Microprocessor Overview

Prehistory: 1906--1947
- 1906: Lee DeForest 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)

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

- 1980: IBM adopts Intel 8086 for IBM PC
- 1980: Patterson et al at Berkeley RISC-I, -II
- 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)

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: 2000–date

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

Even more history: 1986–1999

- 1986: IBM RT
- 1986: MIPS-I (R2000) commercialized
- 1987: Sun SPARC v8
- 1988: Motorola 68000
- 1989: Intel i860
- 1990: IBM Power (RS/6000)
- 199x: 4-bit micro reach 1B/year

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 S1/Cu chip
- 1999: Motorola ships two billionth MC68HC05 8-bit 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 0-201-18557-8

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

Bibliographic resources
- TeX User Group archive:
  [http://www.tug.org/tex/](http://www.tug.org/tex/)
- BibNet Project archive:
- Software:

Architecture books
1985--1997
- John L. Hennessy and David A. Patterson
  *Computer Architecture*
  ISBN 1-55860-069-8, 1-55860-329-8
- *Computer Organization and Design*
  Morgan Kaufmann (1994, 1997)

Architecture journals, conferences, and standards
- *Hot Chips Symposium* (1989–date)
- *Cool Chips Symposium* (1996–date)
- *IEEE Symposium on Computer Arithmetic (ARITH)* (1972–date)
- Michael Storer's *Microprocessor Report*
- ANSI/IEEE 754-1985 Standard for Binary Floating-Point Arithmetic (Draft in 2010)

Other resources
- Nelson H. F. Beebe
  *The Impact of Memory and Architecture on Computer Performance* (1994)
- Mathematical software benchmarks:
  [http://www.math.utah.edu/~beebe/benchmarks](http://www.math.utah.edu/~beebe/benchmarks)
Microprocessor overview

- Acorn RISC Machine (ARM)
- Compaq/DEC Alpha
- HP PA-RISC 1.0-1.2
- IBM Power, PowerPC
- IBM S/390 G5
- Innoset transputer
- Intel 8086,1860, x86, and with HP, IA-64
- Java Virtual Machine (JVM)
- Motorola 68K, 88K
- SGI/MIPS R1000
- Sun Magic, Pico Java, SPARC
- Transmeta Crusoe

Compaq/DEC Alpha

- Richard L. Sites and Richard L. Witek
  *Alpha ARP architecture reference manual*
  Digital Press 1995

- Dilip P. Bhandarkar
  *Alpha implementations and architecture*

Acorn (Advanced) RISC Machine

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

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
- E.M. Schwarz and C. A. Krygowski
  The S/390 G5 Floating-point unit

Inmos transputer
- Ian Graham and Tim King
  The transputer handbook
  Prentice-Hall International 1990

Intel i860
- i860 64-bit Microprocessor Hardware
  Reference Manual
  Intel 1990
  ISBN 1-555-12-106-3
- i860 64-bit Microprocessor Family
  Intel 1991
  ISBN 1-555-12-35-7

Intel i960
- Glenford J. Myers and David L. Budde
  Intel 80086 Microprocessor Architecture
  Wiley 1988

Intel x86
- Walter A. Triebel and Artar 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 (SoPC, SunPC, Sun Wabi, DEC FX32, ...)

- 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 (SoPC, SunPC, Sun Wabi, DEC FX32, ...)

- AMD, Cyrix, IBM, and Nexgen reverse-
  engineer and manufacture x86 workalikes
- HP/Intel IA-64 and Transmeta Crusoe also implement x86
**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)**

- 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**

  Motorola 1984
  ISBN 0-13-541400-8
- MC68000/MC68882 Floating-Point Coprocessor User’s Manual
  Prentice-Hall 1987

**SGI/MIPS Rx00 (x = 20.120)**

- Gerry Kane and Joe Heinrich
  *MIPS RISC Architecture*
  Prentice-Hall 1992 (2nd edition online)
- 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  

Sun SPARC

David I. Weaver and Tom Germond  
The SPARC Architecture Manual — Version 9  
Prentice-Hall PTR 1994  

Instruction set sizes

<table>
<thead>
<tr>
<th>Processor</th>
<th>Sun MAJC</th>
<th>Sun SPARC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cray 1</td>
<td>128</td>
<td>184</td>
</tr>
<tr>
<td>Cray X-MP</td>
<td>246</td>
<td>93</td>
</tr>
<tr>
<td>DEC Alpha</td>
<td>451</td>
<td>154</td>
</tr>
<tr>
<td>HP PA-RISC 165</td>
<td>661</td>
<td></td>
</tr>
<tr>
<td>IBM 801</td>
<td></td>
<td>51</td>
</tr>
<tr>
<td>IBM Power</td>
<td>184</td>
<td></td>
</tr>
<tr>
<td>Intel i860</td>
<td>161</td>
<td></td>
</tr>
</tbody>
</table>

Sun PicoJava

Harlan McMillan and Mike O’Connor  
PicoJava: A direct execution engine for Java bytecode  

J. Michael O’Connor and Marc Tremblay  
PicoJava: The Java Virtual Machine in hardware  

Transmeta Crusoe

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

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

- 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

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

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 MMX (Motion Video Instructions) (21164PC and 21264)
- 1996: HP MAX-2 on PA-RISC
- 1996: Intel "M64x (MultiMedia eXtension, or Matrix Math eXtension) on x86
- 1996: Sun "V5 (Virtual Instruction Set) on UltraSPARC

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

Floating-point architectures

- IBM S/360, S/370, S/380, S/390
- hexagonal normalization, 32-bit, 64-bit, 128-bit
- CDC 60-bit, 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

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

- 1997: Cyrix M2 (MMX compatible)
- 1997: SGI/MIPS IMDX (MIPS Digital Media extension: Mediala) + MIPS-IV
- 1998: AMD 3DNow! (also supports MMX)
- 1998: Motorola Altivec (for PowerPC)

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 (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 (continued)

- NaN, INF. NaN has implications for a lot of scientific software
- Unordered comparisons conflict with ancient Fortran 3-way branch:
  \[
  \text{IF}\ (\text{expr}) \text{THEN n1, n2, n3}
  \]

Floating-point architectures (resumed)

- Rounding modes (-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 some elementary functions (exp, hyperbolic, log, power, sqrt, trig)

Floating-point programming issues (continued)

- NaNs can produce infinite loops (e.g., in EISPACK and LINPACK):
  \[
  \text{tol} = 1.0
  \]
  \[
  \text{WHILE} \ (1.0 + \text{tol}) \text{NE} 1.0
  \]
  \[
  \text{...do some work to reduce tol...}
  \]
  \[
  \text{tol} = \text{new tol}
  \]

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, 0.5 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++ -fpe4).
- CDC 6000, HP PA-RISC 1.0, and Sun SPARC v7 lack integer multiply and divide.

Floating-point architectures (continued)

- Entirely in software (Standard Apple Numeric Environment (SAME), Java Virtual Machine).

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), 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)

- Entirely in hardware (Compaq/DEC Alpha 21264, IBM S/390 and Power, Intel x86 and IA-64, Motorola 68K and 86K, Sun UltraSPARC II (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)

- 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)

- 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 multivord load/store.
Floating-point architectures (continued)

- **Scalar vs ML/W** (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 HP SPARC64-CP, are in hardware; all others are in software, but IBM RS/6000 and SGI terribly brain damaged and buggy)

AMD-K7 MMX and 3DNow! instruction latencies

<table>
<thead>
<tr>
<th>Operation</th>
<th>Latency</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMX ALU</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>MMX Mul</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3DNow! pfadd</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>3DNow! pfmul</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>3DNow! pfrcp</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

DEC Alpha 21264 floating-point instruction cycles

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Cycles (32-bit)</th>
<th>Cycles (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>fadd</td>
<td>4–6</td>
<td>4–6</td>
</tr>
<tr>
<td>fmul</td>
<td>4–6</td>
<td>4–6</td>
</tr>
<tr>
<td>fdiv</td>
<td>9–12</td>
<td>12–15</td>
</tr>
<tr>
<td>fsqrt</td>
<td>15–18</td>
<td>30–33</td>
</tr>
</tbody>
</table>

AMD-K7 floating-point instruction latencies

<table>
<thead>
<tr>
<th>Operation</th>
<th>Latency</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>fadd</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>fmul</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>fdiv</td>
<td>16/20/24</td>
<td>13/17/21</td>
</tr>
<tr>
<td>fsqrt</td>
<td>19/27/35</td>
<td>16/24/32</td>
</tr>
<tr>
<td>fcom</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

DEC Alpha 21164 floating-point instruction cycles

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Cycles (32-bit)</th>
<th>Cycles (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>add</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>mul</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>div</td>
<td>15–31</td>
<td>22–60</td>
</tr>
</tbody>
</table>

MIPS R4400 integer instruction cycles

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Total Cycles</th>
<th>Overlap</th>
</tr>
</thead>
<tbody>
<tr>
<td>div</td>
<td>75</td>
<td>0</td>
</tr>
<tr>
<td>dmul</td>
<td>20</td>
<td>18</td>
</tr>
<tr>
<td>ddiv</td>
<td>139</td>
<td>0</td>
</tr>
</tbody>
</table>
MIPS R4400 floating-point instruction cycles

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Cycles (32-bit)</th>
<th>Cycles (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>mov</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>add</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>mul</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>div</td>
<td>23</td>
<td>36</td>
</tr>
<tr>
<td>sqrt</td>
<td>2—54</td>
<td>2—112</td>
</tr>
</tbody>
</table>

MIPS R10000 floating-point instruction cycles

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Cycles (32-bit)</th>
<th>Cycles (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ld/st</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>fadd</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>fmul</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>fdiv</td>
<td>12</td>
<td>19</td>
</tr>
<tr>
<td>fsqrt</td>
<td>18</td>
<td>33</td>
</tr>
</tbody>
</table>

Sun UltraSPARC II floating-point instruction cycles

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Cycles (32-bit)</th>
<th>Cycles (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>fcmp</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>fadd</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>fmul</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>fdiv/fsqrt</td>
<td>12</td>
<td>22</td>
</tr>
<tr>
<td>padd et al</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>pmul et al</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

Sun UltraSPARC III floating-point instruction cycles

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Cycles (32-bit)</th>
<th>Cycles (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>fadd</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>fmul</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>fdiv</td>
<td>17</td>
<td>20</td>
</tr>
<tr>
<td>fsqrt</td>
<td>24</td>
<td>24</td>
</tr>
</tbody>
</table>

The End