What does the Future Hold?

Hakim Weatherspoon

CS 3410, Spring 2013

Computer Science

Cornell University

Announcements

How to improve your grade?

Submit a course evaluation and drop lowest homework score

- To receive credit, Submit before Tuesday, May 7th

Announcements

Final Project

Design Doc sign-up via CMS

sign up Sunday, Monday, or Tuesday

May 5th, 6th, or 7th

Demo Sign-Up via CMS.

sign up Tuesday, May 14th

or Wednesday, May 15th

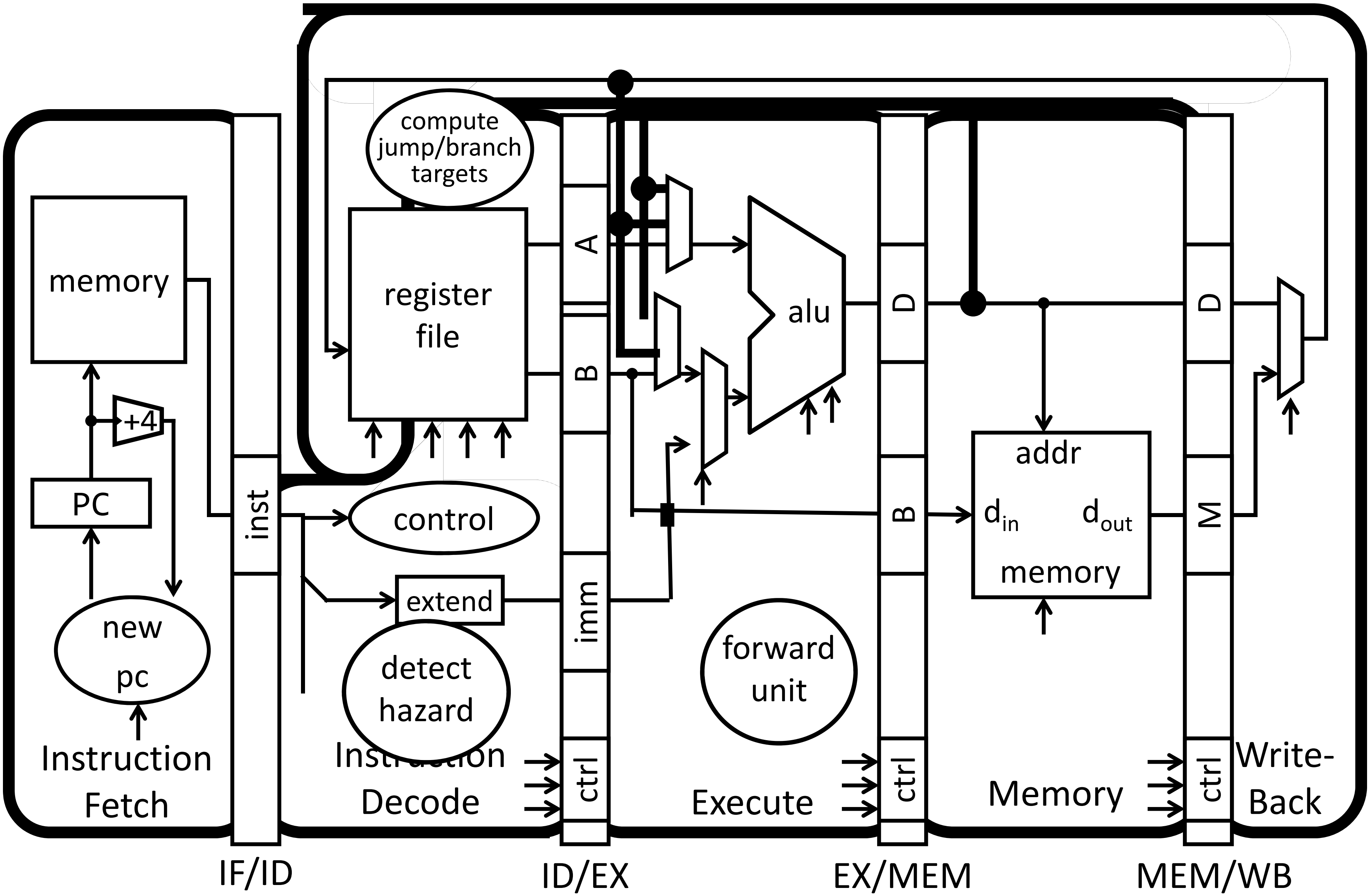
CMS submission due:

- Due 6:30pm Wednesday, May 15th

Big Picture about the Future

Big Picture

How a processor works? How a computer is organized?



What’s next?

More of Moore

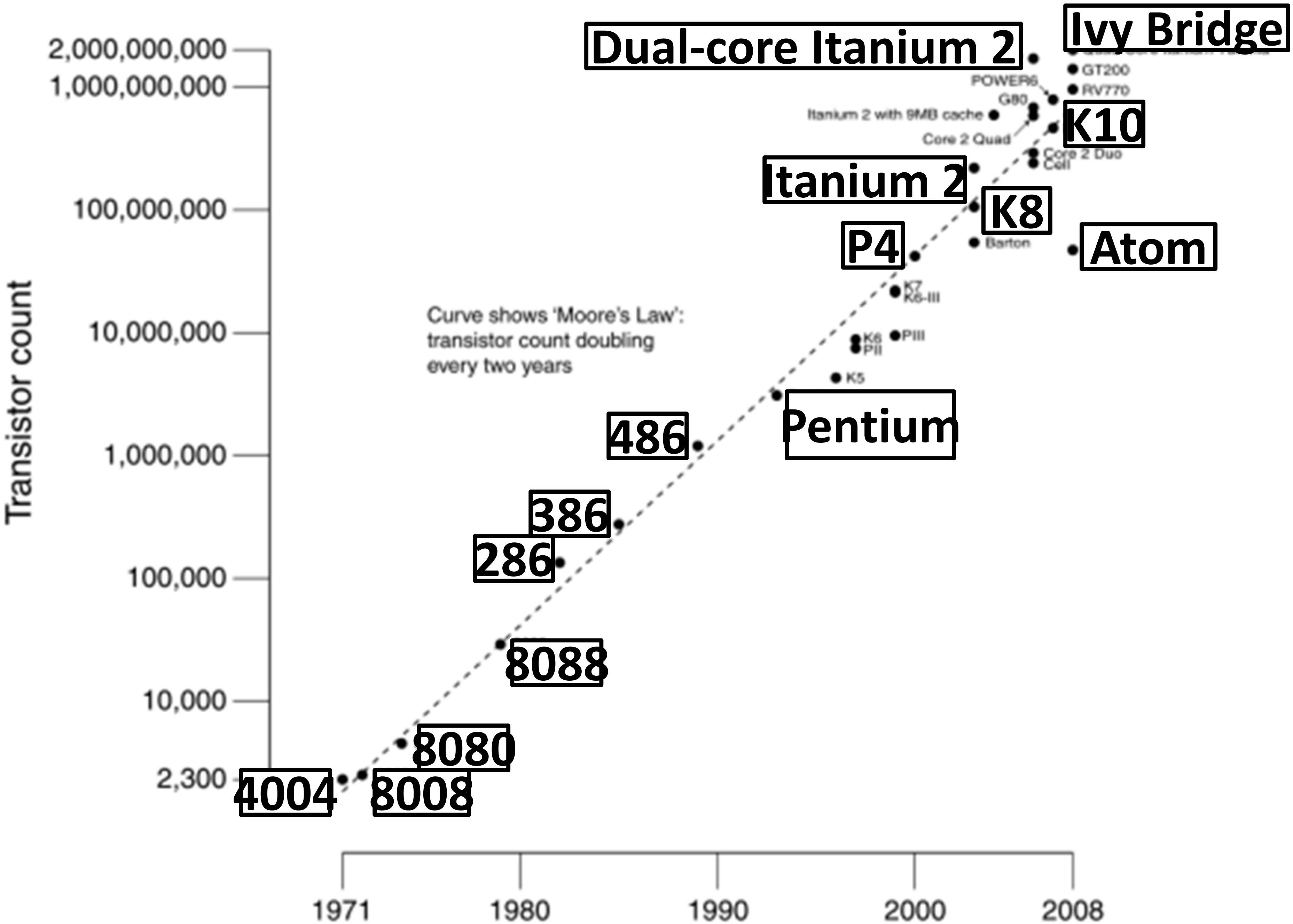
Moore's Law

Moore's Law introduced in 1965

- Number of transistors that can be integrated on a single die would double every 18 to 24 months (i.e., grow exponentially with time).

Amazingly visionary

- 2300 transistors, 1 MHz clock (Intel 4004) - 1971
- 16 Million transistors (Ultra Sparc III)
- 42 Million transistors, 2 GHz clock (Intel Xeon) – 2001
- 55 Million transistors, 3 GHz, 130nm technology, 250mm2 die (Intel Pentium 4) – 2004
- 290+ Million transistors, 3 GHz (Intel Core 2 Duo) – 2007
- 731 Million transistors, 2-3Ghz (Intel Nehalem) – 2009
- 1.4 Billion transistors, 2-3Ghz (Intel Ivy Bridge) – 2012



Ivy Bridge
Dual-core Itanium 2
GT200
POWER6
RV770
G80
Itanium 2 with 9MB cache
K10
Core 2 Quad
Core 2 Duo
Cell
Itanium 2
K8
Barton
P4
Atom
K7
K6-III
K6
PII
PIII
K5
Pentium
486
386
286
8088
8080
4004
8008
Curve shows 'Moore's Law': transistor count doubling every two years
Transistor count
2,000,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
2,300
1971
1980
1990
2000
2008

Why Multicore?

Moore's law

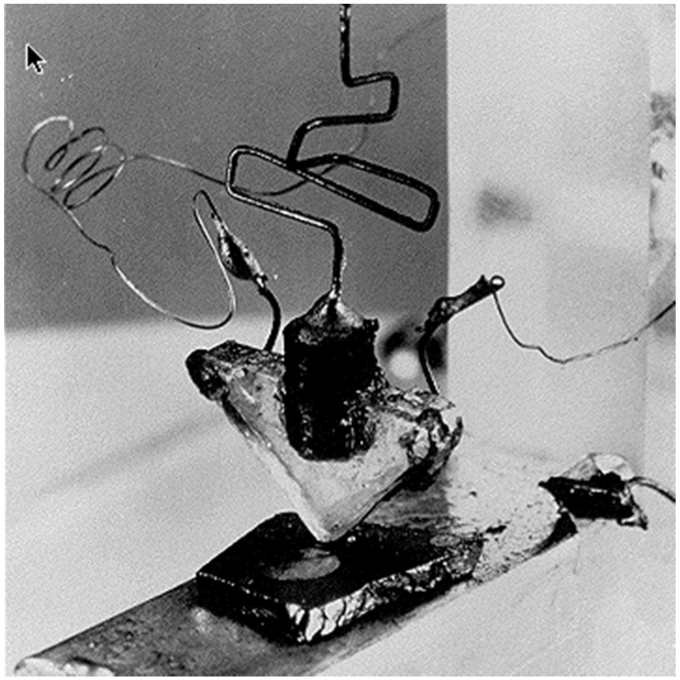
- A law about transistors
- Smaller means more transistors per die
- And smaller means faster too

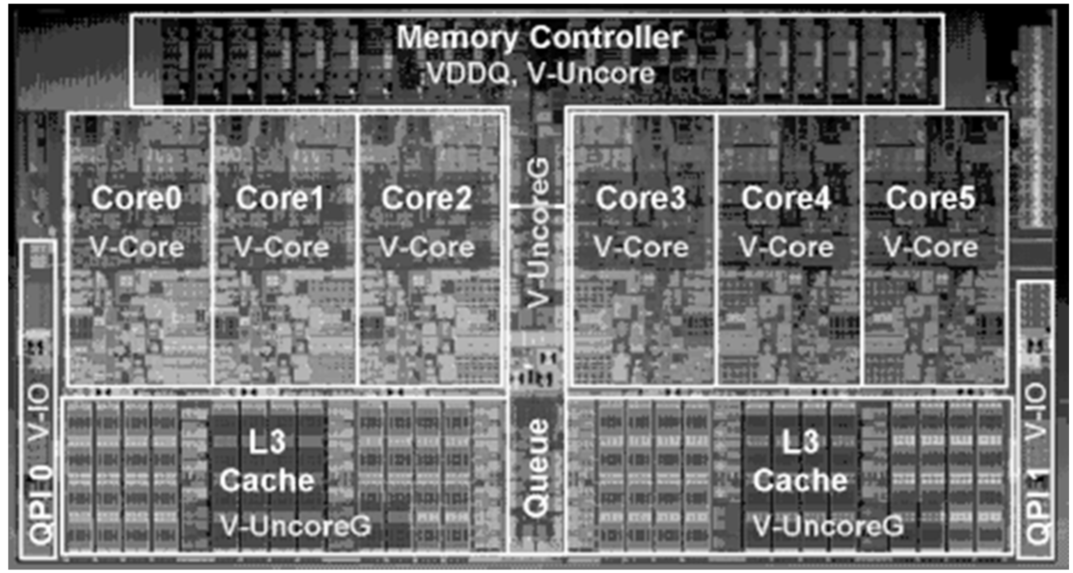
But: Power consumption growing too…

What to do with all these transistors?

Multi-core

Multi-core





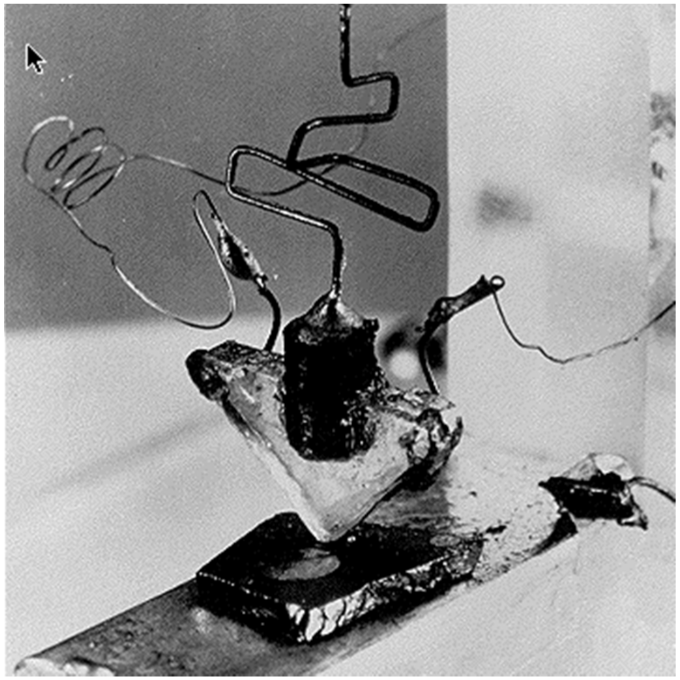
http://www.theregister.co.uk/2010/02/03/intel_westmere_ep_preview/

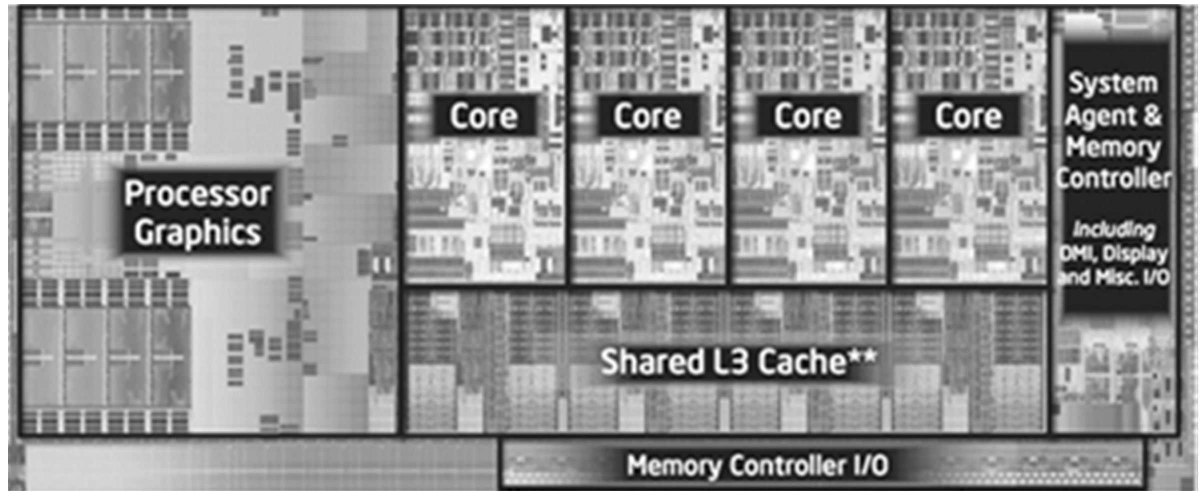
The first transistor

- on a workbench at
  AT&T Bell Labs in 1947
- Bardeen, Brattain, and Shockley

- An Intel Westmere
  - 1.17 billion transistors
  - 240 square millimeters
  - 32 nanometer: transistor gate width
  - Six processing cores
  - Release date: January 2010

Multi-core





http://forwardthinking.pcmag.com/none/296972-intel-releases-ivy-bridge-first-processor-with-tri-gate-transistor

The first transistor

- on a workbench at
  AT&T Bell Labs in 1947
- Bardeen, Brattain, and Shockley

- An Intel Ivy Bridge
  - 1.4 billion transistors
  - 160 square millimeters
  - 22 nanometer: transistor gate width
  - Up to eight processing cores
  - Release date: April 2012

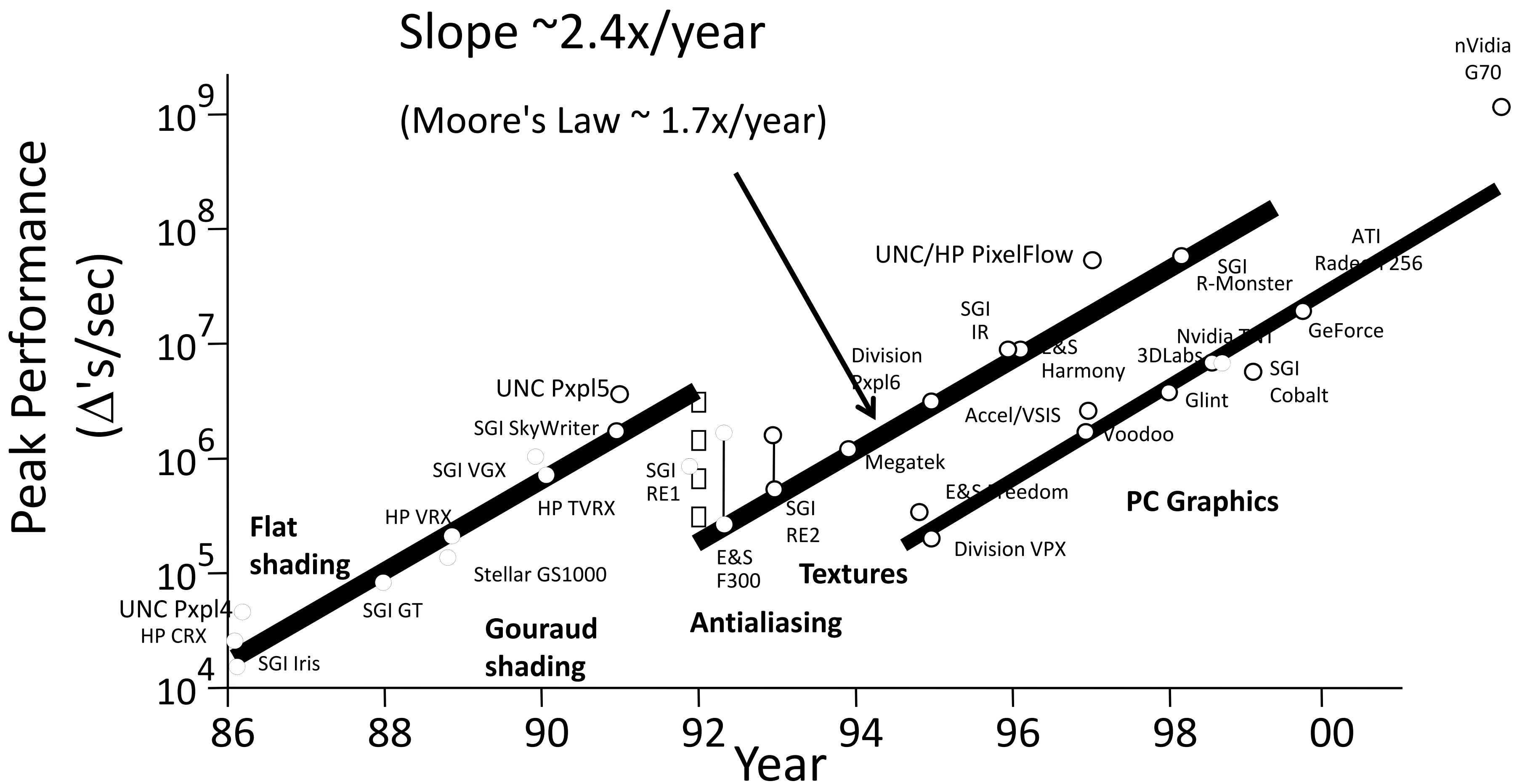
What to do with all these transistors?

Many-core

and Graphical Processing units

Faster than Moore’s Law

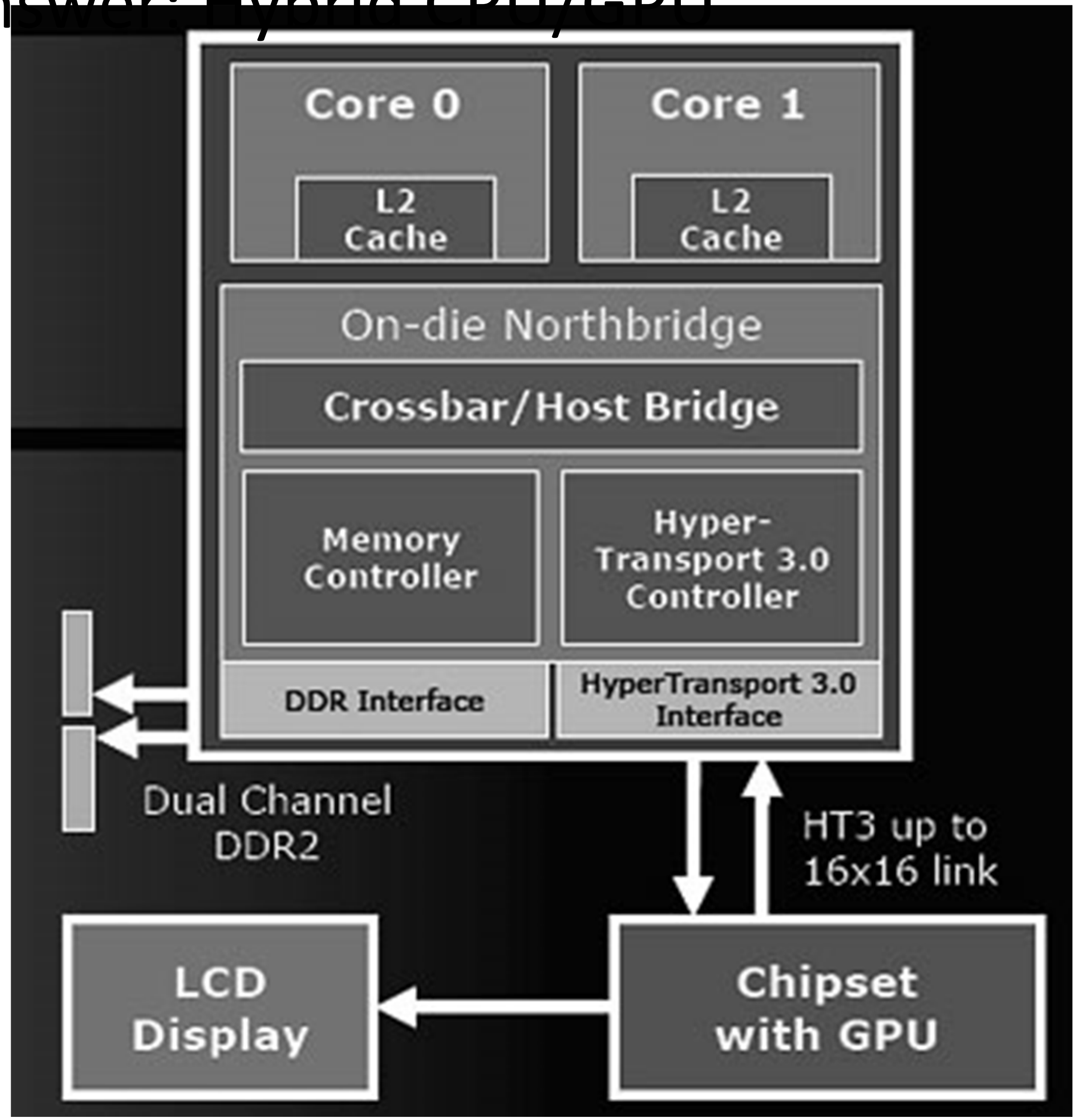
One-pixel polygons (~10M polygons @ 30Hz)



Graph courtesy of Professor John Poulton (from Eric Haines)

AMDs Hybrid CPU/GPU

AMD's Answer: Hybrid CPU/GPU

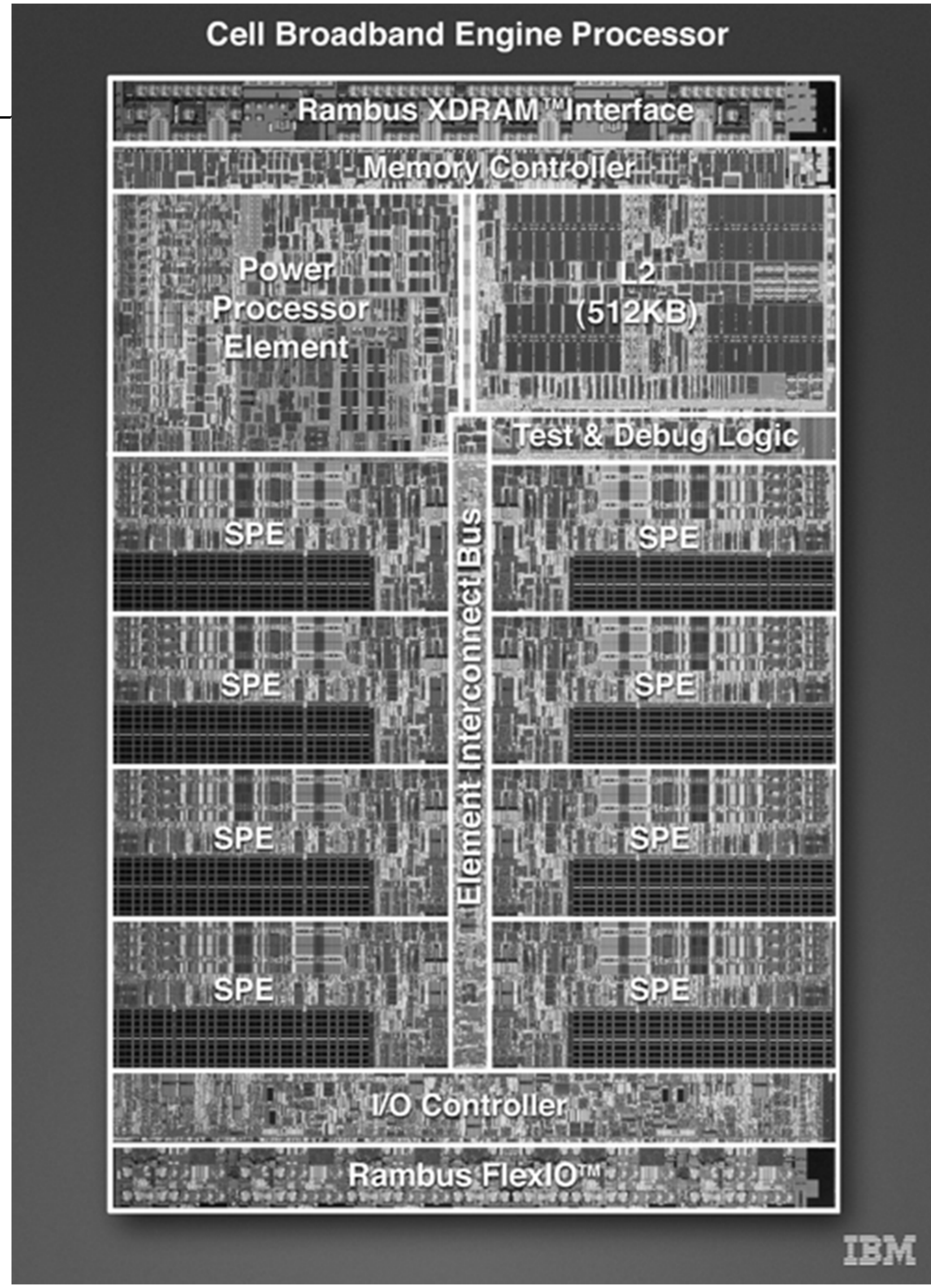


IBM/Sony/Toshiba

Sony Playstation 3

PPE

SPEs (synergestic)



Parallelism

Must exploit parallelism for performance

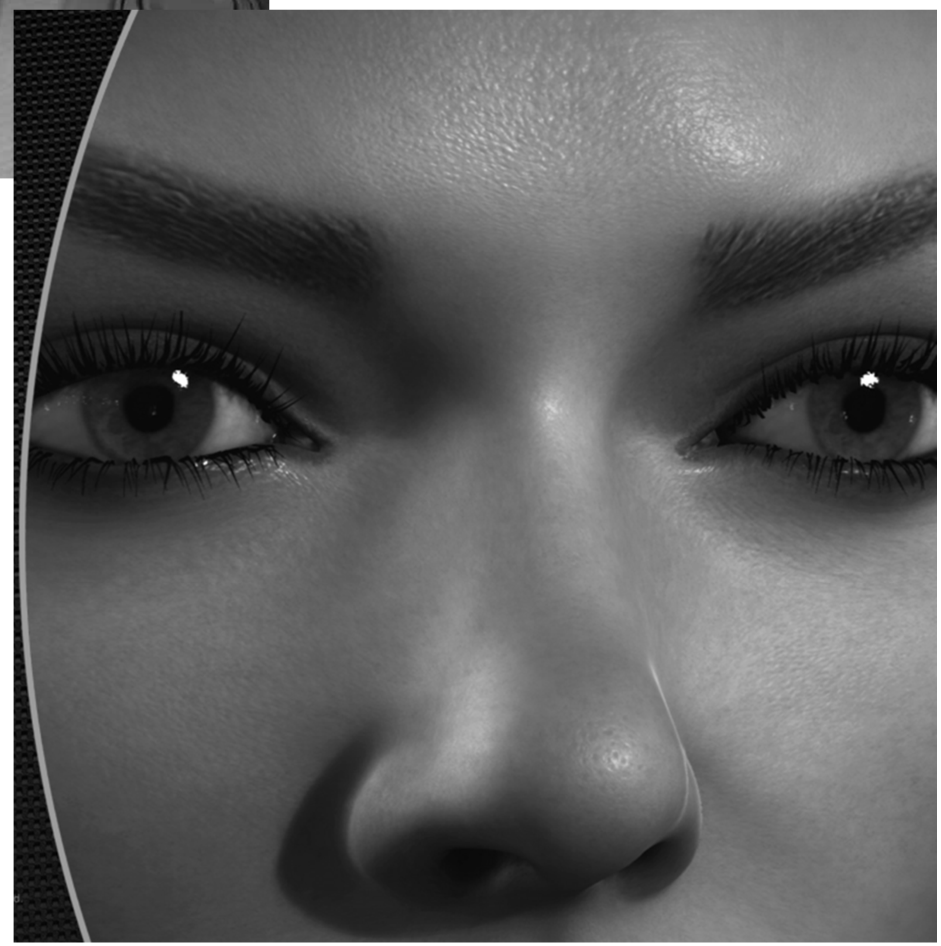
- Lots of parallelism in graphics applications
- Lots of parallelism in scientific computing

SIMD: single instruction, multiple data

- Perform same operation in parallel on many data items
- Data parallelism

MIMD: multiple instruction, multiple data

- Run separate programs in parallel (on different data)
- Task parallelism



NVidia Tesla Architecture

Why are GPUs so fast?

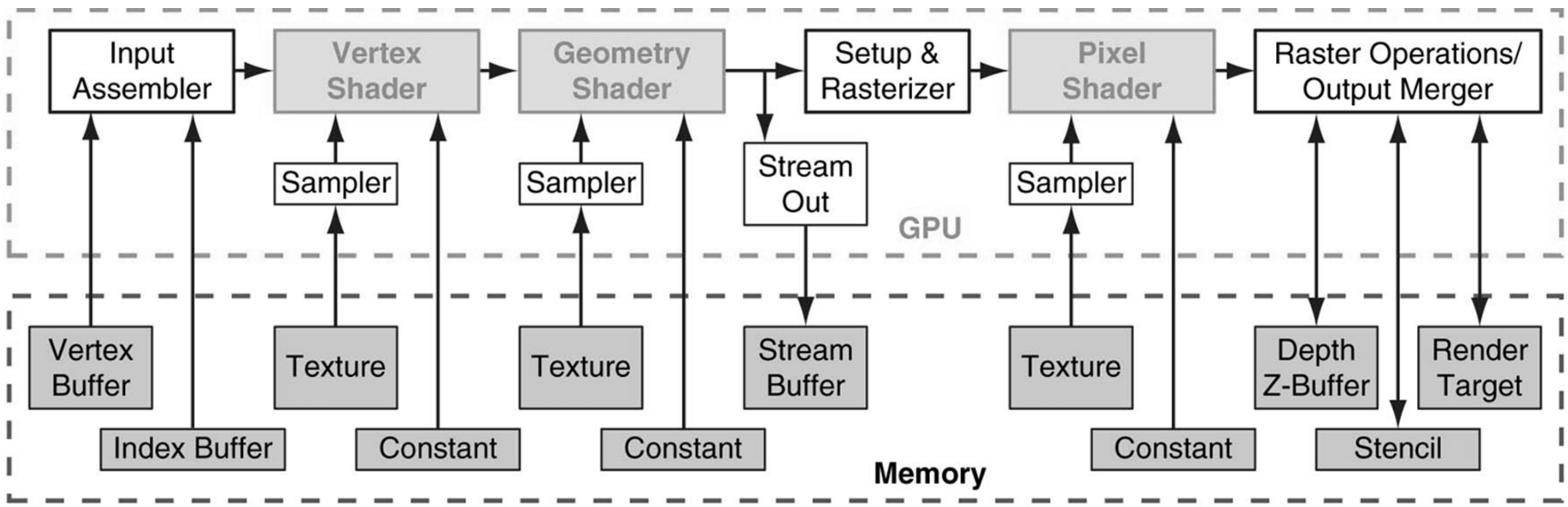


FIGURE A.3.1 Direct3D 10 graphics pipeline. Each logical pipeline stage maps to GPU hardware or to a GPU processor. Programmable shader stages are blue, fixed-function blocks are white, and memory objects are grey. Each stage processes a vertex, geometric primitive, or pixel in a streaming dataflow fashion.

Pipelined and parallel

Very, very parallel: 128 to 1000 cores

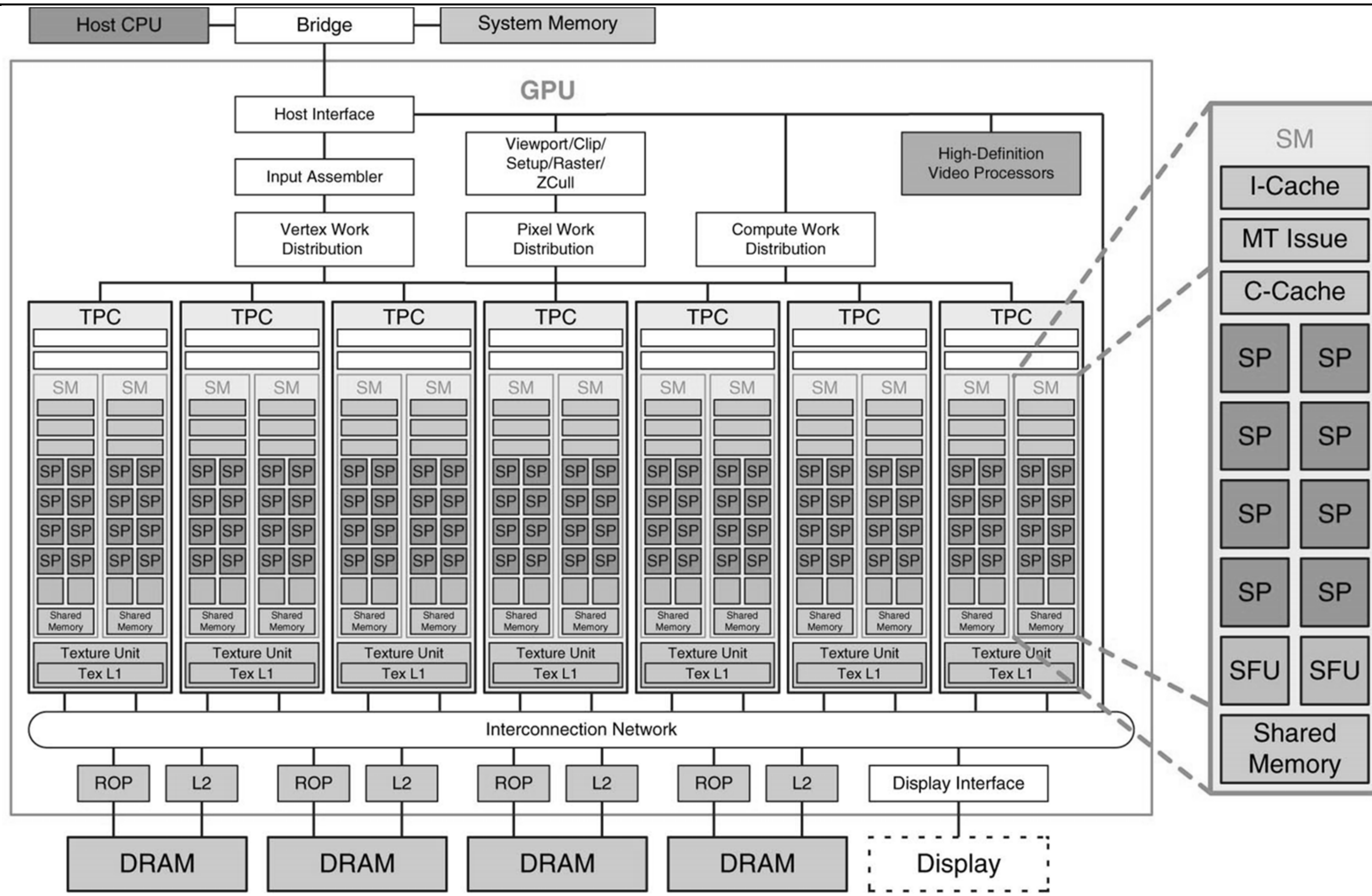


FIGURE A.2.5 Basic unified GPU architecture. Example GPU with 112 streaming processor (SP) cores organized in 14 streaming multiprocessors (SMs); the cores are highly multithreaded. It has the basic Tesla architecture of an NVIDIA GeForce 8800. The processors connect with four 64-bit-wide DRAM partitions via an interconnection network. Each SM has eight SP cores, two special function units (SFUs), instruction and constant caches, a multithreaded instruction unit, and a shared memory.

General computing with GPUs

Can we use these for general computation?

Scientific Computing

- MATLAB codes

Convex hulls

Molecular Dynamics

Etc.

NVIDIA's answer:

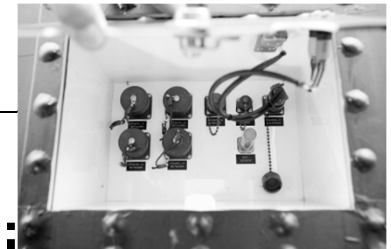
Compute Unified Device Architecture (CUDA)

- MATLAB/Fortran/etc. → "C for CUDA" → GPU Codes

What to do with all these transistors?

Cloud Computing

Cloud Computing



Datacenters are becoming a commodity

Order online and have it delivered

- Datacenter in a box: already set up with commodity hardware & software (Intel, Linux, petabyte of storage)
- Plug data, power & cooling and turn o
  - – typically connected via optical fiber

such datacenters

Cloud Computing = Network of Datacenters

Cloud Computing

Enable datacenters to coordinate over vast distances

- Optimize availability, disaster tolerance, energy
- Without sacrificing performance
- "cloud computing"

Drive underlying technological innovations.

Vision

The promise of the Cloud

- A computer utility; a commodity
- Catalyst for technology economy
- Revolutionizing for health care, financial systems, scientific research, and society

However, cloud platforms today

- Entail significant risk: vendor lock-in vs control
- Entail inefficient processes: energy vs performance
- Entail poor communication: fiber optics vs COTS endpoint

Example: Energy and Performance

Why don't we save more energy in the cloud?

No one deletes data anymore!

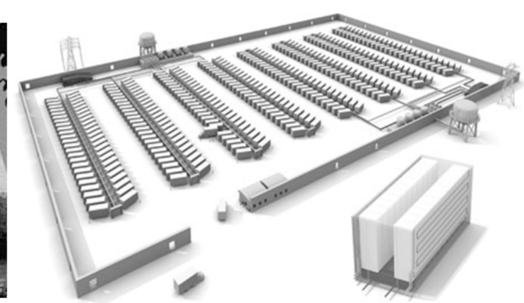
- Huge amounts of seldom-accessed data

Data deluge

- Google (YouTube, Picasa, Gmail, Docs), Facebook, Flickr
- 100 GB per second is faster than hard disk capacity growth!
- Max amount of data accessible at one time << Total data

New scalable approach needed to store this data

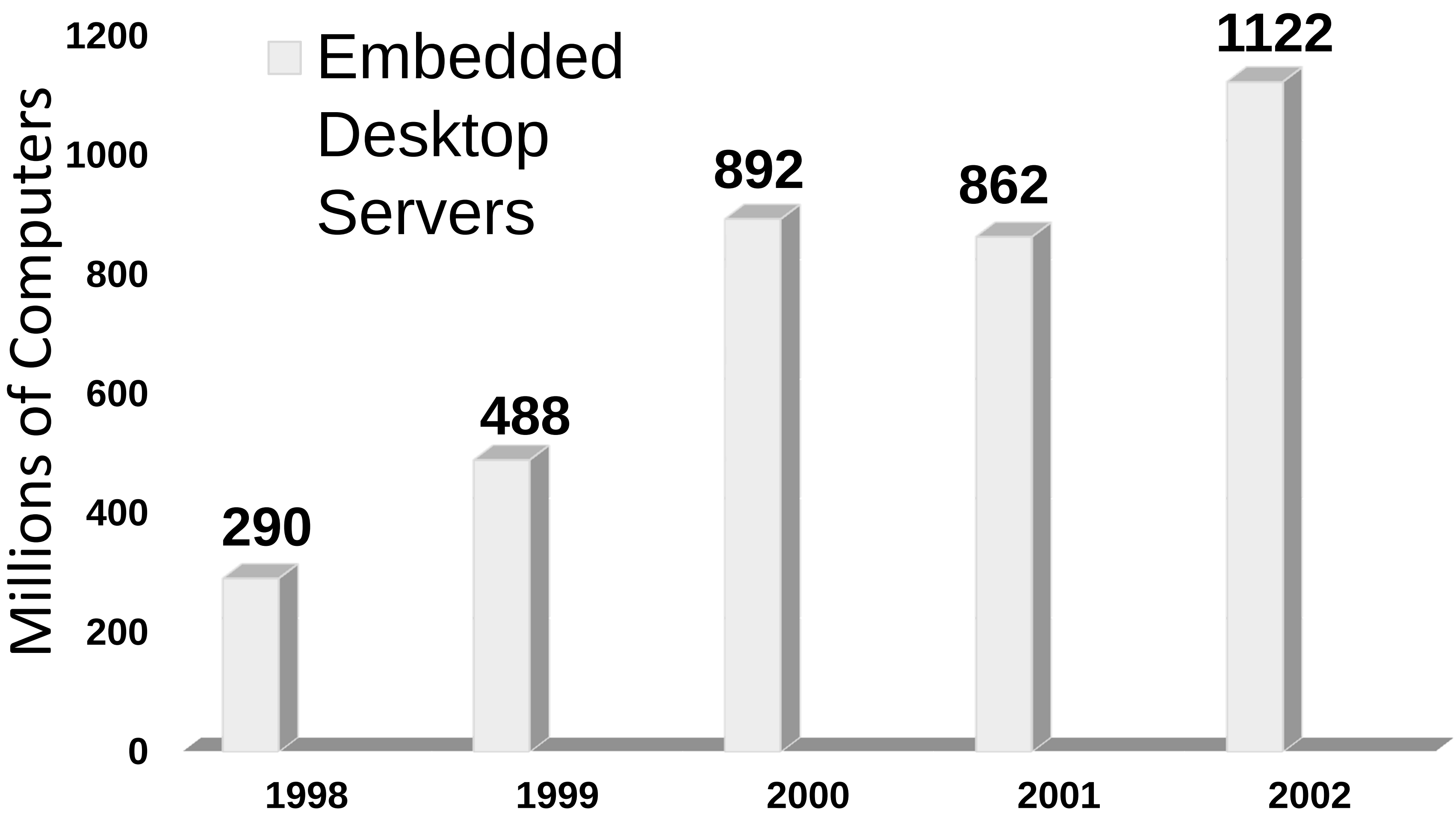
- Energy footprint proportional to number of HDDs is *not* sustainable



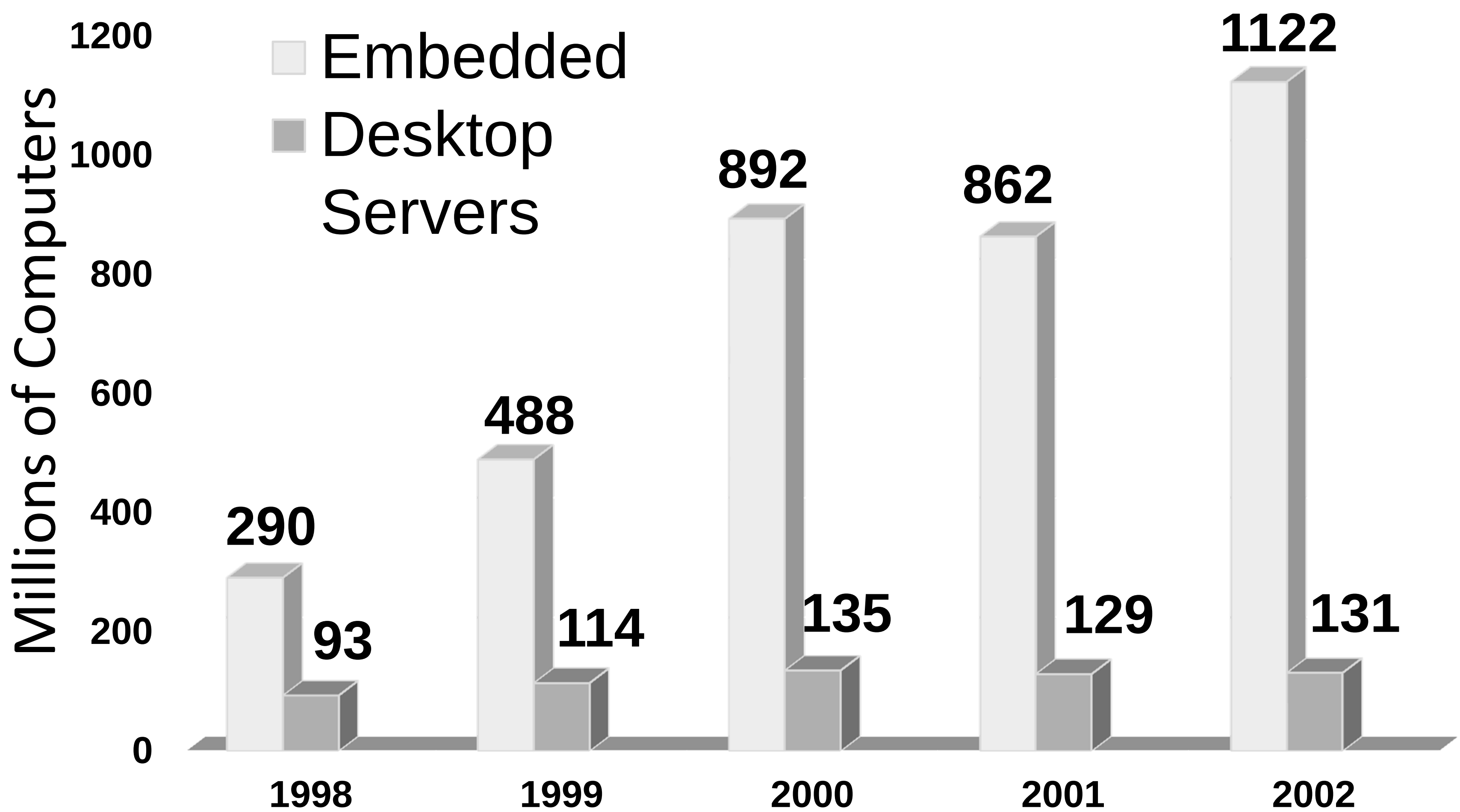
What to do with all these transistors?

Embedded Processors

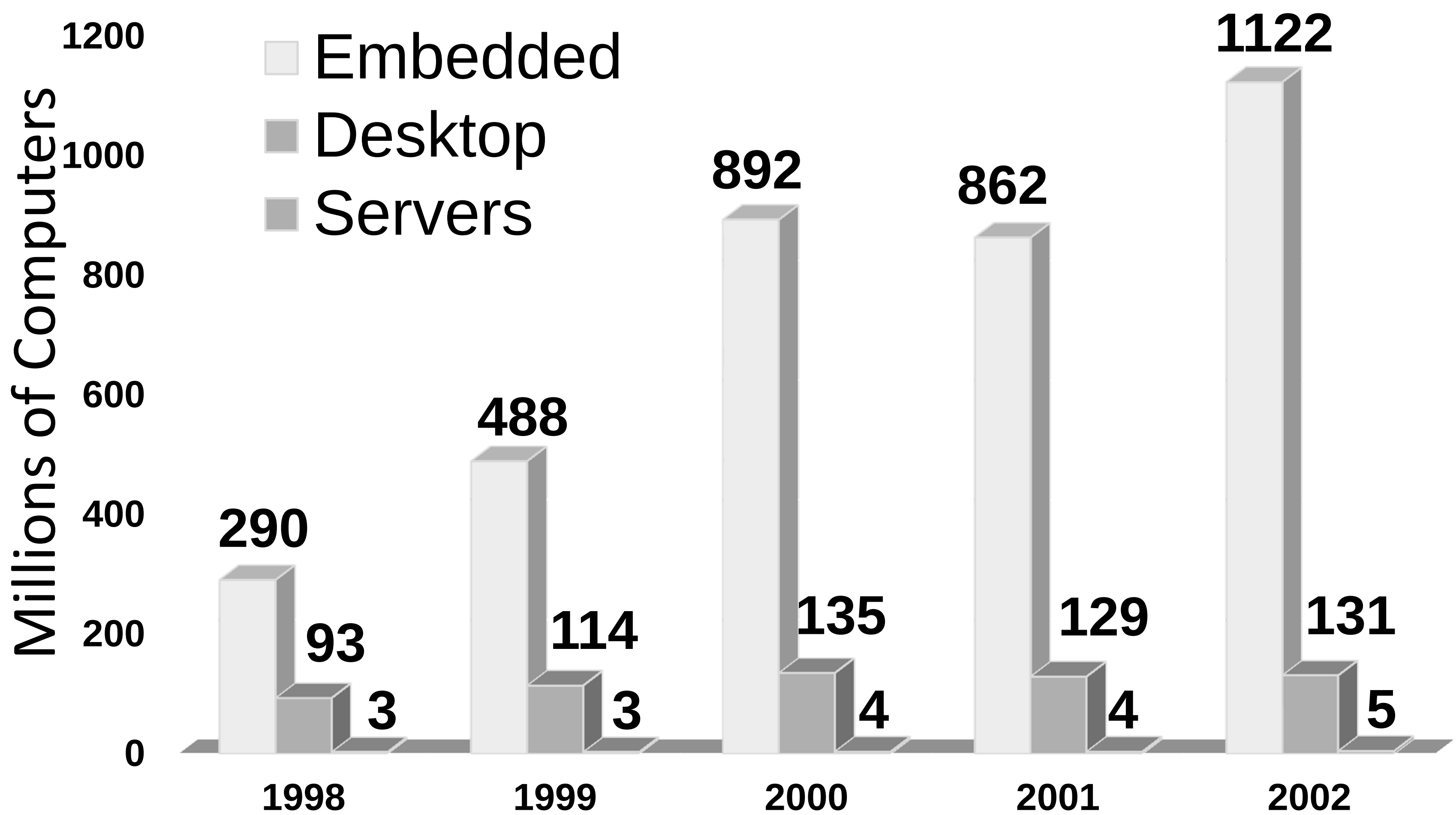
Where is the Market?



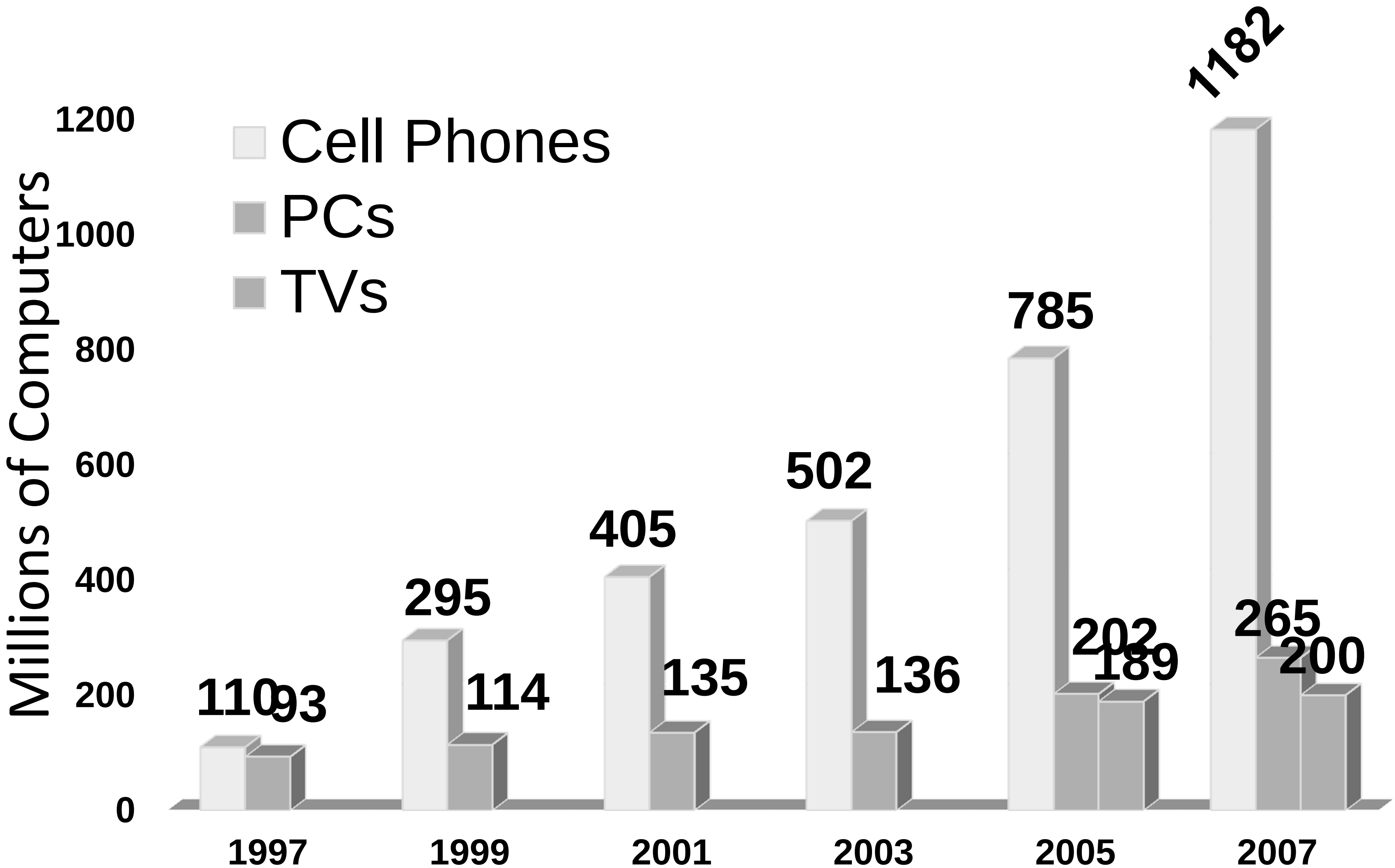
Where is the Market?



Where is the Market?



Where is the Market?

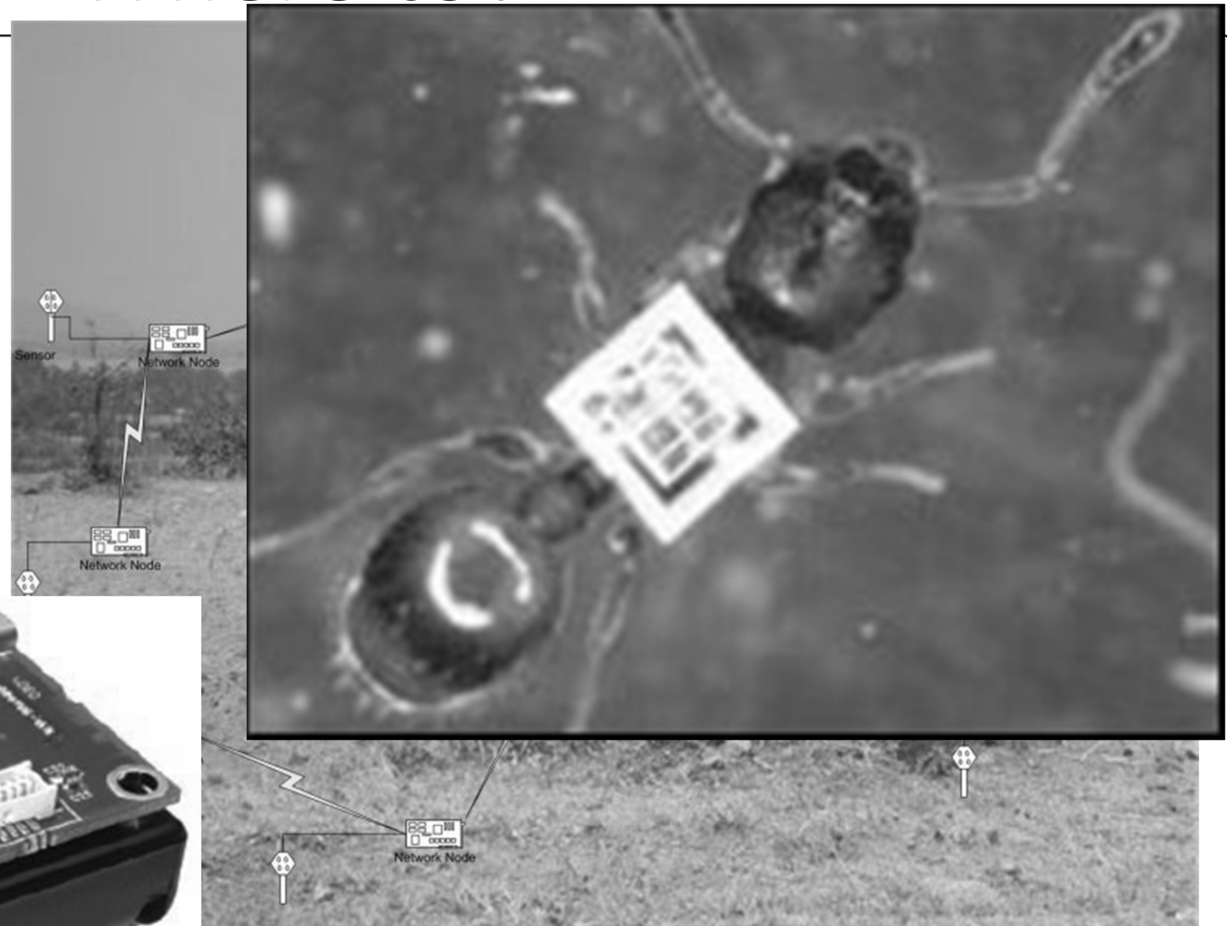


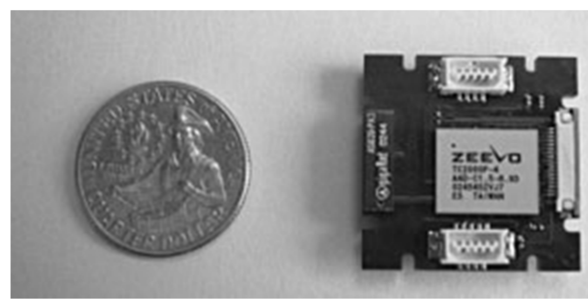

151st St to US-69 S
KIOWA
ARROWHEAD
LINDENWOOD
Arrival
2:16
Menu
Turn In
6.0


at&t
android and me
SAMSUNG

Where to?

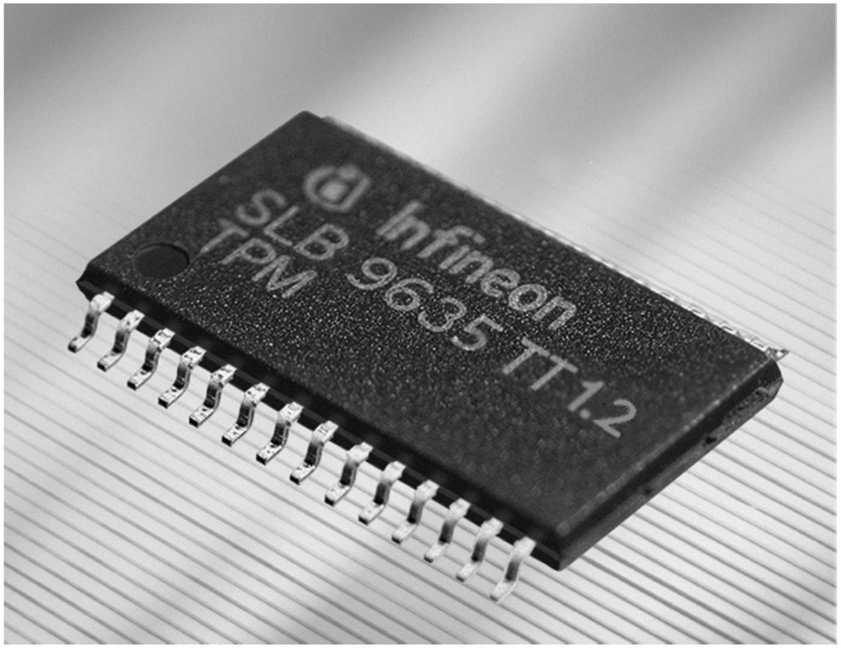
Smart Dust



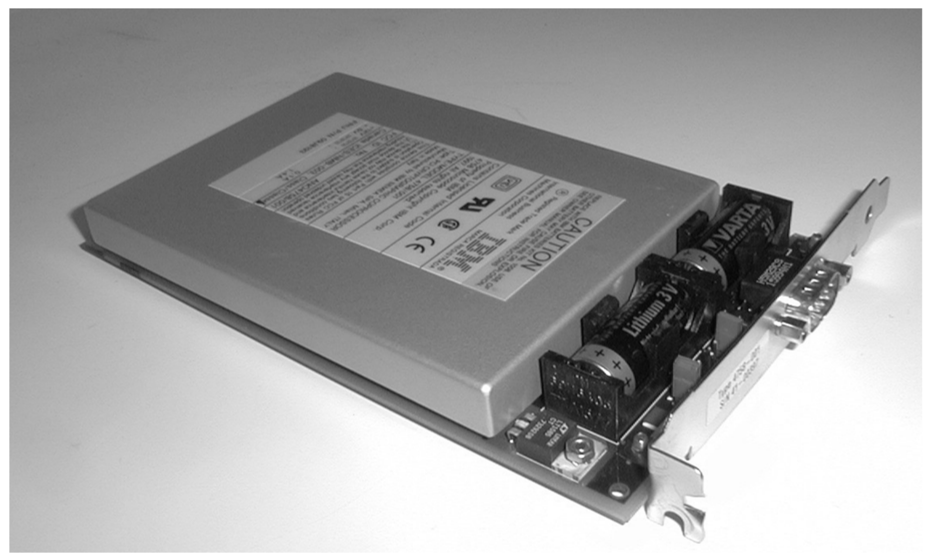


Security?

Cryptography and security…



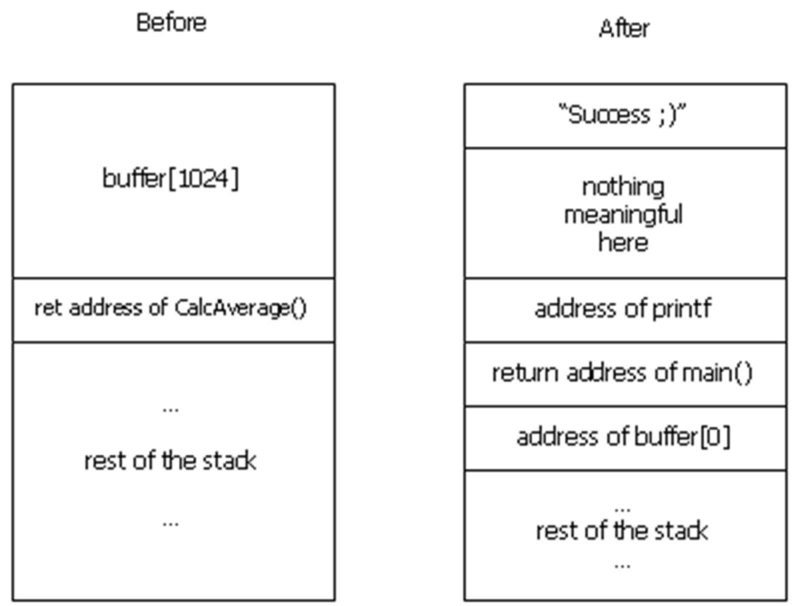
TPM 1.2



IBM 4758
Secure Cryptoprocessor

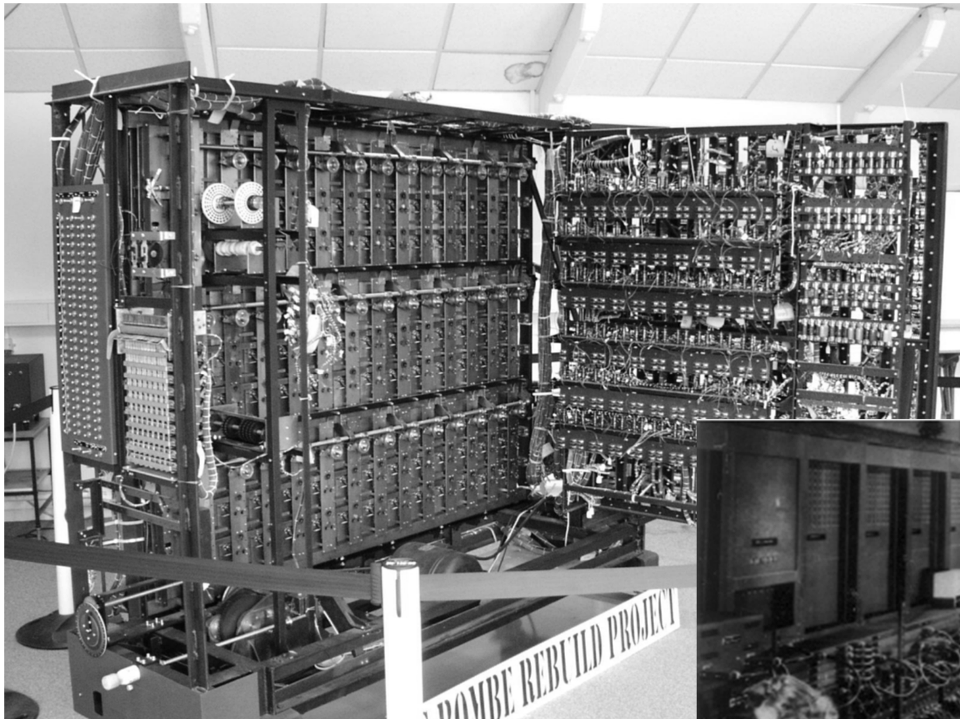
Security?

Stack Smashing…



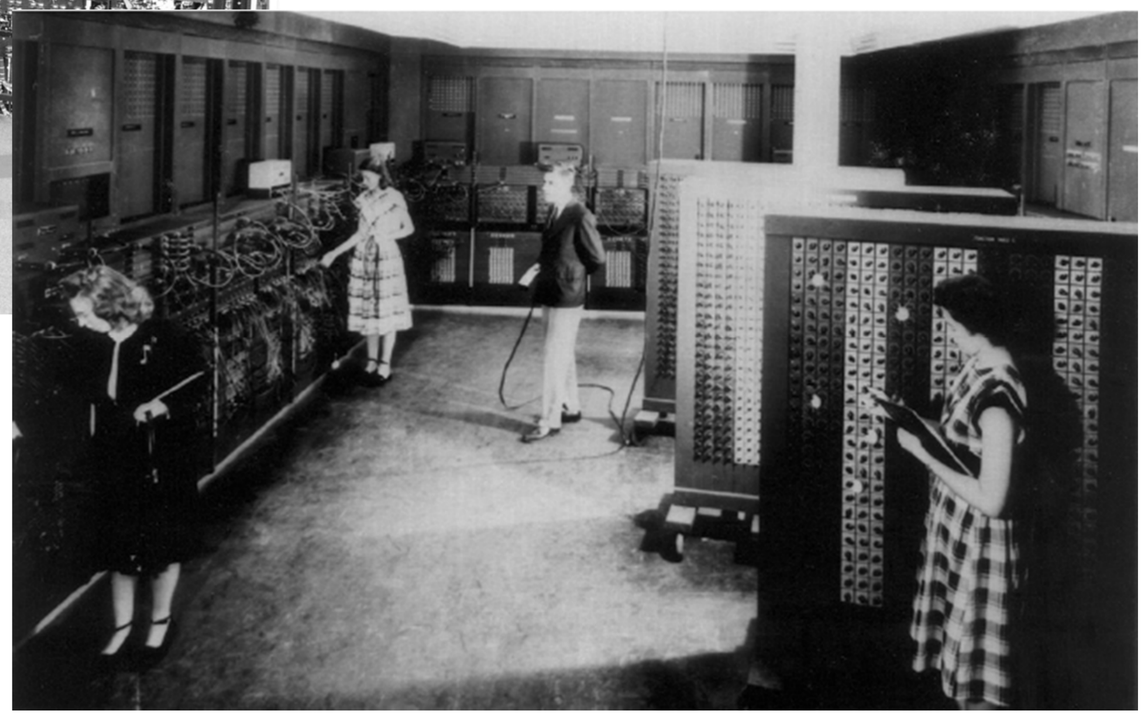
What to do with all these transistors?

You could save the world one day?



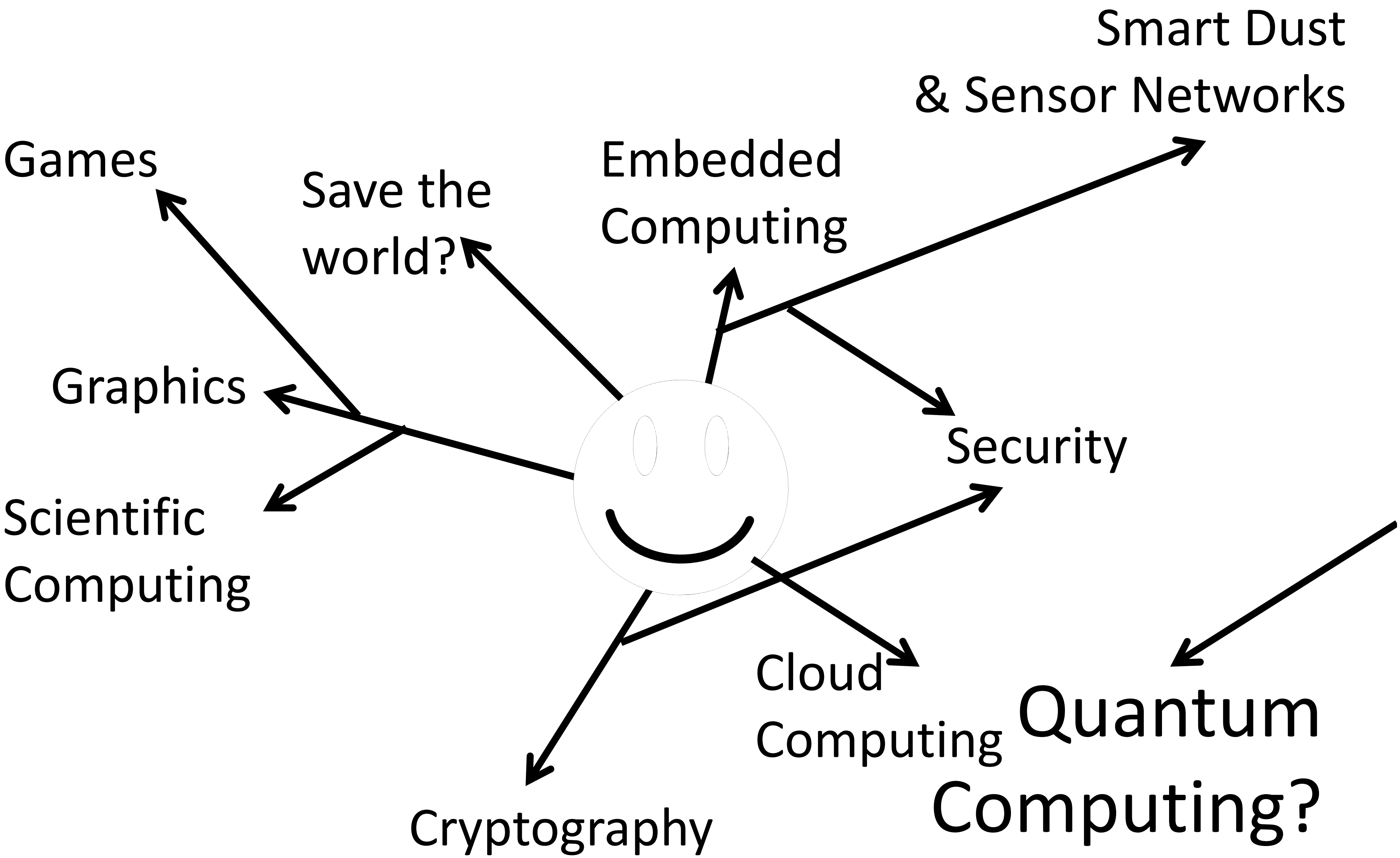
ENIAC - 1946

First general purpose electronic computer. Designed to calculate ballistic trajectories



Alan Turing’s Bombe

Used to crack Germany’s enigma machine


Smart Dust
& Sensor Networks
Games
Save the world?
Embedded Computing
Graphics
Security
Scientific Computing
Cloud Computing
Quantum Computing?
Cryptography

Survey Questions

Are you a better computer scientist and software engineering knowing “the low-level stuff”?

How much of computer architecture do software engineers actually have to deal with?

What are the most important aspects of computer architecture that a software engineer should keep in mind while programming?

Why?

These days, programs run on hardware...

… more than ever before

Google Chrome

→ Operating Systems

→ Multi-Core & Hyper-Threading

→ Datapath Pipelines, Caches, MMUs, I/O & DMA

→ Busses, Logic, & State machines

→ Gates

→ Transistors

→ Silicon

→ Electrons

Why?

Your job as a computer scientist will require knowledge the computer

Research/University

Cornell University
Faculty of Computing and Information Science

Industry

Government

Where to?

CS 3110: Better concurrent programming

CS 4410/4411: The Operating System!

CS 4420/ECE 4750: Computer Architecture

CS 4450: Networking

CS 4620: Graphics

~~CS 4821: Quantum Computing~~

MEng

5412—Cloud Computing, 5414—Distr Computing,

5430—Systems Secuirty,

5300—Arch of Larg scale Info Systems

And many more...

Thank you!

If you want to make an apple pie from scratch, you must first create the universe.

– Carl Sagan