

CS10: The Beauty and Joy of Computing

**Lecture #23
Future of Computing**


bjc

UC Berkeley EECS
Summer Instructor
Ben Chun

2012-07-31

INDIAN POWER GRID FAILURE

Yesterday 370M people in Northern India were without electricity for about 6 hours after a grid failure. (Similar problems have happened all over the world, but this was the biggest by far to date.)




<http://www.thehindu.com/news/national/article3702075.ece>

bjc

Lecture Overview

- Where will today's computers go?
- Quantum Computing
- DNA Computing
- Biological Machines
- Optical Computing
- Smart Grid + Energy



Chun, Summer 2012

UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing [2]

bjc

Computer Technology - Growth!

<ul style="list-style-type: none"> Processor <ul style="list-style-type: none"> Speed 2x / 2 years (since '71) 100X performance last decade Soon: 4 GHz, 32 Cores Memory (DRAM) <ul style="list-style-type: none"> Capacity: 2x / 2 years (since '96) 64x size last decade. Soon: 128 GibiBytes Disk <ul style="list-style-type: none"> Capacity: 2x / 1 year (since '97) 250X size last decade. Soon: 16 TeraBytes 	<p>Kilo (10³) & Kibi (2¹⁰) ↓ Mega (10⁶) & Mebi (2²⁰) ↓ Giga (10⁹) & Gibi (2³⁰) ↓ Tera (10¹²) & Tebi (2⁴⁰) ↓ Peta (10¹⁵) & Pebi (2⁵⁰) ↓ Exa (10¹⁸) & Exbi (2⁶⁰) ↓ Zetta (10²¹) & Zebi (2⁷⁰) ↓ Yotta (10²⁴) & Yobi (2⁸⁰)</p>
---	--

Chun, Summer 2012

UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing [3]

bjc

Kilo, Mega, Giga, Tera, Peta, Exa, Zetta, Yotta

- Kid meets giant Texas people exercising zen-like yoga. – Rolf O
- Kind men give ten percent extra, zestfully, youthfully. – Hava E
- Kissing Mentors Gives Testy Persistent Extremists Zealous Youthfulness. – Gary M
- Kindness means giving, teaching, permeating excess zeal yourself. – Hava E
- Killing messengers gives terrible people exactly zero, yo
- Kindergarten means giving teachers perfect examples (of) zeal (&) youth
- Kissing mediocre girls/guys teaches people (to) expect zero (from) you
- Kissing Mel Gibson, Teddy Pendergrass exclaimed: "Zesty, yo!" – Dan G

Chun, Summer 2012

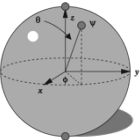
UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing [4]

bjc

Quantum Computing (1)

- Proposed computing device using quantum mechanics**
 - This field in its infancy...
- Normally: bits, which are either 0 or 1**
- Quantum: qubits, either 0, 1 or "quantum superposition" of these**
 - This is the key idea

- If you have 2 bits, they're in exactly one of these:
 - 00, 01, 10 or 11
- If you have 2 qubits, they're in ALL these states with varying probabilities



A Bloch sphere is the geometric representation of 1 qubit

en.wikipedia.org/wiki/Quantum_computer

Chun, Summer 2012

UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing [5]

bjc

Quantum Computing (2)

- Imagine a problem with these four properties:**
 - The only way to solve it is to guess answers repeatedly and check them,
 - There are n possible answers to check,
 - Every possible answer takes the same amount of time to check, and
 - There are no clues about which answers might be better: generating possibilities randomly is just as good as checking them in some special order.
- ...like trying to crack a password from an encrypted file**
- A normal computer**
 - would take (in the worst case) n steps
- A quantum computer**
 - can solve the problem in steps proportional to \sqrt{n}
- Why does this matter?**

Chun, Summer 2012

UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing [7]


Quantum Computing (3)

- **Say the password is exactly 72 bits (0/1)**
- **That's 2^{72} possibilities**
- **Let's say our Mac lab attacked the problem**
 - 30 machines/lab * 8 cores/machine * 3 GHz (say 3 billion checks per second/core)
 - = 720,000,000,000 checks/sec/lab
 - = 720 Gchecks/sec/lab
- **Regular computers**
 - 2^{72} checks needed / 720 Gchecks/sec/lab
 - = 6.6 billion sec/lab
 - = 208 years/lab
- **72-qubit quantum computers in time α to $\sqrt{2^{72}} = 2^{36}$**
 - 2^{36} checks needed / 720 Gchecks/sec/lab
 - = 0.1 sec/lab

Chan, Summer 2012
 UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing (8)

DNA Computing

- **Proposed computing device using DNA to do the work**
 - Take advantage of the different molecules of DNA to try many possibilities at once
 - Ala parallel computing
 - Also in its infancy
- **In 2004, researchers claimed they built one**
 - Paper in "Nature"
 - en.wikipedia.org/wiki/DNA_computing




Chan, Summer 2012
 UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing (9)

www.eecs.berkeley.edu/~maharbiz/Cyborg.html

Biological Machines


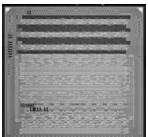
- **Michel Maharbiz and his team at Cal have wired insects (here a giant flower beetle) and can control flight**
 - Implanted as Pupa
- **Vision**
 - Imagine devices that can collect, manipulate, store and act on info from environment



Chan, Summer 2012
 UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing (10)

Optical Computing

- **Electro-optical hybrids**
- **Pure optical**
- **No-memory communication for multi-core processors!**





<http://web.mit.edu/newsoffice/topic/optical-computing.html>

Chan, Summer 2012
 UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing (11)

Smart Grid + Energy


- **Arguably the most important issue facing us today is climate change**
- **Computing can help**
- **Old: generators "broadcast" power**
- **New: "peer-to-peer", with optimal routing**
 - From: ability (to power)
 - To: according to need
- **Energy**
 - Computing helps with climate modeling and simulation
 - "Motes", or "Smart dust" are small, networked computing measurement devices
 - E.g., could sense no motion + turn lights off



Chan, Summer 2012
 UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing (12)

Summary

- **What a wonderful time we live in; we're far from done**
 - What about privacy?
- **Find out the problem you want to solve**
 - Computing can and will help us solve it
- **We probably can't even imagine future software + hardware breakthroughs**



Chan, Summer 2012
 UC Berkeley CS10 "The Beauty and Joy of Computing": Future of Computing (13)