T-61.123 TIETOKONEEN ARKKITEHTUURI (COMPUTER ARCHITECTURE)

Exam on 6th March 2007 by Seppo Haltsonen

Write on each answer sheet:

- name and student number
- department and year of study
- if the project has been done, the year

1. Give a short description of the following concepts:
   a) CPI
   b) CPU
   c) ALU
   d) MIPS (not the name of the computer discussed in the course!)
   e) RISC
   f) DMA

2. a) There is a positive integer in register $a0. Write a program that calculates the sum of all positive integers less than this one. The result has to be in register $a1.

   b) What does branch prediction mean? Explain the different possibilities to implement branch prediction.

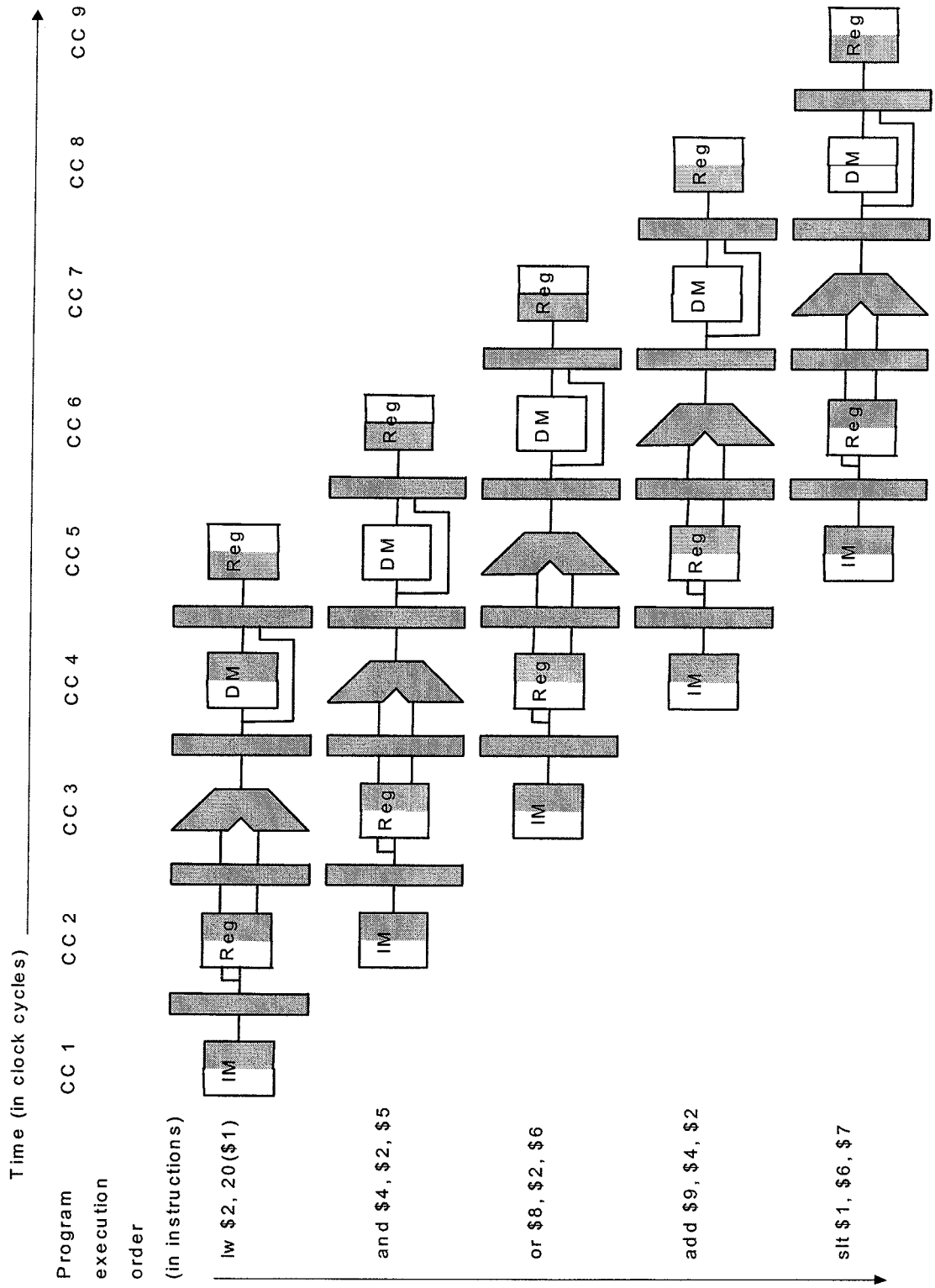
3. What are exceptions and interrupts? How are they dealt with in MIPS?

4. List all data dependencies between the instructions of the attached pipeline figure. Using the figure, describe which ones of the data dependencies are hazards and which ones are not. Describe also how the hazards can be solved.

5. Explain the implementation and operation of direct mapped cache in detail.

MIPS R3000 Fixed-Point Instruction Set Summary

Category	Instruction	Example	Meaning	Comments
Arithmetic	add	add $1,$2,$3	$1 = $2 + $3	3 operands; exception possible
	subtract	sub $1,$2,$3	$1 = $2 - $3	3 operands; exception possible
	add immediate	addi $1,$2,100	$1 = $2 + 100	+ constant; exception possible
	add unsigned	addu $1,$2,$3	$1 = $2 + $3	3 operands; exception not possible
	subtract unsigned	subu $1,$2,$3	$1 = $2 - $3	3 operands; exception not possible
	add immediate unsigned	addiu $1,$2,100	$1 = $2 + 100	+ constant; exception not possible
Logical	and	and $1,$2,$3	$1 = $2 & $3	3 register operands; Logical AND
	or	or $1,$2,$3	$1 = $2 | $3	3 register operands; Logical OR
	and immediate	andi $1,$2,100	$1 = $2 & 100	Logical AND register, constant
	or immediate	ori $1,$2,100	$1 = $2 | 100	Logical OR register, constant
	shift left logical	sll $1,$2,10	$1 = $2 << 10	Shift left by constant
	shift right logical	srl $1,$2,10	$1 = $2 >> 10	Shift right by constant
Data transfer	load word	lw $1,(100)$2	$1 = Memory[$2+100]	Data from memory to register
	store word	sw $1,(100)$2	Memory[$2+100] = $1	Data from register to memory
	load upper immediate	lui $1,100	$1 = 100 * 2^{16}	Load constant in upper 16bits
Conditional branch	branch on equal	beq $1,$2,100	if ($1 == $2) go to PC+4+100	Equal test; PC relative branch
	branch on not equal	bne $1,$2,100	if ($1 != $2) go to PC+4+100	Not equal test; PC relative
	set on less than	slt $1,$2,$3	if ($2 < $3) $1 = 1; else $1 = 0	Compare less than; 2`s complement
	set less than immediate	slti $1,$2,100	if ($2 < 100) $1 = 1; else $1 = 0	Compare < constant; 2`s complement
	set less than unsigned	sltu $1,$2,$3	if ($2 < $3) $1 = 1; else $1 = 0	Compare less than; natural number
	set less than immediate unsigned	sltiu $1,$2,100	if ($2 < 100) $1 = 1; else $1 = 0	Compare constant; natural number
Unconditional jump	jump	j 10000	goto 10000	Jump to target address
	jump register	jr $31	goto $31	For switch, procedure return
	jump and link	jal 10000	$31 = PC + 4;go to 10000	For procedure call


Time (in clock cycles)
CC 1
CC 2
CC 3
CC 4
CC 5
CC 6
CC 7
CC 8
CC 9
Program execution order (in instructions)
lw $2, 20($1)
and $4, $2, $5
or $8, $2, $6
add $9, $4, $2
slt $1, $6, $7
IM
Reg
DM