Logistics

- **Lecture**
  - Tuesdays, 13:15 - 14:45, Auditorium Maximum (Building 33)

- **Tutorial and Exercises (Begin: October 17)**
  - Wednesdays, 11:30 – 13:00, Building 33 Room 2401 (in German)
  - Thursdays, 09:45 - 11:15, Building 43 Room 4/126 (in German)
  - Thursdays, 15:00 - 16:30, Building 33 Room 2216 (in German)
  - Thursdays, 16:45 - 18:15, Building 33 Room 2116 (in English)

- **Exam (tentatively, to be confirmed)**
  - December 18, 2007 – 13:00 – 15:00

Office Hours and Contact

**Prof. Dr. Martin Hepp**
- By appointment only
- (Tuesdays 15:00 – 16:00)
- Building 36, Room 2204
- Phone
  - +49 (89) 6004-4217
  - +49 (89) 6004-4239 (A. Hoffmann)
- mhepp@computer.org

**Alexander Richter**
- Building 36, Room 2205
- Phone: +49 (89) 6004-4218
- a.richter@unibw.de

Learning Resources (2)

- **Course Web page**
  - [http://www.heppnetz.de/teaching/gwi/](http://www.heppnetz.de/teaching/gwi/)

- All slides will be put on-line after the lecture
- Some materials may be password protected
  - User: unibw
  - Password: unibw2007
- Test run: Video recording of the lecture for your exam preparation
  - Will be made available online shortly

Structure of the Lecture

- **Unit 1:** Introduction
- **Unit 2:** Central Processing Units
- **Unit 3:** Storage and Data Structures
- **Unit 4:** Input and Output Devices
- **Unit 5:** Software
- **Unit 6:** Networks, Data Interchange, and the Internet
- **Unit 7:** Design, Development, Deployment, and Operations of Information Systems
- **Unit 8:** Office Applications
- **Unit 9:** Enterprise Applications
- **Unit 10:** Supply Chain Applications and E-Business
- **Unit 11:** Management Support Systems
- **Unit 12:** Exam Review

Link to the Previous Unit

- **Last Unit:** Economic dimension of using computers for mechanizing tasks
- **Today:** Fundamental principles of computers
  - Why are computers able to execute software?
  - What is the basic structure of any computer?
Assignment from Last Week

- Get the books
- Review the slides
- Read the following paper
  [EconLib](http://www.econlib.org/library/Essays/rdPnc1.html),
  [PDF version](http://www.fee.org/pdf/books/I,%20Pencil%202006.pdf)

What's a Computer?

There are lots of intelligent devices...

... but are they computers?

A computer’s functionality is not determined at the time of design, but can be changed at any time just by loading new software.

Computer and Peripherals

CPU, Memory, Input & Output Devices

Input Devices → CPU → Output Devices

Memory

CPU: Central Processing Unit

- The CPU is the heart of any computer.
- It is controlled by an internal or external clock generator.
- The CPU can execute a small set of simple commands.
  - At each “beat” of the clock, it executes the next step
    - Fetch next command
    - Retrieve operands (values)
    - Execute command
    - Store results

Computer: Von Neumann Architecture

Memory contains data AND the program

CPU:

- Central Processing Unit (CPU)
- Control
- Arithmetic Logic Unit (ALU)
- Input/Output (I/O)
Analog vs. Digital Computers

- There are two principal ways of implementing a computer:
  - Analog: A value is represented by a respective voltage (1 = 1 volt, 2 = 2 volts, ...)
  - Digital: Values are represented by a pattern (combination) of multiple signals that can take only one of two voltage levels.

Problems with Analog Systems
- Expensive
- Lack of Reliability
- Transmission Errors

→ Dominance of Digital Computers
Digital: 1

Digital: 2

Digital: 3

Digital: 4

How would the number 5 be represented?
### Binary Numbers

<table>
<thead>
<tr>
<th>4</th>
<th>2</th>
<th>1</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Off</td>
<td>Off</td>
<td>Off</td>
<td>0</td>
</tr>
<tr>
<td>Off</td>
<td>Off</td>
<td>On</td>
<td>1</td>
</tr>
<tr>
<td>Off</td>
<td>On</td>
<td>Off</td>
<td>2</td>
</tr>
<tr>
<td>Off</td>
<td>On</td>
<td>On</td>
<td>3</td>
</tr>
<tr>
<td>On</td>
<td>Off</td>
<td>Off</td>
<td>4</td>
</tr>
<tr>
<td>On</td>
<td>Off</td>
<td>On</td>
<td>5</td>
</tr>
<tr>
<td>On</td>
<td>On</td>
<td>Off</td>
<td>6</td>
</tr>
<tr>
<td>On</td>
<td>On</td>
<td>On</td>
<td>7</td>
</tr>
</tbody>
</table>

### Binary Numbers

<table>
<thead>
<tr>
<th>4</th>
<th>2</th>
<th>1</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>7</td>
</tr>
</tbody>
</table>

### Usually 8 Wires

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>128</td>
</tr>
<tr>
<td>6</td>
<td>64</td>
</tr>
<tr>
<td>5</td>
<td>32</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

8 Bits = 1 Byte

### One byte represents any number from 0 through 255

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>128</td>
</tr>
<tr>
<td>6</td>
<td>64</td>
</tr>
<tr>
<td>5</td>
<td>32</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

### Address and Data Bus

**Address Bus:** Which memory cell is to be used?

**Control Bus:** What should the memory chip do? When is the CPU ready to take the results?

**Data Bus:** For exchanging values between the memory and the CPU

### Address Bus

- **Processor:**
  - Read
  - Write

- **Memory:**
  - Selection of the respective cell
Address and Data Bus: Example

Which memory cell is to be used?
Address Bus: 128+1*64+32+16+8+4+2+1 = 171

Which cell is to be used?
Memory

Control Bus:
What should the memory chip do?
01110010
00010000

When is the CPU ready to take the results?

Data Bus:
Which value is being exchanged?
01110010
01110010

For exchanging values between the memory and the CPU

Memory Size
• KB = 1024 Byte
• MB = 1024 * 1024 Byte
• GB = 1024 * 1024 * 1024 Byte
• TB = 1024 * 1024 * 1024 * 1024 Byte

Why does the processor know what to do?

Memory contains values and instructions

<table>
<thead>
<tr>
<th>Cell No.</th>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>120</td>
<td>ADD to Buffer</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>140</td>
<td>JUMP to …</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>cell no. 10</td>
</tr>
</tbody>
</table>
When you turn on your Computer…

Basic Input / Output System (BIOS)

Processor

READ THIS!
STORE THIS!
01110010
00010000
etc.
Read
Write

Selection of the respective cell

Memory

Decodes and executes it

Basic Input / Output System (BIOS)

Random Access Memory

http://computer.howstuffworks.com/bios.htm/printable

http://computer.howstuffworks.com/ram.htm/printable

Random Access Memory

SIMM
DIMM
SODIMM

http://computer.howstuffworks.com/ram.htm/printable
Thank you!

The slides and additional materials will be available at http://www.heppnetz.de/teaching/piel/ shortly. Don’t forget: Tutorials will not start until Oct 17!