CSc/CpE 142 – Exam 2 Name. [redacted]

1) Fill in the blanks /Short answer (30 points, 3 points each)

a) ..Speculation.. is a method of resolving a branch hazard that assumes a given outcome for the branch and proceeds based on that assumption rather than waiting for the actual outcome.

b) Increasing ..block size.. can improve hit ratio due to temporal locality. -3

c) In a direct-mapped cache each block has only one word. True or False. ✓

d) The five stages in a pipeline implementation require 4, 5, 4, 2, and , 3 nanoseconds. What is the fastest clock rate at which this datapath can operate?..... -3 $\frac{5}{5+5+5+5+5} = \frac{5}{25} = 0.20 =$ 200ms

e) List two primary methods for enhancing instruction-level parallelism.
   I. ..Deeper pipeline..
   II. ..Multi-issue pipeline

f) ..Block.. is the unit of information transferred between the cache and main memory on a miss.

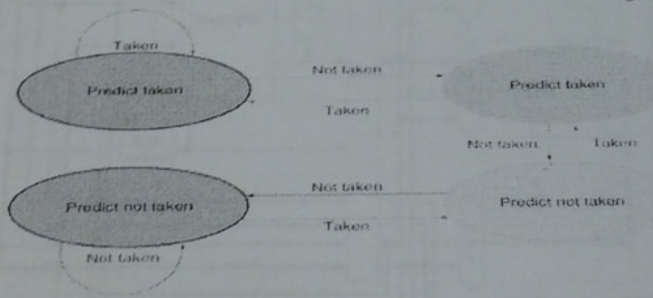
g)..Write through.. -3 is a scheme (policy) which handles writes by updating values only to the block in the cache, then writing the modifies block to the lower level of the hierarchy when the block is replaced.

h) Temporal locality is the principle stating that if a data location is referenced, then it will tend to be referenced again soon.

i) Consider a 2-bit branch prediction scheme illustrated by the following figure. What is the accuracy of this ***2-bit predictor*** for the following repeating pattern of branch outcomes assuming the predictor starts in the top left state? .....

✓ ✗ ✓ ✗ ✗ ✗ ✗ ✗ ✗ ✓
T, NT, NT, NT, T, T, T, T, T, T

$= \frac{3}{10} =$ 30% -3



j) List three types of hazards that can degrade the performance of a pipelined datapath.
   i) Control
   ii) Data
   iii) structural

Name.....

CSc/CpE 142 – Exam 2

2) Consider the pipelined system shown below. Find the number of cycles required to execute the following MIPS code assuming that branch is not taken. Justify your result. (20 points)

```
        SLT $6, $8, $7
        LW $5, 100($6)
        LW $7, 100($15)
        BEQ $8, $15, L1
        SUB $9, $7, $4
        SUB $20, $9, $7
L1:     LW $8, 100($9)
        ADD $11, $8, $12
        SW $8, 104($9)
```

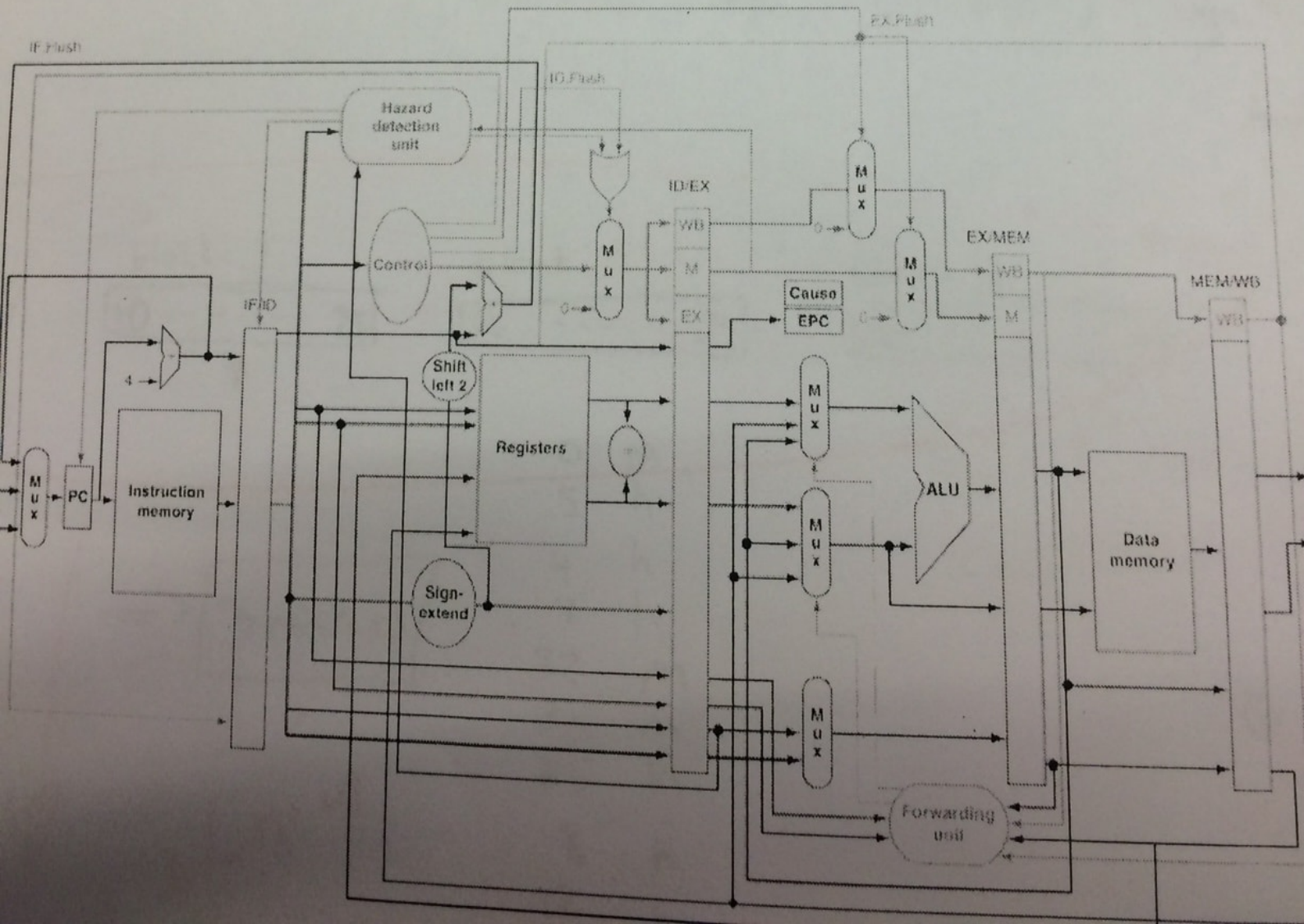
slt	IF	ID	EX	M	WB				
lw		IF	ID	EX	M	WB			
beq			IF	ID	EX	M	WB		
sub	??								
sub									
lw						IF	ID	EX	M
add					I.	??	IF	ID	EX
sw								IF	ID

Stn-1 + (delays)

↓ ↓ ↳ 1(branch taken) + 1(lw hazard)

5+7-1+[1+1] = 13 cycles

−6



2048	1024	512	256	128	64	32	16	8	4	2	1
								2^3	2^2	2^1	2^0

0	0 → F	15
16	10 → 25	31
32	26 → 1F	47
48	42 → 2F	63

0	0→9	9
10	A→F	15
16	10→19	

CSc/CpE 142 – Exam 2 Name.....

3) A computer uses 16-bit addresses, with byte addressable locations. Assume this system has a cache containing 32 (**thirty two**) **4-word** blocks. Each word is 16 bits long. Show the implementation of the cache assuming **4-way set associativity**.

answer on next page →

- You need to show how a given 16-bit address is divided into different parts to perform a cache reference. Show all necessary components of the cache. Provide the size of each component and the widths of its inputs and outputs, if applicable. (**15 points**)

(answer on next page)

- Assume the cache is empty. Find the number of misses for the following hexadecimal address sequence: 0, 5, 4, 7, 20, 21, 23, 8, C, FC. (**10 points**)

cache size: 32 4-word blocks ⇒ 4 words/block
⇒ 32 blocks

$\frac{32 \text{ blocks}}{\text{cache}} \times \frac{\text{set}}{4 \text{ blocks}} = \frac{8 \text{ sets}}{\text{cache}}$

word: 16 bits

4 way set ⇒ 4 blocks/set

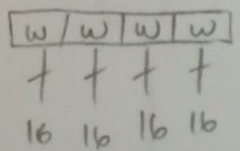
16 bit address

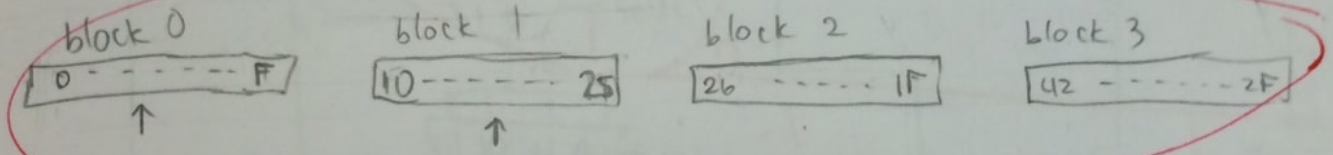
index: $\log_2 8 = 3$ bits

byte offset: $\log_2$ (byte in word) = $\log_2(2) = 1$ bits

word offset: $\log_2$ (word in block) = $\log_2 4 = 2$ bits

bytes in block





0 : miss
5 : hit
4 : hit
7 : hit
20 : miss
21 : miss
23 : hit
8 : hit
c : hit
FC : miss

⇒ 3 misses

0	m
5	h
4	h
7	h
32	m
33	m
35	h
8	h
12	h
1512	m

Wrong solution

-3

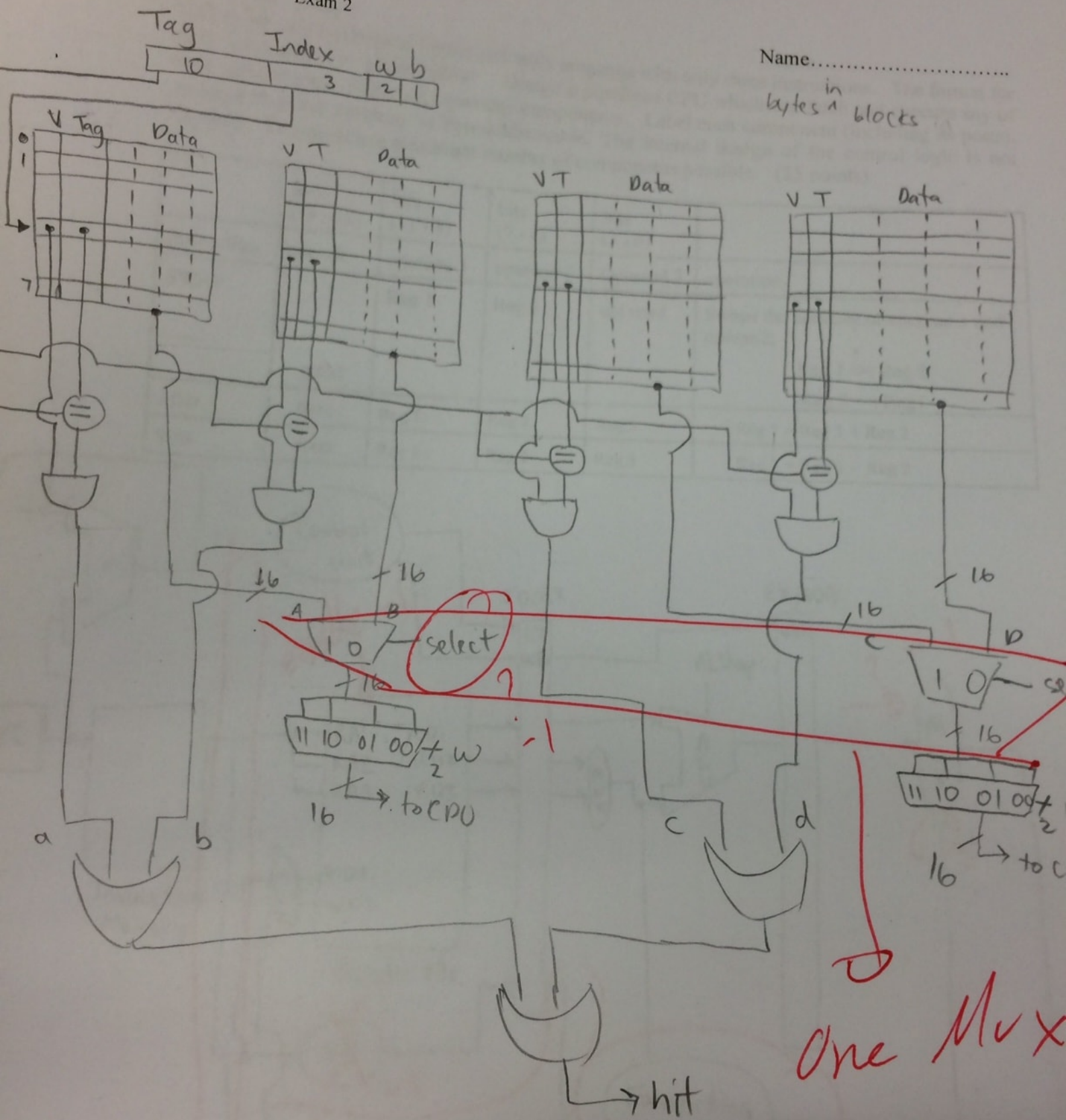
What do you mean by block 0, 1, 2, 3?

A	29
B	28
C	3
D	
E	
F	

16	10
17	11
18	12
19	13
20	14
21	15
22	16
23	17
24	
25	
26	
27	
28	

① ②

Name.....


Tag
Index
w b
10
3
2
1
V Tag Data
V T Data
V T Data
V T Data
16
A
B
C
D
1 0
select
11 10 01 00
2
w
16
to CPU
a
b
c
d
hit
One Mux

Name.....

4) Consider a hypothetical 16-bit assembly language with only three instructions. The format for these instructions is given below. Design a pipelined CPU which can fetch and execute any of these instructions. Show all necessary components. Label each component (including its ports). Assume that the memory is byte-addressable. The internal design of the control logic is not required. You must use minimum number of components possible. (25 points)

	Bits (15 - 12)	bits (11 - 8)	bits (7 - 4)	bits (3 - 0)	
Instruction	opcode	operand 1	operand 2	Operand 3	operation
SWAP	0000	Reg 1	Reg 2	not used	Swaps the contents of operand 1 and operan2; Reg 1 <= Reg 2 Reg 2 <= Reg1
ADD	0010	Reg 1	Reg 2	Reg 3	Reg 1 = Reg 3 + Reg 2
SUB	1000	Reg 1	Reg 2	Reg 3	Reg 1 = Reg 3 - Reg 2