选择题

I. Please choose the best answer and fill it into following table.

1	2	3	4	5	6	7	8	9	10
B	B	B	A	A	D	C	B	D	A

11
A

(1) What is the RISC-V assembly code for the binary:

0000 1110 1001 0101 0010 1000 0010 0011

A、sw x10, 240(x9)

B、sw x9, 240(x10)

C、sd x10, 240(x9)

D、sd x9, 240(x10)

(2) In order to put the content of address A(32 bits) to register x11, which one following is correct?

(A)
```
lui  x10, A_upper
ori  x10, A_lower
lw   x11, 0(x10)
```

(B)
```
lui  x10, A_upper
lw   x11, A_lower(x10)
```

(C)
```
lui  x10, A_upper
orx10, A_lower
lw   x11, 0(x10)
```

(D)
```
lui  x10, A_upper
or   x11, A_lower(x10)
```

where A_upper, A_lower are the high 20 bits and low 12 bits respectively.

(3) For the addition of $0.12345*10^{-2}$ 和 $0.12345*10^{+2}$, the first operation is aligning the exponents. After this operation, the aligned exponent is:

_____。

A、both -2　　B、both +2

C、one is -2,the other is +2　　D、both 0

(4) Which statement is correct?

(A) In virtual memory, the number of entries of a page table is equals to the physical page number.

(B) Increasing associativity can reduce Capacity miss.

(C) asynchronous bus: A bus that uses a handshaking protocol for coordinating usage rather than a clock; can accommodate a wide variety of devices of differing speeds.

(5) What is the right presentation of the result of

$(1.355)_{10}$-$(2.105)_{10}$?

A：1011 1111 0100 0000 0000 0000 0000 0000

B：1111 1111 1100 1000 0000 0000 0000 0000

C：1011 0001 0100 1000 0000 0000 0000 0000

D：1010 0001 0100 1000 0000 0000 0000 0000

(6) Single-cycle RISC-V datapath cannot complete which of the following operations in one clock cycle.

A、Reading data from and writing data to data memory

B、ALU computation and writing data to the register file

C、Updating PC and writing data to data memory

D、Reading data from register file, ALU computation and writing data to data memory

(7) (____) can be reduced by increasing the block size.

A: Compulsory misses

B: Capacity misses

C: Conflict misses

D: All three misses

(8) For a virtual memory with TLB, which of following will be run first during memory access? (____)

A: test cache hit B: test TLB hit

C：test physical memory hit　　D：test dirty bit

(9) In RAID, REDUNDANT means that more disks are used for _____.

A. enlarging capacity of disk system

B. improving transfer rate of system

C. finding and correcting the read/write error

D. improving reliability of disk system

(10) Given a direct-mapped cache with 16 blocks and each block has 4 words. What block number does byte address 840 map to?

A：2　　B：3　　C：4　　D：5

(11) Cache's write-through polity means write operation to main memory _____.

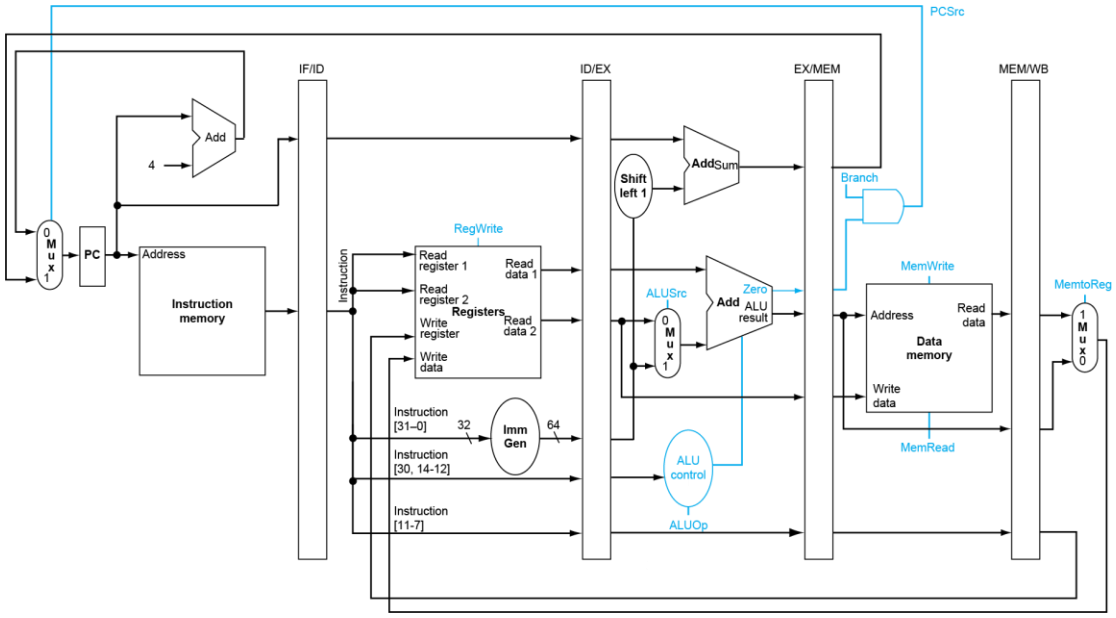
A. as well as to cache　　B. only when the cache is replaced

C. when the difference between cache and main memory is found

D. only when direct mapping is used

Chapter 4

1. Pipeline CPU Design



For the above single-cycle datapath, answer the questions:

(1) When execute sub instruction, the control signals PCSrc, ALUSrc, MemWrite, RegWrite, and MemtoReg generated by the controller are respectively (_0 0 0 1 0 _____).

Assuming that the following codes are executed on the above five-stage pipelined datapath:

```
sub x5, x7, x11
ld   x13, 8(x5)
ld   x7, 4(x2)
add   x13, x5, x13
sd   x13, 0(x5)
```

(2) If there is no forwarding or hazard detection, insert NOPs to ensure correct execution.

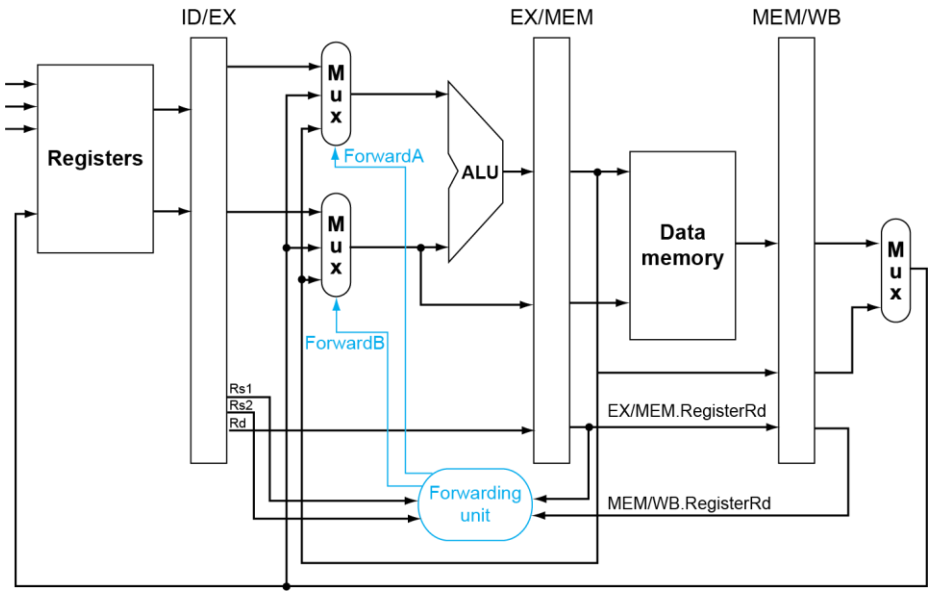
```
sub x5,x7,x11
nop
nop
ld  x13,8(x5)
ld  x7,4(x2)
nop
add x13,x5,x13
sd  x13,0(x15)
```

(3) If the processor has forwarding, but we forgot to implement the hazard detection unit,

redo (2)

```
sub x5, x7, x11
ld x13, 8(x5)
ld x7, 4(x2)
nop
add x13, x5, x13
sd x13, 0(x5)
```

(4) If there is forwarding (shown in following figure), for the first seven cycles, specify the forwarding signals for each cycle.



cycle	ForwardA	ForwardB	
1	X (don't care)	X (don't care)	No instruction in EX stage yet
2	X	X	No instruction in EX stage yet
3	10	X	EX/MEM.RegisterRd=ID/EX.RegisterRs1

4	X	X	EX/MEM.RegisterRd!=ID/EX.RegisterRs1/2&MEM/WB.RegisterRd!=ID/EX.RegisterRs1/2
5	01	X	MEM/WB.RegisterRd=ID/EX.RegisterRs1
6	X	X	EX/MEM.RegisterRd!=ID/EX.RegisterRs1/2&MEM/WB.RegisterRd!=ID/EX.RegisterRs1/2
7	X	X	New instruction unknown

Chapter 5

1. 书后 5.1 In this exercise we look at memory locality properties of matrix computation. The following code is written in C, where elements within the same row are stored contiguously. Assume each word is a 64-bit integer.

```
for (I=0; I<8;I++)
   for(J=0;J<8000;J++)
      A[I][J]=B[I][0]+A[J][I];
```

   1) __2____(the number) 64-bit integers can be stored in a 16-byte cache block.
   2) (ABC) exhibit temporal locality(multiple choices).
      A. I B. J C. B[I][0] D.A[J][I] E. A[I][J]
   3) (A) exhibit spatial locality(multiple choices).
      A. I B. J C. B[I][0] D.A[J][I] E. A[I][J]

2. 书后 5.2 Caches are important to providing a high-performance memory hierarchy to processors. Below is a list of 64-bit memory address references, given as word addresses. 0x03, 0xb4, 0x2b, 0x02, 0xbf, 0x58, 0xbe, 0x0e, 0xb5, 0x2c, 0xba, 0xfd

   1) For each of these references, identify the binary word address, the tag, and the index given a direct-mapped cache with 16 one-word blocks. Also list whether each reference is a hit or a miss, assuming the cache is initially empty.

Word Address	Binary address	Tag	Index	H/M

0x03	00000011	0	3	M
0xb4	10110100	b	4	M
0x2b	00101011	2	b	M
0x02	00000010	0	2	M
0xbf	10111111	b	f	M
0x58	01011000	5	8	M
0xbe	10111110	b	e	M
0x0e	00001110	0	e	M
0xb5	10110101	b	5	M
0x2c	00101010	2	c	M
0xba	10111010	b	a	M
0xfd	11111101	f	d	M

2) For each of these references, identify the binary word address, the tag, the index, and the offset given a direct-mapped cache with two-word blocks and a total size of eight blocks. Also list if each reference is a hit or a miss, assuming the cache is initially empty.

Word Address	Binary address	Tag	Index	H/M
0x03	00000011	0	1	M
0xb4	10110100	b	2	M
0x2b	00101011	2	5	M
0x02	00000010	0	1	H
0xbf	10111111	b	7	M
0x58	01011000	5	4	M
0xbe	10111110	b	6	M
0x0e	00001110	0	7	M
0xb5	10110101	b	2	H
0x2c	00101010	2	6	M
0xba	10111010	b	5	M
0xfd	11111101	f	6	M

Word Address	Binary Address	Tag	Index	Hit/Miss
0x03	0000 0011	0	3	M
0xb4	1011 0100	b	4	M
0x2b	0010 1011	2	b	M
0x02	0000 0010	0	2	M
0xbf	1011 1111	b	f	M
0x58	0101 1000	5	8	M
0xbe	1011 1110	b	e	M
0x0e	0000 1110	0	e	M
0xb5	1011 0101	b	5	M
0x2c	0010 1100	2	c	M
0xba	1011 1010	b	a	M
0xfd	1111 1101	f	d	M

2)

Word Address	Binary Address	Tag	Index	Offset	Hit/Miss
0x03	0000 0011	0	1	1	M
0xb4	1011 0100	b	2	0	M
0x2b	0010 1011	2	5	1	M
0x02	0000 0010	0	1	0	H
0xbf	1011 1111	b	7	1	M
0x58	0101 1000	5	4	0	M
0xbe	1011 1110	b	6	0	H
0x0e	0000 1110	0	7	0	M
0xb5	1011 0101	b	2	1	H
0x2c	0010 1100	2	6	0	M
0xba	1011 1010	b	5	0	M
0xfd	1111 1101	f	6	1	M

3. 书后 5.5 For a direct-mapped cache design with a 64-bit address, the following bits of the address are used to access the cache.

Tag	Index	Offset
63–10	9–5	4–0

Beginning from power on, the following byte-addressed cache references are recorded.

Address											
Hex	00	04	10	84	E8	A0	400	1E	8C	C1C	B4	884
Dec	0	4	16	132	232	160	1024	30	140	3100	180	2180

1) The cache block size (in words) is __4___?
2) The cache have __32_____ blocks?
3) The ratio between total bits required for such a cache implementation over the data storage bits is ___1.21_____.

4. Cache system

(1) Consider a 32 byte direct-mapped write-through & write around cache with 8 byte blocks. Complete the table below for sequence of memory references (occurring from left to right).

Address	00	16	48	08	56	16	08	56	32	00	60
Read/write	r	r	r	r	r	r	r	w	w	r	r
Line#	0	2	1	1	2	2	1	2	2	0	0
Tag	0	0	2	0	2	0	0	2	1	0	3
Hit/miss	M	M	M	M	M	H	H	H	M	H	M

(2) Consider a 128byte 2-way set associative write-back with 16 byte blocks, assuming LRU replacement. Complete the table below for a sequence of memory references (occurring from left to right)

Address	064	032	064	000	112	064	128	048	240	000

Read/write	r	r	r	r	w	w	r	r	r	w
Set #	2	3	2	0	1	2	2	0	0	0
Tag	1	0	1	0	4	1	4	1	9	0
Hit/miss	M	M	H	M	M	H	M	M	M	M
Dirty-set						2			0	0

Please write down the dirty block number 4 , 0 , 1 .

5. **Consider a virtual memory system with the following properties:**

40-bit virtual byte address, 16KB pages, 36-bit physical byte address

(1). What is the total size of the page table for each process on this processor, assuming that the valid, protection, dirty, and use bits take a total of 4 bits and that all the virtual pages are in use? (assuming that disk addresses are not stored in the page table).

2^40/2^16*40=5*2^29

(2). Assume that the virtual memory system has implemented a TLB with a total of 64 TLB entries. Could the TLB hold a program if the program accessed at least 8 MB of memory at a time? If you want the TLB to hold the program, how large the page size should be?

8MB/16KB=512>64,so it can not hold a program.
8MB/64=128KB,the page size should be 128KB