

CMSC 313 Lecture 28

- **Final Exam: Tuesday 12/21, 10:30am - 12:30pm**
- **Semester Review**
- **Sample Final Exam**
- **SCEQ's**

Chapters of M&H

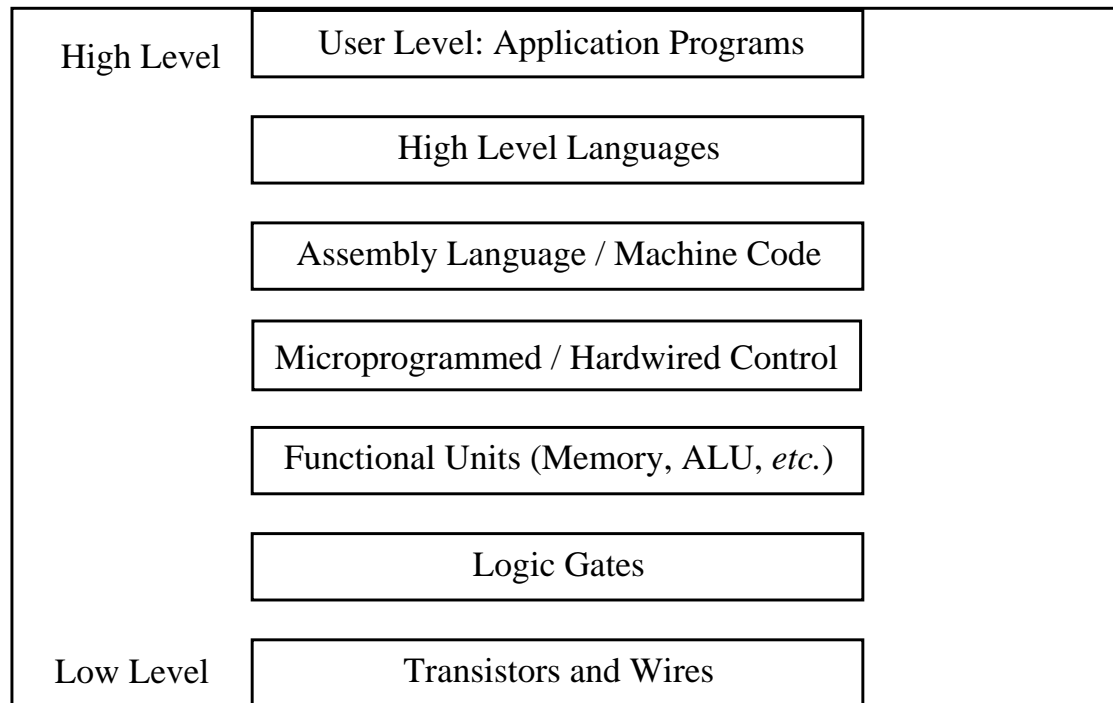
- 1. Introduction** [system bus model, levels of machine]
- 2. Data Representation**
- 3. Arithmetic** [parts]
- 4. Instruction Set Architecture** [skipped, used Intel chip]
- 5. Languages & The Machine** [compiling, assembling, linking & loading]
- 6. Datapath & Control** [skipped, covered in CMSC411]
- 7. Memory**
- 8. Input & output** [parts]
- 9. Communication** [skipped]
- 10. Trends in Computer Architecture** [parts]
 - Appendix A: Digital Logic**
 - Appendix B: Reduction of Digital Logic**

Themes

- **Levels of machines**
- **Modularity**
- **Models**

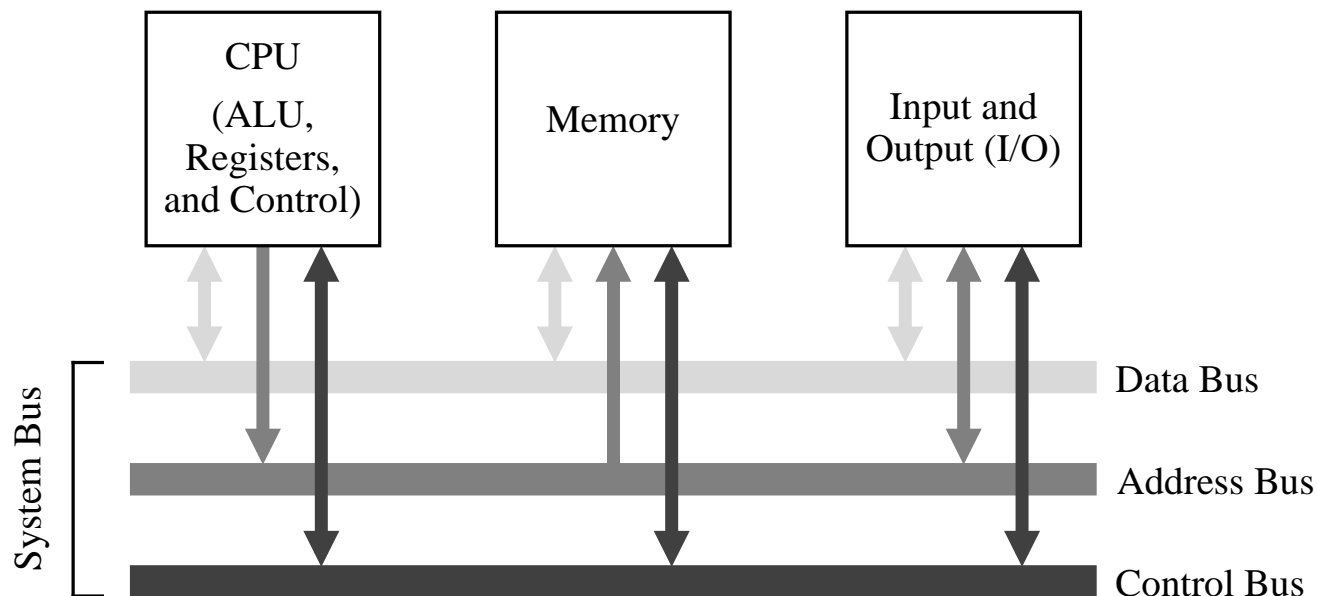
Levels of Machines

- There are a number of levels in a computer (the exact number is open to debate), from the user level down to the transistor level.
- Progressing from the top level downward, the levels become less abstract as more of the internal structure of the computer becomes visible.



The System Bus Model

- A refinement of the von Neumann model, the system bus model has a CPU (ALU and control), memory, and an input/output unit.
- Communication among components is handled by a shared pathway called the *system bus*, which is made up of the data bus, the address bus, and the control bus. There is also a power bus, and some architectures may also have a separate I/O bus.



Concentrated On

- **Basic Assembly Language Programming**
 - ◇ Did RISC-style programming on a CISC chip
- **Optimizing Digital Logic**

Glossed Over

- **Floating Point Instructions**
- **Input-Output**
- **Combinational Logic Components**
- **Programmable Logic Arrays**

What's Next

- **CMSC411 Computer Architecture**

- ◇ Design a CPU using VHDL

- **CMSC421 Operating Systems**

- ◇ Includes more on virtual memory and caching

- **CMSC431 Compiler Design**

- ◇ Write a compiler from scratch for a baby programming language

- **CMSC451 Automata Theory**

- ◇ Theorems about finite state machines, context-free grammars, ...