

Soluzione

Poiché un blocco è costituito da 2B, e la cache è di 4B si avranno $4/2 = 2$ linee. Quindi gli 8 bit di indirizzo saranno suddivisi nel seguente modo: il bit meno significativo individuerà il byte all'interno del blocco, il secondo bit da destra individuerà la linea (0 o 1), ed i restanti bit costituiranno il tag. Mostriamo di seguito l'evoluzione del contenuto dei registri, dei riferimenti a memoria, della cache dati (solo quando cambia) e della memoria (solo quando cambia). Ricordiamo che la politica write-back prescrive l'aggiornamento della memoria principale solo quando il blocco modificato deve essere rimpiazzato in cache. Pertanto, con un (*) vicino ad un blocco ricordiamo che si tratta di un blocco "sporco" (dirty) e che quindi deve essere copiato in memoria principale quando viene rimpiazzato.

codice eseguito	[R1] hex	[R2] hex	[R3] hex	[R4] hex	ind. rif. memoria hex binario	cache dati [linea 0] t: tag r: rif.	modifica memoria mem[ind.] = cont.	
LB R3, 0(R0)	?	?	F	?	00 00000000	[0F 00] t:000000 r:miss		
ADD R2, R0, R0	?	0	F	?				
LB R1, 8(R2)	AE	0	F	?	08 00001000	[AE 13] t:000010 r:miss		
ADDI R2, R2, 5	AE	5	F	?				
SUB R4, R3, R2	AE	5	F	A				
ADDI R1, R1, 7	B5	5	F	A				
SB R1, 8(R2)	B5	5	F	A	0D 00001101	[A1 42] t:000011 r:miss		
						[A1 B5]* t:000011 r:write		
BGTZ R4, -6	B5	5	F	A				
LB R1, 8(R2)	B5	5	F	A	0D 00001101	[A1 B5]* t:000011 r:hit		
ADDI R2, R2, 5	B5	A	F	A				
SUB R4, R3, R2	B5	A	F	5				
ADDI R1, R1, 7	BC	A	F	5				
SB R1, 8(R2)	BC	A	F	5	12 00010010	[A1 B5]* t:000011 r:	[00 00] t:000100 r:miss	
						[A1 B5]* t:000011 r:	[BC 00]* t:000100 r:write	
BGTZ R4, -6	BC	A	F	5				
LB R1, 8(R2)	BC	A	F	5	12 00010010	[A1 B5]* t:000011 r:	[BC 00]* t:000100 r:hit	
ADDI R2, R2, 5	BC	F	F	5				
SUB R4, R3, R2	BC	F	F	0				
ADDI R1, R1, 3	C3	F	F	0				
SB R1, 8(R2)	C3	F	F	0	17 00010111	[A1 B5]* t:000011 r:	[03 71] t:000101 r:miss	mem[12] = BC (write-back) mem[13] = 00 (write-back)
						[A1 B5]* t:000011 r:	[03 C3] t:000101 r:write	
BGTZ R4, -6	C3	F	F	0				