Computer Architecture

Lecture 1

Introduction

Why do you want to study Computer Architecture?

- Because….You won't graduate if you don't take this course.
- Because….You want to design the next great instruction set.
  - Instruction set architecture has largely converged, especially in the desktop/server/laptop space.
  - Dictated by powerful market forces (Intel/ARM).
- Because….You want to become a computer architect and design the next great computer systems.
- Because….The design, analysis, implementation concepts that you will learn are vital to all aspects of computer science and engineering – operating systems, computer networks, compiler, programming languages
- Because….The course will equip you with an intellectual toolbox for dealing with a host of systems design challenges.

From Prof. Fernando C. Colon Osorio's lecture notes

Course Goals

- Understand
  - Interfaces
    - Instruction Set Architecture ("The Hardware/Software Interface")
  - Engineering methodology/ Correctness criteria/ Evaluation methods/ Technology trends involved in the following design techniques
    - Pipelining
    - Cache
    - Multiprocessor
      - Cache Coherence
      - Synchronization
      - Interconnection Network

Interface

Atlas | Reverse Dictionary | Rhyming Dictionary

Dictionary | Thesaurus | Unabridged Dictionary

3 entries found for **interface**.
To select an entry, click on it.

interface[1,noun]
interface[2,verb]
graphical user interface

Go

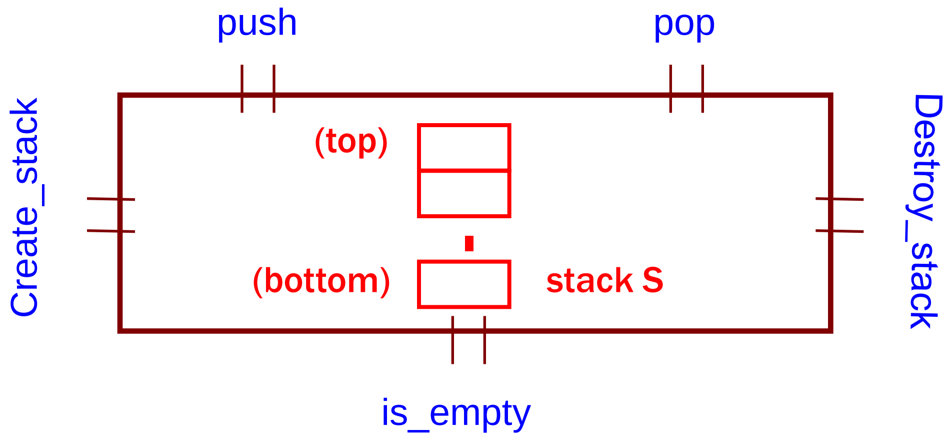
Main Entry: [1]**in·ter·face**
Pronunciation: 'in-t&r-"fAs
Function: *noun*
Date: 1882
1 : a surface forming a common boundary of two bodies, spaces, or phases <an oil-water *interface*> **2 a :** the place at which independent and often unrelated systems meet and act on or communicate with each other <the man-machine *interface*> **b :** the means by which interaction or communication is achieved at an interface
- in·ter·fa·cial /"in-t&r-'fA-sh&l/ *adjective*

Source : http://www.webster.com

Abstract Data Type (ADT) as an Example of Interface

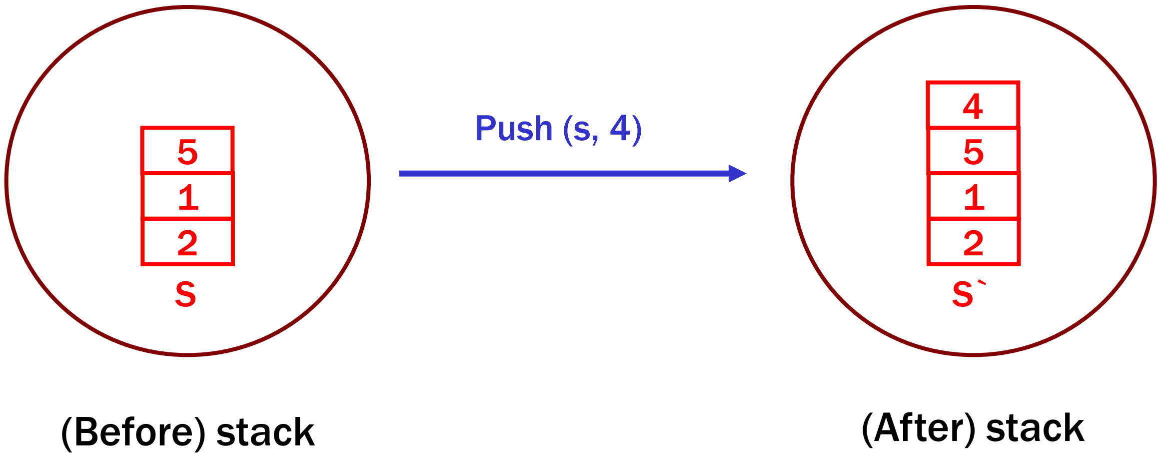
- Abstract data type : A set of data values (state) and associated operations that are precisely specified independent of any particular implementation

- ADT Example : stack



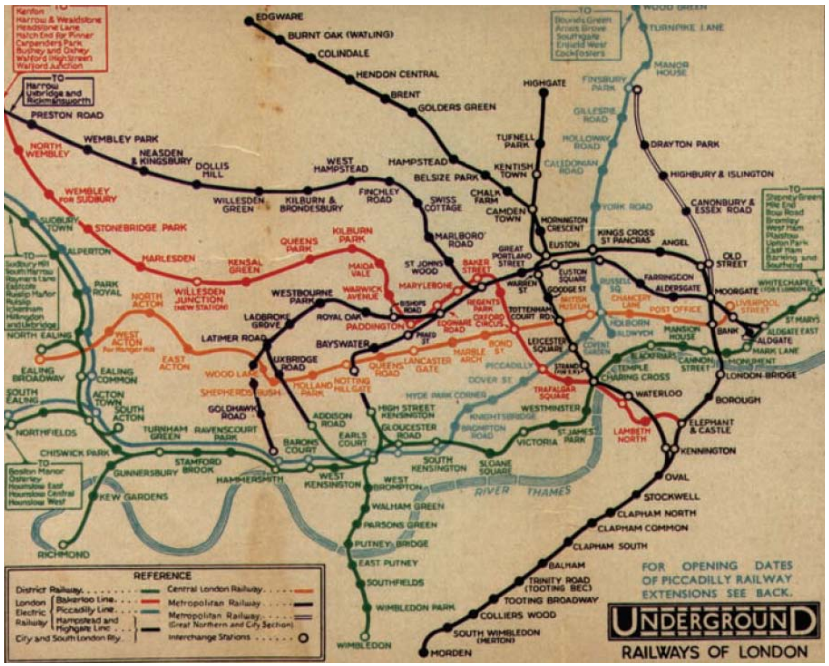
Abstract Data Type (ADT) as an Example of Interface

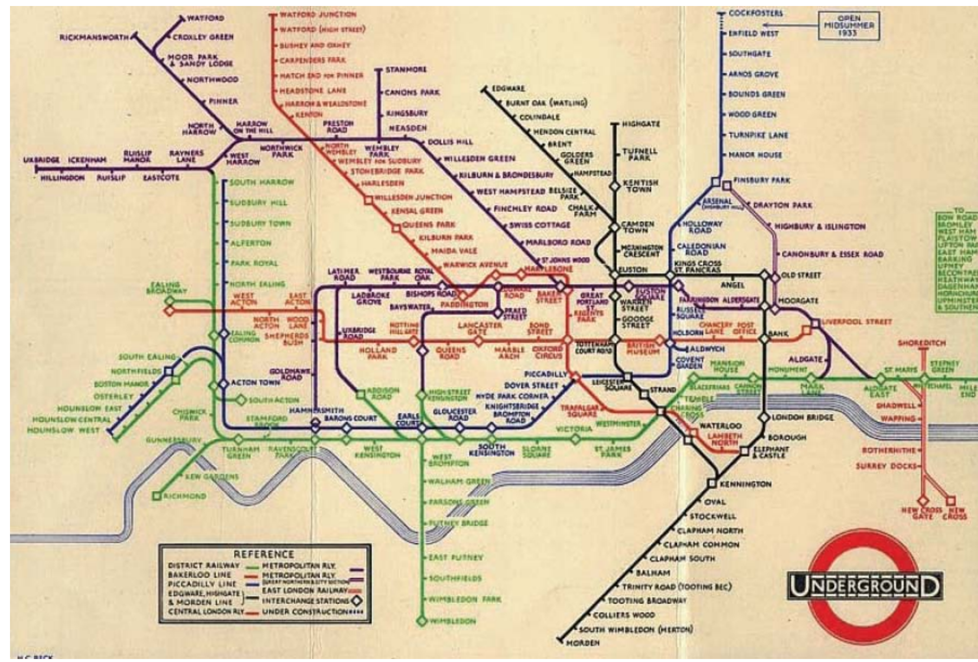
- Operations viewed as state transformation



Abstraction

- (Before)
- (After)





Jeff Kramer, "Is Abstraction the Key to Computing," Communications of ACM, April 2007, Vol. 50, No. 4, pp. 37 - 42.

Abstraction

- (Before)

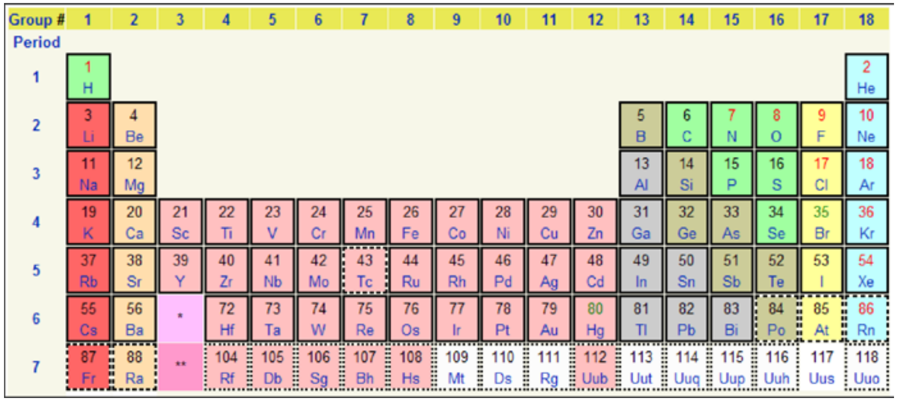
(After)

Abstraction

- (Before)

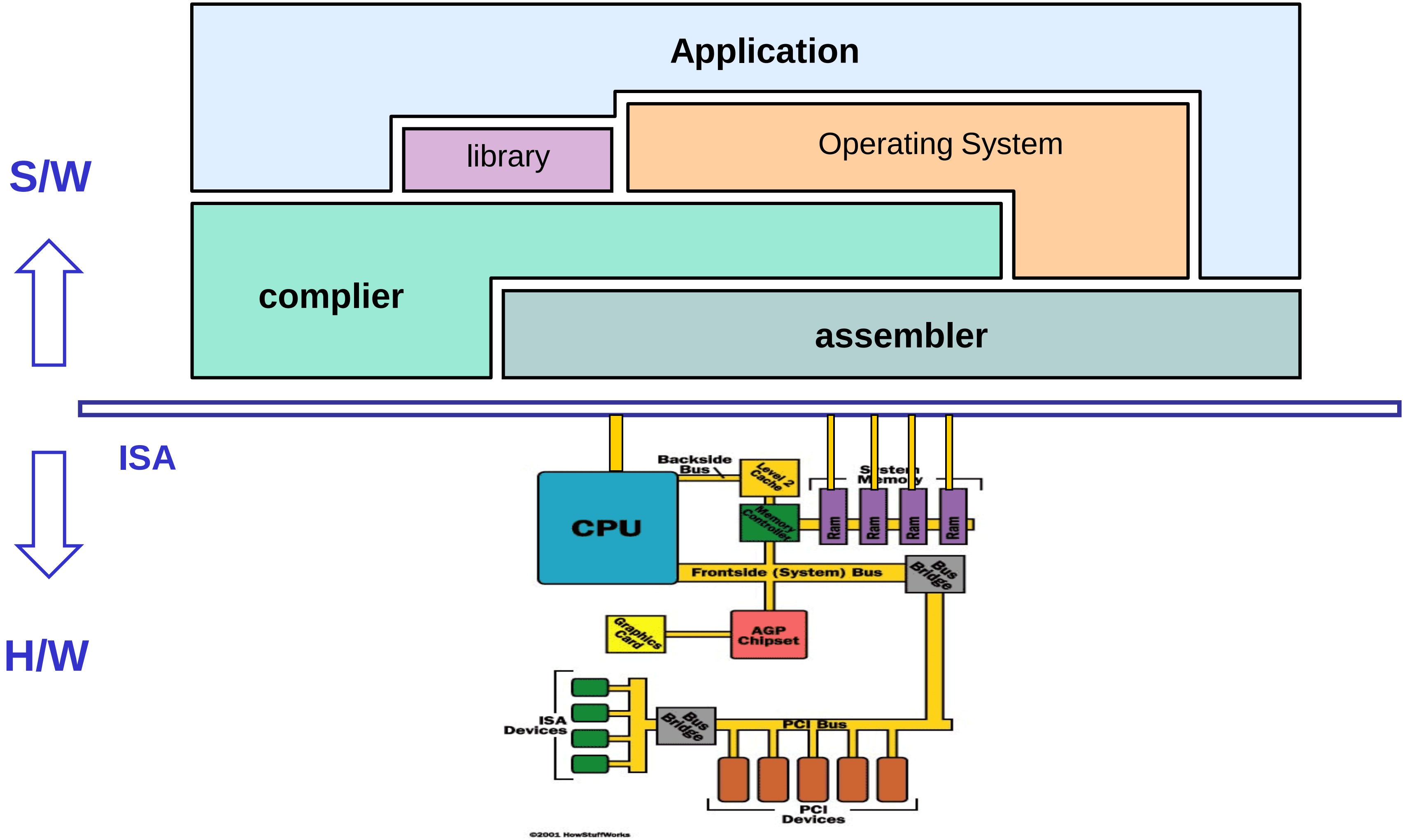
	A	B	C	D	E	F	G	H	I	J	K
1	Element	•P1	•P2	Atomic Num	Atomic Mas	Atomic Radi	Ionic Radius	Ionization E	Electronega	•C1	•C2
2	Ac	140	0	89	227	200	126	51	11	62	56
3	Ag	630	80	47	107	144	129	75	18	124	40
4	Al	750	160	13	27	143	67	60	16	28	25
5	Ar	1050	160	18	39	98	154	158	32	176	51
6	As	870	120	33	75	120	72	98	22	115	33
7	At	990	40	85	210	140	76	95	22	119	22
8	Au	630	40	79	197	144	99	91	25	131	22
9	B	750	200	5	10	85	41	83	20	101	8
10	Ba	80	40	56	137	222	149	51	8	46	56
11	Be	80	200	4	9	112	59	93	15	82	15
12	Bi	870	40	83	209	150	117	73	20	140	27
13	Br	990	120	35	79	114	182	118	30	161	44
14	C	810	200	6	12	77	30	113	25	82	1
15	Ca	80	120	20	40	197	114	60	10	70	51
16	Cd	690	80	48	112	151	109	90	17	113	43
17	Cl	990	160	17	35	100	167	130	32	173	47
18	Co	500	120	27	59	125	83	79	18	120	30
19	Cr	320	120	24	52	128	75	68	17	91	28
20	Cs	20	40	55	132	265	181	39	7	7	56
21	Cu	630	120	29	63	128	87	76	19	118	32
22	F	990	200	9	19	72	119	173	40	39	1
23	Fe	440	120	26	55	126	83	79	18	115	32
24	Fr	20	0	87	223	269	194	40	6	1	56
25	Ga	750	120	31	69	135	76	60	18	89	31
26	Ge	810	120	32	72	122	87	79	20	118	33
27	H	20	240	1	1	32	0	136	22	40	1
28	He	1050	240	2	4	31	93	246	32	1	1
29	Hf	200	40	72	178	159	85	70	12	95	44
30	Hg	690	40	80	200	151	116	103	20	147	27
31	I	990	80	53	126	133	206	105	27	153	44
32	In	750	80	49	114	167	94	58	17	93	42
33	Ir	500	40	77	192	136	82	90	22	116	25
34	K	20	120	19	39	227	152	43	8	37	56
35	Kr	1050	120	36	83	112	169	140	30	163	47

(After)



Jinwook Seo, "Information Visualization Design for Map Use on Future Mobile Devices (Presentation at Samsung Electronics, Dec. 8, 2008)

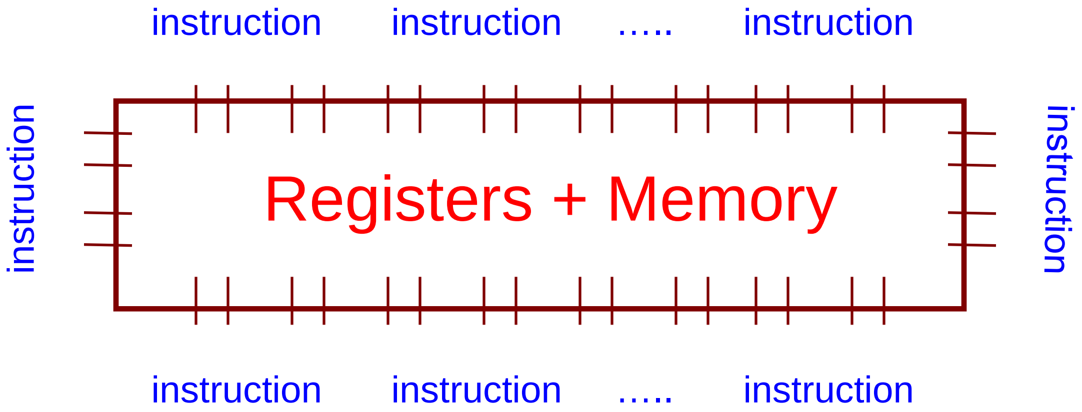
Instruction Set Architecture (ISA)



Instruction Set Architecture as an ADT

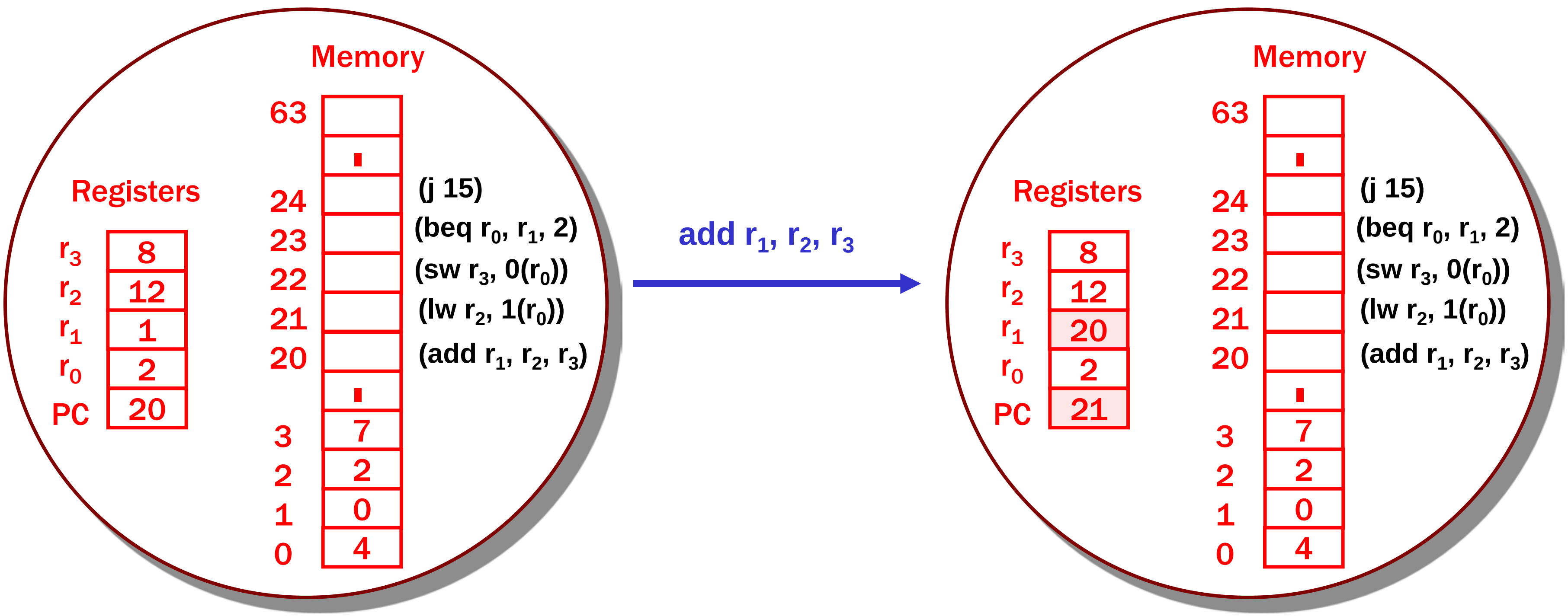
- "…the attributes of a [computing] system as seen by the programmer, i.e. the conceptual structure (state) and functional behavior (operations), as distinct from the organization of the data flow and controls, the logical design, and the physical implementation."

  - Amdahl, Blaauw, and Brooks, 1964



Instruction Set Architecture as an ADT

Assumptions
- 8 bit ISA
- # of registers = 4 + PC (Program Counter)
- Memory size = 64B

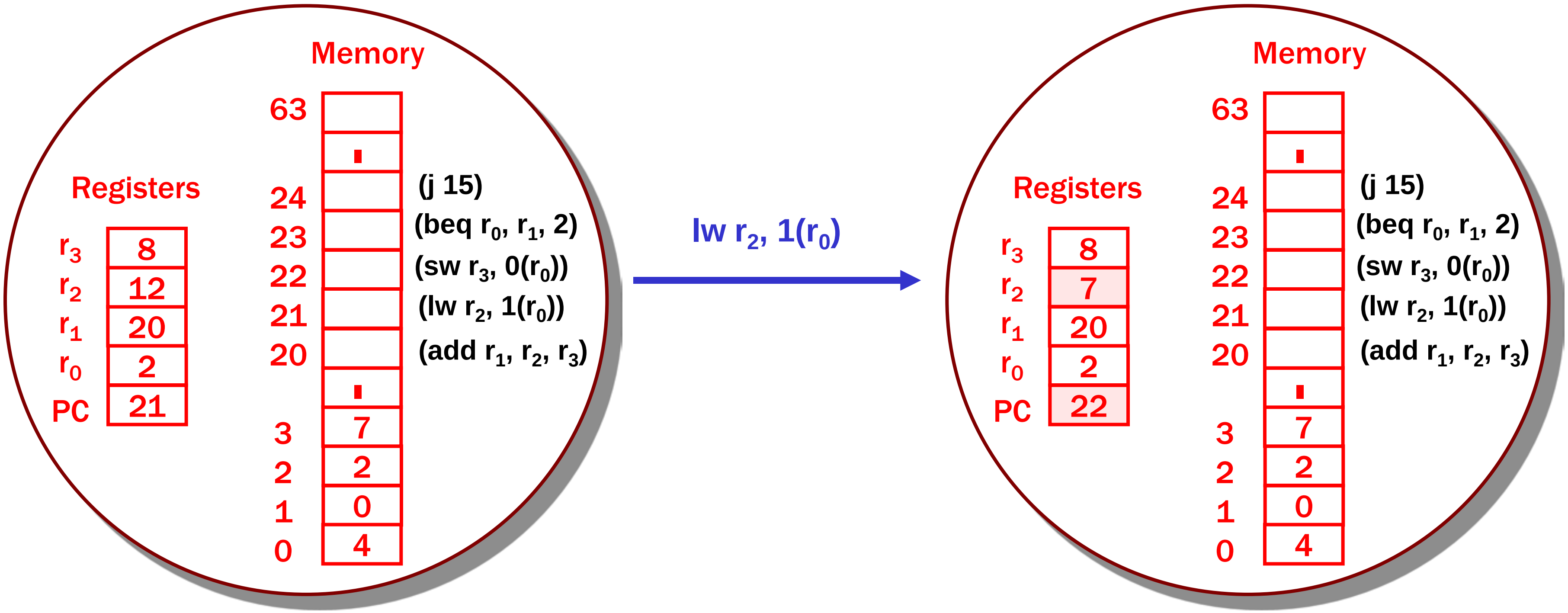


Before Register and Memory **After Register and Memory**

Instruction Set Architecture as an ADT

Assumptions

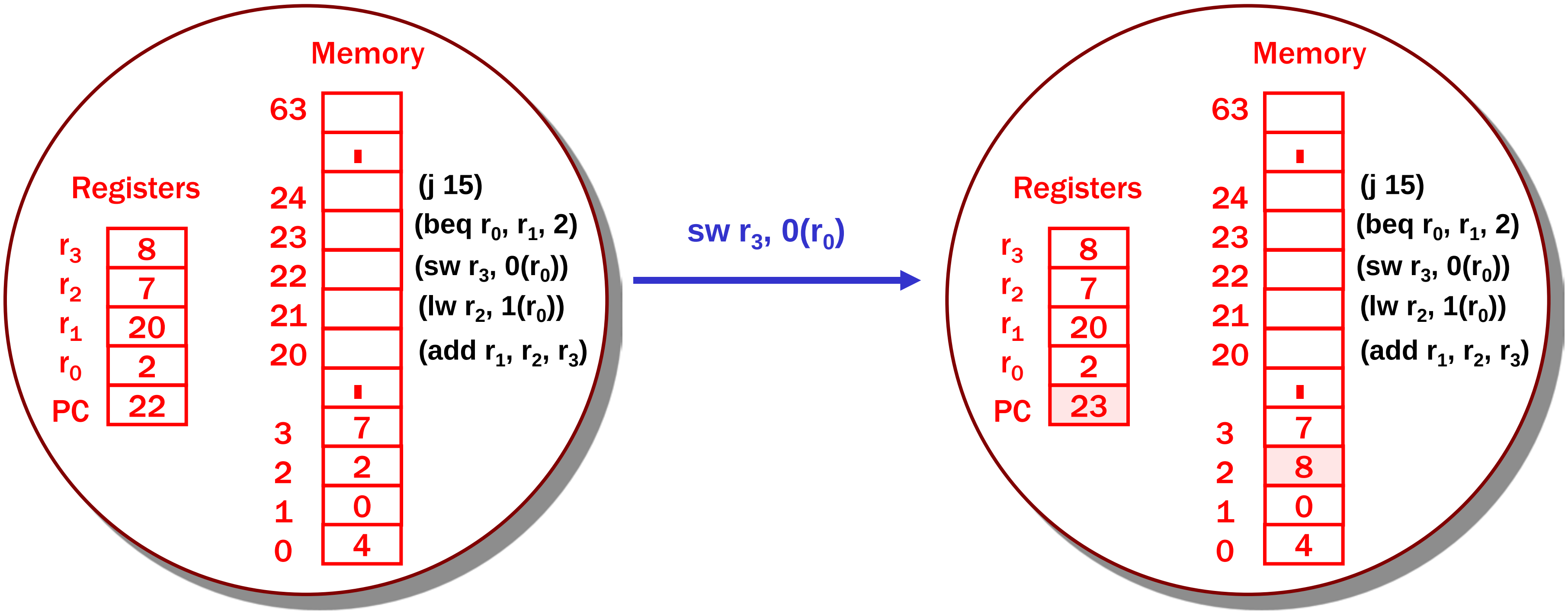
- 8 bit ISA
- # of registers = 4 + PC (Program Counter)
- Memory size = 64B



Before Register and Memory **After Register and Memory**

Instruction Set Architecture as an ADT

Assumptions
- 8 bit ISA
- # of registers = 4 + PC (Program Counter)
- Memory size = 64B

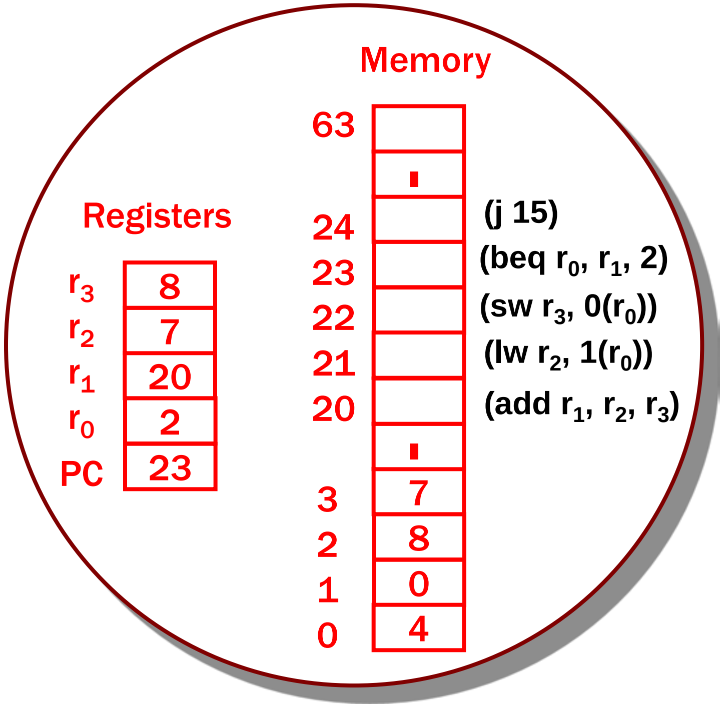


Before Register and Memory **After Register and Memory**

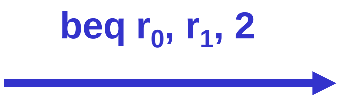
Instruction Set Architecture as an ADT

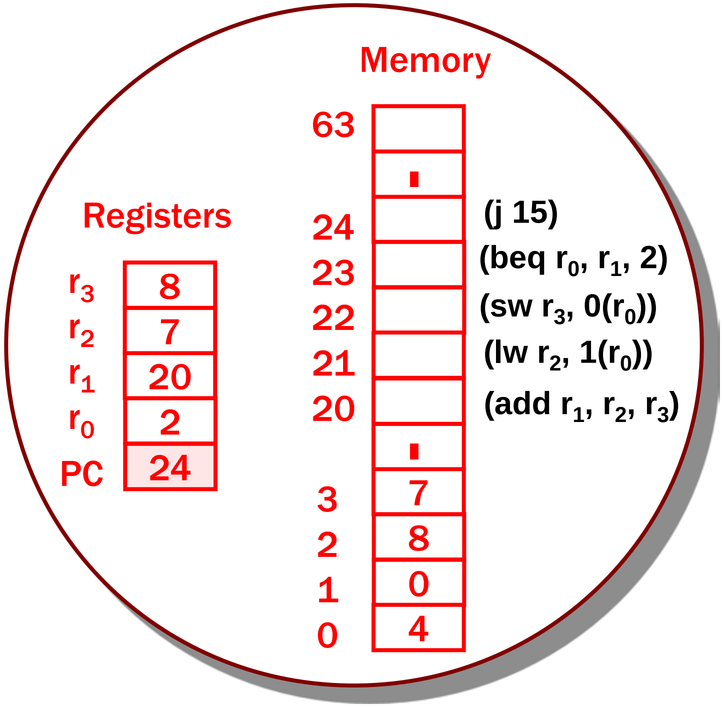
Assumptions

- 8 bit ISA
- # of registers = 4 + PC (Program Counter)
- Memory size = 64B



Before Register and Memory

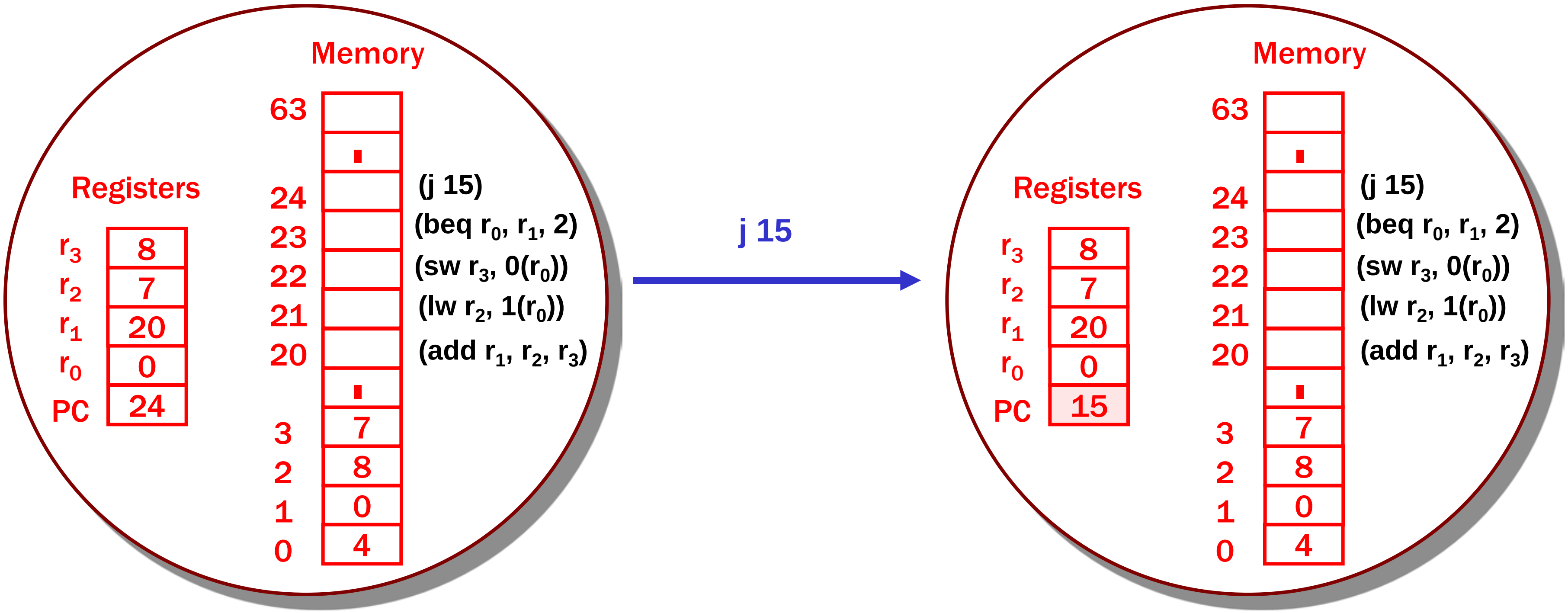




After Register and Memory

Instruction Set Architecture as an ADT

Assumptions
- 8 bit ISA
- # of registers = 4 + PC (Program Counter)
- Memory size = 64B

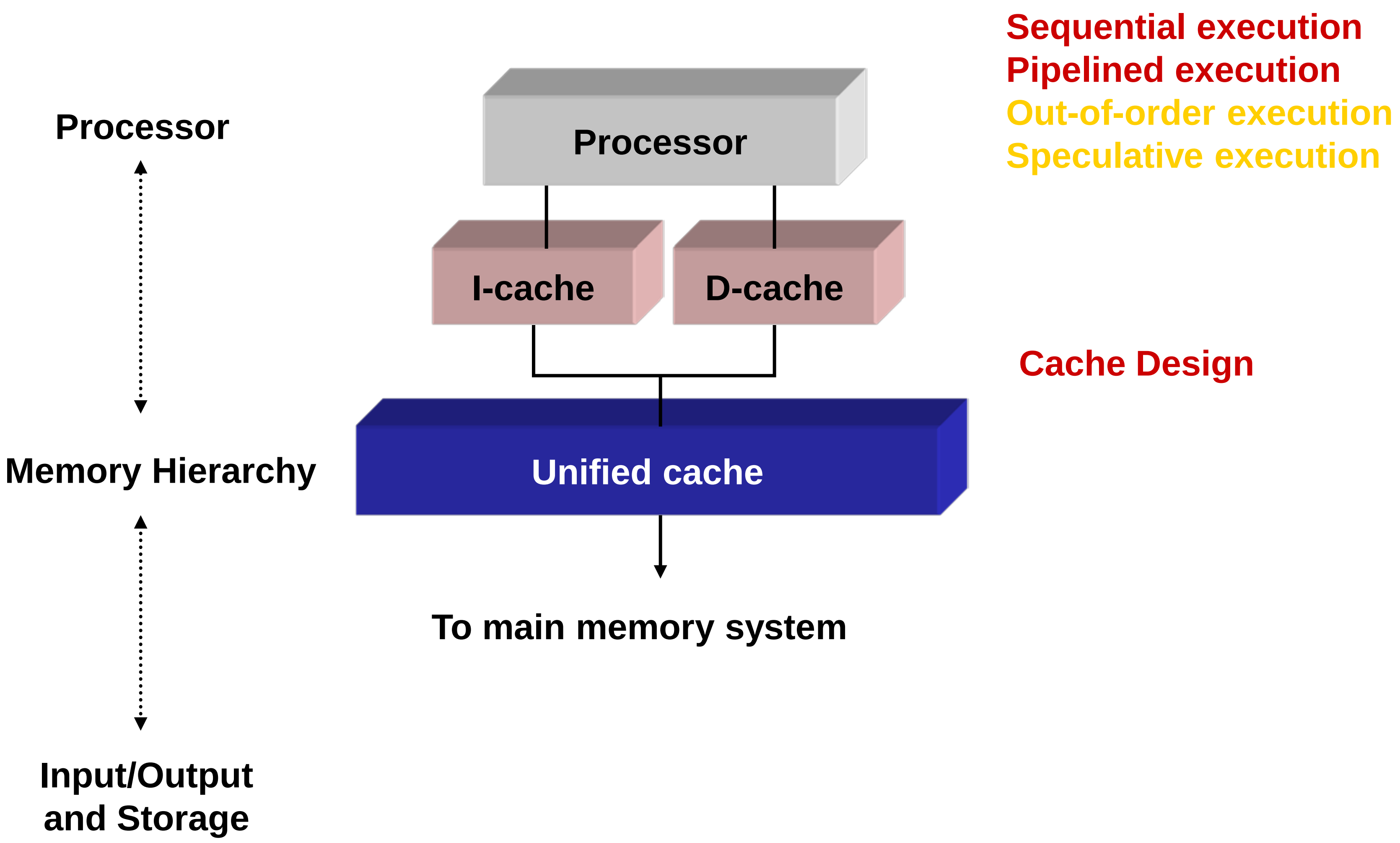


Before Register and Memory **After Register and Memory**

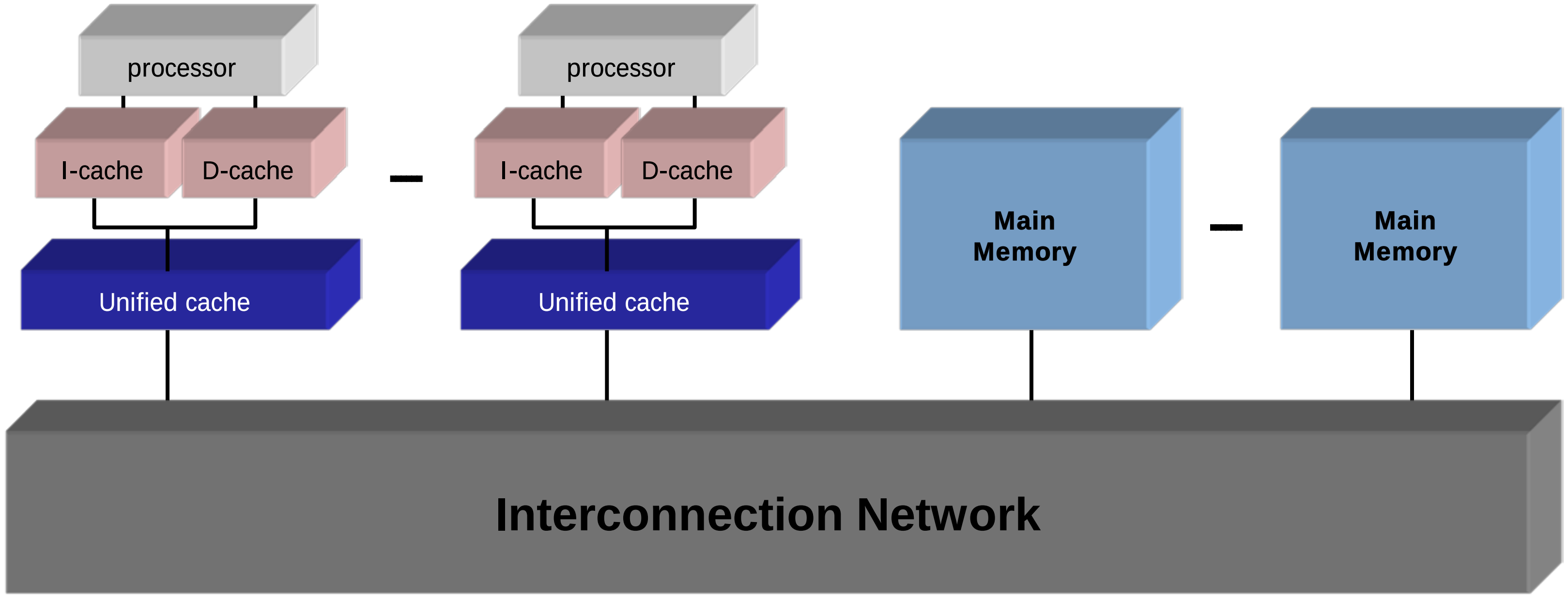
Design Techniques

- Design Techniques
  - Engineering methodology
  - Correctness criteria
  - Evaluation methods
  - Technology trends

Design Techniques



Design Techniques



Cache Coherence, Synchronization, Interconnection network

Engineering methodology

- Rule 1 : Identify and optimize the common case
- Rule 2 : Make the rare case correct and reasonably fast

Correctness criteria

- Examples
  - Pipelined execution : pipelined execution of instructions is correct if the results is as if the instructions were executed sequentially
  - Cache memory : execution of instructions on a system with cache memory is correct if the results is as if the instructions were executed on the same system but without cache memory
  - We'll see a lot of as if's

Performance Evaluation Methods

- Performance types
  - Time
    - response time
    - execution time
  - Rate
    - throughput : MIPS, MFLOPS
    - bandwidth : Mbps
  - Ratio
    - relative performance

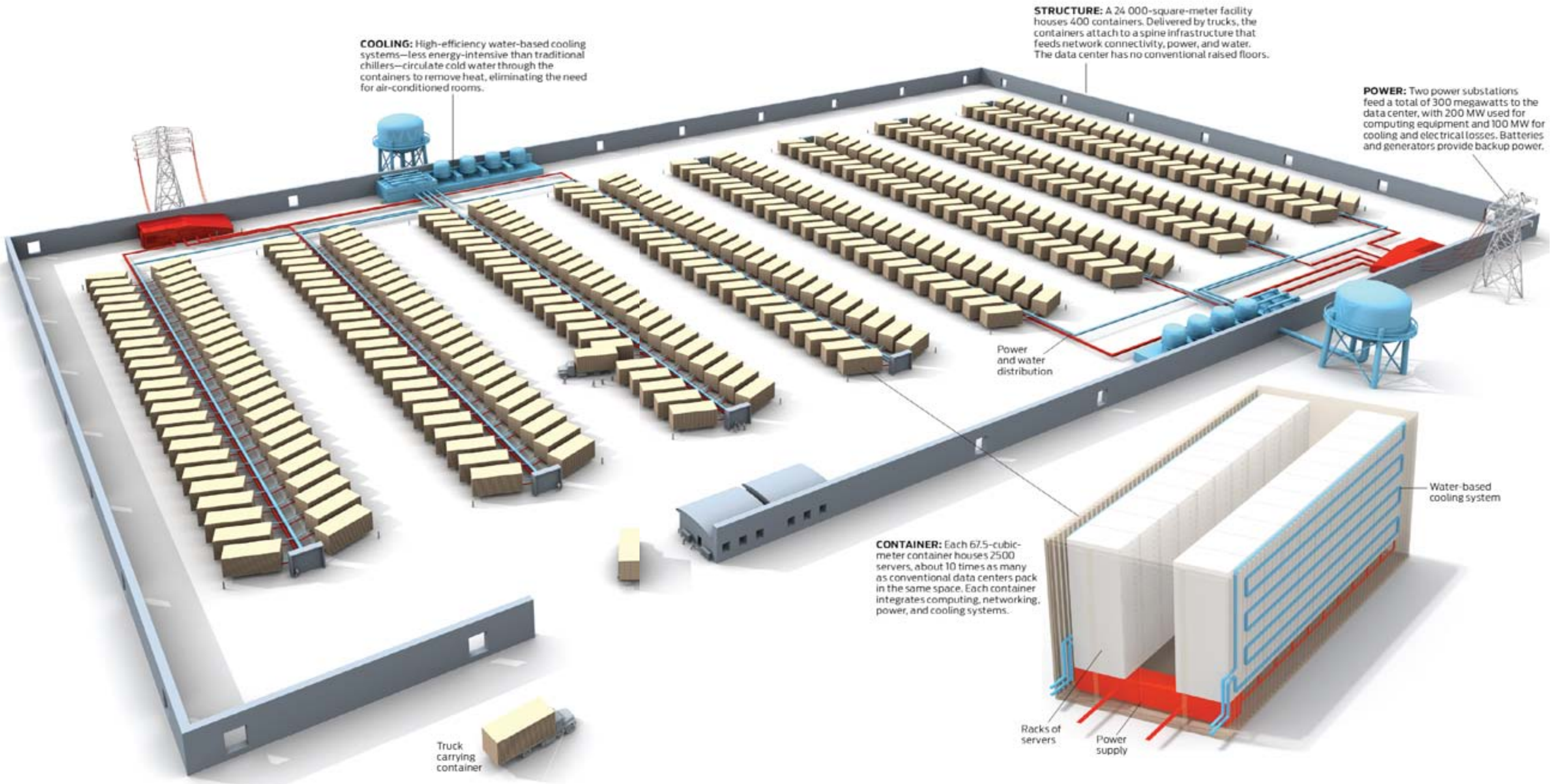
Technology Trends

1965	1977	1998	2005
			
IBM System 360/50	DEC VAX 11/780	Apple iMac	Pentium4
0.15 MIPS 64 KB $1M $6.6M per MIPS $16M per MB	1 MIPS(peak) 0.5 MIPS(estimated) 1 MB $200K $200K to $400 per MIPS $200K per MB	700 MIPS(peak) 427 MIPS(estimated) 32 MB $1229(September 1998) $1.75 to $2.90 per MIPS $38 per MB	~15000 MIPS(peak) ~6000 MIPS(estimated) 512 MB < $1000 $0.07 to $0.17 per MIPS < $2 per MB

A “Big” Picture

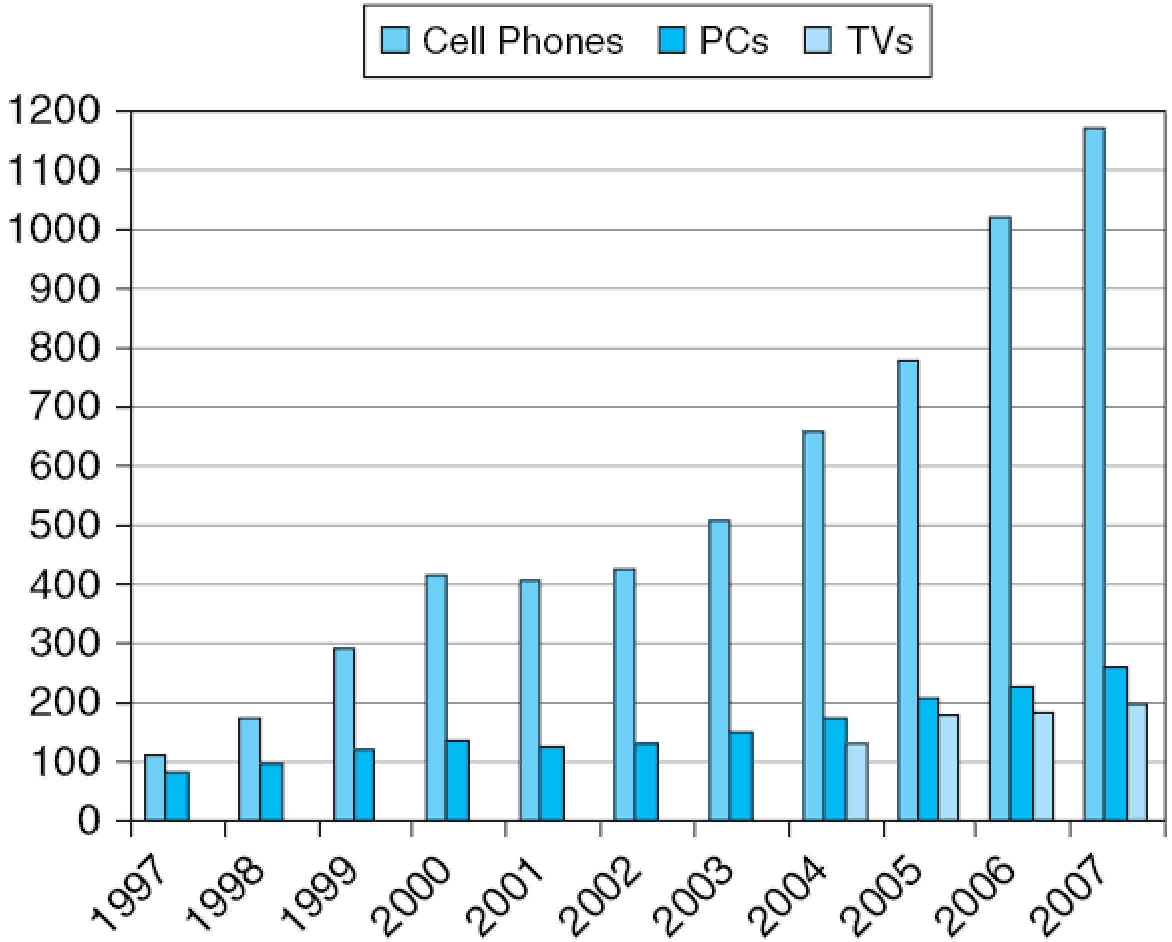
Randy H. Katz, “Tech Titans Building Boom,” IEEE Spectrum, Vol. 46, No. 2, Feb. 2009, pp. 36 – 39.

A “Big” Picture

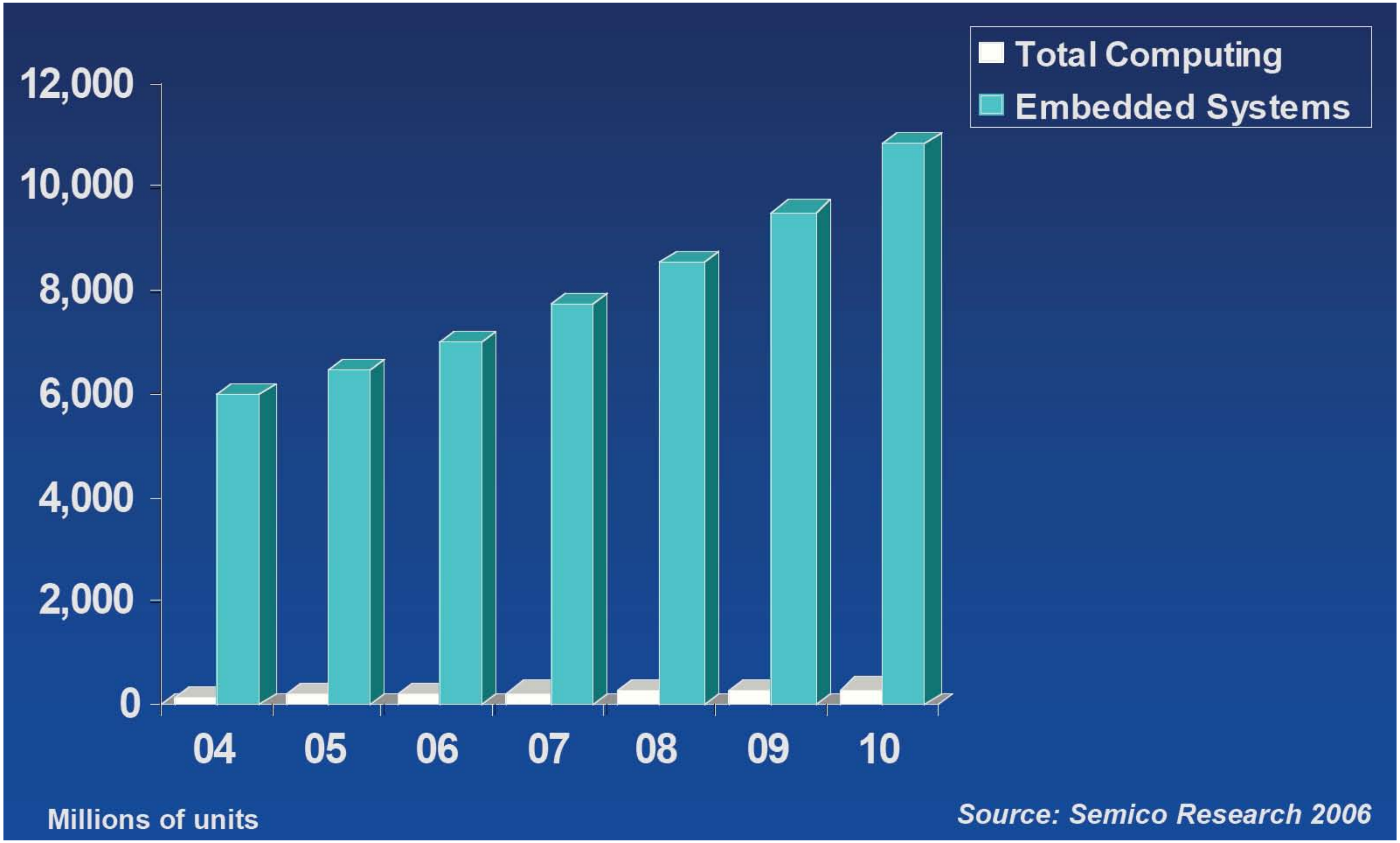


Randy H. Katz, “Tech Titans Building Boom,” IEEE Spectrum, Vol. 46, No. 2, Feb. 2009, pp. 36 – 39.

Embedded Processors

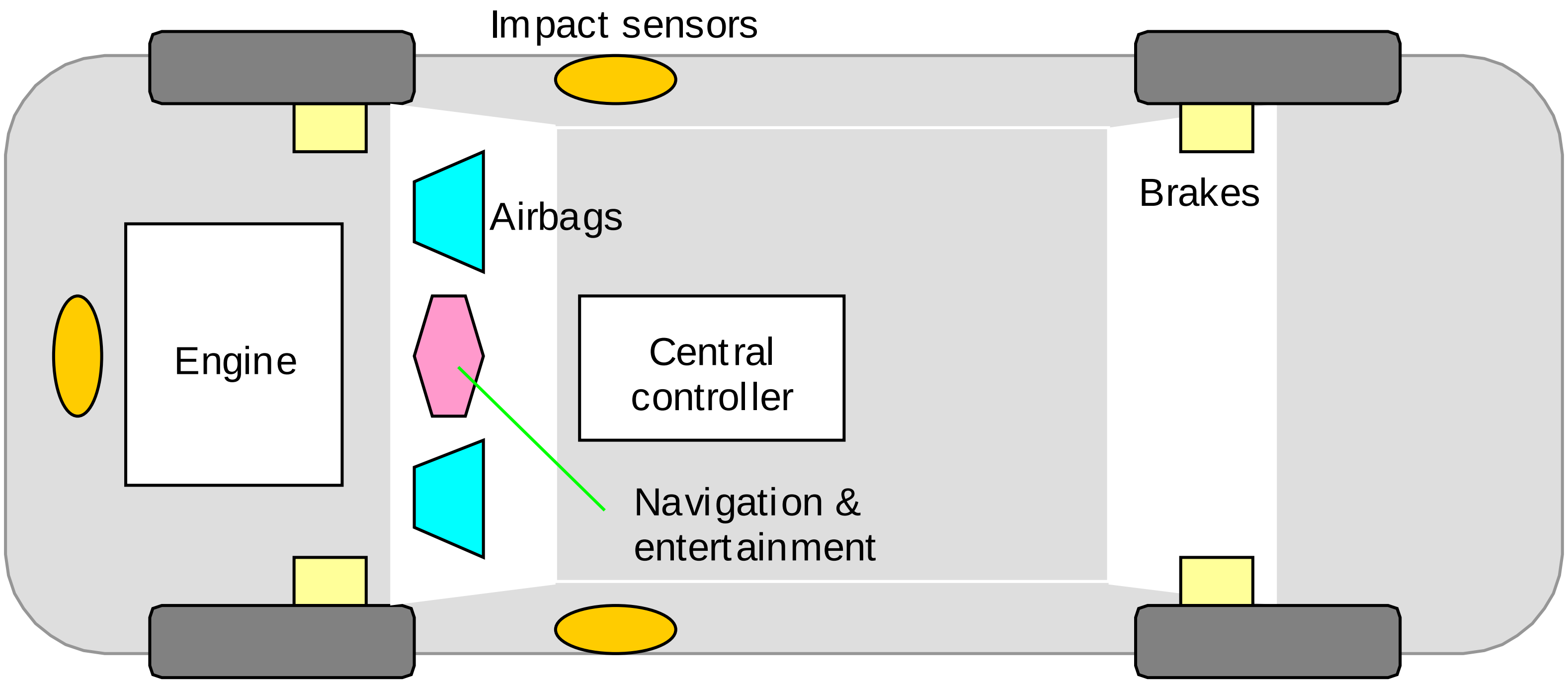


Embedded Processors



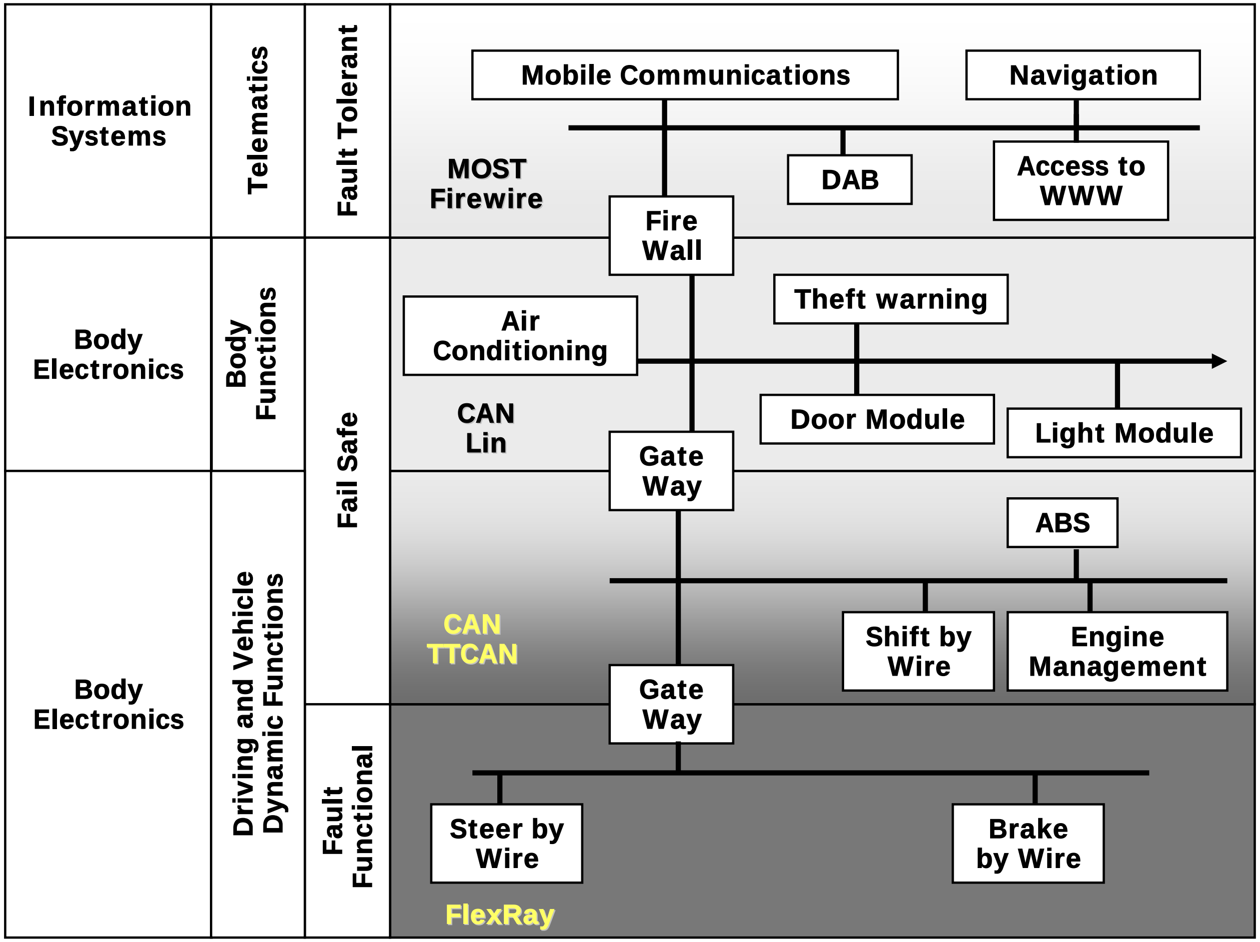
From "Flash and the Embedded Space" by Grady Lambert

Embedded Processing Example



From Prof. Behrooz Parhami's lecture notes

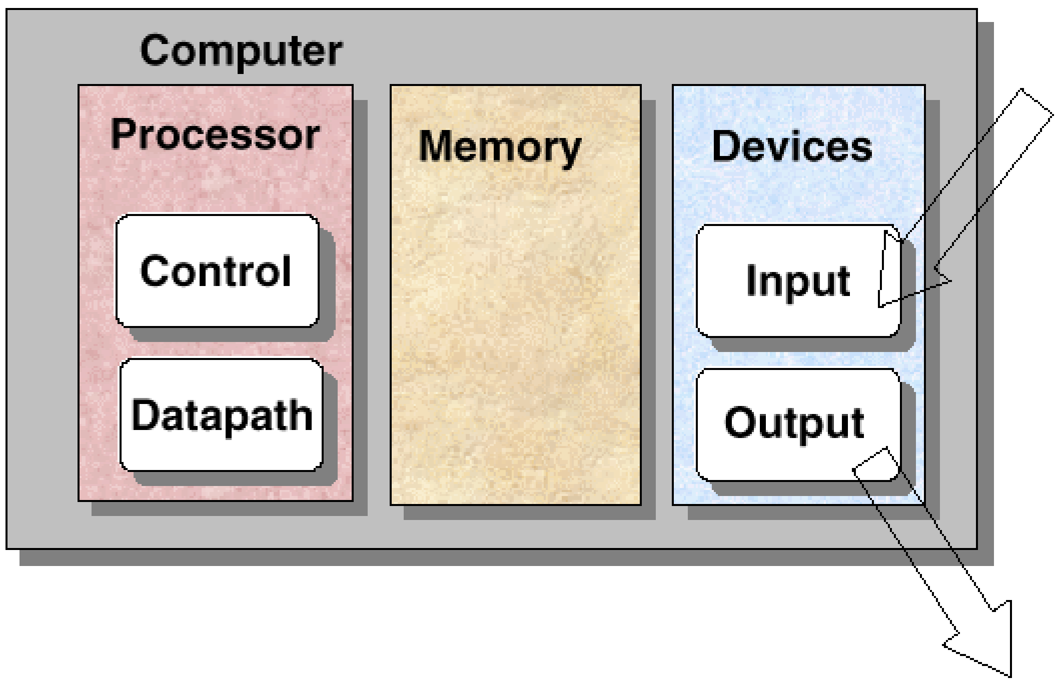
Automotive Electronic System



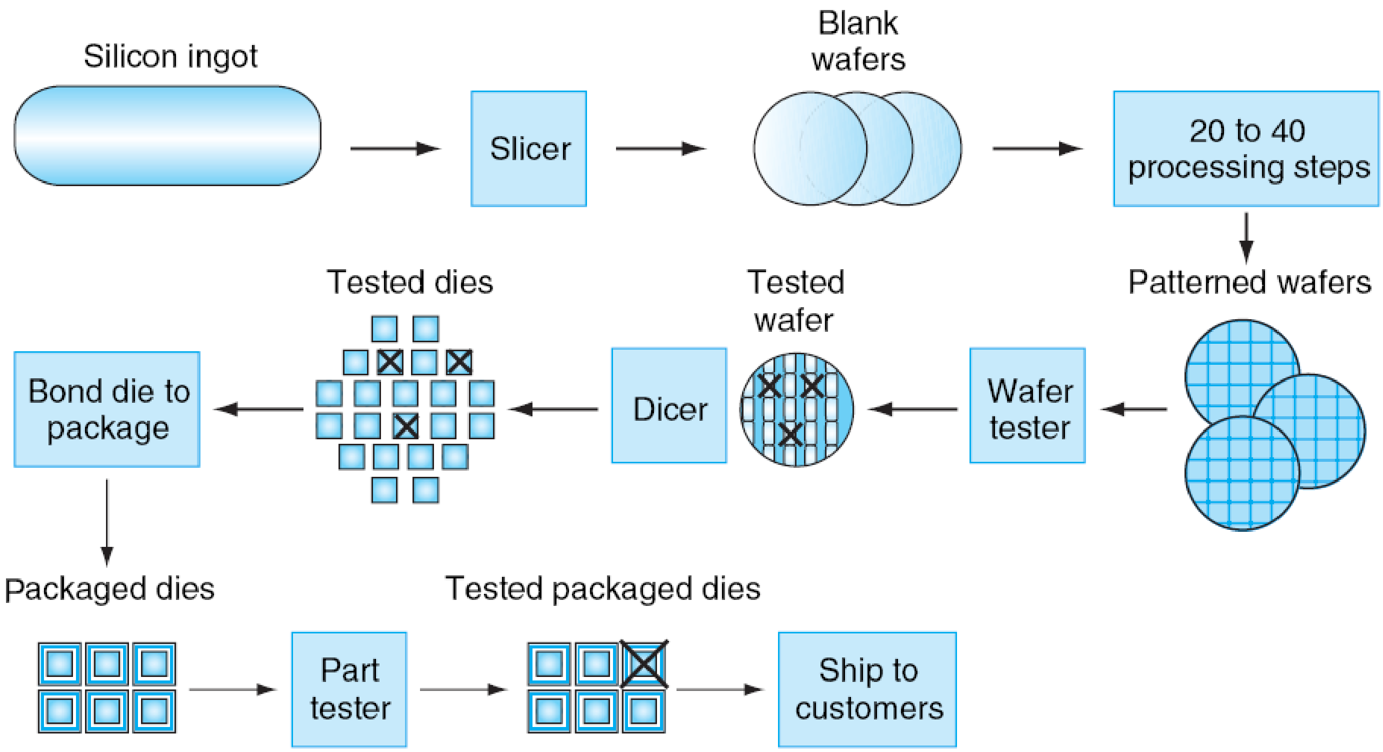
From “Design of Embedded Systems: Methodologies, Tools and Applications” by Alberto Sangiovanni-Vincentelli

Technology Trends

- Five components of a computer system

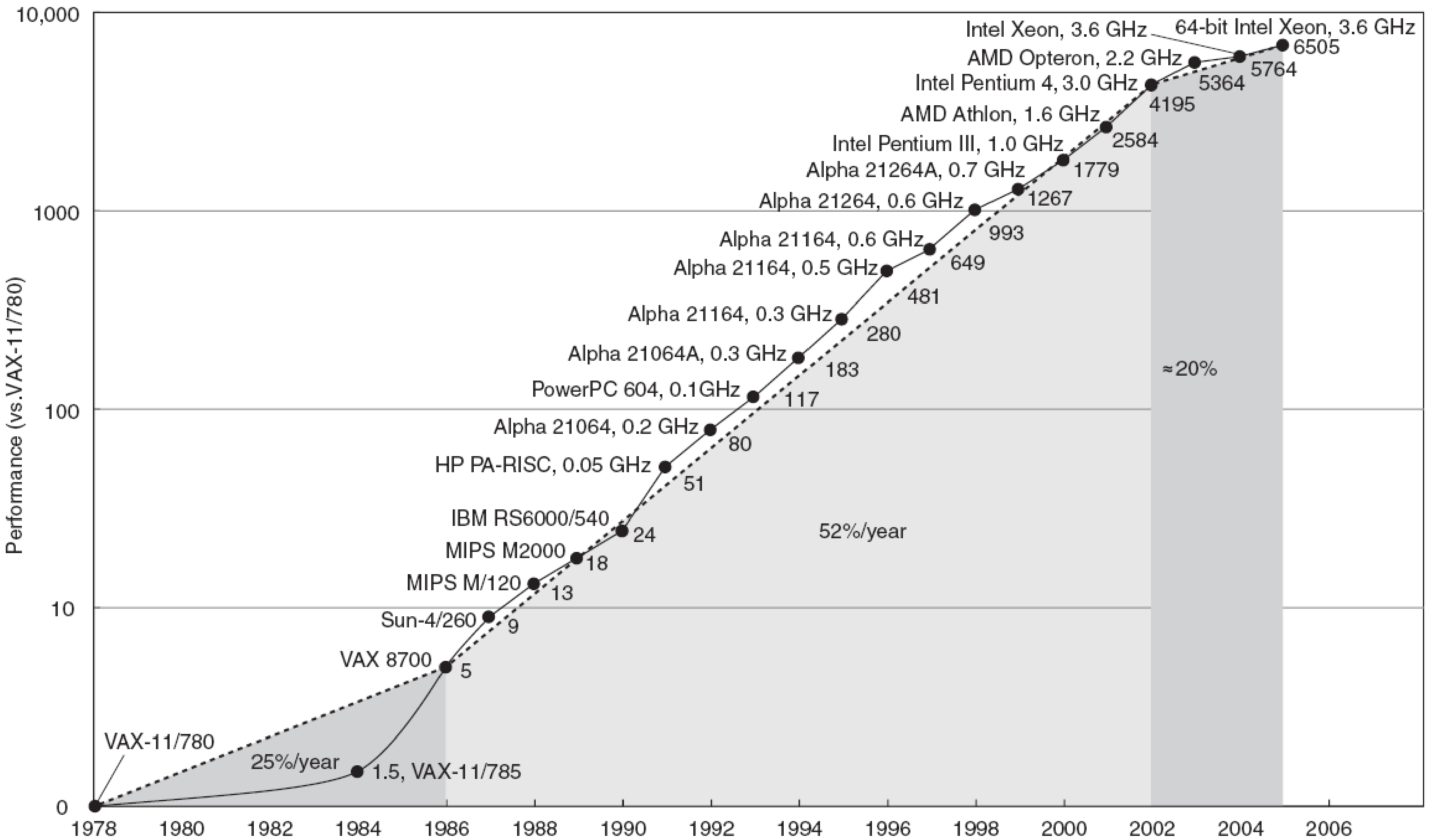


Chip Manufacturing Process

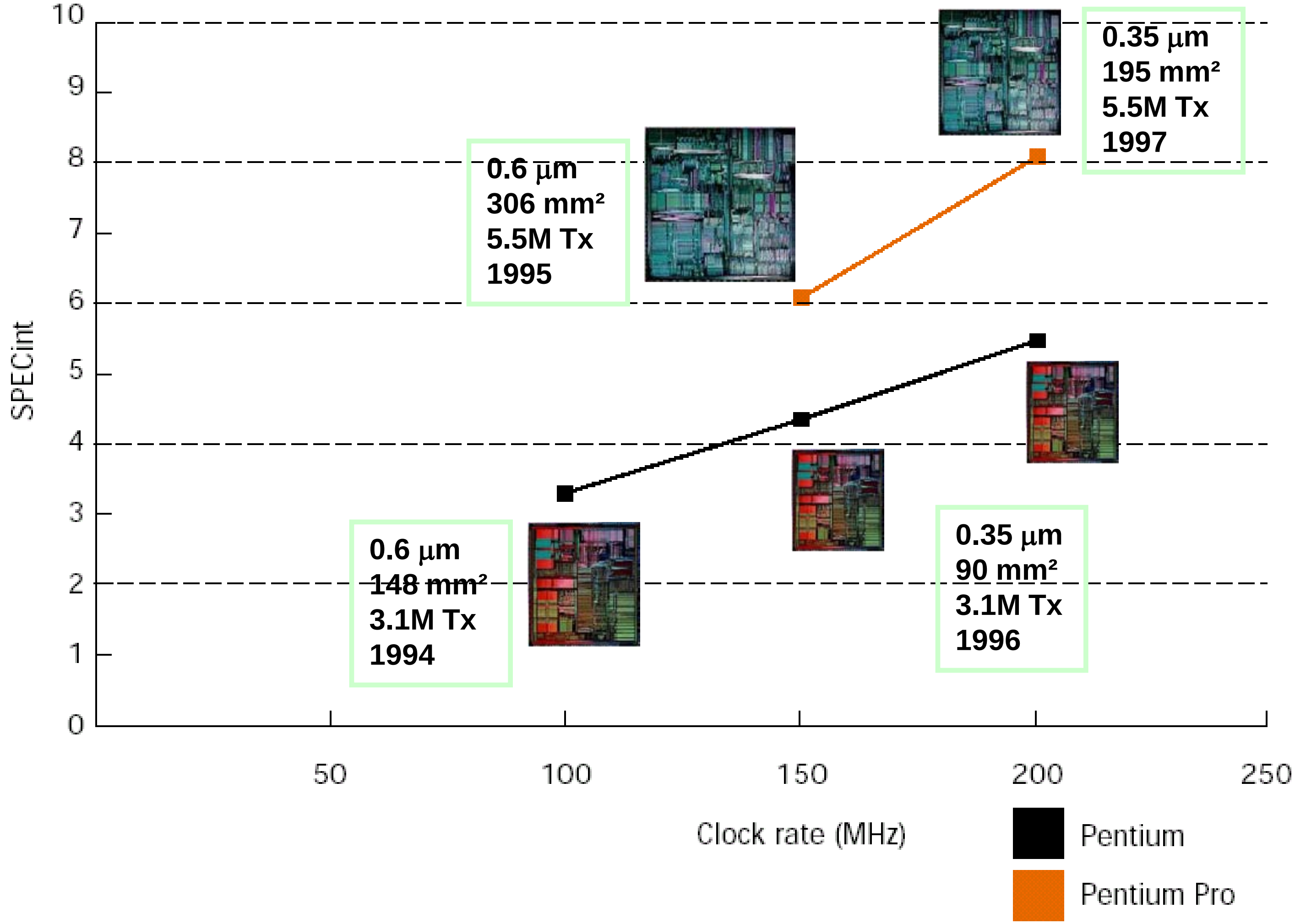


$$\textbf{Chip cost} = \frac{\textbf{Die cost} + \textbf{Testing cost} + \textbf{Packaging cost}}{\textbf{Final test yield}}$$

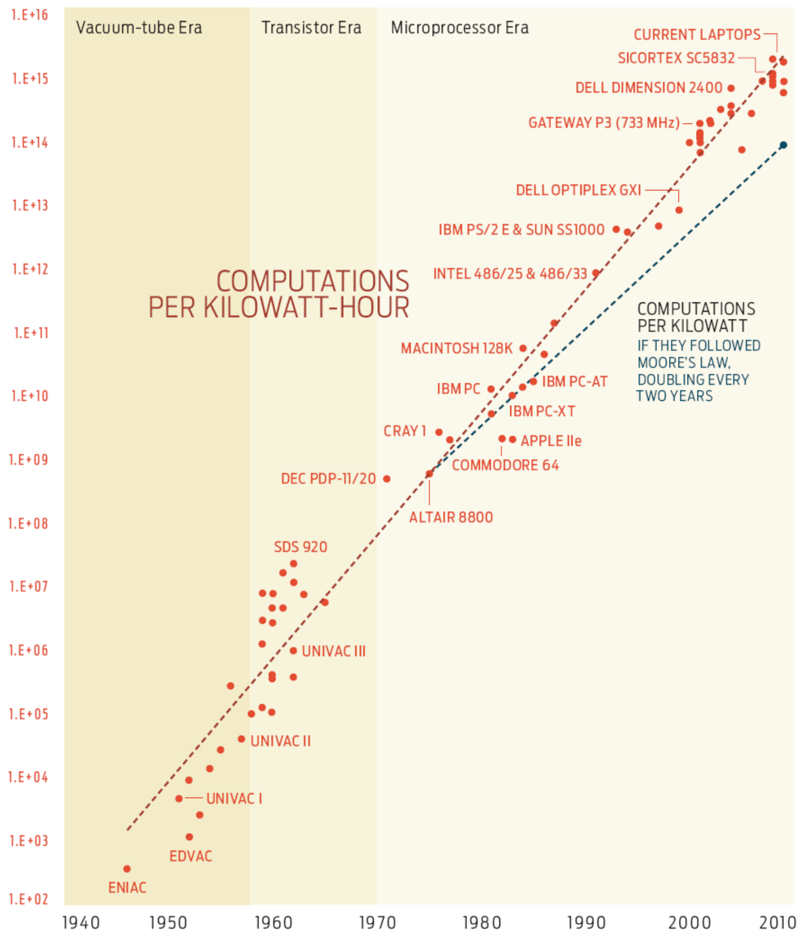
Processor Performance Trends



Performance Improvements by Advances on Lithography (VLSI) Technology

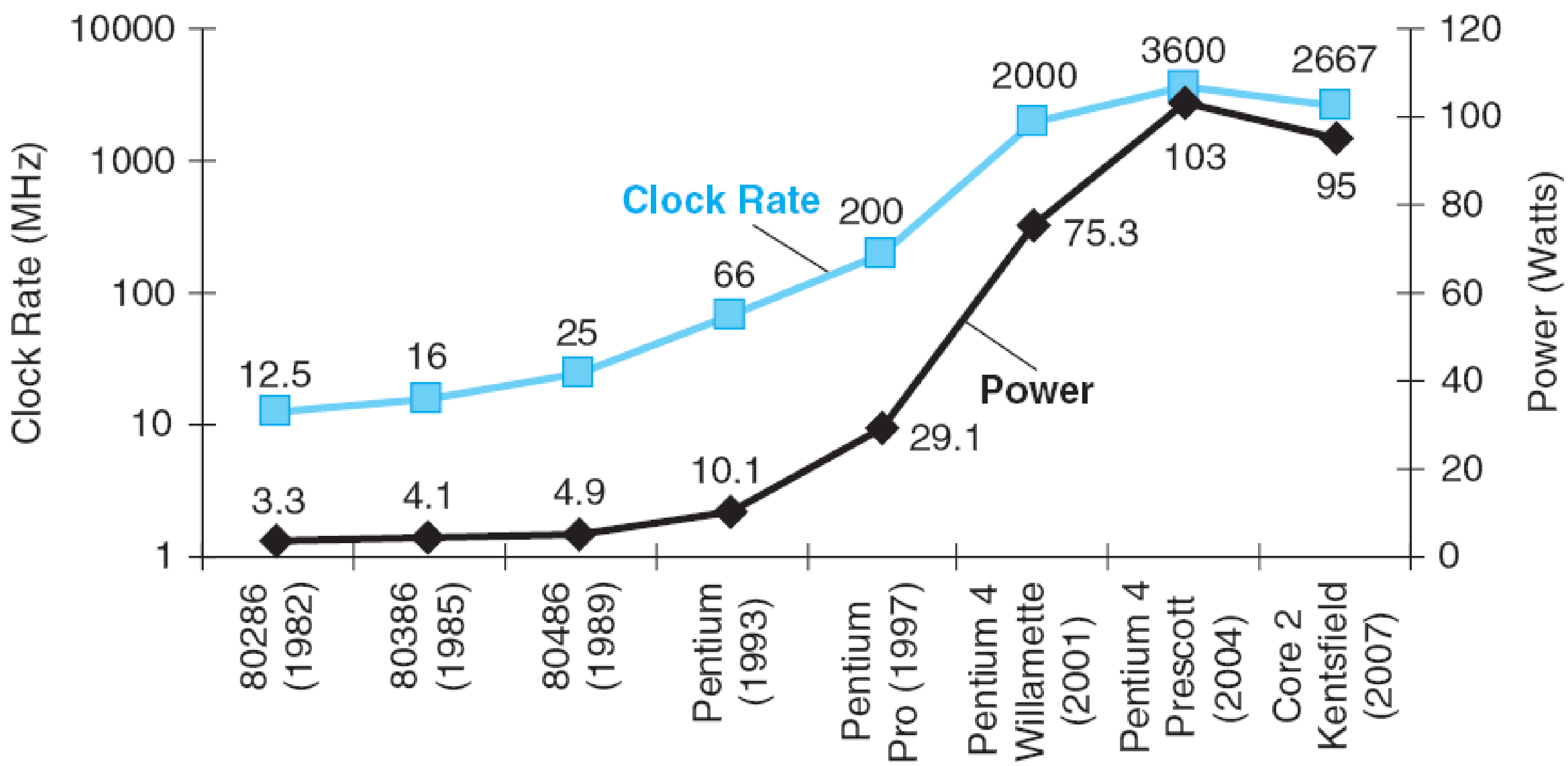


Processor Computations/Energy Trends



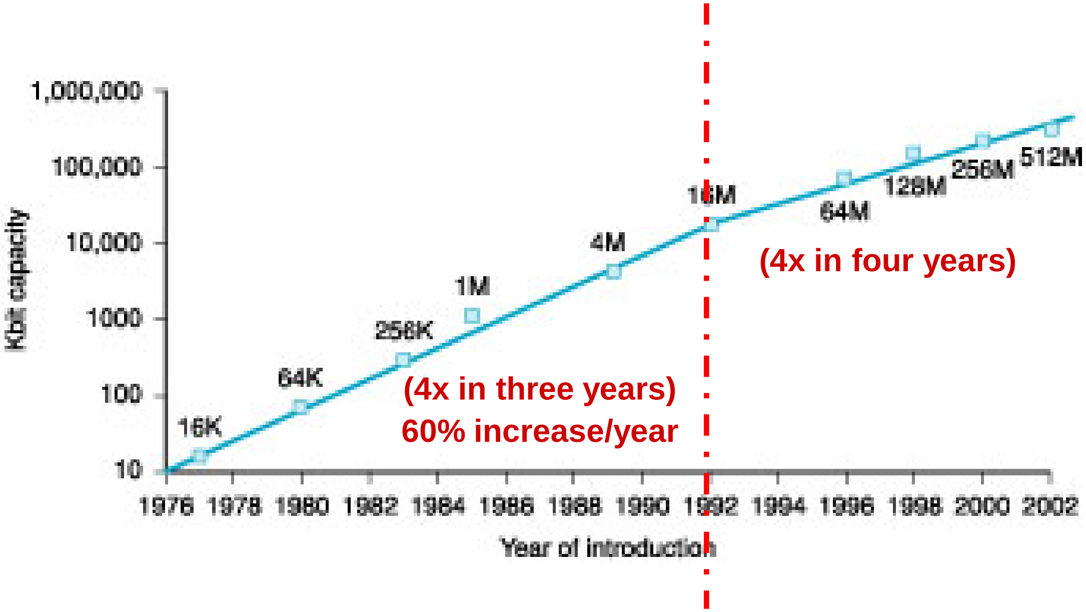
J. G. Koomey, et al. "Outperforming Moore's Law" IEEE Spectrum, Vol. 47, No. 3, Mar. 2010, pp. 68 – 68.

Processor Clock Rate/Power Trends



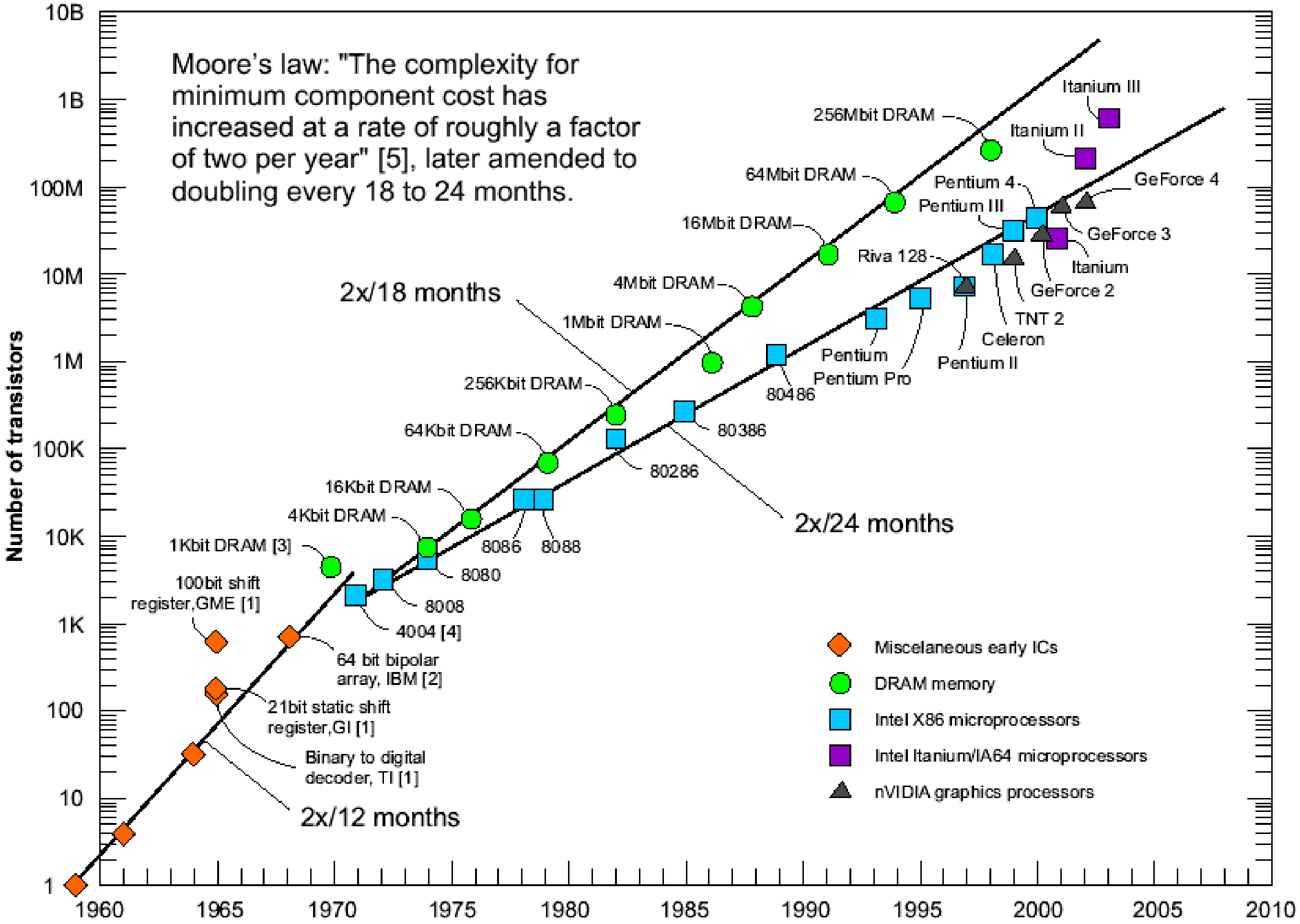
$$\text{Power} = \text{Capacitive load} \times \text{Voltage}^2 \times \text{Frequency}$$

DRAM Technology Trends



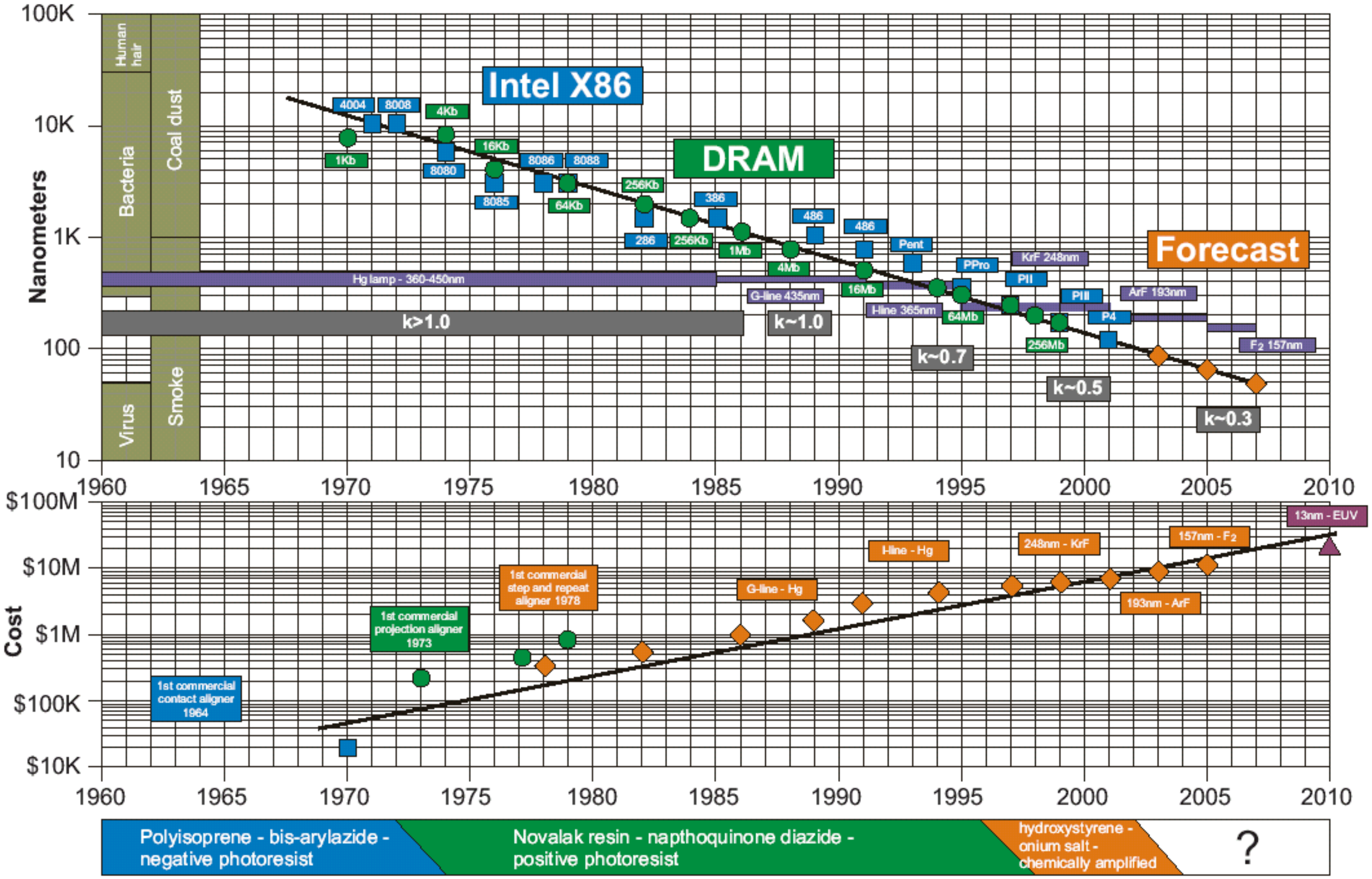
year	size	cycle time
1980	64 Kbits	250 ns
1983	256 Kbits	220 ns
1986	1 Mbits	190 ns
1989	4 Mbits	165 ns
1992	16 Mbits	145 ns
1996	64 Mbits	125 ns
2000	256 Mbits	100 ns

Transistors Per Die Trends



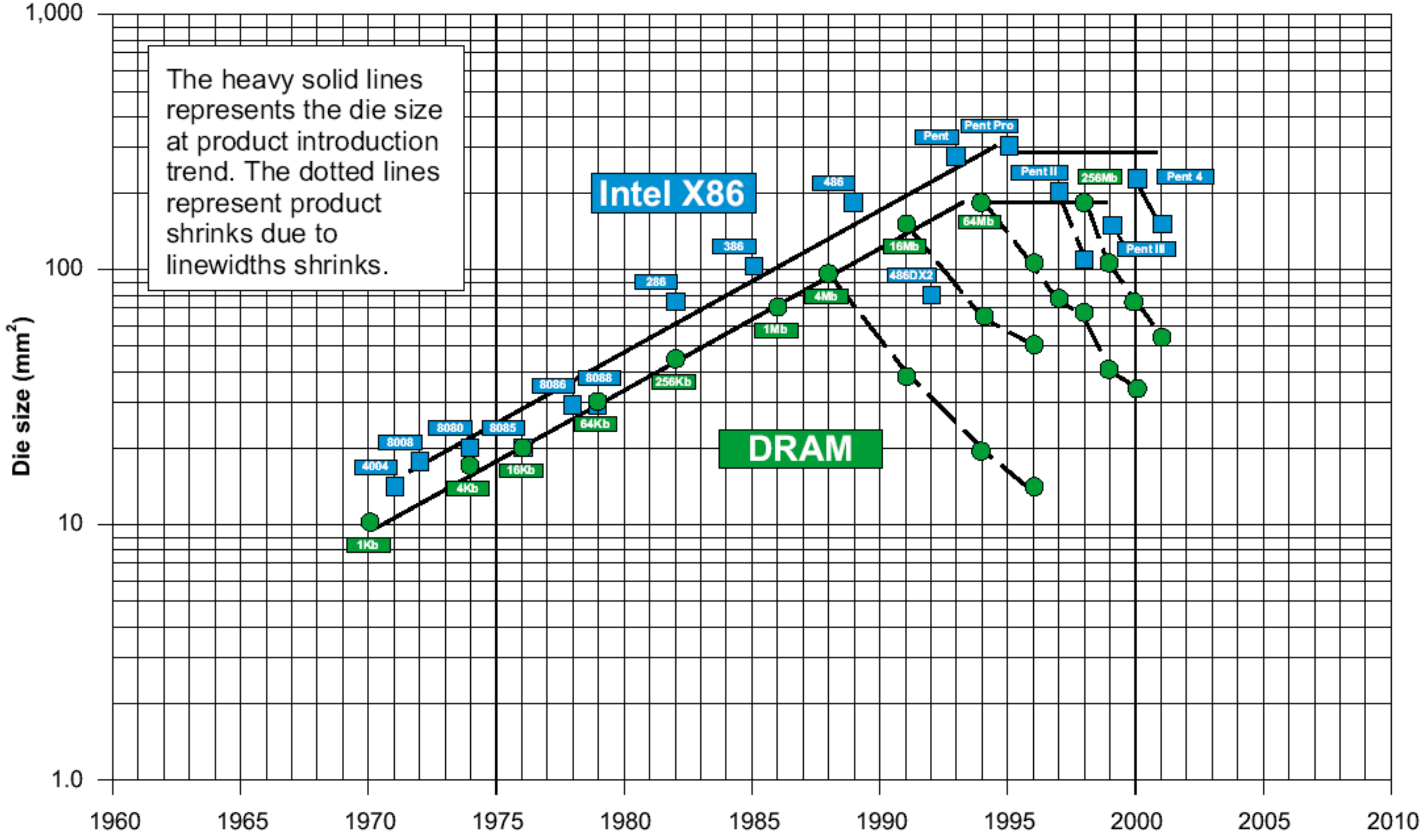
Source: www.icknowledge.com

Lithography Technology Trends



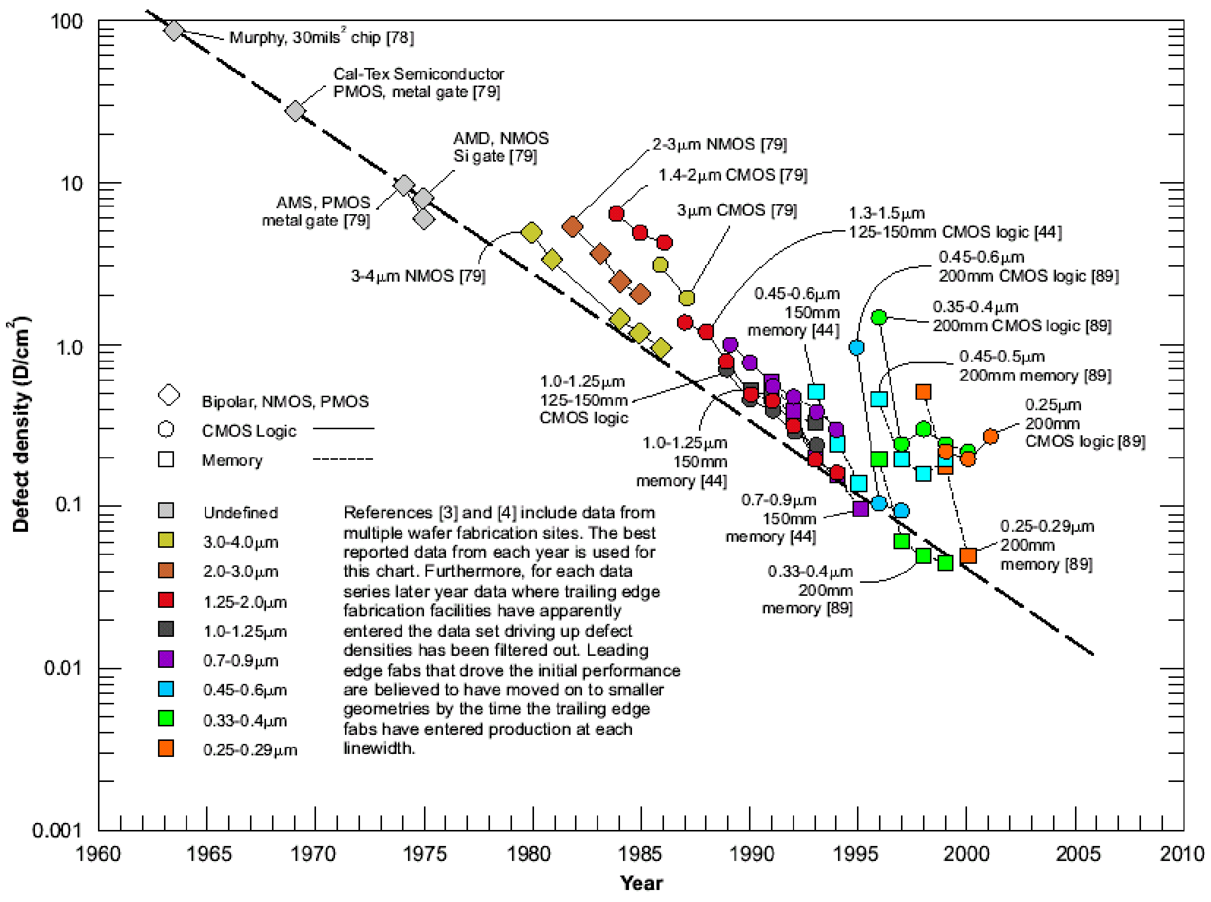
Source: www.icknowledge.com

Die Size Trends



Source: www.icknowledge.com

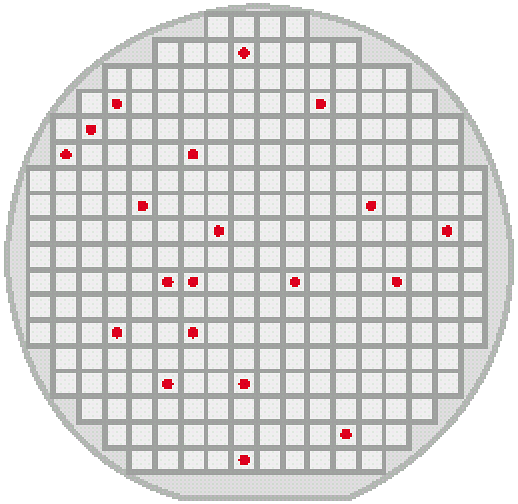
Defect Density Trends

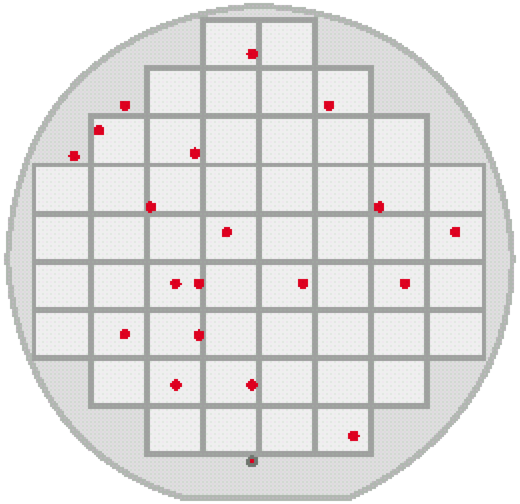


Source: www.icknowledge.com

Die Cost and Yield

- Die Cost $\propto$ f (Die size^{4})



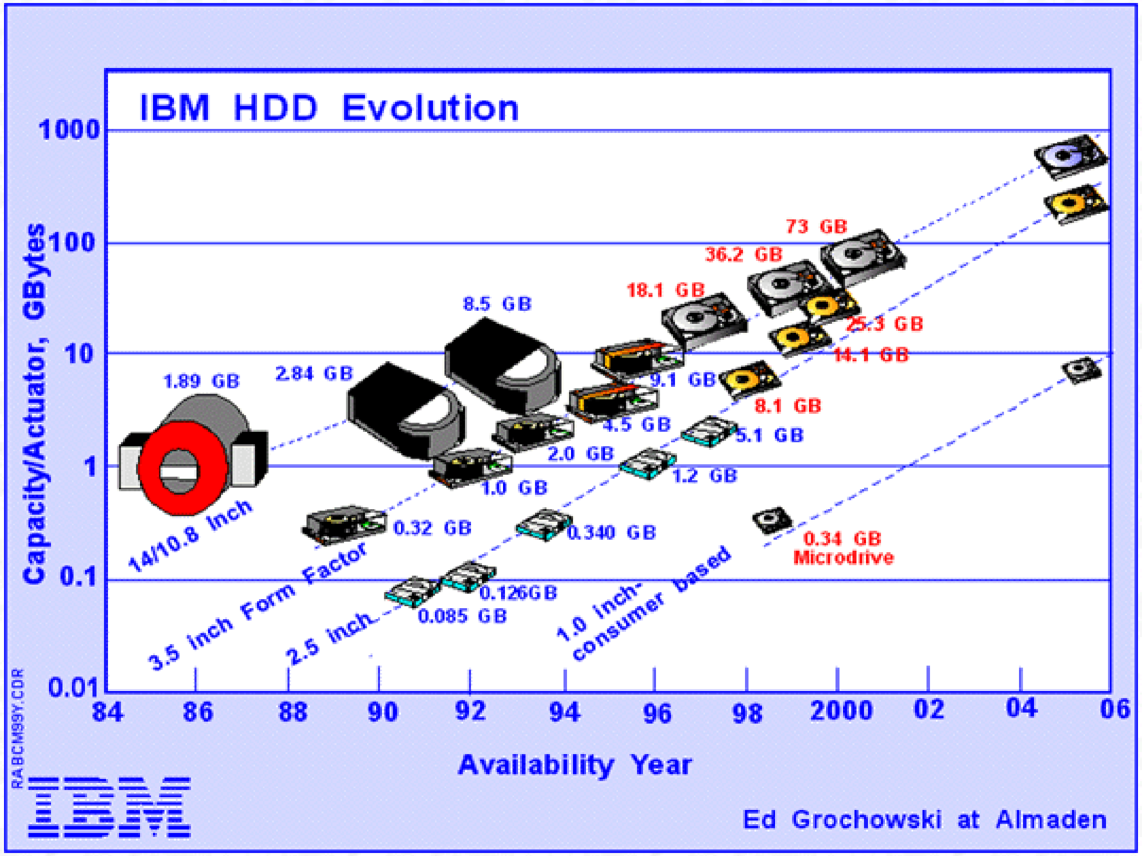


20 Defects
20 Bad Die
264 Gross Die
92% Yield

20 Defects
16 Bad Die
54 Gross Die
70% Yield

Source: www.icknowledge.com

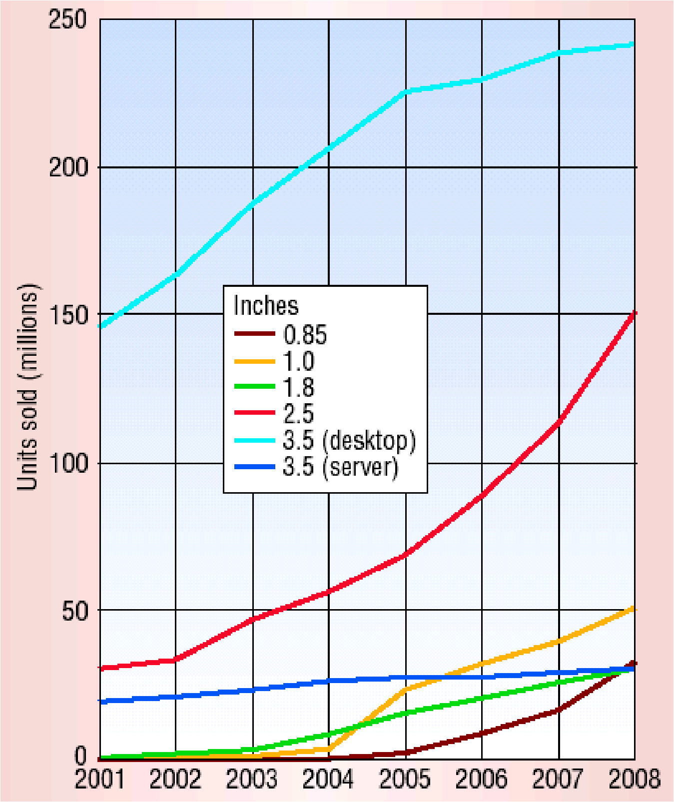
Hard-Disk Technology Trends

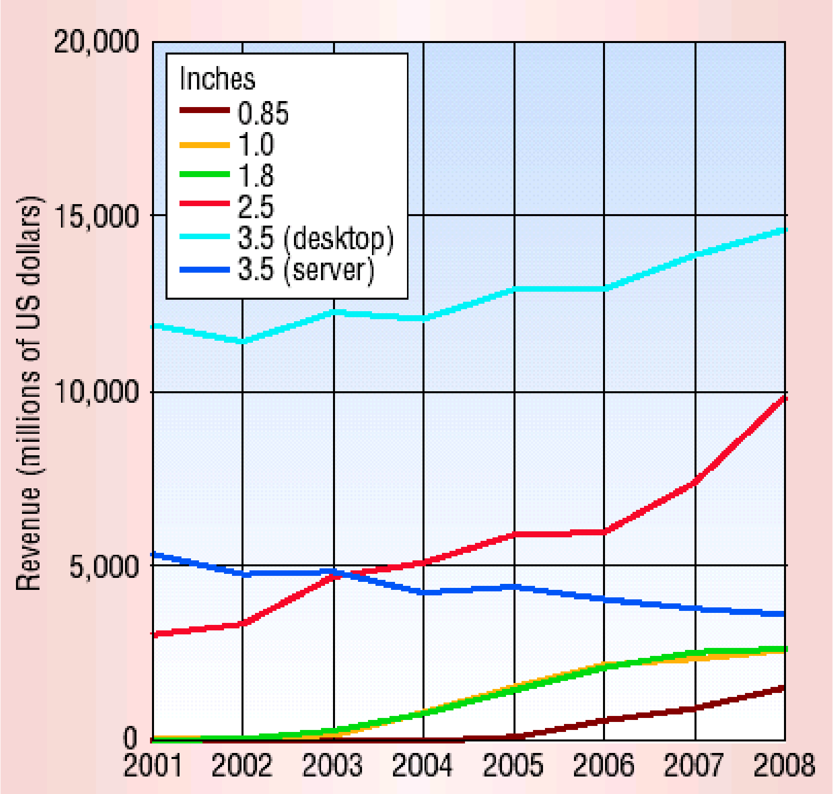


Source: IBM HDD Evolution by Ed Grochowski at Almaden

Disk density: 1.50x - 1.60x per year (4x in three years)

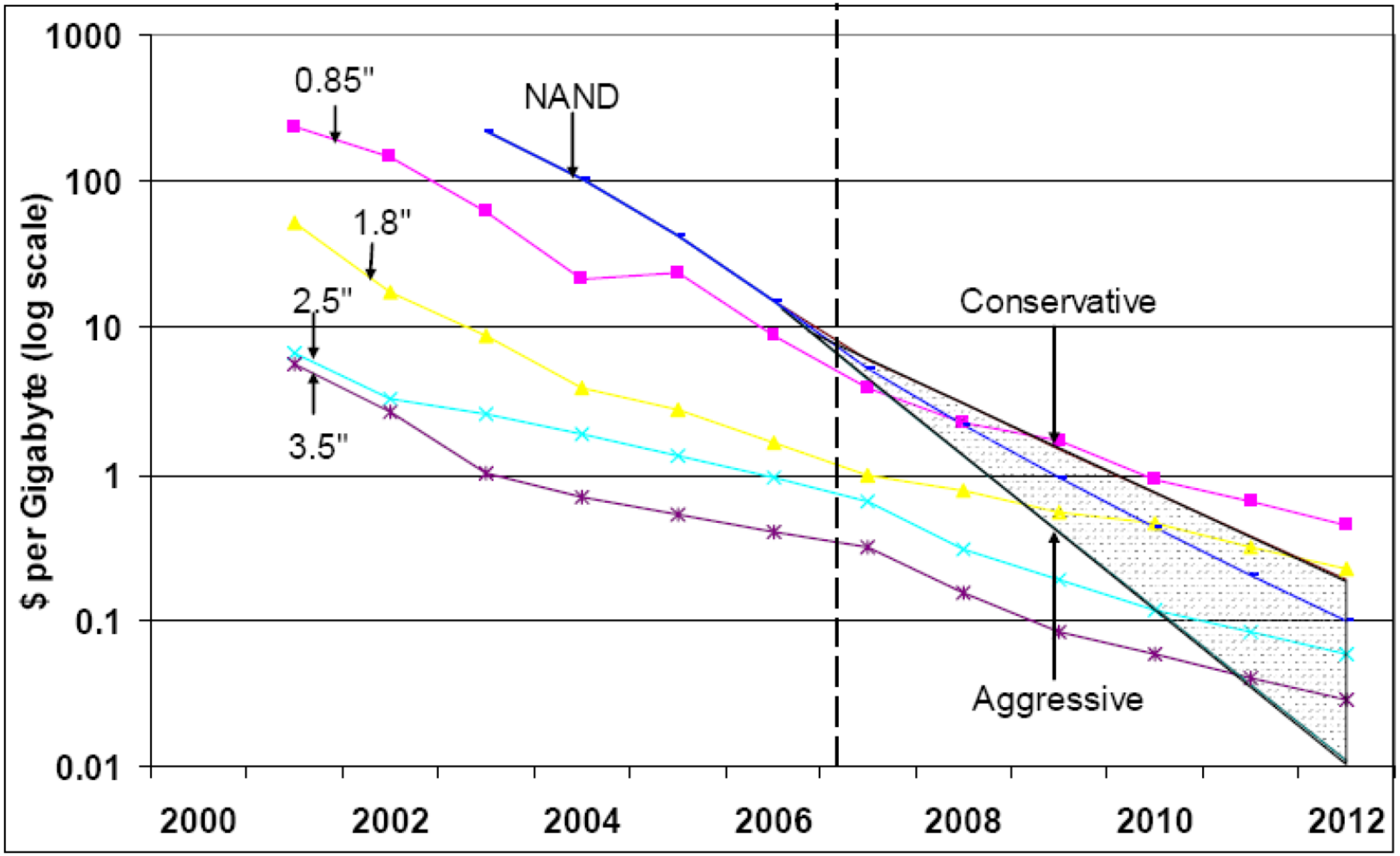
Hard-Disk Technology Trends





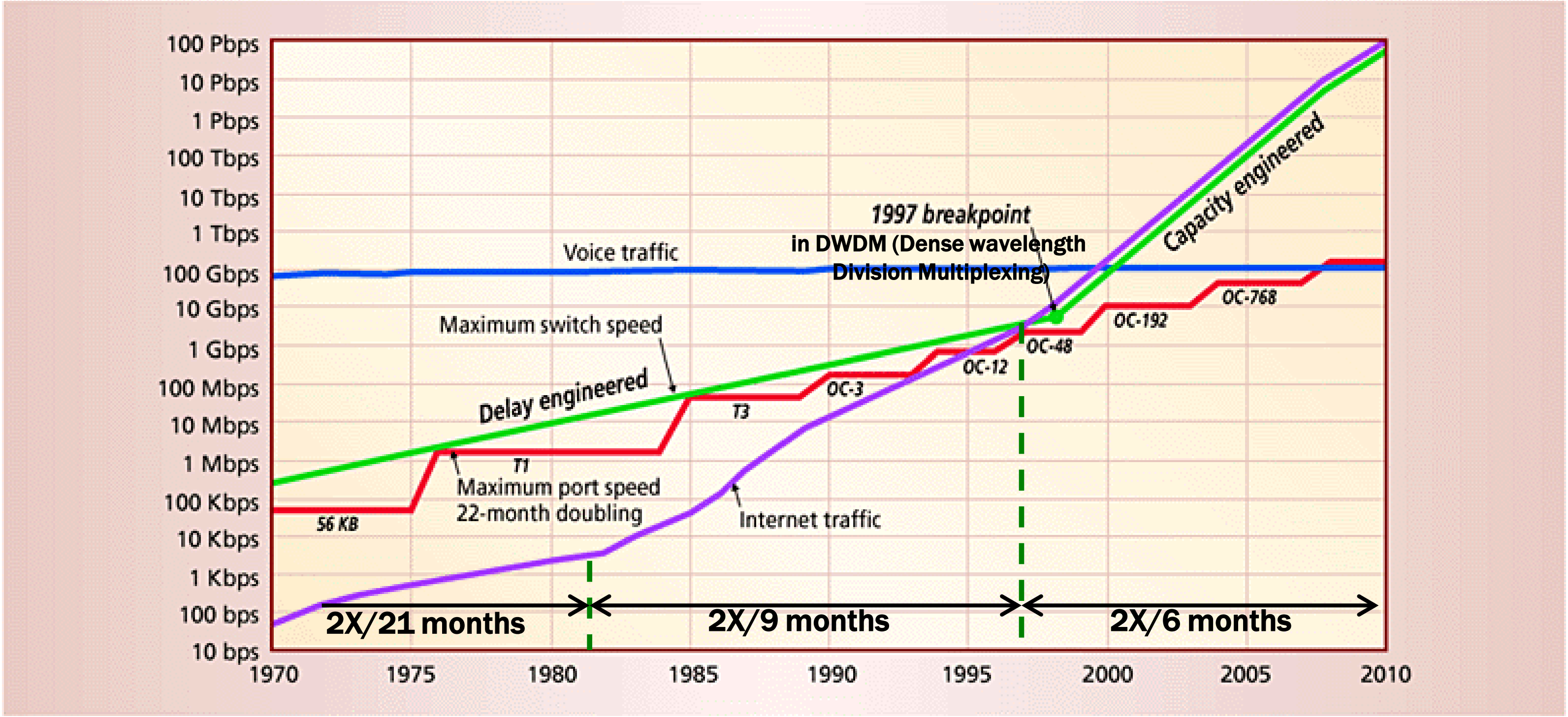
"Will Hard drives Finally Stop Shrinking?" by Linda Dailey Paulson (IEEE Computer, May 2005)

Future Outlook of Flash Memory



Source: Scott Deutsch (SanDisk), "Bringing Solid State Drives to Mainstream Notebooks," Flash Memory Summit 2007.

Internet Technology Trends



Source: Lawrence G. Roberts, Beyond Moore's Law: Internet Growth Trends, IEEE COMPUTER JANUARY 2000, pp. 117-119

Pitfalls of Computer Technology Forecasting

- DOS addresses only 1 MB of RAM because we cannot imagine any applications needing more.” Microsoft, 1980
- “640K ought to be enough for anybody.” Bill Gates, 1981
- “Computers in the future may weigh no more than 1.5 tons.” *Popular Mechanics*
- “I think there is a world market for maybe five computers.” Thomas Watson, IBM Chairman, 1943
- “There is no reason anyone would want a computer in their home.” Ken Olsen, DEC founder, 1977
- “The 32-bit machine would be an overkill for a personal computer.” Sol Libes, *ByteLines*

From Prof. Behrooz Parhami’s lecture notes