FIŞA DISCIPLINEI

1. **Date despre program**

Facultatea	Inginerie Electrică şi Ştiinţa Calculatoarelor
Departamentul	Departamentul de Calculatoare
Domeniul de studii	Calculatoare și tehnologia informației
Ciclul de studii	Masterat
Programul de studii/calificarea	Știința și ingineria calculatoarelor

2. **Date despre disciplină**

Denumirea disciplinei	**PROIECTAREA SISTEMELOR ÎNGLOBATE**						
Anul de studiu	II	Semestrul	3	Tipul de evaluare	E		
Regimul disciplinei	Categoria formativă a disciplinei DF - fundamentală, DS - de specializare, DC – complementară					DS	
	Categoria de opționalitate a disciplinei: DOB – obligatorie, DOP – opțională, DFA - facultativă					DOB	

3. **Timpul total estimat** (ore alocate activităţilor didactice)

I a) Număr de ore, pe săptămână	3,5	Curs	2	Seminar		Laborator		Proiect	1,5
I.b) Totalul de ore (pe semestru) din planul de învăţământ	49	Curs	28	Seminar		Laborator		Proiect	21

Distribuția fondului de timp pe semestru	ore
II. Studiu individual	97
III. Examinări	4
IV. Alte activități (precizați):	

Total ore studiu individual (II+III)	101
Total ore pe semestru (I.b+II+III+IV)	150
Numărul de credite	6

4. **Competenţe specifice acumulate**

Competenţe transversale	CT1 -lucreaza în echipe
Competențe profesionale	CP1 - efectueaza cercetare stiintifica CP5 -asigura managementul de proiect CP9 -proiecteaza hardware CP10 -testeaza hardware CP12 -modeleaza si simuleaza hardware

5. **Rezultatele învățării**

Cunoștințe	Aptitudini	Responsabilitate și autonomie
Studentul/absolventul identifică, descrie critic, sumarizează și utilizează ca bază a unei cercetări originale concepte și metode avansate privitoare la structura si arhitectura sistemelor înglobate.	Studentul / absolventul cercetează, proiectează, implementează și testează sisteme înglobate, incluzând componente hardware și software cu respectarea constrângerilor de timp real, integrează cunoștințele dobândite pentru a propune soluții inovative în cadrul proiectelor de cercetare aplicată și de dezvoltare tehnologică.	Studentul / absolventul poate desfășura în condiții de autonomie activități care implică inovație tehnologică, cercetare științifică și comunicare tehnică în ştiinţa şi ingineria calculatoarelor.

6. **Obiectivele disciplinei** (reieşind din grila competenţelor specifice acumulate)

Obiectivul general al disciplinei	● Obiectivul general al disciplinei este sinteza, organizarea şi dobândirea de noi cunoştinţe şi deprinderi pentru proiectarea unor sisteme înglobate utilizând microcontrolere.
	●

7. **Conținutul predării și învățării**

Curs	Nr. ore	Metode de predare	Observaţii
Introducere în Ingineria software a sistemelor înglobate (Inginerie software, Sisteme încorporate, Sisteme în timp real, Exemplu de sistem hard în timp real, Caracteristicile evenimentelor în timp real, Provocări în proiectarea sistemului în timp real, Procesul de construire a software-ului sistemului încorporat, Arhitecturi distribuite și multiprocesor, Software pentru sisteme încorporate,Sstraturi de abstracție hardware pentru sisteme încorporate)	2h	expunerea, prelegerea-dezbatere,	
Sisteme de operare **(**Sisteme în prim-plan/în fundal, Kernel-uri în timp real, RTOS (sistem de operare în timp real), Atribuirea priorităților la sarcini(task-uri, fire de execuție), Determinarea dimensiunii unei stive, Programare cu suspendare, , Puncte de planificare, Planificare Round-Robin, Schimbarea contextului, Managementul întreruperilor, Tick-ul ceasului (sau Tick-ul sistemului), Managementul resurselor, Sincronizare 196)	4h	expunerea, prelegerea-dezbatere, demonstraţia.	
Construcția blocurilor de timp real: evenimente și declanșatoare (triggers) **(**Evenimente și declanșatoare, unitate de temperatură în cameră, sisteme de evenimente, manipularea evenimentelor, metode pentru evenimente, structura datelor de tip eveniment, Reentrancy, procesarea evenimentelor,...**)**	2h	expunerea, prelegerea-dezbatere, demonstraţia.	
Procesul de dezvoltare a software-ului (Noțiuni introductive, Planificarea proiectului, Cerințe, Arhitectura, Proiectarea, Implementarea, Derularea împreună: Dezvoltare agilă)	2h	expunerea, prelegerea-dezbatere, demonstraţia.	
Arhitectura sistemelor incorporate și multicore - proiectare și optimizare **(**Introducere, Calea corectă și calea greșită, Înțelegerea cerințelor, Maparea aplicației, Ajutarea compilatorului și a instrumentelor de construcție , Optimizarea consumului de putere)	1h	expunerea, prelegerea-dezbatere, demonstraţia.	
Tehnici de bază de programare (Introducere, Prezentare generală a platformei de referință, Instalare SDK, Configurarea și inițializarea sistemului țintă, Exemple de programare)	1h	expunerea, prelegerea-dezbatere, demonstraţia.	
Programare și ghiduri de implementare (Introducere, Pornirea proiectului de software încorporat, Structura variabilelor)	2h	expunerea, prelegerea-dezbatere, demonstraţia.	
Optimizarea software-ului și a compilatoarelor pentru microcontrollere, procesoare incorpate și dsp-uri (Introducere, Prezentare generală a instrumentelor de dezvoltare, Înțelegerea țintei încorporate, Arhitectura, Obiective și practici de de bază pentru optimizare, Transformări generale în buclă)	2h	expunerea, prelegerea-dezbatere, demonstraţia.	
Software înglobat (embedded) calitate, integrare și tehnici de testare (Ce este testarea software-ului? De ce ar trebui să testăm software-ul? Câte teste sunt suficiente? Când ar trebui să	2h	expunerea, prelegerea-dezbatere, demonstraţia.	

aibă loc testarea? Cine ia deciziile? Stabilirea standardului Când ai de-a face cu ceva neobișnuit)			
Dezvoltarea de software înglobat pe multicore (Multiprocesare simetrică și asimetrică, Paralelismul economisește putere, Căutați oportunități de paralelism, Localizarea unei aplicații multicore, modele de programare multicore, Performanța și optimizarea sistemelor multicore, Exemplu de extensii de limbaj—OpenMP, Tragând totul împreună)	2h		
Dezvoltarea de software critic pentru siguranță (care sunt cerințele pentru siguranță, strategii de planificare a unui proiect, erori, căderi, hazard, și analiza riscului, arhitecturi critice pentru siguranță)	2h	expunerea, prelegerea-dezbatere,	
Proprietate intelectuală (Introducere, Este acest software al tău? Patente, Probleme)	2h	expunerea, prelegerea-dezbatere,	
Managementul dezvoltării software-ului înglobat (Integrarea modelului de maturitate a capacităţilor, modelul OSI, Dezvoltare software, Organizare, Carta programului, Părțile interesate (stakeholders) și echipa de bază, Managementul ciclului de viață al produsului, Managementul portofoliului, Ciclul de viață al managementului proiectelor, Ciclul de viață al proiectului, Rezolvarea problemelor, Comunicaţii)	2h	prelegerea-dezbatere, demonstraţia.	
Software înglobat pentru aplicații automotive (Scurt istoric, Segmentul automotive și cum diferă acesta, Calitatea în automotive, testare și dezvoltare, diagnostic în automotive)	2h	prelegerea-dezbatere, demonstraţia.	

Bibliografie minimală recomandată

[1] **Robert Oshana, Mark Kraeling - Software Engineering for Embedded Systems Methods, Practical Techniques, and Applications, Second Edition,** ISBN: 978-0-12-809448-8**, Elsevier 2019**

[2] **Robert Oshana, Mark Kraeling - Software Engineering for Embedded Systems Methods, Practical Techniques, and Applications, ISBN:978-0-12-415917-4, Elsevier 2013**

[3] Embedded Systems Architecture A Comprehensive Guide for Engineers and Programmers By Tammy Noergaard, Elsevier 2013, ISBN: 978-0-12-382196-6.

[4] Vasile Gheorghiță GĂITAN, Nicoleta-Cristina GĂITAN, Ioan UNGUREAN, Valentin POPA - Utilizarea specificațiilor OPC DA pentru implementarea aplicațiilor distribuite de tip SCADA, Editura DrukArt, Cernăuți, Ucraina, 2013, ISBN 978-966-2021-69-1.

[5] Yiu, Joseph - The definitive guide to the ARM Cortex-M3, second edition. Newnes 2010, ISBN 978-1-85617-963-8.

[6] INTEGRATION TECHNOLOGIES FOR INDUSTRIAL AUTOMATED SYSTEMS - Edited by Richard Zurawski, Taylor & Francis Group, 2007, ISBN 978-0-8493-9262-7.

[7] NETWORKED EMBEDDED SYSTEM - Edited by Richard Zurawski, Taylor & Francis Group, 2009, ISBN 978-1-4398-0761-3.

[8] Kai Qian, David den Haring, Li Cao - Embedded Software Development with C, Springer 2009, ISBN 978-1-4419-0605-2.

[9] Wolfgang Ecker, Wolfgang Müller, Rainer Dömer - Hardware-dependent Software - Principles and Practice, Springer 2009, ISBN 978-1-4020-9435-4.

[10] Daniel W. Lewis – Fundamental of Embedded Software with the ARM Cortex-M3, 2Ed., Pearson 2013, ISBN 978-0-13-335722-6.

[11] Suport de curs in format electronic, Google Classroom (actualizat 2026).

Aplicaţii (Seminar/laborator/proiect)	Nr. ore	Metode de predare	Observaţii
Tema principală a proiectului: Să se implementeze un sistem distribuit de achiziții de date bazat pe microcontrolere ARM care să cuprindă un gateway MODBUS, care implementează protocoalele MODBUS (TCP/IP, RTU și ASCII), și cinci stații MODBUS Slave care implementează protocoalele MODBUS RTU și ASCII.	21h	Discuții, dezbateri, demonstrații	Pe durata celor 21 ore de proiect se vor discuta, prezenta, demonstra soluții pentru toate cele 17 subteme. Subtemele

Conexiunea GATEWAY-ului cu PC-ul se realizează prin: 1) Ethernet (DLL in C++); 2) USB (DLL in C++) Tema principală cuprinde următoarele subteme: **(pe un calculator gazdă)** 1) un utilitar în Python pentru testare și configurare ce poate lucra utilizând fie MODBUS TCP/IP fie USB (care transportă mesajele în format MODBUS TCP/IP); 2) un driver MODBUS TCP/IP (scris în C++) pentru a permite conexiunea la serverul OPC DA; 3) un driver USB MODBUS TCP/IP (scris în C++, care transportă mesajele în format MODBUS TCP/IP) pentru a permite conexiunea la serverul OPC DA; **(la nivelul gateway-ului)** 4) un task pentru protocolul MODBUS TCP-IP (Ethernet - se vor utiliza socket-uri BSD); 5) un task USB-MODBUS TCP-IP mesaje 6) pe PC se va afla aplicația client, iar pe GATEWAY aplicația server și, ca urmare, va fi prevăzut un task care să asigure funcționarea GATEWAY-ul ca un SLAVE (server) care poate executa funcțiile clasice MODBUS (un task de funcții MODBUS clasa 0,1 și 2); 7) un task pentru achiziții de date (analogice și digitale) și comenzi (analogice și digitale); 8) un task MODBUS TCP-IP, MODBUS RTU/ASCII – gateway-ul propriu-zis; 9) un task care să definească un ciclu de achiziții; 10) un task de configurare; 11) un task data logger; **(la nivelul celor cinci stații SLAVE RTU/ASCII).** 12) un task pentru implementarea protocolului MODBUS ASCII/RTU; 13) un task de achiziție de date și control; 14) un task pentru data logger și configurare; (**proiectare hardware)** 15) Gateway-ul, 16) Stația slave **(la nivelul întregului sistem)** 17) Intregrare de sistem			pot fi luate în grupuri de 2-3 studenți.

Bibliografie minimală recomandată

[1] Vasile Gheorghiță GĂITAN, Nicoleta-Cristina GĂITAN, Ioan UNGUREAN, Valentin POPA - Utilizarea specificațiilor OPC DA pentru implementarea aplicațiilor distribuite de tip SCADA, Editura DrukArt, Cernăuți, Ucraina, 2013, ISBN 978-966-2021-69-1.

[2] Yiu, Joseph - The definitive guide to the ARM Cortex-M3, second edition. Newnes 2010, ISBN 978-1-85617-963-8.

[3] INTEGRATION TECHNOLOGIES FOR INDUSTRIAL AUTOMATED SYSTEMS - Edited by Richard Zurawski, Taylor & Francis Group, 2007, ISBN 978-0-8493-9262-7.

[4] NETWORKED EMBEDDED SYSTEM - Edited by Richard Zurawski, Taylor & Francis Group, 2009, ISBN 978-1-4398-0761-3.

[5] Kai Qian, David den Haring, Li Cao - Embedded Software Development with C, Springer 2009, ISBN 978-1-4419-0605-2.

[6] Daniel W. Lewis – Fundamental of Embedded Software with the ARM Cortex-M3, 2Ed., Pearson 2013, ISBN 978-0-13-335722-6.

[7] www.arm.com

[8] www.st.com

[9] www.modbus.org

[10] Îndrumar de laborator: https://apollo.eed.usv.ro/~cristinag/ , http://www.eed.usv.ro/~cristinag (2026)

8. **Evaluare**

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Evaluare finală	● Nota acordată pentru participarea activă în timpul cursurilor. ● Nota acordată la examinarea finală	Evaluare continuă Evaluare prin probă finală de tip test grilă cu întrebări din conținutul cursului sub controlul mediului de învățare - examinare Moodle, sau opțional prezentarea individuală a unei teme de casă bazată pe subiecte din curs dar prezentate într-o manieră diferită de aceea din curs.	10 % 40 %
Evaluare pe parcurs	Respectarea etapelor de proiectare Corectitudinea soluțiilor alese. Originalitatea soluțiilor alese.	Evaluare continuă, Evaluare continuă, Evaluare sumativă.	10 % 20 % 20 %

Fișa disciplinei include, dacă este cazul, elemente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

Data completării	Grad didactic, nume, prenume, semnătura titularului de curs	Grad didactic, nume, prenume, semnătura titularului de aplicație
23.09.2026	prof.dr.ing. Vasile Gheorghita GĂITAN	prof.dr.ing. Vasile Gheorghita GĂITAN conf.dr.ing. Nicoleta Cristina GĂITAN

Data avizării	Grad didactic, nume, prenume, semnătura responsabilului de program
24.09.2026	prof. dr. ing. Stefan-Gheorghe PENTIUC

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
24.09.2026	Prof. dr. ing. Ovidiu-Andrei SCHIPOR

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
25.09.2026	Prof. dr. ing. Laurentiu- Dan MILICI