CE 431
Parallel Computer Architecture
Spring 2019

Case Study:
Intel's Core i7 microarchitecture

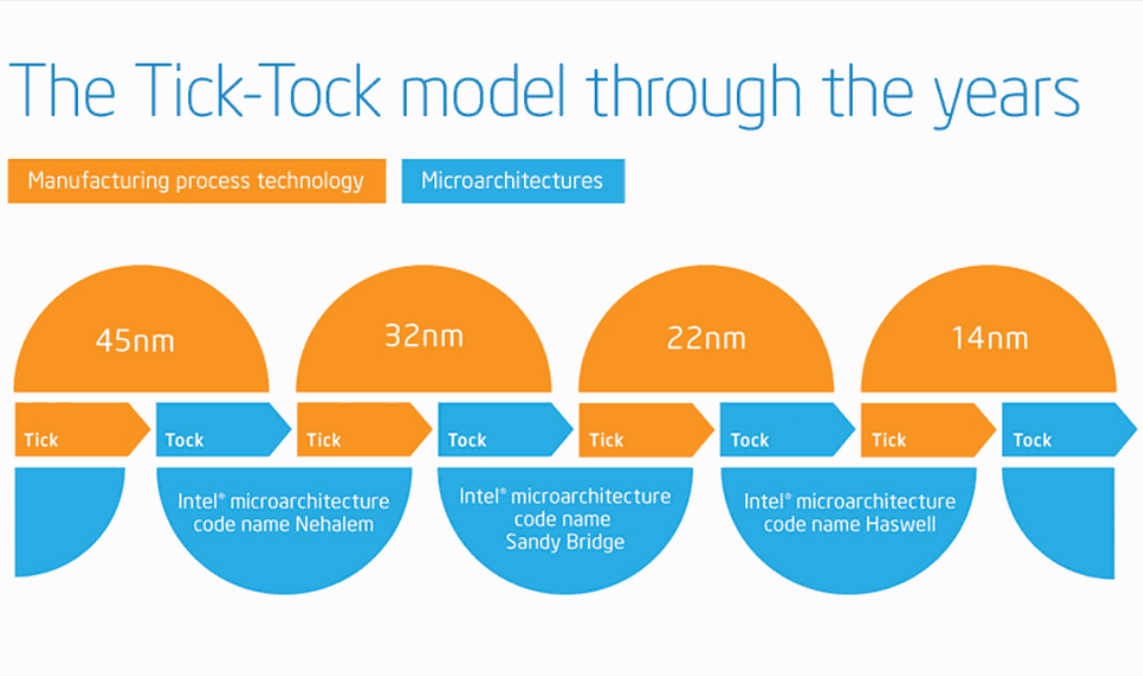
Nikos Bellas

Electrical and Computer Engineering Department
University of Thessaly

Outline

- Introduction to Intel's architecture
  - 80x86 ISA
- Intel Nehalem microarchitecture and Core i7 processor
  - Pipeline
  - Memory subsystem
  - Multi-core
- Power management

Intel's Tick-Tock Development Model

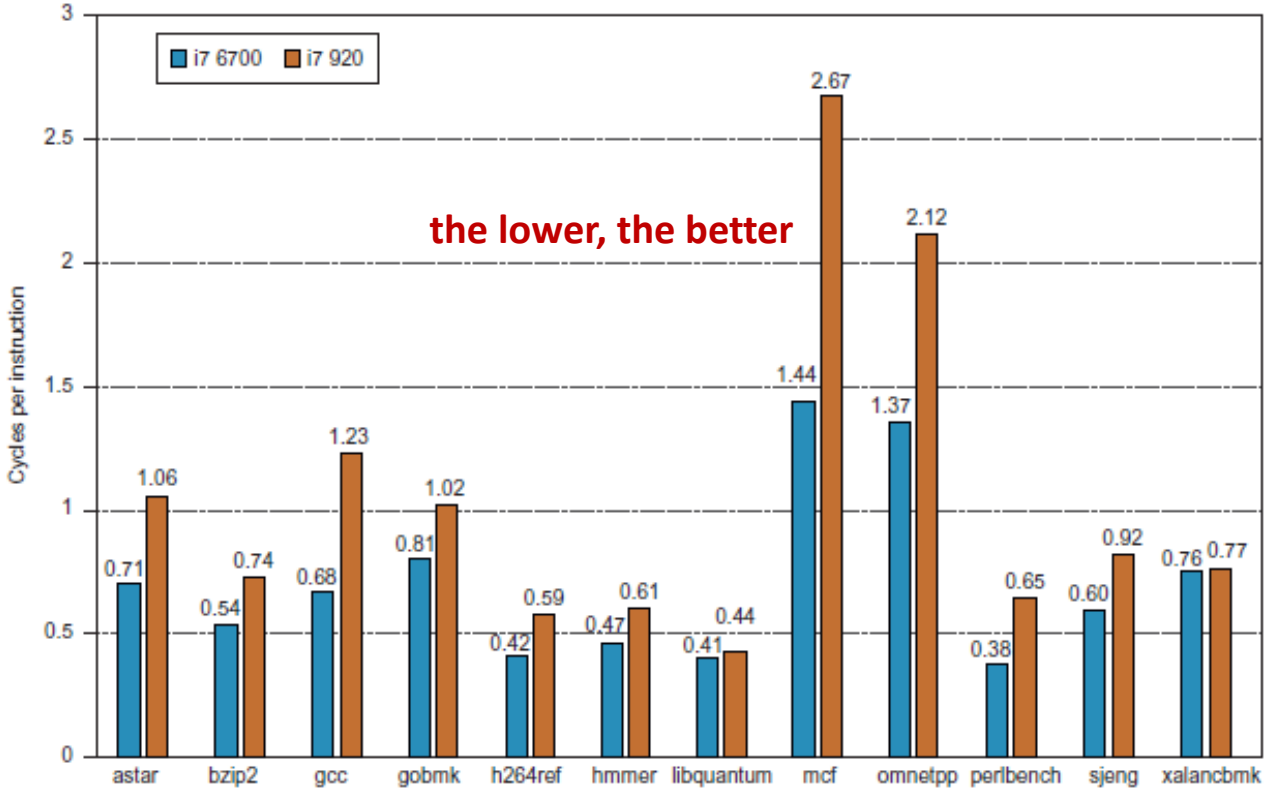


- **2017-2016 (TICK-TOCK)**
  - TICK is process shrink, same microarchitecture
  - TOCK is a new microarchitecture, same process
  - There is a TICK or TOCK every 12-18 months
- **2016 - (Process, Architecture, Optimization)**
  - Similar to TICK TOCK but with an additional Optimization Phase

Some Intel terminology first

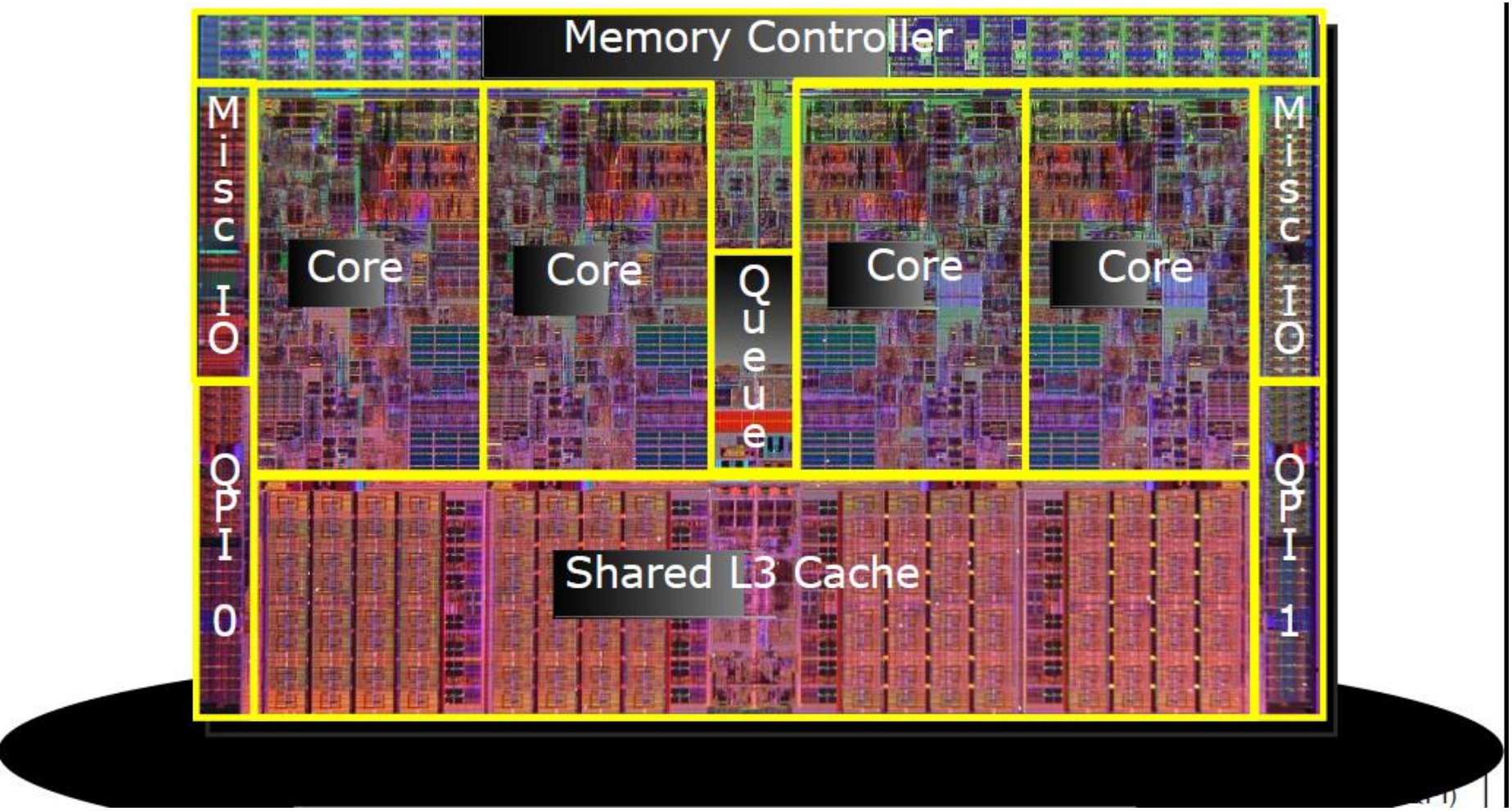
- Brands/families of Intel CPUs
  - **Atom** (ultra-low-voltage microprocessors mainly used in netbooks, embedded applications, IoT)
  - **Core** (mid- to high-end consumer, workstation, and enthusiast)
    - Core i3, i5, i7, i9 → From Lower to Higher Performance
  - **Xeon** (non-consumer workstation, server, and embedded system markets
  - **Itanium** (enterprise servers and high-performance computing systems). NOT x86
- Microarchitectures (TICK-TOCK system).
  - 45 nm : Penryn (2007) → Nehalem (2008)
  - 32 nm: Westmere (2010) → Sandy Bridge (2011)
  - 22 nm : Ivy Bridge (2012) → Haswell (2013)
  - 14 nm : Broadwell (‘14) → Skylake (‘15) → Coffee Lake (‘16) → Kaby Lake (‘17)
  - 10 nm : Cannon Lake (2018)
- Each brand/family can include all microarchitectures
  - Except Itanium which is another world, altogether

CPI comparison



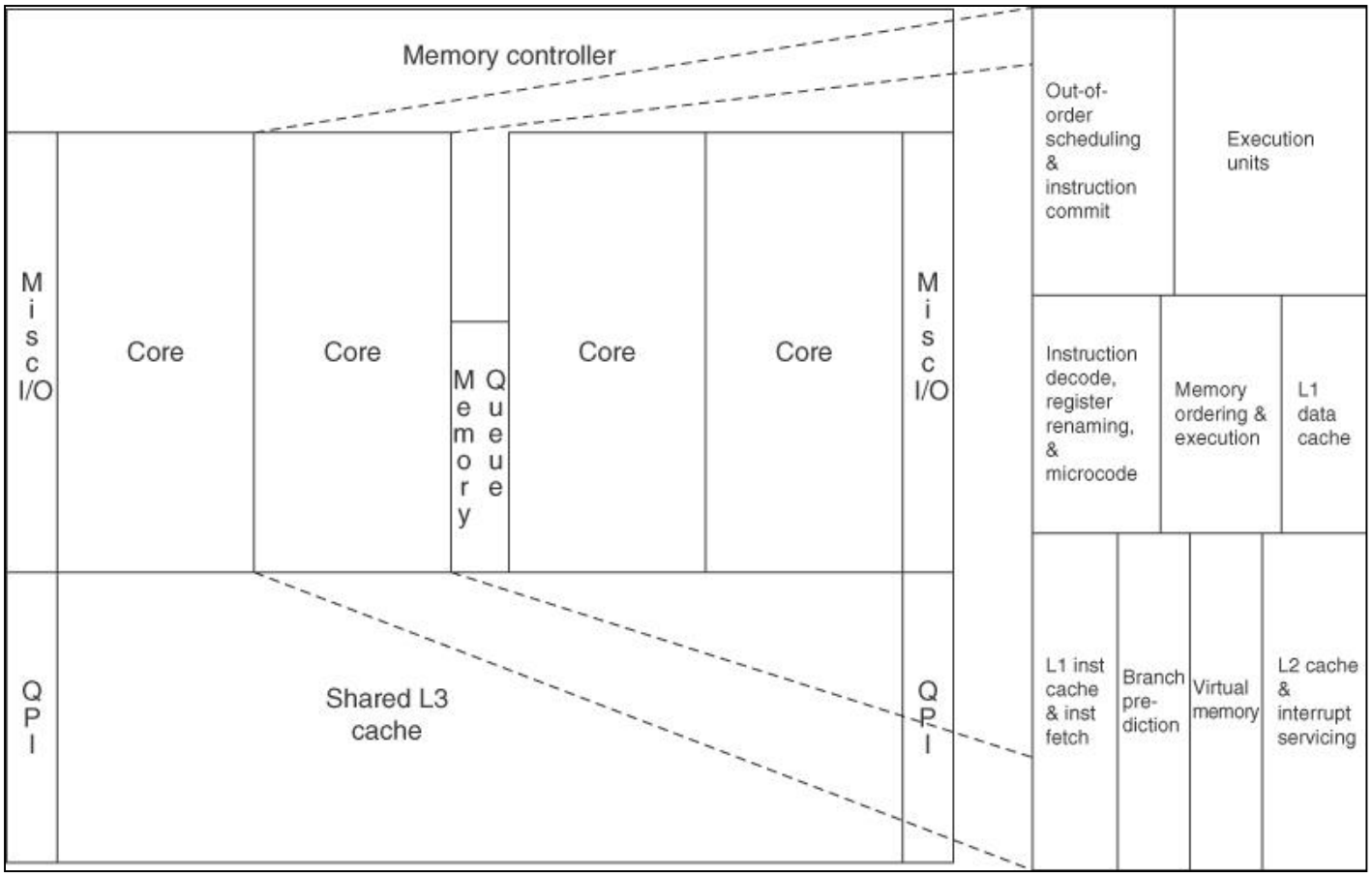
- Nehalem i7-920 (2008) vs Skylake i7-6700 (2015) microarchitectures
- SPECint 2006 benchmark suite
- Skylake microarchitecture has lower CPI mainly due to lower L1 miss rate and better branch prediction

Intel Core i7 microprocessor die



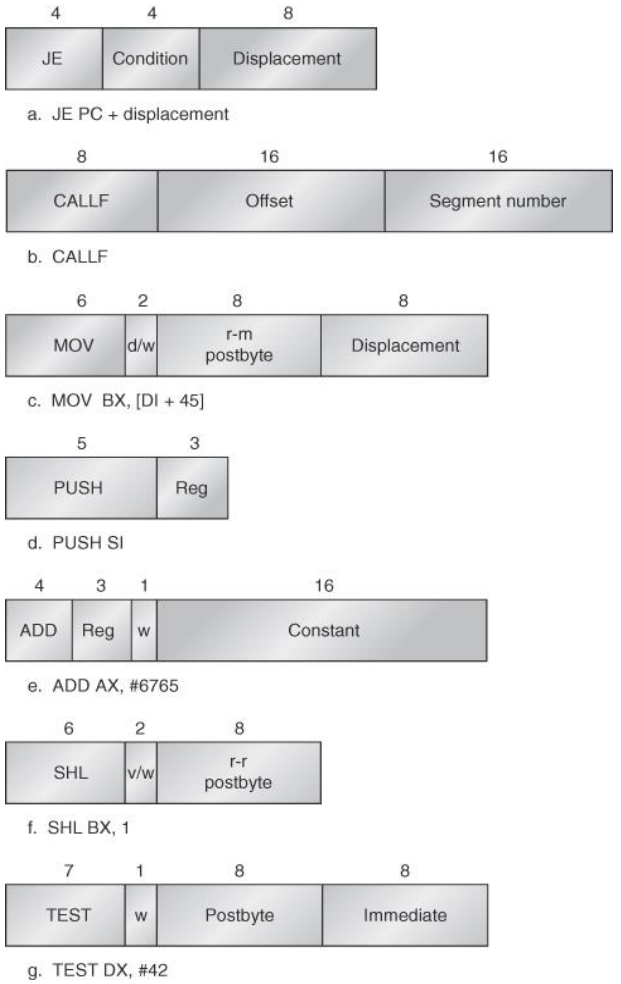
Core i7 is first implementation of Nehalem uarch. The dimensions are 18.9 mm by 13.6 mm (257 mm2) in a 45 nm process.

Intel Core i7 microprocessor die



80x86 ISA

- *CISC* architecture (Complex Instruction Set Computer)
- Many different instruction formats.
- Simpler instructions 1 byte, when there are no operands,
- More complex instructions up to 6 bytes, when the instruction contains a 16-bit immediate and uses 16-bit displacement addressing.
- x86-64 is the 64-bit architecture introduced in 2004 (after AMD64)
- 16 registers, 64-bit
- SSE ISA for SIMD operations
    - 128- and 256-bit registers
- Compare to MIPS ISA which is RISC

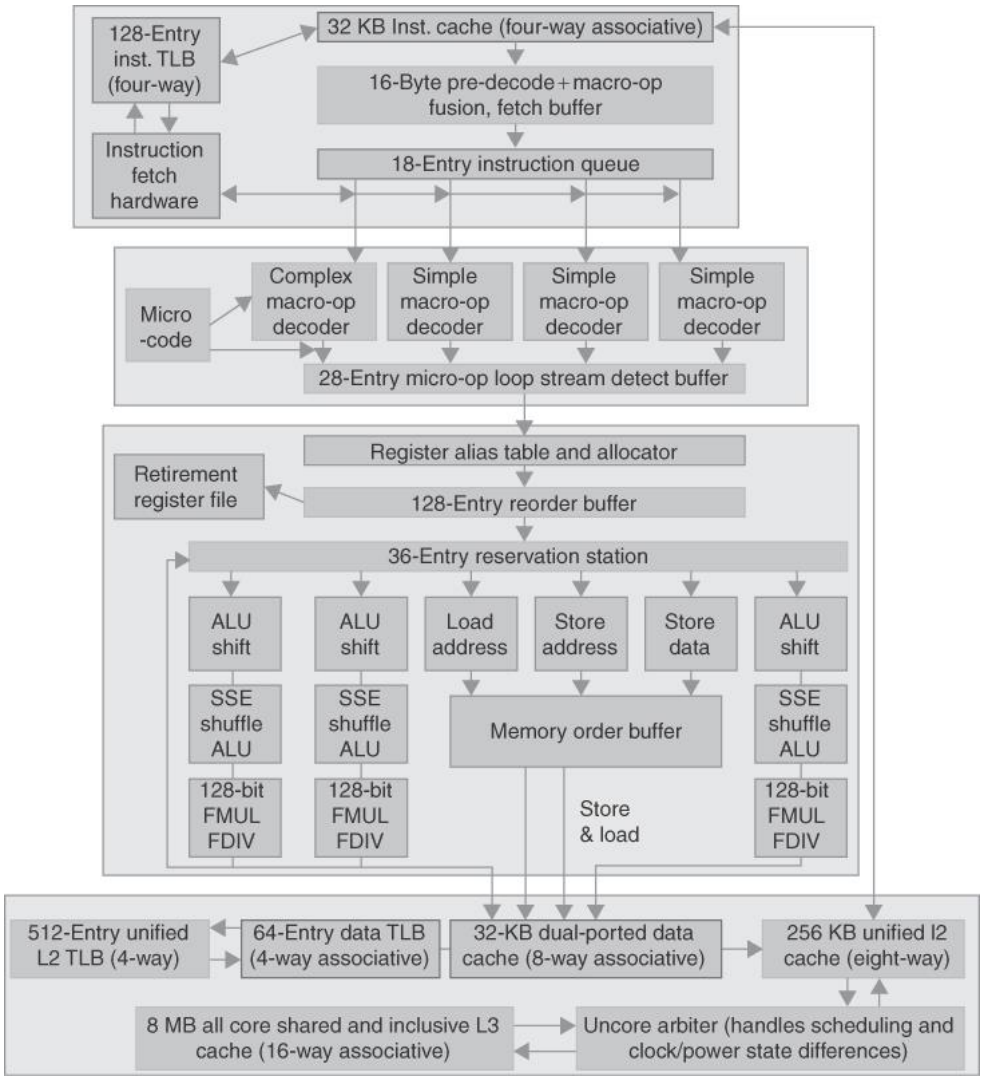


80x86 ISA

- 80x86 complexity is problematic for modern high-performance processors
  - Difficult to fetch and decode variable-length instructions
    - An instruction may span cache lines
  - Difficult as a compiler code generation target
- Individual x86 instructions translated into MIPS-like micro-ops (uops)
  - Done by hardware in ID unit (not the compiler!)
  - First appeared in Pentium Pro (1995)
- Easier to pipeline and execute
- CISC front-end, RISC execution

Intel Core i7 pipeline architecture

- Main characteristics
  - •4-core multiprocessor
  - •Two threads per core, each thread dynamically scheduled (SMT or Hyperthreading)
  - •Pipeline depth 14 cycles
  - •15 cycles for branch misprediction
  - •6 independent Functional Units can➔ 6 uops/cycle
  - • 32KB I/32 KB DCache –L1. One per core (4 cycles, pipelined, 64-byte cache line)
  - •256 KB unified L2 Cache. One per core. (10 cycles)
  - •4x2MB = 8MB common L3 Cache (35 cycles)



Intel Core i7 pipeline architecture

- Instruction Fetch (IF)

•Like most modern HP processors, IF decoupled from back-end

• Multi-level Branch Target Buffer (BTB)

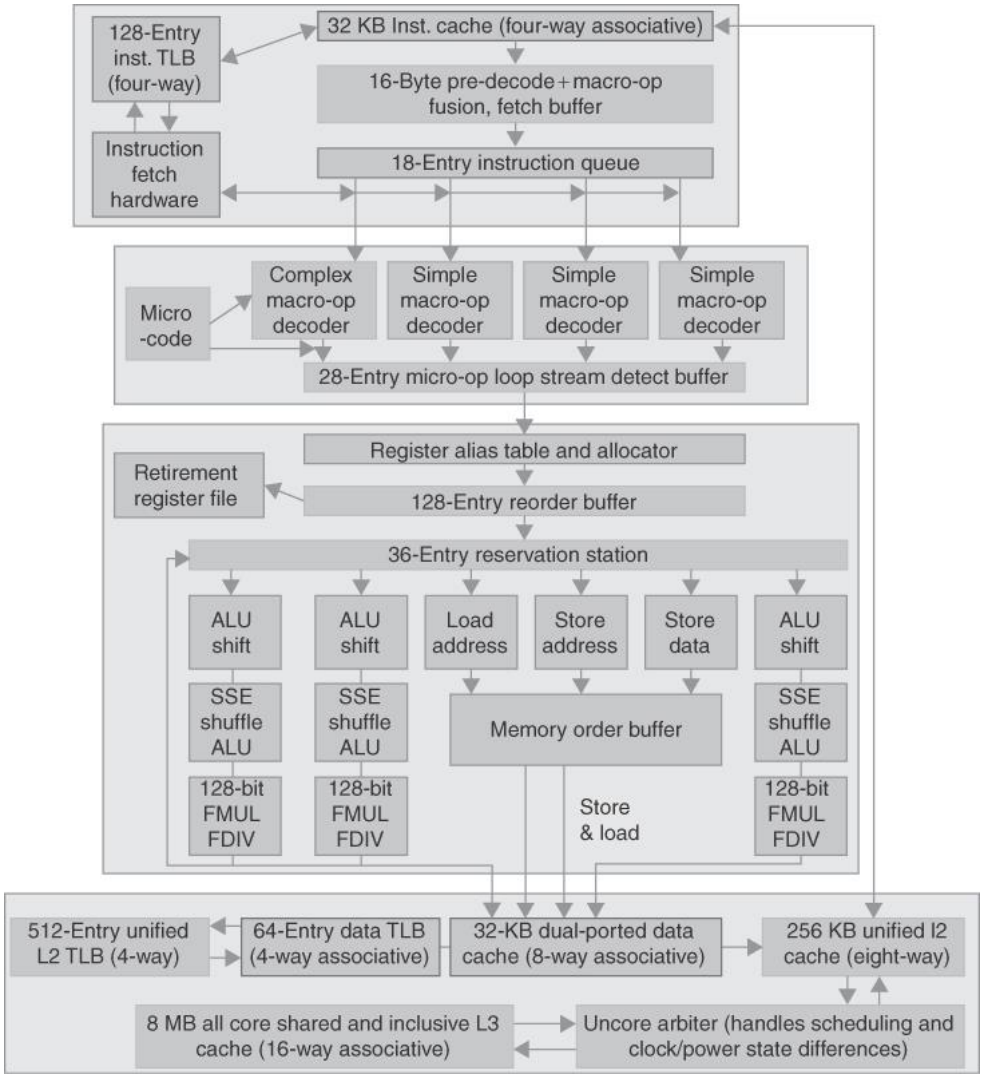
• 16-bytes are fetched every cycle from the 64 byte ICache line to pre-decode buffer

• Break 16 bytes in individual x86 instructions

•Macro-op *fusion* combines specific clusters of x86 operations in one macro-operation.

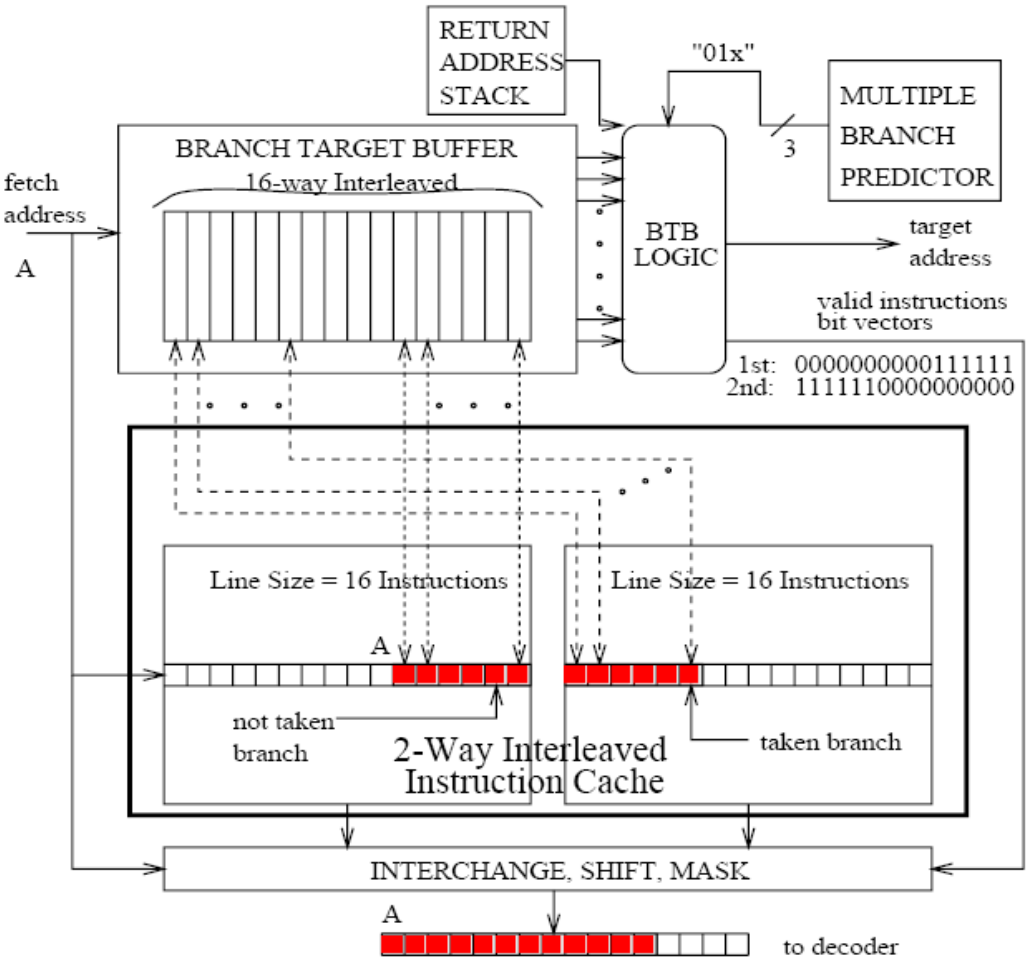
- e.g. CMP+J*cc* → executed and committed as one instruction

• All x86 instructions are placed in instruction queue



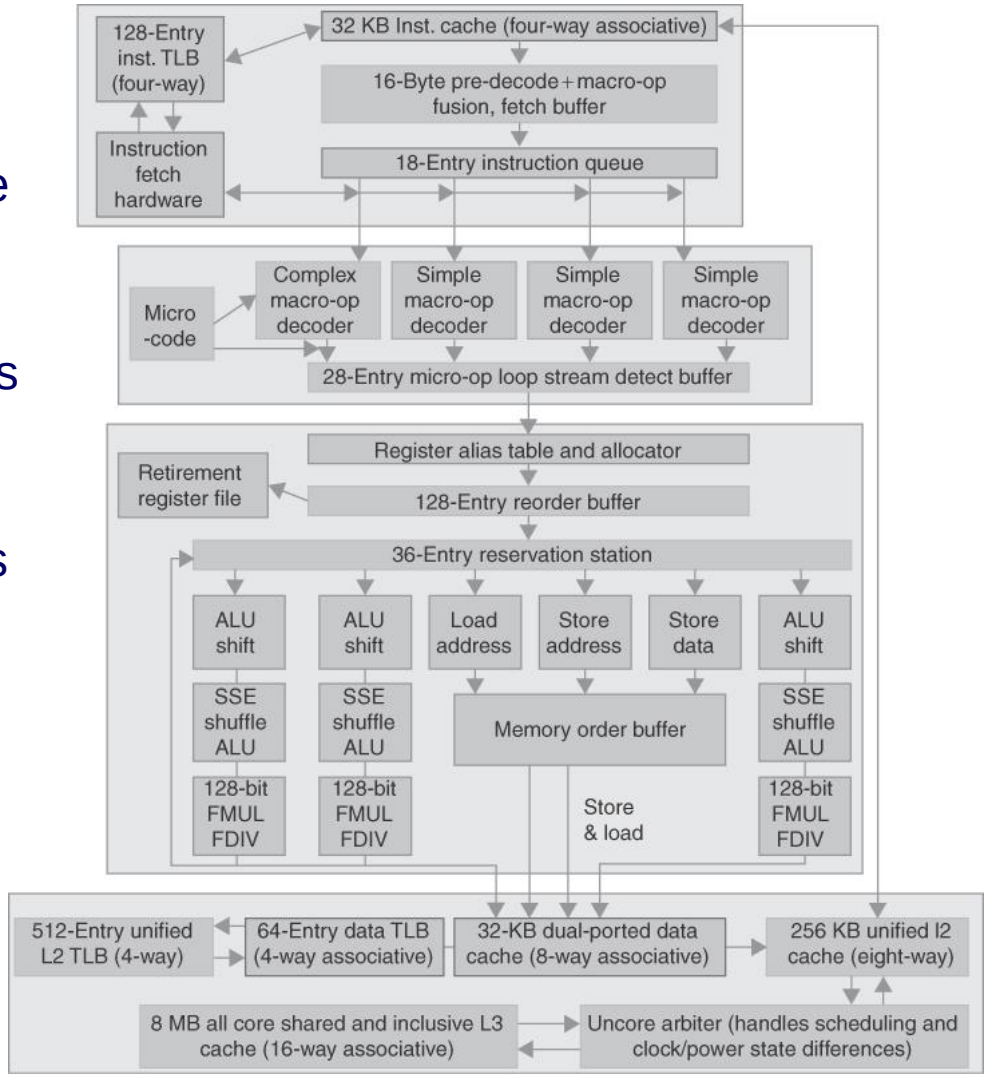
Front End Design

- In modern processors, the front end is a separate autonomous unit, decoupled from the execution unit.
- Not just a pipeline stage
- Up to 16 instructions/cycle in this example

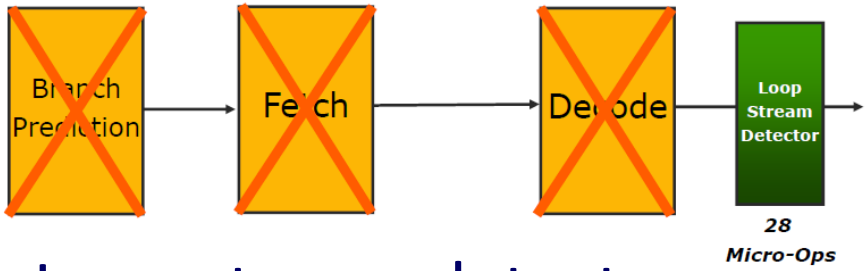


Intel Core i7 pipeline architecture

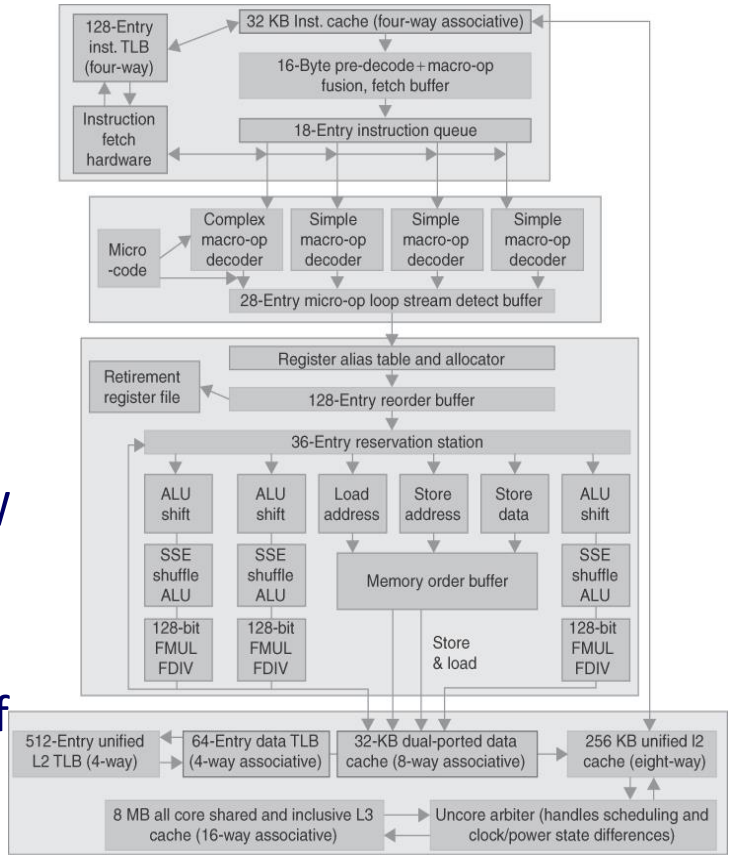
- Instruction Decode (ID)
  - x86 instructions move to ID unit to be translated into RISC-like uops
  - Three decoders used for direct 1-1 translation from simple x86 instructions to micro-ops
  - Microcode unit used to generate micro-ops for complex x86 instructions
  - All micro-ops placed in the 28-entry buffer



Intel Core i7 pipeline architecture

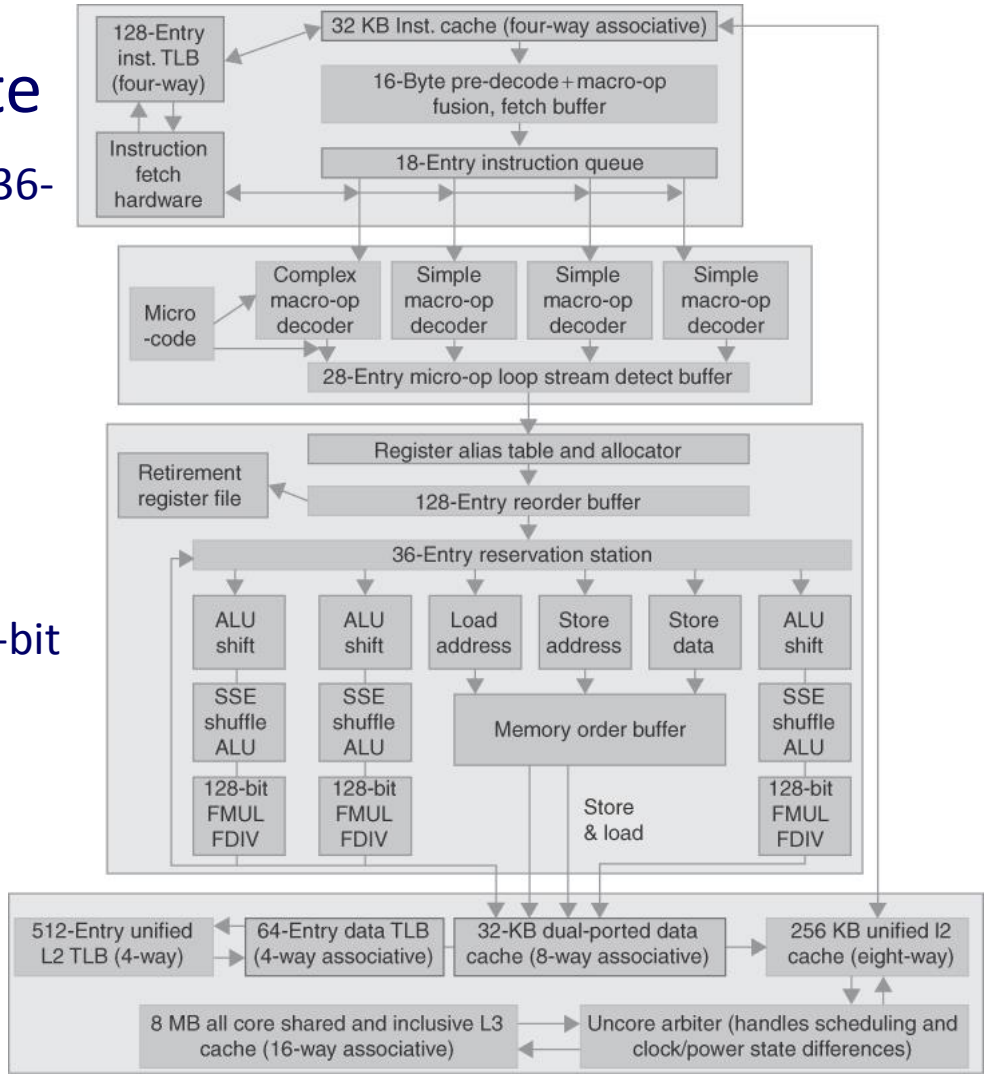


- Loop stream detector
  - Loop Stream Detector identifies tight software loops
  - Take advantage of knowledge of loops in HW and avoid fetching and decoding same instructions over and over
  - Stream from Loop Stream Detector instead of normal path
  - Disable unneeded blocks of logic for ***power savings***
  - ***Higher performance*** by removing instruction fetch limitations
  - Microfusion to combine pairs of uops



Intel Core i7 pipeline architecture

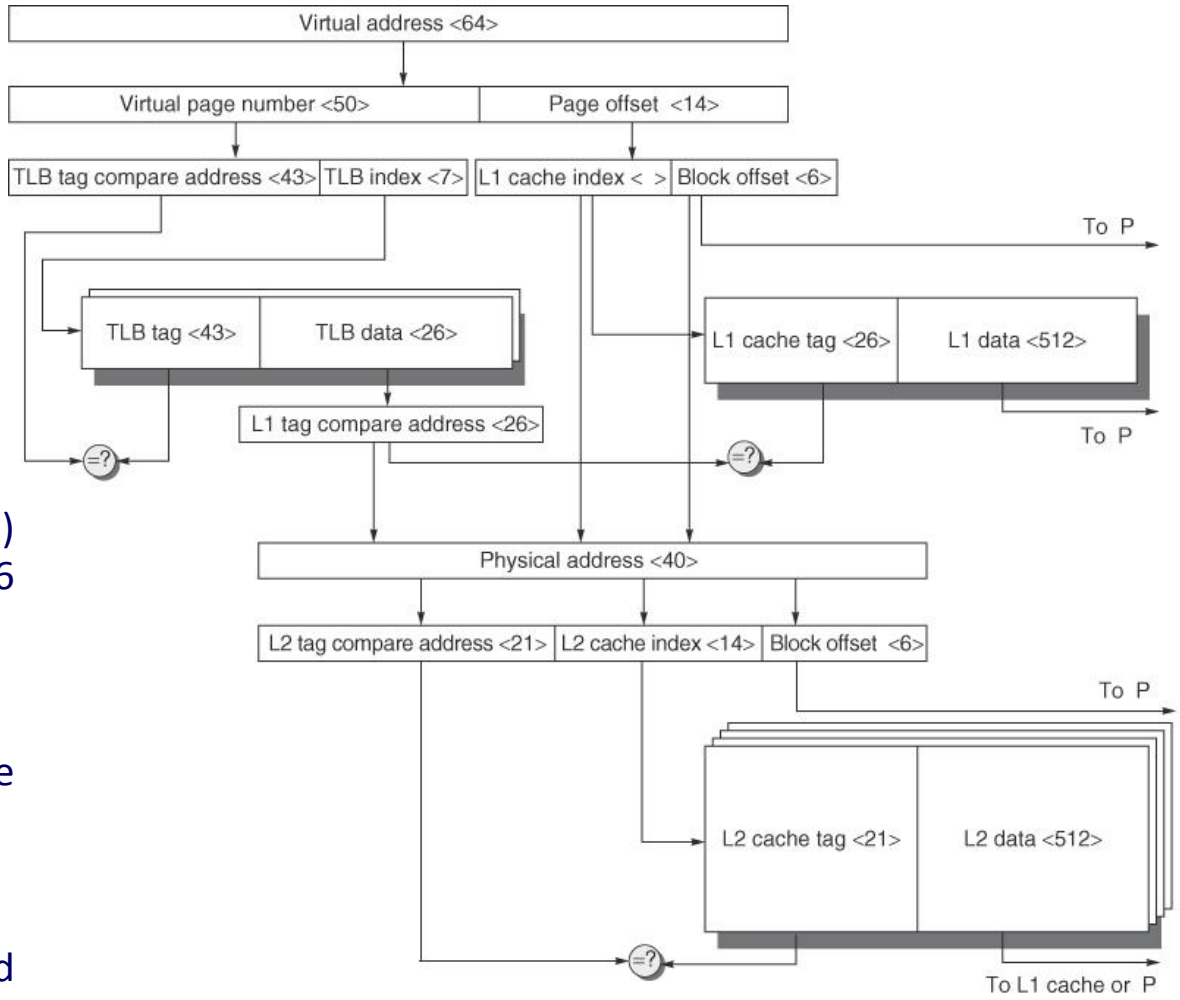
- Instruction Issue and Execute
  - Tomasulo for Dynamic scheduling with 36-entry centralized reservation station
  - Reorder Buffer (128 entry)
  - Up to 6 uops/cc issued
    - 3 memory uops
    - 3 computational uops
  - Advanced Digital Media boost with 128-bit wide SSE



Memory Hierarchy 101

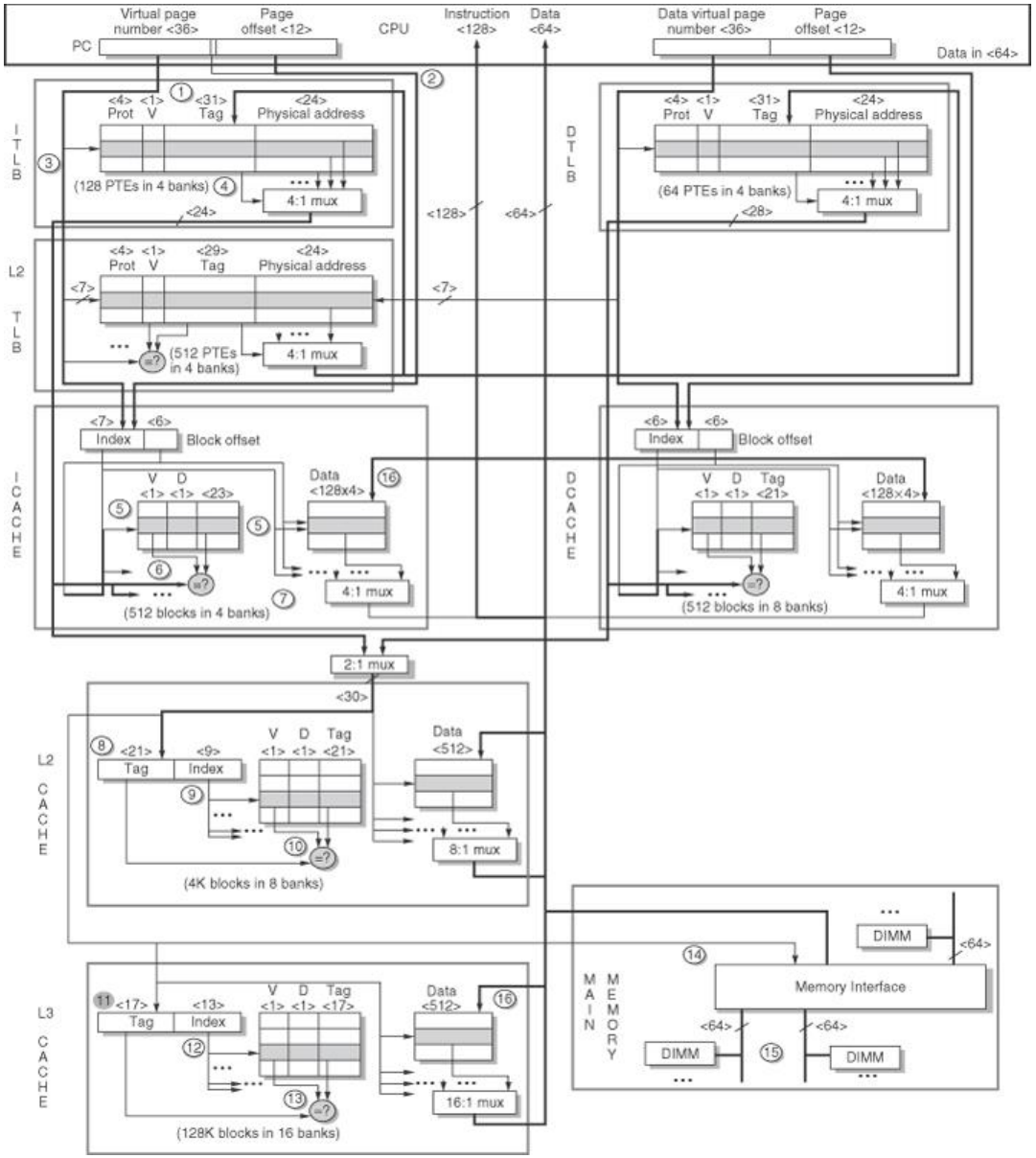
Overall picture of a *hypothetical* memory hierarchy from virtual address to L2 cache access

- Virtual address 64 bits
- Physical address 40 bits
- Page size 16 KB
- Translation Look Aside Buffer (TLB) is two-way set associative with 256 entries.
- L1 cache direct-mapped 16 KB
- L2 cache four-way set associative with a total of 4 MB
- Both 64-byte blocks.
- Virtually indexed, physically tagged L1 Cache



Intel Core i7 memory hierarchy

- 48-bit virtual address, 36-bit physical address
- Virtually indexed, physically tagged caches overlap data reads with TLB translation
- L2, L3 caches are unified and physically tagged
- Two levels of TLB for higher TLB hit rate
- Miss penalty 135 cycles (data from main memory)

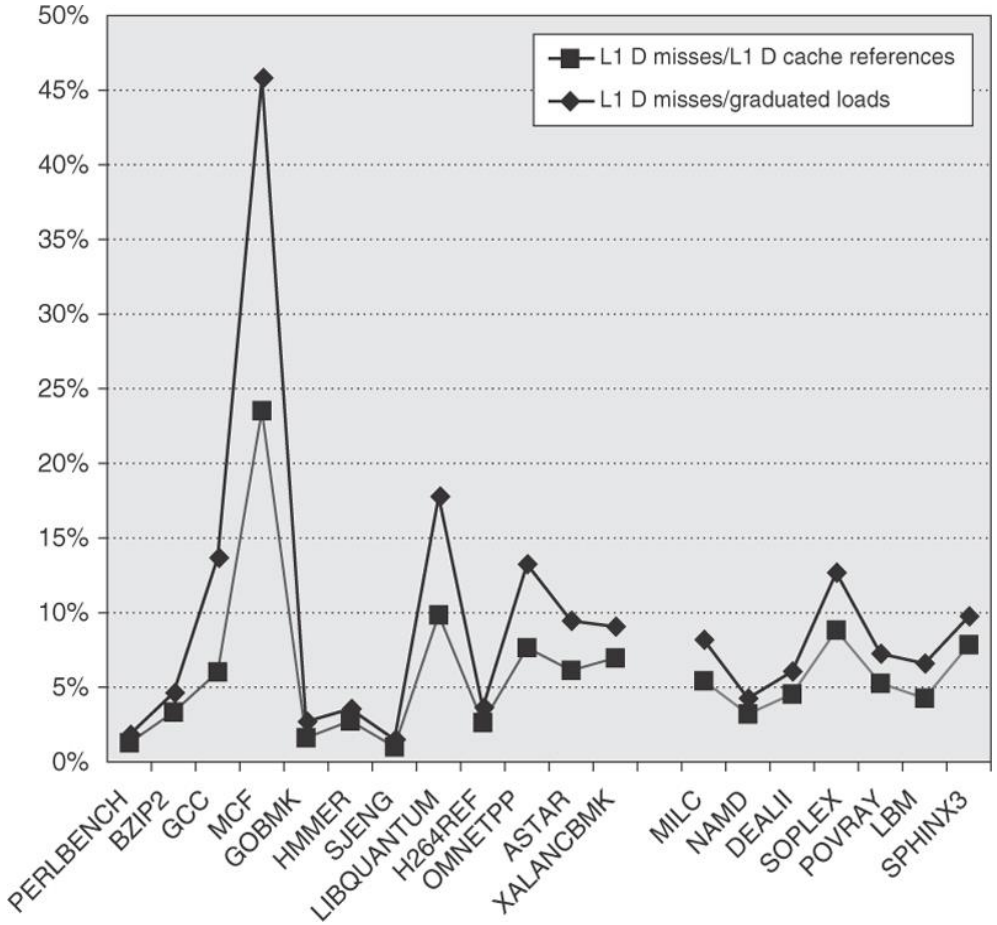


L1 DCache miss rate

L1 data cache miss rate for 17 SPECCPU2006 benchmarks

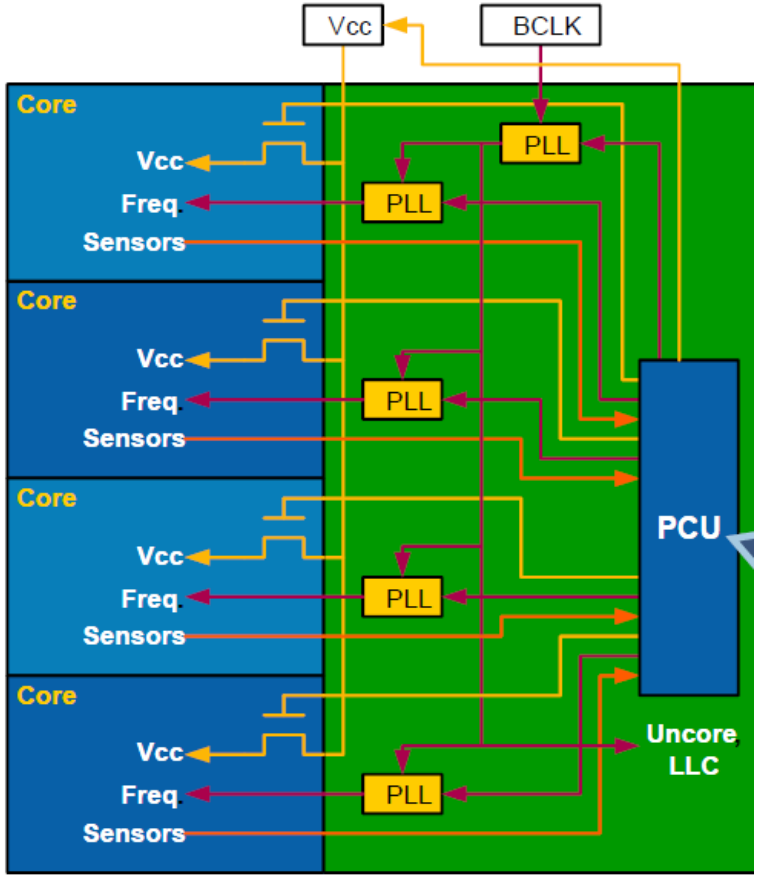
- Relative to the actual loads that complete execution successfully
- Relative to all the references to L1
    - includes prefetches, speculative loads that do not complete, and writes, which count as references, but do not generate misses.

• Misses from real (non-speculative) loads are lower



Power Management

- Power Control Unit (PCU)
  - Small programmable microcontroller used exclusively for power management
  - Monitors temperature, current and power at runtime using sensors
  - Individually for each core
  - PCU analyzes sensor readings data
  - switches qualifying cores to power-saving mode by adjusting their frequency and voltage.
  - PCU may disable inactive cores and put them in deep sleep state where their power consumption will be close to 0.



Power Management

- PCU responsible for placing cores in one of the following states:
  - C0: CPU active state
  - C1: stop core pipeline by stopping most core clocks
  - C3: stop remaining core clocks
  - C6: save architectural state and turn off power. Eliminates leakage as well
- PCU under the control of the OS