



Dipartimento di Elettronica e Informazione

Politecnico di Milano

20133 Milano (Italia)
Piazza Leonardo da Vinci, 32
Tel. (+39) 02-2399.3400
Fax (+39) 02-2399.3411

Advanced Computer Architecture

June 25, 2020

Prof. Marco Santambrogio

Name Marco && Davide
Last Name Santambrogio && Conficconi

The student has to choose 2 out of the 3 exercises.

If all the 3 exercises will be completed, just the 2 with the lowest score will be considered.

This is an open book exam, which means the students can you use all their notes (we do not care of the format) but they cannot share their exams with anyone during the test.

At the end of the exam, when the professor will call it, each student has to upload 1 PDF file (named using the “codice persona” – CodicePersona.pdf) using the Microsoft Form share via the Zoom chat at the beginning of the exam.

In the first page of the uploaded PDF, the professor has to find the name and the surname of the student.

Problem 1 (50%)	
Problem 2 (50%)	
Problem 3 (50%)	
Total (100%)	

Problem 1

Question1 - To implement cache coherency one of the protocols is the MESI concurrency protocol implemented for a write-invalidate write-back cache. Describe briefly the algorithm (also in term of graph) and complete the following table.

State	Cache up to date?	Memory up to date?	Others have a copy?	Cache can respond other's reads?
Modified				
Exclusive				
Shared				
Invalid				

Problem 1

Question1 - To implement cache coherency one of the protocols is the MESI concurrency protocol implemented for a write-invalidate write-back cache. Describe briefly the algorithm (also in term of graph) and complete the following table.

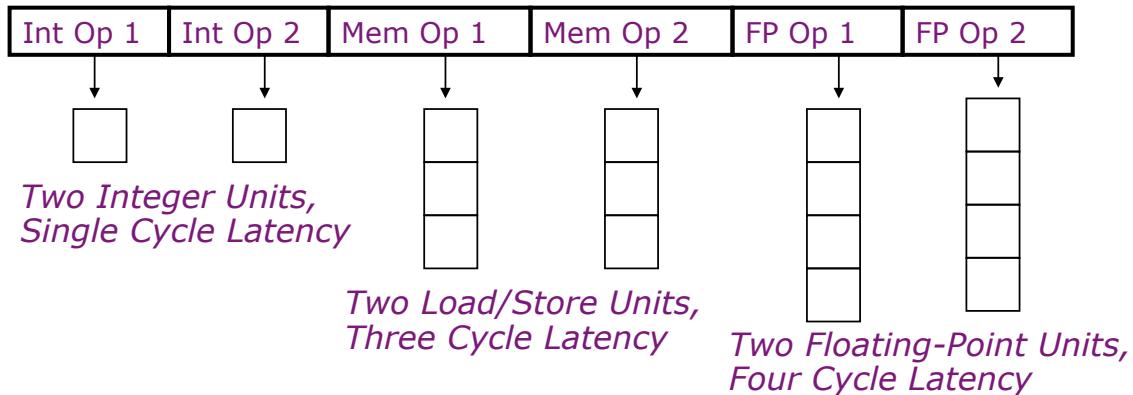
State	Cache up to date?	Memory up to date?	Others have a copy?	Cache can respond other's reads?
Modified				
Exclusive				
Shared				
Invalid				

Question 2 - Consider the following access pattern on a two-processor system with a direct-mapped, write-back cache with one cache block and a two cache block memory. Assume the MESI protocol is used, with write-back caches, and invalidation of other caches on write (instead of updating the value in the other caches). **NOTE: you can recreate the table in excel, you can draw it on paper, you can create a "ASCII" version of it... the only thing is that, at the end, the information has to be clear and not ambiguous.**

Time	After Operation	P1 cache block state	P2 cache block state	Memory at block 0 up to date?	Memory at block 1 up to date?
0	P1: read block 1				
1	P2: read block 0				
2	P1: write block 0				
3	P2: read block 1				
4	P1: read block 1				
5	P2: write block 1				
6	P1: read block 1				
7	P2: write block 1				
8	P1: read block 0				
9	P2: write block 1				
10	P2: write block 1				
11	P1: read block 1				
12	P2: write block 1				
13	P1: read block 1				
14	P2: read block 1				

Problem 2

Considering the following VLIW “architecture”:



Considering the following portion of assembly:

Loop:

```
LD F0, 0(R1)
LD F6, 8(R1)
LD F10, 16(R1)
LD F14, 24(R1)
FADD F4, F0, F2
FADD F8, F6, F0
FADD F12, F10, F14
FADD F16, F14, F2
SD F4, 0(R1)
SD F8, 8(R1)
ADD R1, 32
SD F12, -16(R1)
BNE R1 R2 Loop
SD F16, -8(R1)
```

Question1 - Describe the corresponding VLIW code:

NOTE: you can recreate the table in excel, you can draw it on paper, you can create a "ASCII" version of it... the only thing is that, at the end, the information has to be clear and not ambiguous.

Answer:

Clock	Int1	Int2	Mem1	Mem2	FP+ (=ADD)	FPx (x= MUL)
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						

Question2 – How many FLOPS/cycle?

Answer:

Problem 3

Given the following assembly code and knowing that R0 is set to 100 and R1 is set to 0:

LOOP:	LD	F1	0	R0
	ADDD	F2	F1	F1
	ADDI	R1	R1	100
LOOP2:	MULTD	F2	F2	F1
	SUBI	R1	R1	1
	BNEZ	R1		LOOP2
	SUBI	R0	R0	1
	BNEZ	R0		LOOP

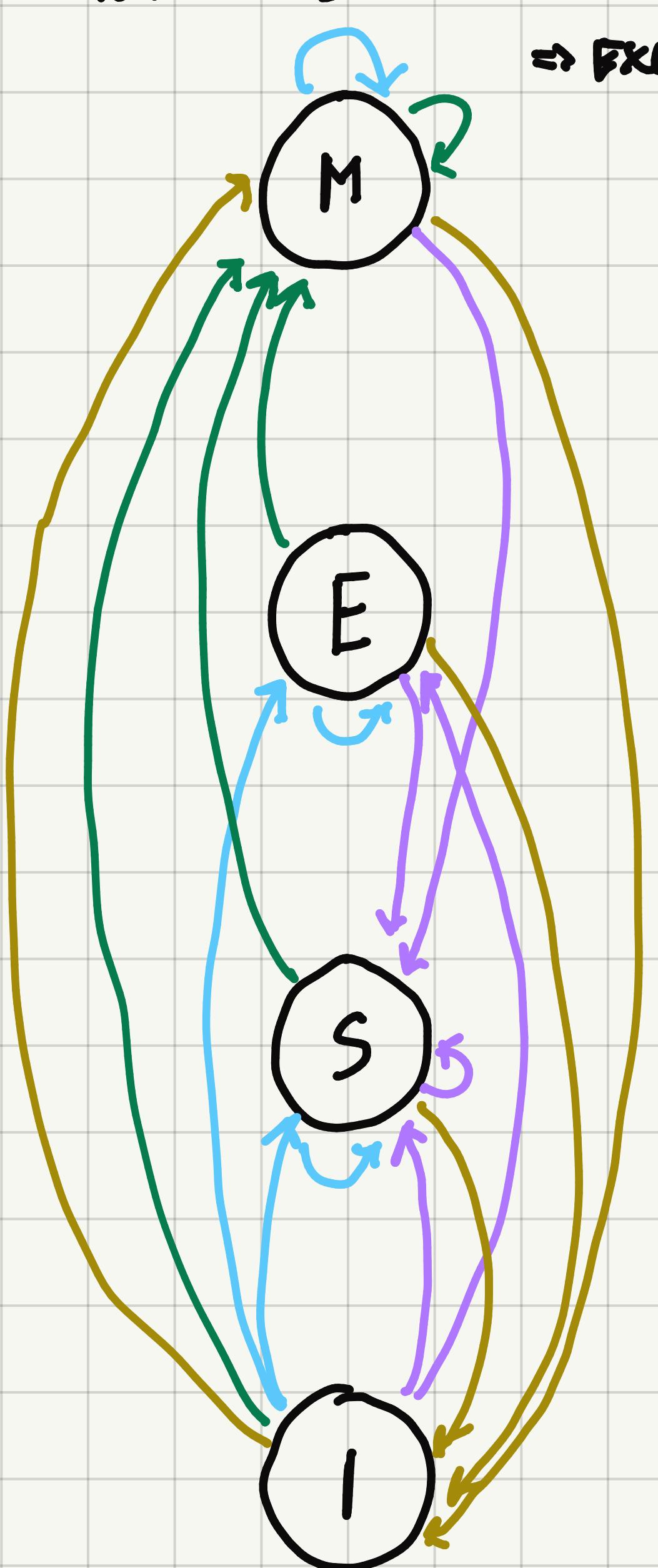
Question 1 - Design and describe (all the decisions have to be motivated) a 1-BHT and a 2-BHT able to execute the previous code. The obtained result, in terms of miss predictions, is inline with theoretical characteristics of the two predictors? Please effectively support your answer.

Answer:

Question 2 – Is it true that the 1-BHT is always outperformed by the 2-BHT. Please, effectively support your answer.

1 MESI (Modified, Exclusive, Shared, Invalid) is an example of snooping protocols, common in the context of MIMD architectures with centralized shared memory. It can tell, for each operation, how it affects the whole states

- Pr RI → (LOCAL) PROCESSOR READ READ FROM MEMORY
- Pr Wr → (LOCAL) PROCESSOR WRITE WRITE ON MEMORY
- Bus Rd → BUS READ ANOTHER PROCESSOR ATTEMPTS TO READ
- Bus Rdx → BUS READ EXCLUSIVE ANOTHER PROCESSOR ATTEMPTS TO WRITE
⇒ EXCLUSIVE READ AND INVALIDATION OF OTHER COPIES



State	Cache up to date?	Memory up to date?	Others have a copy?	Cache can respond other's reads?
Modified	YES	NO	NO	YES
Exclusive	YES	NO	NO	YES
Shared	YES	YES	YES	No
Invalid	NO	MAYBE	MAYBE	NO

Question 2 - Consider the following access pattern on a two-processor system with a direct-mapped, write-back cache with one cache block and a two cache block memory. Assume the MESI protocol is used, with write-back caches, and invalidation of other caches on write (instead of updating the value in the other caches). **NOTE: you can recreate the table in excel, you can draw it on paper, you can create a "ASCII" version of it... the only thing is that, at the end, the information has to be clear and not ambiguous.**

Time	After Operation	P1 cache block state	P2 cache block state	Memory at block 0 up to date?	Memory at block 1 up to date?
0	P1: read block 1	E(1)	I	YES	YES
1	P2: read block 0	E(1)	E(0)	YES	YES
2	P1: write block 0	M(0)	I	NO	YES
3	P2: read block 1	M(0)	E(1)	NO	YES
4	P1: read block 1	S(1)	S(1)	YES	YES
5	P2: write block 1	I	M(1)	YES	NO
6	P1: read block 1	S(1)	S(1)	YES	YES
7	P2: write block 1	I	M(1)	YES	NO
8	P1: read block 0	E(0)	M(1)	YES	NO
9	P2: write block 1	E(0)	M(1)	YES	NO
10	P2: write block 1	E(0)	M(1)	YES	NO
11	P1: read block 1	S(1)	S(1)	YES	YES
12	P2: write block 1	I	M(1)	YES	NO
13	P1: read block 1	S(1)	S(1)	YES	YES
14	P2: read block 1	S(1)	S(1)	YES	YES

COMMIT 0 TO READ 1 COMMIT WHEN NEEDED

2

Question1 - Describe the corresponding VLIW code:

NOTE: you can recreate the table in excel, you can draw it on paper, you can create a "ASCII" version of it... the only thing is that, at the end, the information has to be clear and not ambiguous.

Answer:

Clock	Int1	Int2	Mem1	Mem2	FP+ (+=ADD)	FPx (x= MUL)
1			ld F0, 0(R1)		1	
2			ld F6, 8(R1)		2	1
3			ld F10, 16(R2)		1 3	2
4			ld F14, 24(R2)	1 2	Fadd F4, F0, F2 3	
5				2 3	Fadd F3, F6, F0	
6				5 3		
7				4	Fadd F2, F10, F4	
8			sd F4, 0(R1)		2 Fadd M6, F14, F0	
9	add R4, 32 1		sd F8, 8(R2)		3	
10	BNE R2, R2, loop				4	
11			sd F12, 16(R2)			
12			sd F16, 32(R2)			

Question2 – How many FLOPS/cycle?

Answer:

$$\frac{\text{#FPt} + \text{#FPx}}{\text{CC}} = \frac{4}{12} = \frac{1}{3}$$

Problem 3

Given the following assembly code and knowing that R0 is set to 100 and R1 is set to 0:

LOOP:	LD	F1	0	R0
	ADDD	F2	F1	F1
	ADDI	R1	R1	100
LOOP2:	MULTD	F2	F2	F1
	SUBI	R1	R1	1
	BNEZ	R1	LOOP2	
	SUBI	R0	R0	1
	BNEZ	R0	LOOP	

Question 1 - Design and describe (all the decisions have to be motivated) a 1-BHT and a 2-BHT able to execute the previous code. The obtained result, in terms of miss predictions, is inline with theoretical characteristics of the two predictors? Please effectively support your answer.

Answer:

Question 2 – Is it true that the 1-BHT is always outperformed by the 2-BHT. Please, effectively support your answer.

1

$R_0=100$ $R_1=0$

LOOP:	LD	F1	0	R0	→ $R_1=100$
	ADDD	F2	F1	F1	
	ADDI	R1	R1	100	
LOOP2:	MULTD	F2	F2	F1	LOOP2 100 ITERATIONS
	SUBI	R1	R1	1	
	BNEZ	R1	LOOP2		
	SUBI	R0	R0	1	
	BNEZ	R0	LOOP		

Problem 3

Given the following assembly code and knowing that R0 is set to 100 and R1 is set to 0:

LOOP:	LD	F1	0	R0
	ADDD	F2	F1	F1
	ADDI	R1	R1	100
LOOP2:	MULTD	F2	F2	F1
	SUBI	R1	R1	1
	BNEZ	R1	LOOP2	
	SUBI	R0	R0	1
	BNEZ	R0	LOOP	

Question 1 - Design and describe (all the decisions have to be motivated) a 1-BHT and a 2-BHT able to execute the previous code. The obtained result, in terms of miss predictions, is inline with theoretical characteristics of the two predictors? Please effectively support your answer.

Answer:

Question 2 – Is it true that the 1-BHT is always outperformed by the 2-BHT. Please, effectively support your answer.

1

$R_0 = 100$ $R_1 = 0$

LOOP:	LD	F1	0	R0	→ $R_1 = 100$
	ADDD	F2	F1	F1	
	ADDI	R1	R1	100	
LOOP2:	MULTD	F2	F2	F1	LOOP2 100 ITERATIONS
	SUBI	R1	R1	1	
	BNEZ	R1	LOOP2		
	SUBI	R0	R0	1	
	BNEZ	R0	LOOP		

1-BHT NO collision

LOOP:	LD	F1	0	R0]
	ADDD	F2	F1	F1	
	ADDI	R1	R1	100	
LOOP2:	MULTD	F2	F2	F1	
	SUBI	R1	R1	1	
T	BNEZ	R1	LOOP2	100	

$$T \rightarrow NT \quad NT \rightarrow T \quad T \rightarrow NT \\ 1 + (1+1) \cdot (100-1) +$$

$$+ 1 = 200$$

2-BHT NO collision

LOOP:	LD	F1	0	R0]
	ADDD	F2	F1	F1	
	ADDI	R1	R1	100	
LOOP2:	MULTD	F2	F2	F1	
NT _λ	SUBI	R1	R1	1	
T _λ	BNEZ	R1	LOOP2	100	

$$T_L \rightarrow T_W \quad T_{L \rightarrow T_W} \\ 1 \cdot 100 + 1 = 101$$

$$NT \rightarrow T \quad T_L \rightarrow T_W \quad NT \rightarrow T \quad T_{L \rightarrow T_W} \\ W.C.) \quad 2 + 1 \cdot 100 + 2 + 1 = 105$$

W.C. 2BHT > B.C. 1-BHT \Rightarrow INNING WITH THEORETICAL CHARACTERISTICS

1-BHT WITH COLLISION

LOOP:	LD	F1	0	R0
	ADDD	F2	F1	F1
	ADDI	R1	R1	100
LOOP2:	MULTD	F2	F2	F1
	SUBI	R1	R1	1
T	BNEZ	R1	LOOP2	
	SUBI	R0	R0	1
	BNEZ	R0	LOOP	

$$\begin{array}{l}
 T \rightarrow NT \quad NT \rightarrow T \\
 \uparrow \qquad \uparrow \\
 (1+1) \cdot (100-1) + 1 = 199
 \end{array}$$

SINCE WE SURELY KNOW, BASED
 ON THE CODE, THAT THE 1ST BRANCH
 IS TAKEN, WE SMART THE

PREDICTION AS T

2-BHT WITH COLLISION

LOOP:	LD	F1	0	R0
	ADDD	F2	F1	F1
	ADDI	R1	R1	100
LOOP2:	MULTD	F2	F2	F1
	SUBI	R1	R1	1
T _b	BNEZ	R1	LOOP2	
	SUBI	R0	R0	1
	BNEZ	R0	LOOP	

$$\begin{array}{l}
 T_1 \rightarrow T_W \quad T_W \rightarrow T_4 \\
 (1+0) \cdot (100-1) + 1 + 1 = 101
 \end{array}$$

THE RESULTS ARE THEORETICALLY OK, AS 2-BHT SHOULD OUTPERFORM 1-BHT

BUT, IN GENERAL, IT IS NOT TRUE (I.E.: NESTED LOOPS)