03 Febbraio 2021 – Architetture dei Sistemi di Elaborazione

A1

Name, Student ID ....

Considerare la seguente architectura MIPS64:

- Integer ALU: 1 clock cycle
- Data memory: 1 clock cycle
- FP multiplier unit: pipelined 6 stages
- FP arithmetic unit: pipelined 2 stages
- FP divider unit: not pipelined unit that requires 7 clock cycles
- branch delay slot: 1 clock cycle, and the branch delay slot disabled
- forwarding enabled
- è possibile completare lo stage EXE di una istruzion in modo out-of-order.

○ Facendo riferimento al frammento di codice riportato, si mostrino le tempistiche relative all’esecuzione ciascuna istruzione e si calcoli il numero totale di clock cycles necessari per eseguire completamente il programma:

```
for (i = 0; i < 100; i++) {
   v5[i] = v1[i]/v2[i];
   v6[i] = (v1[i]*v2[i])+(v3[i]/v4[i]);
}
```

.data																																								Clock cycles
V1: .double “100 values”																																								
V2: .double “100 values”																																								
V3: .double “100 values” … V5: .double “100 zeros”																																								
V4: .double “100 values”																																								
V5: .double “100 values”																																								
V6: .double “100 values”																																								
.text																																								
main: daddui r1,r0,0	F	D	E	M	W																																			5
daddui r2,r0,100		F	D	E	M	W																																		1
loop: l.d f1,v1(r1)			F	D	E	M	W																																	1…2
l.d f2,v2(r1)				F	D	E	M	W																																1
l.d f3,v3(r1)					F	D	E	M	W																															1
l.d f4,v4(r1)						F	D	E	M	W																														1

03 Febbraio 2021 – Architetture dei Sistemi di Elaborazione

A1

Name, Student ID .....

div.d f5,f1,f2							F	D	/	/	/	/	/	/	/	M	W																						7
s.d f5,v5(r1)								F	D	E	S	S	S	S	S	S	M	W																					1
mul.d f6,f1,f2									F	D	*	*	*	*	*	*	S	M	W																			1
div.d f7,f3,f4										F	D	S	S	S	S	/	/	/	/	/	/	/	M	W														5
add.d f1,f6,f7											F	S	S	S	S	D	S	S	S	S	S	S	+	+	M	W												2
s.d f1,v6(r1)																F	S	S	S	S	S	S	D	E	S	M	W											1
daddui r1,r1,8																							F	D	S	E	M	W										1
daddi r2,r2,-1																								F	S	D	E	M	W									1
bnez r2,loop																										F	S	D	E	M	W							2
halt																												F	N	N	N	N						0…1
Total	6 + 100 * 26																																					2606

Question 2

Considerando il programma precedente e l'architettura del processore superscalare descritto in seguito; completare la tabella relativa alle prime 2 iterazioni.
Processor architecture:

- Issue 2 instructions per clock cycle
- jump instructions require 1 issue
- handle 2 instructions commit per clock cycle
- timing facts for the following separate functional units:
    i. 1 Memory address 1 clock cycle
    ii. 1 Integer ALU 1 clock cycle
    iii. 1 Jump unit 1 clock cycle
    iv. 1 FP multiplier unit, which is pipelined: 6 stages
    v. 1 FP divider unit, which is not pipelined: 7 clock cycles
    vi. 1 FP Arithmetic unit, which is pipelined: 2 stages
- Branch prediction is always correct
- There are no cache misses
- There are 2 CDB (Common Data Bus).

# iteration		Issue	EXE	MEM	CDB x2	COMMIT x2
1	l.d f1,v1(r1)	1	2	3	4	5
1	l.d f2,v2(r1)	1	3	4	5	6
1	l.d f3,v3(r1)	2	4	5	6	7
1	l.d f4,v4(r1)	2	5	6	7	8
1	div.d f5,f1,f2	3	6	—	13	14
1	s.d f5,v5(r1)	3	6	—	—	14
1	mul.d f6,f1,f2	4	6	—	12	15
1	div.d f7,f3,f4	4	13	—	20	21
1	add.d f1,f6,f7	5	21	—	23	24
1	s.d f1,v6(r1)	5	7	—	—	24
1	daddui r1,r1,8	6	7	—	8	25
1	daddi r2,r2,-1	6	8	—	9	25
1	bnez r2,loop	7	10	—	—	26

2	l.d f1,v1(r1)	8	9	10	11	26
2	l.d f2,v2(r1)	8	10	11	12	27
2	l.d f3,v3(r1)	9	11	12	13	27
2	l.d f4,v4(r1)	9	12	13	14	28
2	div.d f5,f1,f2	10	20	—	27	28
2	s.d f5,v5(r1)	10	13	—	—	29
2	mul.d f6,f1,f2	11	13	—	19	29
2	div.d f7,f3,f4	11	27	—	34	35
2	add.d f1,f6,f7	12	35	—	37	38
2	s.d f1,v6(r1)	12	14	—	—	38
2	daddui r1,r1,8	13	14	—	15	39
2	daddi r2,r2,-1	13	15	—	16	39
2	bnez r2,loop	14	17	—	—	40