dce

COMPUTER ARCHITECTURE
REVIEW

Tran Ngoc Thinh
HCMC University of Technology
http://www.cse.hcmut.edu.vn/~tnthinh

dce

Performance

1. **You have a computer with a program execution time of 60% of the multiplication instructions, 30% of the division instructions. Suppose you can improve the multiplication instructions up to 6 times, division instructions up to 12 times. To increase your computer's performance by 2 times with only one component improvement, what component is selected?**
   a. multiplication instructions
   b. division instructions
   c. Both a & b are correct
   d. Both a & b are wrong.

- **Speedup_nhan = 1/((1-F)+F/S) = 1/((1-0.6)+0.6/6) = 2**
- **Speedup_chia = 1/((1-0.3) +0.3/12) = 1.38**

2. If you improve both multiplication and division in the above question, the performance of the new computer will increase:
1. 4.000 times
2. 4.175 times
3. 4.444 times
4. None of the above

- Speedup = 1/((1-0.6-0.3)+0.6/6+0.3/12) = 4.444

Performance

3. A program with 4 million instructions executes on a computer with a CPU clock rate of 2.5 GHz. The program run time is 8 milliseconds. The average CPI of this program is

a. 5
b. 10
c. 20
d. None of the above

- **T = I x CPI x C => CPI = $(8*10^{-3} *2.5*10^{9}) /(4*10^{6}) = 5$**

4. To improve the runtime of the program in the question above, a new compiler with 5 million instructions and a new CPI of 2, CPU clock rate does not change. Speedup is

a. 1.5
b. 4
c. 2
d. None of the above

- **Tnew = $(5*10^{6} *2) /(2.5*10^{9}) = 4*10^{-3}$ => Speedup = Told/Tnew = 8/4 =2**

MIPS

1. **In MIPS instruction set, we can use a following technique to substitute NOT A instruction.**
   a. A NAND 0
   b. A XOR 0
   c. A NOR 0
   d. None of the above

2. **Microprocessor performance can be measured by**
   a. MIPS
   b. Throughput
   c. MFLOP
   d. All the above

3. **MIPS architecture is a type of**
   a. Stack
   b. Accumulator
   c. Register /register
   d. Register/ memory

4. Given the following MIPS sequence, determine the value of *$t1*

```
• lui     $t1, 0x1234
• addi    $t1, $t1, 0x5678
```

a. 0x00000000
b. 0x1234
c. 0x5678
d. 0x12345678

5. x86 architecture is:

a. Big-Endian
b. Most-significant byte at highest address of a word
c. Most-significant byte at lowest address of a word
d. None of the above

dce

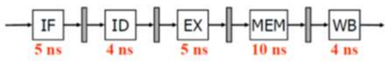
Pipelining

1. **Choose a FALSE statement about techniques to reduce hazards using compilers.**
   a. Forwarding
   b. Scheduling
   c. Branch prediction
   d. Branch delay slots
2.

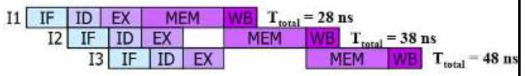
   Which data dependence is not a data hazard?
   a. Read-after-read (RAR)
   b. Read-after-write (RAW)
   c. Write-after-read (WAR)
   d. Write-after-write (WAW)
3.

   Which is the required time of 100 tasks in an ideal pipeline with 6 stages and an execution time of 2ns per stage?
   a. 12ns
   b. 200ns
   c. 210ns
   d. 212ns

The figure of 5-stage pipelining is used for 3 following questions



4. **The total execution time of 3 consecutive pipelining instructions is?**
   a. 38ns
   b. 43ns
   c. 48ns
   d. 53ns



5. **The time of STALL in the 4th instruction is?**
   a. 5ns
   b. 10ns
   c. 15ns
   d. None of the above

dce

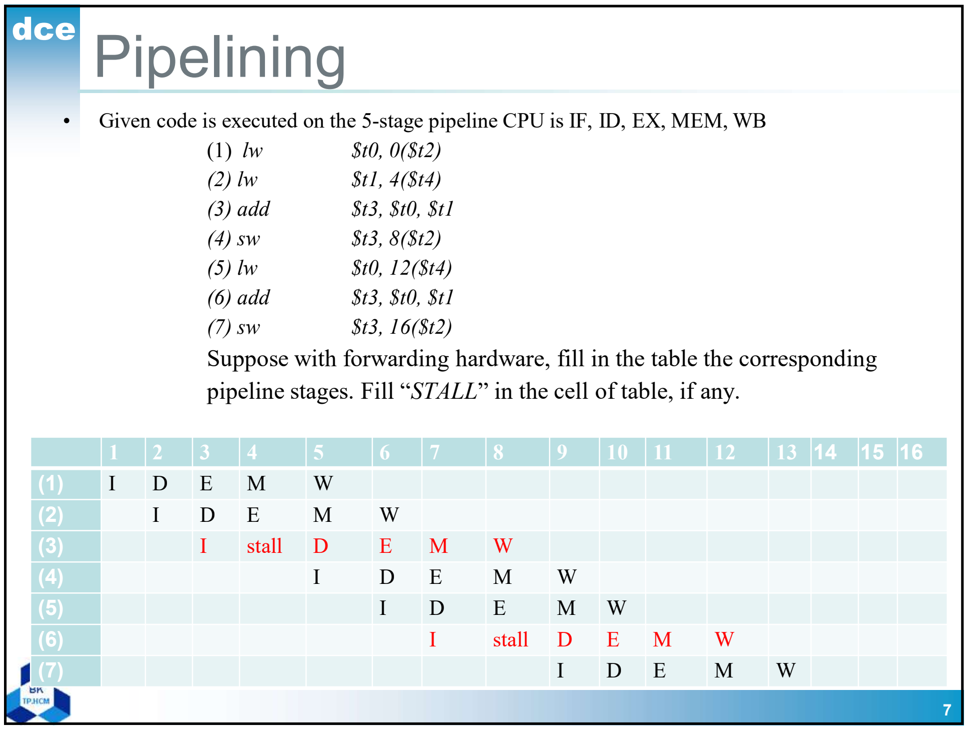
Pipelining

- Given code is executed on the 5-stage pipeline CPU is IF, ID, EX, MEM, WB

  (1) *lw* *$t0, 0($t2)*
  (2) lw *$t1, 4($t4)*
  (3) add *$t3, $t0, $t1*
  (4) sw *$t3, 8($t2)*
  (5) lw *$t0, 12($t4)*
  (6) add *$t3, $t0, $t1*
  (7) sw *$t3, 16($t2)*

  Suppose that without forwarding hardware, fill in the table the corresponding pipeline stages. Fill "*STALL*" in the cell of table, if any.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
(1)	I	D	E	M	W																
(2)		I	D	E	M	W															
(3)			I	stall	stall	D	E	M	W												
(4)						I	stall	stall	D	E	M	W									
(5)									I	D	E	M	W								
(6)										I	stall	stall	D	E	M	W					
(7)													I			D	E	M	W		



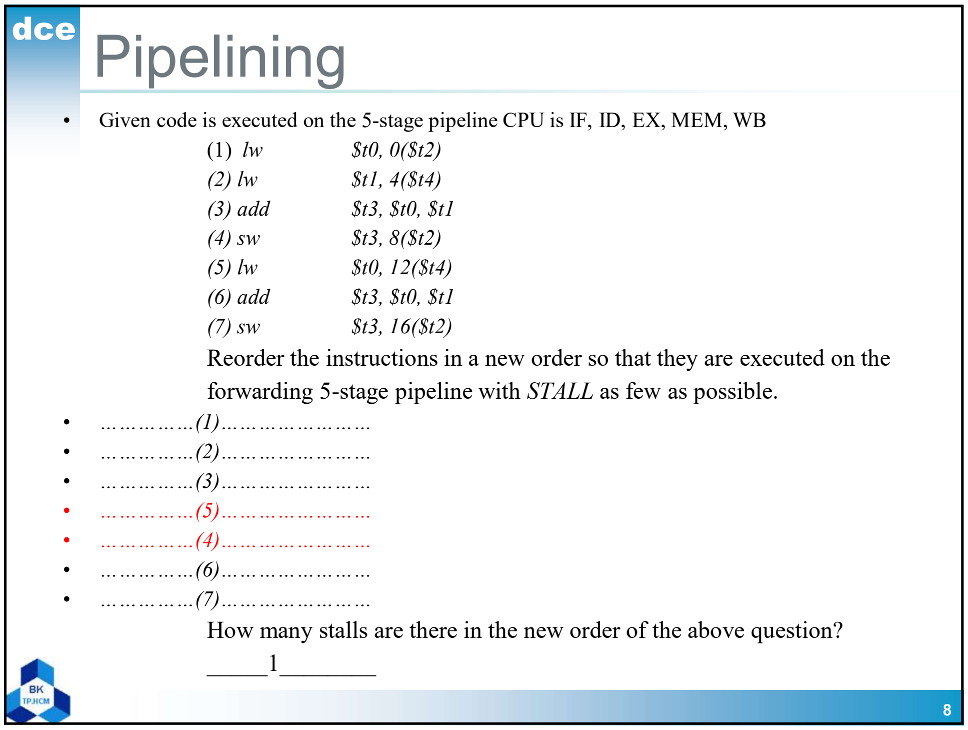
Pipeline

- Given code is executed on the 5-stage pipeline CPU is IF, ID, EX, MEM, WB

(1) *lw*	*$t0, 0($t2)*
(2) lw	*$t1, 4($t4)*
(3) add	*$t3, $t0, $t1*
(4) sw	*$t3, 8($t2)*
(5) lw	*$t0, 12($t4)*
(6) add	*$t3, $t0, $t1*
(7) sw	*$t3, 16($t2)*

Suppose with forwarding hardware, fill in the table the corresponding pipeline stages. Fill "*STALL*" in the cell of table, if any.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
(1)	I	D	E	M	W											
(2)		I	D	E	M	W										
(3)			I	stall	D	E	M	W								
(4)					I	D	E	M	W							
(5)						I	D	E	M	W						
(6)							I	stall	D	E	M	W				
(7)									I	D	E	M	W			



Pipeline

- Given code is executed on the 5-stage pipeline CPU is IF, ID, EX, MEM, WB

(1) *lw*	*$t0, 0($t2)*
(2) lw	*$t1, 4($t4)*
(3) add	*$t3, $t0, $t1*
(4) sw	*$t3, 8($t2)*
(5) lw	*$t0, 12($t4)*
(6) add	*$t3, $t0, $t1*
(7) sw	*$t3, 16($t2)*

Reorder the instructions in a new order so that they are executed on the forwarding 5-stage pipeline with *STALL* as few as possible.

- ………………(1)………………………
- ………………(2)………………………
- ………………(3)………………………
- ………………(5)………………………
- ………………(4)………………………
- ………………(6)………………………
- ………………(7)………………………

How many stalls are there in the new order of the above question?

______1_________

dce

Cache

1. The algorithm to remove and place new contents into the cache is called

- ***a. Replacement algorithm***
- ***b. Renewal algorithm***
- ***c. Updation***
- ***d. None of the above***

2. L2 cache (level 2) helps

- ***a. Reduces miss-rate of L1 cache***
- ***b. Reduce hit-time of L1 cache***
- ***c. Increase L1 cache size to 2x***
- ***d. Reduce miss-penalty of L1 cache***

dce

Cache

- ***Give the following data for the next 4 questions***
- A CPU equipped with an 8KB data cache implemented in a 2-way associative with a cache line of 16 bytes. Know that the CPU uses 32-bit addresses to access the cache

8. The bit number of Offset field is

a. 2 b. 4 c. 8 d.16

- **a cache line of 16 bytes => Offset b=4 bit**

9. The bit number of Index field is

a. 2 b. 4 c. 8 d.16

- **8KB data cache => 8KB/16 = 512 block frames, 2-way => 512/2 = 256 sets =>Index k=8 bit**

10. The bit number of Tag field is

a. 16 b. 18 c. 20 d. 22

- **Tag field t= N-k-b = 32-8-4 = 20**

11. After reset, how many times the cache miss to access the following 5 addresses

- *0x00000000, 0x00000014, 0x00000008, 0x00000010, 0x00000020*

a. 2
b. 3
c. 4
d. 5

dce

VM

1. **In virtual memory**
   a. A program may be larger than physical memory
   b. The same address in 2 different programs can be different
   c. The elements A[n] and A[n + 1] in the array have not to lie side by side in the physical memory
   d. All the above
2. **The binary address issued to data or instructions are called as**
   a. Physical address
   b. Location
   c. Relocatable address
   d. Logical address
3. **The page table is for storing**
   a. Data of the physical page
   b. Virtual page data
   c. Physical page address
   d. Virtual page address
4. **Read data at an address in a paging system including**
   a. 1 physical address lookup and 1 physical data reading
   b. 1 virtual address lookup and 1 virtual data reading
   c. 1 virtual address lookup and 1 physical data reading
   d. 1 physical data reading, if miss 1 more virtual data reading

dce

VM

5. **Page-fault exception is**
   a) Exception indicates cache-miss
   b) Exception indicates page-table miss
   c) Page size is larger than standard size
   d) Physical page and virtual page have difference sizes
6. **TLB is used to**
   a) Accelerate physical data access
   b) Accelerate hard disk access
   c) Accelerate the translation from virtual address o the physical address
   d) All the above
7. **When TLB miss occurs then**
   a) Page-fault occurs
   b) No page-fault occurs
   c) The system is reset
   d) Page-fault can occur or cannot occur
8. **In FIFO page replacement algorithm, when a page must be replaced**
   a) Newest page is chosen
   b) Oldest page is chosen
   c) Random page is chosen
   d) None of the mentioned
9. **The algorithm which replaces the block which has not been referenced for a while is called**
   a) LRU
   b) ORF
   c) Direct
   d) Both LRU and ORF