3 Febbraio 2020 - ASE - Architetture moderne

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 divider unit: not pipelined unit that requires 6 clock cycles
 branch delay slot: 1 clock cycle, and the branch delay slot disabilitato
- forwarding abilitato
- è possibile completare lo stage EXE di una istruzion in modo out-of-order.
- o 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 comletamente il programma.

.data

V1: .double "100 values"

.double "100 values"

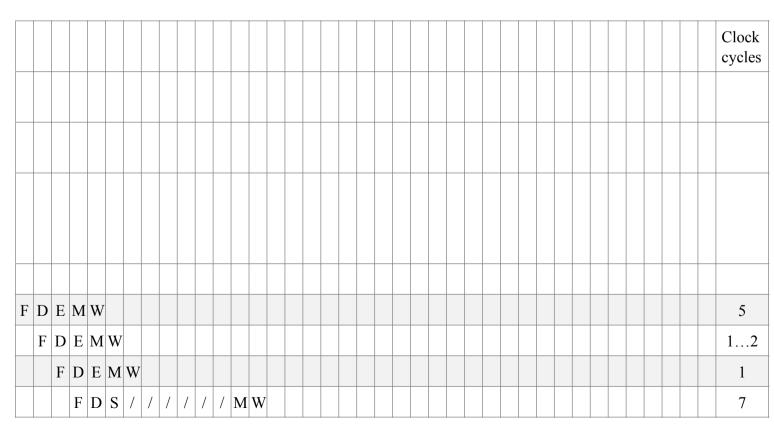
V3: .double "100 values"

.double "100 V5:

.text

main: daddui

loop: l.d fl,v1(r1)1.d f2,v2(r1)div.d f5,f1,f2



3 Febbraio 2020 - ASE - Architetture moderne

Name, Student ID

1.d f3,v3(r1) F S D E M W 0 F|D|E|M|W|1.d f4,v4(r1)0 F D S S S / / / / / / / MW div.d f6,f5,f3 6 F S S S D S S S S S + + H W add.d f6,f6,f4 3 F | S | S | S | S | D | E | S | S | M | W s.d f6,v5(r1) F|D|S|S|E|M|W|daddui r1,r1,-8 F S S S D E M W bnez r1,loop 2 |F|N|N|N|N|halt 0...1 **Total** 5 + 100 * 232305

3 Febbraio 2020 - ASE - Architetture moderne

Name, Student ID

Considerato un programma basato su loop, ed assumendo che il processore utilizzato sia un MIPS64 che implementa multiple-issue e speculation:

- Issue di 2 instruzioni per clock cycle
- Instruzioni jump richiedono 1 issue
- Esegui il commit di 2 istruzioni per clock cycle
- Le unità funzionali hanno le seguenti caratteristiche:
 - 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: 8 stages
 - v. 1 FP divider unit, which is not pipelined: 8 clock cycles
 - vi. 1 FP Arithmetic unit, which is pipelined: 4 stages
- La predizione di salto è sempre corretta
- Non ci sono cache misses
- Essitono 2 CDB (Common Data Bus).
- Si complete la tabella mostrando il comportamento del processore durante le 3 iniziali iterazioni

# iterazione	Instruction	ISSUE	EXE	MEM	CDBx2	COMMITx2
1	l.d f1,v1(r1)	1	2	3	4	5
1	1.d f2,v2(r1)	1	3	4	5	6
1	mul.d f1,f1,f1	2	5	_	13	14
1	mul.d f2,f2,f2	2	6	_	14	15
1	div.d f5,f1,f2	3	15	_	23	24
1	s.d f5,v3(r1)	3	4	_	_	24
1	daddui r1,r1,-8	4	5	_	6	25
1	bnez r1,loop	5	7	_	_	25
2	1.d f1,v1(r1)	6	7	8	9	26
2	1.d f2,v2(r1)	6	8	9	10	26

B1

3 Febbraio 2020 - ASE - Architetture moderne

Name, Student ID

2	mul.d f1,f1,f1	7	10	_	18	27
2	mul.d f2,f2,f2	7	11	_	19	27
2	div.d f5,f1,f2	8	23	_	31	32
2	s.d f5,v3(r1)	8	9	_	_	32
2	daddui r1,r1,-8	9	10	_	11	33
2	bnez r1,loop	10	12	_	_	33
3	l.d f1,v1(r1)	11	12	13	14	34
3	l.d f2,v2(r1)	11	13	14	15	34
3	mul.d f1,f1,f1	12	15	_	23	35
3	mul.d f2,f2,f2	12	16	_	24	35
3	div.d f5,f1,f2	13	31	_	39	40
3	s.d f5,v3(r1)	13	14	_	_	40
3	daddui r1,r1,-8	14	15	_	16	41
3	bnez r1,loop	15	17	_	_	41