

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/Electronică Aplicată
1.3 Catedra	—
1.4 Domeniul de studii (denumire/cod ³)	Inginerie electronică, telecomunicații și tehnologii informaționale/20/20/10/100
1.5 Ciclul de studii	Licență
1.6 Programul de studii (denumire/cod/calificarea)	Electronică Aplicată/20/20/10/100/10/Electronică Aplicată

2. Date despre disciplină

2.1 Denumirea disciplinei/Categoria formativă ⁴	Sisteme Embedded/DS						
2.2 Titularul activităților de curs	Șef lucr. dr. ing. Radu MÎRȘU						
2.3 Titularul activităților aplicative ⁵	Șef lucr. dr. ing. Ciprian SEICULESCU						
2.4 Anul de studii ⁶	4	2.5 Semestrul	2	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei ⁷	DO

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	3	3.3 ore seminar/laborator/proiect	0/3/0
3.1* Număr total de ore asistate integral/sem.	42 , format din:	3.2* ore curs	21	3.3* ore seminar/laborator/proiect	0/21/0
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	5.1 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			1,5
		ore studiu individual după manual, suport de curs, bibliografie și notițe			1,5
		ore pregătire seminar/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			2,1
3.7* Număr total de ore activități neasistate/semestru	36 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			10.5
		ore studiu individual după manual, suport de curs, bibliografie și notițe			10.5
		ore pregătire seminar/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			15
3.8 Total ore/săptămână ⁹	8.1				
3.8* Total ore/semestru	78				
3.9 Număr de credite	3				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	<ul style="list-style-type: none"> Circuite integrate digitale, Sisteme de prelucrare numerica cu procesoare, Sisteme de achizitii de date
-------------------	---

¹ Se înscrie numele facultății care gestionează programul de studiu căruia îi aparține disciplina.

² Se înscrie numele departamentului căruia i-a fost încredințată susținerea disciplinei și de care aparține titularul cursului.

³ Se înscrie codul prevăzut în HG – privind aprobarea Nomenclatorului domeniilor și al specializărilor/programelor de studii, actualizată anual.

⁴ Disciplina se încadrează potrivit planului de învățământ în una dintre următoarele categorii formative: disciplină fundamentală (DF), disciplină de domeniu (DD), disciplină de specialitate (DS) sau disciplina complementară (DC).

⁵ Prin activități aplicative se înțeleg activitățile de: seminar (S) / laborator (L) / proiect (P) / practică (Pr).

⁶ Anul de studii în care este prevăzută disciplina în planul de învățământ.

⁷ Disciplina poate avea unul din următoarele regimuri: disciplină impusă (DI) sau disciplină obligatorie (DOb)-pentru alte domenii fundamentale de studii oferite de UPT, disciplină opțională (DO) sau disciplină facultativă (Df).

⁸ Numărul de ore de la rubricile 3.1*, 3.2*,...,3.8* se obțin prin înmulțirea cu 14 (săptămâni) a numărului de ore din rubricile 3.1, 3.2,..., 3.8. Informațiile din rubricile 3.1, 3.4 și 3.7 sunt chei de verificare folosite de ARACIS sub forma: (3.1)+(3.4) ≥ 28 ore/săpt. și (3.8) ≤ 40 ore/săpt.

⁹ Numărul total de ore / săptămână se obține prin însumarea numărului de ore de la punctele 3.1, 3.4 și 3.7.

4.2 de competențe	• Programare C
-------------------	----------------

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. Competențe la formarea cărora contribuie disciplina

Competențe specifice	<ul style="list-style-type: none"> • Programarea sistemelor embedded
Competențele profesionale în care se înscriu competențele specifice	<ul style="list-style-type: none"> • Rezolvarea problemelor tehnologice din domeniile sistemelor embedded. •
Competențele transversale în care se înscriu competențele specifice	<ul style="list-style-type: none"> • Analiza metodică a problemelor întâlnite în activitate, identificând elementele pentru care există soluții consacrate, asigurând astfel îndeplinirea sarcinilor profesionale • Definirea activităților pe etape și repartizarea acestora subordonațiilor cu explicarea completă a îndatoririlor, în funcție de nivelurile ierarhice, asigurând schimbul eficient de informații și comunicarea interumană • Adaptarea la noile tehnologii, dezvoltarea profesională și personală, prin formare continuă folosind surse de documentare tipărite, software specializat și resurse electronice în limba română și, cel puțin, într-o limbă de circulație internațională

7. Obiectivele disciplinei (asociate competențelor de la punctul 6)

7.1 Obiectivul general al disciplinei	• Prezentarea arhitecturii și principiilor de utilizare a sistemelor embedded cu procesor Cortex M0
7.2 Obiectivele specifice	<ul style="list-style-type: none"> • 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 instrucțiuni, modul Thumb, - stiva, implementare și utilizare specifică	6	Prelegere susținută de prezentări PPT, conversații, explicații, exemplificări, platforma tip Moodle „Campus Virtual”
2. Porturi digitale - arhitectura și implementarea hardware a unui port digital - registrii de control pentru configurarea și utilizarea portului digital - aplicații și programe exemplu	3	
3. Sistemul de întreruperi - arhitectura hardware a controlerului de întreruperi	3	

¹⁰ Se detaliază toate activitățile didactice prevăzute prin planul de învățământ (tematicile prelegerilor și ale seminariilor, lista lucrărilor de laborator, conținuturile etapelor de elaborare a proiectelor, tematica fiecărui stagiu de practică). Titlurile lucrărilor de laborator care se efectuează pe standuri vor fi însoțite de notația „(*)”.

¹¹ Prezentarea metodelor de predare va include și folosirea noilor tehnologii (e-mail, pagină personalizată de web, resurse în format electronic etc.).

- surse de intreruperi, prioritati - executia intreruperilor - exemple de aplicatii		
4. Interfete analogice - principii de conversie analog-numeric 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	6	
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 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.		
8.2 Activități aplicative¹³	Număr de ore	Metode de predare
1. Prezentare placa de dezvoltare si infrastructurii software de programare	3	Expunere temă, discuții, întrebări, Programare perifericelor de pe placa de dezvoltare folosind C/C++ cu KEIL uVision si mbed Compiler.
2. Programare funcții in asamblare si apelarea funcțiilor din C. Prezentare teme proiecte.	3	
3. Programare periferice PIT, GPIO, UART.	3	
4. Programare periferice SPI, I2C, Accelerometru.	3	
5. Prezentare biblioteca mbed OS si mbed Compiler. Programarea perifericelor folosind mbed OS	3	
6. Prezentare algoritmi de calibrare si filtrare. Analiza reprezentări numerelor asupra performantei 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. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

¹² Cel puțin un un titlu trebuie să aparțină colectivului disciplinei iar cel puțin un titlu trebuie să se refere la o lucrare de referință pentru disciplină, de circulație națională și internațională, existentă în biblioteca UPT.

¹³ Tipurile de activități aplicative sunt cele precizate în nota de subsol 5. Dacă disciplina conține mai multe tipuri de activități aplicative atunci ele se trec consecutiv în liniile tabelului de mai jos. Tipul activității se va înscrie într-o linie distinctă sub forma: „Seminar:”, „Laborator:”, „Proiect:” și/sau „Practică:”.

¹⁴ Cel puțin un titlu trebuie să aparțină colectivului disciplinei.

- Majoritatea angajatorilor reprezentativi (Continental, Hella, Elster, Yazaki, Alcatel, Flextronics, Intel, etc.) din domeniul aferent programului de studiu solicita cunoștințe legate de funcționarea sistemelor embedded.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare ¹⁵	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Tratare aspecte teoretice	Examinare scrisa	1/3
10.5 Activități aplicative	S:		
	L: Implementarea programelor propuse in limbaj de programare C	Examinare orala/scrisa (pe calculator)	1/3
	P¹⁶:		
	Pr: Implementarea programelor propuse in limbaj de programare C	Examinare orala/scrisa (pe calculator)	1/3
10.6 Standard minim de performanță (se prezintă cunoștințele minim necesare pentru promovarea disciplinei și modul în care se verifică stăpânirea lor ¹⁷)			
<ul style="list-style-type: none"> Cel puțin nota 5 (jumătate din subiecte rezolvate corect) acordata pentru tratarea întrebărilor cu răspuns preponderent teoretic Cel puțin nota 5 (jumătate din subiecte rezolvate corect) acordata pentru tratarea problemelor Cel puțin nota 5 pentru activitatea practica (proiectarea, testarea unui circuit electronic de complexitate medie, prezenta 			

Data completării

22.05.2020

**Titular de curs
(semnătura)**



**Titular activități aplicative
(semnătura)**



**Director de departament
(semnătura)**

Data avizării în Consiliul Facultății¹⁸

26.05.2020

**Decan
(semnătura)**

¹⁵ Fișele disciplinelor trebuie să conțină procedura de evaluare a disciplinei cu precizarea criteriilor, a metodelor și a formelor de evaluare, precum și cu precizarea ponderilor atribuite acestora în nota finală. Criteriile de evaluare se formulează în mod distinct pentru fiecare activitate prevăzută în planul de învățământ (curs, seminar, laborator, proiect). Ele se vor referi și la formele de verificare pe parcurs (teme de casă, referate ș.a.)

¹⁶ În cazul când proiectul nu este o disciplină distinctă, în această rubrică se va preciza și modul în care rezultatul evaluării proiectului condiționează admiterea studentului la evaluarea finală din cadrul disciplinei.

¹⁷ Nu se va explica cum se acorda nota de promovare.

¹⁸ Avizarea este precedată de discutarea punctului de vedere al board-ului de care aparține programul de studii cu privire la fișa disciplinei.