Concepts Introduced in Chapter 1

- introduction
- classes of computers
- trends
- measures of computers
- principles of computer design

Computer Architecture

- Computer architecture covers three aspects of computer design.
  - instruction set architecture (ISA) - programmer visible instruction set that serves as the boundary between the software and the hardware
  - organization (microarchitecture) - high level aspects of the design implementation, which includes the memory system and CPU
  - hardware - low level aspects of the design implementation, which includes the logic design, packaging technology

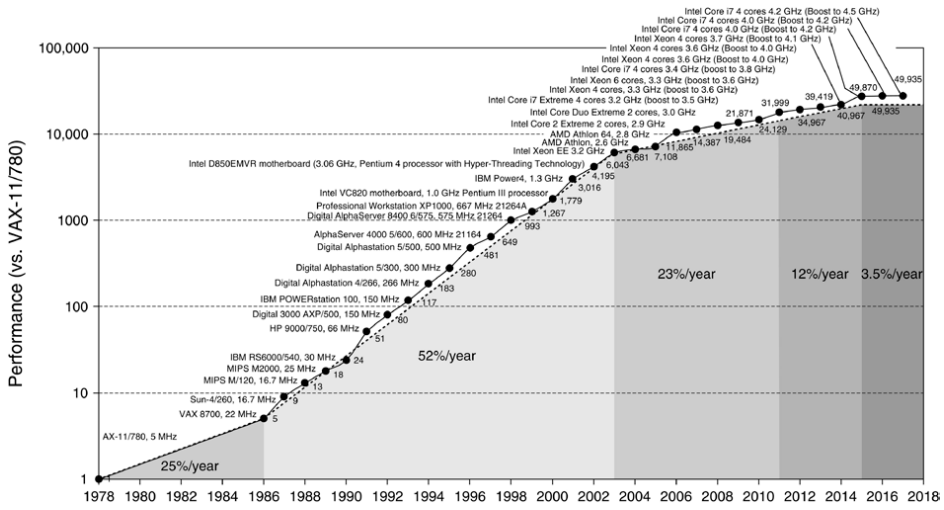
Factors That Allow Architectures to Be Quickly Introduced

- little assembly language programming
- standard operating systems

Computer Architect Goals

- low cost
- high performance
- low power

Growth in Processor Performance Since the Late 1970s



Effects of Dramatic Growth in Processor Performance

- Enhanced capability available to users.
- Led to new classes of computers.
- Led to dominance of microprocessor based computers across the range of computer design.
- Allows programmers to trade performance for productivity.
- Nature of applications are also changing.

Classes of Computers

- personal mobile devices (PMDs)
  - wireless devices with multimedia user interfaces
  - smart phones and tablets
  - emphasis on cost and energy efficiency
- desktops
  - personal computers intended for use at a stationary location
  - notebooks and workstations
  - emphasis on cost and performance
- servers
  - assessed by other computers to provide computation and/or storage
  - emphasis on availability, scalability, and throughput

Classes of Computers (cont.)

- clusters/warehouse-scale computers (WSCs)
  - collections of computers connected by a network to act as a single powerful computer
  - emphasis on cost, performance, and power
  - scalability and availability handled through the network
- embedded computers
  - computers contained in other devices
  - cannot run externally developed software (unlike PMDs)
  - emphasis on cost
  - have more diverse ISAs

Five Mainstream Computing Classes

Feature	Personal mobile device (PMD)	Desktop	Server	Clusters/warehouse-scale computer	Internet of things/ embedded
Price of system	$100–$1000	$300–$2500	$5000–$10,000,000	$100,000–$200,000,000	$10–$100,000
Price of microprocessor	$10–$100	$50–$500	$200–$2000	$50–$250	$0.01–$100
Critical system design issues	Cost, energy, media performance, responsiveness	Price-performance, energy, graphics performance	Throughput, availability, scalability, energy	Price-performance, throughput, energy proportionality	Price, energy, application-specific performance

Important Functional Architectural Requirements

Functional requirements	Typical features required or supported
Application area	*Target of computer*
Personal mobile device	Real-time performance for a range of tasks, including interactive performance for graphics, video, and audio; energy efficiency (Chapters 2–5 and 7; Appendix A)
General-purpose desktop	Balanced performance for a range of tasks, including interactive performance for graphics, video, and audio (Chapters 2–5; Appendix A)
Servers	Support for databases and transaction processing; enhancements for reliability and availability; support for scalability (Chapters 2, 5, and 7; Appendices A, D, and F)
Clusters/warehouse-scale computers	Throughput performance for many independent tasks; error correction for memory; energy proportionality (Chapters 2, 6, and 7; Appendix F)
Internet of things/embedded computing	Often requires special support for graphics or video (or other application-specific extension); power limitations and power control may be required; real-time constraints (Chapters 2, 3, 5, and 7; Appendices A and E)
Level of software compatibility	*Determines amount of existing software for computer*
At programming language	Most flexible for designer; need new compiler (Chapters 3, 5, and 7; Appendix A)
Object code or binary compatible	Instruction set architecture is completely defined—little flexibility—but no investment needed in software or porting programs (Appendix A)
Operating system requirements	*Necessary features to support chosen OS* (Chapter 2; Appendix B)
Size of address space	Very important feature (Chapter 2); may limit applications
Memory management	Required for modern OS; may be paged or segmented (Chapter 2)
Protection	Different OS and application needs: page versus segment; virtual machines (Chapter 2)
Standards	*Certain standards may be required by marketplace*
Floating point	Format and arithmetic: IEEE 754 standard (Appendix J), special arithmetic for graphics or signal processing
I/O interfaces	For I/O devices: Serial ATA, Serial Attached SCSI, PCI Express (Appendices D and F)
Operating systems	UNIX, Windows, Linux, CISCO IOS
Networks	Support required for different networks: Ethernet, Infiniband (Appendix F)
Programming languages	Languages (ANSI C, C++, Java, Fortran) affect instruction set (Appendix A)

Parallelism in Applications

- data-level parallelism (DLP) - data items can be operated on at the same time
- task-level parallelism (TLP) - tasks can be created to independently operate

Ways Hardware Exploits Application Parallelism

- instruction-level parallelism (ILP) - exploits DLP by processing more than one instruction at a time
- vector units (VUs), graphic processor units (GPUs), and multimedia instruction sets - exploits DLP by applying a single operation to a set of data items in parallel
- thread-level parallelism (TLP) - exploits DLP or TLP through multiple flows of execution in tightly-coupled hardware
- request-level parallelism (RLP) - exploits TLP among largely decoupled tasks

Flynn Taxonomy

- single instruction stream, single data stream (SISD) - standard sequential computer, but includes support for ILP
- single instruction stream, multiple data stream (SIMD) - single instruction is applied to multiple items of data in parallel, includes VUs and GPUs
- multiple instruction stream, single data stream (MISD) - no commercial multiprocessor of this type has been built
- multiple instruction stream, multiple data stream (MIMD) - exploits TLP in both tightly and loosely coupled processors

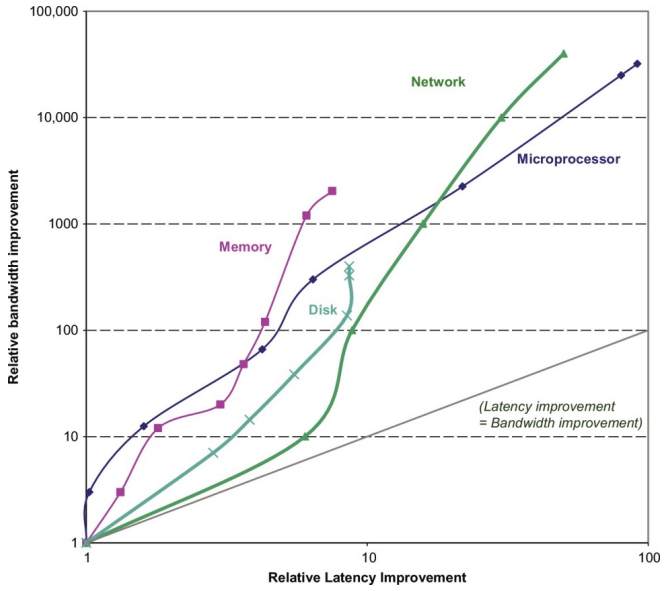
Trends in Implementation Technology

- Transistor count on a chip has historically increased by about 40% to 55% a year, or doubling every 18 to 24 months (Moore's law). In recent years this rate of increase has slowed.
- DRAM capacity per chip is increasing by about 25% to 40% a year, doubling every two to three years. In recent years this rate of increase has also slowed.
- Flash capacity per chip is increasing by about 50% to 60% a year, doubling every two years. Flash memory is currently 8 to 10 times cheaper per byte than DRAM.
- Disk density has been increasing about 40% per year, doubling every three years. But recently the increase has slowed to less than 5% a year. Disks per byte are currently 8 to 10 times cheaper than flash.

Performance Trends: Bandwidth over Latency

- Bandwidth (throughput) is the total amount of work done in a given period of time.
- Latency (response time) is the time between the start and completion of an event.
- Bandwidth typically improves faster than latency for most memory technologies.

Log-Log Plot of Bandwidth and Latency Milestones



Performance Milestones over the Last Four Decades

Microprocessor	16-Bit address/ bus, microcoded	32-Bit address/ bus, microcoded	5-Stage pipeline, on-chip I & D caches, FPU	2-Way superscalar, 64-bit bus	Out-of-order 3-way superscalar	Out-of-order superpipelined, on-chip L2 cache	Multicore OOO 4-way on chip L3 cache, Turbo
Product	Intel 80286	Intel 80386	Intel 80486	Intel Pentium	Intel Pentium Pro	Intel Pentium 4	Intel Core i7
Year	1982	1985	1989	1993	1997	2001	2015
Die size (mm^2)	47	43	81	90	308	217	122
Transistors	134,000	275,000	1,200,000	3,100,000	5,500,000	42,000,000	1,750,000,000
Processors/chip	1	1	1	1	1	1	4
Pins	68	132	168	273	387	423	1400
Latency (clocks)	6	5	5	5	10	22	14
Bus width (bits)	16	32	32	64	64	64	196
Clock rate (MHz)	12.5	16	25	66	200	1500	4000
Bandwidth (MIPS)	2	6	25	132	600	4500	64,000
Latency (ns)	320	313	200	76	50	15	4
Memory module	DRAM	Page mode DRAM	Fast page mode DRAM	Fast page mode DRAM	Synchronous DRAM	Double data rate SDRAM	DDR4 SDRAM
Module width (bits)	16	16	32	64	64	64	64
Year	1980	1983	1986	1993	1997	2000	2016
Mbits/DRAM chip	0.06	0.25	1	16	64	256	4096
Die size (mm^2)	35	45	70	130	170	204	50
Pins/DRAM chip	16	16	18	20	54	66	134
Bandwidth (MBytes/s)	13	40	160	267	640	1600	27,000
Latency (ns)	225	170	125	75	62	52	30
Local area network	Ethernet	Fast Ethernet	Gigabit Ethernet	10 Gigabit Ethernet	100 Gigabit Ethernet	400 Gigabit Ethernet	
IEEE standard	802.3	803.3u	802.3ab	802.3ac	802.3ba	802.3bs	
Year	1978	1995	1999	2003	2010	2017	
Bandwidth (Mbits/seconds)	10	100	1000	10,000	100,000	400,000	
Latency (µs)	3000	500	340	190	100	60	
Hard disk	3600 RPM	5400 RPM	7200 RPM	10,000 RPM	15,000 RPM	15,000 RPM	
Product	CDC WrenI 94145-36	Seagate ST41600	Seagate ST15150	Seagate ST39102	Seagate ST373453	Seagate ST600MX0062	
Year	1983	1990	1994	1998	2003	2016	
Capacity (GB)	0.03	1.4	4.3	9.1	73.4	600	
Disk form factor	5.25 in.	5.25 in.	3.5 in.	3.5 in.	3.5 in.	3.5 in.	
Media diameter	5.25 in.	5.25 in.	3.5 in.	3.0 in.	2.5 in.	2.5 in.	
Interface	ST-412	SCSI	SCSI	SCSI	SCSI	SAS	
Bandwidth (MBytes/s)	0.6	4	9	24	86	250	
Latency (ms)	48.3	17.1	12.7	8.8	5.7	3.6	

Need for Energy Efficient Processors

- Extend battery life for mobile systems.
- Reduce heat dissipation for general-purpose processors.
- Energy cost for computing is increasing.

Energy and Power within a Microprocessor

- Energy is the capacity to do work and power is the rate at which work is done. Energy is power integrated over time and power is the rate at which energy is transmitted.
- Dynamic power comes from switching transistors.

  $Power \propto 1/2\ X\ Capacitive\ load\ X\ Voltage^2\ X\ Frequency\ switched$

- Static power mainly comes from leakage which is proportional to the number of devices being powered.

Techniques to Improve Energy Efficiency

- Turn off the clock (clock gating) for inactive modules (do nothing efficiently).
- Use a lower clock frequency during periods of low activity (called dynamic voltage-frequency scaling or DVFS).
- Use a low power mode for memory and storage when not being accessed (sometimes called a drowsy state).
- Completely turn off power to subsets of chip when not used (power gating).

Manufacturing Computer Component Terms

- A wafer is a slice of a silicon crystal ingot.
- A die or chip is a portion of the wafer that is an independent component.
- Yield is the percentage of manufactured devices that passes the testing procedures.
- Designers have been including redundancy as a means to raise yield.

N Times Faster

- Often people state that a machine X is n times faster than a machine Y. What does this mean?

$$\frac{Performance_X}{Performance_Y} = n = \frac{Execution\_Time_Y}{Execution\_Time_X}$$

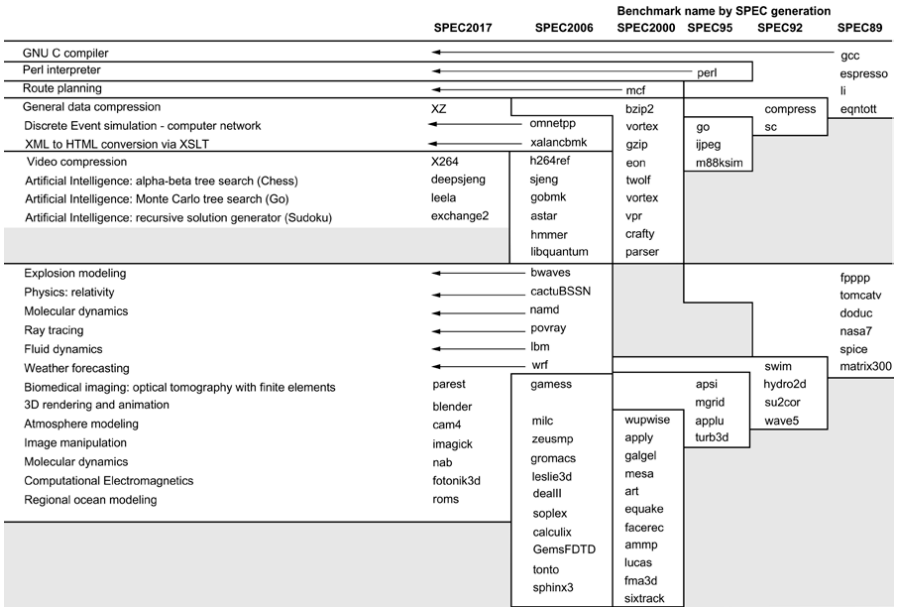
Measures of Clock Speed

- clock periods
  - millisecond (ms) - 10^{-3} of a second
  - microsecond (μs) - 10^{-6} of a second
  - nanosecond (ns) - 10^{-9} of a second
  - picosecond (ps) - 10^{-12} of a second
  - femtosecond (fs) - 10^{-15} of a second
- clock rates
  - kilohertz (KHz) - 10^3 cycles per second
  - megahertz (MHz) - 10^6 cycles per second
  - gigahertz (GHz) - 10^9 cycles per second
  - terahertz (THz) - 10^{12} cycles per second
  - petahertz (PHz) - 10^{15} cycles per second

Programs Used to Evaluate Machine Performance

- kernels - an extraction of small, key pieces from popular real programs (Livermore loops, Linpack, EEMBC)
- toy benchmarks - small constructed programs that produce a known result (puzzle, sieve, quicksort, arraymerge, matmult)
- synthetic benchmarks - attempts to match the frequency of operations and operands from real programs (dhrystone, whetstone)
- real applications - application programs that are actually used (SPEC)

Evolution of the SPEC Benchmarks over Time



Classes of Measurement Techniques

- instruction set interpretation
- instrumented execution
- hardware performance counters

Locality of Reference

- Programs tend to reuse data and instructions they have recently accessed.
- Temporal Locality - recently accessed items are likely to be accessed in the near future
- Spatial Locality - items whose addresses are near one another tend to be referenced close together in time
- Locality of reference is extensively exploited in memory hierarchies.

Memory Hierarchy

- Memory hierarchies are useful due to locality of reference.
- Smaller memories are faster, require less energy to access, and are more expensive per byte.
- Larger memories are slower, require more energy to access, and are less expensive per byte.
- A memory hierarchy is cost effective since most references are captured by the fastest memory and most instructions and data are retained in the least expensive memory.
- A memory hierarchy also reduces energy usage since smaller structures require less energy to access.

Amdahl's Law

- Amdahl's Law states that the performance improvement to be gained from using some faster mode of execution is limited by the fraction of the time the faster mode can be used.
- Amdahl's Law depends on two factors:
  - The fraction of the time the enhancement can be exploited.
  - The improvement gained by the enhancement while it is exploited.

$$execution\_time_{new} = execution\_time_{old} * (1 - fraction_{enhanced} + \frac{fraction_{enhanced}}{speedup_{enhanced}})$$

$$speedup_{overall} = \frac{execution\_time_{old}}{execution\_time_{new}} = \frac{1}{(1 - fraction_{enhanced}) + \frac{fraction_{enhanced}}{speedup_{enhanced}}}$$

CPU Time

- CPU time ignores I/O and the time for executing other processes.

$$CPU\_time = CPU\_clock\_cycles * clock\_cycle\_time = \frac{CPU\_clock\_cycles}{clock\_rate}$$

$$CPI = \frac{CPU\_clock\_cycles}{instruction\_count}$$

$$CPU\_time = Instruction\_count * CPI * clock\_cycle\_time$$

Fallacies and Pitfalls

- Fallacy: a commonly held misbelief
  - Multiple microprocessors on a chip are a silver bullet.
  - Peak performance tracks observed performance.
- Pitfall: an easily made mistake
  - Falling prey to Amdahl's law.
  - A single point of failure.