



BROWN

Computer System Design Lecture 20: Virtual Memory

Prof. R. Iris Bahar
EN164
March 14, 2007

Reading: Appendix C, sections C.4



BROWN

Homeworks and Lab 2

- Mickey has them graded. Stop by Mickey's office later today to pick them up.
- Download of processor for lab 2 is done.
- Please check the website for additional information regarding lab 2



BROWN

Improving DRAM Access Time

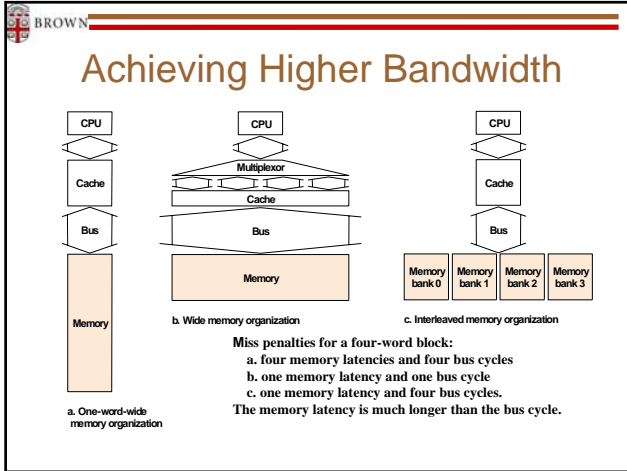
- The column decoder has access to many bits of data
 - Many sequential bits can be forwarded to the CPU without additional row accesses (fast page mode)
- Each word is sent asynchronously to the CPU
 - Every transfer entails overhead to synchronize with the controller
 - By introducing a clock, more than one word can be sent without increasing the overhead
 - Called synchronous DRAM (SDRAM)



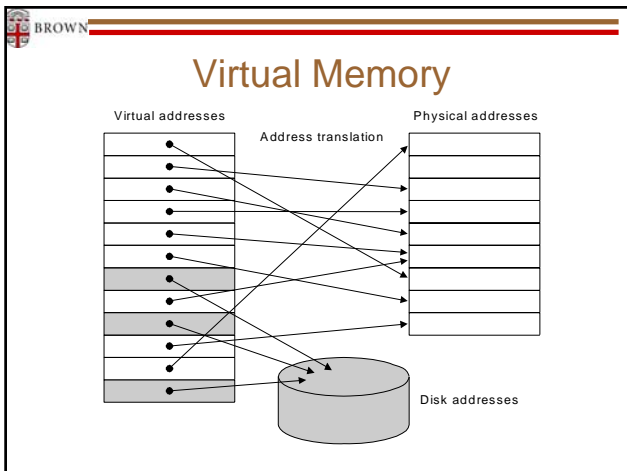
BROWN

Increasing Bus Bandwidth

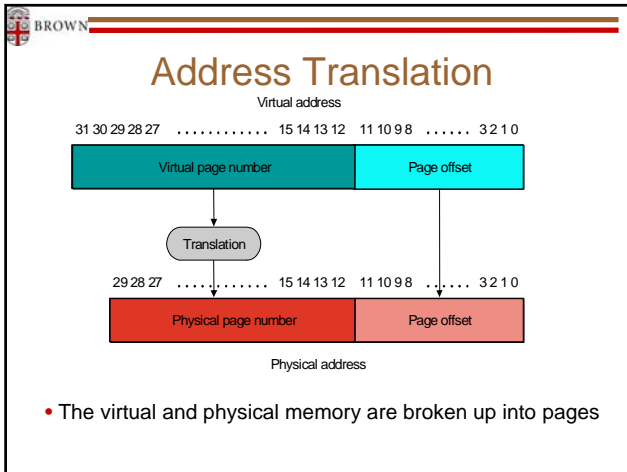
- By increasing the memory width (number of memory chips and the connecting bus), more bytes can be transferred together
 - Increases cost
- Interleaved memory
 - since the memory is composed of many chips, multiple operations can happen at the same time
 - a single address is fed to multiple chips, allowing us to read sequential words in parallel



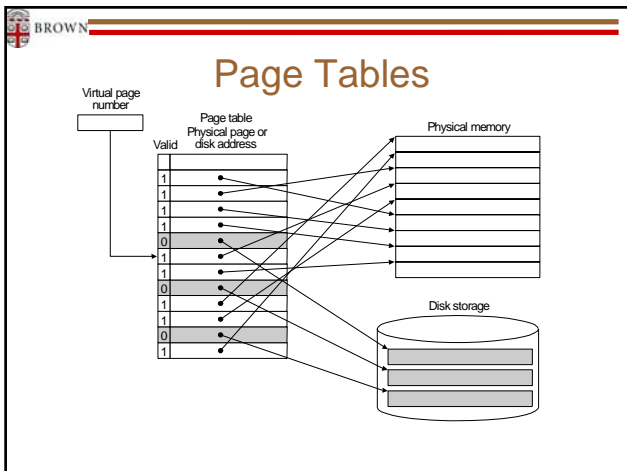
- Virtual Memory**
- Main memory can act as a “cache” for the secondary storage
 - large virtual address space used in each program
 - process places part of its virtual memory in this physical memory and the rest is stored on disk
 - Motivations
 - efficient and safe sharing of memory among multiple programs
 - remove programming burdens of a small main memory
 - Advantages:
 - illusion of having more physical memory
 - program relocation
 - protection



- Pages: Virtual Memory Blocks**
- CPU produces a virtual address
 - translated by a combination of hardware and software to a physical address
 - *Virtual address*: virtual page number and page offset
 - *Physical address*: physical page number and page offset
 - *Page fault*: data is not in memory, retrieve it from disk



- ## Virtual Memory Design
- A virtual memory page can be placed anywhere in physical memory (fully-associative)
 - A page table (indexed by virtual page number) is used for translating virtual to physical page number
 - Huge miss penalty, if data is not found in main memory (page fault)
 - thus pages should be fairly large (4 - 64 kB).
 - Replacement policy is important (LRU is worth the price).
 - Page faults can be handled in SW instead of HW (overhead small compared to the access time to disk)
 - Virtual memory systems use write-back.
 - using write-through is too expensive
 - Dirty bit indicates whether a page needs to be copied back to disk when replaced



- ## Translation Lookaside Buffer
- Since the number of pages is very high, the page table capacity is too large to fit on chip
 - A translation lookaside buffer (TLB) caches the virtual to physical page number translation for recent accesses
 - A TLB miss requires us to access the page table, which may not even be found in the cache
 - two expensive memory look-ups to access one word of data!
 - A large page size can increase the coverage of the TLB and reduce the capacity of the page table, but also increases memory wastage

