

OBIECTIVELE DISCIPLINEI CN2

- Aprofundarea cunostintelor care privesc arhitectura sistemelor de calcul, inclusiv arhitectura setului de instructiuni.
- Studiul principalelor tipuri de procesoare: CISC, RISC si al metodelor de evaluare a performantei acestora (SPECint si SPECfp).
- Proiectarea si implementarea unui procesor de tip RISC, cu unitate de comanda conventionala si microprogramata.
- Proiectarea unui procesor care opereaza intr-un singur ciclu de ceas.
- Proiectarea unui procesor care efectueaza o instructiune in mai multe cicluri de ceas.
- Operarea in banda de asamblare; Operatiile in cadrul sectiunilor Benzii de Asamblare; Problemele Benzii de Asamblare: Hazardele structural, de date, de control; Performanta Benzii de Asamblare.
- Structura si arhitectura procesoarelor modern.
- Memoria principala. Tendinte in domeniul tehnologiei memoriilor. Memorii statice. Memorii dinamice. Organizare memoriei principale. Parametri Performante.
- Organizarea memoriei in sistemele de calcul. Memorii asociative. Memoria intermediara. Memoria Virtuala. Organizarea pe blocuri a memoriei principale.

COMPETENTE SPECIFICE

Dupa parcurgerea cursului CN2, inclusiv Laboratorul, studentii vor fi capabili:

- sa evalueze in mod obiectiv performantele diferitelor tipuri de procesoare , cu arhitecturi si structuring diferite;
- sa proiecteze logic, sa simuleze si sa implementeze in tehnologia FPGA procesoare care opereaza intr-un singur ciclu de ceas, procesoare care efectueaza o instructiune in mai multe cicluri de ceas., procesoare care opereaza in banda de asamblare.
- sa compare si sa evalueze diferitele tipuri de organizare a memoriei sistemelor numerice;
- sa inteleaga solutiile arhitecturale si structurale folosite in procesoarele modern;

1. CONTINUTUL TEMATIC

Curs: CALCULATOARE NUMERICE 2

1. Arhitectura, Organizarea si Structura Calculatoarelor Numerice

- 1.1 Arhitectura calculatorului: definitie, exemple
- 1.2 Arhitectura si Ingineria Calculatoarelor numerice
- 1.3 Organizarea calculatorului: definitie, exemple.
- 1.4 Structura calculatorului: definitie, exemple.
- 1.5 Arhitectura Setului de Instructiuni, exemple.

2. Proiectarea unui procesor simplu de tip RISC. (DLX/MIPS)

- 2.1 Specificatii de proiectare.
- 2.2 Organizarea procesorului pentru operarea in virgula fixa.

- 2.3 Resursele hardware.
- 2.4 Formatele instructiunilor.
- 2.5 Moduri de adresare.
- 2.6 Descrierea operatiilor efectuate de catre instructiuni
- 2.7 Performanta procesorului; SPEC-uri, exemple
- 2.8 Probleme.

3. Implementarea procesorului DLX/MIPS

- 3.1 Controlul operatiilor.
- 3.2 Comanda cablata.
- 3.3 Tehnici de optimizare.
- 3.4 Implementarea intreruperilor
- 3.5 Descrierea detaliata a grupurilor de instructiuni.
- 3.6 Performanta comenzii cablate.
- 3.7 Comanda microprogramata
- 3.8 Structura microinstructiunii.
- 3.9 Tipuri de microoperatii/operatii elementare.
- 3.10 Prezentarea microprogramului care interpreteaza setul de instructiuni al procesorului DLX/MIPS.
- 3.11 Performanta comenzii microprogramate. Solutii de imbunatatirea performantei.

4. Proiectarea unui procesor care opereaza intr-un singur ciclu de ceas.

- 4.1 Etapele proiectarii
- 4.2 Componentele unitatii de executie.
- 4.3 Metodologia de sincronizare.
- 4.4 Descrierea detaliata a derularii instructiunilor la nivel logic.
- 4.5 Descrierea detaliata a derularii instructiunilor la nivelul transferurilor intre registre.
- 4.6 Stabilirea operatiilor elementare.
- 4.7 Proiectarea unitatii de comanda sub forma cablata
- 4.8 Calculul perioadei ceasului pentru cazul cel mai defavorabil al derularii unei instructiuni.
- 4.9 Concluzii.

5. Proiectarea unui procesor care efectueaza o instructiune in mai multe cicluri de ceas.

- 5.1 Schema bloc
- 5.2 Etapele derularii diferitelor tipuri de instructiuni.
- 5.3 Abordari in proiectare: top-down, botom-up, mixt
- 5.4 Limitarile principale ale procesorului care opereaza intr-un singur ciclu de ceas.
- 5.5. Proiectarea pe etape.
- 5.6 Unitatea de comanda
- 5.7 Evaluarea performantei
- 5.8. Anexe
- 5.9 Intrebari.

6. Operarea in banda de asamblare.

- 6.1 Introducere
- 6.2 Bnda de asamblare pentru DLX

- 6.3 Operatiile in cadrul sectiunilor Benzii de Asamblare
- 6.4 Problemele Benzii de Asamblare: Hazardele
- 6.5 Hazarde structurale
- 6.6 Hazarde de date
- 6.7 Hazarde de control
- 6.8 Solutii pentru hazarde
- 6.9 Performanta Benzii de Asamblare.

7. Memoria principala.

- 7.1 Tendinte in domeniul tehnologiei memoriilor
- 7.2 Terminologia utilizata
- 7.3 Memorii statice
- 7.4 Memorii dinamice
- 7.5 Organizare memoriei principale. Parametri
- 7.6 Performante.
- 7.7 Concluzii.

8. Organizarea memoriei in sistemele de calcul

- 8.1 Generalitati
- 8.2 Organizarea ierarhica a memoriei
- 8.3 Memorii asociative
- 8.4 Memoria intermediara
- 8.5 Proiectarea unui subsistem de memorie intermediara
- 8.6 Tipuri de organizare pentru memoria intermediara/cache
- 8.7 Memoria Virtuala, organizare
- 8.8 Proiectarea unui subsistem de memorie virtuala.
- 8.9 Ansamblul: memorie intermediara, memorie principala, memorie virtuala
- 8.10 Organizarea pe blocuri a memoriei principale.
- 8.11 Proiectarea unui subsistem de memorie organizata pe blocuri.