# Introduction

Tom Kelliher, CS 220

Aug. 29, 2011

## 1 Administrivia

### Announcements

### Assignment

Read 1.4.

### Outline

1. Syllabus.
2. Introduction.

### Coming Up

Performance measurement.

## 2 Introduction

What is computer organization and why is it important? The three dimensions involved in optimizing traditional performance:

1. Algorithms.
2. Organization/architecture.
3. Technology.

A new performance criterion: power. Determined by voltage, transistor count, clock rate.

Examples of problems to be solved:

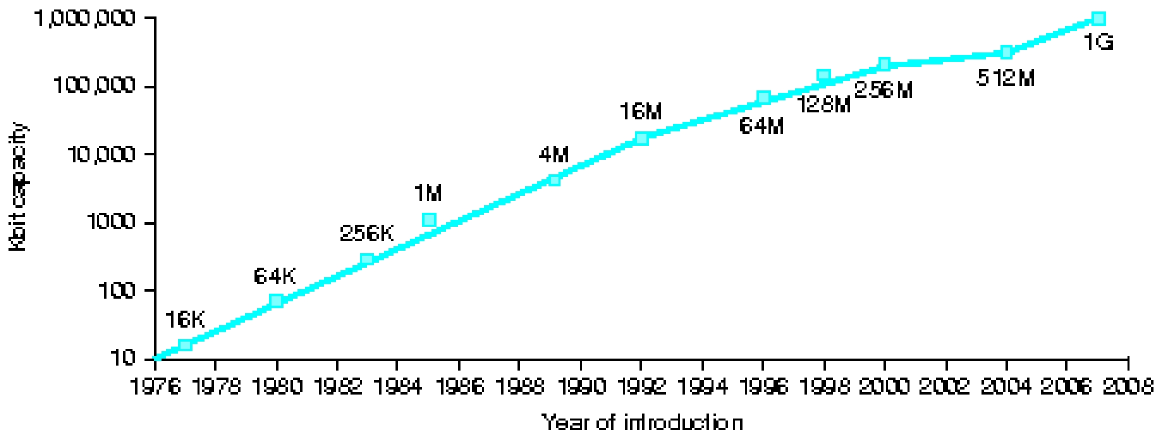
1. Encoding/decoding video/audio.
2. Data mining.
3. Sequence matching.
4. Simulation
5. Finding/organizing/querying data.

Questions to consider:

1. How do we translate human readable programs into machine readable programs? What are the steps?
2. What is architecture — the crux of the software/hardware interface.
3. Performance. What is it? How do we improve it?
4. What has fueled the transition from uniprocessing to multiprocessing (multiple cores, multiple CPU chips)? What are the consequences? How was program parallelism handled earlier?

# 3 The March of Technology

Moore's law: the number of transistors on a chip doubles every two years. What has this given us?




Some more recent figures:

| **Processor** | **Year** | **Transistor Count** |
|---|---|---|
| AMD Athlon 64 | 2003 | 105,900,000 |
| Intel Core 2 Duo | 2006 | 291,000,000 |
| Intel Core 2 Quad | 2006 | 582,000,000 |
| NVIDIA G80 | 2006 | 681,000,000 |
| Intel Dual Core Itanium 2 | 2006 | 1,700,000,000 |
| Six Core Xeon 7400 | 2008 | 1,900,000,000 |
| AMD RV770 | 2008 | 956,000,000 |
| NVIDIA GT200 | 2008 | 1,400,000,000 |
| Eight Core Xeon Nehalem-EX | 2010 | 2,300,000,000 |
| 10 Core Xeon Westmere-EX | 2011 | 2,600,000,000 |
| AMD Cayman | 2010 | 2,640,000,000 |
| NVIDIA GF100 | 2010 | 3,000,000,000 |
| Altera Stratix V | 2011 | 3,800,000,000 |

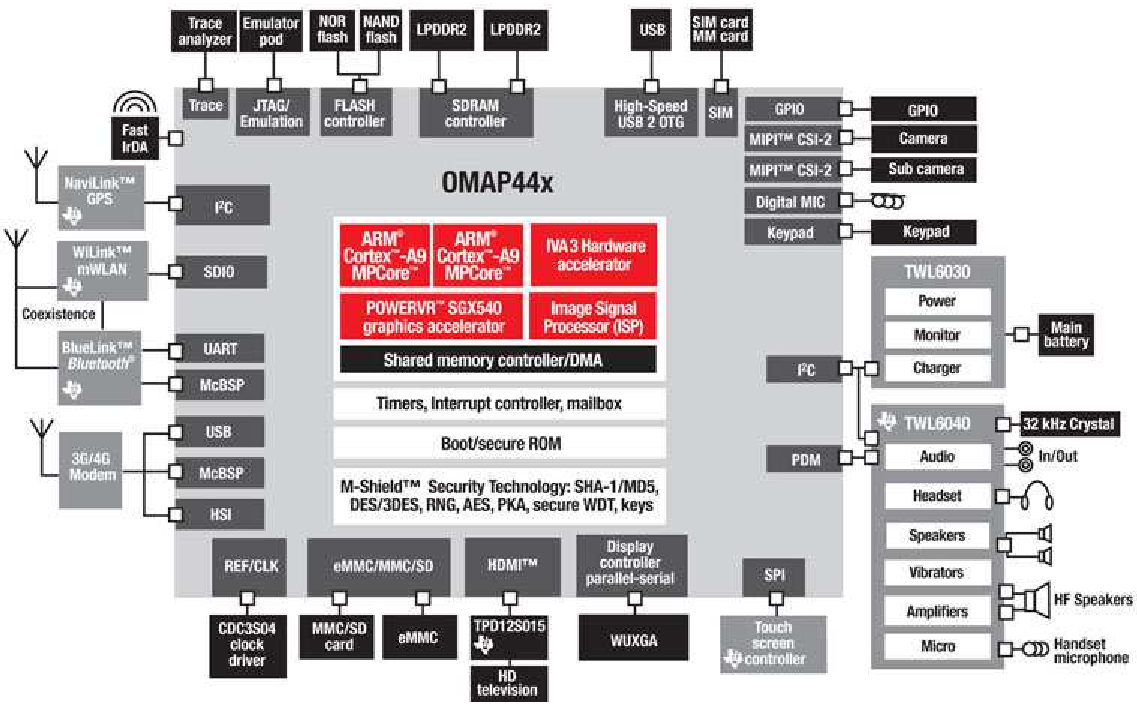
(G80: 128 stream processors — FPUs; RV770 800 SPs; GT200 240 SPs; GF100 512 SPs)

What have architects done with these transistors?

CPUs: lots of transistors tied up in caches.

GPUs: FPU-intensive.

Interestingly, it's not easy to get transistor counts for mobile platforms, such as the OMAP 4. Why?




# 4 Computing Systems

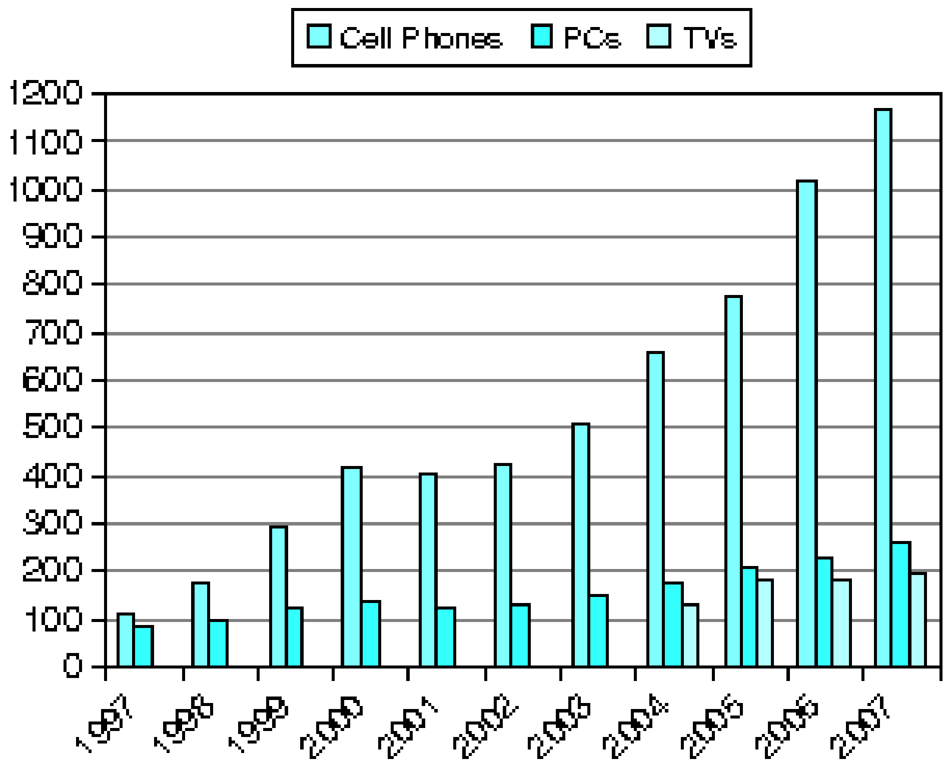
1. Personal systems: desktop and laptops.

2. Servers: Today's "mainframes." File servers have more storage and faster I/O; CPU speed not so critical. Compute servers tend to have more of everything.

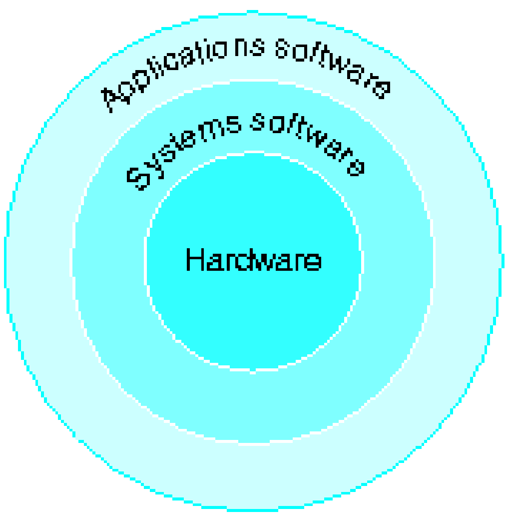
3. Supercomputers: super servers. Large scale simulations — weather, automotive, nuclear.

4. Embedded: the largest category. Where are they?

   Most CPU sales: ARM processors, in cell phone handsets




Layered system design:




1. Hardware.

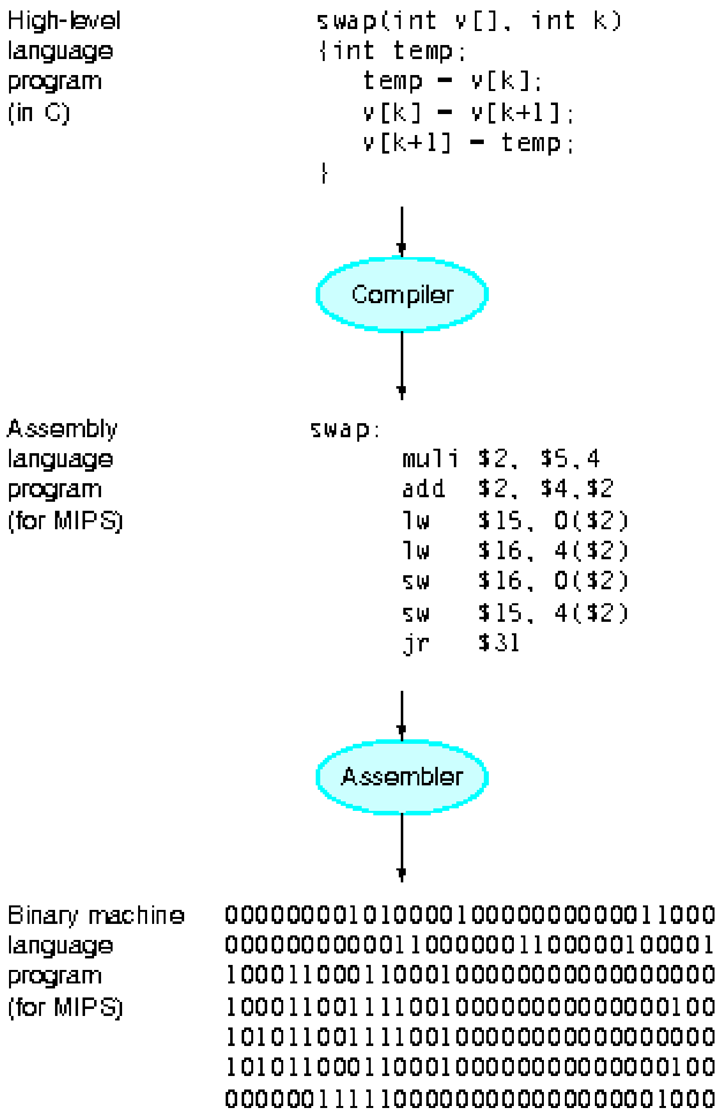
2. Operating system.

3. System software.

4. Application software.

5. User.

Compilation process:




1. HLL and compiler.

2. Assembly and assembler.

   One-to-one correspondence to machine code (usually).

3. Binary machine code.

How does Java fit into this model?

Components of a computer:

1. Input, output.
2. Memory.

   Hierarchy:

   (a) Registers.

   (b) L1 and L2 caches.

   (c) Memory.

   (d) Hard disk.

   (e) Floppy, CD, Zip, flash drive, tape, etc.

   Technologies:

   (a) Flip flops.

   (b) Static, dynamic RAM.

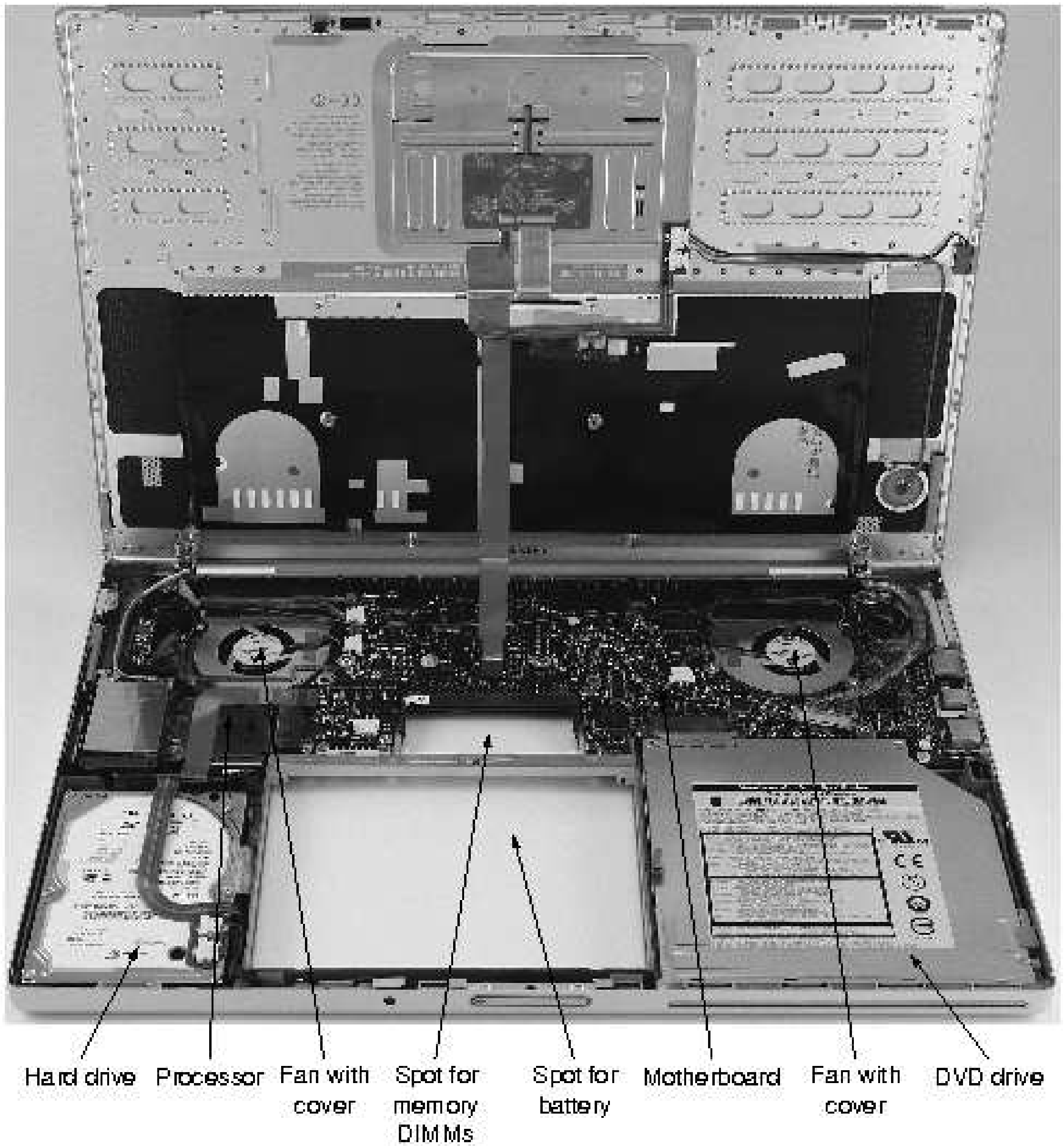
   (c) Flash

   (d) Disk technology.
3. CPU. Control, datapath.

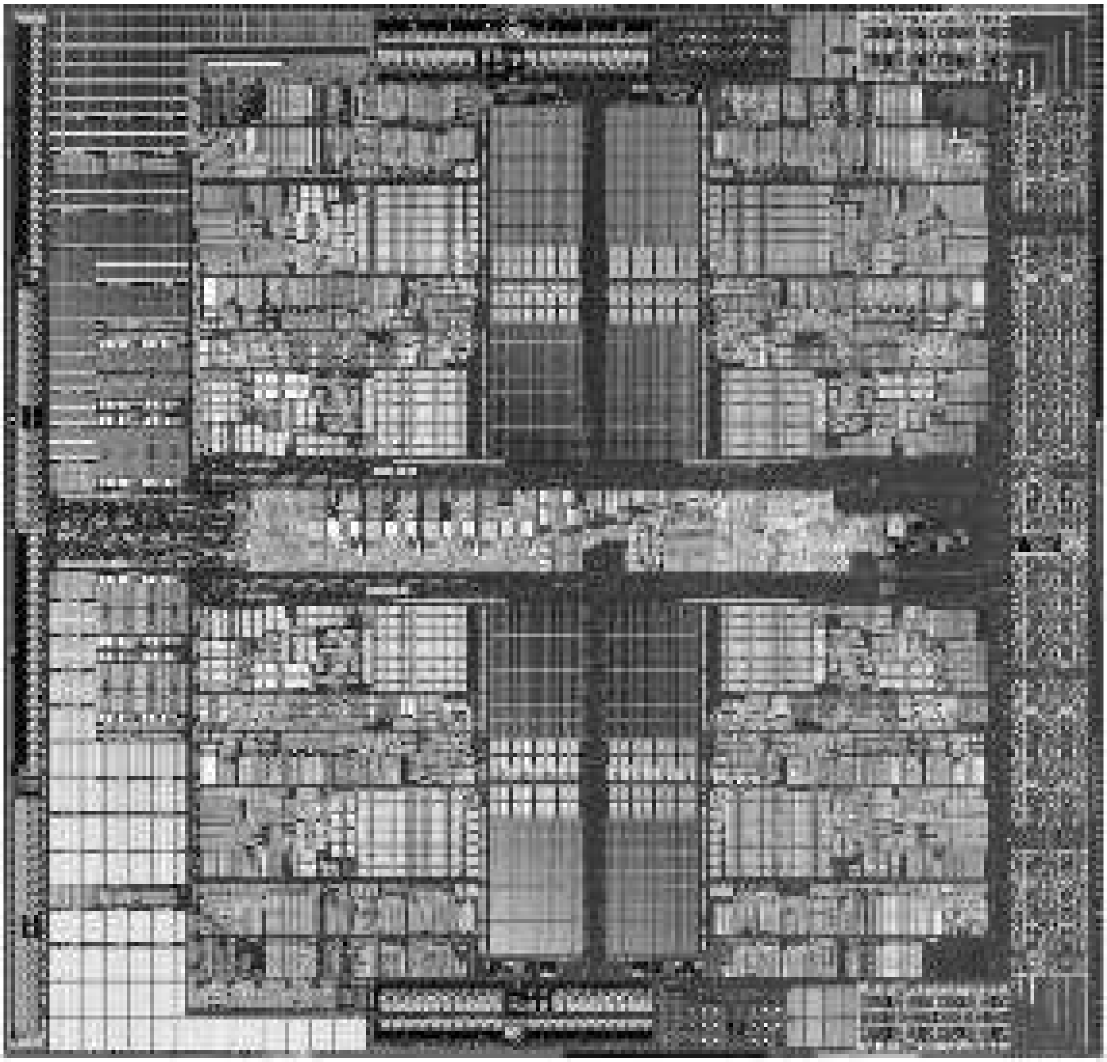
   Memory and I/O can't always keep up.

# 5 Odds and Ends

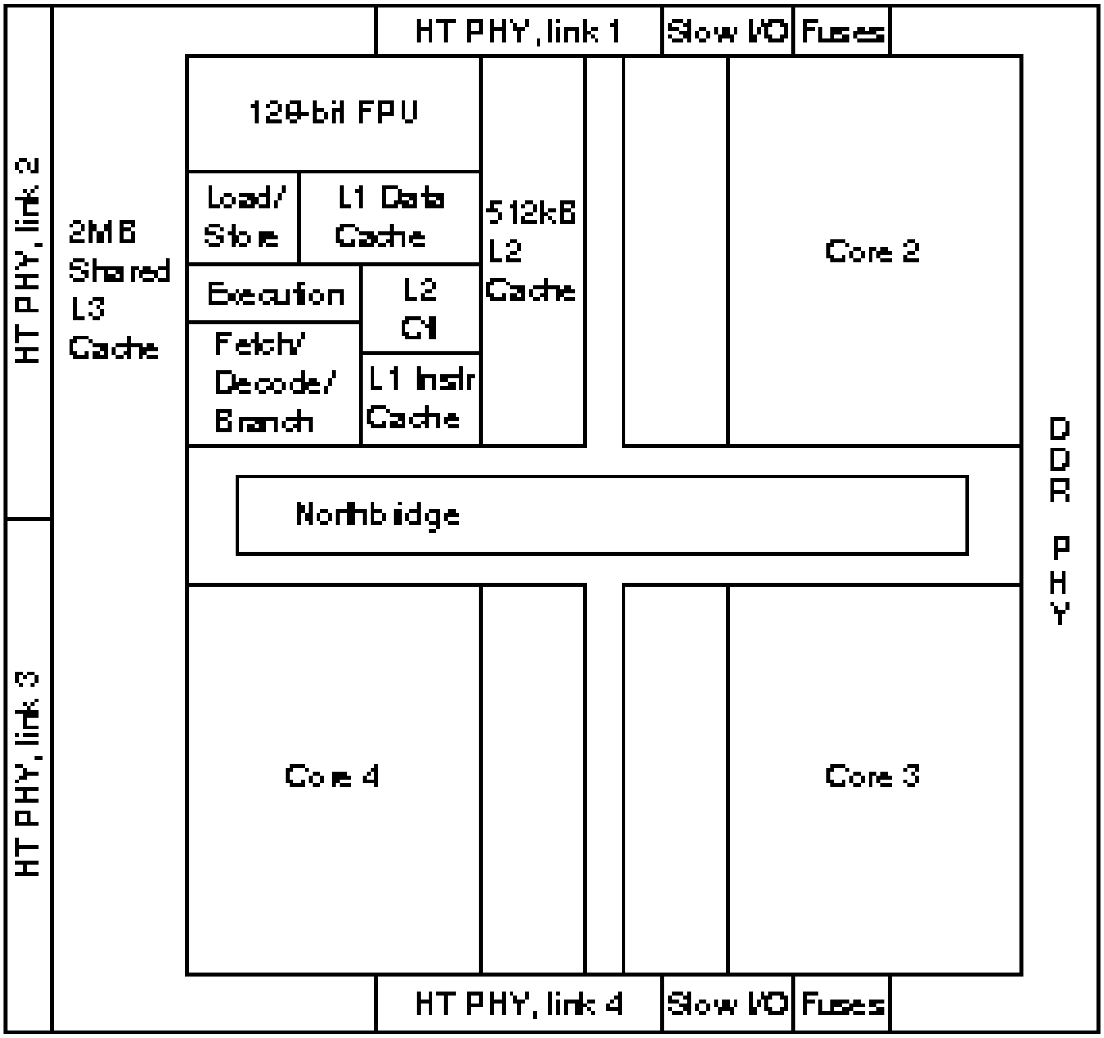
1. An awful lot if packed into a laptop (or an iPhone):



2. Photomicrograph of a 4 core AMD chip:



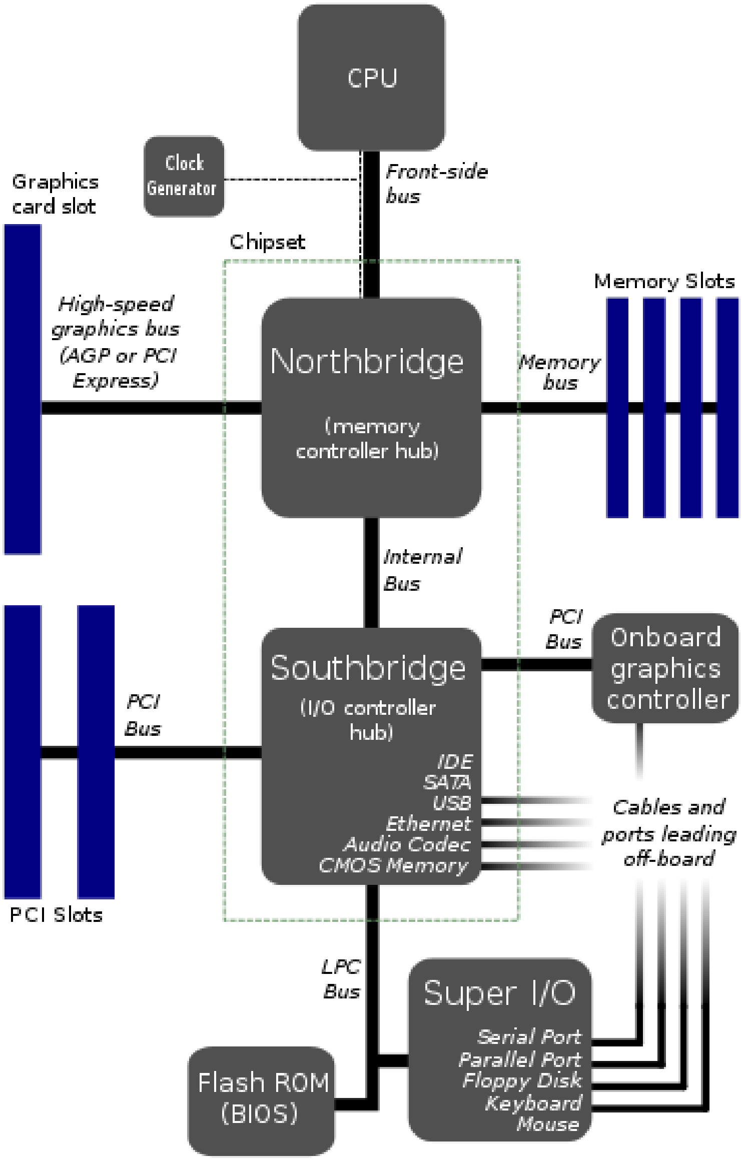
3. Block diagram of same chip:




Buses:

(a) Hypertransport: front-side bus, multiprocessor interconnect, etc.

(b) Northbridge: "fast" I/O devices.




Relative performance of various technologies:

| Year | Technology used in computers | Relative performance/ unit cost |
|---|---|---|
| 1951 | Vacuum tube | 1 |
| 1965 | Transistor | 35 |
| 1975 | Integrated circuit | 900 |
| 1995 | Very large-scale integrated circuit | 2,400, 000 |
| 2005 | Ultra large-scale integrated circuit | 6,200, 000,000 |