Memory Systems

Instructor: Saraju P. Mohanty

- Memory Hierarchy
- Locality of Reference
- Cache Memory
- Virtual Memory

Sources

- Logic and Computer Design Fundamentals by M. M. Mano and C. R. Kime.
- Dr. Valavanis lectures

Memory Hierarchy

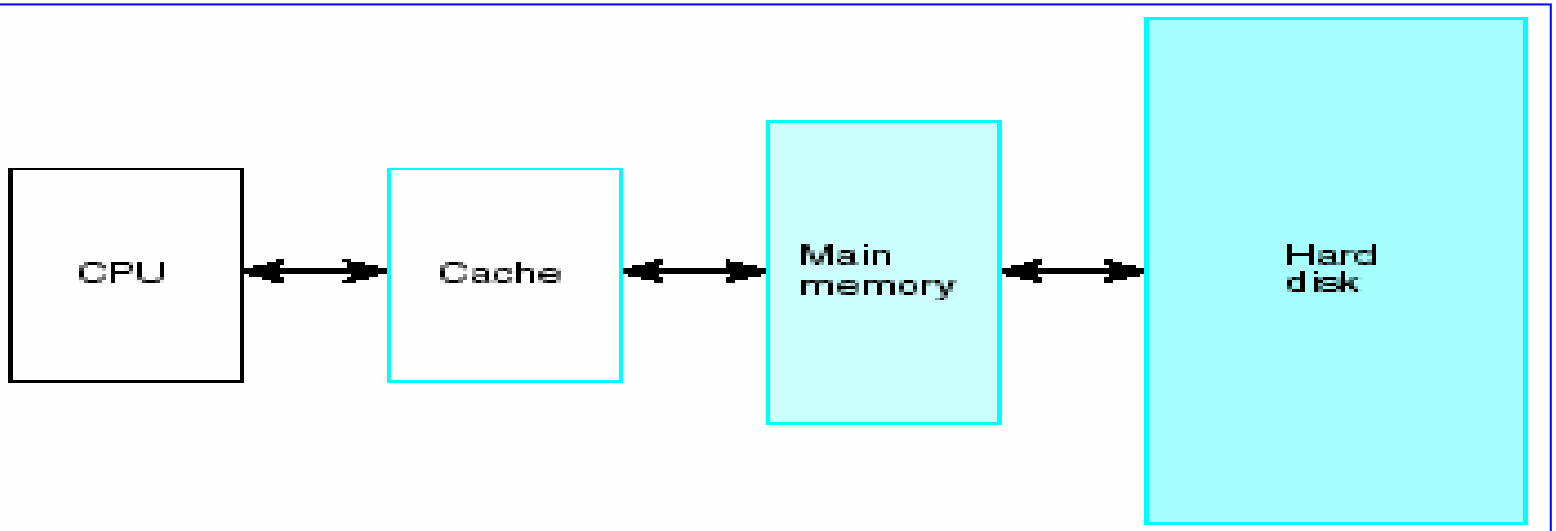


Fig. 12-1 Memory Hierarchy

- Lowest Level: Cache – a small, fast memory. a large proportion of CPU instruction and operand fetches come from cache.
- Next Level: Main memory – serves directly most of the CPU instruction and operand fetches not satisfied by the cache.
- Top Level: Hard Disk – accessed only in the very infrequent cases in which a CPU instruction or operand fetch is not found in main memory.

Memory Hierarchy

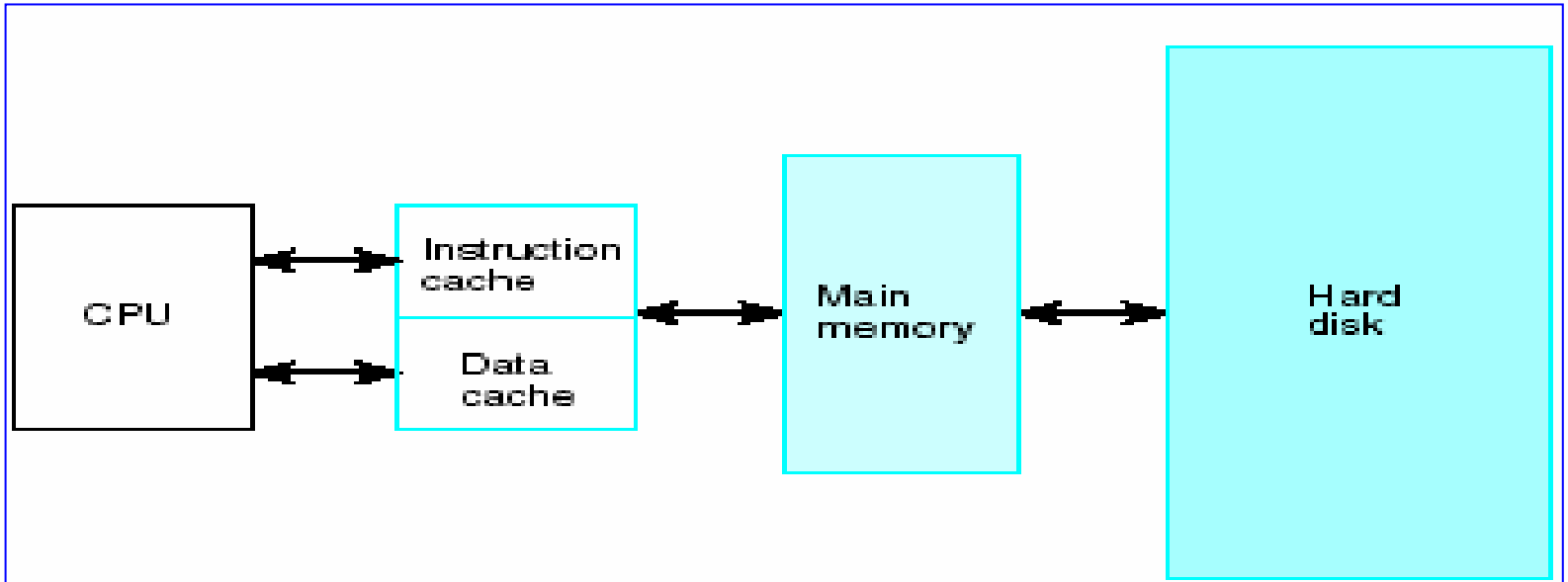


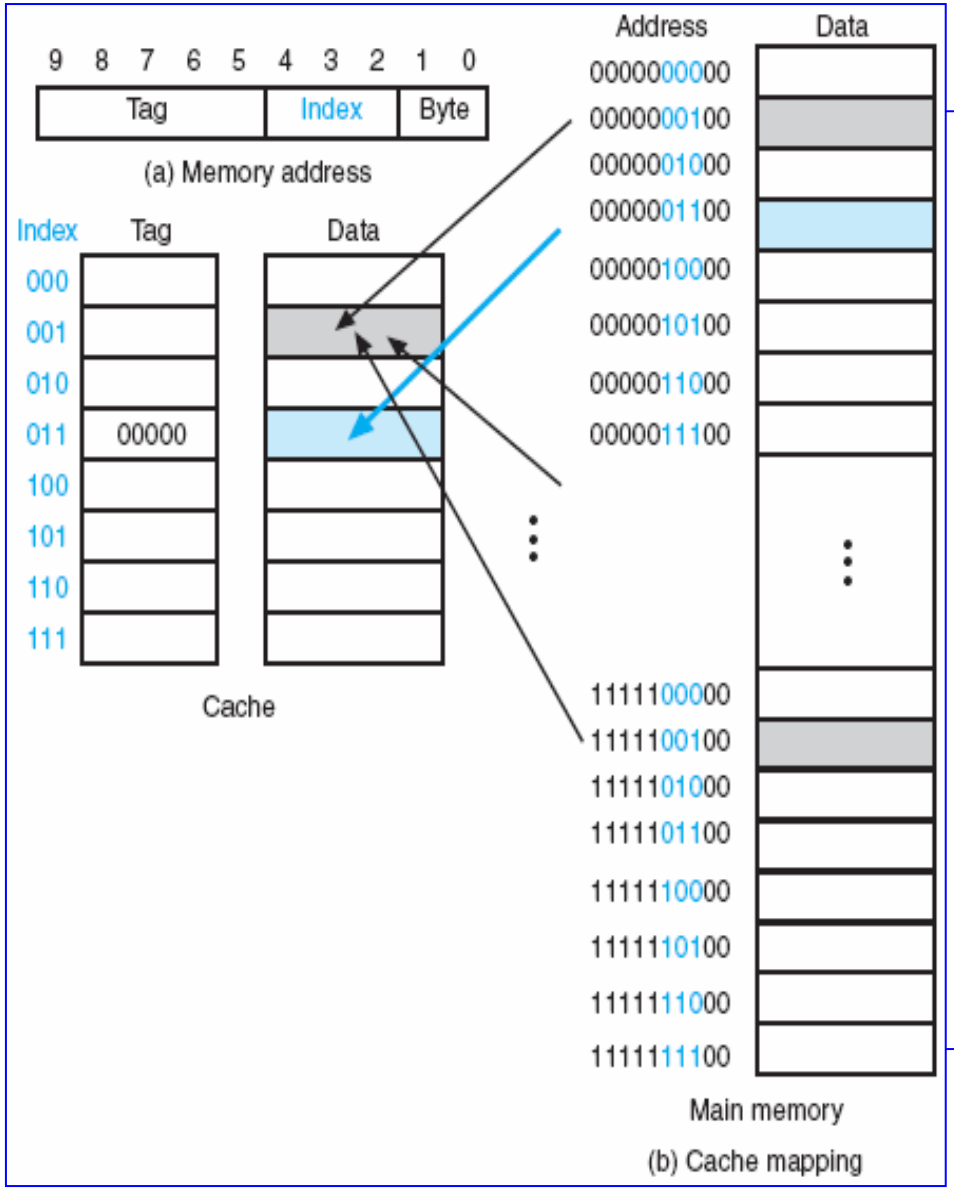
Fig. 12-2 Example of Memory Hierarchy

- Two separate caches – one for instructions and one for data.
- The use of 2 caches permits one instruction and one operand to be fetched, or one instruction to be fetched and one result to be stored in a single clock cycle.

Locality of Reference

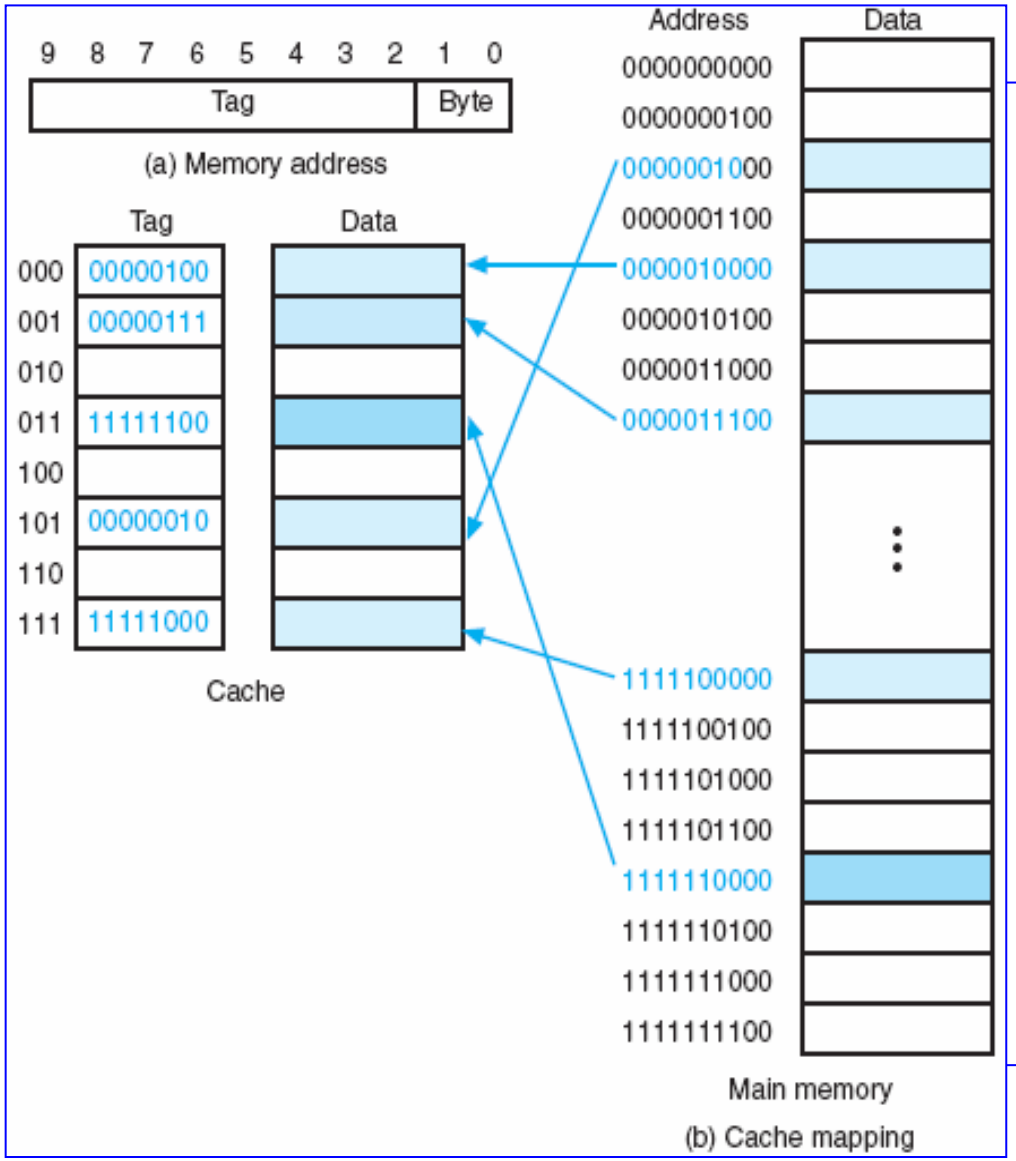
- A principle that makes having a memory hierarchy a good idea.
- Two types of locality: temporal locality and spatial locality.
- Temporal locality: If an item is referenced, it will tend to be referenced again soon
- Spatial locality: If an item is referenced, nearby items will tend to be referenced soon.
- Terminology:
  - block: minimum unit of data
  - hit: data requested is in the upper level
  - miss: data requested is not in the upper level

Direct-Mapped Cache



- The cache has 8 32-bit words, main memory is 1KB (256 words).
- Cache address – 3 bits , Main memory address – 10 bits.
- *Index* – 2-4 bits of the main memory address, which are used as cache address.
- *Byte* – 0-1 bits of the main memory address, used as byte offset in a word.
- *Tag* – 5-9 bits of the main memory address, identifies an address along with index and byte fields.

Fully-Associative Cache Mapping



- Direct mapping – only one location in cache can contain the word from particular main memory location.
- Fully-associative – Any location in memory can be mapped to any one of the eight addresses in the cache. *Memory address is divided as 8 tag bits and 2 byte offset bits.*

Associative Memory

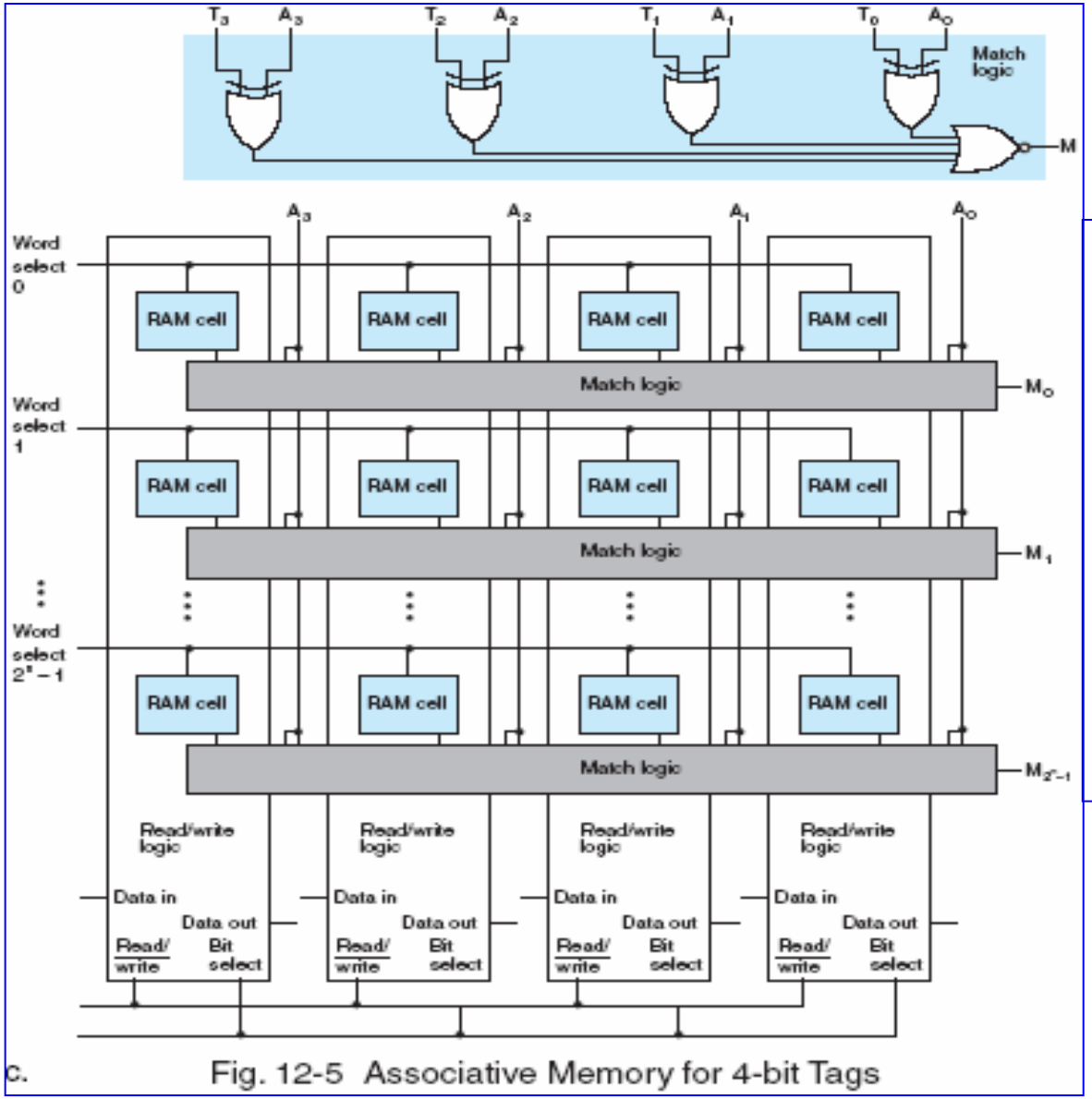


Fig. 12-5 Associative Memory for 4-bit Tags

- Match logic does an equality comparison or match between the tag T and applied address A from CPU.
- The match logic for each tag is composed of XOR gates for each bit and a NOR gate.

Two-way Set-Associative Cache

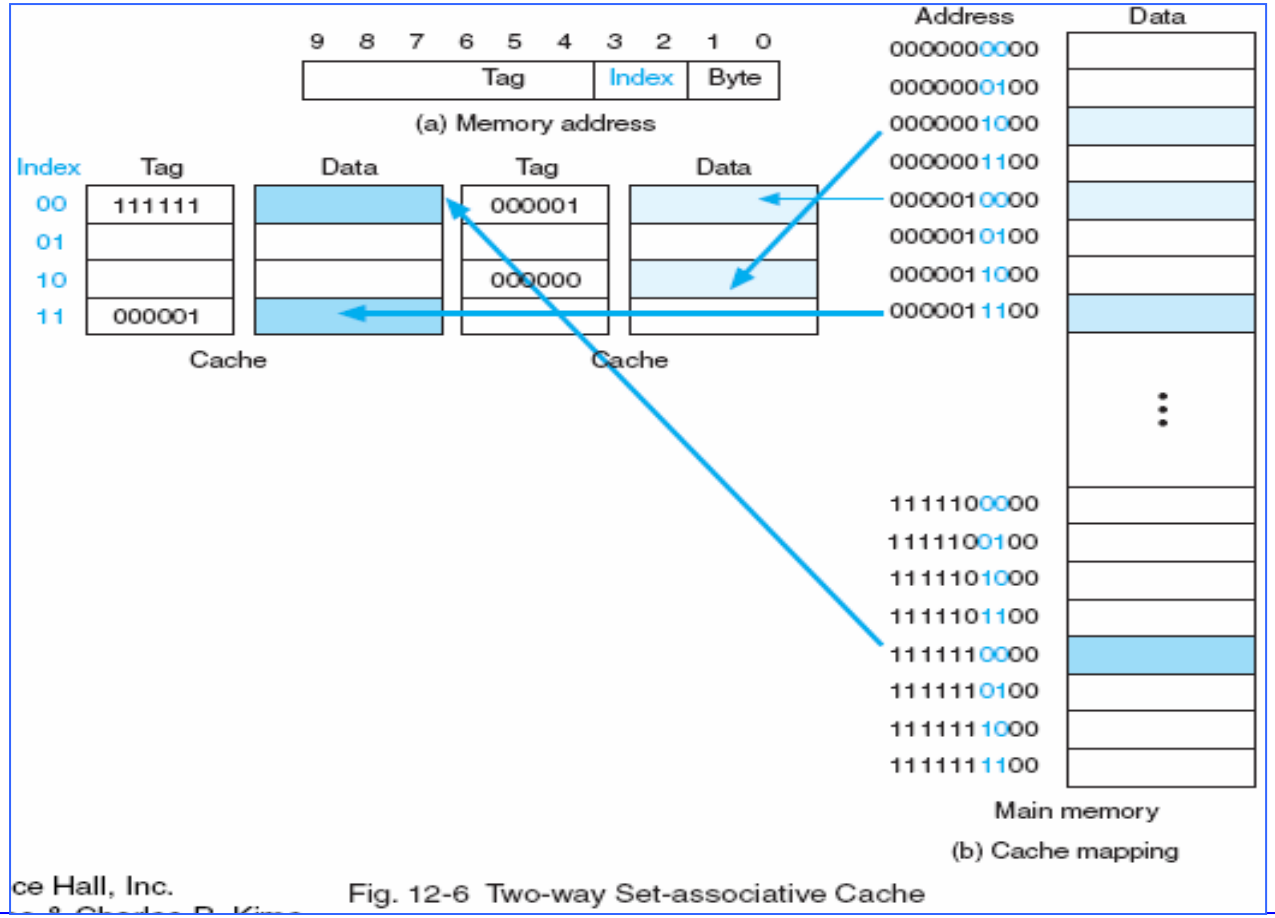


Fig. 12-6 Two-way Set-associative Cache

- *Set-associative mapping* - There are sets of locations and associativity is used on sets.
- *Two-way set-associative* – Each set consists of two locations.

Set-Associative Cache

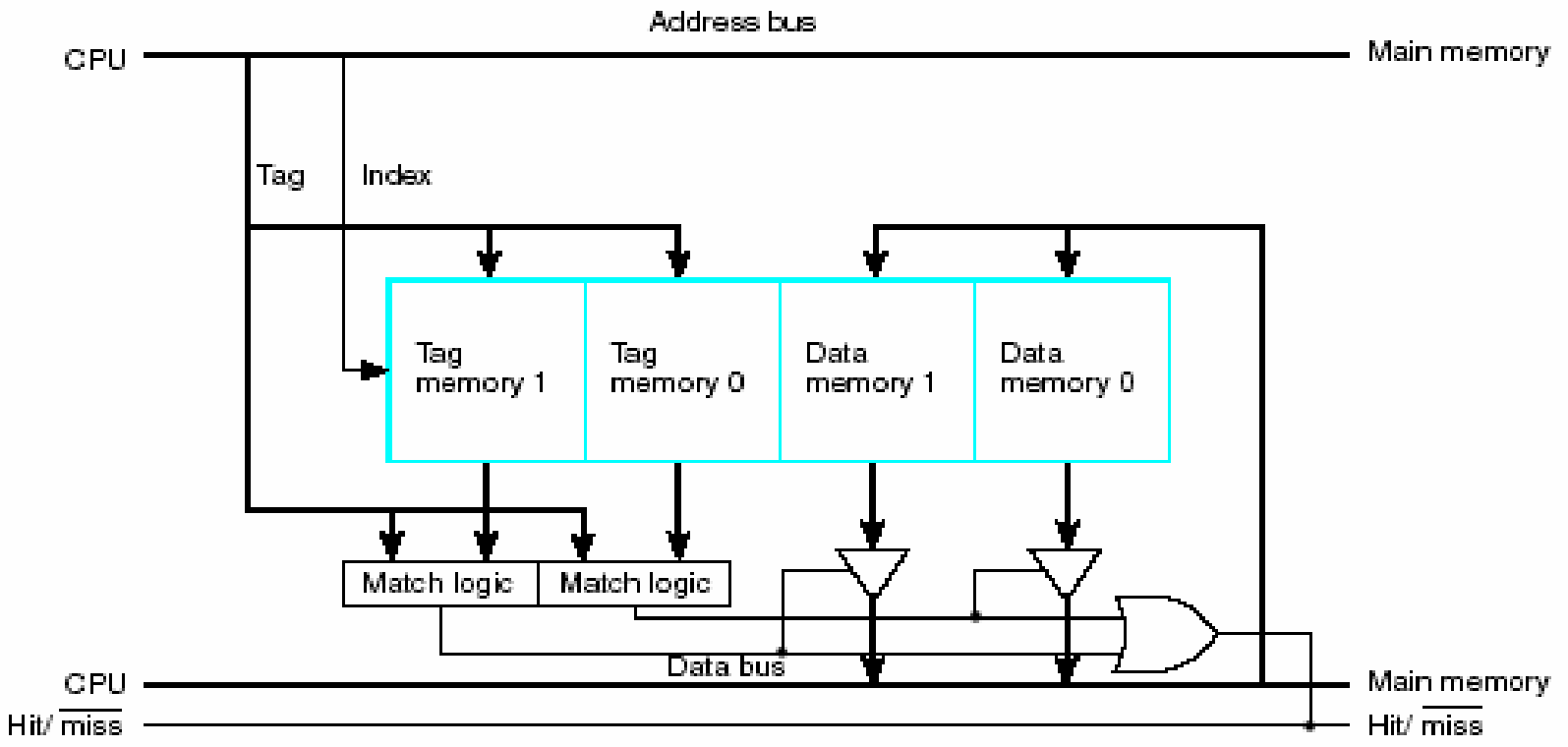


Fig. 12-7 Partial Hardware Block Diagram for Set-associative Cache

- Index is used to address each row of the cache memory.
- Two tags read from the tag memories are compared to the tag part of the address on the address bus from the CPU.
- If tags match, a three-state buffer is activated and data is placed on to the data bus.

Set-Associative Cache

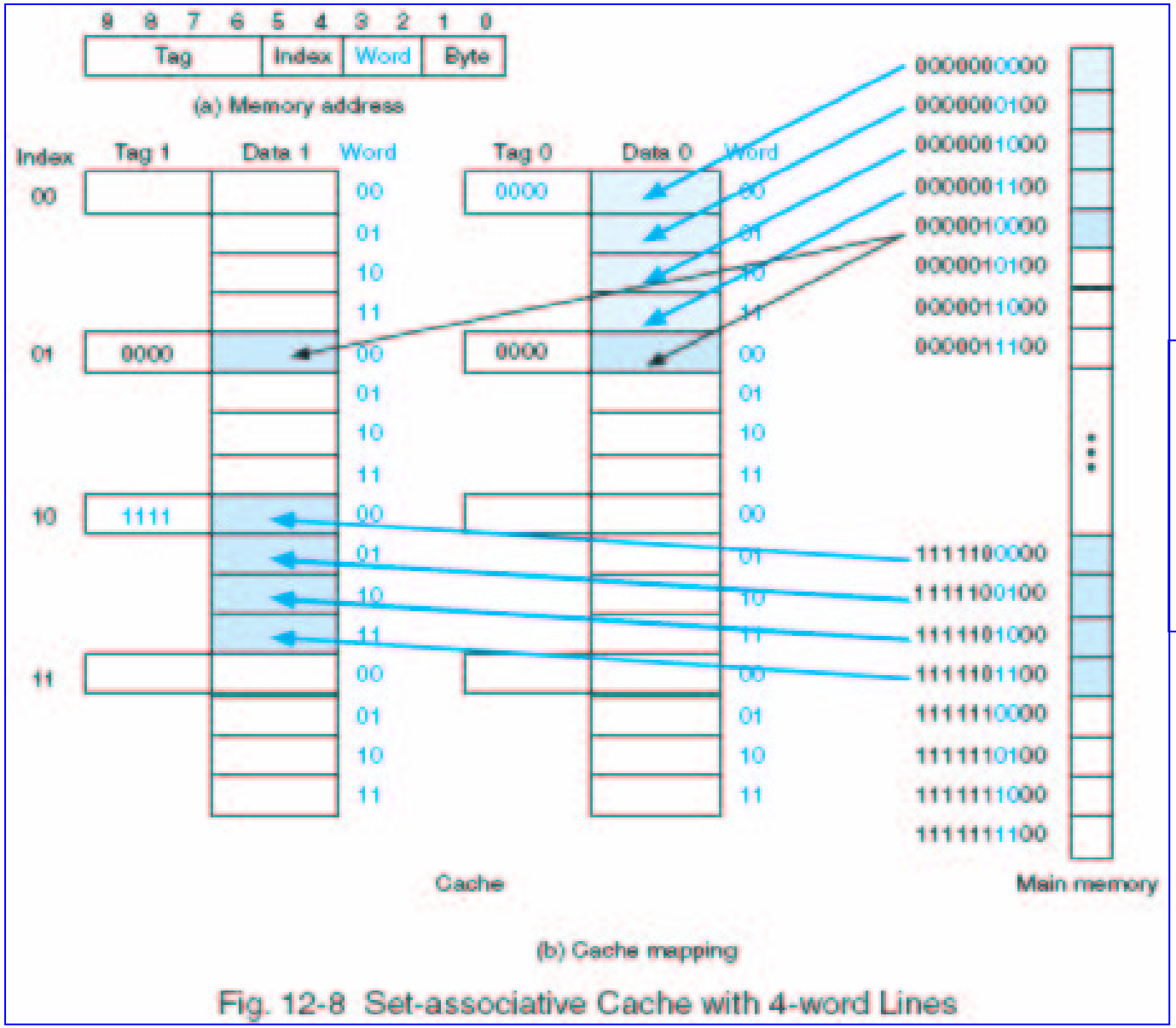


Fig. 12-8 Set-associative Cache with 4-word Lines

A block of l words called a line is fetched to exploit spatial locality.

Cache-block diagram

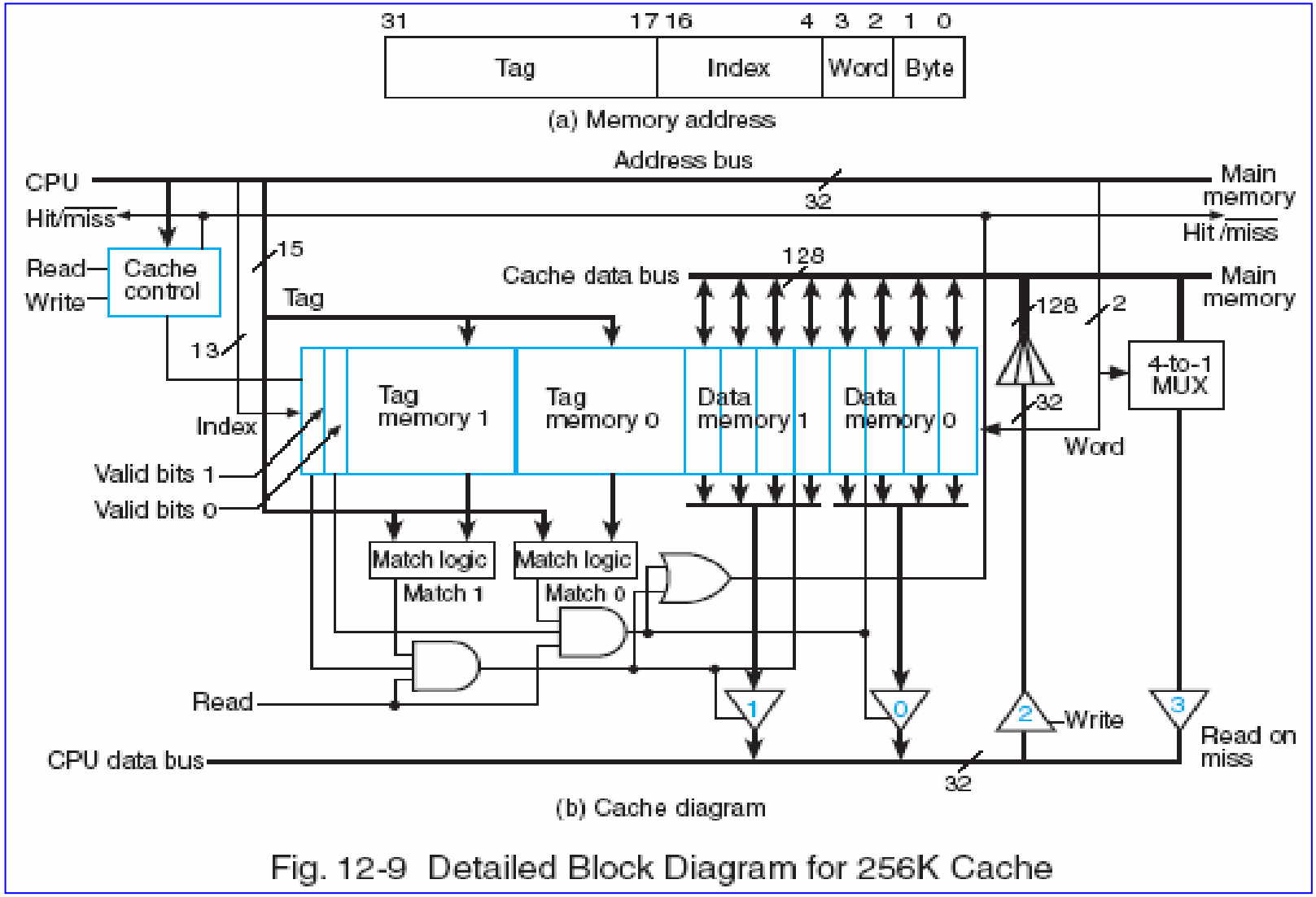


Fig. 12-9 Detailed Block Diagram for 256K Cache

Cache – Read and Write Operations

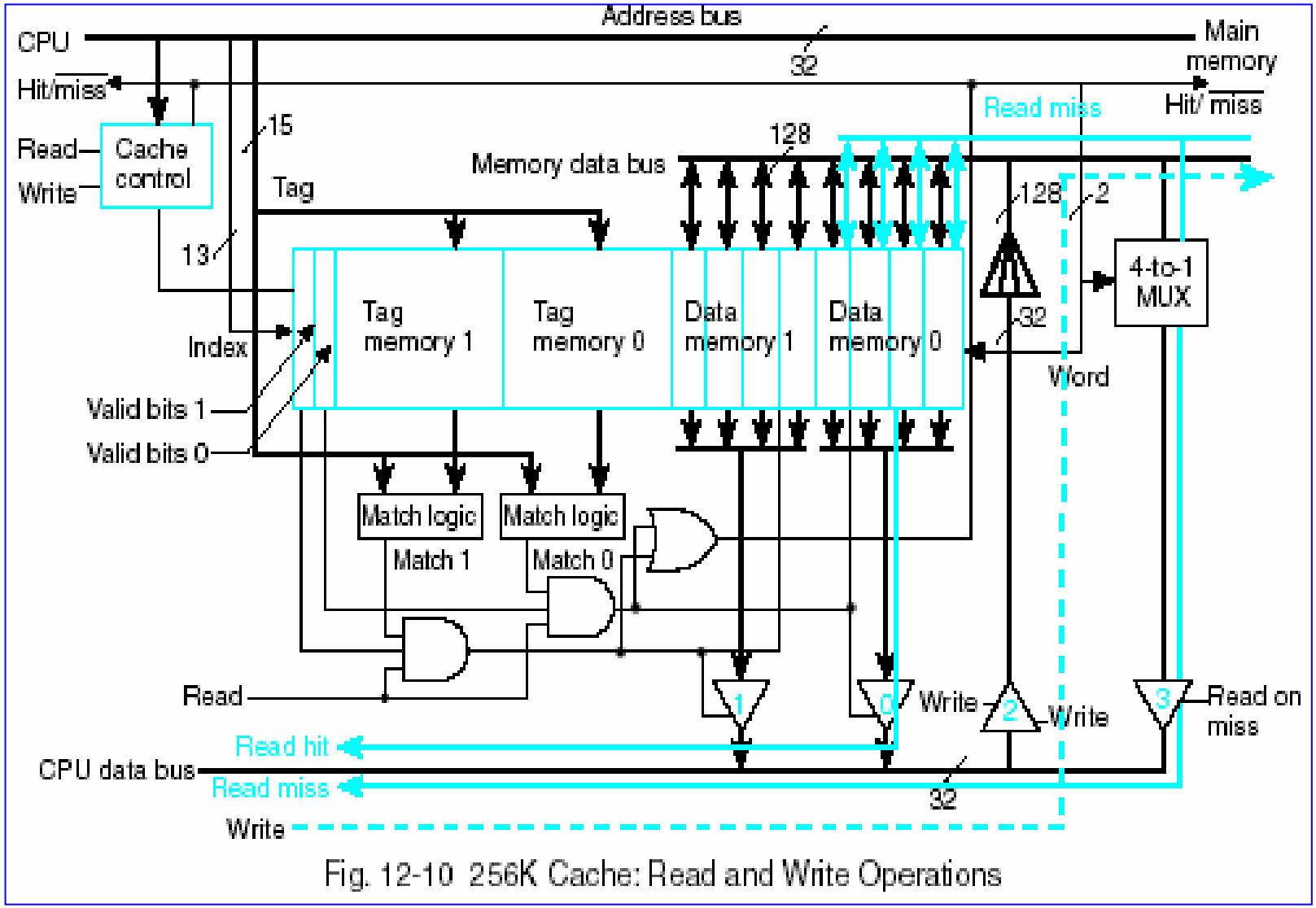


Fig. 12-10 256K Cache: Read and Write Operations

Virtual Memory

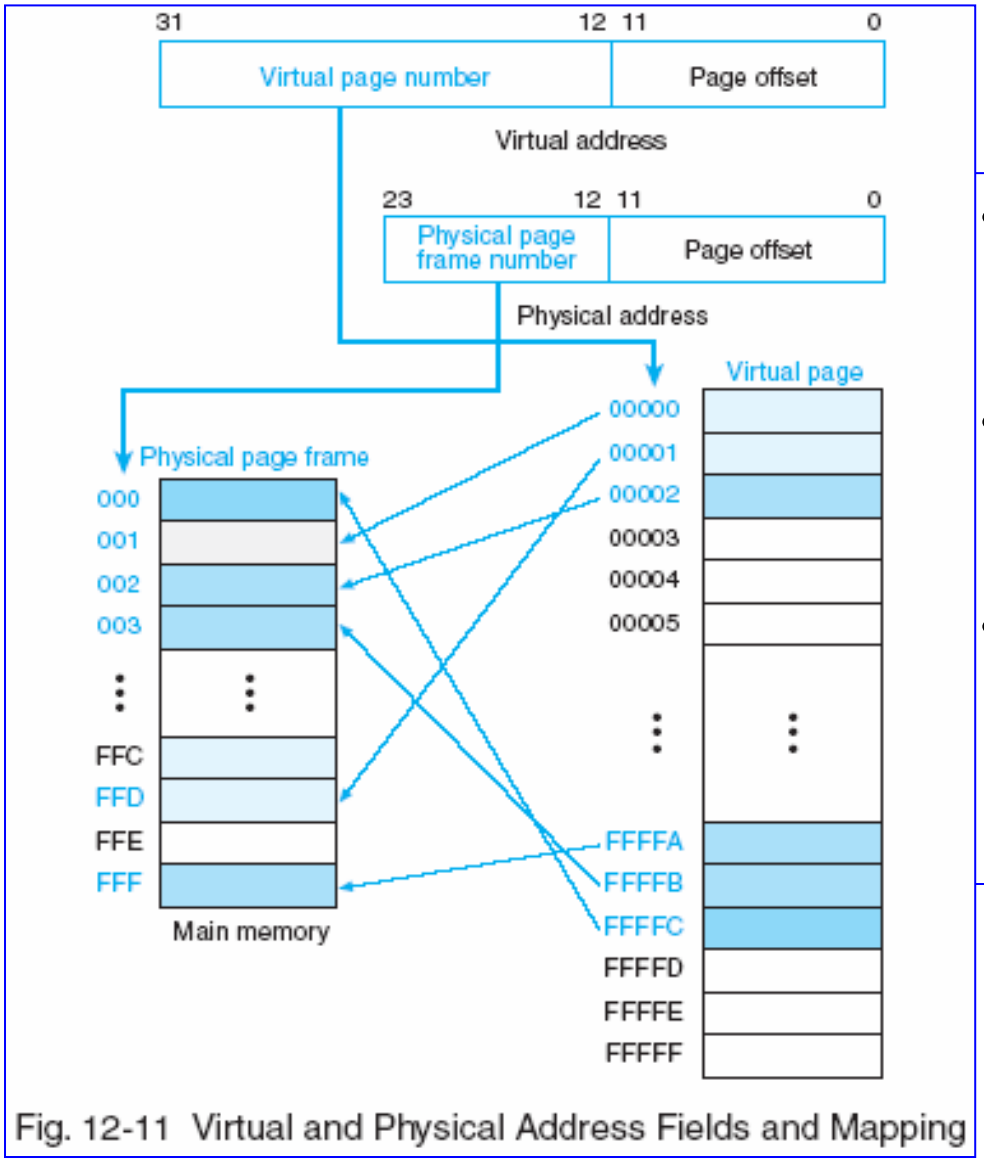


Fig. 12-11 Virtual and Physical Address Fields and Mapping

- *Virtual Address space* – The address space used by a program.
- *Physical Address space* – The address space available in the main memory.
- Virtual Memory implementation – mapping the program's virtual address to a physical address in main memory.

Virtual Memory: Page Tables

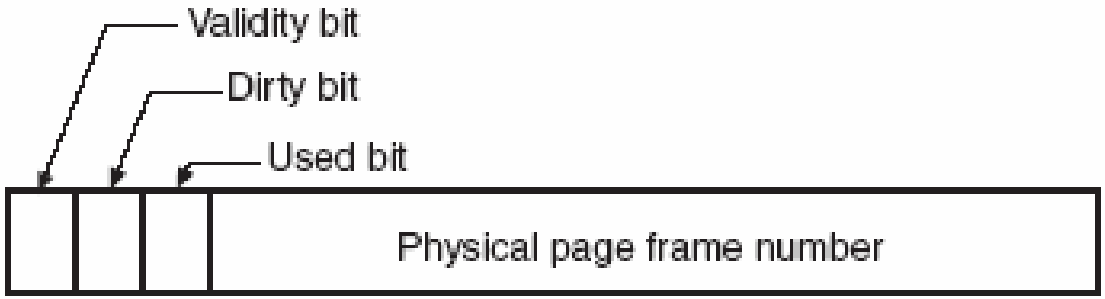


Fig. 12-12 Format for Page Table Entries

- Page Tables – store the mappings of virtual addresses to physical addresses.
- Page frame number – location where the page is stored in main memory.
- Validity bit – when set to 1, the page frame in memory is valid, when 0, the page frame does not correspond to correct code or data (i.e., invalid).
- Dirty bit – when set to 1, indicates a write to at least one byte in the page since it was placed in main memory. When 0, indicates no writes have been made to the page since it was placed in main memory.
- Used bit – used to implement an approximation to an LRU replacement scheme.

Virtual Memory: Page Table Structure

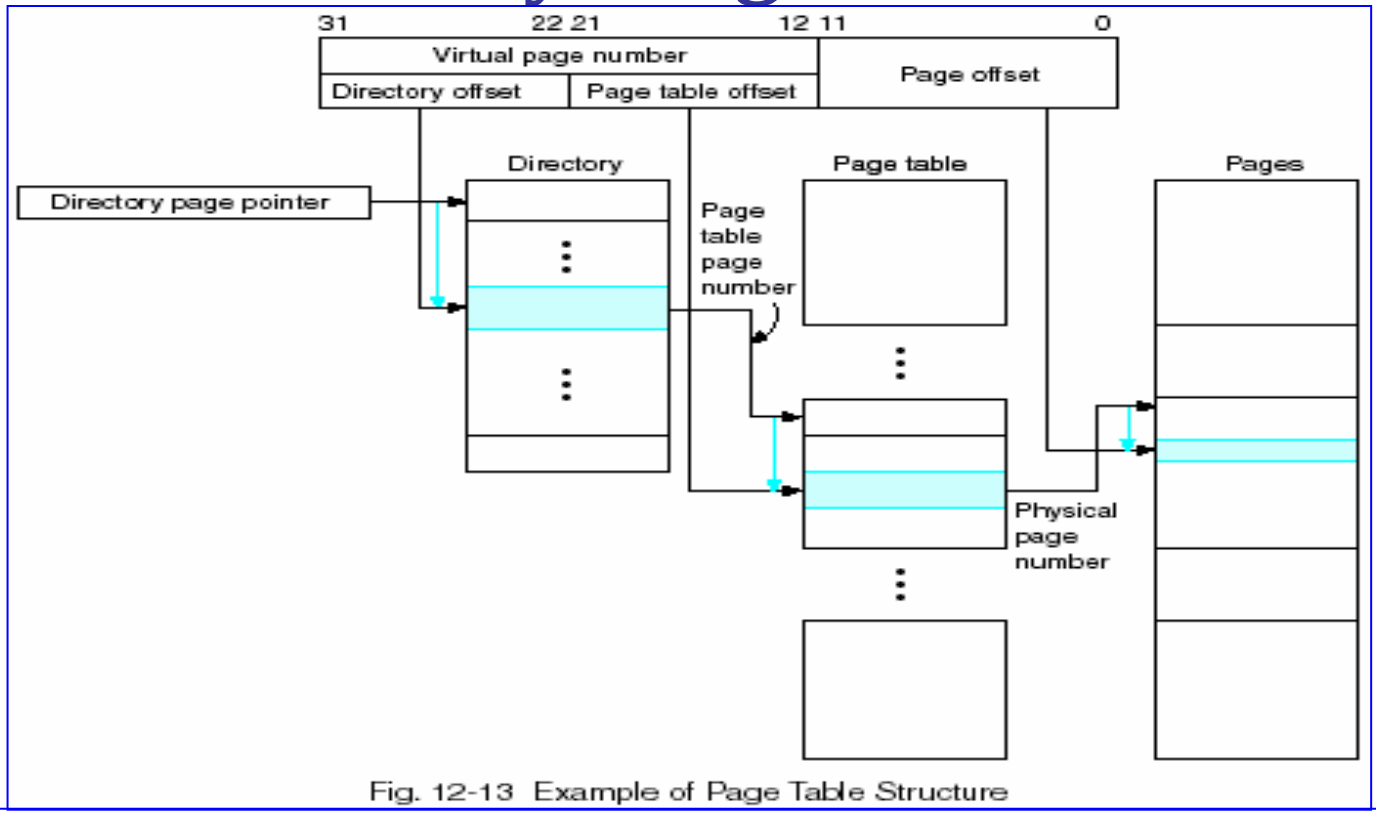


Fig. 12-13 Example of Page Table Structure

- Directory page pointer – a register that points to the location of the directory page in main memory.
- Directory offset – most significant 10 bits of virtual page number used to access the page table.
- Page table offset – least significant 10 bits of virtual page number used to access a page table entry.

Virtual Memory: Translation Lookaside Buffer

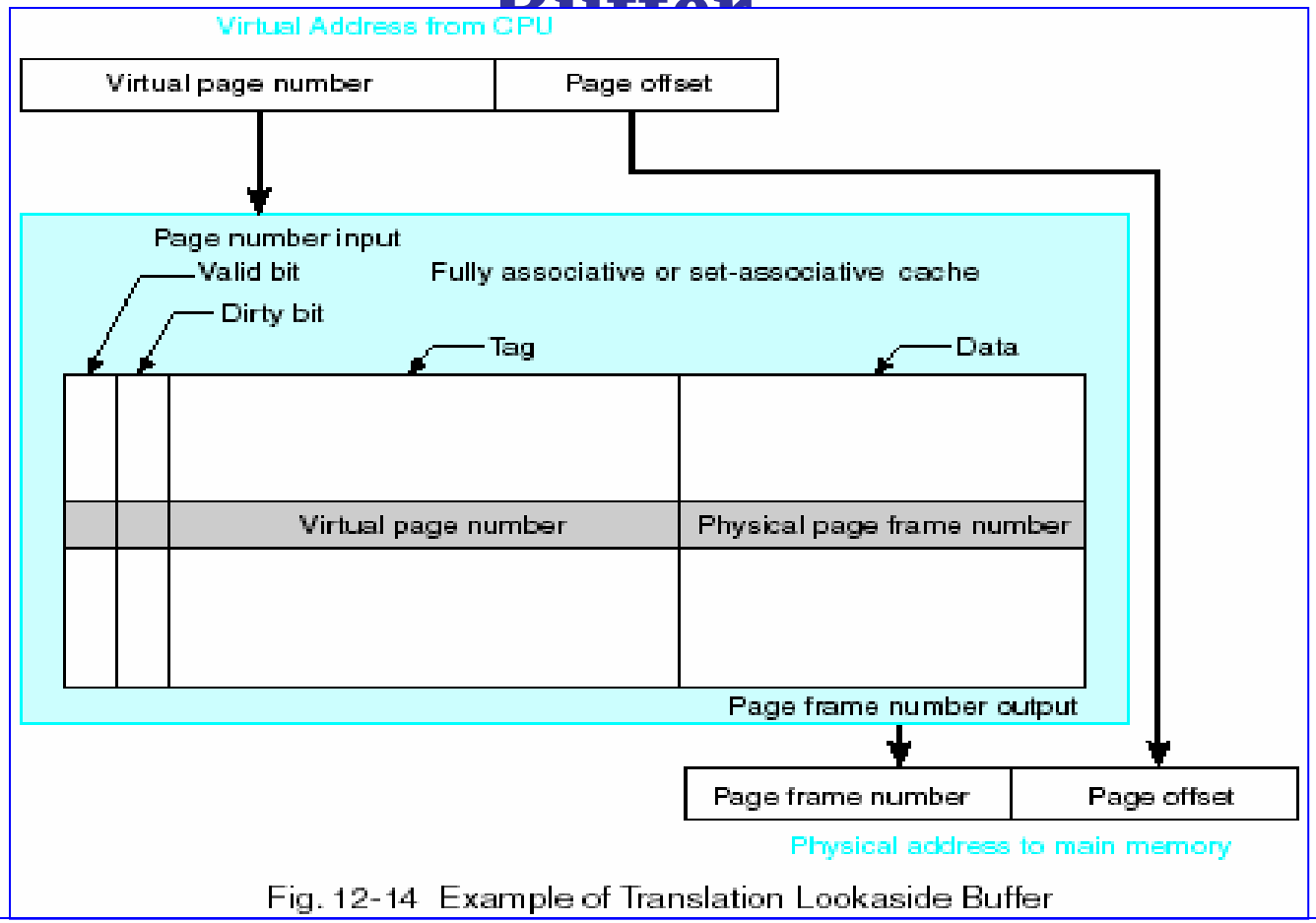


Fig. 12-14 Example of Translation Lookaside Buffer

- TLB – a cache used to speed up the process of translating virtual address into corresponding physical address.
- Holds locations of recently addressed pages to speed access to cache or main memory.