How Computers Work
Lecture 11

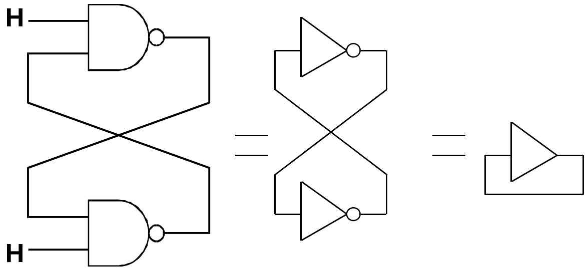
Introduction to the Physics of Computation

Recall the *essence* of data transmission:

Q: What form does information take during transmission?

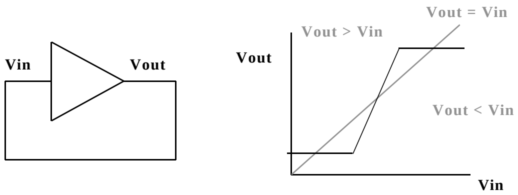
A: Energy

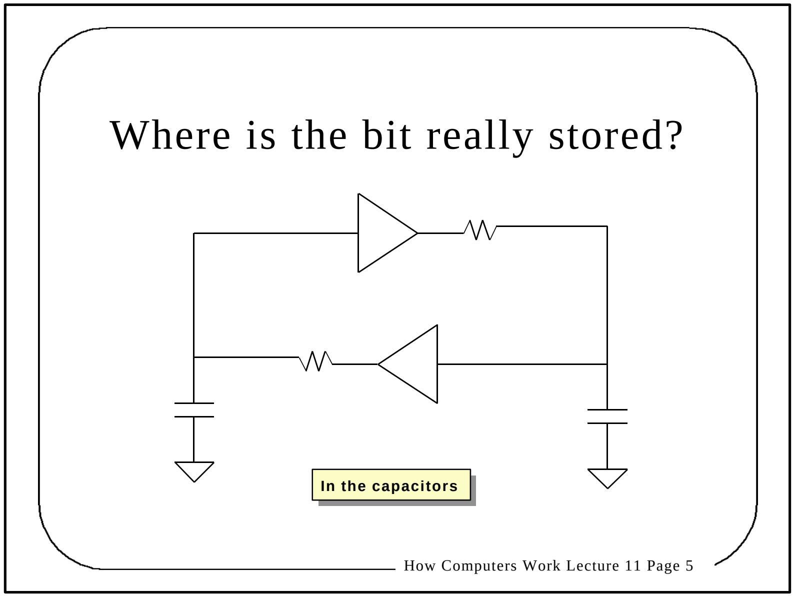
Recall the SR Flop in the Store State



Is this an adequate way to explain storage of a bit?

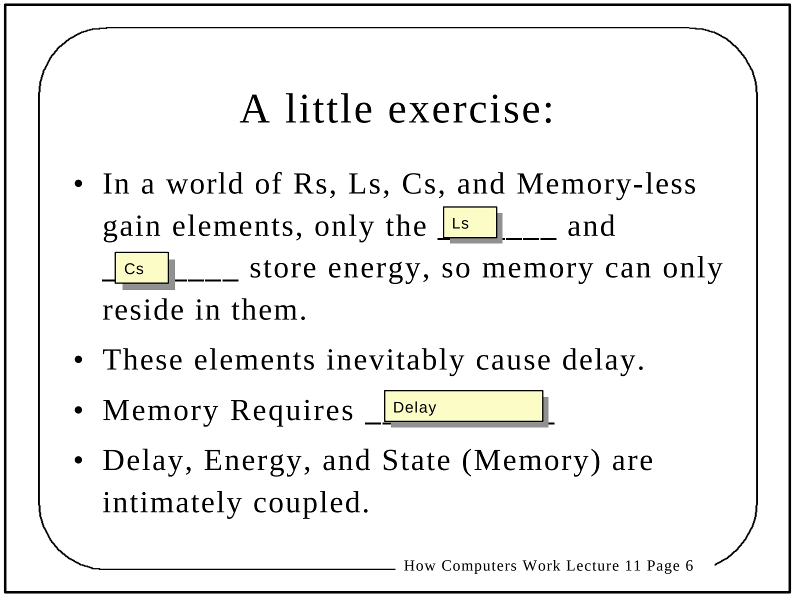
- A:___No___





Where is the bit really stored?

In the capacitors

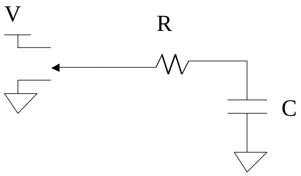


A little exercise:

- In a world of Rs, Ls, Cs, and Memory-less gain elements, only the _Ls_____ and _Cs_____ store energy, so memory can only reside in them.
- These elements inevitably cause delay.
- Memory Requires __Delay__
- Delay, Energy, and State (Memory) are intimately coupled.

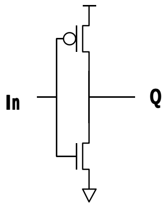
(Linear) Capacitor Energetics

- Q: How much energy does a capacitor charged to voltage V store?
  - A: $(1/2)\ C\ V^2$
- Q: If I charge an originally discharged capacitor to a voltage V through a resistor, how much energy is dissipated in the resistor?
  - A: $(1/2)\ C\ V^2$
- Q: How much total energy is lost charging (to V) and discharging a capacitor?
- A: $C\ V^2$



Power Loss in the CMOS Inverter

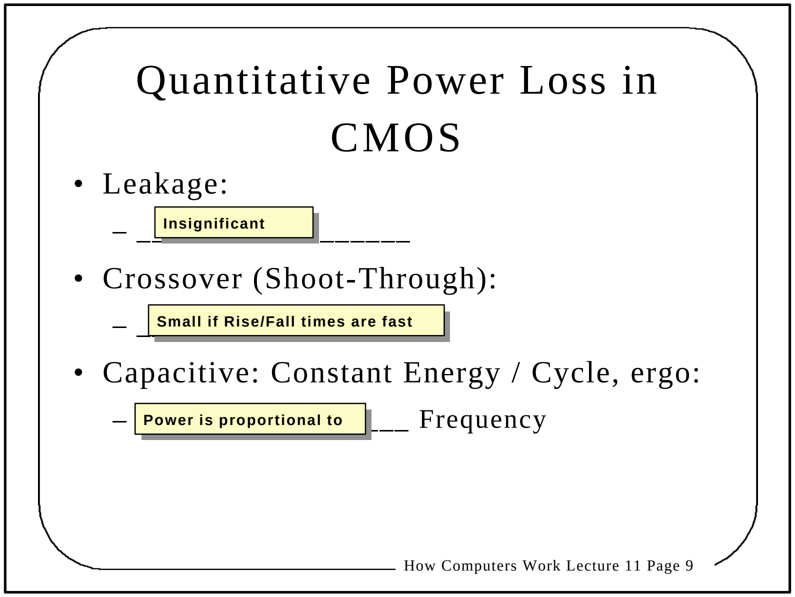
Power is lost due to:



1) Leakage

2) Shoot-Through

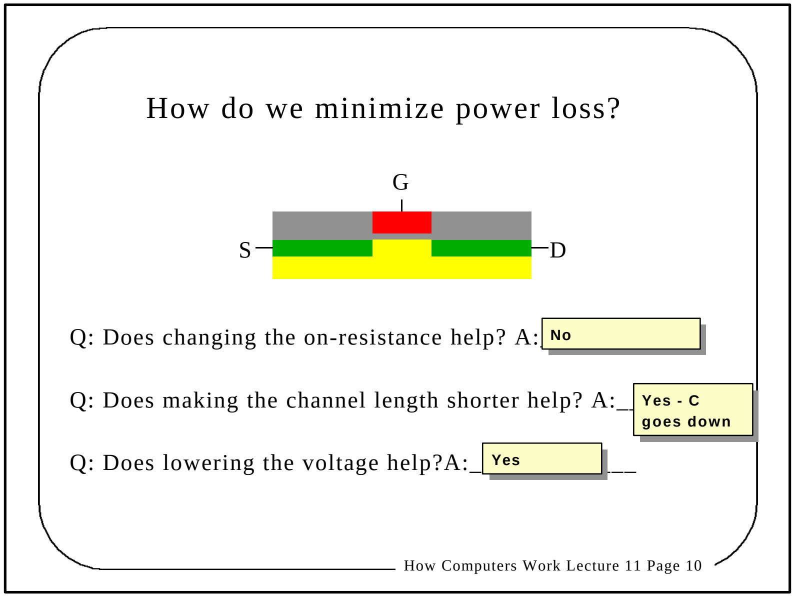
3) Capacitive Charge/Discharge



Quantitative Power Loss in CMOS

- Leakage:
  - __Insignificant______
- Crossover (Shoot-Through):
  - _Small if Rise/Fall times are fast
- Capacitive: Constant Energy / Cycle, ergo:
  - Power is proportional to___ Frequency

How Computers Work Lecture 11 Page 9



How do we minimize power loss?

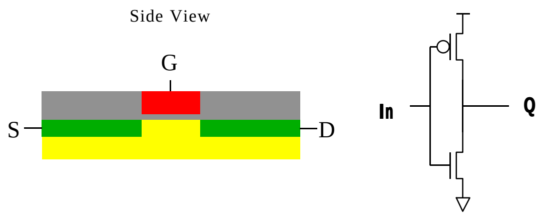
Q: Does changing the on-resistance help? A: No

Q: Does making the channel length shorter help? A:_Yes - C goes down

Q: Does lowering the voltage help?A:_Yes__

How Computers Work Lecture 11 Page 10

But do MOS transistors turn on enough at low voltages?



A: ___Yes_____ as long as you doctor the channel a bit, thus lowering the turn-on threshold.

How about speed?

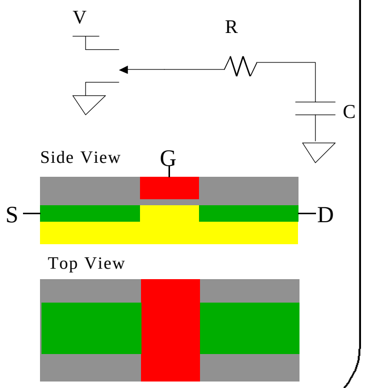
Unlike power dissipation, lowering R does lower Tpd.

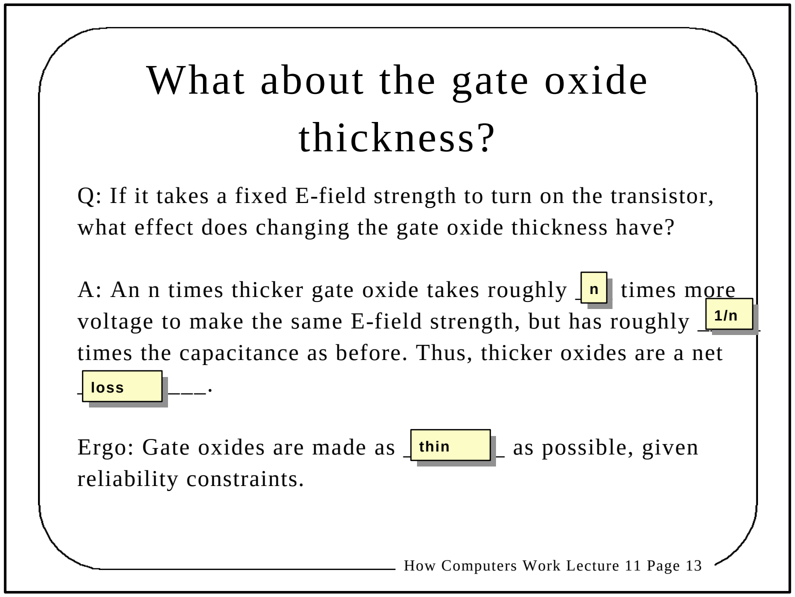
Q: How do we lower R?

A: Make FET channel _Shorter_

Q: Does making the FET channel wider help?

A:__No__ because it raises C while lowering R.



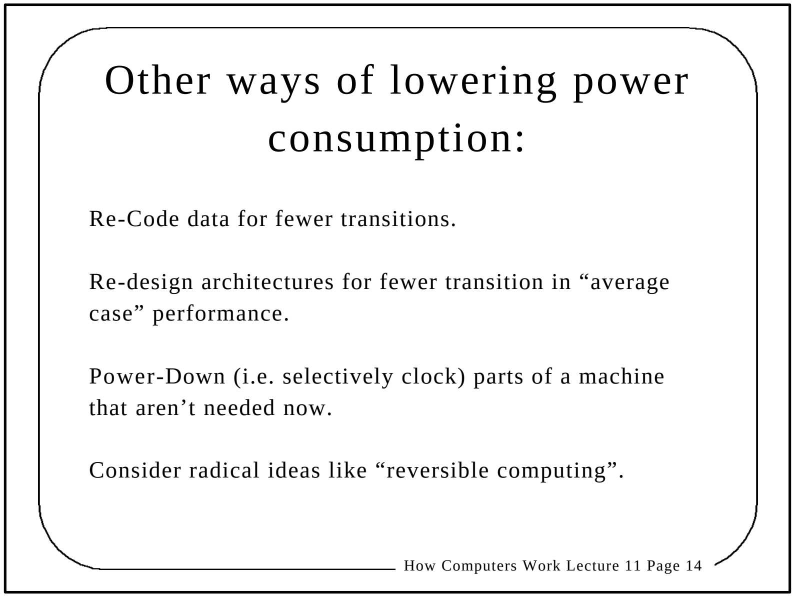


What about the gate oxide thickness?

Q: If it takes a fixed E-field strength to turn on the transistor, what effect does changing the gate oxide thickness have?

A: An n times thicker gate oxide takes roughly _n_ times more voltage to make the same E-field strength, but has roughly _1/n_ times the capacitance as before. Thus, thicker oxides are a net _loss___.

Ergo: Gate oxides are made as _thin_ as possible, given reliability constraints.



Other ways of lowering power consumption:

Re-Code data for fewer transitions.

Re-design architectures for fewer transition in “average case” performance.

Power-Down (i.e. selectively clock) parts of a machine that aren’t needed now.

Consider radical ideas like “reversible computing”.

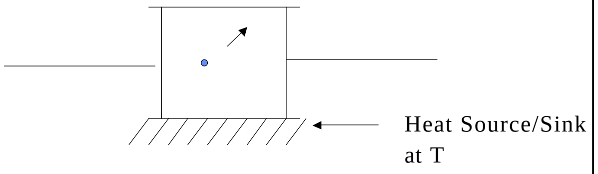
Reversible Computing?

Q1: How little energy can be used to represent a bit?
Q2: Is there a minimum energy is takes to do computation?

Intuitive (in this case, wrong) answers:

A1: It can take arbitrarily little energy to represent a bit
A2: Computation must consume power

The smallest energy system we know that can represent a bit:

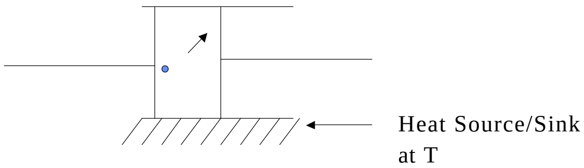


1 particle of gas exists in a 2-piston iso-thermal cylinder at temperature T.

Q: What is the kinetic energy of the particle?

A:__T__________________

Q: Does it take energy to slowly compress the gas to the left side?

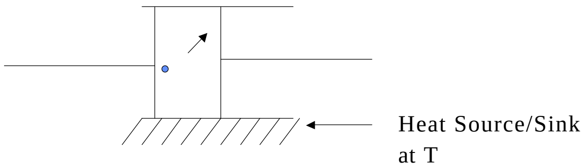


A: Yes

Q: Is the kinetic energy of the particle any different?

A: No

Then why did it take energy?

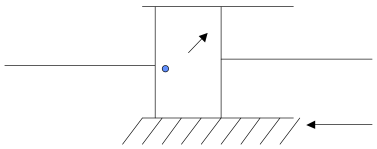


A: Because the particle was bouncing against the piston

Q: Where did this energy go?

A1: Into the heat sink
A2: Into information!

How many bits of information have we created?



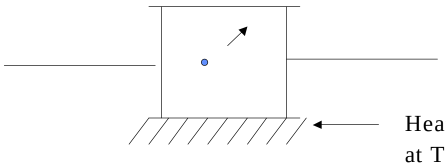
Heat Source/Sink
at T

A:__1

Q: How much energy did this take:

A: $\log_e(2)\ k\ T$

Can we get this energy back?



Heat Source/Sink
at T

A:__Yes_____!!!!!! IF WE DO IT SLOWLY!!!!!

Summary:

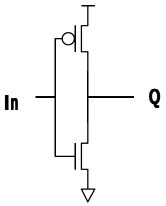
- Slow (i.e. reversible) thermodynamic processes can recover the energy put into creating a bit.
- Fast (i.e. irreversible) processes loose part of this energy.
- We can recover an arbitrarily high fraction of the energy put in by going slowly enough.
- There's nothing special about the isothermal heat sink - adiabatic (insulated) cylinders work too, it's the speed that's important.

But that's bit storage. How about computing?

- A: Computing is nothing more than creating bits whose value is determined by examining other bits.
- Examination of bits is energetically free. It is their creation and destruction that is tied to energy.
- When destroying a bit, we can do so *reversibly* (i.e. slowly) or *irreversibly* (i.e. fast).

A typical CMOS Inverter

- Has fixed power supply rails.
- Is driven as fast as possible.
- Operates non-reversibly.
- Throws away its bit energy

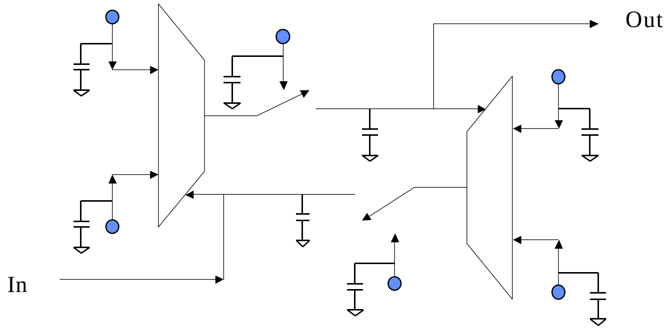


But there’s another way:

- Reversible Computing:
  - To Create a bit:
    - Start with two power supply rails at the same voltage.
    - Connect the parasitic capacitance to the appropriate power supply rail.
    - Slowly separate the power supply rail voltages, raising one and lowering the other.
  - To (reversibly) destroy a bit:
    - Start with two power supply rails separated by some voltage.
    - Connect the (already charged) parasitic capacitance to the appropriate rail.
    - Slowly bring the power supply rails together.
    - Except for (arbitrarily small) resistive losses, the power supply can recover all of the energy!

An example: Younis and Knight's SCRL

● = Clock Lines



An interesting consequence:

We must remember the value of a bit in order to recover it's energy, so every computation must be reversed after it is done.

Thus: For any computation (e.g. AND) that destroys input information, we must remember enough input variables
to be able to UNDO or REVERSE the computation, and recover the energy that would be otherwise lost due to the DESTRUCTION OF BITS.

This is sometimes practical and sometimes not. By being clever and only throwing away bits when it is very inconvenient to remember them, we can make reversible computation practical.

Reversible Computing?
Some real answers

Q1: How little energy can be used to represent a bit?
Q2: Is there a minimum energy is takes to do computation?

Real answers:

A1: k T $\log_e(2)$ is the minimum energy to reliably store a bit
CMOS gates typically use 10^8 kT per bit
RNA duplication typically uses 100 kT per bit

A2: Computation does NOT need to consume power, as long as it is done reversibly (i.e. slowly enough, and without destroying information)

Related Trivia : The awake human brain consumes approximately 40 Watts of power.

To Learn More Read:

- Feynman Lectures on Computation
- http://www.ai.mit.edu/people/tk/lowpower/crl.ps
- http://www.ai.mit.edu/people/tk/lowpower/low94.ps
- “Thermodynamics of Computation - A Review” Charles H. Bennett, *International Journal of Theoretical Physics* **21**, 905(1982)