Tárgytematika / Course Description
Computer Architectures

GKNM_TATA112

Tárgyfelelős neve / Teacher's name: dr. Lencse Gábor Sándor
Félév / Semester: 2026/27/1
Beszámolási forma / Assesment: Vizsga
Tárgy heti óraszáma / Teaching hours(week): 2/2/0
Tárgy féléves óraszáma / Teaching hours(sem.): 0/0/0

OKTATÁS CÉLJA / AIM OF THE COURSE

The aim of the course is to teach the design, operation and features of modern computers. Knowing the main characteristics of the hardware enables the development of efficient software that makes better use of the resources of computers.

TANTÁRGY TARTALMA / DESCRIPTION

Topic: Introduction

Introduction to information processing models. Control flow architectures: Neumann architecture, Harvard architecture, modified Harvard architecture. Characteristics of instruction sets, CISC-RISC strategies.

Topic: Input-output peripherals

Dedicated I/O instructions and memory mapped peripheral management. Flow control. Processing peripheral signals: polling, interrupt, interrupt in multiprocessor environment, interrupt moderation. Processor offloading: DMA, I/O processor. Interconnections: bus, point-to-point, serial, parallel, timing, arbitration. Single-bus, multi-bus, bridge-based systems. PCI, PCI Express and USB interfaces.

Topic: Mass storage devices

Operation of hard disks: concept and parts of a sector, zoned bit recording. Main components of data transfer service times. Queuing and reordering of commands. How SSD devices work: concept and role of pages, blocks. Management and side effects of read/write operations. Causes and importance of aging. Functions of SSD controllers.

Topic: Memory

Synchronous DRAM based memory systems: concept and operation of memory controller, module, rank, bank. DRAM commands and their timing, out-of-order execution of commands. Virtual memory management: address translation, TLB, page table implementations, single level and hierarchical page tables. Cache memory: role of locality principles, cache organization, cache organization and virtual memory management. Cache content management: cache pollution prevention, block prefetching, block replacement algorithms. Locality-aware programming techniques.

Topic: Processor

Pipeline instruction processing. Concepts and handling of hazards. Implementation of a simple 5 stage pipeline. Exception handling, precise interrupt handling. Handling arithmetic operations with different delays. Dynamic scheduling (out-of-order instruction execution). The concept of precedence graph and data flow-based instruction scheduling. The role and implementation of the instruction queue, register renaming, and the reorder buffer. The Tomasulo algorithm. Wide pipelines: superscalar, VLIW and EPIC architectures. Branch prediction: predicting the outcome of the jump condition and the jump target address. Branch prediction-aware programming. Attacks against speculative execution.

Topic: Parallel processing

Data parallelism: vector processors, SIMD instruction set extensions. Multiprocessor systems: notion of explicit parallelism, multi-threaded processors, classification of multiprocessor systems, interconnection networks. Shared memory management: cache coherence protocols, memory consistency problems and solutions.

The material of the classroom practices:

- Review of digital design basics through a simple hardware-software design project
- Calculating the relative load of the CPU for peripheral processing with polling and interrupt based I/O peripheral management
- HDD latency and throughput calculations, manual walkthrough of SSD write management algorithms
- Memory management: DRAM command scheduling, command latency and throughput calculations, exercises related to virtual memory management with and without TLB
- Cache memory: practicing cache organization, cache error rate calculation and code optimization for simple C programs
- Pipeline scheduling: scheduling low-level programs with various instruction pipelines, determining optimal instruction order
- Advanced pipeline techniques: dependency analysis, elimination of anti-dependencies by register renaming, following the operation of branch predictors using simple C programs

---

SZÁMONKÉRÉSI ÉS ÉRTÉKELÉSI RENDSZERE / ASSESMENT'S METHOD

The students are strongly advised to participate all lectures and classroom practices because there are no

lectures notes for the course and it is rather difficult to understand the material from the slides without explanation.

There are two mid-term tests. The students must pass at least one mid-term test to get the signature. Those who achieve at least grade 4 at both mid-term tests will be entitled for a recommended grade (4 or 5 depending on their results).

Those who acquire the signature but are not entitled for a recommended grade (or do not accept it) may take an exam.

KÖTELEZŐ IRODALOM / OBLIGATORY MATERIAL

Lecture and classroom practice slides published after the each class on the web page of the course available at: https://www.tilb.sze.hu/cgi-bin/tilb.cgi?0=m&1=targyak&2=GKNM_TATA112

AJÁNLOTT IRODALOM / RECOMMENDED MATERIAL

Bruce Jacob, Spencer W. Ng, Samuel Rodriguez: Memory Systems, Morgan Kaufmann Publishers, 2008.

Lee Hutchinson: Solid-state revolution: in-depth on how SSDs really work, Ars Technica, 2012, online

William Stallings: Computer Organization and Architecture, 8th Edition, 2010.

Jean-Loup Baer: Microprocessor Architecture, Cambridge University Press, 2010.

David A. Patterson, John L. Hennessy: Computer Organization and Design, Morgan Kaufmann Publishers, 2009.

Ravi Budruk, Don Anderson, Tom Shanley: PCI Express System Architecture, Addison-Wesley, 2003.

Don Anderson: Universal Serial Bus System Architecture, Addison-Wesley, 2001.