Computer Organization and Architecture

Introduction

Chapters 1-2

Architecture & Organization 1

- Architecture is those attributes visible to the programmer
  - Instruction set, number of bits used for data representation, I/O mechanisms, addressing techniques.
  - e.g. Is there a multiply instruction?
- Organization is how features are implemented, typically hidden from the programmer
  - Control signals, interfaces, memory technology.
  - e.g. Is there a hardware multiply unit or is it done by repeated addition?

Architecture & Organization 2

- ⌘ All Intel x86 family share the same basic architecture
- ⌘ The IBM System/370 family share the same basic architecture
- ⌘ This gives code compatibility
  - At least backwards
  - But... increases complexity of each new generation. May be more efficient to start over with a new technology, e.g. RISC vs. CISC
- ⌘ Organization differs between different versions

Structure & Function

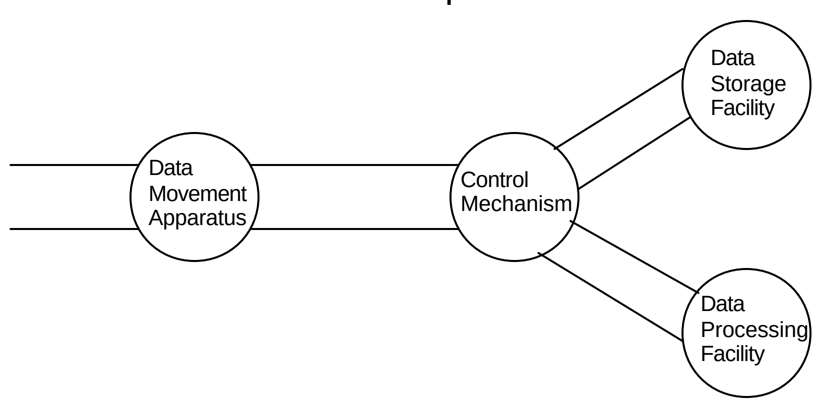
- ⌘ Computers are complex; easier to understand if broken up into hierarchical components. At each level the designer should consider
  - **Structure** : the way in which components relate to each other
  - **Function** : the operation of individual components as part of the structure
- ⌘ Let's look at the computer top-down starting with function.

Function

⌘All computer functions are:
- Data processing
- Data storage
- Data movement
- Control

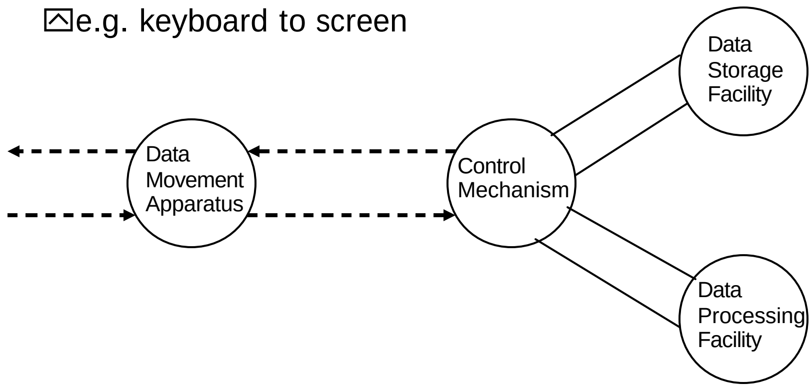
Functional view

⌘Functional view of a computer



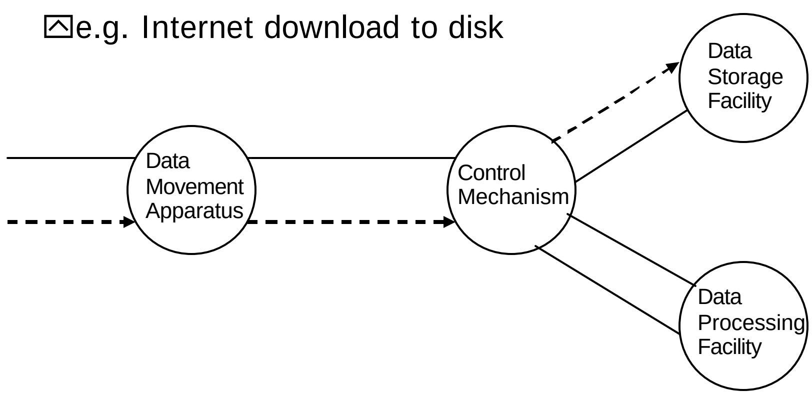
Operations (1)

- Data movement
  - e.g. keyboard to screen



Operations (2)

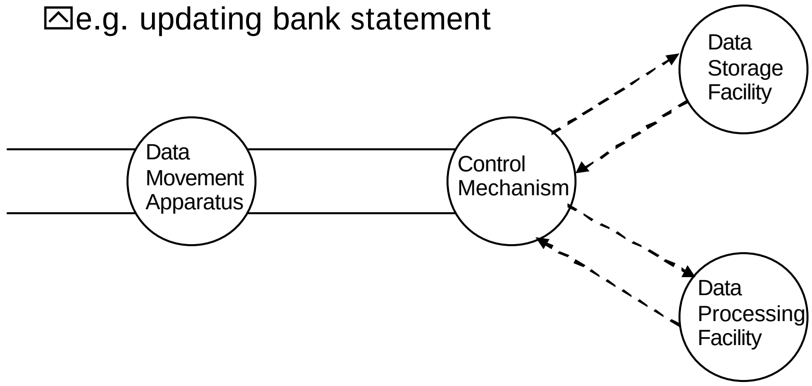
- Storage
  - e.g. Internet download to disk



Operation (3)

⌘Processing from/to storage

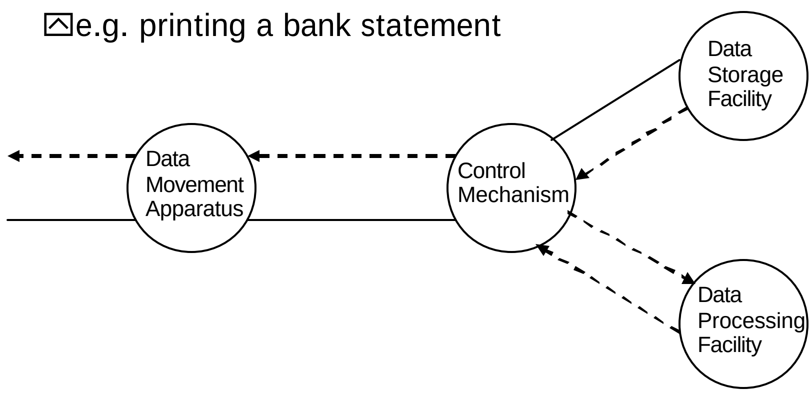
- ◰e.g. updating bank statement



Operation (4)

⌘Processing from storage to I/O

- ◰e.g. printing a bank statement

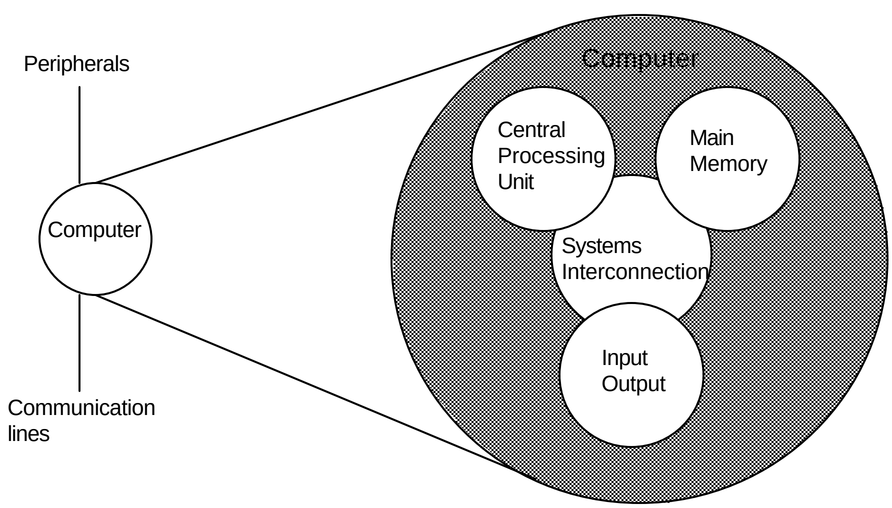


Structure

⌘Major Components of a Computer

- Central Processing Unit (CPU) – Controls the operation of the computer and performs data processing
- Main Memory – Stores data
- Input Output (I/O) – Moves data between the computer and the external environment
- System Interconnect – Some mechanism that provides for communications between the system components, typically a **bus** (set of wires)

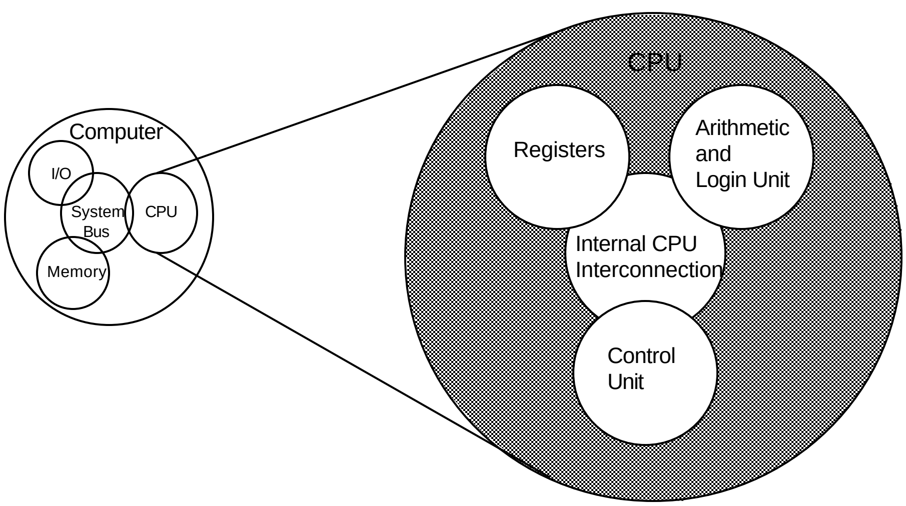
Structure - Top Level



Structure - CPU

⌘Major components of the CPU
- Control Unit (CU) – Controls the operation of the CPU
- Arithmetic and Logic Unit (ALU) – Performs data processing functions, e.g. arithmetic operations
- Registers – Fast storage internal to the CPU, but contents can be copied to/from main memory
- CPU Interconnect – Some mechanism that provides for communication among the control unit, ALU, and registers

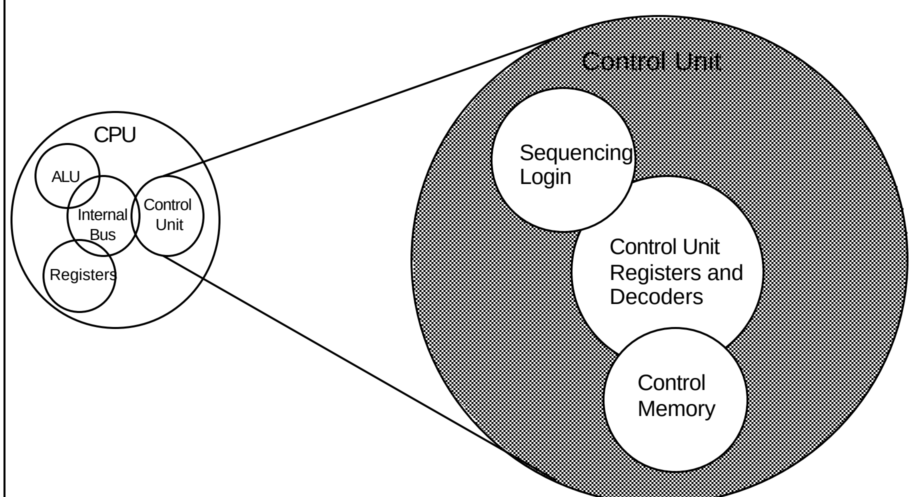
Structure - The CPU



Structure – Inside the CPU

- ⌘The implementation of registers and the ALU we will leave primarily to EE 241
- ⌘We will say a bit about the architecture of the control unit, there are many possible approaches.
  - A common approach is the microprogrammed control unit, where the control unit is in essence itself a miniature computer, where a CPU instruction is implemented via one or more “micro instructions”
    - ☒Sequencing Logic – Controlling the order of events
    - ☒Microprogram Control Unit – Internal controls
    - ☒Microprogram Registers, Memory

Structure – A Microprogrammed Control Unit



Outline of the Stallings Text (1)

- ⌘Computer Evolution and Performance
- ⌘Computer Interconnection Structures
- ⌘Internal Memory
- ⌘External Memory
- ⌘Input/Output
- ⌘Operating Systems Support
- ⌘Computer Arithmetic
- ⌘Instruction Sets

Outline of the Stallings Text (2)

- ⌘CPU Structure and Function
- ⌘Reduced Instruction Set Computers
- ⌘Superscalar Processors
- ⌘Control Unit Operation
- ⌘Microprogrammed Control
- ⌘Multiprocessors and Vector Processing
- ⌘Digital Logic (Appendix)

Computer Evolution and Performance

Better, Faster, Cheaper?

History: ENIAC background

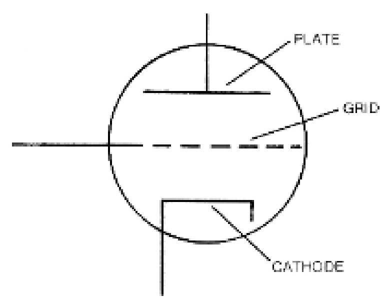
- Electronic Numerical Integrator And Computer
- Eckert and Mauchly
- University of Pennsylvania
- Trajectory tables for weapons, BRL
- Started 1943
- Finished 1946
  - Too late for war effort
- Used until 1955

ENIAC - details

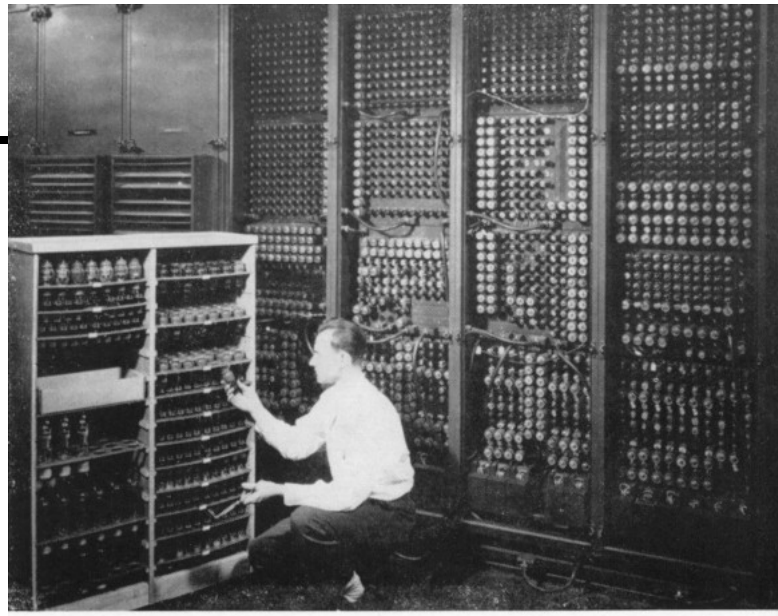
- ⌘Decimal (not binary)
- ⌘20 accumulators of 10 digits (ring of 10 tubes)
- ⌘Programmed manually by switches
- ⌘18,000 vacuum tubes
- ⌘30 tons
- ⌘15,000 square feet
- ⌘140 kW power consumption (about $10/hr today)
- ⌘5,000 additions per second

Vacuum Tubes

vacuum tube from the early 1900's



Grid regulates flow from of electrons from the cathode

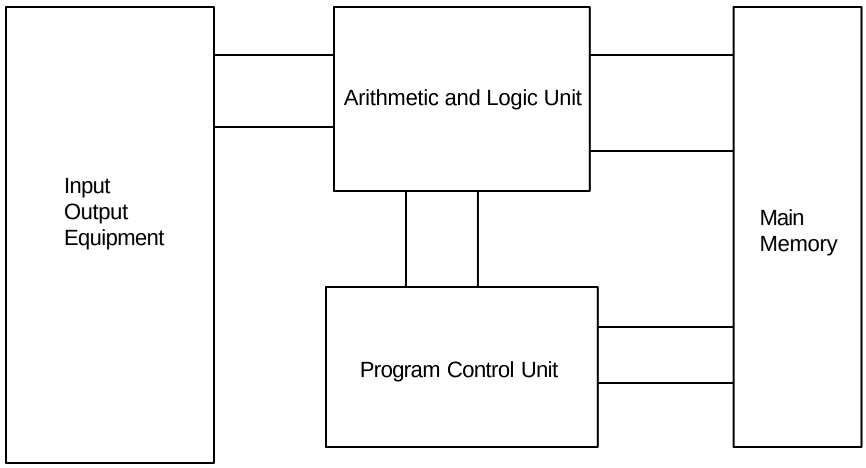


Replacing a bad tube meant checking among ENIAC's 19,000 possibilities.

von Neumann/Turing

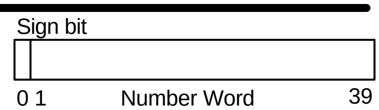
- ENIAC : Very tedious to manually wire programs
- von Neumann architecture:
  - Stored Program concept
  - Main memory storing programs and data
  - ALU operating on binary data
  - Control unit interpreting instructions from memory and executing
  - Input and output equipment operated by control unit
  - Princeton Institute for Advanced Studies
    - IAS
  - Completed 1952

Structure of von Neumann machine

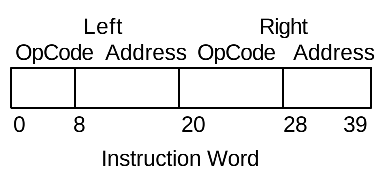


IAS - details

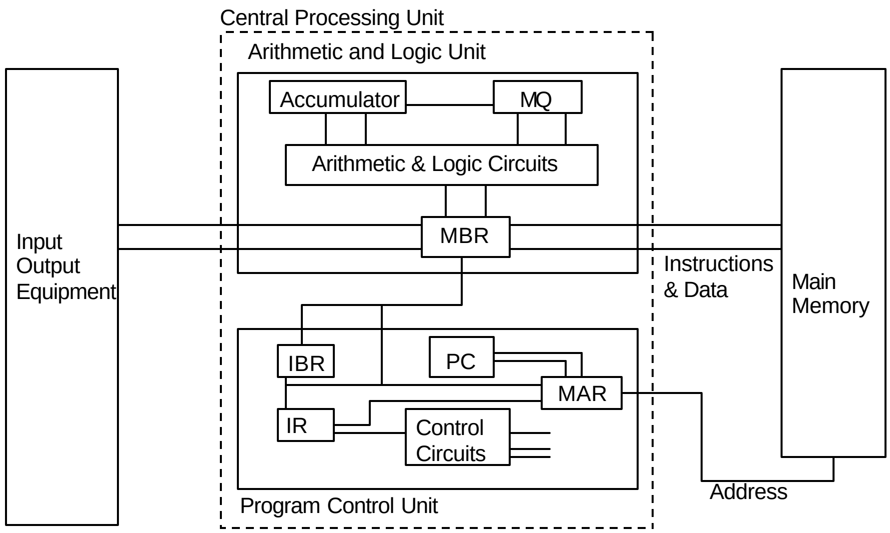
- 1000 x 40 bit words
  - Binary number
  - 2 x 20 bit instructions



- Set of registers (storage in CPU)
  - Memory Buffer Register
  - Memory Address Register
  - Instruction Register
  - Instruction Buffer Register
  - Program Counter
  - Accumulator
  - Multiplier Quotient



Structure of IAS - detail



IAS Instruction Cycle

- ⌘ The IAS repetitively performs the instruction cycle:
  - Fetch
    - Opcode of the next instruction is loaded into the IR
    - Address portion is loaded into the MAR
    - Instruction either taken from the IBR or obtained from memory by loading the PC into the MAR, memory to the MBR, then the MBR to the IBR and the IR
      - To simplify electronics, only one data path from MBR to IR
  - Execute
    - Circuitry interprets the opcode and executes the instruction
    - Moving data, performing an operation in the ALU, etc.
- ⌘ IAS had 21 instructions
  - Data transfer, Unconditional branch, conditional branch, arithmetic, address modification

Commercial Computers

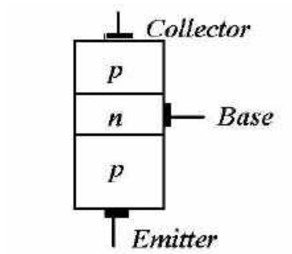
- ⌘1947 - Eckert-Mauchly Computer Corporation
- ⌘UNIVAC I (Universal Automatic Computer)
- ⌘US Bureau of Census 1950 calculations
- ⌘Became part of Sperry-Rand Corporation
- ⌘Late 1950s - UNIVAC II
  - Faster
  - More memory
  - Upward compatible with older machines

IBM

- ⌘Punched-card processing equipment
- ⌘1953 - the 701
  - IBM's first stored program computer
  - Scientific calculations
- ⌘1955 - the 702
  - Business applications
- ⌘Lead to 700/7000 series

Transistors

- ⌘Replaced vacuum tubes
- ⌘Smaller
- ⌘Cheaper
- ⌘Less heat dissipation
- ⌘Solid State device
- ⌘Made from Silicon (Sand)
- ⌘Invented 1947 at Bell Labs
- ⌘Shockley, Brittain, Bardeen



Transistor Based Computers

- ⌘Second generation of machines
- ⌘NCR & RCA produced small transistor machines
- ⌘IBM 7000
- ⌘DEC - 1957
  - Produced PDP-1

IBM 7094

- ⌘Last member of the 7000 series
  - 50 times faster than the 701
    - 1.4 uS vs. 30 uS cycle
  - 32K memory vs. 2K
  - Main memory: Core memory vs. Tubes
  - CPU memory: transistors vs. Tubes
  - 185 vs. 24 opcodes
  - Instruction fetch overlap, reduced another trip to memory (exception are branches)
  - Data channels, independent I/O module for devices

3rd Generation: Integrated Circuits

- ⌘Self-contained transistor is a discrete component
  - Big, manufactured separately, expensive, hot when you have thousands of them
- ⌘Integrated Circuits
  - Transistors “etched” into a substrate, bundled together instead of discrete components
  - Allowed thousands of transistors to be packaged together efficiently

Microelectronics

- Literally - “small electronics”
- A computer is made up of gates, memory cells and interconnections
- These can be manufactured on a semiconductor, e.g. silicon wafer
  - Thin wafer divided into chips
  - Each chip consists of many gates/memory cells
  - Chip packaged together with pins, assembled on a printed circuit board

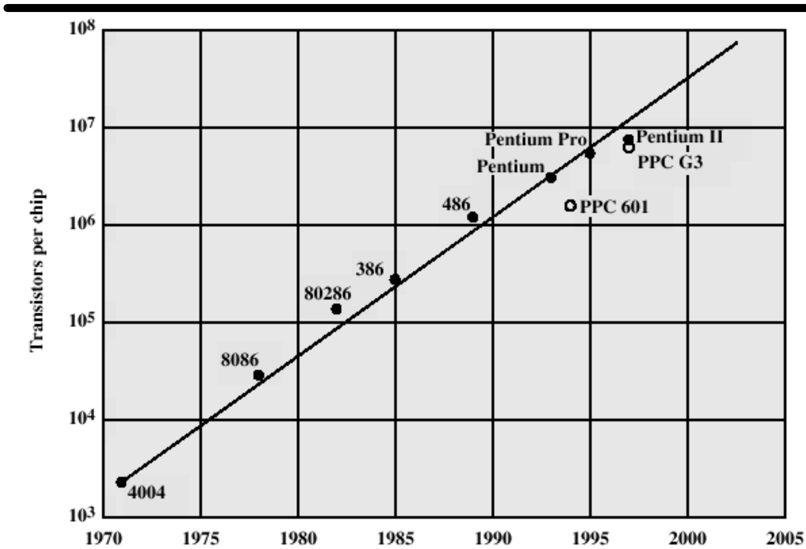
Generations of Computer

- Vacuum tube - 1946-1957
- Transistor - 1958-1964
- Small scale integration - 1965 on
  - Up to 100 devices on a chip
- Medium scale integration - to 1971
  - 100-3,000 devices on a chip
- Large scale integration - 1971-1977
  - 3,000 - 100,000 devices on a chip
- Very large scale integration - 1978 to date
  - 100,000 - 100,000,000 devices on a chip
  - Pentium IV has about 40 million transistors
- Ultra large scale integration
  - Over 100,000,000 devices on a chip (vague term)

Moore’s Law

- ⌘ Increased density of components on chip
- ⌘ Gordon Moore : co-founder of Intel
- ⌘ Number of transistors on a chip will double every year
- ⌘ Since 1970’s development has slowed a little
  - Number of transistors doubles every 18 months
- ⌘ Cost of a chip has remained almost unchanged
- ⌘ Higher packing density means shorter electrical paths, giving higher performance
- ⌘ Smaller size gives increased flexibility
- ⌘ Reduced power and cooling requirements
- ⌘ Fewer interconnections increases reliability

Growth in CPU Transistor Count



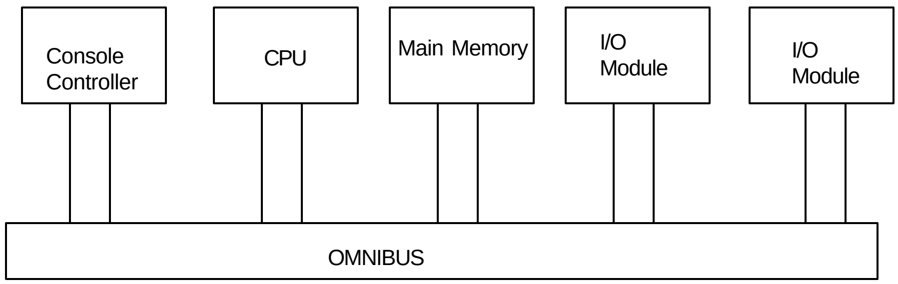
IBM 360 series

- ⌘ 1964
- ⌘ Replaced (& not compatible with) 7000 series
  - Reason: Needed to break out of constraints of the 7000 architecture
- ⌘ First planned “family” of computers
  - Similar or identical instruction sets
  - Similar or identical O/S
  - Increasing speed
  - Increasing number of I/O ports (i.e. more terminals)
  - Increased memory size
  - Increased cost (not always the case today!)
- ⌘ Multiplexed switch structure

DEC PDP-8

- ⌘ 1964
- ⌘ First minicomputer (after miniskirt!)
- ⌘ Did not need air conditioned room
- ⌘ Small enough to sit on a lab bench
- ⌘ $16,000
  - $100k+ for IBM 360
- ⌘ Embedded applications & OEM
- ⌘ BUS STRUCTURE

DEC - PDP-8 Bus Structure



96 separate signal paths to carry control, address, data signals
Highly flexible, allowed modules to be plugged in for different configurations

Other Innovations - Semiconductor Memory

- 1970
- Fairchild
- Size of a single core
  - i.e. 1 bit of magnetic core storage
  - Held 256 bits
- Non-destructive read
- Much faster than core
- Capacity approximately doubles each year

Intel

- ⌘ 1971 - 4004
  - First microprocessor
  - All CPU components on a single chip
  - 4 bit
- ⌘ Followed in 1972 by 8008
  - 8 bit
  - Both designed for specific applications
- ⌘ 1974 - 8080
  - Intel's first general purpose microprocessor
- ⌘ Evolution: 8086, 8088, 80286, 80386, 80486, Pentium Pentium Pro, Pentium II, Pentium III, Pentium IV, Itanium

Speeding it up

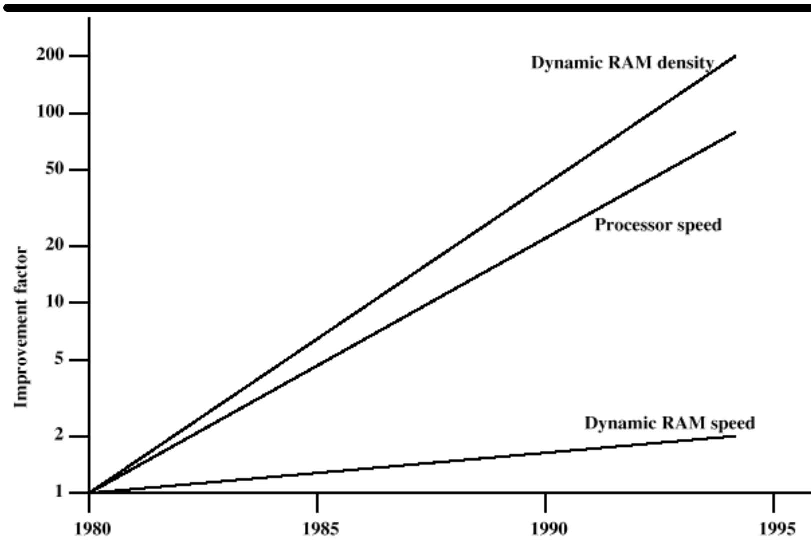
- ⌘ Smaller manufacturing process (0.11 micron)
- ⌘ Pipelining
- ⌘ On board cache
- ⌘ On board L1 & L2 cache
- ⌘ Branch prediction
- ⌘ Data flow analysis
- ⌘ Speculative execution
- ⌘ Parallel execution

Performance Mismatch

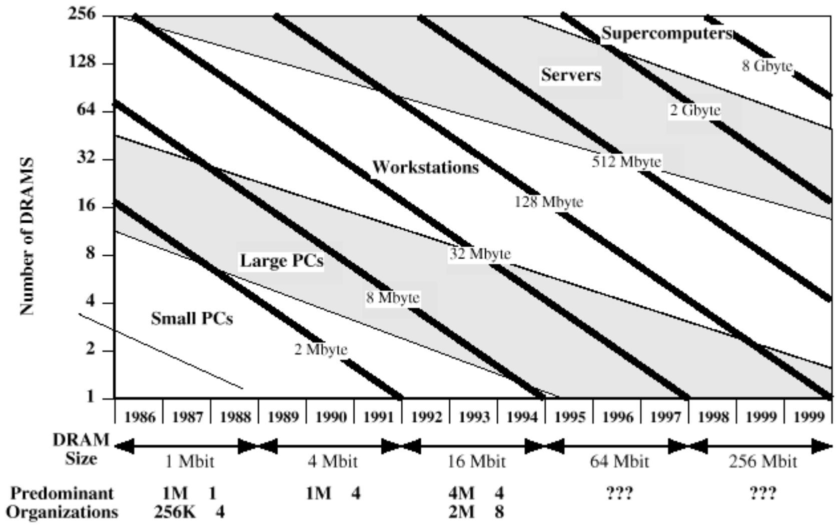
- ⌘Processor speed increased
- ⌘Memory capacity increased
- ⌘Memory speed lags behind processor speed

- ⌘Common memory chip technology
  - DRAM = Dynamic Random Access Memory

DRAM and Processor Characteristics



Trends in DRAM use



Solutions

- ⌘ Increase number of bits retrieved at one time
  - Make DRAM “wider” rather than “deeper”
- ⌘ Change DRAM interface
  - Cache
- ⌘ Reduce frequency of memory access
  - More complex cache and cache on chip
- ⌘ Increase interconnection bandwidth
  - High speed buses
  - Hierarchy of buses

- ⌘ Similar problems with I/O devices, e.g. graphics, network
- ⌘ Need balance in computer design