



# Beginning Snapshots

Chapter 0.

# Chapter Objectives

## Parts of the Picture

- What is Computer Science
- The History of Computing
- Introduction to Computer Systems

# Chapter Objectives

- To give an overview, to demonstrate the breadth of Computer Science
- To provide the context of today's computing by noting significant events from the past
- To describe basic components of computer systems
- To understand methods and features of programming

# What is Computer Science?

○ It is not just writing computer programs

○ Computer science includes:

- Algorithms
- Data structures
- Architecture
- Artificial intelligence
- Robotics
- Human-Computer communication
- Numerical and symbolic computation
- Operating Systems
- Programming Languages
- Software Engineering
- Ethical issues



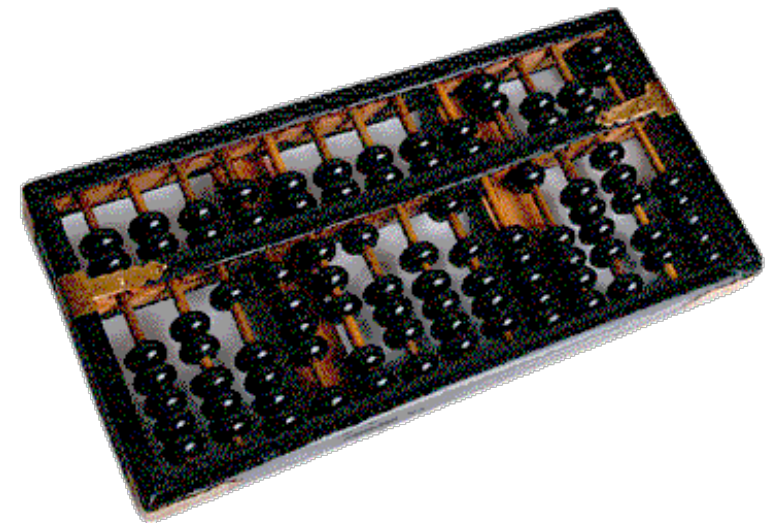
# The History of Computing



- Check out Computer History web site.
- Important concepts that shaped the history of computing:
  - The mechanization of arithmetic
  - The stored Program
  - Graphical user interface
  - The computer network

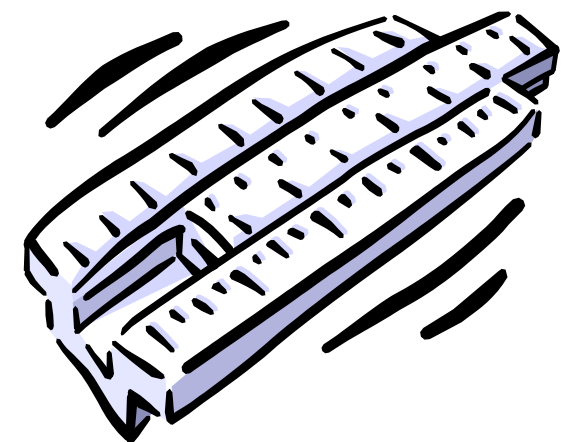
# Machines to Do Arithmetic

The abacus



- Napier's Bones
- The Pascaline
- Leibniz' calculator

The slide rule



# The Stored Program

- Program: a sequence of instructions for the computer to follow. Also called “software”
- Hardware: the chips, wires, switches, etc. on which the software instructions are executed
- Primitive example: the Jacquard Loom
  - The loom was the hardware
  - The weaving pattern cards was the software
  - The “program” was “stored” on punched cards

# Mechanical Computers

- Charles Babbage
  - Difference engine computed polynomials
  - Analytical engine
    - Provided a memory of 1000 50-digit numbers
    - Had processing, storage, input, output
    - Was never built – the technology of his day could not build his design
  - Babbage called the “Father of Computing”

# Mechanical Computers

- Ada Augusta
  - Understood Babbage's machines
  - Developed "programs"
  - Could be called the first programmer
- In the 1980's a computer language (Ada) was named after her

# Electromechanical Computers

- Herman Hollerith
- Developed a punched card tabulating machine
- Used for the 1890 census
- His company was one of several which began IBM

# Electromechanical Computers

- Konrad Zuse
  - Proposed use of vacuum tubes for switching of binary circuits
  - Hitler refused to fund his design
- Alan Turing
  - Computer theorist
  - Worked on the Colossus, used to decrypt German military messages, WW2
- Grace Hopper – coder on the Harvard Mark I
  - It used electrical relays
  - Sponsored by US Navy to compute navigational tables



# Early Electronic Computers

- ABC computer
  - Developed by Atanasoff & Berry at Iowa State to do math & physics calculations
- ENIAC
  - Used 18,000 vacuum tubes, caused lights to dim in Philadelphia neighborhoods when turned on
  - Programmed by rewiring panels



# Early Electronic Computers

- John von Neumann
  - Inventor of stored program concept
- Eckert and Mauchly
  - Built the UNIVAC (UNIversal Automatic Computer)
  - Used by the Census Bureau in 1950s

# Second-Generation Computers

- Characterized by use of transistors in place of vacuum tubes
- During late 1950s, early 1960s, programming languages developed
  - FORTRAN (FORmula TRANslation)
  - LISP (LIst Processing)
  - COBOL (Common Business Oriented Language)

# Third-Generation Computers

- Characterized by integrated circuits
  - Pioneered by Jack Kilby and Robert Noyce
- IBM System/360
  - First of 3rd generation computers
- Operating Systems developed
  - UNIX (1971)
  - MS-DOS (1981)
- PDP-8 first commercially successful minicomputer
- ARPANET beginning of the Internet (1969)
  - ARPA Advanced Research Projects Agency

# Fourth-Generation Computers

- Characterized by Very Large Scale Integrated circuits (VLSI)
  - Beginning of microprocessors and personal computers
- Other events of 1970s
  - C language developed by Dennis Ritchie
  - Ethernet
  - Altair 8800 first hobby-kit computer
  - Bill Gates, Pallen write BASIC compiler for Altair
  - Steve Jobs, Steve Wosniak develop first Apple Computers
  - First super computer CRAY 1



# Albuquerque



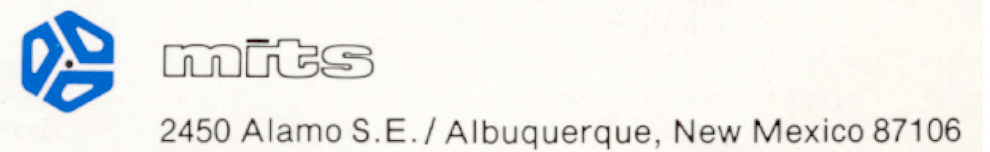
1975: speeding and  
driving without a license

1977: running a stop sign

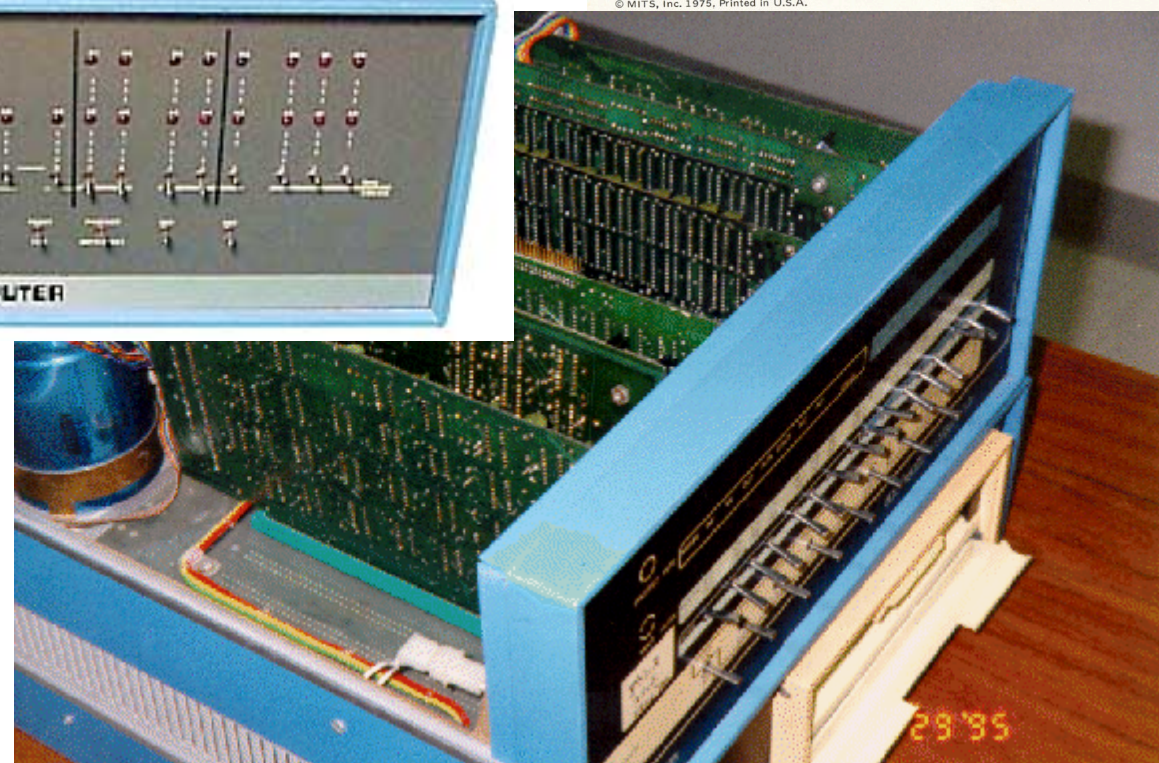
Sundowner Hotel (2005), just west of  
Walgreen's at Central and San Pedro







1975 MITS (Micro  
Instrumentation and  
Telemetry Systems)  
Altair 8800



The name "Altair" came from an episode of Star Trek,  
or simply the name of a star ...



# The Graphical User Interface (or GUI)

- Human-computer interaction was done with a difficult and cryptic command line interface
- Doug Englebart at Xerox PARC developed graphical interface, first to use a “mouse”
  - Concept went unused until Steve Jobs saw it
  - He adapted the concept for the Macintosh
  - Microsoft responded with Windows operating system
  - X Window system developed at M.I.T. for Unix

# Networks

- Definition  $\Rightarrow$  two or more computers connected to exchange resources
  - Hardware resources
  - Software resources
  - Data
- Early networking
  - Timesharing, mainframe to terminals via modems
  - ARPANET connected research center computers for the Department of Defense



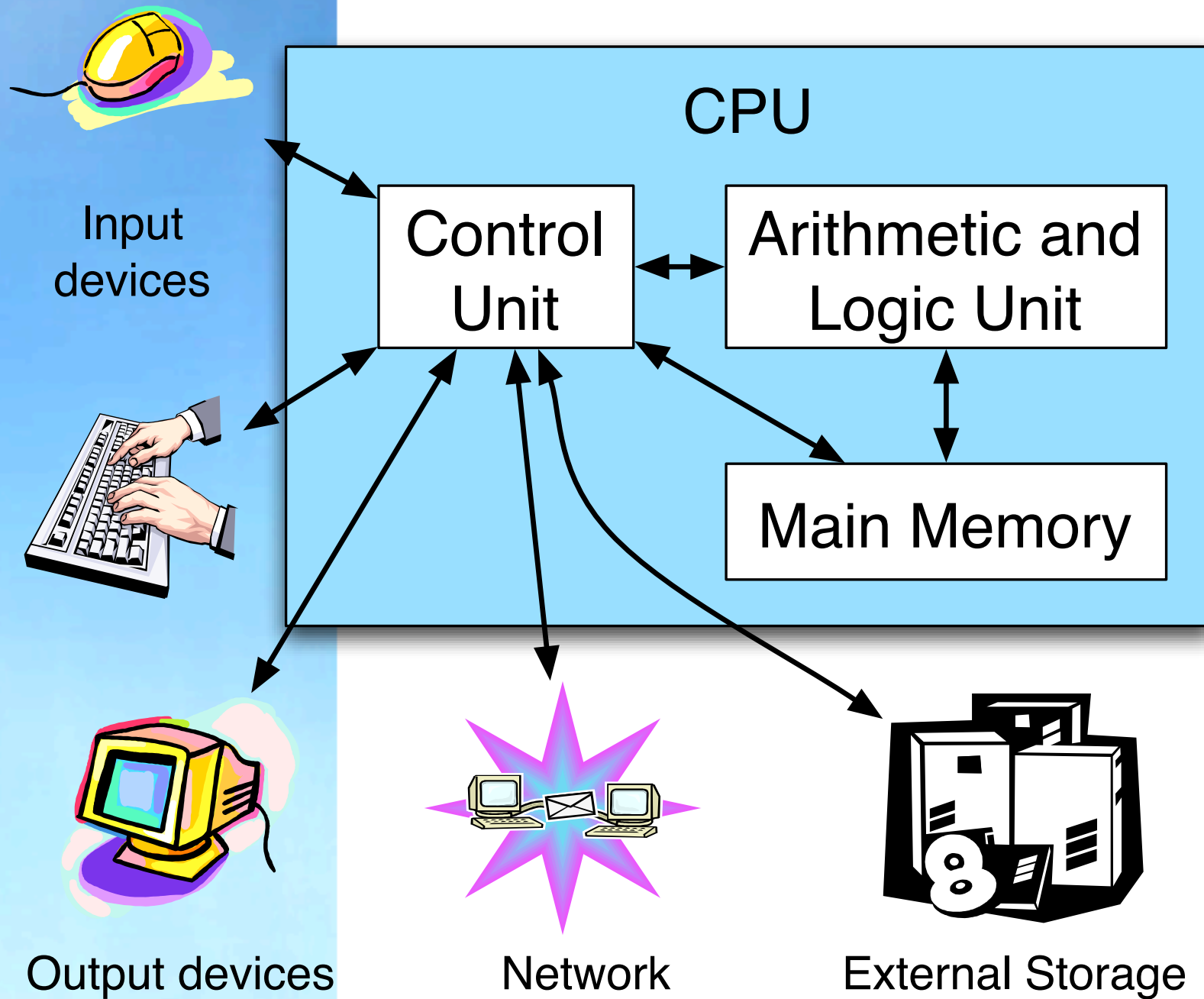
# Networks

- The Internet
  - Grew out of the ARPANET as popularity and the number of host computers grew
  - Internet service providers enable even home computers to be wired into the global digital infrastructure
- Local Area Networks (LANs)
  - Enabled by Ethernet hardware and network operating systems
  - PC users share resources

# Introduction to Computer Systems

- Babbage's Analytical Engine was designed with capabilities of ...
  - Processing
  - Storage
  - Input
  - Output
- This is still a common feature in most modern computers

# Processing



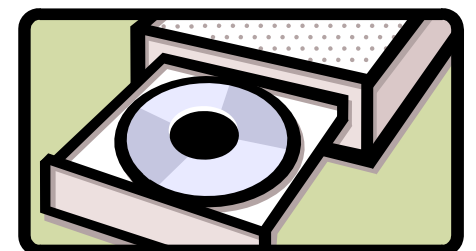
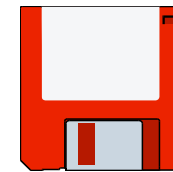
- Major components of a computer
  - CPU or Central Processing Unit
  - Primary (internal) memory
  - Secondary (external memory)
  - Control Unit
  - Arithmetic Logic Unit
  - Input devices
  - Output devices

# Storage

- Main Memory
  - Internal, primary, random access memory (RAM)
  - Stores instructions and data
- Cache memory
  - Smaller quantity of high speed memory
- Registers
  - Specific high speed memory locations used repeatedly by instructions
  - Three orders of magnitude faster than RAM

# Storage

- Secondary, external, auxiliary storage
- Needed because internal memory is volatile – loses contents when power is off
- Media used
  - Magnetic disks (floppy disks, hard drives)
  - Optical media (CDs, DVDs)



# Storage

- Binary storage
  - Storage locations are a collection of two-state values (either 0 or 1 – Binary digiTs)
  - These are called BITS
- Bytes
  - 8 bits make a byte
  - 1024 bytes make a kilobyte
  - Note kilo (1000) not quite accurate
  - Computer memory established in multiples of powers of 2 ...  $2^{10} = 1024$
  - 1024 kilobytes = 1 Megabyte = 1,048,576 bytes
  - 1024 Megabytes = 1 Gigabyte = 1,073,741,824 bytes

# Input and Output

- Input devices

- Convert instructions and data into binary form

- Transmit to the CPU

- Output devices

- Convert binary contents of memory into meaningful symbols for humans to view

- Outputs information as graphics, sounds, video, and even robotic movement





# Input and Output

- Communication between CPU and I/O devices
- Specific or general connections to these peripheral devices called ports
- Ports connect to the computer's bus



# Operating Systems

- Required for a computer to be general purpose
- The tasks of the operating system (OS) include...
  - Interface between user and system hardware
  - Environment in which other software programs can run
- Hardware and OS together make up a platform
- Examples:
  - DOS, Windows, UNIX

# Programming

- Definition  $\Rightarrow$  instructions for the hardware to perform
- Instructions are stored in memory
- Written in machine language
- Made up of ...
  - Command or operator
  - Address of the value to be operated upon, operand
- Stored in binary form

# Programming

- Assembly language
  - Easier to read, understand than list of binary codes
  - Uses mnemonics in place of numeric codes
  - Translation program called an assembler converts mnemonics into binary machine code

# Programming

- High level languages
  - Instructions read like English and algebra
    - Called source code
  - Easier to read/understand than assembly language
  - Must be translated into machine language by compiler
    - Called object code

# Programming

- Text editor creates source code
- Compiler creates object code
- Linker gathers portions of object code from ...
  - Compiler output
  - Libraries of special routines
- ... produces executable code

