

**Nr. 2**

Email/Gr/Nume+prenume [Cu litere MARI]

**Examen: Arhitectura  
sistemelor de calcul  
Gr 134, 18-Ian-2010**

.....  
.....

**T1:** Dați răspunsuri scurte (1-2 fraze) la următoarele întrebări:

1. Cum se implementează “testul la zero” într-un ALU pe 32b?
2. Prin ce diferă cele 3 metode de împărțire în hardware prezentate la curs?
3. Enumerați tipurile de instrucțiuni MIPS și dați un exemplu din fiecare.
4. Câte porți de citire și de scriere folosește fișierul de regiștrii RF? (semnalele de control nu se numără)
5. Unde este plasata data cu adresa 2007B într-un cache cu 64 blocuri cu 16B fiecare?
6. Ce este un multiplexor? Indicați o formulă logică de specificare.
7. Din ce componente se compune prețul de vânzare [list price] al PC-urilor?
8. Prin ce diferă sumatorul secvențial (ripple-carry adder) de cel rapid (carry-lookahead adder)?
9. La procesorul cu pipeline, cum se setează semnalele de control pentru fazele IF, ID în caz de întârziere?
10. Care este principiul folosit la arbitrarea distribuită cu auto-selectare pentru gestionarea accesului la o magistrală? **fiecare, 1.5 puncte**

**T2:** Prezentați pe scurt (1-2 pagini) unitatea de control folosită la procesorul cu faze multiple.

**7.5 puncte**

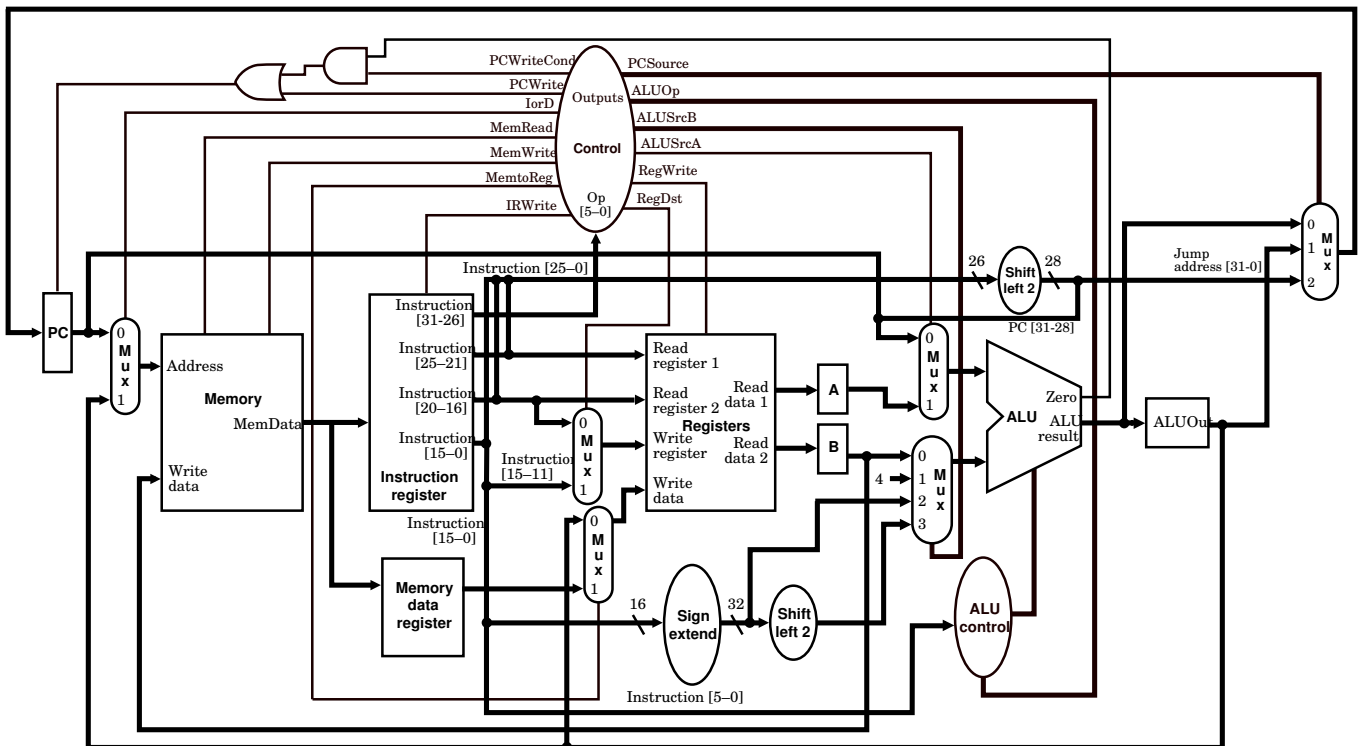
**T3:** Aplicați a 2-a metodă de înmulțire în hardware dată în curs pentru a înmulți 5 cu 3 (valori din baza 10) și descrieți pas cu pas conținutul regiștrilor folosiți.

**7.5 puncte**

**P1:** Scrieți un program MIPS care să verifice dacă printre cifrele unui număr natural  $n$  apare și cifra 7. Faceți și o descriere informală a metodei folosite.

**10 puncte**

**P2:** Considerăm schema completă de procesor cu cicluri multiple cu punctele de tăietură A-J indicate în figură:



Găsiți codul binar care reprezintă instrucțiunea “and \$7,\$3,\$7”.

Completați valorile regiștrilor și ale semnalelor de control în fiecare ciclu în punctele A-J când se procesează instrucțiunea de mai sus și \$3 conține 65, iar \$7 conține 1025. (Folosiți NA pentru valori nedefinite.)

Descrieți sumar ce operație se face în fiecare pas.

**10 puncte**

**P3:** Considerăm programul

1	add	\$t1,\$t2,\$t2;	6	bne	\$t1,\$t4,\$t2;
2	bne	\$t2,\$zero,\$t1;	7	add	\$t1,\$t2,\$t3;
3	lw	\$t2,20(\$t4);	8	sub	\$t2,\$t3,\$t4;
4	sw	\$t1,20(\$t4);	9	Et2	addi \$t4,\$t4,4;
5	Et1	addi \$t2,\$t2,4;	10	addi	\$t4,\$t4,4;

1. Determinați timpul de execuție (numărul de cicluri de ceas) când se rulează programul de mai sus, cu probabilitate 25% de a accepta testele 2 și 6 și folosind un procesor pipeline cu tehnicile de “avansare” și “întârziere” incluse (ilustrați execuția instrucțiunilor).
2. Se poate rearanja codul (păstrând funcția calculată de program) astfel încât să se reducă timpul de execuție? Motivați răspunsul.

**10 puncte**