6.823
Computer System Architecture

Lecturers: *Krste Asanovic, Srini Devadas*

What's 6.823 About?

- What's under the "hood" of your desktop?
- How powerful is the "engine"?
- How much "gas" does it burn?
- How do you build one?

➢ From the Beta (6.004) RISC processor to the Pentium-4

We won't emphasize:

VLSI implementations
Parallel processing
Quantitative evaluation

Course Information

You must sign up for the course through the web

~7 Homeworks (30%) *Midterm (30%)* *Final (40%)*

Tutorials : One session / week
1 hour / session

All students must help grade homeworks once during semester, signup sheets distributed during class

Course Information (contd.)

Textbook: Hennessy and Patterson – Computer Architecture: A Quantitative Approach (strongly recommended)

Prerequisite material: Patterson and Hennessy – Hardware/Software Interface book

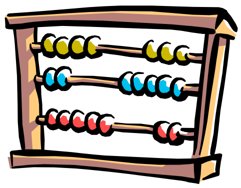
- Some material in lectures will be from other sources

Tutorials: No new material introduced, reinforcement of lecture content using examples

Problem Set 0

- Goal is to help you judge for yourself whether you have prerequisites for this class
- We assume that you understand digital logic, a simple 5-stage pipeline, and simple caches
- For this problem set only, work by yourself – not in groups
- Due at start of class.

History of Calculation and Computer Architecture



The Importance of Calculation

- **Plato in *Philebus* ("On Pleasure") 4th century BC**
  - **On the very first level of knowledge are to be found number and calculation**
- **Pythagorean mysticism**
  - **Do numbers rule the world?**
- **Scientists needed to escape the hardships of human calculation**
  - **Chinese abacus (suan pan) – 13th century AD: First instrument to provide a simple means to carry out all the operations of arithmetic**
  - **Abacus still in use today!**

PLATO

If you cannot calculate, you cannot speculate on future pleasure and your life will not be that of a human, but that of an oyster or jellyfish.

LUIGI MENABREA

How the prospects of long and arid calculations have demoralized great thinkers, who seek only time to meditate but see themselves swamped by the sheer mass of arithmetic to be done by an inadequate system! And yet, it is the path of laborious analysis that leads to truth, but they cannot follow that path without employing the guide of number, for which number there is no way to life the veil which covers the mysteries of nature – 1884

CHINESE ABACUS 13th Century AD Several rods, five lower balls on each rod, 2 upper balls, divided by a slat

JAPANESE ABACUS soroban, more refined, reduced to 1 upper ball, and 4 lower balls

Calculating Machines

- **Wilhelm Shickard’s Calculating Clock (1623) was the earliest calculating machine in history!**
  - multiplication and division required several interventions by the operator
  - Operated on the principle of “Napier’s bones”
- **Possibility of mechanizing arithmetic was demonstrated by Blaise Pascal in the machine “Pascaline” (1642)**
  - Series of toothed gears, each numbered 0 to 9
  - First calculating machine to be commercialized!
- **Arithmometer (1822), keyboard (1850), printer (1885), Electro-mechanical Arithmometer (1913)**

On 20th September 1623, Shickard wrote as follows to his friend Kepler: “The calculations which you do by hand, I have recently attempted to achieve mechanically … I have constructed a machine which, immediately and automatically, calculates with given numbers, which adds, subtracts, multiplies and divides. You will cry out with joy when you see how it carries forward tens and hundreds, or deducts them in subtractions …”

Kepler sure would have appreicated such an invention to create his tables of the movements of the planets, but unfortunately, Schickard’s one and only copy of his own machine was destroyed by fire on 22 February 1624.

Napier’s bones or rods (1617). Ten wooden rods, of square cross-section. Each of the four sides of each rod corresponded to one of the digits from 0 to 9, and marked down its length in nine divisions, were the multiples of the corresponding digit. A kind of multiplication table, where you read off the product horizontally when the rods were placed side by side.

Pascaline was not very reliable. When one wheel completed a revolution, the next wheel would advance a step. The automatic carrying mechanism tended to jam when several wheels were simultaneously at 9, necessitating several simultaneous carries (999 to 1000)

Thomas, (director of a French insurance company) Arithmometer was the first machine to be commercialized on a large scale.

Programmability?

- **Problems with all these machines lay with their limited capability to carry out a linked sequence of calculations**
  - **Needed to transcribe and enter all intermediate results!**
- **Vaucanson's programmable androids (1738)**
- **Vaucanson (1749) constructed the first automatic loom**
  - **Accepted commands by means of a perforated metal cylinder**
- **Jacquard (1804) perfected the programmable loom**
  - **Activated by a *sequence* of punched-cards!**

Basile Bouchon (1725) invented a loom that accepted commands by means of a punched tape

Vaucanson in 1738 developed the Digesting Duck, an artificial automaton for Louis XV.

"He stretches out his neck to go and take the grain from the hand, he swallows it, digests it, and excretes it, once digested through the normal channels; all the gestures of a duck swallowing rapidly, speeding up the movement in his throat to pass the food into his stomach, are copied from nature".

Vaucanson turned his loom into a programmable but cylical automaton, one in which the commands were inscribed by means of perforations on a hydraulically-driven drum, and were regularly repeated.

Jacquard combined the use of a moving drum equipped with a sequence of punched cards and the concept of a swinging arm that lifted hooks.

Charles Babbage 1791-1871

Lucasian Professor of Mathematics
Cambridge University, 1827-1839

Difference Engine 1823

Application?
- Mathematical Tables - Astronomy
- Nautical Tables – Navy

Background
- Any continuous function can be approximated by a polynomial --- *Weierstrass*

Technology
- mechanical - gears, Jacquard's loom, simple calculators

First digital calculators with an automatic capability of effecting chained sequences of operations following a program set up in advance in a control mechanism were the difference machines.

Difference Engine

A machine to compute mathematical tables

Weierstrass:

- Any continuous function can be approximated by a polynomial
- Any polynomial can be computed from *difference* tables

$f(n) = n^2+n+41$

$d1(n) = f(n) - f(n-1) = 2n$

$d2(n) = d1(n) - d1(n-1) = 2$

$f(n) = f(n-1) + d1(n) = f(n-1) + (d1(n-1) + 2)$

n	0	1	2	3	4 ...
d2(n)			2	2	2
d1(n)		2	4		
f(n)	41	43	47		

all you need is an adder!

Difference Engine

1823 - Babbage's paper is published

1834 - The paper is read by Scheutz brothers in Sweden

1842 - Babbage gives up the idea of building it; (he is onto Analytic Engine!)

1855 - Scheutz displays his machine at the Paris World Fare
- Can compute any 6^{th} degree polynomial
- *Speed:* 33 to 44 32-digit numbers per minute!

Now the machine is at the Smithsonian

Babbage was funded by the British Association for the advancement of Science, but failed to build the machine.

He was unable to moderate his ambitions, and gradually lost interest in the original difference engine

Scheutz brothers' machine was the first working calculator in history that did print out the results.

Analytic Engine

1833 - Babbage's paper is published

conceived during a hiatus in the development of the difference engine

Inspiration:*Jacquard's Loom*

The set of cards with fixed punched holes dictated the pattern of weave ⇒ *program*
The same set of cards could be used with different colored threads ⇒ *numbers*

1871 - Babbage dies - the machine remains unrealized.

It is not clear if the analytic engine could be built even today using only mechanical technology

However, near the end of his life he became depressed. "If I could live just a few more years, the Analytical Engine would exist and its example spread across the entire planet". Then he added, even more pathetically, "If any man, who is not rebuffed by my example, one day produces a machine containing within it all of the principles of mathematical analysis, then I have no fear for my memory, for he alone will be able to appreciate fully the nature of my efforts and the value of the results I have obtained"

Analytic Engine

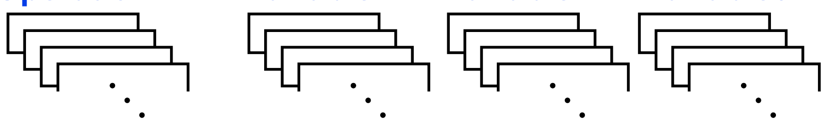
The first conception of a general purpose computer

1. The *store* in which all variables to be operated upon, as well as intermediate results are placed.

2. The *mill* into which the quantities about to be operated upon are always brought.

The *program*

Operation variable1 variable2 variable3



An operation in the *mill* required feeding two punched cards and producing a new punched card for the *store.*

An operation to alter the sequence was also provided!

One of the most striking features of Babbage's Analytical Engine is the way conditional operations were to be handled. Proposed that a lever only move if the result of the calculation was negative, and that is should be used to advance or roll back the cards on the Jacquard mechanism to any specified extent.

The first programmer

Ada Byron *aka* "Lady Lovelace"

Babbage's ideas had a lot of influence later, primarily because of

- ***Luigi Menabrea*, who published notes of Babbage's lectures in Italy**
- ***Lady Lovelace*, who translated Menabrea's notes in English and thoroughly expanded them.**
  "... Analytic Engine weaves *algebraic patterns*...."

Ada's tutor was Babbage himself!

In the early 20th century - the focus shifted to analog computers but …

Countess Lovelace (1815-1852) was Lord Byron's only daughter. Hard to imagine a greater contrast between the poet and his daughter, who had applied herself to the exact and arduous study of calculating machines.

She devised a certain number of programs with the idea of one day introducing them to the machine of her friend and master. "We can say that the Analytical Engine will weave algebraic patterns, just as Jacquard looms weave flowers and leaves".

Harvard Mark I

Built in 1944, in the IBM laboratories at Endicott by Howard Aiken – Professor of Physics at Harvard

Essentially mechanical but had some electro-magnetically controlled relays and gears

Weighed *5 tons* and had *750,000* components
A synchronizing clock that beat every *0.015* seconds

Performance:
- **0.3 seconds for addition**
- **6 seconds for multiplication**
- **1 minute for a sine calculation**

Broke down once a week!

Over 500 miles of electrical wiring, and 3 million solder joints!

72 registers of 23 bits each

At best took a few minutes to repair, averaged 20 minutes, sometimes several hours.

During the final months of World War II it was used exclusively by the US Navy to solve problems in ballistics.

Decommissioned in 1959.

Linear Equation Solver

John Atanasoff, Iowa State University

1930's: Atanasoff and Clifford Berry built the Linear Equation Solver. It had 300 tubes!

Application:

Linear and Integral differential equations

Background:

Vannevar Bush's Differential Analyzer

--- ***an analog computer***

Technology:

Tubes and Electromechanical relays

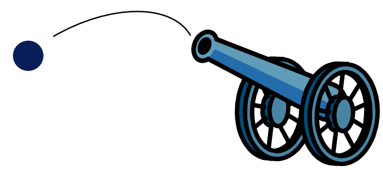
Atanasoff decided that the correct mode of computation was by electronic digital means.

The physical and logical structures of the machine were fairly rudimentary and it was never an analytical calculator in the true sense of the term (in that it was not programmable). Furthermore, it never worked properly.

Electronic Numerical Integrator and Computer (ENIAC)

- Inspired by Atanasoff and Berry, Eckert and Mauchly designed and built ENIAC (1943-45)
- The first, completely electronic, operational, general-purpose analytical calculator!

- 30 tons, 72 square meters, 200KW
- Performance
  - Read in 120 cards per minute
  - Addition took 200 μs, Division 6 ms
  - 1000 times faster than Mark I



- Not very reliable!

Designers were obliged to install a ventilation shaft and a cooling system in the room. Machine had 18,000 vacuum tubes.

The slogan was: The ENIAC is able to calculate the trajectory of a large-caliber naval shell in less time than the shell takes to reach its target!

To change a program, it was necessary to change the instructions, the connector panels, and the positions of the switches all at the same time.

The inventors of ENIAC themselves admitted that, after taking into account human error, the machine only got the correct result 20 times out of 100!

Electronic Discrete Variable Automatic Computer (EDVAC)

- **ENIAC's programming system was external**
  - **Sequences of instructions were executed independently of the results of the calculation**
  - **Human intervention required to take instructions "out of order"**
- **Eckert, Mauchly, John von Neumann and others designed EDVAC (1944) to solve this problem**
  - **Solution was the stored program computer**
- ***First Draft of a report on EDVAC*** **was published in 1945, but just had von Neumann's signature!**
- **In 1973 the court of Minneapolis attributed the honor of** ***inventing the computer*** **to John Atanasoff**

Von Neumann (1903-1957) was born in Budapest. Great quantities of work in set theory to quantum physics. Celebrated for his theory of games and its application to economics.

Report was the formal specification of EDVAC. Before that the documentation on ENIAC or EDVAC was non-existent. Von Neumann devised a mathematical logical notation which expressed fundamental ideas of the fetch-execute-decode loop.

Five major components: Arithmetic unit, control unit, the memory, input devices and output devices.

Stored Program Computer

Program = A sequence of instructions

How to control instruction sequencing?

manual control	**calculators**
automatic control	
external (paper tape)	**Harvard Mark I , 1944**
	Zuse's Z1, WW2
internal	
plug board	**ENIAC 1946**
read-only memory	**ENIAC 1948**
read-write memory	**EDVAC 1947 *(concept)***

⇒ The same storage can be used to store program and data

EDSAC	**1950**	**Maurice Wilkes**

EDSAC: Electronic Delay Storage Automatic Calculator, translation into machine language of commands input in symbolic form; automatic loading of the program into core memory, etc.

The First True Computers

- **Eckert and Mauchly founded their own company and built the BINAC 1947-49**
  - Two processors that checked each other for reliability
- **Whirlwind I of MIT was another von Neumann computer built between 1946-51 by Jay Forrester**
  - Had a magnetic-core memory and a programming language
  - Used by the US Air Defense
- **First *commercial* American computer was UNIVAC-I designed and built by Eckert and Mauchly**
  - Used for opinion polls in 1952 Presidential elections

BINAC was the first electronic computer built in the United States.

There were several von Neumann computer efforts, including SEAC.
UNIVAC-I. CBS television polls, accurately predicted Eisenhower's victory.

Dominant Problem: *Reliability*

Mean time between failures (MTBF)
MIT's Whirlwind with an MTBF of 20 min. was perhaps the most reliable machine !

Reasons for unreliability:

1. **Vacuum Tubes**

2. **Storage medium**
   acoustic delay lines
   mercury delay lines
   Williams tubes
   Selections

CORE	**J. Forrester**	**1951**

Commercial Activity -- 1948-52

IBM's SSEC

Selective Sequence Electronic Calculator

- Vacuum tubes in the control unit, and electro-magnetic relays everywhere else
- 150 word store, stored program machine
- Instructions, constraints, and tables of data were read from paper tapes.
- 66 Tape reading stations!
- Serious inconsistencies of logic.

Exhibited by IBM at the start of 1948 in the shop windows of a busy New York avenue.

Fascinated the public, who came in their thousands to look at the lights on the calculator blinking away.

Was a “near-computer” because there a lack of synchronization in the calculation due to its hybrid nature.

IBM Computers

IBM 701 -- 30 machines were sold in 1953-54

**IBM 650 -- a cheaper, drum based machine,
more than 120 were sold in 1954
and there were orders for 750 more!**

Users stopped building their own machines.

Why was IBM late getting into computer technology?

They were making too much money!

Even without computers, IBM revenues were doubling every 4 to 5 years in 40's and 50's.

Defense Calculator was another name for the IBM 701.

Commissioned during the Korean War.

Software Developments

up to 1955 Libraries of numerical routines
- Floating point operations
- Transcendental functions
- Matrix manipulation, equation solvers, . . .

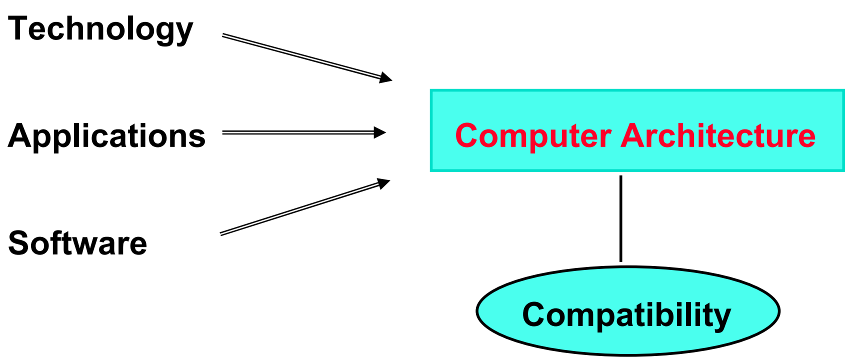
1955-60 *High level Languages* - Fortran 1956
Operating Systems -
- Assemblers, Loaders, Linkers, Compilers
- Accounting programs to keep track of usage and charges

Machines required *experienced operators*

⇒ Most users could not be expected to understand these programs, much less write them

⇒ Machines had to be sold with a lot of resident software

Factors that Influence Computer Architecture



Software played almost no role in defining an architecture before mid fifties.

special-purpose *versus* general-purpose machines

Compatibility

Essential for *portability* and *competition*

Its importance increases with the market size
but it is also the most *regressive* force

Instruction Set Architecture (ISA) compatibility

The same assembly program can run on any upward compatible model
***then* IBM 360/370 ... *now* Intel x86**

System and application software developers expect more than ISA compatibility ***(API's)***

applications
operating system
proc + mem + I/O

Java?

Wintel

Microprocessor Economics

- Designing a state-of-the-art microprocessor requires a huge design team
  - Pentium ~300 engineers
  - PentiumPro ~ 500 engineers
- Huge investments in fabrication lines
  - ⇒ need to sell 2 to 4 million units to be profitable
- Continuous improvements are needed to improve yields and clock speed
  - ⇒ price drops to one tenth in 2-3 years
- Fast new processors also require new peripheral chips (memory controller, I/O) ⇒ $$$

Cost of launching a new ISA is prohibitive and the advantage is not so clear!

View of Computer Architecture

Language/ Compiler/ System software designer		*Architect/Hardware designer*
Need mechanisms to support important abstractions	⇒	**Decompose each mechanism into essential micro-mechanisms and determine its feasibility and cost effectiveness**
Determine compilation strategy; new language abstractions	⇐	**Propose mechanisms and features for performance**

Architects' main concerns are cost-performance, performance, and efficiency in supporting a broad class of software systems.