Hacettepe University
Computer Engineering Department
BBM234 Computer Organization

Homework 6

Assigned date : 08.05.2017
Due date : 16.05.2017 (All Sections)

You should hand in your homework in class.
Don't email your homework! Stamp your homework papers. Do not put them in the plastic bags.
LATE HOMEWORKS WILL NOT BE ACCEPTED…

Questions: (Each one is 20 points.)

Q1. You are given the following MIPS code. You have 5 stage pipelined MIPS processor running the code.

a) If there is no hazard unit in the MIPS processor, **insert enough NOPs between the instructions** to have correct execution of the code.
   i. How many cycles does it take to execute all instructions if the branch is not taken?
   ii. How many cycles does it take to execute all instructions if the branch is taken? Show your calculations or explain how you found the cycle time.

```
# MIPS assembly code
      lw  $s0, 0($0)

      lw  $s1, 4($0)
      2 NOPS
      beq $s0, $s1, L
      3 NOPS
      add $t0, $t0, $s0

      add $t1, $t1, $s1
      2 NOPS
      add $t2, $t0, $t1
      2 NOPS
L:    addi $t2, $t2, 1
```

If branch is not taken, after beq, all instructions execute. We have total of 16 instructions including NOPs. Then, we fetch 16 instrcutions in 16 cycles. Last instruction needs 4 more cycles to complete. Then, it takes 20 cycles to execute all instrcutions if branch is not taken.

If branch is taken, after beq, it takes 3 cycles to determine the next address of the branch. We have 4 instructions before beq, beq itself, 3 instcrutions after beq and the last instruction where branch is (L). Then, there will be 9 instructions that will execute. The total cycles will be 13.

b) If there is data hazard unit (data forwarding, stalling, and early branch resolution hardware) in the MIPS processor, insert enough NOPs between the instructions to have correct execution of the code.
   i. How many cycles does it take to execute all instructions if the branch is not taken?
   ii. How many cycles does it take to execute all instructions if the branch is taken? Show your calculations or explain how you found the cycle time.

 # MIPS assembly code

```
    lw  $s0, 0($0)

    lw  $s1, 4($0)
    1 NOP
    beq $s0, $s1, L

    add $t0, $t0, $s0

    add $t1, $t1, $s1

    add $t2, $t0, $t1

L:  addi $t2, $t2, 1
```

If branch is not taken, after beq, all instructions execute. We have total of 8 instructions including NOPs. Then, we fetch 8 instrcutions in 8 cycles. Last instruction needs 4 more cycles to complete. Then, it takes 12 cycles to execute all instrcutions if branch is not taken.

If branch is taken, after beq, it takes 1 cycles to determine the next address of the branch since we have early branch resolution hardware. We have 3 instructions before beq, beq itself, 1 instcrution after beq and the last instruction where branch is (L). Then, there will be 6 instructions that will execute. The total cycles will be 10.

Q2. Bir işlemci tasarlanırken işlemcinin bloklarının çalışma süreleri aşağıdaki gibi hesaplanmıştır. Pipeline register çalışma süresi ise 1ns olarak bulunmuştur. (ns=10^{-9})

IF	ID	EX	MEM	WB
7ns	7ns	6ns	9ns	5ns

a) Single cycle bir işlemci tasarlarsak, işlemcinin clock süresi ne olur? [3]

Clock=7+7+6+9+5=34ns

b) 5 stage pipeline işlemci tasarlarsak, işlemcinin clock süresi ne olur? [4]

En uzun stage MEM aşamasıdır. Pipeline register süresini de eklersek:

Clock = 9+1=10ns

c) Pipeline işlemcinin clock süresini azaltmak istiyoruz. Bunun için stagelerden birini iki aşamaya bölme kararı aldık ve işlemcinin pipeline stage sayısını 6 stage yapacağız. Hangi aşamayı bölersiniz? Böldükten sonra işlemcinin clock süresi ne olur? [4]

MEM aşaması bölümürse clock süresi azalır. MEM aşaması bölünürse, en uzun stage 7ns olacağından:

Clock = 7+1=8ns

d) Yukarıdaki üç işlemcide de aynı 1000 tane komutu çalıştırmak istiyoruz. Programda hiçbir data ve control hazard olmadığını düşünün. Bu üç işlemcide bu program kaç saniye çalışır? [9]

CPU Time Single Cycle = $1000x1x34.10^{-9}=34.10^{-6}$sn

CPU Time five stage = $1000x1004/1000x10.10^{-9}= 10{,}04.10^{-6}$sn

CPU Time five stage = $1000x1005/1000x8.10^{-9}= 8{,}04.10^{-6}$sn

Q3. A cache has the following parameters: *b*, block size given in numbers of words; *S*, number of sets; *N*, number of ways; and *A=32*, number of address bits.

You are given the following MIPS code with the lw addresses given in hexadecimal:

```
# MIPS assembly code
        addi $t0, $0, 5
loop:   beq  $t0, $0, done
        lw   $t1, 0x10($0)
        lw   $t2, 0x14($0)
        lw   $t3, 0xF0($0)
        lw   $t3, 0x1C($0)
        addi $t0, $t0, -1
        j    loop
done:
```

Suppose C=8 words and b=1 word. Assume the cache is initially empty. Answer the following questions.

a) If N=1 (direct mapped), what is the miss rate? Explain which memory accesses cause misses.

b) If N=2, what is the miss rate? Explain which memory accesses cause misses.

c) If N=1 (direct mapped) and b=4, what is the miss rate? Explain which memory accesses cause miss.

For all questions, Show where the addresses 10, 0C, F0, and 48 are mapped on the cache on the figures.

---

a)

Data	
1C	Set 7 (111)
	Set 6 (110)
14	Set 5 (101)
10,F0	Set 4 (100)
	Set 3 (011)
	Set 2 (010)
	Set 1 (001)
	Set 0 (000)

In the first loop, all memory accesses cause miss (4 misses – compulsory). In each loop, 10 and F0 causes conflict misses (2 misses).
Loop iterates 5 times, in each loop there are 4 memory accesses.
Total memory accesses = 5x4=20
First loop: 4 misses
Second, third, fourth, and fifth loops:2 misses (total of 8)

Miss rate =12/20

0x10 --> 000**10**000 (mapped to set 100, the last two bits are for byte offset)
0x14 --> 000**10**100 (mapped to set 101)
0xF0 --> 111**10**000 (mapped to set 100)
0x1C --> 000**11**100 (mapped to set 111)

---

b)

Way 1	Way 0	
Data	Data	
	1C	Set 3 (11)
		Set 2 (10)
	14	Set 1 (01)
F0	10	Set 0 (00)

There are only 4 compulsory misses in the first loop.

Miss rate = 4/20

0x10 --> 0001**00**00 (mapped to set 00, the last two bits are for byte offset)
0x14 --> 0001**01**00 (mapped to set 01)
0xF0 --> 1111**00**00 (mapped to set 00)
0x1C --> 0100**11**00 (mapped to set 10)

---

c)

Data	Data	Data	Data	
1C, FC	18, F8	14, F4	10, F0	Set 1 (1)
				Set 0 (0)

0x10 --> 000**1**0000 (mapped to set 1, block 00. The last two bits are for byte offset, the next two bits are block offset). 10, 14, 18, 1C are mapped to set 1 in the first memory access.
0x14 --> 000**1**0100 (mapped to set 1, block 01)
0xF0 --> 111**1**0000 (mapped to set 1, block 00). F0, F4, F8, FC are mapped when memory is accessed for F0.
0x1C --> 000**1**1100 (mapped to set 1, block 11)

In the first loop, 10 is miss (compulsory). 14 is hit. F0 is miss (conflict). 1C is miss (conflict). (3 misses)
In the second loop, 10 and 14 are hit. F0 is miss (conflict). 1C is miss (conflict). (2 misses)
In the third loop, 10 and 14 are hit. F0 is miss (conflict). 1C is miss (conflict). (2 misses)
In the forth loop, 10 and 14 are hit. F0 is miss (conflict). 1C is miss (conflict). (2 misses)
In the fifth loop, 10 and 14 are hit. F0 is miss (conflict). 1C is miss (conflict). (2 misses)

So, **Miss rate=11/20.**

Q4.
Aşağıdaki parametrelere sahip bir cache düşünün:
Fiziksel adres boyutu: 32 bit
Assosiativity (N): 2
Word size: 8 Byte
Block size: 2 words
Cache size (C): 128 Byte

a) Bu cache bellekte kaç tane set bulunur? Hesaplayın. [5]

Bellekteki bir bloğun boyutu (b) = 8*2=16 Byte
Cache bellekteki blok sayısı (B) = 128 / 16 = 8

Set sayısı (S) = B/N=8/2=4

b) Fiziksel bir adreste hangi bitler ne için kullanılır, aşağıdaki bitler üzerinde gösterin. [10]

Tag 26 bits	Set 2 bits	Blok offset 1 bit	Byte offset 3 bits
31-6	5-4	3	2-0

c) Bir cache satırında, V ve U bitleri dahil olmak üzere toplam kaç bit bulunur? Cache'in bir satırını blok halinde çizip bit sayılarını yazın ve toplamı hesaplayın. [5]

Way 1				**Way 0**			
V	**U**	**TAG**	**DATA**	**V**	**TAG**	**DATA**	**Toplam**
1 bit	**1 bit**	**26 bit**	**128 bit**	**1 bit**	**26 bit**	**128 bit**	**311 bits**

Q5. Aşağıda main memory'de tutulan page table'ın ilk 8 satırı verilmiştir. Page boyutu 4KB'tır. İşlemcinin TLB'ye erişim süresi 1ns, main memory erişim süresi 100ns olsun. Mimaride cache bulunmadığını farzedin.

V	Physical Page Number (PPN)	Virtual Page Number (VPN)
..	..	..
0	0x0002	7
0		6
1	0x0001	5
0		4
0		3
1	0x7FFF	2
0		1
0	0x0005	0

İşlemci aşağıdaki virtual adreslere sırasıyla erişmek istiyor.

0x50F0, 0x2FF0, 0x5FF4, 0x2020

a) Yukarıdaki virtual adreslerin fiziksel adresleri nedir? [6]

Virtual page address	Physical page address
0x50F0	0x10F0
0x2FF0	0x7FFFFF0
0x5FF4	0x1FF4
0x2020	0x7FFF020

b) Eğer TLB kullanılmıyorsa, bu adreslerdeki veriler işlemciye toplam ne kadar sürede ulaşır? Hesaplayın. [7]

Her address translation için main memory'e erişilir. Ayrıca veriye ulaşmak için yine Main memory'e erişilir. Toplamda 8 main memory erişimi vardır. Toplam erişim süresi 8x100ns=800ns olur.

c) Eğer iki değer tutabilen bir TLB varsa ve başlangıç durumu aşağıdaki gibiyse, bu adreslerdeki veriler işlemciye toplam ne kadar sürede ulaşır? Hesaplayın. [7]

Entry 1			Entry 0		
V	VPN	PPN	V	VPN	PPN
0			1	0x0002	0x7FFF

0x50F0 için TLB'ye bakılır. TLB'de address translation için veri bulunamadığından main memorye bir kez adres translation bir kez veriyi getirmek için erişilir. Ayrıca bu adresin değeri TLB'ye yazılır.
0x2FF0 TLB'de olduğundan bir kez TLB'ye bir kez main memorye erişilir. 0x5FF4 ve 0x2020 adresleri de TLB'den translate edildiğinden her ikisi için de bir kez TLB bir kez main memory erişimi olur.
Toplamda 5 main memory ve 4 TLB erişimi yapılır. Toplam erişim süresi 5x100ns+4*1ns=504ns olur.