

FIȘA DISCIPLINEI

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Petrol-Gaze din Ploiești
1.2. Facultatea	Inginerie Mecanică și Electrică
1.3. Departamentul	Automatică, Calculatoare și Electronică
1.4. Domeniul de studii universitare	Calculatoare și Tehnologia Informației
1.5. Ciclul de studii universitare	Licență
1.6. Programul de studii universitare	Calculatoare

2. Date despre disciplină

2.1. Denumirea disciplinei	STRUCTURA ȘI ORGANIZAREA CALCULATOARELOR
2.2. Titularul activităților de curs	Șef lucr. dr. ing. Iulian Vlad
2.3. Titularul activităților seminar/laborator	Șef lucr. dr. ing. Iulian Vlad
2.4. Titularul activității proiect	-
2.5. Anul de studiu	3
2.6. Semestrul *	5
2.7. Tipul de evaluare	E
2.8. Categoria formativă** / regimul*** disciplinei	DD/DOB

* numărul semestrului este conform planului de învățământ;

** DF - Discipline fundamentale; DS - discipline de specializare; DC - discipline complementare

*** obligatorie/impusă = DOB; opțională = DOP; facultativă = DFA

3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1. Număr de ore pe săptămână	5	din care: 3.2. curs	3	3.3. Seminar/laborator	2	3.4. Proiect	
3.5. Total ore din planul de învățământ	70	din care: 3.6. curs	42	3.7. Seminar/laborator	28	3.8. Proiect	
3.9. Total ore studiu individual (studiu după suport de curs, bibliografie și notițe, documentare suplimentară în bibliotecă, pe platformele electronice de specialitate, pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri)							55
3.10. Total ore pe semestru							125
3.11. Numărul de credite							5

4. Condiții (acolo unde este cazul)

4.1. de curriculum	<ul style="list-style-type: none"> ➤ Calculatoare Numerice, ➤ Proiectare logică ➤ Programarea calculatoarelor și limbaje de programare
4.2. de desfășurare a cursului	<ul style="list-style-type: none"> ➤ Sală de curs, dotată cu tablă, laptop, videoproiector ➤ Online: Conexiune Internet, Google Classroom, Meet, Forms, Sheets, E-learning, etc; ➤ Cursul se desfășoară în format clasic: predare, urmată de dezbateri
4.3. de desfășurare a seminarului/laboratorului	<ul style="list-style-type: none"> ➤ Sală de laborator dotată cu tablă, computere, videoproiector și software adecvat (Vivado ML Standard Edition); ➤ Plăci de dezvoltare Nexys A7-100T ➤ Online: Conexiune Internet, Google Classroom, Meet, Forms, Sheets, E-learning, etc.

5. Competențe specifice acumulate și rezultatele învățării* care stau la baza acestora

Competențe profesionale	Rezultatele învățării*
<p>1. Înțelegerea și aplicarea principiilor fundamentale ale arhitecturii calculatoarelor, inclusiv structura internă a acestora, unitățile de procesare, memoriile și interfețele de comunicație</p>	<p>C1: Studentul va cunoaște structura internă a unui calculator, inclusiv componentele hardware fundamentale: CPU, memorie, unități de stocare, și interfețele de comunicație.</p> <p>C2: Studentul va înțelege diferitele tipuri de arhitecturi ale calculatoarelor (ex. arhitectura Von Neumann, arhitectura Harvard, etc.) și rolul fiecărei componente hardware în execuția unui program.</p> <p>A1: Studentul va fi capabil să explice modul în care datele sunt procesate în cadrul unui sistem de calcul, inclusiv fluxul de informații între CPU, memorie și dispozitivele periferice.</p> <p>A2: Studentul va fi capabil să analizeze și să aplice tehnici de optimizare a performanței în cadrul arhitecturilor de calcul, în special în ceea ce privește memoriile cache, pipeline-ing și paralelismul.</p> <p>RA1: Studentul va demonstra autonomie în analiza și compararea diferitelor tipuri de arhitecturi și va putea alege soluții adecvate pentru proiecte de implementare hardware sau software în domeniul arhitecturii calculatoarelor.</p> <p>RA2: Studentul va respecta principiile etice în dezvoltarea și implementarea de soluții hardware/software, asigurându-se că designul arhitectural respectă standardele de performanță și fiabilitate necesare.</p>
<p>2. Cunoașterea și utilizarea limbajelor, mediilor și tehnicilor de programare pentru investigarea arhitecturii calculatoarelor</p>	<p>C1: Studentul va cunoaște limbajele de programare utilizate pentru investigarea arhitecturii calculatoarelor, cum ar fi: C, C++, VHDL și Verilog.</p> <p>C2: Studentul va înțelege mediile de dezvoltare și instrumentele de simulare necesare pentru analiza performanței și comportamentului arhitecturilor calculatoarelor (ex. Vivado).</p> <p>A1: Studentul va fi capabil să folosească limbajele de programare pentru a dezvolta programe care interacționează cu arhitectura internă a calculatoarelor (ex. manipularea registrilor, gestionarea memoriei).</p> <p>A2: Studentul va putea aplica tehnici de profiling și debugging pentru a analiza performanța aplicațiilor pe diverse arhitecturi hardware.</p> <p>A3: Studentul va fi capabil să dezvolte și să implementeze programe în Assembly pentru a testa și verifica funcționarea componentelor hardware ale unui calculator.</p> <p>RA1: Studentul va demonstra autonomie în utilizarea limbajelor de programare și a tehnicilor de testare pentru a explora și evalua comportamentele arhitecturale ale calculatoarelor.</p> <p>RA2: Studentul va respecta bunele practici de dezvoltare și va respecta standardele de optimizare a performanței și eficiență energetică în programarea arhitecturilor de calcul.</p>
<p>3. Proiectarea și implementarea unităților funcționale ale unui calculator</p>	<p>C1: Studentul va cunoaște arhitectura internă a unui calculator, incluzând procesorul, memoriile, dispozitivele periferice și magistralele.</p> <p>C2: Studentul va înțelege principiile de funcționare a unității de control și modul în care aceasta coordonează execuția instrucțiunilor și accesul la memorie într-un sistem de calcul.</p> <p>C3: Studentul va ști să implementeze circuite de memorie, precum și alte componente ale calculatorului folosind principiul modulației</p> <p>A1: Studentul va fi capabil să proiecteze o unitate funcțională simplă a unui procesor (de exemplu, o unitate aritmetică simplă ALU) utilizând limbaje de descriere hardware precum VHDL sau Verilog.</p> <p>A2: Studentul va putea implementa unități de control și va aplica tehnici de sinteză logică pentru a crea un circuit funcțional pe platforme hardware cum ar fi FPGA.</p>

	<p>A3: Studentul va fi capabil să simuleze și să verifice comportamentul unităților proiectate utilizând instrumente software pentru dezvoltarea de arhitecturi (ex. Vivado).</p> <p>RA1: Studentul va demonstra autonomie în realizarea proiectelor de arhitectură internă a procesorului și va respecta standardele industriei în ceea ce privește performanța și fiabilitatea unităților de procesare.</p> <p>RA2: Studentul va respecta standardele de testare și validare ale unităților funcționale și va aplica cele mai bune practici pentru optimizarea performanței acestora.</p>
Competențe transversale	Rezultatele învățării*
<p>1. Comportarea onorabilă, responsabilă, în spiritul eticii profesionale și respectării legislației curente (inclusiv a drepturilor de proprietate intelectuală), pentru a asigura reputația statutului de student și a profesiei alese</p>	<p>C1: Studentul va cunoaște principiile etice fundamentale din domeniul ingineriei, inclusiv importanța respectării drepturilor de proprietate intelectuală și a legislației în domeniul tehnologiilor informației.</p> <p>C2: Studentul va înțelege reglementările internaționale și locale privind utilizarea software-ului, cercetarea științifică, și proiectarea hardware-ului, inclusiv standardele de sustenabilitate și protecția datelor.</p> <p>A1: Studentul va fi capabil să aplice corect principiile etice în dezvoltarea și implementarea proiectelor, fie că este vorba de software sau hardware, respectând standardele profesionale și drepturile de autor.</p> <p>A2: Studentul va fi capabil să colaboreze eficient în cadrul unei echipe interdisciplinare, demonstrând comportamente etice și responsabilitate față de colegi și față de profesia aleasă.</p> <p>A3: Studentul va fi capabil să recunoască și să raporteze eventualele încălcări ale principiilor etice sau ale legislației de protecție a proprietății intelectuale, atunci când acestea apar în proiectele de cercetare sau dezvoltare.</p> <p>RA1: Studentul va demonstra responsabilitate în utilizarea resurselor software și hardware** și va respecta normele legale privind licențele software și proiectarea hardware.</p> <p>RA2: Studentul va acționa în mod autonom în ceea ce privește deciziile etice, asigurându-se că toate proiectele, cercetările și colaborările respectă atât legislația locală cât și normele internaționale de bună practică profesională.</p>
<p>2. Capacitatea de a sesiza, înțelege și promova calitatea și creativitatea</p>	<p>C1: Studentul va înțelege conceptele fundamentale ale calității și creativității în ingineria sistemelor și informatică, inclusiv standardele internaționale de calitate (ex. ISO 9001) și metodele de evaluare a calității în dezvoltarea de software și hardware.</p> <p>C2: Studentul va cunoaște tehnicile și abordările de creație și inovare utilizate în proiectarea și dezvoltarea sistemelor informatice, inclusiv cele care permit integrarea de soluții inovative în arhitecturi de calculatoare.</p> <p>A1: Studentul va fi capabil să aplice tehnici și instrumente pentru evaluarea și îmbunătățirea calității unui produs software sau hardware (ex. testare software, verificare hardware, optimizare).</p> <p>A2: Studentul va putea dezvolta soluții creative pentru problemele complexe din domeniul arhitecturii calculatoarelor, abordând proiectele cu o perspectivă inovativă și originală.</p>

	<p>A3: Studentul va fi capabil să analizeze și să propună îmbunătățiri creaționale pentru arhitecturi sau algoritmi de calcul, implementând soluții care cresc eficiența și performanța sistemului.</p> <p>RA1: Studentul va demonstra responsabilitate în implementarea soluțiilor de calitate și creativitate, având grijă de impactul acestora asupra performanței și fiabilității sistemelor informatice.</p> <p>RA2: Studentul va promova o cultură a calității și a creativității în echipele de proiect, contribuind activ la îmbunătățirea proceselor de dezvoltare software și hardware și încurajând colaborarea interdisciplinară pentru a găsi soluții inovative.</p>
--	---

* C – cunoștințe; A – aptitudini; RA – responsabilitate și autonomie.

6. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

6.1. Obiectivul general al disciplinei	<p>➤ La sfârșitul cursului studenții vor fi familiarizați, prin cunoștințele dobândite, cu problemele de natură hardware și software specifice domeniului. Scopul principal este deprinderea operării cu fundamente de proiectare / configurare a resurselor hardware specifice științei calculatoarelor.</p>
6.2. Obiectivele specifice	<p>➤ La sfârșitul cursului, studenții vor fi capabili:</p> <ul style="list-style-type: none"> ▪ să înțeleagă natura legăturilor hardware-software la nivelul sistemelor de calcul; ▪ să formuleze și să rezolve probleme din domeniul cursului

7. Conținuturi

7.1. Curs	Nr. ore	Metode de predare	Observații
<p>Scurt istoric al structurii și arhitecturii calculatoarelor:</p> <ul style="list-style-type: none"> - Primele Calculatoare - Clasificarea Calculatoarelor - Dimensiunea structurală a sistemelor de calcul - Arhitecturi generale de sisteme de calcul 	6	<ul style="list-style-type: none"> • Expunere • Explicație 	
<p>Elemente fundamentale de infrastructură a calculatoarelor numerice</p> <ul style="list-style-type: none"> - Evoluția calculatoarelor numerice din punctul de vedere al resurselor fizice - Evoluția calculatoarelor numerice din punctul de vedere al sistemelor de operare 	4	<ul style="list-style-type: none"> • Prelegere, • Dezbateri, 	
<p>Componente principale ale calculatorului:</p> <ul style="list-style-type: none"> - Unitatea centrală de prelucrare - Magistrale de date, de adrese și de comenzi - Memoria - Porturi (interfețe) - Periferice 	4	<ul style="list-style-type: none"> • Problematizare, • Algoritmizare, • Exercițiu, • Tehnici de implementare 	
<p>Subsistemul unitatea centrală de prelucrare (UCP)</p> <ul style="list-style-type: none"> - Funcțiile UCP - Unități de tip RISC - Unități de tip CISC - Unitatea aritmetico-logică - Microprogramarea - Arhitectura familiei de procesoare Intel® 80x86 	6	<ul style="list-style-type: none"> • Prezentare multimedia 	
<p>Magistrale de comunicație</p> <ul style="list-style-type: none"> - Elemente de caracterizare a unei magistrale - Considerații de natură fizică asociate magistrelor 	6		

- Magistrale sincrone - Magistrale asincrone - Familii și standarde de magistrale			
Subsistemul memorie - Clasificarea și caracteristicile tehnice de bază ale dispozitivelor de memorie - Ierarhizarea memoriei - Memoria cu acces aleatoriu (RAM) - Memoriile permanente (ROM) - Memoria internă - Memoria cache - Memoria externă - Memoria virtuală	6	<ul style="list-style-type: none"> • Expunere • Explicație • Prelegere, • Dezbateri, • Problematizare, • Algoritmizare, • Exercițiu, 	
Subsistemul de intrare/ieșire - Operații de intrare/ieșire efectuate sub control programat - Operații de intrare/ieșire efectuate prin întreruperi - Operații de intrare/ieșire efectuate prin acces direct la memorie - Dispozitive periferice de Intrare/ieșire	6	<ul style="list-style-type: none"> • Tehnici de implementare 	
Indicatori de calitate ai unui calculator.	2	Prezentare multimedia	
Calculatorul cuantic – prezentare generală	2		
Bibliografie:			
1.- Rădulescu G. , <i>Arhitectura Calculatoarelor</i> , Editura UPG, Ploiești, 2016 .			
2.- William Stallings , <i>Computer Organization and Architecture</i> , Editura Pearson, 2021 ;			
3.- Paraschiv N. , <i>Structura și Arhitectura Calculatoarelor</i> , Editura UPG Ploiești, 2006 ;			
4.- Linda Null , <i>Essentials of Computer Organization and Architecture</i> , Editura Jones & Bartlett, 2023 ;			
5.- Tanenbaum A. <i>Organizarea structurată a calculatoarelor</i> , 4th Edition. Editura Computer Press Agora, 1999 ;			
6.- Vlad, I.T. <i>Note de curs</i> , suport electronic			
7.2. Seminar / laborator	Nr. ore	Metode de predare	Observații
1. Instalarea pachetului Xilinx și utilitarul Vivado ML Standard Edition	2	<ul style="list-style-type: none"> • Expunere • Explicație • Prelegere, • Dezbateri, • Problematizare, • Algoritmizare, • Teme și exerciții • Tehnici de implementare • Evaluare individuală a referatelor • Testare finală 	
2. Programarea FPGA	2		
3. Introducere in Vivado ML Standard Edition	2		
4. Prezentarea modulelor Programului Vivado ML Standard Edition	2		
5. Prezentarea plăcii de dezvoltare Nexys A7-100T (caracteristici tehnice).	2		
6. Proiectarea și implementarea porții logice AND cu două intrări a câte doi biți	2		
7. Implementarea contoarelor cu placa de dezvoltare Nexys A7 – 100T.	2		
8. Utilizarea afișajelor digitale cu 7 segmente și implementarea convertoarelor binar-hexazecimal, binar-zecimal pe placa de dezvoltare Nexys A7-100T	4		
9. Implementarea unui sumator a două numere de câte 4 biți cu carry-out pe placa de dezvoltare Nexys A7-100T	4		
10. Simularea unui multiplexor cu 8 intrări cu ajutorul plăcii de dezvoltare Nexys A7-100T	2		
11. Implementarea circuitelor de tip Latch și Flip-Flop cu ajutorul plăcii de dezvoltare Nexys A7-100T	2		
12. Verificarea cunoștințelor	2		

<u>Bibliografie:</u>			
1.- Williams J. <i>Digital VLSI Design with Verilog</i> , Springer, 2008			
2.- William Stallings , <i>Computer Organization and Architecture</i> , Editura Pearson, 2021 ;			
3.- Fook L. W. <i>Verilog Coding for Logic Synthesis</i> , Wiley-Interscience, 2003			
4.- AMD Xilinx , <i>Vivado Design Suite User Guide, Release Notes, Installation, and Licensing</i> , 2025			
5.- AMD Xilinx , <i>Documentation Portal</i> , webpage: https://docs.xilinx.com/home			
6.- Vlad, I.T. <i>Note de laborator</i> , suport electronic			
7.3. Proiect	Nr. ore	Metode de predare	Observații
Bibliografie			

8. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

<ul style="list-style-type: none"> ➤ Conținutul disciplinei este în concordanță cu ceea ce se face în alte centre universitare din țară și din străinătate. ➤ Pentru o mai bună adaptare a conținutului disciplinei la cerințele pieței muncii au avut loc întâlniri atât cu reprezentanți ai mediului de afaceri cât și cu profesori din învățământul preuniversitar. <p>Titularul de curs anunță studenții despre întâlnirile organizate de Departamentul Automatică, Calculatoare și Electronică, de Facultatea de Inginerie Mecanică și/sau de Universitatea Petrol – Gaze cu reprezentanții comunitatii epistemice, ai asociațiilor profesionale și cu angajatori reprezentativi din domeniul aferent programului. Titularul de curs participă împreună cu studenții la aceste întâlniri..</p>

9. Evaluare

Tip activitate	9.1. Criterii de evaluare	9.2. Metode de evaluare	9.3. Pondere din nota finală
9.4. Curs	Evaluare finală	Test grilă	50%
	Prezență și interacțiune la orele de curs (criterii ce vizează aspecte precum: atitudinea: seriozitatea, interesul pentru studiul individual)	Lista de prezență și dezbateri despre noțiunile prezentate	10%
9.5. Seminar/laborator	Referate pentru fiecare lucrare de laborator	Se notează îndeplinirea sarcinilor pentru fiecare lucrare de laborator	30%
	Prezență și interacțiune la orele de laborator (criterii ce vizeaza aspecte precum: răspunsul la întrebări, atitudinea și interesul)	Rezolvarea exercițiilor și prezența la data, ora și sala stabilite în orar.	10%

9.6. Proiect			
9.7. Standard minim de performanță			
<p>Înșușirea corectă a noțiunilor teoretice de bază și aplicarea acestora în rezolvarea unor aplicații simple. Pentru nota 5 trebuie să fie îndeplinite cumulativ următoarele condiții:</p> <ul style="list-style-type: none"> • efectuarea tuturor laboratoarelor și trimiterea referatelor în termenul limită; • nota finală de la laborator cel puțin 5; • nota la testul grilă de la examen cel puțin 5. • demonstrarea însușirii semnificației principalilor termeni utilizați în domeniu și a capacității de explicare a acestora în cadrul discuțiilor generale de la laborator; 			

Data
completării
25.09.2025

Semnătura titularului de curs
Șef lucr.dr. ing. Vlad Iulian

Semnătura titularului de laborator
Șef lucr.dr. ing. Vlad Iulian

Semnătura titularului
de proiect

Data avizării în
departament

26.09.2025

Director de departament
Conf. dr. ing. Pricop Emil

Decan
Conf. dr. ing. Bădicioiu Marius
