

## Problema 1 (2punts):

Es disposa de 4 xips de ROM:

- 2 xips 2K x 4 bits
- 1 xip 4K x 8 bits
- 1 xip 1K x 16 bits

Es vol implementar una única ROM de 8 bits i maximitzar la seva capacitat d'emmagatzematge, utilitzant només els 4 xips de què es disposa.

Es demana:

- a) Especifiqueu les entrades, sortides i grandària en kilobytes de la ROM de 8 bits a implementar.
- b) Dibuixeu el circuit complet incloent totes les entrades i sortides de la ROM. Es disposa de busos, comportes, multiplexors, buffers, decodificadors, etc. Justifiqueu la solució proposada.

Problema 2 (3 punts): Un ordinador té una unitat de memòria de 4 k paraules i una memòria cau de 256 paraules. La memòria cau és associativa per conjunts, amb 16 conjunts i 2 particions per conjunt. Quan es necessita una substitució a la memòria cau, en el conjunt corresponent es reemplaça la partició més antiga.

- a) Format de l'adreça

	Etiqueta	Conjunt	Paraula
Nº de bits			

- b) Suposant que inicialment la memòria cau és buida. Ompliu la taula senyalant les fallades i substitucions que es produiran a la memòria cau si es llegeixen consecutivament les adreces de la MP.

Adreça	Etiqueta		Conjunt	Fallada (si/no)	Bloc substituït
	L1	L2			
0000 0000 1111					
0000 0100 1111					
0001 0000 1111					
0001 0000 1100					
0010 1000 1111					
0000 0000 1101					

Problema 3 (5 punts): Sigui un computador capaç d'executar 10 MIPS ( $10^7$  instruccions per segon). Es vol connectar a l'ordinador, un sol perifèric amb una velocitat de transferència de 20.000 bytes / s. i sobre el qual es realitzen operacions de lectura de blocs de 1.024 bytes. Es pretén veure el comportament de la parella ordinador - perifèric davant les diferents tècniques d'entrada/sortida (programada, mitjançant interrupcions i per DMA)

Se sap que:

- La rutina de transferència d'E / S programada consta de 10 instruccions.
- La rutina de tractament d'interrupció en l'E/S mitjançant interrupcions consta de 20 instruccions.
- La rutina de inicialització del DMA consta de 8 instruccions. I en cada operació d'escriptura d'una dada a memòria el controlador ocupa els busos durant 500 ns.

Es demana que digueu el nombre d'instruccions d'altres processos que pot realitzar l'ordinador durant cadascun dels tipus d'E/S previstos.

Nom: .....

Problema 1: Si tenim les següents declaracions de variables globals, emmagatzemades a memòria a partir de l'adreça 0x10010000:

```
.data
.word    0x100100004
.byte    0xAA
.half    0x00BB
.word    0x88AACCEE
.half    0x0403
.byte    0x40, 0x50, 0x60, 0x70, 0x80, 0x90
```

- a) Indiqueu el contingut inicial de la memòria, en hexadecimal, segons el format que utilitza el simulador QtSpim si el processador MIPS està treballant en *little endian*. Podeu suposar que les posicions de memòria abans de la càrrega del programa valen zero.

Adreça	Valor (0)	Valor (4)	Valor (8)	Valor (c)
0x10010000				
0x10010010				

- b) Quin serà el valor dels registres modificats, en hexadecimal, després d'executar cadascuna de les línies de codi?

```
la $t0, a1      # Carrega al registre l'adreça de memòria      $t0: .....
lw $t1, 0($t0)  # Carrega al registre la paraula (32 bits)      $t1: .....
lb $t2, 2($t1)  # Carrega al registre el byte (fa l'extensió del signe) $t2: .....
sra $t2, $t2, 2 # Desplaçament lògic a la dreta (2)            $t2: .....
sh $t2, 4($t1)  # Emmagatzema una paraula de 16 bits a memòria  Mem(.....): .....
lw $t1, 4($t1)  # Carrega al registre la paraula (32 bits)      $t1: .....
```

Problema 2: El fragment de codi s'executa en un DLX amb segmentació. Supposeu que una dada es pot escriure en el banc de registres i llegir el seu nou valor en el mateix cicle. Recordeu que els cicles són: IF, ID, EX, MEM i WB

- a) Calculeu el nombre de cicles necessaris per executar aquest codi si no hi ha cap curtcircuit (anticipació d'operands).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
sub r5, r2, r3																								
add r4, r5, r6																								
sub r5, r4, r8																								
add r7, r2, r3																								
add r6, r7, r3																								
lw r1, 0(r6)																								
add r3, r1, r4																								
sub r6, r7, r8																								

- b) Calculeu el nombre de cicles si hi ha curtcircuits (anticipació d'operands).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
sub r5, r2, r3																								
add r4, r5, r6																								
sub r5, r4, r8																								
add r7, r2, r3																								
add r6, r7, r3																								
lw r1, 0(r6)																								
add r3, r1, r4																								
sub r6, r7, r8																								

- c) Tracteu de reordenar el codi per aconseguir que el nombre de cicles sigui mínim, si hi ha curtcircuit (anticipació d'operands). Quants cicles són necessaris en aquest cas?.

Problema 3: Representeu en l'estàndard IEEE 754 (puny flotant) de simple precisió els números 14.0 i -3.5

14.0:

-3.5:

Problema 4: Suposeu que a la memòria de la màquina senzilla hi ha els continguts següents:

@	Contingut	@	Contingut
0	add 100,101	100	1
1	mov 100, 102	101	65536
2	Cmp 100, 102	102	0

Suposant que el **PC** conté inicialment un **0**,

- a) Quin és el contingut final de les posicions de memòria involucrades en les instruccions?

@	Contingut
100	
101	
102	

- b) Com i quan varia el valor d'FZ?.

Descriviu el cicle d'execució de les següents instruccions suposant les dades inicials de l'apartat anterior i que el programa comença a l'adreça 0.

@	Contingut
0	add 100, 101
1	cmp 101, 102
2	beq 0

Quantes vegades s'executa la instrucció de l'adreça 0?