

SYLLABUS
pentru disciplina:

“Fundamente de Ingineria Calculatoarelor”

FACULTATEA: AUTOMATICA SI CALCULATOARE
DOMENIUL / SPECIALIZAREA: CALCULATOARE SI TEHNOLOGIA INFORMATIEI/
CALCULATOARE

Anul de studii: III

Semestrul I

Titularul cursului: Prof.dr.ing. Mircea VLADUTIU
Colaboratori: S.I. Lucian PRODAN, s.I.Mihai UDRESCU, As. Versavia ANCUSA, Prep. Alexandru AMARICAI, Prep. Oana Boncalo

Numar de ore/saptamana/Verificarea/Credite					
Curs	Seminar	Laborator	Proiect	Examinare	Credite
2	-	1	1	E	5

A. OBIECTIVELE CURSULUI

Cursul se concentreaza asupra modelelor si metodelor utilizate in analiza si designul sistemelor tolerante la defecte si a sistemelor cu fiabilitate ridicata. Toleranta la defecte constituie un obiectiv primordial in implementarea sistemelor specializate pe aplicatii precum retelele de comunicatii si controlul zborului. Cursul urmareste familiarizarea cu conceptele de baza si state-of-the-art conexe analizei si designului sistemelor tolerante la defecte. Vor fi studiate sisteme tolerante la defecte existente, impreuna cu tehnicile aplicate. Aspecte practice ale tolerantei la defectare vor fi observate prin intermediul lucrarilor practice de laborator, a modelarilor si simularilor in VHDL.

B. SUBIECTELE CURSULUI

Capitolul 1. Metrici de evaluare a performantei in sistemele de calcul (4 ore)

- 1.1 Tendinte in tehnologia si utilizarea sistemelor de calcul
- 1.2. Masurarea performantei. Programe benchmark
- 1.3 Principii cantitative in designul sistemelor de calcul
- 1.4. Performanta procesorului si componentele sale
- 1.5. Ierarhii de memorii swi implicatia acestora in performanta

Capitolul 2. Controlul pipeline in sistemele de calcul (12 ore)

- 2.1. Presentare generală
- 2.2. O arhitectură DLX cu pipeline
- 2.3. Problemele fundamentale: hazardurile
- 2.4. Hazardul structural
- 2.5 Hazardul de date
- 2.6. Hazardul de control
- 2.7. Excepții în DLX
- 2.8. Extinderea pipeline-ului pentru operații multiciclu
- 2.9. Probleme de design privind setul de instrucții
- 2.10. Pipeline-ul MIPS R4000
- 2.11. Concluzii

Capitolul 3. Memorii cache si ierarhii de memorii (12 ore)

- 3.1. Memorii cache
 - 3.1.1. Replasarea blocurilor și probleme specifice. Structura cache la Alpha AXP 21064
 - 3.1.3. Criterii de performanță. Ameliorarea performanței cache-urilor
- 3.2. Reducerea cache misses
 - 3.2.1. Mărirea dimensiunii blocurilor VS mărirea asociativității
 - 3.2.3. Victim cache si cache set-asociativ
 - 3.2.5. Hardware pre-fetching pentru instrucții și date
 - 3.2.6. Optimizări de compilator
- 3.3. Reducerea cache miss penalty
- 3.4. Reducerea hit time

- 3.5. Memoria principală. Tehnologii și organizare
- 3.6. Memoria virtuală. Replasarea blocurilor de memorie și translatarea adreselor
- 3.7. Protecția în memoria virtuală
- 3.8. Ierarhia de memorie la Alpha AXP 21064

C. SUBIECTELE APLICATIILOR (laborator, seminar, proiect)

1. Alocarea resurselor hardware în timpul execuției unei secvențe de cod. Identificarea situațiilor de dependență. **(2 ore)**
2. Prezentarea pipeline-ului DLX pentru operații cu întregi și identificarea hazardurilor. **(2 ore)**
3. Încărcarea pipeline-ului pentru o secvență de cod cu operații în virgulă flotantă. **(2 ore)**
4. Desfășurarea buclelor de cod. Identificarea și eliminarea ciclurilor de stall datorate hazardurilor. **(2 ore)**
5. Analiza dependențelor existente într-o secvență de cod. Listarea problemelor apărute și posibilități de rezolvare hardware-software. **(2 ore)**
6. Analiza unei secvențe de cod prin metoda scoreboard. **(2 ore)**
7. Analiza unei secvențe de cod prin metoda Tomasulo. **(2 ore)**
8. Desfășurarea de bucle pentru varianta DLX superscalar. Identificarea situațiilor de dependență. **(2 ore)**
9. Analiza algoritmilor de replasare pentru memorii cache. **(2 ore)**
10. Evaluarea parametrilor cache pentru o secvență de cod, într-o ierarhie de memorie dată. **(2 ore)**
11. Analiza unor scenarii de execuție pentru o secvență de cod rulată pe mașini cu ierarhii de memorie diferite. **(2 ore)**
12. Translatarea adreselor și probleme de replasare într-o ierarhie complexă de memorie. **(2 ore)**
13. Analiza tehnicilor VLIW. **(2 ore)**

D. BIBLIOGRAFIE

1. David A. Patterson, John L. Hennessy: "Computer Architecture. A Quantitative Approach", 3rd Edition, Morgan Kaufmann Publishers Inc., 2002
2. William Stallings: "Computer Organization and Architecture. Designing for Performance", 7th Edition, Prentice Hall International, 2005
3. John L. Hennessy and David A. Patterson: "Computer Organization and Design: The Hardware/Software Interface", 3rd Edition Morgan Kaufmann Publishers Inc., 2007

E. PROCEDURA DE EVALUARE

Cursul se încheie cu un examen scris la sfârșitul semestrului cu durata de 2 ore și care va trata un număr de subiecte pe baza celor tratate la curs (2-4 subiecte). În nota finală se iau în considerare activitatea la laborator (40%) și nota la examenul scris (60%).

F. COMPATIBILITATE INTERNATIONALA

*University of California at Berkeley, Computer Science 252 – Graduate Computer Architecture
Swansea University (UK), High Performance Microprocessors Course
MIT, 6.033 – Computer System Engineering (SMA 5501), Spring 2005*

Data:

DIRECTOR/SEF DEPARTAMENT/CATEDRA

TITULAR DE DISCIPLINĂ,