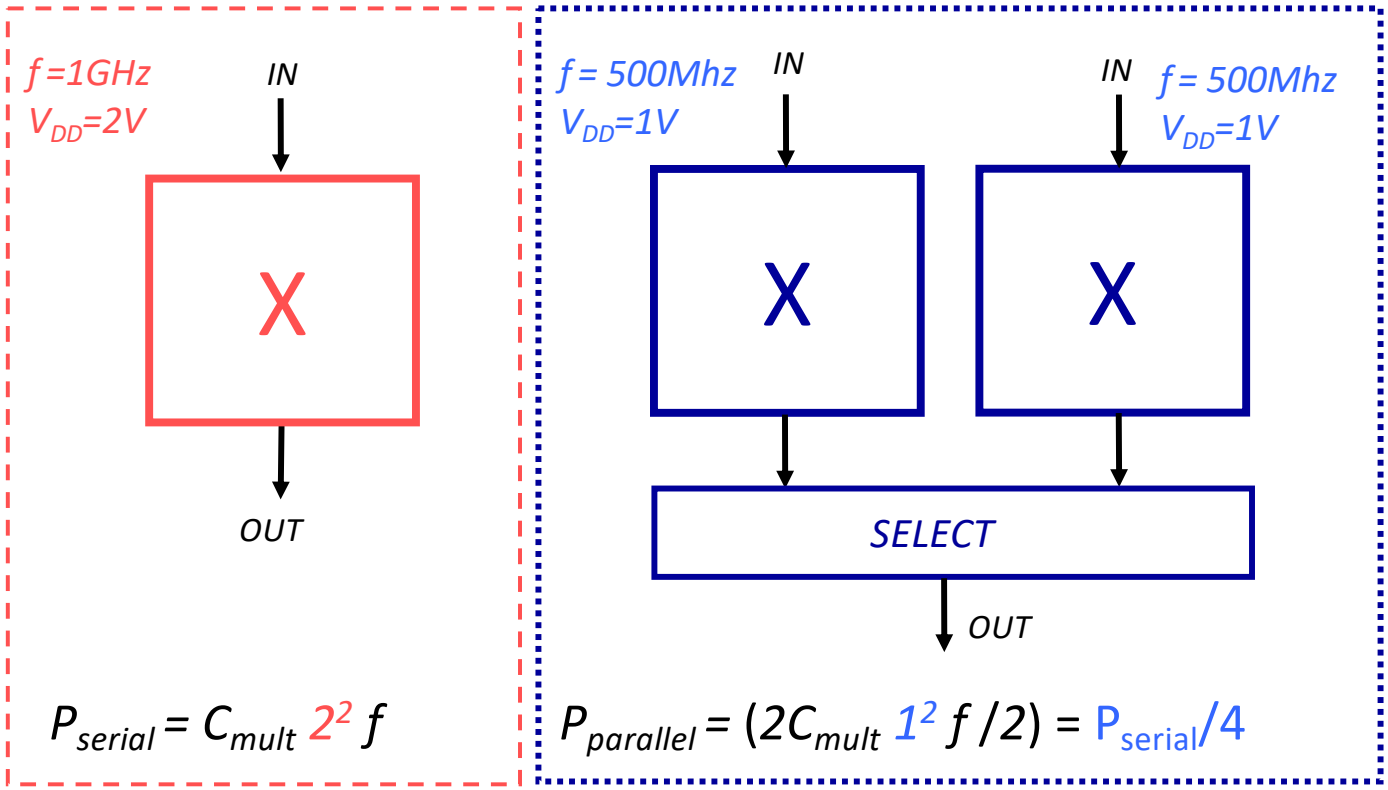


$$Delay = \frac{C_L \cdot \Delta V}{i_D} = \frac{C_L \cdot \frac{V_{DD}}{2}}{\frac{k}{2}(V_{DD}-V_T)^2} \propto \frac{V_{DD}}{(V_{DD}-V_T)^2} \approx \frac{1}{V_{DD}}$$

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

Transistors Are Basically Free…

$$P \propto CV^2 f$$

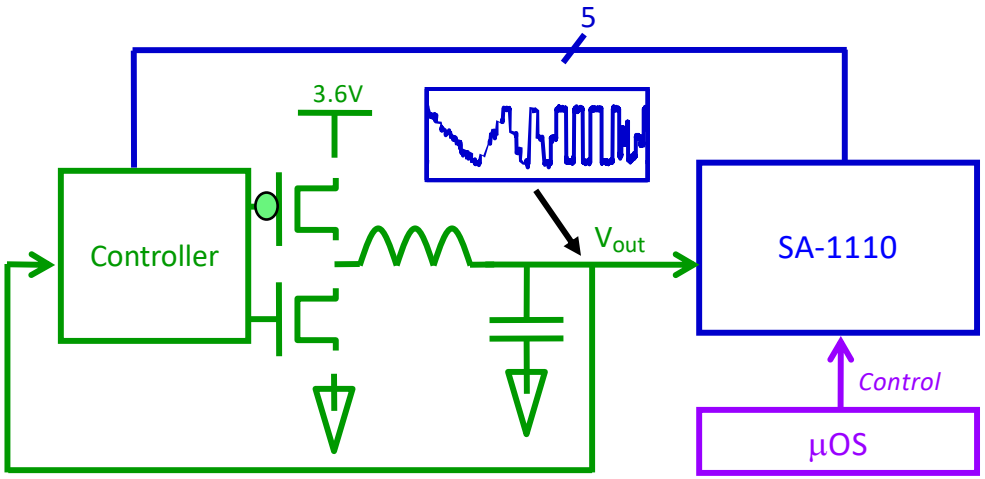


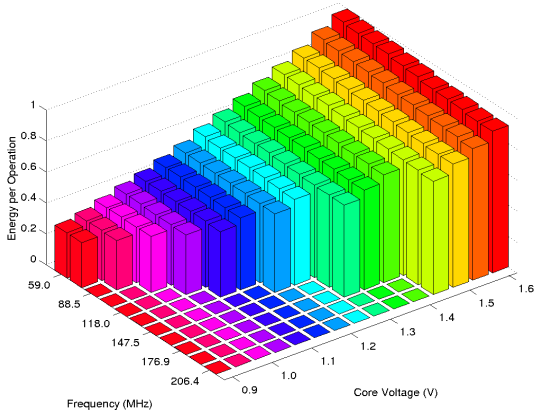
Trade Area for Low Power, Possibly higher latency

Dynamic Voltage Scaling on a Processor

- Some systems take to extreme and dynamically scale their voltage!

Digitally adjustable DC-DC converter powers SA-1110 StrongArm core





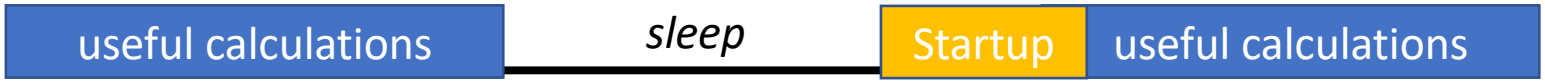
µOS selects appropriate clock frequency based on workload and latency constraints

Even more extreme…the Threat of Static Power Loss

- If you're a major loss mechanism becomes static phenomena, then there will become a point where a cranking the clock could be beneficial!
- Run as fast as possible with the best hardware as possible (32 bit MCU if appropriate vs. 8 bit or something)
- Then sleep/power down! (the static loss monster won't get you if you're in sleep)
- Not necessarily the right solution, particularly as new transistor models come in and keep static loss at bay, but you never know.

Powering On and Off

- Historically, turning on and off a computational device had significant startup costs:
  - A processor might need 100,000 clock cycles to startup during which time it used energy but couldn't do useful work



- The length of sleep vs. required startup restricted system (why go to sleep if you need more energy to startup anyways)
- More recently new circuit architectures are showing up that allow like ~15 cycle startups and stuff for really low-energy designs

Mingoo Seok
https://www.ee.columbia.edu/~mgseok/publications.html

Alternative Energy Sources

- While personal computers have gotten much more efficient, we tend to just use that as an excuse to do more (so they use ~similar amounts of power total throughout recent years)
- For given tasks, computation has become much more efficient…that means in many specialized applications, the power to do something is much less. That means we can power these circuits with tiny amounts of energy:
  - Energy Harvesting – movement
  - Energy Harvesting – thermoelectric generator
  - Ambient RF
  - Grapes
  - Gastric fluids

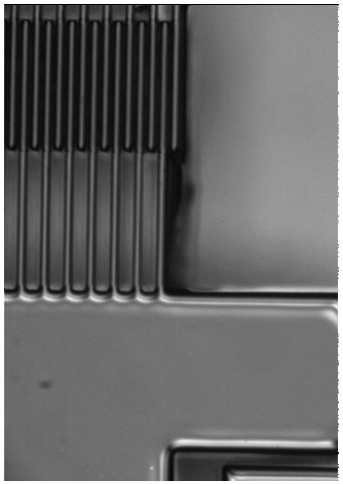
Energy Scavenging

- Solar is by far the most mature technology
- Been in commercial products as far back as 1960s

https://en.wikipedia.org/wiki/Solar_cell

Energy Scavenging: Mechanical

MEMS Generator



Jose Mur Miranda/
Jeff Lang

Vibration-to-Electric Conversion

~ 10mW

Power Harvesting Shoes

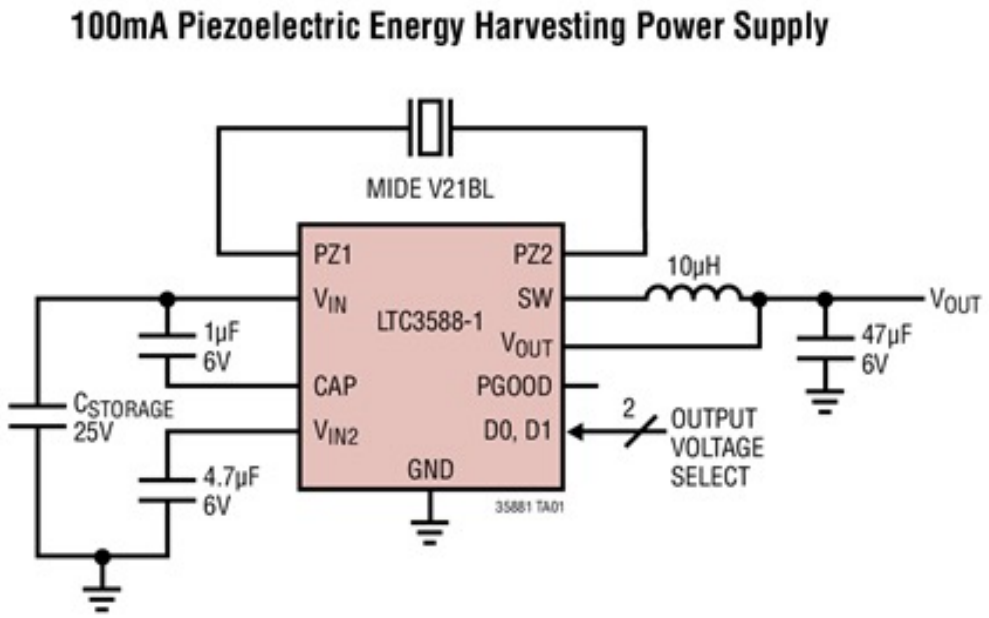
Joe Paradiso
(Media Lab)

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

~10mW

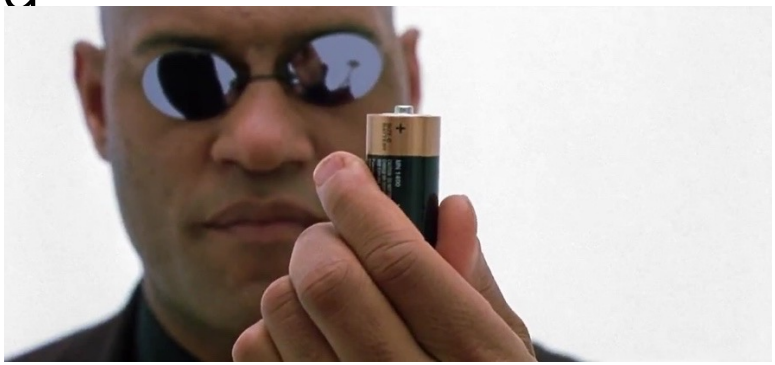
Energy Scavenging: Mechanical

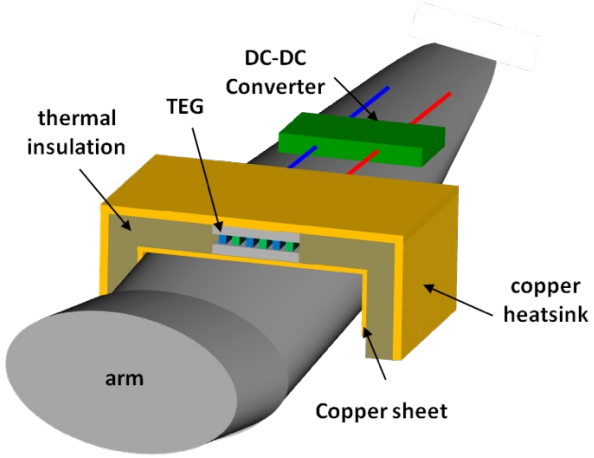
Optimized for :
collecting 60 Hz vibrations at low sub-g accelerations!

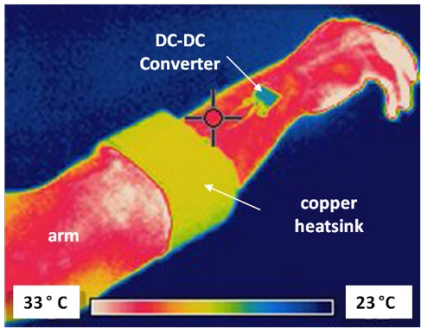


LTC3588

Low-Profile Wearable Body-Powered Thermoelectric Generator





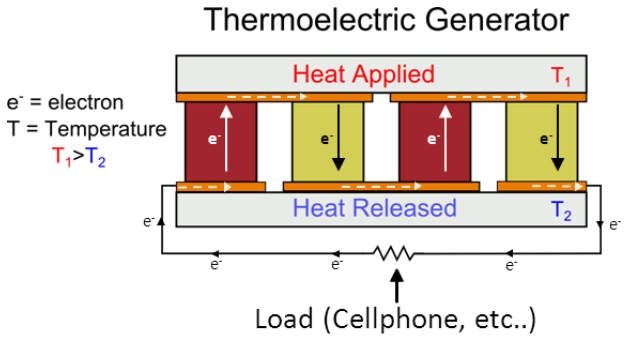


- Low profile, lightweight, conformal.
- Utilization of small temperature difference
- Utilization of natural convection for cooling

Credit: Krishna Settaluri MIT '2010

Energy harvesting

- Thermo-electric generator
  - Thermoelectric material converts temperature difference into voltage

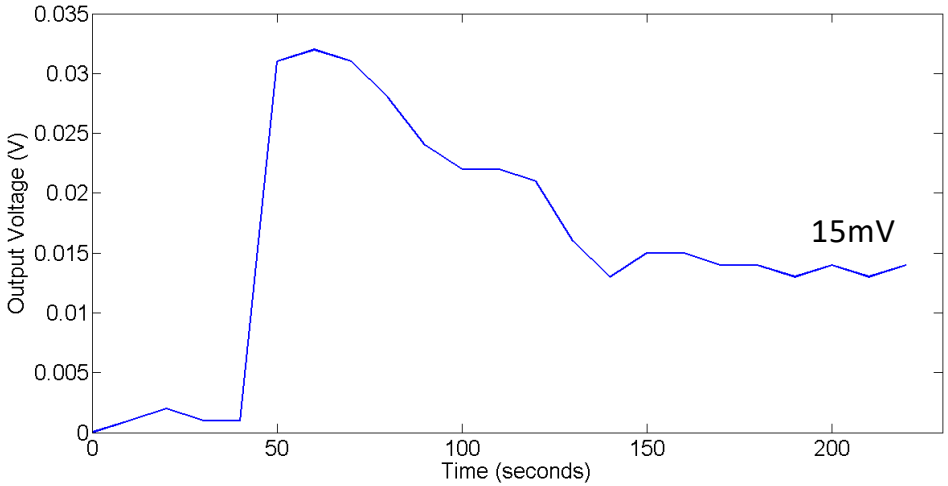


40 K temp difference
1.8 V @ 368 mA

https://www.adafruit.com/products/700
http://electronicdesign.com/content/content/73937/73937-fig2.gif

Experimental Results

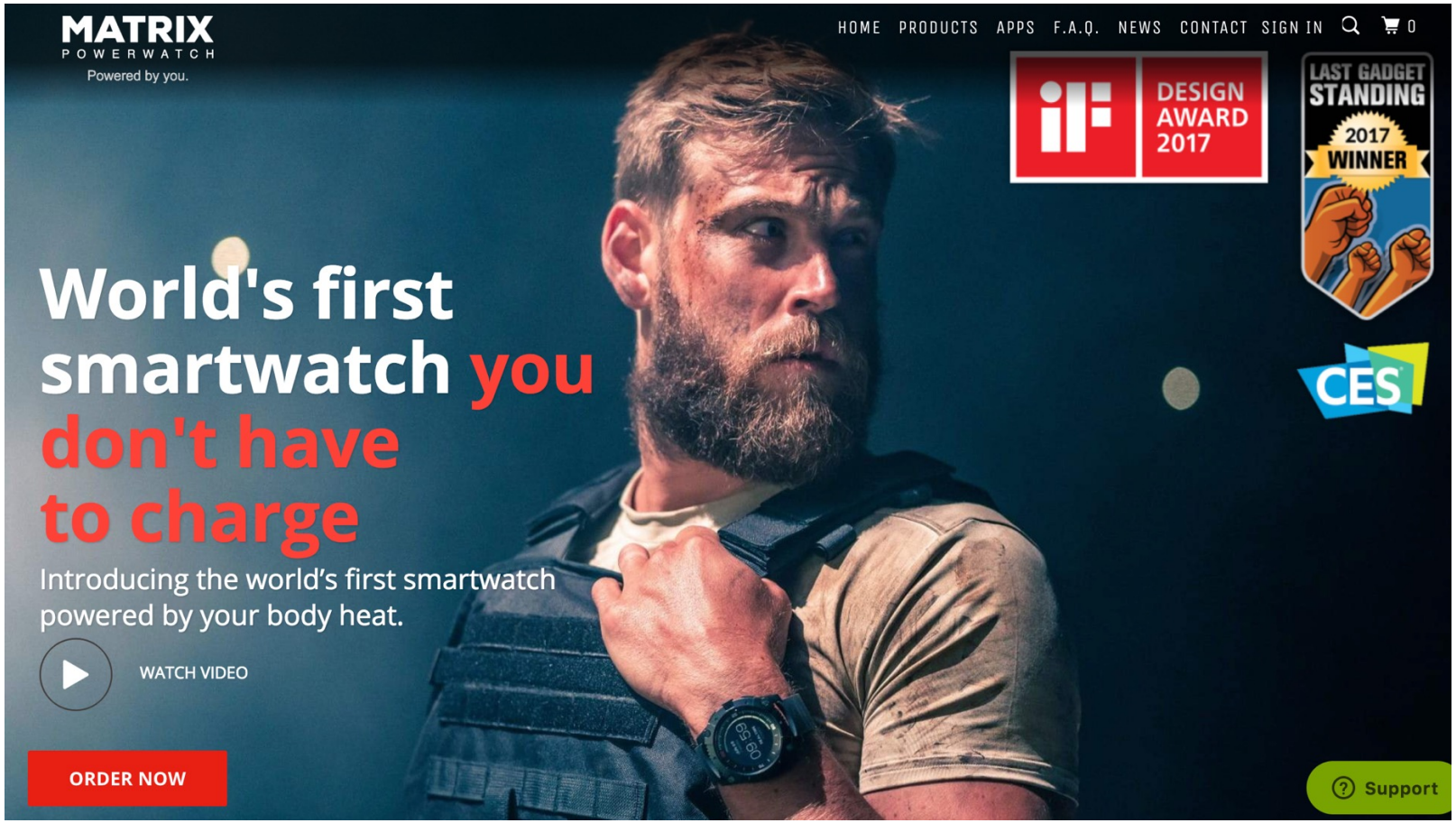
16 TEG Islands (2 TEG modules)



Optimal Electrical Load Resistance	33Ω (20Ω theoretical)
Optimized Power	11µW

Credit: Krishna Settaluri MIT '2010

Apparently Powered by Body Heat



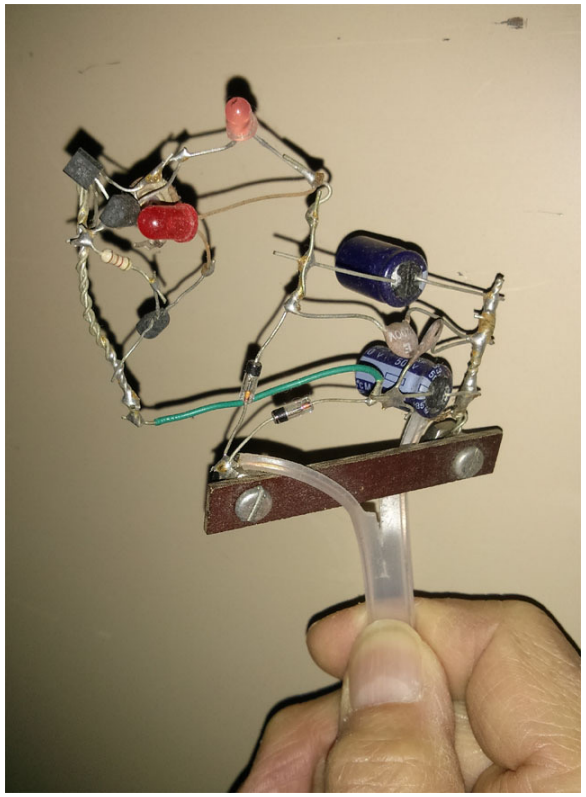
https://www.powerwatch.com/

Ambient RF

Prudential Center
FM Stations:
WZLX 100.7, WBMX 104.1, WMJX 106.7, and WXKS-FM 107.9, WBOS 92.9, WBQT 96.9, and WROR-FM 105.7.

Power output:
22,000 watts

Recovered:
~ 0.2 milliwatt



Ambient RF

TABLE V
HARVESTERS CHARGE AND DISCHARGE TIMES (t_c, t_d, RESPECTIVELY) FOR A SPECIFIED LOAD

Band	Wire					Tape				
	t_c (s) load independent	t_d (s) load dependant	t_{cycle} (s) load dependant	$P_{dc}(t_d)$ (μW)	$P_{dc}(t_{cycle})$ (μW)	t_c (s) load independent	t_d (s) load dependant	t_{cycle} (s) load dependant	$P_{dc}(t_d)$ (μW)	$P_{dc}(t_{cycle})$ (μW)
DTV	26	12	38	9.6	3	14	18	32	8.2	3.6
GSM900	14	10	24	11.5	4.8	8	13	21	14.4	5.5
GSM1800	43	15	58	7.7	2	22	27	49	5.2	2.4
3G v2	167	3	170	38.4	0.7	96	5	101	1.2	1.1
Multiband ΣV	43	7	50	66	2.3	-	-	-	-	-
Multiband ΣI	55	5	60	92.2	2	-	-	-	-	-

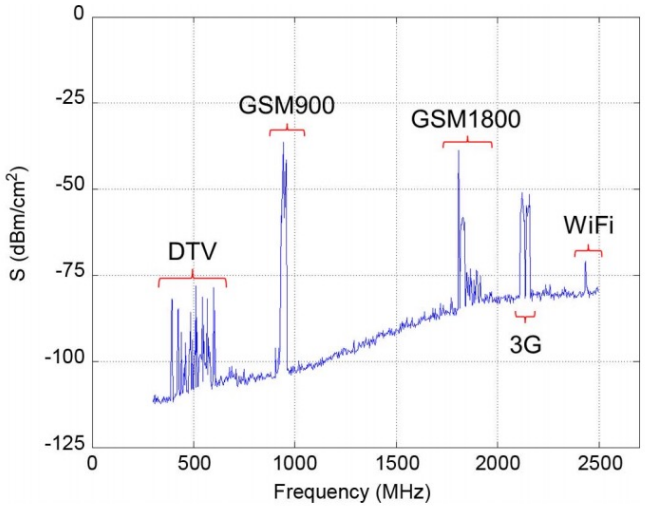


Fig. 1. Input RF power density measurements outside the Northfields London Underground station.

Energy to be had in the signals that are all around us

Previously not practical since even simple circuits used lots of power, but as transistors have scaled…gotten reasonable….couple with ASICs and you could be in business

Ambient RF Energy Harvesting in Urban and Semi-Urban Environments, Manuel Piñuela, Student Member, IEEE, Paul D. Mitcheson, Senior Member, IEEE, and Stepan Lucyszyn, Senior Member, IEEE 2013

WiFi Harvesting

Nature, 2019

LETTER

https://doi.org/10.1038/s41586-019-0892-1

Two-dimensional MoS_2-enabled flexible rectenna for Wi-Fi-band wireless energy harvesting

Xu Zhang[1], Jesús Grajal[2], Jose Luis Vazquez-Roy[3], Ujwal Radhakrishna[1], Xiaoxue Wang[4], Winston Chern[1], Lin Zhou[1], Yuxuan Lin[1], Pin-Chun Shen[1], Xiang Ji[1], Xi Ling[5], Ahmad Zubair[1], Yuhao Zhang[1], Han Wang[6], Madan Dubey[7], Jing Kong[1], Mildred Dresselhaus[1,8] & Tomás Palacios[1]*

The mechanical and electronic properties of two-dimensional materials make them promising for use in flexible electronics[1–3]. Their atomic thickness and large-scale synthesis capability could enable the development of 'smart skin'[1,3–5], which could transform ordinary objects into an intelligent distributed sensor network[6]. However, although many important components of such a distributed electronic system have already been demonstrated (for example, transistors, sensors and memory devices based on two-dimensional materials[1,2,4,7]), an efficient, flexible and always-on energy-harvesting solution, which is indispensable for self-powered systems, is still missing. Electromagnetic radiation from Wi-Fi systems operating at 2.4 and 5.9 gigahertz[8] is becoming increasingly ubiquitous and would be ideal to harvest for powering future distributed electronics. However, the high frequencies used for Wi-Fi communications have remained elusive to radiofrequency harvesters (that is, rectennas) made of flexible semiconductors owing to their limited transport properties[9–12]. Here we demonstrate an atomically thin and flexible rectenna based on a MoS_2 semiconducting–metallic-phase heterojunction with a cutoff frequency of 10 gigahertz, which represents an improvement in speed of roughly one order of magnitude compared with current state-of-the-art flexible rectifiers[9–12]. This flexible MoS_2-based rectifier operates up to the X-band[8] (8 to 12 gigahertz) and covers most of the unlicensed industrial, scientific and medical radio band, including the Wi-Fi channels. By integrating the ultrafast MoS_2 rectifier with a flexible Wi-Fi-band antenna, we fabricate a fully flexible and integrated rectenna that achieves wireless energy harvesting of electromagnetic radiation in the Wi-Fi band with zero external bias (battery-free). Moreover, our MoS_2 rectifier acts as a flexible mixer, realizing frequency conversion beyond 10 gigahertz. This work provides a universal energy-harvesting building block that can be integrated with various flexible electronic systems.

that exhibit a cutoff frequency of 1.6 GHz[11]. However, the random distribution of particle sizes and separation distances results in a low on/off current ratio and unreliable turn-on voltage, which deteriorates their rectification performance and reliability for large-scale production. In addition, almost all the above methods use a vertical structure to increase the effective device area and thereby to reach a sufficiently high on-current, I_{on}. However, in such a structure, the top and bottom electrodes of the diode inevitably form a parallel-plate capacitor with large parasitic capacitance, which considerably hinders its high-speed applications. Lateral p–intrinsic–n (PIN) diodes made from single-crystal silicon[18] and germanium[19] nanomembranes can be fabricated on flexible substrates for operation at 10 GHz. However, the use of PIN diodes is usually limited to RF switches and power attenuators, and such diodes are not applicable to energy harvesting[8]. Besides, the high cost of single-crystal silicon and germanium nanomembranes, as well as the complexity of their materials and processing, render them unfavourable for practical applications.

Nowadays, Wi-Fi is becoming increasingly ubiquitous in both indoor and outdoor environments and provides an abundant source of always-on RF energy. It would be highly desirable if wearable electronics could directly harvest the radiation in the Wi-Fi band (2.4 GHz and 5.9 GHz) for wireless charging. However, owing to the aforementioned challenges, a flexible RF rectifier that is fast enough to achieve Wi-Fi-band wireless energy harvesting has not been demonstrated. In this work, we present an atomically thin and fully flexible MoS_2-based rectifier with a cutoff frequency of 10 GHz at zero external bias, using a self-aligned fabrication technique. MoS_2 is an emerging two-dimensional (2D) semiconductor with high mechanical robustness and low-cost large-scale synthesis technology[2,20,21]. By patterning MoS_2 into a metallic–semiconducting (1T/1T′–2H) phase heterostructure[22] (Fig. 1a), we demonstrate a lateral Schottky diode with junction capacitance lower than 10 fF. In combination with a reduc-

Grape Power

I guess

Grape Juice Voltage

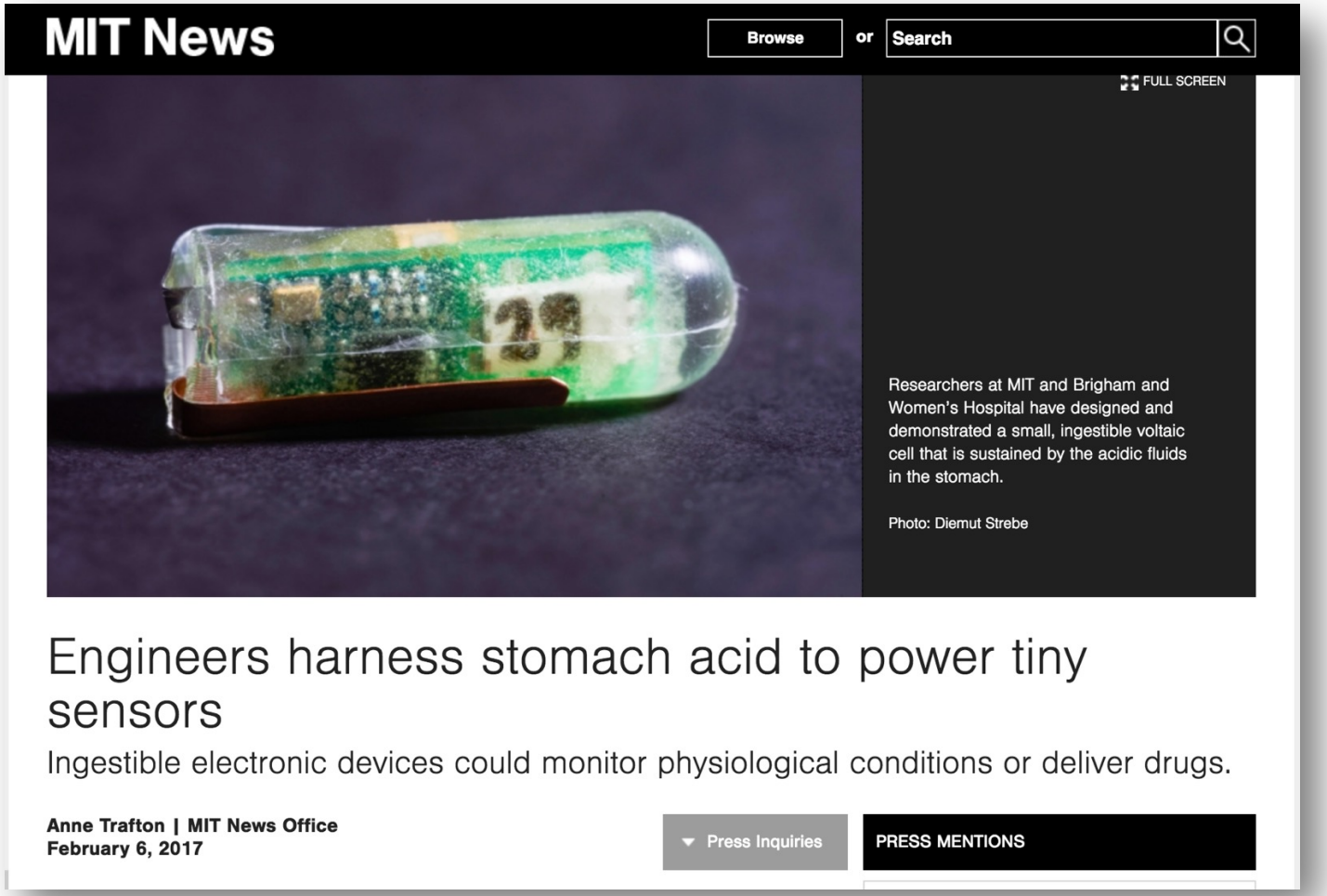
Copper penny
Zinc screw

Newman's Own
Grape Juice

Infinite Power!

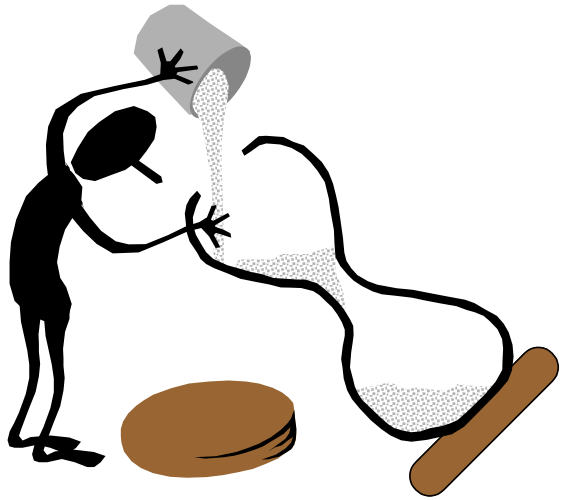
If we ignore huge amount of energy required to grow grapes that have been so selectively bred to the point of absurdity that they will die at the slightest environmental deviations, process said grapes, ship said grape juice, extract petroleum from the earth to synthesize the plastics that form the container, maintain a significant geopolitical presence to keep petroleum at low prices, as well as manufacture fertilizer to make the plants in the first place and to clear the land, etc…

More seriously though…



Nadeau et al…2017 Nature Biomedical Engineering

Done…Final Project Time… See you in lab!



- Final project represents 72 **good** hours of credit, so you should average 2-3 hours/day of good work on your project assuming you give yourself the occasional day off…

- Do not try to do 72 hours in three days. While technically possible, it is not likely to happen