Computer Architecture

Fall, 2019

Week 3

2019.9.23

組員簽名：_____

[group7] (對抗賽)

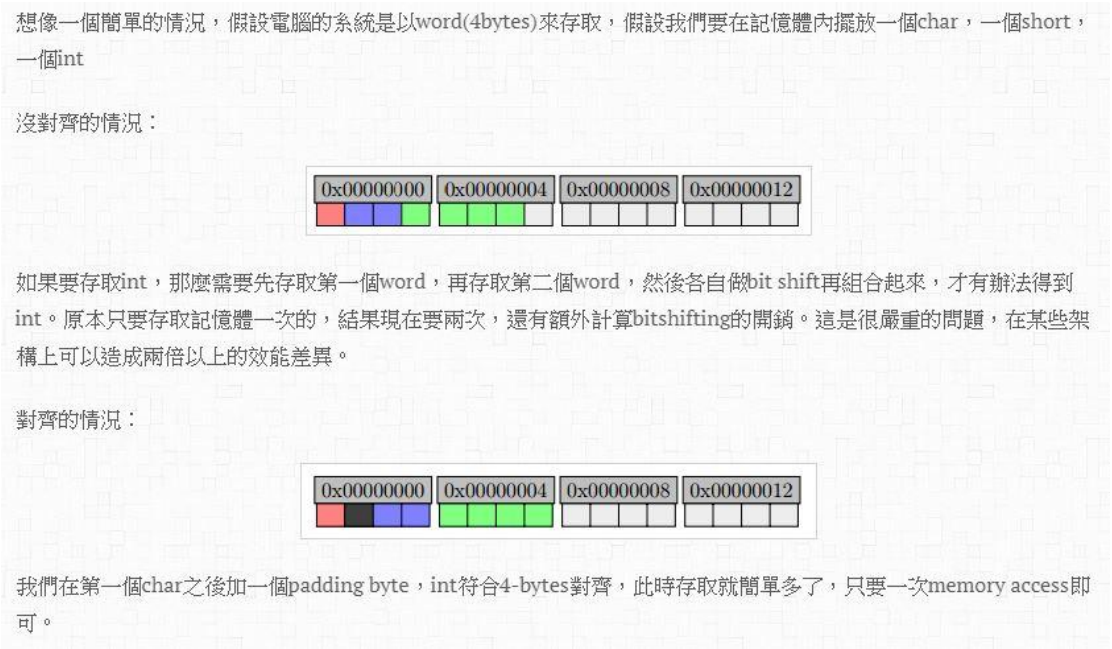
1. (a)為什麼需要存取資料時需要對齊(Alignment)記憶體?
   (b)如果記憶體不對齊，那存取資料時可能會發生什麼情況?舉例說明。

A：

(a)進行存取會最有效率。

word 是存取記憶體最自然的單位。word 的大小是電腦架構所定義，一般現代的系統通常 word 不是 4bytes(32bit)就是 8bytes(64bit)。早期的電腦記憶體只能以 word 為單位進行存取，因此所有的記憶體存取都要落在以 word 為倍數的邊界上。但現代的電腦架構通常能夠存取小至單獨的 byte，大至多個 word。

(b)



[group5] (對抗賽)

2. There is no instruction move (move the data from one register to another register), but we still can move the data from one register to another, how?

A2:by using add (ex:add $t1 $s1 $zero)

[group11] (對抗賽)

3. What is the difference between register and memory? Consider from 3 distinct perspective, including access frequency, size, access speed.

Ans:

1. Register holds the data that the CPU is currently processing, whereas the memory holds the data which will be required for processing
2. The register ranges from 32-bits register to 64-bits register, whereas the memory capacity ranges from some GB to some TB
3. The processor accesses register faster than the memory

[group9] (對抗賽)

4. Which statement below is true for MIPS?
   (A) Assembly use RAM as a place for temporary storage.
   (B) The data stored in a 32 bits wide register can also be called as a word.
   (C) ADD $0, 1, 1 is a valid assembly code.
   (D) Normally each register is 32bits, while Floating Point register is 64bits.

ANS.

(A) Assembly use **register file** as a place for temporary storage.
(B) true
(C) ADD need **three register operands**
(D) Floating Point register is **32bits**.

[group10] (對抗賽)

5. 下列關於 MIPS 敘述哪些符合 design principle : **Simplicity favors regularity?**
   (A) 所有 instructions 皆為 32 bits
   (B) Basic MIPS arithmetic/logic instructions 中為 1 operator, 3 operands
   (C) 在每種 instruction 格式中，暫存器欄位(register fields)固定在相同地方
   (D) 在每種 instruction 格式中，指令欄位(opcode fields)固定在相同地方

ANS : ABCD

[group2] (對抗賽)

6.

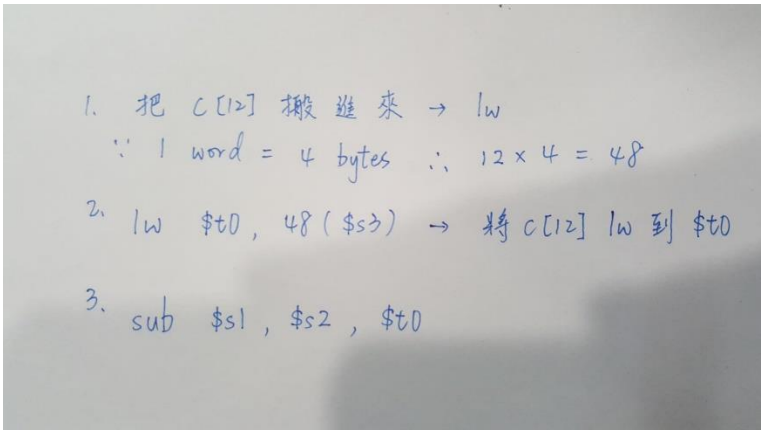
$s1:a , $s2:b , $s3:base address of C

a = b - C[12]

Q1:Please write down the assembly code.

Q2:Please explain your assembly code.

答案 :



[group12] (對抗賽)

7. The following is the part of MIPS code, after executing it, what is the result of $t3? (Suppose a integer array A, A[i]=2*i, the base address of A is in $s3)

```
addi $t0, $0, 5
add $t0, $t0, $t0
add $t0, $t0, $t0  # or it could be sll $t0, $t0, 2
add $t1, $s3, $t0
lw $t2, 0($t1)
add $t2, $t2, $t2
add $t2, $t2, $t2
add $t1, $s3, $t2
lw $t3, 0($t1)
```

A: 20

[group4]

8. For the following MIPS assembly instructions above, what is a corresponding C statement?

```
add f, g, h
add f, i, f
```

ANS: f = i + (g+ h)