

COMP 3221

Microprocessors and Embedded Systems

Lectures 37: Virtual Memory - II

<http://www.cse.unsw.edu.au/~cs3221>

October, 2003

Saeid Nooshabadi

saeid@unsw.edu.au

Some of the slides are adopted from David Patterson (UCB)

COMP3221 lec37-vm-IL.1

Saeid Nooshabadi

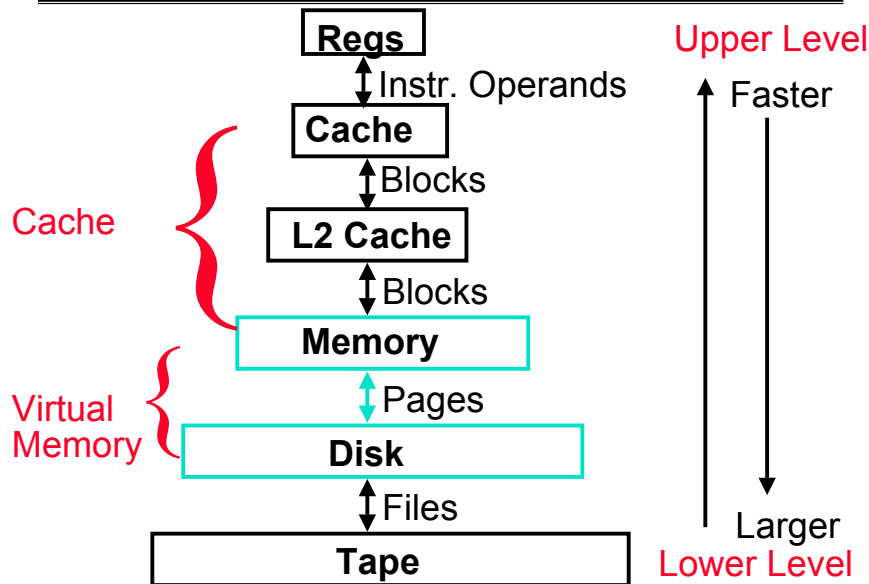
Overview

- Page Table
- Translation Lookaside Buffer (TLB)

COMP3221 lec37-vm-IL.2

Saeid Nooshabadi

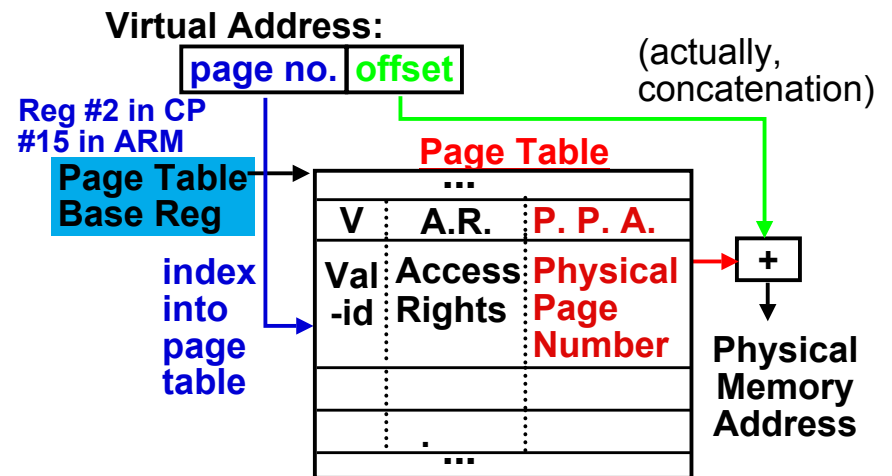
Review: Memory Hierarchy



COMP3221 lec37-vm-IL.3

Saeid Nooshabadi

Review: Address Mapping: Page Table

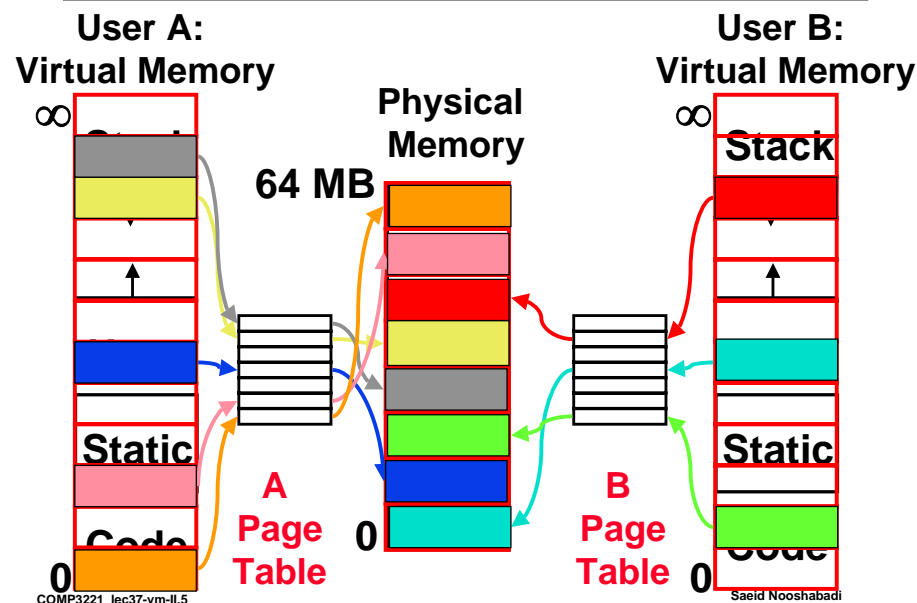


Page Table located in physical memory

COMP3221 lec37-vm-IL.4

Saeid Nooshabadi

Paging/Virtual Memory for Multiple Processes



Analogy

- Book title like **virtual address** (ARM System On-Chip)
- Library of Congress call number like **(QA76.5.F8643 2000)** **physical address**
- Card (or online-page) catalogue like **page table**, indicating mapping from book title to call number
- On card (or online-page) info for book, indicating in local library vs. in another branch like **valid bit** indicating in main memory vs. on disk
- On card (or online-page), available for 2-hour in library use (vs. 2-week checkout) like **access rights**

COMP3221 lec37-vm-IL.6

Saeid Nooshabadi

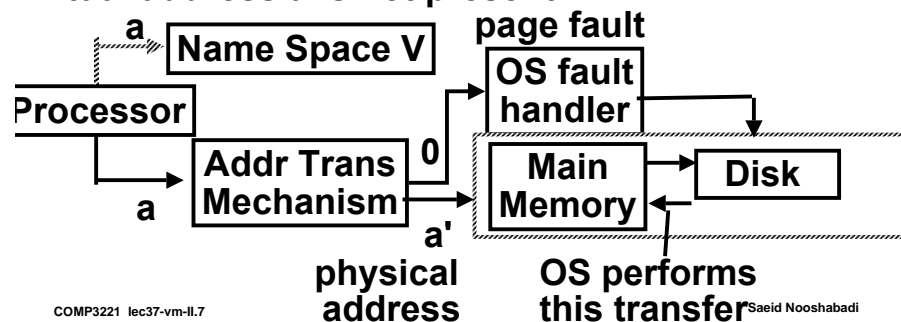
Address Map, Mathematically Speaking

$V = \{0, 1, \dots, n - 1\}$ virtual page address space ($n > m$)

$M = \{0, 1, \dots, m - 1\}$ physical page address space

MAP: $V \rightarrow M \cup \{\emptyset\}$ page address mapping function

MAP(a) = a' if data at virtual address a is present in physical address a' and $a' = \emptyset$ if data at virtual address a is not present in M



Saeid Nooshabadi

Comparing the 2 Levels of Hierarchy

- | | |
|-----------------------|---------------------------|
| ◦ Cache Version | Virtual Memory vers. |
| ◦ Block or Line | Page |
| ◦ Miss | Page Fault |
| ◦ Block Size: 32-64B | Page Size: 4K-8KB |
| ◦ Placement: | Fully Associative |
| Direct Mapped, | |
| N-way Set Associative | |
| ◦ Replacement: | Least Recently Used (LRU) |
| LRU or Random | |
| ◦ Write Thru or Back | Write Back |

COMP3221 lec37-vm-IL.8

Saeid Nooshabadi

Notes on Page Table

- Solves Fragmentation problem: all chunks same size, so all holes can be used
- OS must reserve “**Swap Space**” on disk for each process
- To grow a process, ask Operating System
 - If unused pages, OS uses them first
 - If not, OS swaps some old pages to disk
 - (Least Recently Used to pick pages to swap)
- Each process has its own Page Table
- Will add details, but Page Table is essence of Virtual Memory

COMP3221 lec37-vm-IL.9

Saeid Nooshabadi

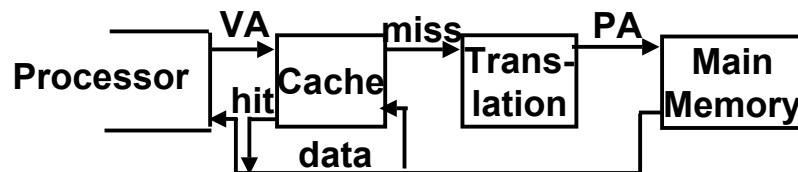
Virtual Memory Problem #1

- **Not enough physical memory!**
 - Only, say, 64 MB of physical memory
 - N processes, each 4GB of virtual memory!
 - Could have 1K virtual pages/physical page!
- Spatial Locality to the rescue
 - Each page is 4 KB, lots of nearby references
 - No matter how big program is, at any time only accessing a few pages
 - “**Working Set**”: recently used pages

COMP3221 lec37-vm-IL.10

Saeid Nooshabadi

Virtual Address and a Cache (#1/2)

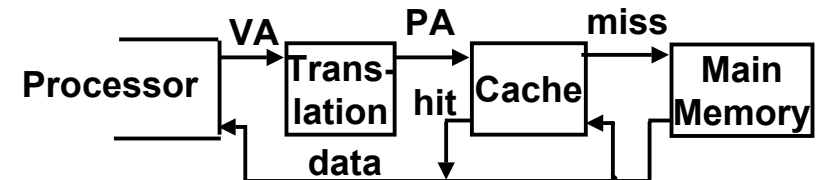


- Cache operates on Virtual addresses.
- ARM Strategy
 - The advantage: If in cache the translation is not required.
 - Disadvantage: Several copies of the the same physical memory location may be present in several cache blocks. (**Synonyms** problem). Gives rise to some complications!

COMP3221 lec37-vm-IL.11

Saeid Nooshabadi

Virtual Address and a Cache (#2/2)



- Cache typically operates on physical addresses on most other systems.
- Address Translation (Page Table access) is another memory access for each program memory access!
 - Accessing memory for Page Table to get Physical address → (**Slow Operation**)
 - Need to fix this!

COMP3221 lec37-vm-IL.12

Saeid Nooshabadi

Reading Material

- ° Steve Furber: ARM System On-Chip; 2nd Ed, Addison-Wesley, 2000, ISBN: 0-201-67519-6. **Chapter 10.**

Virtual Memory Problem #2

- ° Map every address \Rightarrow 1 extra memory accesses for every memory access
- ° Observation: since locality in pages of data, must be locality in virtual addresses of those pages
- ° Why not use a cache of virtual to physical address translations to make translation fast? (small is fast)
- ° For historical reasons, this cache is called a Translation Lookaside Buffer, or TLB

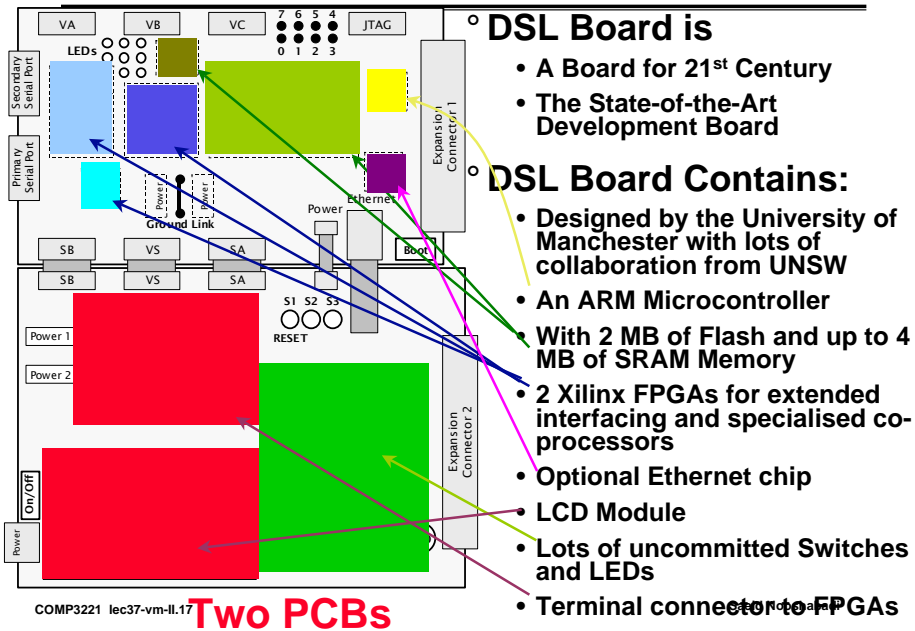
Typical TLB Format

Virtual Address	Physical Address	Dirty	Ref	Valid	Access Rights

- TLB just a cache on the page table mappings
- TLB access time comparable to cache (much less than main memory access time)
- Ref: Used to help calculate LRU on replacement
- Dirty: since use write back, need to know whether or not to write page to disk when replaced

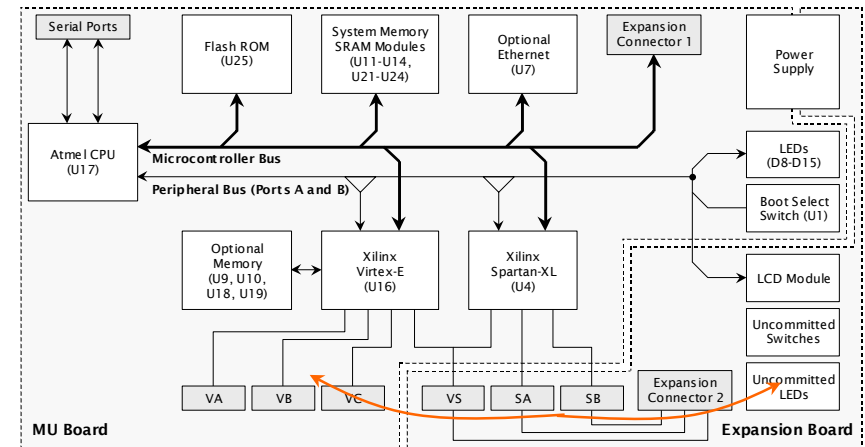
° **PROJECTS IN DIGITAL
HARDWARE DESIGN FOR 4th
Year**

Digital Systems Laboratory Hardware



Two PCBs

DSL MU Hardware Block Diagram



COMP3221 lec37-vm-IL.18

Saeid Nooshabadi

Projects with Digital Systems Lab Board

- **Project 1: Design of a Vector Floating Point Co-Processor for ARM Core:**
- **Aim:** The Aim of this project is to build Floating Point Vector Processor On the Vertex FPGA.
- **Degree of Difficulty:** Hard, and Challenging
- **Ability:** Number Representation, Digital System Design, DSP

COMP3221 lec37-vm-IL.19

Saeid Nooshabadi

Projects with Digital Systems Board

- **Project 2: Design of a Fixed Point DSP for ARM Core:**
- **Aim:** The Aim of this project is to build DSP Optimised Processor (Single Cycle MAC Processor/ Distributed Arithmetic) On the Vertex FPGA.
- **Degree of Difficulty:** Hard, and Challenging
- **Ability:** Number Representation, Digital System Design, DSP

COMP3221 lec37-vm-IL.20

Saeid Nooshabadi

Projects with Digital Systems Board

- **Project 3: Development of a Simple Multi tasking Operating System**
- **Aim:** The Aim of this project is Development of a Simple Multi tasking Operating System with Simple Virtual Memory Protection for on-board program monitoring and debugging.
- **Degree of Difficulty:** Moderate, and some Challenges
- **Ability:** Software Development, Basic Operating Systems
- **Application:** Our Undergraduate/Post Graduate Teaching
- **Advantage:** Make yourself Immortal!

COMP3221 lec37-vn-ll.21

Saeid Nooshabadi

Projects with Digital Systems Board

- **Project 4: Interfacing GNU debugging tool with on-board emulator (Komodo)**
- **Aim:** The Aim of this project is to Interface GNU debugging tool with on-board emulator program to facilitate source-level debugging and monitoring.
- **Degree of Difficulty:** Moderate to Hard with some Challenges
- **Ability:** Software Development, Basic Operating Systems
- **Application:** Our Undergraduate/Post Graduate Teaching
- **Advantage:** Make yourself Immortal!

COMP3221 lec37-vn-ll.22

Saeid Nooshabadi

Projects with Digital Systems Board

- **Project 5: Porting of uClinux to DSL Board**
- **Aim:** The Aim of this project is to port real-time Operating System uClinux to DSL Board.
- **Degree of Difficulty:** Hard, and Challenging
- **Ability:** Software Development, Basic Operating Systems
- **Start:** <http://www.uclinux.org>

COMP3221 lec37-vn-ll.23

Saeid Nooshabadi

Projects with Digital Systems Board

- **Project 6: Design of an embedded Internet enabled device for remote control and monitoring**
- **Aim:** The aim of this project is to design an embedded interface device using programmable microcontrollers and FPGAs, to wireless devices in one hand and modem/lan on the other hand. The unit collects the data through the wireless picodevices and in turn transfers the data via modem or lan to a remote server via TCP/IP stacks. It should also be possible to control the picodevices remotely. The challenge is to build the various software layers, on a realtime operating system like eCos to do the task.
- **Degree of Difficulty:** Hard, and Challenging
- **Ability:** Software Development, Basic Operating Systems, basic Hardware Interfacing

COMP3221 lec37-vn-ll.24

Saeid Nooshabadi

Projects with Digital Systems Board

- **Project 7: Design of an embedded Internet enabled device for remote control and monitoring**
- **Aim:** The aim of this project is to design an embedded interface device using programmable microcontrollers and FPGAs, to control and monitor the viewing of cable TV channels. This embedded unit controls a TV tuner, in a real time fashion, based on the control information it receives remotely. The unit is connected to the internet via a cable modem. It can be controlled and monitored via a remote device (a PC). The challenge is to build the various software layers, on a realtime operating system like eCos to do the task.
- **Degree of Difficulty:** Hard, and Challenging
- **Ability:** Software Development, Basic Operating Systems, basic Hardware Interfacing

COMP3221 lec37-vm-IL.25

Saeid Nooshabadi

Projects with Digital Systems Board

- **Project 8: PS/2 and USB Controller For DSL Board**
- **Aim:** The aim of this project is to design an PS/2 and USB Controller using the on-board FPGA chips.
- **Degree of Difficulty:** Hard, and Challenging
- **Ability:** Hardware Development, Basic Software development

COMP3221 lec37-vm-IL.26

Saeid Nooshabadi

Projects with Digital Systems Board

- **Project 9: Frame Grabber Device for CMOS Digital Camera**
- **Aim:** The aim of this project is to build a high resolution web camera using a Kodak KAC-1310 CMOS image sensor. The pixel data would be buffered in SRAM/SDRAM, compressed and uploaded through the Internet interface for display on a PC.
- **Degree of Difficulty:** Very Hard, and Challenging
- **Ability:** DSP, Hardware Development, Basic Software development

COMP3221 lec37-vm-IL.27

Saeid Nooshabadi

Projects with Digital Systems Board

- **Project 10: Audio Signal Processing**
- **Aim:** The aim of this project is to interface a codec to a DSP Audio Signal Processor/ Compressor built on the on-board FPGAs
- **Degree of Difficulty:** Moderate to Hard, and Challenging
- **Ability:** DSP, Hardware Development, Basic Software development

COMP3221 lec37-vm-IL.28

Saeid Nooshabadi

Things to Remember (#1/2)

- Apply Principle of Locality Recursively
- Reduce Miss Penalty? add a (L2) cache
- Manage memory to disk? Treat as cache
 - Included protection as bonus, now critical
 - Use **Page Table** of mappings vs. tag/data in cache
- Virtual memory to Physical Memory Translation too slow?
 - Add a cache of Virtual to Physical Address Translations, called a **TLB**

Things to Remember (#2/2)

- Virtual Memory allows protected sharing of memory between processes with less swapping to disk, less fragmentation than always swap or base/bound
- Spatial Locality means Working Set of Pages is all that must be in memory for process to run fairly well

Things to Remember

- Spatial Locality means Working Set of Pages is all that must be in memory for process to run fairly well
- Virtual memory to Physical Memory Translation too slow?
 - Add a cache of Virtual to Physical Address Translations, called a **TLB**
 - TLB to reduce performance cost of VM