Department of Computing Examinations — 2007–2008 Session *Confidential*

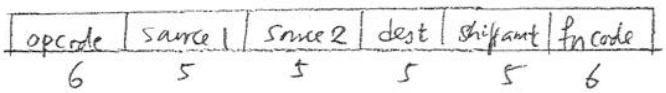
MODEL ANSWER and MARKING SCHEME

First Examiner wl Paper Code C210 = E2·13

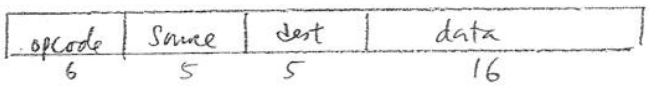
Second Examiner kkleung Question 1 Page 1 *out of* 1

Question labels in left margin Mark allocations in right margin

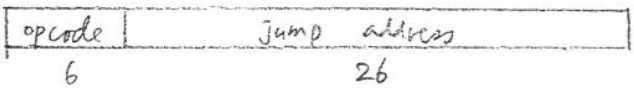
1a MIPS format R type



I type

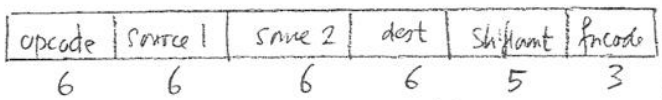


J type



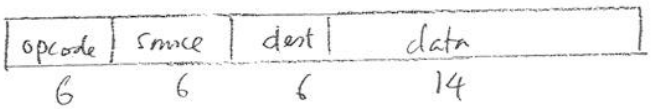
6

b 64 registers, need 6 bits to address register file



R type: shiftamt is only 5 bits, so may need 2 shift instructions

I type: data field reduced to 14 bits, reducing range of branches by a factor of 4

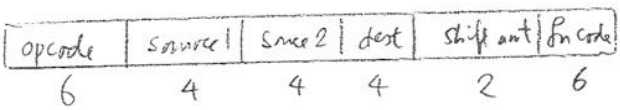


J type: no change since no registers are involved

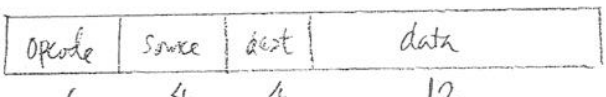
6

c 16 registers so need 4 bits to address register file

R type: shiftamt reduced to 2 bits, so may need multiple shift instr.



I type: data field reduced to 12 bits



J type: jump address reduced to 20 bits

advantage: smaller instruction, so smaller size, possibly faster

disadvantage: reduce data field or jump address field

8

Department of Computing Examinations — 2007–2008 Session	*Confidential*

MODEL ANSWER and MARKING SCHEME

First Examiner	wl	Paper Code	C210 = E2.13
Second Examiner	kk leung	Question 2	Page 1 *out of* 1

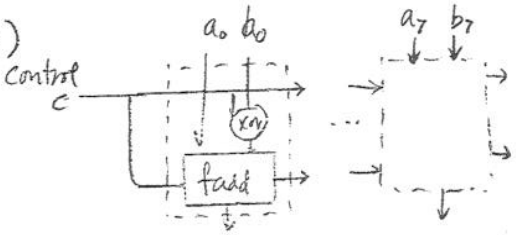
Question labels in left margin — Mark allocations in right margin

2a value of n-bit 2's comple. num: $a_{n-1} \cdots a_0$ is $-2^{n-1} a_{n-1} + 2^{n-2} a_{n-2} + \cdots + 2a_1 + a_0$

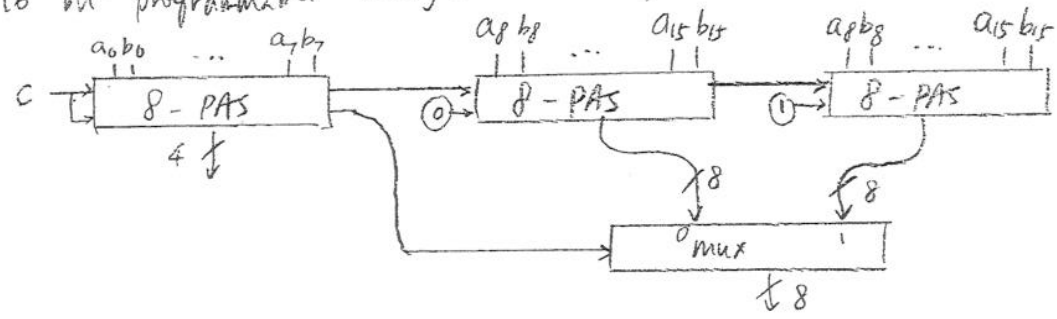
programmable adder/subtractor (8-PAS) — 5

$c = 0$: performs add

$c = 1$: " subtr



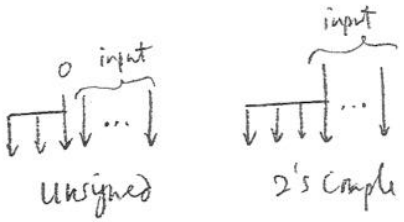
b 16 bit programmable carry select adder/subtractor — 6



c Sign extension: include more bits in the most significant position without changing the value of the representation. — 4

unsigned: MSBs zero

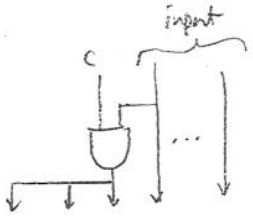
2's Comple: replicate the MSB



d programmable sign. ext. circuit — 5

$c = 0$ unsigned

$c = 1$ 2's comple



Department of Computing Examinations — 2007–2008 Session	*Confidential*

MODEL ANSWER and MARKING SCHEME

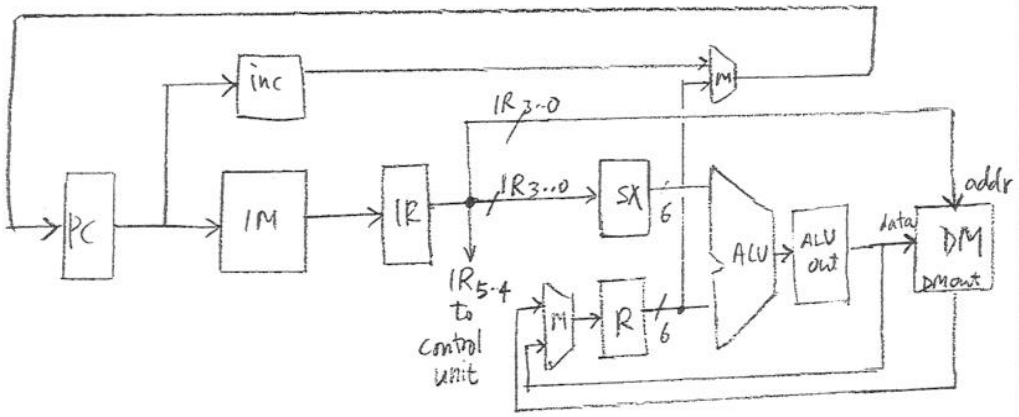
First Examiner wl	Paper Code C 210 = E2.13
Second Examiner kkleung	Question 3 Page 1 *out of* 1

Question labels in left margin — Mark allocations in right margin

3a add: R = R + C load: R = DM [C], store: DM [C] = R — 4

Jump: PC = IM [R]

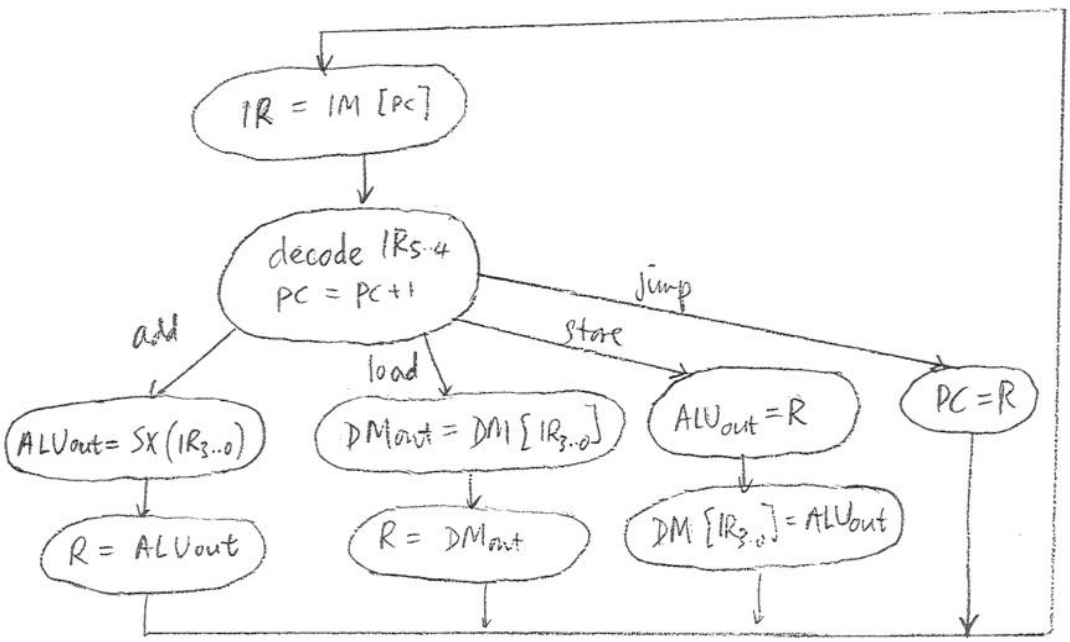
b



8

M = Mux

c



8

Department of Computing Examinations — 2007–2008 Session *Confidential*

MODEL ANSWER and MARKING SCHEME

First Examiner wl Paper Code C210 = E213

Second Examiner kkleung Question 4 Page 1 *out of* 1

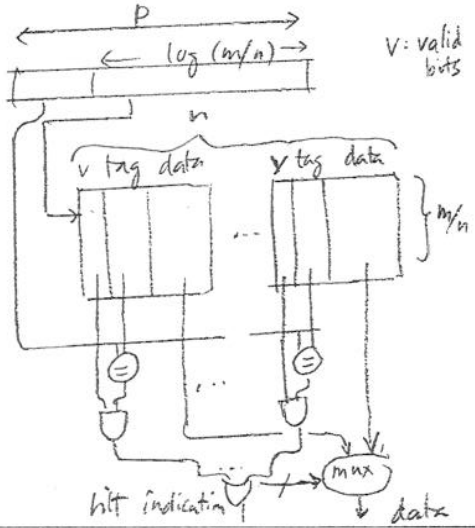
Question labels in left margin Mark allocations in right margin

4a dir-mapped - adv: simple, fast, disadv: high conflict misses — 4
fully assoc - adv: low conflict misses, disadv: complex, costly

b n-way set assoc:
- contains a number of sets, each set contains n-blocks
- each block maps to a unique set in the cache given by the index field
- a block can be placed in any element of that set
- all n-blocks need to be searched for a given block, so increasing n would increase size, reduce cache speed — 6

c TLB: translation lookaside buffer, a cache for page table, — 3
Increase assoc. reduces misses, and misses are costly in TLB so a fully associative cache minimises misses

d



number of sets = m/n (m blocks)

total number of tag bits
$= m(p - \log(m/n))$

total number of bits = num of valid bits + num of tag bits + number of bits for page addr data

$= m + m(p - \log(m/n)) + mp$

$= m(1 + 2p - \log(m/n))$ — 7

$\frac{m}{n} \cdot (p - \log(m/n)) \cdot n$
(m/n: num of sets; $p - \log(m/n)$: tag width; n: n-way)