Microprocessor Overview

Nelson H. F. Beebe
Center for Scientific Computing
University of Utah
Department of Mathematics
155 S 1400 E
Salt Lake City, UT 84112-0090
USA
http://www.math.utah.edu/~beebe

Prehistory: 1906--1947

- 1906: Lee De Forest invents vacuum tube
- 1946: J. Presper Eckert and John Mauchly at U Pennsylvania invent ENIAC: boxcar size, 30 tons, 18,000 vacuum tubes, 150kW, computing power of 100 humans
- 1947: John Bardeen, William Shockley, and Walter Brattain invent *transistor* at Bell Labs

Prehistory: 1956--1961

- 1956: *Nobel Prize in Physics* for transistor
- 1959: Jack Kilby (Texas Instruments) and Robert Noyce (Fairchild Semiconductor) invent integrated circuit (IC)
- 1960: DEC PDP-1
- 1961: IBM 7030 Stretch (first scientific computer)

Prehistory: 1964--1970

- 1964: IBM S/360 announced
- 1964: ASCII defined, too late for S/360
- 1964: CDC 6600
- 1968: DEC PDP-10
- 1969: Apollo moon landing
- 1970: DEC PDP-11

Early history: 1971--1975

- 1971: Intel 4004 4-bit microprocessor; equals ENIAC in compute power: 2300 transistors, 0.75MHz, 8KB ROM, 16KB RAM
- 1972: Intel 8008 8-bit microprocessor
- 1973: French Micral with 8008: first desktop personal computer (some credit Altair)
- 1975: IBM 801 (first RISC microprocessor)
- 1975: Zilog Z80

Early history: 1976--1979

- 1976: Cray 1
- 1978: DEC VAX
- 1978: Intel 8086
- 1979: Motorola 68000

More history: 1980--1986

- 1980: IBM adopts Intel 8086 for IBM PC
- 1980: Patterson et al: Berkeley RISC-I, -II
- 1981: Hennessy et al: Stanford MIPS
- 1983: Time magazine: PC "Man of the Year"
- 1984: Apple Macintosh (Motorola 68000)
- 1985: Acorn RISC first commercial RISC
- 1986: HP Precision Architecture (PA-RISC)

Even more history: 1986--199x

- 1986: IBM RT
- 1986: MIPS-I (R2000) commercialized
- 1987: Sun SPARC v8
- 1988: Motorola 88000
- 1989: Intel i860
- 1990: IBM Power (RS/6000)
- 199x: 4-bit micros reach 1B/year

Recent history: 1992--1995

- 1992: DEC Alpha
- 1993: IBM/Motorola/Apple PowerPC
- 1994: Intel Pentium divide flaw: cover-up, and recovery costing > $480M
- 1995: Microprocessor revenues > $100B/year
- 1995: More than 10B microprocessors sold in 25 years

Recent history: 1996--1999

- 1996: 25th anniversary of microprocessor: IEEE Micro April and December issues
- 1997: 48M MIPS processors shipped: first RISC architecture to exceed Motorola 68K CISC volume
- 1999: IBM ships *one millionth* Si/Cu chip
- 1999: Motorola ships *two billionth* MC68HC05 8-bit microprocessor

Recent history: 2000--date

- 2000: *Nobel Prize in Physics* to Jack Kilby (for *integrated circuit*), Zhores Alferov and Herbert Kroemer (for *optoelectronics*)
- 2000: HP/Intel IA-64 MLIW/RISC
- 2001: 30th anniversary of microprocessor

Industry revenues

- DRAM: $30B (Wall Street Journal 7-Dec-2000)
- Semiconductors: > $150B in 1995
- Microprocessors: > $100B in 1995 (3% of US GDP)

Architecture books 1833--1985

- Gerrit A. Blaauw and Frederick P. Brooks, Jr.
  Computer Architecture: Concepts and Evolution
  Addison-Wesley 1997
  ISBN 0-201-10557-8

Architecture books 1985--1997

- John L. Hennessy and David A. Patterson
  Computer Architecture
  Morgan Kaufman (1995, 1997)
  ISBN 1-55860-069-8, 1-55860-329-8

  Computer Organization and Design
  Morgan Kaufmann (1994, 1997)
  ISBN 1-55860-281-X, 1-55860-281-X

Other books

- Michael S. Malone
  The Microprocessor: a biography
  Telos 1995
  ISBN 0-387-94342-0
- Carl Shapiro and Hal Varian
  Information Rules
  Harvard Business School Press 1999
  ISBN 0-87584-863-X

Architecture journals, conferences, and standards

- *IEEE Micro* (1981--date)
- *Hot Chips Symposia* (1989--date)
- *Cool Chips Symposia* (1998--date)
- *IEEE Symposia on Computer Arithmetic* (*ARITH'nn*) (1972--date)
- Michael Slater's *Microprocessor Report*
- *ANSI/IEEE 754-1985 Standard for Binary Floating-Point Arithmetic* (draft in 1980)

Bibliographic resources

- TeX User Group archive:
  http://www.math.utah.edu/pub/tex/bib/
- BibNet Project archive:
  http://www.math.utah.edu/bibnet/
- Software:
  http://www.math.utah.edu/~beebe/software/
  http://www.math.utah.edu/pub/mg/

Other resources

- Nelson H. F. Beebe
  The Impact of Memory and Architecture on Computer Performance
  (1994)
  http://www.math.utah.edu/~beebe/memperf.pdf
- Mathematical software benchmarks:
  http://www.math.utah.edu/pub/benchmarks

Microprocessor overview

- Acorn RISC Machine (ARM)
- Compaq/DEC Alpha
- HP PA-RISC 1.0-1.2
- IBM Power, PowerPC
- IBM S/390 G5
- Inmos transputer
- Intel i860, i960, x86, and with HP, IA-64
- Java Virtual Machine (JVM)
- Motorola 68K, 88K
- SGI/MIPS Rxx000
- Sun MAJC, PicoJava, SPARC
- Transmeta Crusoe

Acorn (Advanced) RISC Machine

- VLSI Technology
  Acorn RISC Machine (ARM) Family Data Manual
  Prentice-Hall 1990
  ISBN 0-13-781618-9
- Company renamed from *Acorn ...* to *Advanced ...* in 1990
- DEC, Intel, et al manufacture *StrongARM*

Compaq/DEC Alpha

- Richard L. Sites and Richard L. Witek
  Alpha AXP architecture reference manual
  Digital Press 1995
  ISBN 1-55558-145-5
- Dileep P. Bhandarkar
  Alpha implementations and architecture
  Digital Press 1996, ISBN 1-55558-130-7

HP PA-RISC

- Gerry Kane
  PA-RISC 2.0 Architecture
  Prentice-Hall PTR 1996
  ISBN 0-13-182734-0

IBM Power, PowerPC

- Shlomo Weiss and James E. Smith
  Power and PowerPC: Principles, Architecture, Implementation
  Morgan-Kaufmann 1994
  ISBN 1-55860-279-8

IBM Power, PowerPC (continued)

- IBM Corporation
  PowerPC Architecture: A Specification for a New Family of RISC Processors
  Morgan-Kaufmann 1994
  ISBN 1-55860-316-6

IBM S/390 G5/G6

- G5 announced 7-May-1998
- Tim Slegel et al
  IBM's S/390 G5 microprocessor design
  IEEE Micro 19(2):12--23 (1999)
- E. M. Schwarz and C. A. Krygowski
  The S/390 G5 floating-point unit
  IBM J. Res. Dev. 43(5/6):707-721 (1999).

Inmos transputer

- Ian Graham and Tim King
  The transputer handbook
  Prentice-Hall International 1990
  ISBN 0-13-929134-2

Intel i860

- *i860 64-bit Microprocessor Hardware Reference Manual*
  Intel 1990
  ISBN 1-55512-106-3
- *i860 64-bit Microprocessor Family Programmer's Reference Manual*
  Intel 1991
  ISBN 1-55512-135-7

Intel i960

- Glenford J. Myers and David L. Budde
  Intel 80960 Microprocessor Architecture
  Wiley 1988
  ISBN 0-471-61857-8

Intel x86

- Walter A. Triebel and Avtar Singh
  The 8088 and 8086 Microprocessors: ...
  Prentice-Hall 2000
  ISBN 0-13-010560-0
- John F. Palmer and Stephen P. Morse
  The 8087 Primer
  Wiley 1984
  ISBN 0-471-87569-4

Intel x86 (continued)

- AMD, Cyrix, IBM, and Nexgen reverse-engineer and manufacture x86 workalikes
- HP/Intel IA-64 and Transmeta Crusoe also implement x86
- Several x86 software emulators or translators (SoftPC, SunPC, Sun Wabi, DEC FX!32, ...)

Java Virtual Machine (JVM)

- Tim Lindholm and Frank Yellin
  The Java Virtual Machine Specification
  Second edition
  Addison-Wesley 1999
  ISBN 0-201-43294-3

Java Virtual Machine (JVM) (continued)

- 32-bit software architecture (initially)
- Strictly-controlled evaluation order in Java hinders optimization
- Strict 32-bit and 64-bit ANSI/IEEE 754 floating-point subset (round-to-nearest only: hard to do interval arithmetic, and extended intermediate precision is forbidden!)

Java Virtual Machine (JVM) (continued)

- No pointers, and all array accesses are bounds-checked *each* time
- Multidimensional array layout problems
- 16-bit Unicode characters, but Unicode 3.0 needs at least 21 bits!

Motorola 68000

- *MC68000 16/32-bit Microprocessor Programmer's Reference Manual*
  Motorola 1984
  ISBN 0-13-541400-8

- *MC68881/MC68882 Floating-Point Coprocessor User's Manual*
  Prentice-Hall 1987
  ISBN 0-13-566936-7

Motorola 88000

- *Motorola 88100 RISC Microprocessor User's Manual*
  Motorola 1989
  ISBN 0-13-567090-X

SGI/MIPS RxOO (*x = 20..120*)

- Gerry Kane and Joe Heinrich
  MIPS RISC Architecture
  Prentice-Hall 1992 (2nd edition online)
  ISBN 0-13-590472-2
- Paul Chow
  The MIPS-X RISC Microprocessor
  Kluwer 1989
  ISBN 0-7923-9045-8

Sun MAJC

- Marc Tremblay et al
  The MAJC Architecture: A synthesis of parallelism and scalability
  IEEE Micro 20(6):12–25 (2000).

Sun PicoJava

- Harlan McGhan and Mike O'Connor
  PicoJava: A direct execution engine for Java bytecode
  Computer 31(10):22--30 (1998)
- J. Michael O'Connor and Marc Tremblay
  PicoJava-I: The Java Virtual Machine in hardware
  IEEE Micro 17(2):45--53 (1997)

Sun SPARC

- David L. Weaver and Tom Germond
  The SPARC Architecture Manual --- Version 9
  Prentice-Hall PTR 1994
  ISBN 0-13-099227-5

Transmeta Crusoe

David R. Ditzel
Transmeta's Crusoe: A Low-Power x86-Compatible Microprocessor Built with Software
Cool Chips 3: 2000
Hot Chips 12: 2000

Instruction set sizes

Cray 1	128	Intel i960	184
Cray X-MP	246	MIPS R2000	93
DEC Alpha	451	MIPS R4000	154
HP PA-RISC	165	MIPS64	661
IBM 801	ca. 118	Motorola 88000	51
IBM Power	184	Sun SPARC	72
Intel i860	161		

Memory model and size

- *Segmented* (Intel x86 (64KB), IBM 370-XA (2GB)) vs *uniform linear* address space
- 8-bit (256B), 16-bit (64KB), 32-bit (1GB, 2GB, or 4GB), 64-bit (18PB = 1.8e+19B)
- IBM S/360 April 1964: $1/B ($5.55/B in 2000 dollars) vs IBM PC RAM in early 2000: $0.60/MB

Memory model and size

- At 100MB/sec, 32-bit space fills in 45 sec, but 64-bit space fills in > 5000 years!

Registers, cache, and RAM

- Access time: register == 1 cycle, cache == 2–16, RAM == 10--100, network > 50K, disk > 1M
- Up to 3 levels of cache (level 1 (sometimes 2) on chip, others on system board in expensive memory technology)
- Write-through cache vs write-back vs ...

Registers, cache, and RAM

- *Direct-mapped* cache vs *n-way set-associative* vs *fully-associative*
- *Combined* vs *separate* instruction and data
- Multiprocessor cache-coherence problem
- User control of cache reuse through clever programming (USI TR 3, November 1990)
- Turn off cache for some types of jobs

Imbench memory performance on Sun SPARC models

CPU	MHz	Register	L1	L2	RAM
10/412	40	1	2.0	15.7	16.2
20/512	50	1	2.0	8.2	55.8
LX	50	1	2.0	9.8	10.2
US170	167	1	2.0	8.0	47.3
E250	300	1	1.8	9.9	79.5
E5500	400	1	1.6	10.0	102.8

Register sets

- Set size: 1, 4, 8, 16, 32, 64, 128, 256, ...
- General, or separate floating-point, registers
- Instruction-dedicated (Intel x86) vs general purpose
- Vector vs scalar registers
- Register-use masks to reduce CALL/RETURN overhead

Register sets

- Overlapping register windows (Sun SPARC)
- Multiple register sets on chip to reduce context switch time
- Register renaming on chip
- Stack (Intel x86 floating-point) vs random addressing (every sensible architecture)

Visual instruction sets (integer fixed-point, and floating-point)

- 1995: HP *PA-7100LC* on PA-RISC
- 1996: Compaq/DEC Alpha *MVI (Motion Video Instructions)* (21164PC and 21264)
- 1996: HP MAX-2 on PA-RISC
- 1996: Intel *MMX* (MultiMedia eXtension, or Matrix Math eXtension) on x86
- 1996: Sun *VIS* (Visual Instruction Set) on UltraSPARC

Visual instruction sets (integer fixed-point, and floating-point)

- 1997: Cyrix M2 (MMX compatible)
- 1997: SGI/MIPS *MDMX* (MIPS Digital Media eXtension: MaDMaX) *+ MIPS-V*
- 1998: AMD *3DNow!* (also supports MMX)
- 1998: Motorola *AltiVec* (for PowerPC)

Visual instruction set compiler support

- Possibly useful for 32-bit floating-point, although only subsets implemented
- Except for Motorola and SGI, no other vendor provides high-level language compiler support for use of these instructions, only assembly code
- *Icc* compiler on Intel x86 supports MMX

Visual instruction set cost and benefit

- Cyrix M2: added 1% to die size (20K transistors)
- Cyrix M2: Inverse Discrete Cosine Transform (IDCT) (core of MPEG decoding): x86: 220 cycles vs MMX: 31 cycles

Floating-point architectures

- IBM S/360, S/370, S/380, S/390 hexadecimal normalization, 32-bit, 64-bit, 128-bit
- CDC 60-bit, and in software, 120-bit
- Cray 64-bit, and in software, 128-bit
- DEC PDP-10 and Unisys 36-bit and 72-bit
- DEC PDP-11 and VAX binary normalization 32-bit, 64-bit, and on VAX in software, 128-bit

Floating-point architectures (continued)

- ANSI/IEEE 754-1985 (almost all CPUs since 1980) 32-bit, 64-bit, 80-bit (Intel and Motorola 68K), 128-bit (software, except IBM S/390 G5 and HAL SPARC64-GP)
- Wider format means wider exponent range (unlike IBM S/360 and DEC PDP-11, VAX)
- *Nonstop* computing model

Floating-point architectures (continued)

- *Gradual underflow* (denormalized numbers)
- *NaN* (Not-a-Number) and *Infinity*

Floating-point programming issues

- NaN .NE. NaN has implications for a *lot* of scientific software
- Unordered comparisons conflict with ancient Fortran 3-way branch:
  IF (expr) n1, n2, n3

Floating-point programming issues (continued)

- Unordered comparisons also conflict with modern 2-way branch:

```
IF (expr < 0.0)
  { then-part }
ELSE
  { else-part }
```


Floating-point programming issues (continued)

- NaNs can produce infinite loops (e.g., in EISPACK and LINPACK):

```
tol = 1.0
WHILE ((1.0 + tol) .NE. 1.0)
{
  ...do some work to reduce tol...
  tol = newtol
}
```


Floating-point architectures (resumed)

- Rounding modes (to *+Infinity*, *-Infinity*, *zero*, *nearest*), and support for interval arithmetic (Sun Workshop 6 compilers for Fortran 90 , 95, and C++)
- Instruction repertoire: basic +, -, *, / and type conversion, or also elementary functions (arc, hyperbolic, log, power, sqrt, trig)

Floating-point architectures (continued)

- Divide and square root sometimes replaced by reciprocal approximation and Newton-Raphson iteration (some models of Cray, and HP/Intel IA-64), and result may be incorrectly rounded
- Intel x86 and IA-64 do not produce correct rounding (1 bit error on Pentium, 0.6 bit error on IA-64: helped by 80/82-bit format)

Floating-point architectures (continued)

- Some vendors do not default to correct nonstop ANSI/IEEE 754 behavior unless asked by additional compilation switches (e.g., Compaq/DEC Fortran *-fpe3* or *-fpe4*, and C/C++ *-ieee*)
- CDC 6600, HP PA-RISC 1.0, and Sun SPARC v7 lack integer multiply and divide

Floating-point architectures (continued)

- Entirely in *hardware* (Compaq/DEC Alpha 21264, IBM S/390 and Power, Intel x86 and IA-64, Motorola 68K and 88K, Sun UltraSPARC III (check??))
- Partly in *hardware*, plus *software* for complicated parts (Compaq/DEC Alpha, HP PA-RISC, SGI/MIPS, Sun SPARC); software may be set at compile time, or at run time

Floating-point architectures (continued)

- Entirely in *software* (Standard Apple Numeric Environment (SANE), Java Virtual Machine)

Floating-point architectures (continued)

- Some implement only partial ANSI/IEEE 754 (e.g., no denormals, NaN, or Infinity, or only a subset of rounding modes, or no runtime setting of rounding modes)
- Some vendors offer old-style arithmetic as well as ANSI/IEEE 754 (Compaq/DEC Alpha, Convex, IBM S/390)

Floating-point architectures (continued)

- Multiply-add/subtract: form exact double-width product, then add with single rounding (HP PA-RISC, HP/Intel IA-64, IBM Power (first, in 1990), SGI/MIPS MIPS-IV (R5000, R8000, R10000, R12000), but *not* Compaq/DEC Alpha, Intel x86, Java Virtual Machine, or Sun SPARC)
- Get 2x performance boost for free!

Floating-point architectures (continued)

- Good compilers offer user control over whether multiply-add is used or not
- MIPS64 ISA has integer multiply-add too!
- MIPS-V has paired-single operations, for 2x boost
- Preloading, and multiword load/store

Floating-point architectures (continued)

- *Scalar* vs *MLIW* (visual instruction sets and Intel IA-64) vs *vector* (8 -- 256 elements)
- Quadruple-precision (only IBM S/390's hexadecimal and ANSI/IEEE 754 formats, and Hal SPARC64-GP, are in hardware; all others are in software, but IBM RS/6000 and SGI terribly brain damaged and buggy)

AMD-K7 floating-point instruction latencies

Operation	Latency	Throughput
fadd	4	1
fmul	4	1
fdiv	16/20/24	13/17/21
fsqrt	19/27/35	16/24/32
fcom	2	1

AMD-K7 MMX and 3DNow! instruction latencies

Operation	Latency	Throughput
MMX ALU	2	1
MMX Mul	3	1
3DNow! pfadd	4	1
3DNow! pfmul	4	1
3DNow! pfrcp	3	1

DEC Alpha 21164 floating-point instruction cycles

Instruction	Cycles (32-bit)	Cycles (64-bit)
add	4	4
mul	4	4
div	15—31	22—60

DEC Alpha 21264 floating-point instruction cycles

Instruction	Cycles (32-bit)	Cycles (64-bit)
fadd	4—6	4—6
fmul	4—6	4—6
fdiv	9—12	12—15
fsqrt	15—18	30—33

MIPS R4400 integer instruction cycles

Instruction	Total Cycles	Overlap
mul	12	10
div	75	0
dmul	20	18
ddiv	139	0

MIPS R4400 floating-point instruction cycles

Instruction	Cycles (32-bit)	Cycles (64-bit)
mov	1	1
add	4	4
mul	7	8
div	23	36
sqrt	2—54	2—112

MIPS R10000 floating-point instruction cycles

Instruction	Cycles (32-bit)	Cycles (64-bit)
fld/fst	3	3
fadd	2	2
fmul	2	2
fdiv	12	19
fsqrt	18	33

Sun UltraSPARC IIi floating-point instruction cycles

Instruction	Cycles (32-bit)	Cycles (64-bit)
fcmp	1	1
fadd	3	3
fmul	3	3
fdiv/fsqrt	12	22
padd et al	1	1
pmul et al	3	3

Sun UltraSPARC III floating-point instruction cycles

Instruction	Cycles (32-bit)	Cycles (64-bit)
fadd	4	4
fmul	4	4
fdiv	17	20
fsqrt	24	24

The End