Introduction to Computer Architecture

Why, How, and What for ???

Smruti R. Sarangi

Department of Computer Science
Indian Institute of Technology
New Delhi, India

Outline

1. Introduction
2. History
    - Ancient Age
    - Middle Age
    - Modern Age
3. Theoretical Developments
    - Abstract Machine Models
    - Theoretical Instruction Sets

What is Computer Architecture?

Figure 1: Courtesy: www.psychology-today.com

Computer Architecture

- The CPU is the brain of a computer system.
- It works both consciously and subconsciously.
- Consciously : Executes a program
- Sub-consciously : Runs the operating system, co-ordinates with I/O devices

Computer Architecture : Study of the CPU and the peripherals

Where does it fit in?

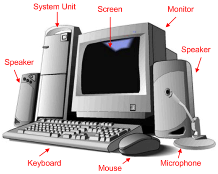

Figure 2: courtesy: www.coolnerds.com

Example

- Computer Architecture –> Brain
- Networking –> Nervous and Circulatory System
- Computer Vision –> Eyes
- Operating System–> Endocrine and Immune System
- Databases –> Memory
- Algorithms –> Intelligence
- Prog. Languages –> Linguistic Center
- ...

Why Study Computer Architecture?

- Understanding
    - Learn the inner workings of processors
    - Understand hardware/software interaction
    - Design better operating systems and compilers
- Career Prospects
    - Companies directly working in architecture
        - Intel, AMD, Sun/Oracle, Arm, IBM
    - Systems Software
        - Google, Samsung, VMWare, Wind River, McAfee
- Higher Studies ...

Outline

1. Introduction
2. History
   - Ancient Age
   - Middle Age
   - Modern Age
3. Theoretical Developments
   - Abstract Machine Models
   - Theoretical Instruction Sets

History of Computing

History of Computing

- 2400 BC: The babylonians invented the abacus.

History of Computing

- 2400 BC: The babylonians invented the abacus.
- 500 BC: Mathematicians in India started using zero.
- 500 BC: Indian grammarian, gave the first Turing complete grammar for programming languages. This led to the famous paradigm, *Panini-Backus* form.

History of Computing

- 2400 BC: The babylonians invented the abacus.
- 500 BC: Mathematicians in India started using zero.
- 500 BC: Indian grammarian, gave the first Turing complete grammar for programming languages. This led to the famous paradigm, *Panini-Backus* form.

History of Computing

- 2400 BC: The babylonians invented the abacus.
- 500 BC: Mathematicians in India started using zero.
- 500 BC: Indian grammarian, gave the first Turing complete grammar for programming languages. This led to the famous paradigm, *Panini-Backus* form.
- 300 BC: Indian mathematician, Pingala, proposed the binary number system.

History of Computing

- 2400 BC: The babylonians invented the abacus.
- 500 BC: Mathematicians in India started using zero.
- 500 BC: Indian grammarian, gave the first Turing complete grammar for programming languages. This led to the famous paradigm, *Panini-Backus* form.
- 300 BC: Indian mathematician, Pingala, proposed the binary number system.
- 100 BC: Chinese invent negative numbers.

History of Computing

- 2400 BC: The babylonians invented the abacus.
- 500 BC: Mathematicians in India started using zero.
- 500 BC: Indian grammarian, gave the first Turing complete grammar for programming languages. This led to the famous paradigm, *Panini-Backus* form.
- 300 BC: Indian mathematician, Pingala, proposed the binary number system.
- 100 BC: Chinese invent negative numbers.
- 60 BC: Heron of Alexandria invented a mechanical machine, which follows completed a sequential series of actions.

History of Computing

- 2400 BC: The babylonians invented the abacus.
- 500 BC: Mathematicians in India started using zero.
- 500 BC: Indian grammarian, gave the first Turing complete grammar for programming languages. This led to the famous paradigm, *Panini-Backus* form.
- 300 BC: Indian mathematician, Pingala, proposed the binary number system.
- 100 BC: Chinese invent negative numbers.
- 60 BC: Heron of Alexandria invented a mechanical machine, which follows completed a sequential series of actions.
- 600 AD: Indian mathematician, Brahmagupta, described the place value system.

Outline

1 Introduction

2 History
- Ancient Age
- Middle Age
- Modern Age

3 Theoretical Developments
- Abstract Machine Models
- Theoretical Instruction Sets

History of Computing - II

- 1206 AD: Arab engineer, Al Jazari, invented a basic robot. This was a human mannequin, which could move its hands. It was hydro powered.
- 1400 AD: Kerala school of astronomy, mathematicians, invent the floating point number system.
- 1492 AD: Leonardo Da Vinci invents the mechanical calculator.
- 1622 AD: William Oughtred invented the slide rule.

Russian Mechanical Calculator

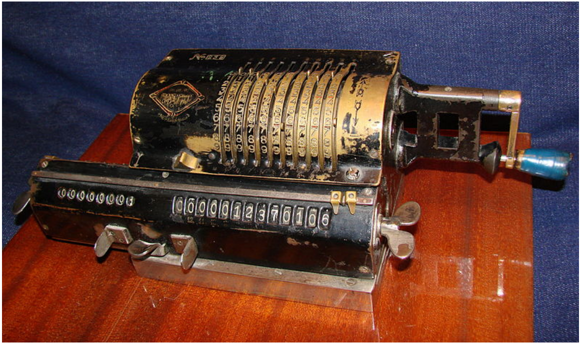

Figure 3: courtesy wikipedia

Slide Rule

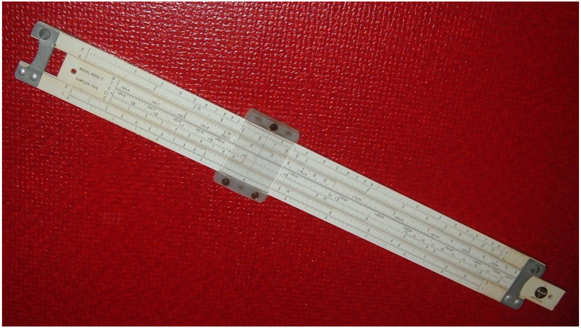

Figure 4: courtesy wikipedia

History of Computing - Middle Age

- 1642: Blaise Pascal made the first widely used mechanical calculator, Pascaline. It could add, subtract, multiply, and divide. it consisted of a series of wheels.

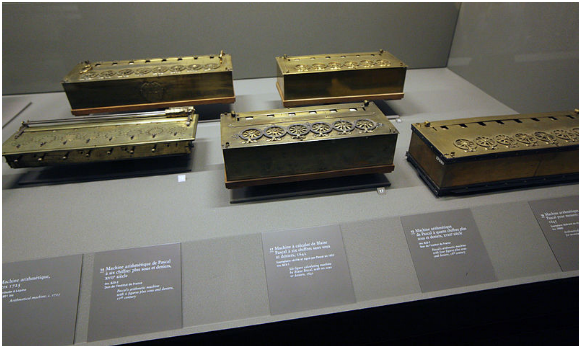

Figure 5: The Pascaline, courtesy wikipedia

History of Computing - Middle Age - II

- 1801: Joseph-Marie Jacquard developed an automatic loom. This loom was controlled by punch cards. First example of a stored program machine.

History of Computing - Middle Age - II

- 1801: Joseph-Marie Jacquard developed an automatic loom. This loom was controlled by punch cards. First example of a stored program machine.
- 1834: Charles Babbage designed the first general purpose mechanical computer called the Analytical Engine.
  - It had the notion of a program. This was stored in punch cards.
  - It had an arithmetic unit that could perform all arithmetic operations, compare numbers, and compute square roots.
  - It had the notion of a memory that could save programs and data.
- 1848: British mathematician, George Boole, invented Boolean algebra.

Punch Cards

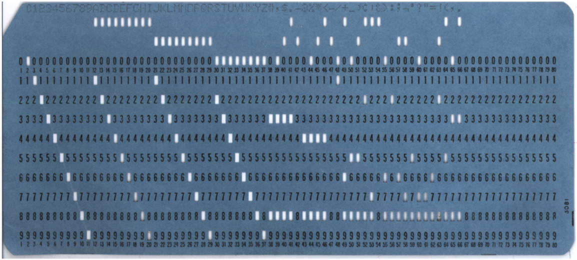

Figure 6: Punched Card,courtesy wikipedia

Jacquard Mill

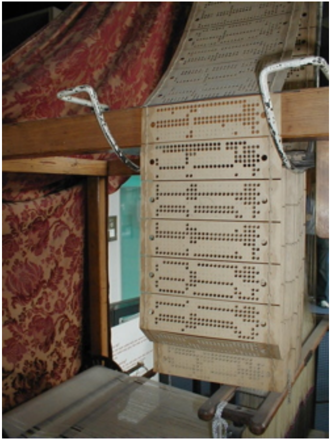

Figure 7: Jacquard Mill,courtesy wikipedia

Outline

1. Introduction
2. History
   - Ancient Age
   - Middle Age
   - Modern Age
3. Theoretical Developments
   - Abstract Machine Models
   - Theoretical Instruction Sets

History of Computing - Modern Age

- 1890: Herman Hollerith won a competition to build a machine for the US census bureau. He founded the Computing Tabulating and Recording Corporation. This company went on to become IBM.
- 1892: Burroughs found the Arithmetic Arithmometer Company. This computer went on to become Burroughs, and then Unisys.
- 1900-1950: Computers started using vacuum tubes and electro-mechanical relays.
- 1900-1950: Other than two world wars, something else happened in the world of computing.

How do we solve problems?

Let us rewind back to 1900 AD ...

- Computing was still at its infancy.
- We could at the most add a few hundred numbers.
- Very limited storage space.
- Computers weighed a few tons.
- Uses limited to: accounting, surveying

What about the rest of the engineering fraternity?

State of the Art in 1939

What did the 19^{th} and the early 20^{th} century give us ...

- Steel
- Electricity
- Internal Combustion Engine
- Telegraph and Telephone
- Railroads
- Antiseptics, Anaesthesia, Antibiotics
- X-Rays
- Discovery of the atomic structure
- Skyscrapers

The Empire State Building

Figure 8: courtesy wikipedia

Did computers help us design this wonder of the world ... ,
WHY NOT

What was missing?

- Computer Architecture is a marriage of two things:
  - A theoretical model of computation
  - A realization of this model on practical devices

Answer: We had none

Why

Without any formal models of computation, and with extremely slow electro-mechanical devices like manually turned knobs, water powered computers, or vacuum tubes, the computing industry was very primitive.

Outline

1. Introduction
2. History
   - Ancient Age
   - Middle Age
   - Modern Age
3. Theoretical Developments
   - Abstract Machine Models
   - Theoretical Instruction Sets

The Turing Machine

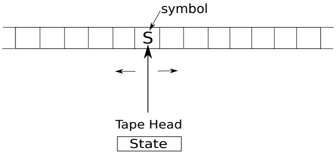

Figure 9: The Turing Machine

Turing Machine

- It consists of an unbounded tape divided into slots. Each slot contains a symbol.
- The tape head is associated with a *state*.
- At any step we have the following transition:

$$< Symbol, State > \Longrightarrow < NewSymbol, NewState, Left|Right >$$

Turing Machine - II

- This machine is powerful enough to do most common computations.
  - All kinds of arithmetic operations, solving differential equations, algebraic formulae, ...
  - Programs of arbitrary complexity, recursive function calls

Definition (Turing Complete)

A programming language, or a computational machine, is said to be Turing Complete if it can be used to simulate a Turing Machine.

Other Models

At the same time, other models were proposed, which are equivalent to a Turing machine ...

- American mathematician, Alonzo Church, proposed λ calculus.
    - It only consists of one input functions.
    - Two operations : Function definition and application
    - Forms the basis of functional languages like Scheme & ML
- Church along with Kleene and Rosser, formed a new formalism based on repeated function recursion.

Church-Turing Thesis

Question

- Can I design a computer that can compute anything from differential equations to estimating the number of people who like Chicken Tandoori in Delhi ???
- What if in the future, I get a problem, which this problem cannot compute? For example, that only does addition and subtraction, it is impossible to write a program that prints out my name *n* times. *n* is an user input.

Church Turing Thesis - II

Answer

Everything computable is computable by a Turing machine.

- This is a thesis, not a theorem.
- It has held for the last 75 years.
- There are some functions that are not computable by a Turing machine.
    - Write a function to find if a program contains an infinite loop.

99.9999 ... 999% of the programs that we encounter everyday, can be solved by a Turing Machine.

Outline

1. Introduction
2. History
    - Ancient Age
    - Middle Age
    - Modern Age
3. Theoretical Developments
    - Abstract Machine Models
    - Theoretical Instruction Sets

Von Neumann Machine

One Instruction Set Computer

Basic Instructions