Memory System

The size of a memory is determined by the number of unique addressable memory locations.

$$k \text{ address bits} = 2^k \text{ addresses}$$

Memory Access Time: the time that elapses between the initiation of an operation to transfer a word of data and the completion of that operation. This is referred to as the memory access time
Memory Cycle Time: the minimum time delay required between the initiation of two successive memory operations
RAM: A memory unit is called a random-access memory (RAM) if the access time to any location is the same, independent of the locations address.
Cache memory: a small, fast memory inserted between the larger, slower main memory and the processor. It holds the currently active portions of a program and their data.
Virtual memory: With this technique, only the active portions of a program are stored in the main memory, and the remainder is stored on the much larger secondary storage device. Sections of the program are transferred back and forth between the main memory and the secondary storage device in a manner that is transparent to the application program.
Block Transfers: transfers between the main memory and the cache and between the main memory and the disk do not occur one word at a time. Data are always transferred in contiguous blocks involving tens, hundreds, or thousands of words.

$$\text{bandwidth} = \text{words} \times \frac{\text{bytes}}{word} \times \frac{\text{frequency}}{\text{cycles}}$$

Number of cycles:

1. Async DRAM: $= (RAS + CAS + 1) \times burst\ size$
2. Fast page mode DRAM $= \Big((RAS + CAS + 1) \times 1 + (CAS + 1) \times (burst\ size - 1)\Big)$
3. Synchronous DRAM (SDRAM) $= RAS + CAS + burst\ size$
4. DDR - SDRAM: $= RAS + CAS + \frac{burst\ size}{2}$

Memory Sizes:

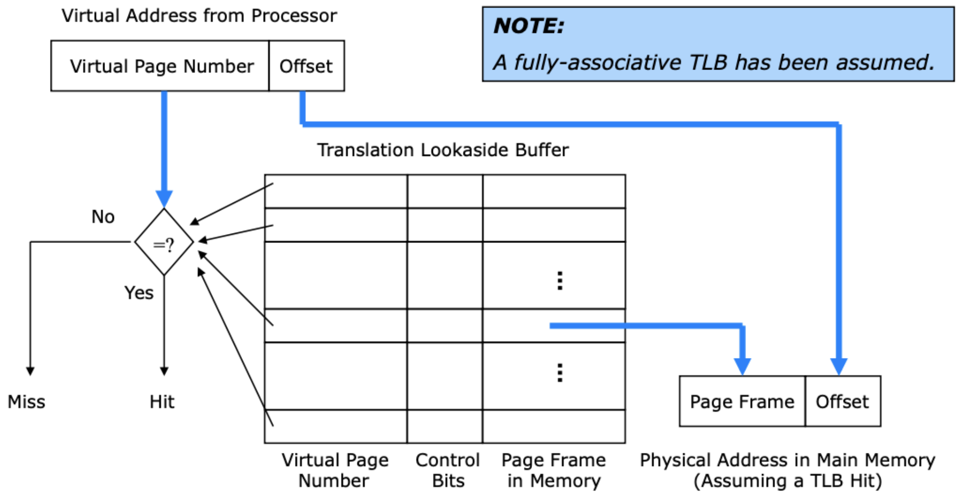
- KiB: $2^{10} = 1024\, bytes$
- MiB: $KiB^2 = 1024^2\ bytes$
- GiB: $KiB^3 = 1024^3\ bytes$
- TiB: $KiB^4 = 1024^4\ bytes$

offset bits $= log_2(Block\ Size)$
The first and the last address can be generated by having offsets 000000, 111111 be concatenated with the block address.
Direct-mapped cache:

- offset bits: log_2(block size)
- index: log_2(number of blocks)
- Tag: number of address bits - (index bits + offset bits)

number of cache lines $= \frac{Cache\ size}{Block\ size}$
N-way set-associative cache:

- offset bits $= log_2$(block size)
- number of sets = number of cache lines / N
- index bits $= log_2$(number of sets)
- Tag = number of address bits - (index bits + offset bits)


Virtual Address from Processor
Virtual Page Number
Offset
NOTE:
A fully-associative TLB has been assumed.
Translation Lookaside Buffer
No
=?
Yes
Miss
Hit
Page Frame
Offset
Virtual Page Number
Control Bits
Page Frame in Memory
Physical Address in Main Memory (Assuming a TLB Hit)

Translation divides address bits into 2 fields

- Lower bits give offset of word within page (page offset)
- Upper bits give virtual page number (vpn)

Page offset = log_2(page size)
# of virtual page number (VPN) bits = #virtual address bits - page offset
# of physical page number (PPN) bits = #physical address bits - page offset
It is possible to improve the performance of virtual memory address translation by moving the page table into the MMU:

- The entire page table is often too large to be stored within the MMU
- A portion of the page table, known as a translation lookaside buffer (TLB) is typically stored in the MMU and holds recently accessed entries of page table
- The TLB is analogous to a cache for page table entries