

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Politehnica Timișoara
1.2 Facultatea ¹ / Departamentul ²	Electronică, Telecomunicații și Tehnologii Informaționale / Electronica Aplicata
1.3 Domeniul de studii (denumire/cod ³)	Inginerie electronică, telecomunicații și tehnologii informaționale/ 20/20/10/100
1.4 Ciclul de studii	Licență
1.5 Programul de studii (denumire/cod/calificarea)	Electronica Aplicata/ 20/20/10/100/10/ Inginer Electronica Aplicata

2. Date despre disciplină

2.1a Denumirea disciplinei/Categoria formativă ⁴			Sisteme Incorporate/DS				
2.1b Denumirea disciplinei în limba engleză			Embedded Systems				
2.2 Titularul activităților de curs			S.I. dr. ing. Radu Mirsu				
2.3 Titularul activităților aplicative ⁵			Drd. Ing. Badoi Ionut				
2.4 Anul de studii ⁶	4	2.5 Semestrul	8	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei ⁷	DOP

3. Timp total estimat - ore pe semestru: activități didactice directe (asistate integral sau asistate parțial) și activități de pregătire individuală (neasistate)⁸

3.1 Număr de ore asistate integral/săptămână	3 , format din:	3.2 ore curs	1.5	3.3 ore seminar/laborator/proiect	1.5
3.1* Număr total de ore asistate integral/sem.	42 , format din:	3.2* ore curs	21	3.3* ore seminar/laborator/proiect	21
3.4 Număr de ore asistate parțial/săptămână	, format din:	3.5 ore practică		3.6 ore elaborare proiect de diplomă	
3.4* Număr total de ore asistate parțial/semestru	, format din:	3.5* ore practică		3.6* ore elaborare proiect de diplomă	
3.7 Număr de ore activități neasistate/săptămână	4 .14 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	1.14	ore studiu individual după manual, suport de curs, bibliografie și notițe	1
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri	2		
3.7* Număr total de ore activități neasistate/semestru	58 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	16	ore studiu individual după manual, suport de curs, bibliografie și notițe	14
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri	28		
3.8 Total ore/săptămână ⁹	7.14				
3.8* Total ore/semestru	100				
3.9 Număr de credite	4				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Circuite integrate digitale, Sisteme de prelucrare numerică cu procesoare, Sisteme de achiziții de date
4.2 de rezultatele învățării	• Programare C, arhitectura sistemelor cu microprocesor

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• Sală curs mare și materiale suport: laptop, videoproiector, tablă
5.2 de desfășurare a activităților practice	• Laborator minim 5 posturi de lucru (calculatoare și echipament electronic specific)

6. Rezultatele învățării la formarea cărora contribuie disciplina

Cunoștințe	• C1. Studentul/absolventul identifică și descrie concepte, principii și metode de bază din matematică, fizică, grafică asistată de calculator, bazele electrotehnicii, limbaje de programare. • C4. Studentul/absolventul descrie, identifică, sumarizează concepte și metode elementare de achiziție, analiză și prelucrare a semnalelor, implementate în sisteme cu procesoare de uz general sau procesoare de semnal și modul lor de aplicare în probleme concrete. • C5. Studentul/absolventul descrie, identifică, sumarizează concepte și metode elementare privitoare la arhitectura sistemelor de calcul, microcontrolere, limbaje și tehnici de programare și modul lor de aplicare în probleme concrete. • C24. Studentul/absolventul explică și analizează principiile fundamentale ale circuitelor electronice și regulile de reprezentare și interpretare a schemelor electronice de putere. • C29. Studentul/absolventul descrie și analizează specificații de proiectare electronică, metode de proiectare asistată de calculator (CAD) și principii de concepere a circuitelor integrate. • C33. Studentul/absolventul descrie conceptele și principiile de funcționare ale senzorilor (mecanici, electrici, optici, biomedicali), precum și domeniile lor de aplicare. • C35. Studentul/absolventul cunoaște limbajele și mediile software utilizate pentru modelarea, simularea și controlul senzorilor. • C38. Studentul/absolventul descrie principiile de funcționare și integrare ale componentelor electronice utilizate în circuite și sisteme hardware. • C41. Studentul/absolventul descrie conceptele fundamentale de programare a microcontrolerelor, arhitecturi hardware și interfețe de comunicație. •
Abilități	A15. Studentul/absolventul proiectează blocuri funcționale de complexitate mică/medie de analiză și prelucrare digitală a semnalelor și le implementează pe procesoare de semnal, microcontrolere sau procesoare dedicate. A17. Studentul/absolventul specifică cerințe, elaborează programe în limbaje de programare de uz general (C, etc.) și /sau obiect-orientate (C++, Java, etc.), execută, depanează și interpretează rezultatele programelor realizate în vederea rezolvării unei probleme concrete. A18. Studentul/absolventul realizează proiecte de complexitate mică/medie care implică utilizarea simultană de componente hardware (microprocesoare sau microcontrolere) și componente software (programele aferente). A29. Studentul/absolventul dezvoltă circuite, sisteme și produse analogice și digitale, electronice și de telecomunicații. A37. Studentul/absolventul este capabil să: analizeze și evalueze proiecte ingineresti conform criteriilor tehnice, economice și de calitate. A63. Studentul/absolventul creează softuri pentru achiziția de date, prelucrare de semnal și integrare a senzorilor în sisteme complexe. A70. Studentul/absolventul programează firmware, implementând algoritmi, drivere și rutine de control în limbaje dedicate (ex. C, C++, ASM). A71. Studentul/absolventul proiectează firmware, elaborând specificații, structuri modulare și scheme de integrare între software și hardware. •
Responsabilitate și autonomie	• RA1. Studentul/absolventul aplică valorile eticii și deontologiei profesiei de inginer. • RA2. Studentul/absolventul practică raționamentul logic, evaluarea și autoevaluare în luarea deciziilor. • RA4. Studentul/absolventul este angajat în învățarea pe tot parcursul vieții pentru dobândirea și implementarea cunoștințelor, după cum este necesar, folosind strategii de învățare adecvate. • RA5. Studentul/absolventul promovează dialogul, cooperarea, respectul față de ceilalți și interculturalitatea. • RA15. Studentul/absolventul demonstrează capacitatea de a lucra autonom și lua decizii fundamentate în evaluarea și validarea proiectelor. • RA17. Studentul/absolventul demonstrează capacitatea de a-și asuma responsabilitatea implementării și validării procedurilor de încercare, garantând acuratețea și relevanța rezultatelor. • RA19. Studentul/absolventul demonstrează capacitatea de a lucra autonom în proiectarea și aplicarea procedurilor de încercare, luând decizii fundamentate tehnic. • RA21. Studentul/absolventul demonstrează capacitatea de a-și asuma responsabilitatea coordonării proiectelor ingineresti, inclusiv alocarea resurselor și respectarea termenelor.

	• RA40. Studentul/absolventul își asumă responsabilitatea pentru calitatea proiectării și testării senzorilor, asigurând conformitatea cu cerințele tehnice și normative. • RA47. Studentul/absolventul își asumă responsabilitatea pentru calitatea și funcționalitatea firmware-ului dezvoltat, respectând cerințele aplicației și reglementările domeniului. • RA48. Studentul/absolventul demonstrează autonomie în proiectarea și programarea firmware-ului pentru sisteme embedded. • RA49. Studentul/absolventul reflectă critic și etic asupra impactului soluțiilor de firmware în raport cu performanța, siguranța și securitatea produselor electronice. •
--	---

7. Obiectivele disciplinei (asociate rezultatelor învățării de la punctul 6)

- Prezentarea arhitecturii și principiilor de utilizare a sistemelor embedded cu procesor Cortex M0
- Pentru fiecare periferic prezentat se vor aborda aplicații concrete care se vor dezvolta complet de la faza de proiectare până la implementarea în limbaj de programare C

8. Conținuturi¹⁰

8.1 Curs	Număr de ore	Metode de predare ¹¹
1. Procesorul Cortex M0 - caracteristici generale, arhitectura, pipeline - registrii, registrii speciali, harta de memorie - setul de instructiuni, modul Thumb, - stiva, implementare si utilizare specifica	3 3	Prelegere susținută de prezentări PPT, conversații, explicații, exemplificări, platforma tip Moodle „Campus Virtual
2. Porturi digitale - arhitectura si implementarea hardware a unui port digital - registrii de control pentru configurarea si utilizarea portului digital - aplicatii si programe exemplu	3	
3 . Sistemul de intreruperi - arhitectura hardware a controlerului de intreruperi - surse de intreruperi, prioritati - executia intreruperilor - exemple de aplicatii	3	
4. Interfete analogice - principii de conversie analog-numerica si numeric-analogica - convertorul cu aproximari succesive, avaluarea calitatii conversiei - prezentarea convertoarelor si a compartorului analogic existent pe placa embedded KL25Z - registrii de control, utilizare, aplicatii exemplu cu program	3 3	
5. Comunicatie seriala - principii generale de comunicatie seriala - UART, SPI, I2C - registrii de control pentru placa KL25Z, utilizare - aplicatii exemplu cu program	3	
Bibliografie ¹² [1] Materiale ARM University, intranet. [2] KL25 Sub-Family Reference Manualhttps://www.nxp.com/docs/en/reference-manual/KL25P80M48SF0RM.pdf. [3] David A. Patterson and John L. Hennessy. 1990. Computer architecture: a quantitative approach. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA..		
8.2 Activități aplicative ¹³	Număr de ore	Metode de predare
1. Prezentare placa de dezvoltare si infrastructurii software de programare	3	Expunere temă, discutii, întrebări,

2. Programare funcții în asamblare și apelarea funcțiilor din C. Prezentare teme proiecte	3	Programare perifericelor de pe placa de dezvoltare folosind C/C++ cu KEIL uVision și mbed Compiler
3. Programare periferice PIT, GPIO, UART	3	
4. Programare periferice SPI, I2C, Accelerometru	3	
5. Prezentare bibliotecă mbed OS și mbed Compiler. Programarea perifericelor folosind mbed OS	3	
6. Prezentare algoritmi de calibrare și filtrare. Analiza reprezentării numerelor asupra performanței de calcul.	3	
7. Îndrumare proiect. Prezentări proiecte	3	
Bibliografie ¹⁴ [1] Materiale ARM University, intranet. [2] KL25 Sub-Family Reference Manual https://www.nxp.com/docs/en/reference-manual/KL25P80M48SF0RM.pdf . [3] David A. Patterson and John L. Hennessy. 1990. Computer architecture: a quantitative approach. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.		

9. Evaluare

Tip activitate	9.1 Criterii de evaluare ¹⁵	9.2 Metode de evaluare	9.3 Pondere din nota finală
9.4 Curs	Tratare aspecte teoretice Rezolvare aplicații	Examinare scrisă	1/2
9.5 Activități aplicative	S:		
	L: Implementarea programelor propuse în limbaj de programare C	Examinare orală, scrisă (pe calculator)	1/2
	P¹⁶:		
	Pr:		
9.6 Standard minim de performanță (se prezintă cunoștințele minim necesare pentru promovarea disciplinei și modul în care se verifică stăpânirea lor ¹⁷)			
• Cel puțin nota 5 (jumătate din subiecte rezolvate corect) acordată pentru tratarea întrebărilor cu răspuns preponderent teoretic • Cel puțin nota 5 (jumătate din subiecte rezolvate corect) acordată pentru tratarea problemelor			

Data completării

01.09.2025

Titular de curs
(semnătura)

Titular activități aplicative
(semnătura)

Director de departament
(semnătura)

Data avizării în Consiliul Facultății¹⁸

07.10.2025

Decan
(semnătura)