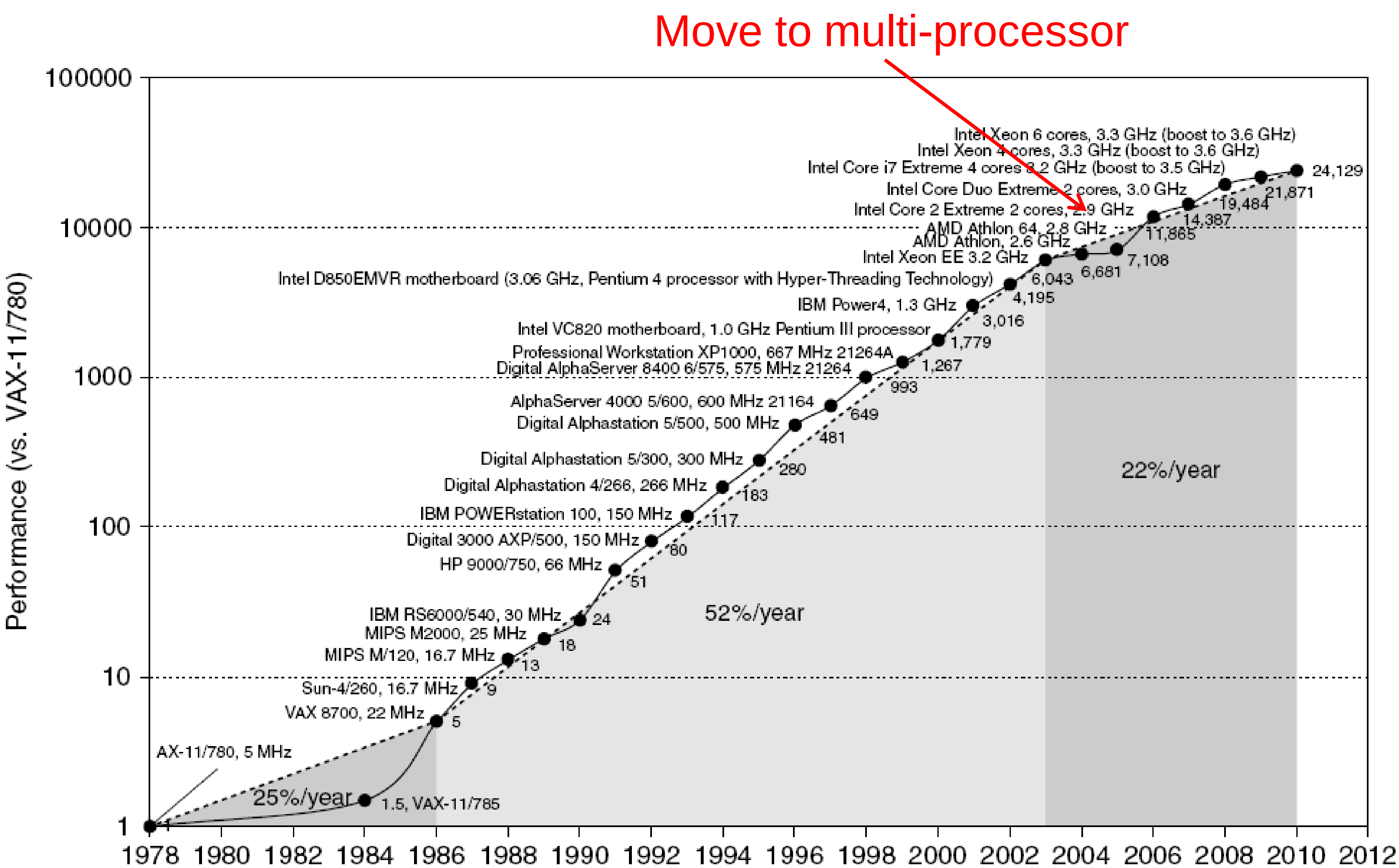
FUNDAMENTALS OF QUANTITATIVE DESIGN

Contents and objective

- Getting familiar with terms & concepts
- Overall trends
- Figure of merit (metrics)
- Computer design principles

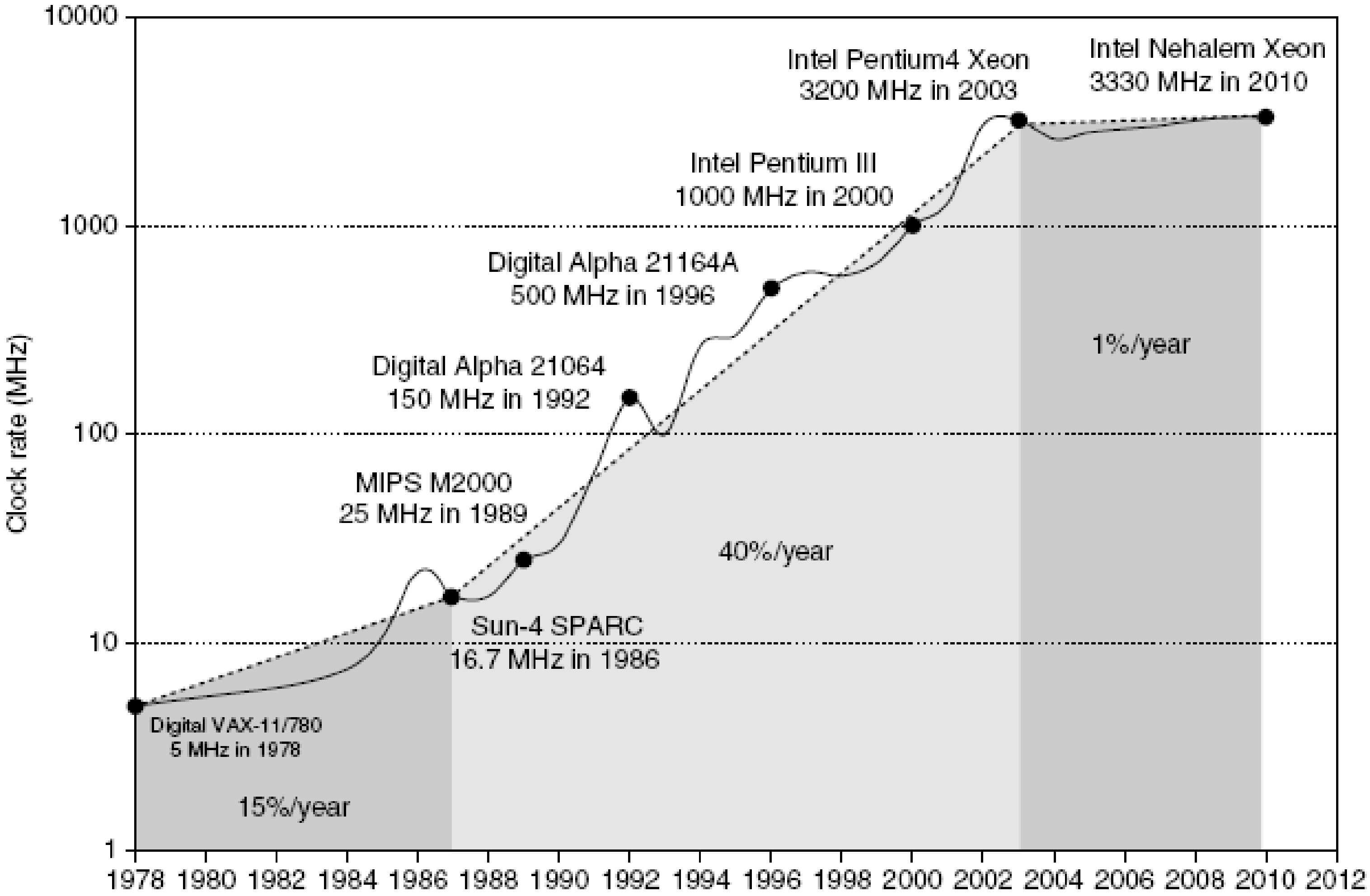
MicroProcessor Performance



Points to Note

- The 52% growth per year is because of faster clock speeds and architectural innovations (led to 25x higher speed)
- Clock speed increases have dropped to 1% per year in recent years
- The 22% growth includes the parallelization from multiple cores
- Moore's Law: transistors on a chip double every 18-24 months

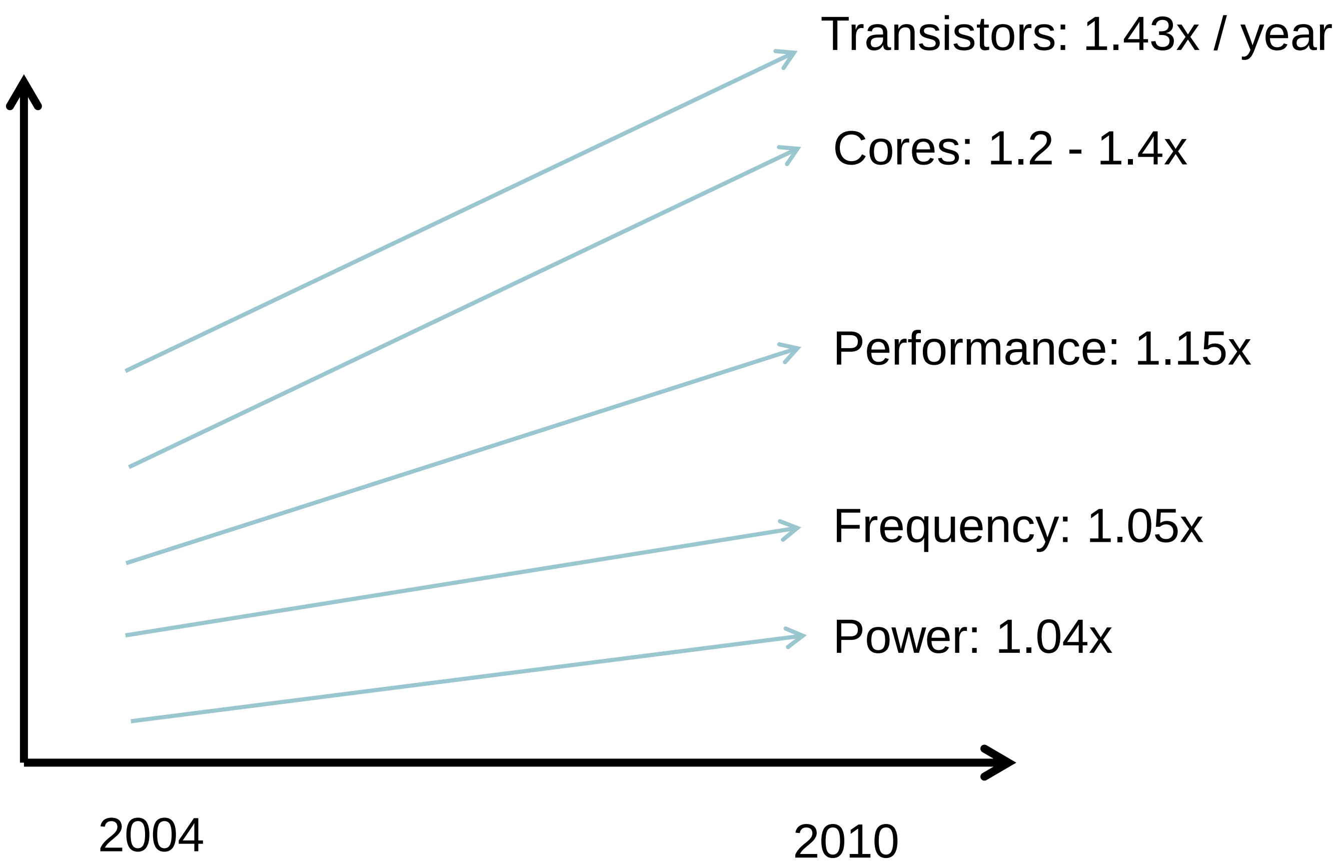
Clock Speed Increases



5

Source: H&P textbook

Recent Microprocessor Trends



Source: Micron University Symp.

Classes of Computers

- Personal Mobile Device (PMD)
  - e.g. start phones, tablet computers
  - Emphasis on energy efficiency and real-time
- Desktop Computing
  - Emphasis on price-performance
- Servers
  - Emphasis on availability, scalability, throughput
- Clusters / Warehouse Scale Computers
  - Emphasis on availability and price-performance
  - Sub-class: Supercomputers, emphasis: floating-point performance and fast internal networks
- Embedded Computers
  - Emphasis: price

Choosing Programs to Evaluate Perf.

- Toy benchmarks
  - e.g., quicksort, puzzle
  - No one really runs. Scary fact: used to prove the value of RISC in early 80’s
- Synthetic benchmarks
  - Attempt to match average frequencies of operations and operands in real workloads.
  - e.g., Whetstone, Dhrystone
  - Often slightly more complex than kernels; But do not represent real programs
- Kernels
  - Most frequently executed pieces of real programs
  - Good for focusing on individual features not big picture
  - Tend to over-emphasize target feature
- Real programs
  - e.g., gcc, spice, SPEC2006 (standard performance evaluation corporation), TPCC, TPCD, PARSEC, SPLASH

Defining Computer Architecture

- "Old" view of computer architecture:
  - Instruction Set Architecture (ISA) design
  - i.e. decisions regarding:
    - registers, memory addressing, addressing modes, instruction operands, available operations, control flow instructions, instruction encoding
- "Real" computer architecture:
  - Specific requirements of the target machine
  - Design to maximize performance within constraints: cost, power, and availability
  - Includes ISA, microarchitecture, hardware

METRICS

Transistor dimension (Area)

- Feature size
  - Also called geometry, process node
  - Minimum size of transistor or wire in x or y dimension
  - 10 microns in 1971 to .032 microns in 2011, .016 microns in 2016
  - Leads to chip-miniaturization
  - Allows fitting more transistors on a chip

Throughput and Latency

- Bandwidth or throughput
  - Total work done in a given time
  - 10,000-25,000X improvement for processors
  - 300-1200X improvement for memory and disks
- Latency or response time
  - Time between start and completion of an event
  - 30-80X improvement for processors
  - 6-8X improvement for memory and disks

Static Power

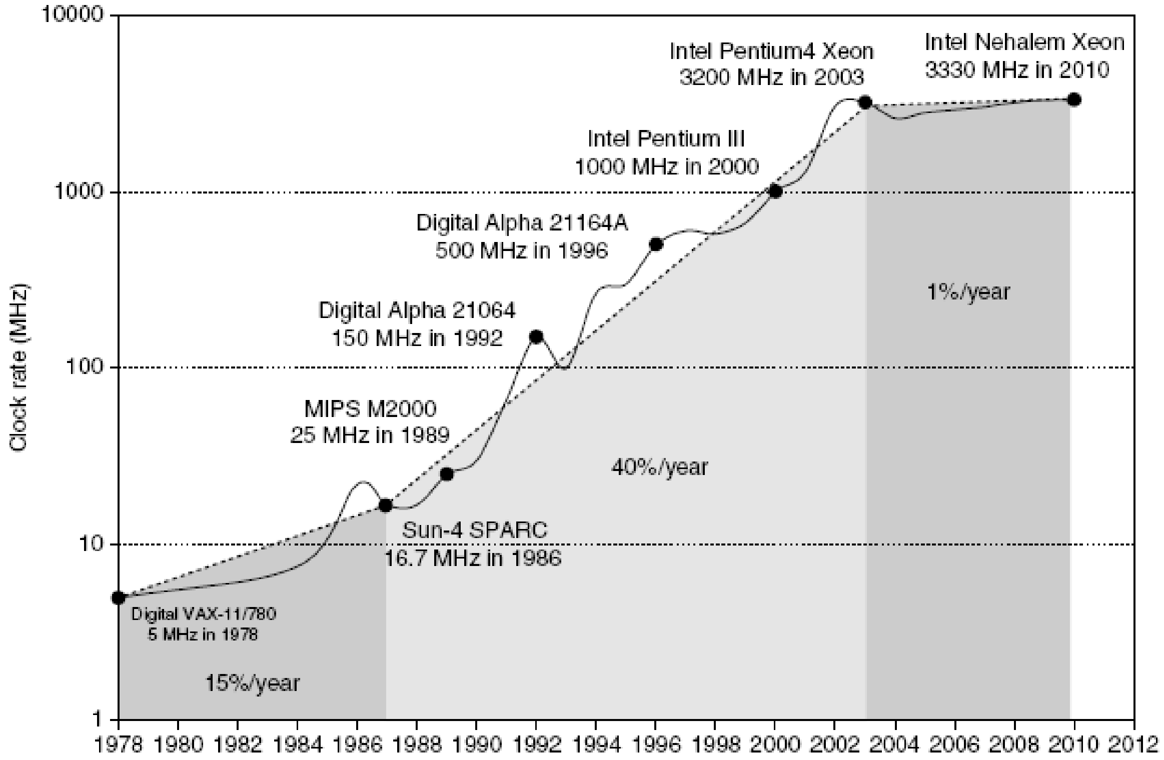
- Power = static + dynamic power
- Static power consumption
  - $\text{Current}_{\text{static}}$ x Voltage
  - Scales with number of transistors

Dynamic Energy and Power

- Dynamic energy
  - Transistor switch from 0 -> 1 or 1 -> 0
  - ½ x Capacitive load x Voltage^2
- Dynamic power
  - ½ x Capacitive load x Voltage^2 x Frequency switched
- Reducing clock rate reduces power, not energy

Power

- Intel 80386 consumed ~ 2 W
- 3.3 GHz Intel Core i7 consumes 130 W
- Heat must be dissipated from 1.5 x 1.5 cm chip
- This is the limit of what can be cooled by air



Power Vs. Energy

- Energy is the ultimate metric: it tells us the true “cost” of performing a fixed task
- Power (energy/time) poses constraints; can only work fast enough to max out the power delivery or cooling solution
- If processor A consumes 1.2x the power of processor B, but finishes the task in 30% less time, its relative energy is 1.2 X 0.7 = 0.84; Proc-A is better, assuming that 1.2x power can be supported by the system

Measuring Performance

- Typical performance metrics:
  - Response time
  - Throughput
- Speedup of X relative to Y
  - Execution time_Y / Execution time_X
- Execution time
  - Wall clock time: includes all system overheads
  - CPU time: only computation time

Defining Reliability and Availability

- A system toggles between
  - Service accomplishment: service matches specifications
  - Service interruption: services deviates from specs
- The toggle is caused by *failures* and *restorations*
- Reliability measures continuous service accomplishment and is usually expressed as mean time to failure (MTTF)
- Availability measures fraction of time that service matches specifications, expressed as MTTF / (MTTF + MTTR)