Welcome to CS220: Computer Organization

Mainak Chaudhuri
Indian Institute of Technology Kanpur

Sketch

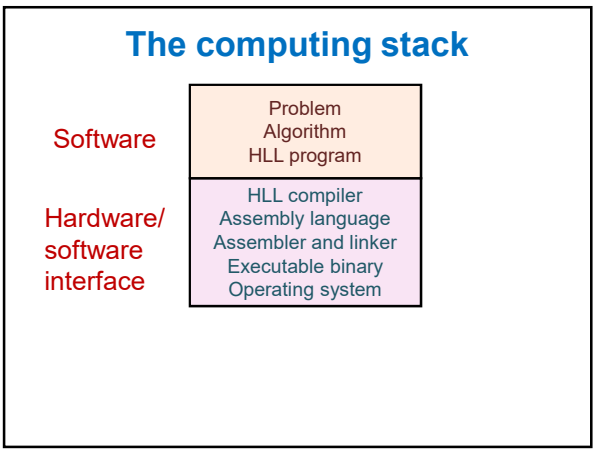
- The computing stack
- Anatomy of a computer system

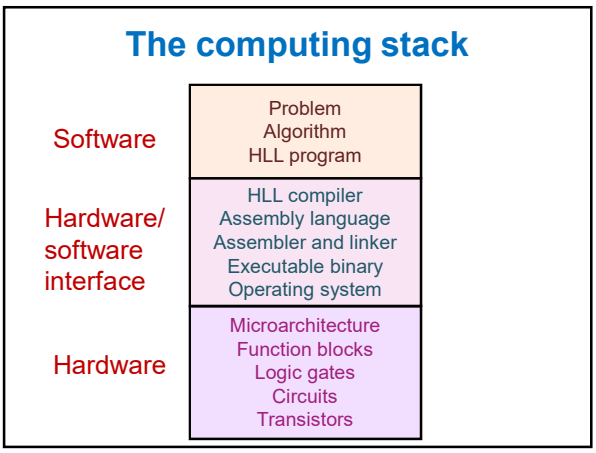
The computing stack

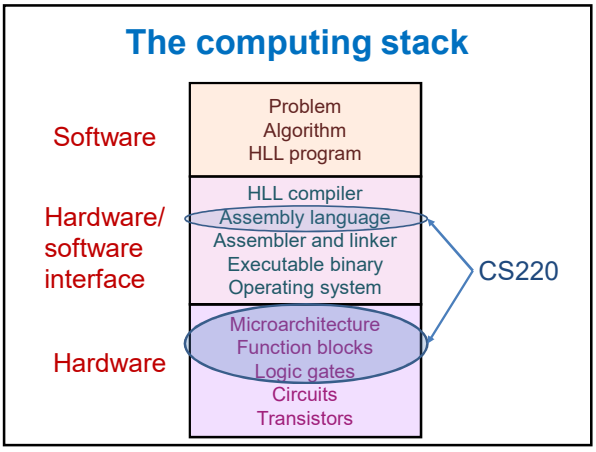
The computing stack

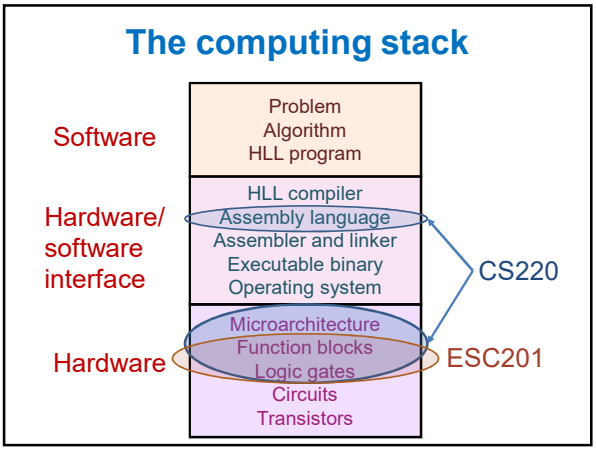
Software

Problem
Algorithm
HLL program

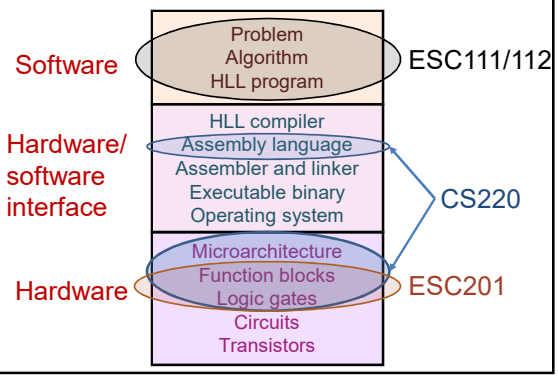

The computing stack
Software
Problem
Algorithm
HLL program
Hardware/
software
interface
HLL compiler
Assembly language
Assembler and linker
Executable binary
Operating system


The computing stack
Software
Problem
Algorithm
HLL program
Hardware/
software
interface
HLL compiler
Assembly language
Assembler and linker
Executable binary
Operating system
Hardware
Microarchitecture
Function blocks
Logic gates
Circuits
Transistors

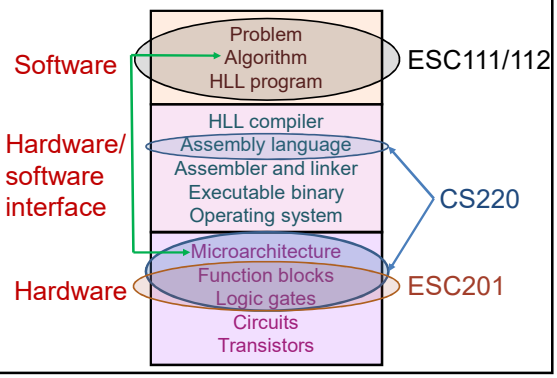

The computing stack
Software
Problem
Algorithm
HLL program
Hardware/
software
interface
HLL compiler
Assembly language
Assembler and linker
Executable binary
Operating system
Hardware
Microarchitecture
Function blocks
Logic gates
Circuits
Transistors
CS220


The computing stack
Software
Problem
Algorithm
HLL program
Hardware/
software
interface
HLL compiler
Assembly language
Assembler and linker
Executable binary
Operating system
Hardware
Microarchitecture
Function blocks
Logic gates
Circuits
Transistors
CS220
ESC201

The computing stack



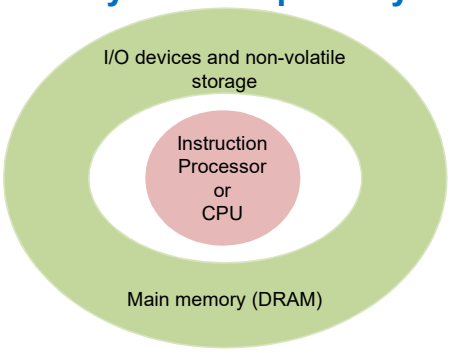
The computing stack



Logic gates to microarchitecture

- Example of a ripple-carry adder
  - Logic gates for building a function block for adding two bits with a carry input
    - Known as a full adder
  - An array of full adders used to design the microarchitecture of an adder
    - Encapsulates an algorithm for adding two n-bit numbers
    - A microarchitecture is always linked to an underlying algorithm for executing the task
- A digital computer represents one of the most complex pieces of microarchitecture
  - A complex algorithm implemented in hardware

Anatomy of a computer system



Anatomy of a computer system

- The central processing unit (CPU)
  - Also known as microprocessor
  - Dictates how a task will be done, but cannot do anything on its own
  - Needs to be told what to do next in the form of a stream of "instructions"
    - These instructions are generated from a program that represents an algorithm for accomplishing the task
  - Can store intermediate/final results of a computation in main memory
    - Dynamic random access memory (DRAM); volatile
  - Can store information on persistent non-volatile storage media e.g., magnetic hard disk

Anatomy of a computer system

- Peripheral I/O devices
  - Plug-ins to the CPU for communicating with the world
  - Display (CRT, LCD, touchscreen)
  - Keyboard
  - Mouse
  - DVD reader/writer
  - Speaker
  - Microphone
  - Camera
  - Wireless communication
  - Wired Ethernet communication

Anatomy of a computer system

- How does an instruction execute inside the processor?
  - Every entity residing inside a computer has an address
    - An instruction also has an address
  - The processor maintains the address of the next instruction in a register called program counter
  - The instruction is fetched from main memory and placed in an instruction register
  - The instruction is decoded to generate the control signals for executing the instruction
    - Send to adder if this is an addition instruction

Anatomy of a computer system

- How does an instruction execute inside the processor?
  - Most instructions require source operands for execution
    - a+b
  - After decoding the instruction, the operands are fetched
    - Operand addresses are typically encoded in the instruction or could be implicit
    - These addresses are known after decoding the instruction
  - The instruction can now execute
    - Operands are sent to the adder

Anatomy of a computer system

- How does an instruction execute inside the processor?
  - Most instructions generate a result
    - c=a+b
  - The address of the result (or destination) operand is typically encoded in the instruction or implicit
    - This address is known after decoding the instruction
  - The result is stored in the destination operand location

Anatomy of a computer system

- How does an instruction execute inside the processor?
  - The execution of an instruction requires the appropriate control and data paths to be activated
- Data path is usually slow because main memory is much slower than the processor
  - Commonly used optimizations for speeding up the data path:
    - Reasonably large set of general-purpose registers inside the processor
    - Fast memory (known as cache) inside the processor