Computer Organization &
Assembly Language Programming

CSE 2312
Lecture 2 Introduction of Computers

Junzhou Huang, Ph.D.
Department of Computer Science and Engineering
Administration Reviewing

• Course CSE2312
  – What: Computer Organization & Assembly Language Programming
  – When: Mon. & Wed. 3:00 ~ 4:20pm
  – Where: SH 332
  – Who: Junzhou Huang (Office ERB 650) jzhuang@uta.edu
  – Office Hour: Mon. & Wed. 1:00 ~ 3:00pm and/or appointments
  – Homepage: [http://ranger.uta.edu/~huang/teaching/CSE2312/](http://ranger.uta.edu/~huang/teaching/CSE2312/)
  (You’re required to check this page regularly)

• Lecturer
  – PhD in CS from Rutgers, the State University of New Jersey
  – Research areas: medical imaging (hardware and software design), machine learning, computer vision, image and signal processing

• GTA
  – Yeqing Li (Office ERB 101), yeqing.li@mavs.uta.edu
  – Office hours: Mon. & Wed. 10:00am ~ 12:00pm and/or appointments
Today: Computer Family and Evolution

• Chapter 1 in the Tanenbaum’s Textbook
  – You have read it, right?

• Acknowledgments
  – The slides today are mainly based on slides provided by Andrew S. Tanenbaum to accompany his textbook
  – Some material available on the web (via google search, wikipedia)
  – Additional materials and modifications are copyrighted by Junzhou Huang
Basic Concepts

• **Program**
  – A sequence of instructions describing how to perform a certain task

• **Architecture**
  – The set of data types, operations, and features of each level
  – **Computer Architecture** is to study how to design parts of a computer system that are visible to the programmers

• **Machine Language**
  – Formed by a computer’s primitive instructions
  – Enable people to communicate with the computer
  – Should be simple due to the complexity and costs of electronics

• **Gap in Computer Design**
  – Convenient for people (preferred natural language) or for computer (preferred binary)? How to balance?
How to bridge gap?

• Idea
  – Designing the L1 language including a new set of instructions that is more convenient for people to use than those in built-in machine instructions (L0 language)

• Solution: Translation
  – Executing a program written in L1 is first replace each instruction in it by an equivalent sequence of instructions in L0
  – The computer execute the new L0 program instead of the old L1 program

• Solution: Interpretation
  – Write a program in L0 to take programs in L1 as input data
  – Examine each instruction in turn and execute the equivalent sequence of L0 instruction directly
Translation and Interpretation

• **Similarity**
  – In both of them, the computer carried out instructions in L1 by executing equivalent sequences of instructions in L0

• **Dissimilarity**
  – In translation, the entire L1 program is converted to a L0 program. Then the new L0 program is loaded into the memory and executed. During Execution, the new L0 program is running and in control of computer
  – In interpretation, after each L1 instruction is examined and decoded, it is carried out immediately. The interpreter is in control of computer. In this case, L1 program is just data.
Languages, Levels, Virtual Machines

Level n
Virtual machine M_n, with machine language L_n

Level 3
Virtual machine M_3, with machine language L_3

Level 2
Virtual machine M_2, with machine language L_2

Level 1
Virtual machine M_1, with machine language L_1

Level 0
Actual computer M_0, with machine language L_0

Programs in L_n are either interpreted by an interpreter running on a lower machine, or are translated to the machine language of a lower machine.

Programs in L_2 are either interpreted by interpreters running on M_1 or M_0, or are translated to L_1 or L_0.

Programs in L_1 are either interpreted by an interpreter running on M_0, or are translated to L_0.

Programs in L_0 can be directly executed by the electronic circuits.

A multilevel machine
Contemporary Multilevel Machines

Level 5  Problem-oriented language level
         Translation (compiler)

Level 4  Assembly language level
         Translation (assembler)

Level 3  Operating system machine level
         Partial interpretation (operating system)

Level 2  Instruction set architecture level
         Interpretation (microprogram) or direct execution

Level 1  Microarchitecture level
         Hardware

Level 0  Digital logic level

A six-level computer. The support method for each level is indicated below it.
Evolution of Multilevel Machines

- **Invention of microprogramming**
  - 1st digit computer in 1940 only had ISA level and digital logic level
  - It made the digital logic level’ circuit very complicate, unreliable, difficult to understand and build
  - Microprogramming is designed (Wilkes, 1951) as unchangeable interpreter to execute the ISA-level programs by interpretation
  - This enables fewer electronic circuits and hence enhance the reliability

- **Invention of operating system**
  - Earlier computers were open-shop, which meant the programmers had to operate the computer by themselves
  - First designed in 1960 to reduce the amount of wasting time by automating the operator’s job
Operating System Tasks

- **Steps**
  - Put the big green deck labeled FORTRAN complier into the card reader, start
  - Put his/her Fortran program into the card reader. Read it two times.
  - The translation is completed and check if there exist errors. If no, the machine language is punched out into the cards
  - Put the translated machine language cards and subroutine library deck in
  - The program began executing
    Programmers are checking problem while CPU is idle.

- **Sample job for the FMS operating system (IBM 709)**
Evolution of Multilevel Machines

• **Migration of functionality to microcode**
  – Add new instructions (hardware) by extending microprogram
  – The added ones are not essentially but slightly faster than a sequence of existing instructions
  – Speeding up the particular activity

• **Elimination of microprogramming**
  – Computer design had come full circle
  – The microprogramming is too fat. The computer become slower and slower
  – Eliminating the microprogram
  – Vastly reducing the instruction set
  – Having the remaining instructions be directly executed by hardware
Milestones in Computer Architecture (1)

- Some milestones in the development of the modern digital computer

<table>
<thead>
<tr>
<th>Year</th>
<th>Name</th>
<th>Made by</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1834</td>
<td>Analytical Engine</td>
<td>Babbage</td>
<td>First attempt to build a digital computer</td>
</tr>
<tr>
<td>1936</td>
<td>Z1</td>
<td>Zuse</td>
<td>First working relay calculating machine</td>
</tr>
<tr>
<td>1943</td>
<td>COLOSSUS</td>
<td>British gov’t</td>
<td>First electronic computer</td>
</tr>
<tr>
<td>1944</td>
<td>Mark I</td>
<td>Aiken</td>
<td>First American general-purpose computer</td>
</tr>
<tr>
<td>1946</td>
<td>ENIAC I</td>
<td>Eckert/Mauchley</td>
<td>Modern computer history starts here</td>
</tr>
<tr>
<td>1949</td>
<td>EDSAC</td>
<td>Wilkes</td>
<td>First stored-program computer</td>
</tr>
<tr>
<td>1951</td>
<td>Whirlwind I</td>
<td>M.I.T.</td>
<td>First real-time computer</td>
</tr>
<tr>
<td>1952</td>
<td>IAS</td>
<td>Von Neumann</td>
<td>Most current machines use this design</td>
</tr>
<tr>
<td>1960</td>
<td>PDP-1</td>
<td>DEC</td>
<td>First minicomputer (50 sold)</td>
</tr>
<tr>
<td>1961</td>
<td>1401</td>
<td>IBM</td>
<td>Enormously popular small business machine</td>
</tr>
<tr>
<td>1962</td>
<td>7094</td>
<td>IBM</td>
<td>Dominated scientific computing in the early 1960s</td>
</tr>
<tr>
<td>1963</td>
<td>B5000</td>
<td>Burroughs</td>
<td>First machine designed for a high-level language</td>
</tr>
<tr>
<td>1964</td>
<td>360</td>
<td>IBM</td>
<td>First product line designed as a family</td>
</tr>
</tbody>
</table>
Milestones in Computer Architecture (2)

- Some milestones in the development of the modern digital computer

<table>
<thead>
<tr>
<th>Year</th>
<th>Name</th>
<th>Made by</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1965</td>
<td>PDP-8</td>
<td>DEC</td>
<td>First mass-market minicomputer (50,000 sold)</td>
</tr>
<tr>
<td>1970</td>
<td>PDP-11</td>
<td>DEC</td>
<td>Dominated minicomputers in the 1970s</td>
</tr>
<tr>
<td>1974</td>
<td>8080</td>
<td>Intel</td>
<td>First general-purpose 8-bit computer on a chip</td>
</tr>
<tr>
<td>1974</td>
<td>CRAY-1</td>
<td>Cray</td>
<td>First vector supercomputer</td>
</tr>
<tr>
<td>1978</td>
<td>VAX</td>
<td>DEC</td>
<td>First 32-bit superminicomputer</td>
</tr>
<tr>
<td>1981</td>
<td>IBM PC</td>
<td>IBM</td>
<td>Started the modern personal computer era</td>
</tr>
<tr>
<td>1981</td>
<td>Osborne-1</td>
<td>Osborne</td>
<td>First portable computer</td>
</tr>
<tr>
<td>1983</td>
<td>Lisa</td>
<td>Apple</td>
<td>First personal computer with a GUI</td>
</tr>
<tr>
<td>1985</td>
<td>386</td>
<td>Intel</td>
<td>First 32-bit ancestor of the Pentium line</td>
</tr>
<tr>
<td>1985</td>
<td>MIPS</td>
<td>MIPS</td>
<td>First commercial RISC machine</td>
</tr>
<tr>
<td>1987</td>
<td>SPARC</td>
<td>Sun</td>
<td>First SPARC-based RISC workstation</td>
</tr>
<tr>
<td>1990</td>
<td>RS6000</td>
<td>IBM</td>
<td>First superscalar machine</td>
</tr>
<tr>
<td>1992</td>
<td>Alpha</td>
<td>DEC</td>
<td>First 64-bit personal computer</td>
</tr>
<tr>
<td>1993</td>
<td>Newton</td>
<td>Apple</td>
<td>First palmtop computer</td>
</tr>
</tbody>
</table>
Computer Generations

• **Zeroth Generation**
  – Mechanical Computers (1642 – 1945)

• **First Generation**
  – Vacuum Tubes (1945 – 1955)

• **Second Generation**
  – Transistors (1955 – 1965)

• **Third Generation**

• **Fourth Generation**
  – Very Large Scale Integration (1980 – ?)
Recalling: Von Neumann Architecture

- Model of a computer that used stores programs
  - Both Data and Program stored in memory
  - Allows the computer to be “Re-programmed”
Original Von Neumann Machine
PDP-8 Innovation – Single Bus

The PDP-8 omnibus
# IBM 360

The initial offering of the IBM product line.

<table>
<thead>
<tr>
<th>Property</th>
<th>Model 30</th>
<th>Model 40</th>
<th>Model 50</th>
<th>Model 65</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relative performance</td>
<td>1</td>
<td>3.5</td>
<td>10</td>
<td>21</td>
</tr>
<tr>
<td>Cycle time (in billionths of a sec)</td>
<td>1000</td>
<td>625</td>
<td>500</td>
<td>250</td>
</tr>
<tr>
<td>Maximum memory (bytes)</td>
<td>65,536</td>
<td>262,144</td>
<td>262,144</td>
<td>524,288</td>
</tr>
<tr>
<td>Bytes fetched per cycle</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>Maximum number of data channels</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>
Moore Law

- **Moore Law**
  - Gordon Moore was an Inter Engineer
  - An observation about improvements in hardware
  - Predicts a 60-percent annual increase in the number of transistors that can be put on a chip.

- **Processor Speed**
  - Processors Speed 2x every 18 months
  - Number of transistors on a chip double every 18 months

- **Memory Capacity**
  - 2x every 2 years

- **Disk Capacity**
  - 2x every year
Applied to Transistors on a Chip

• Moore’s law
  – Predicts a 60-percent annual increase in the number of transistors that can be put on a chip.
  – The data points given in this figure are memory sizes, in bits.
Clock Speed
Processor Performance

The graph illustrates the improvement in processor performance over time, measured in MIPS. The plot shows a logarithmic increase in performance from 1970 to 2005, indicating a doubling time of about 1.8 years. Key processor generations marked on the graph include the 486, Pentium, Pentium II, Pentium 4, and Xeon.
Growth in Processor Performance

Figure 1.1 Growth in processor performance since the mid-1980s. This chart plots performance relative to the VAX 11/780 as measured by the SPECint benchmarks (see Section 1.8). Prior to the mid-1980s, processor performance growth was largely technology driven and averaged about 25% per year. The increase in growth to about 52% since then is attributable to more advanced architectural and organizational ideas. By 2002, this growth led to a difference in performance of about a factor of seven. Performance for floating-point-oriented calculations has increased even faster. Since 2002, the limits of power, available instruction-level parallelism, and long memory latency have slowed uniprocessor performance recently, to about 20% per year. Since SPEC has changed over the years, performance of newer machines is estimated by a scaling factor that relates the performance for two different versions of SPEC (e.g., SPEC92, SPEC95, and SPEC2000).
DRAM Capacity

TOTAL DRAM BITS SHIPPED / TOTAL CHIPS SHIPPED

4X/3Yr pace

Actual

YEAR


Mb/chip

Source: WSTS
Memory Granularity

![Graph showing the evolution of memory granularity](image_url)

Source of image at: [www.ieee.org/~08Winter&file=Isaac.xml](http://www.ieee.org/~08Winter&file=Isaac.xml)
Memory Price

Retail $50 to $100/GB
Hard Disk Drive Capacity

1985-2008
Grew from 10MB to 1,000GB

2008-2020
Forecast to grow from 1TB to 80TB

2008, 1,000 GB = 1TB
Hard Disk Drive Growth

30$ to 50 $ per 100 G , 1 TB for 100 $ to
## Power Ratings

- **Atom Processor 1.6 Ghz , 4 W .. netbook**
- **AMD Athlon 1.4 Ghz, 64 W … desktop**

<table>
<thead>
<tr>
<th>Model</th>
<th>Clock Speed</th>
<th>Power</th>
<th>Clock Speed to Power Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pentium Pentium</td>
<td>75MHz</td>
<td>8.0W</td>
<td>9.4</td>
</tr>
<tr>
<td>Pentium Pentium</td>
<td>90MHz</td>
<td>9.0W</td>
<td>10</td>
</tr>
<tr>
<td>Pentium Pentium</td>
<td>100MHz</td>
<td>10.1W</td>
<td>9.9</td>
</tr>
<tr>
<td>Pentium Pentium</td>
<td>120MHz</td>
<td>11.9W</td>
<td>10.1</td>
</tr>
<tr>
<td>Pentium Pentium</td>
<td>133MHz</td>
<td>11.2W</td>
<td>11.5</td>
</tr>
<tr>
<td>Pentium Pentium</td>
<td>150MHz</td>
<td>11.6W</td>
<td>12.9</td>
</tr>
<tr>
<td>Pentium Pentium</td>
<td>166MHz</td>
<td>14.5W</td>
<td>11.4</td>
</tr>
<tr>
<td>Pentium Pentium</td>
<td>200MHz</td>
<td>15.5W</td>
<td>12.9</td>
</tr>
</tbody>
</table>
Battery Lifetime

- Netbook 5 to 6 hours
- Laptop 3 to 6 hours
- iPAD 8 to 10 hours
- Smart Phones
- 7 to 8 hours, Standby 24 hours
## Battery Lifetime

<table>
<thead>
<tr>
<th>Compare for yourself⁴</th>
<th>HP EliteBook 6930p Notebook PC</th>
<th>Dell Latitude E6400</th>
<th>HP EliteBook 6930p Notebook PC*</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Battery Life (up to ...)</strong></td>
<td>17.25 hours</td>
<td>19 hours</td>
<td>24 hours</td>
</tr>
<tr>
<td>Starting price</td>
<td>$1,199</td>
<td>$879</td>
<td>$1,227</td>
</tr>
<tr>
<td>Included battery</td>
<td>6 cell</td>
<td>6 cell</td>
<td>6 cell</td>
</tr>
<tr>
<td>Upgrade primary battery</td>
<td>n/a</td>
<td>9 cell</td>
<td>n/a</td>
</tr>
<tr>
<td>Add on battery</td>
<td>Ultra Capacity Battery</td>
<td>12-Cell High Capacity Slice</td>
<td>Ultra Capacity Battery</td>
</tr>
<tr>
<td>Display</td>
<td>Standard WXGA</td>
<td>Not Disclosed</td>
<td>HP Illumi-Lite LED backlit display</td>
</tr>
<tr>
<td>Storage/Hard Drive</td>
<td>n/a</td>
<td>64GB Ultra Performance Solid State Drive</td>
<td>80GB Intel SSD</td>
</tr>
<tr>
<td><strong>Final Price</strong></td>
<td><strong>$1,388</strong></td>
<td><strong>$2,026</strong></td>
<td><strong>$1,696</strong></td>
</tr>
</tbody>
</table>

Source: www.hp.com
Battery Technology

Super store
Rechargeable-battery capacity
World trends, Wh/kg

Source: Avicenne  NB: Dashed lines denote forecast data

New Li technology
Li-ion/Poly
NiMH
NiCd

1970  80  90  2000  10  15