國　立　清　華　大　學　命　題　紙

95 學年度 電機領域聯合招生 系（所） 組碩士班入學考試

科目 計算機系統 科目代碼 9911 共 3 頁第 1 頁 *請在【答案卷卡】內作答

(1) (a) (5%) Can you come up with a MIPS instruction that behaves like a NOP? The instruction is executed by the pipeline but does not change any state.

(b) (5%) In a MIPS computer, a main program can use "*jal procedureaddress*" to make a procedure call and the callee can use "*jr $ra*" to return to the main program. What is saved in register *$ra* during this process?

(c) (5%) Name and explain the three principle components that can be combined to yield runtime.

(2) (a) (5%) Briefly explain the purpose of having a write buffer in the design of a write-through cache.

(b) (5%) Large cache block tends to decrease cache miss rate due to better spatial locality. However, it has been observed that too large a cache block actually increases miss rate, especially in a very small cache. Why?

(3) (a) (5%) Dynamic branch prediction is often used in today's machine. Consider a loop branch that branches nine times in a row, and then is not taken once. What is the prediction accuracy for this branch, assuming a simple 1-bit prediction scheme is used and the prediction bit for this branch remains in the prediction buffer? Briefly explain your result.

(b) (5%) What is the prediction accuracy if a 2-bit prediction scheme is used? Again briefly explain your result.

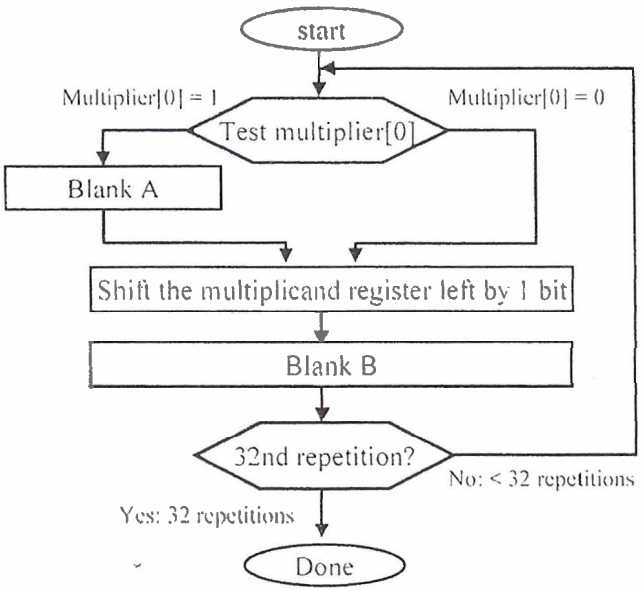
(4) (15%) Answer the following questions briefly.

(a) (5%) In a pipelined CPU design, what kind of problem may occur as it executes instructions corresponding to an *if-statement* in a C program? Name one possible scheme to get around this problem more or less.

(b) (5%) Consider the possible actions in the *Instruction Decode* stage of a pipelined CPU. In addition to setting up the two input operands of ALU, what is the other possible action? (Hint: consider the execution of a jump instruction)

(c) (5%) What is x if the maximum number of memory words you can use in a 32-bit MIPS machine in a single program is expressed as 2^x? (Note: MIPS uses a byte addressing scheme.)

(5) (10%) Consider the following flow chart of a sequential multiplier. We assume that the 64-bit multiplicand register is initialized with the 32-bit original multiplicand in the right half and 0 in the left half. The final result is to be placed in a *product* register. Fill in the missing descriptions in blanks A and B.



(6) (10%) Schedule the following instruction segment into a superscaler pipeline for MIPS. Assume that the pipeline can execute one ALU or branch instruction and one data transfer instruction concurrently. For the best, the instruction segment can be executed in four clock cycles. Fill in the instruction identifiers into the table. Note that data dependency should be taken into account.

(Identifier)		(Instruction)	
In-1	Loop:	lw	$t0, 0($s1)
In-2		addu	$t0, $t0, $s2
In-3		sw	$t0, 0($s1)
In-4		addi	$s1, $s1, -4
In-5		bne	$s1, $zero, Loop

Clock Cycle	ALU or branch instruction	Data transfer instruction
1		
2		
3		
4		

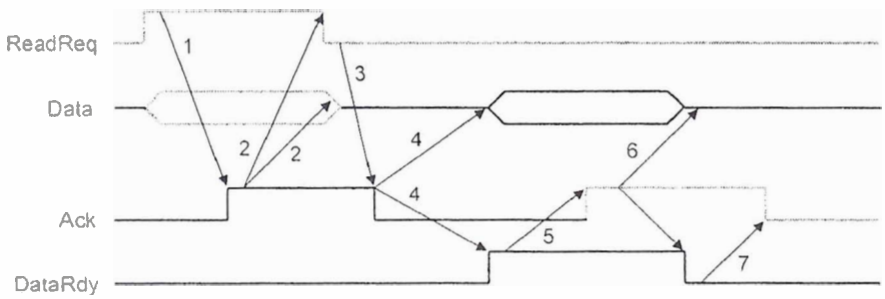
(7) (10%) Suppose a computer's address size is k bits (using byte addressing), the cache size is S bytes, the block size is B bytes, and the cache is A-way set-associative. Assume that B is a power of two, so $B = 2^b$. Figure out what the following quantities are in terms of S, B, A, b and k:

(a) (3%) the number of sets in the cache

(b) (3%) the number of index bits in the address

(c) (4%) the number of bits needed to implement the cache

(8) (8%) To compare the maximum bandwidth for a synchronous and an asynchronous bus, assume that the synchronous bus has a clock cycle of 50 ns, and each bus transmission takes 1 clock cycle. The asynchronous bus requires 40 ns per handshake and the asynchronous handshaking protocol consists of seven steps to read a word from memory and receive it in an I/O device as shown below. The data portion of both buses is 32 bits wide. Find the bandwidth for each bus in MB/sec when performing one-word reads from a 200-ns memory.



(9) (12%) Bus arbitration is needed in deciding which bus master gets to use the bus next in a computer system. There are a wide variety of schemes for bus arbitration; these may involve special hardware or extremely sophisticated bus protocols. In a bus arbitration scheme, a device (or the processor) wanting to use the bus signals a bus request and is later granted the bus. After a grant, the device can use the bus, later signaling to the arbiter that the bus is no longer required. The arbiter can then grant the bus to another device. Most multiple-master buses have a set of bus lines for performing bus requests and grants. A bus release line is also needed if each device does not have its own request line. Sometimes the signals used for bus arbitration have physically separate lines, while in other systems the data lines of the bus are used for this function. Arbitration schemes usually try to balance two factors in choosing which device to grant the bus, namely, the priority and the fairness. In general, bus arbitration schemes can be divided into four broad classes. What are those four classes? Briefly explain those four classes of bus arbitration schemes.