期末考题 （new）

**1.** [5points] Prior to the early 1980s, machines were built with more and more complex instructionset. The MIPS is a RISC machine. Why has there been a move to RISC machines away from complex instruction machines?

There are number of reasons for the move towards RISC machines away from CISC. Some of them are:

* + Since early computers had limited memory capacities and were expensive, having a CISC instruction set enabled performing complex operations with very few instructions (encoded within a smaller memory size compared to a corresponding RISC program). Since then memories have got cheaper and there has been a lot of advances in the design of cache hierarchies (for example, a dedicated on-chip instruction cache, prefetching techniques, etc.) that permit RISC machines work-around longer instruction sequences
  + Writing a compiler to generate efficient code is easier for a RISC architecture than for a CISC architecture as the compiler can take advantage of a lot of registers provided by the RISC architecture than a CISC
  + RISC instructions are easier to pipeline than CISC instructions

2.[12 points] Perform the following operations by converting the operands to 2’s complement binary numbers and then doing the addition or subtraction shown. Please show all work in binary, operating on 16-bit numbers.

(a) 35 + 12

0000 0000 0010 0011 (35)

* 0000 0000 0000 1100 (12)

0000 0000 0010 1111

The decimal value of the result is 47

(b) 13 – 8

0000 0000 0000 1101 (13)

* 1111 1111 1111 1000 (-8)

0000 0000 0000 0101

The decimal value of the result is 5 (we ignored the carry here)

(c) 15– 16

0000 0000 0000 1111 (5)

* 1111 1111 1111 0000 (-6)

1111 1111 1111 1111

The decimal value of the result is -1

3.[5points]Assuming single precision IEEE 754 format, what decimal number is represent by this word:

1 01111100 00100000000000000000000

(Hint: remember to use the biased form of the exponent.)

Answer:

The decimal number= (-1)1 \* (2^(124-127))\*(1.001) 2

= (-1)\*(0.125)\*(0.125)

=- 0.015625

4.[5 points] Show the IEEE 754 binary representation of the number 1.75 ten in single precision.

Answer: P189

5.[5points] Assume a color display using 8 bits for each of the primary colors (red, green, blue) per pixel and a frame size of 1440X900.

What is the minimum size in bytes of the frame buffer to store a frame?

Answer: 1440×900 pixels =1296000 pixels =>1296000×3= 3888000 bytes/frame.

6.[5points]Consider three different processors P1, P2, and P3 executing the same instruction set. P1 has a 4 GHz clock rate and a CPI of 2. P2 has a 3.0 GHz clock rate and has a CPI of 1.5.P3 has a 2.5 GHz clock rate and a CPI of 1.0.

Which processor has the highest performance expressed in instructions per second?

Answer:

performance of P1 (instructions/sec) = 4×10 9 /2=2 × 10 9

performance of P2(instructions/sec) = 3 × 10 9 /1.5 = 2× 10 9

performance of P3 (instructions/sec) = 2.5 × 10 9 /1.0 = 2.5 × 10 9

P3 has the highest performance expressed in instructions per second.

7.[10points]Assume one byte data value is 10101101 two . First show the Hamming ECC code

for that byte, and then invert bit 11and show that the ECC code finds and corrects the single bit error.

8.(1)（5分）简述原码不恢复余数法(加减交替法)的原理

答：

原码不恢复余数法(加减交替法)

计算机中串行除法是通过左移余数和加（减）除数来实现除法运算处理的。首先我们看一看原码除法运算规则。

原码不恢复余数法运算法则：

首先用被除数减去除数，然后按下列过程重复：

当余数为正时，商1，余数左移1位，减除数

当余数为负时，商0，余数左移1位，加除数

（2）（5分）设Ｘ＝0.1011,Y＝－0.1101，用原码不恢复余数法求Ｘ÷Ｙ。

9.[3 points] The memory architecture of a machine X is summarized in the following table.

|  |  |  |  |
| --- | --- | --- | --- |
| Virtual Address (bits) | Physical DRAM | Installed Page Size | PTE Size (byte) |
| 50 | 16GiB | 16 K bytes | 4 |

For a single-level page table, how many page table entries (PTEs) are needed? How much physical memory is needed for storing the page table?

Answer:The page table is indexed by the virtual page number which uses 50–14 = 36 bits. The number of entries are therefore 236 . Each PTE has 4 bytes. So the total size of the page table is

238 bytes.

8. [7 points] Virtual memory uses a page table to track the mapping of virtual addresses to physical addresses. Th e following data constitutes a stream of virtual addresses as seen on a system. Assume 4 KiB pages, a 4-entry fully associative TLB, and true LRU replacement. If pages must be brought in from disk, increment the next largest page number.

34588,13197,2228,48871

TLB

|  |  |  |
| --- | --- | --- |
| Valid | Tag | Physical Page Number |
| 1 | 11 | 12 |
| 1 | 7 | 4 |
| 1 | 3 | 6 |
| 0 | 4 | 9 |

Page table

|  |  |
| --- | --- |
| Valid | Physical Page or in Disk |
| 1 | 5 |
| 0 | Disk |
| 0 | Disk |
| 1 | 6 |
| 1 | 9 |
| 1 | 11 |
| 0 | Disk |
| 1 | 4 |
| 0 | Disk |
| 0 | Disk |
| 1 | 3 |
| 1 | 12 |

Given the address stream shown, and the initial TLB and page table states provided above, show the final state of the system. Also list for each reference if it is a hit in the TLB, a hit in the page table, or a page fault.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Address | Virtual Page | TLB H/M | TLB | | |
| Valid | Tag | Physical Page |
| 34588  (871CH) |  |  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| 13197  (338DH) |  |  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| 2228  (8B4H) |  |  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| 48871  (BEE7H) |  |  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

10. [5points]Given the following MIPS assembly code (and assuming all registers start at 0):

addi $1 $0 3

addi $2 $0 8

loop add $1 $1 $2

and $3 $1 $2

addi $2 $2 -2

bne $0 $2 loop

Describe the function of above program.

11.[5 points] Write the following sequence of code into MIPS assembler:

z=x+y-z+q;

Assume that x,y,z,q are stored in registers $s5-$s8.

Answer: The MIPS assembly sequence is as follows:

Add $t0,$s5,$s6

Sub $t1,$t0,$s7

Sub $s7,$t1,$s8

12. In general, cache access time is proportional to capacity. Assume that main memory accesses take 70 ns and that memory accesses are 36% of all instructions.

Th e following table shows data for L1 caches attached to each of two processors, P1 and

P2.

|  |  |  |  |
| --- | --- | --- | --- |
|  | L1 Size | L1 Miss Rate | L1 Hit Time |
| P1 | 2 KiB | 8.0% | 0.66 ns |
| P2 | 4 KiB | 6.0% | 0.90 ns |

5.6.1 [5] < § 5.4> Assuming that the L1 hit time determines the cycle times for P1 and

P2, what are their respective clock rates?

5.6.2 [5] < § 5.4> What is the Average Memory Access Time for P1 and P2?

5.6.3 [5] < § 5.4> Assuming a base CPI of 1.0 without any memory stalls, what is the

total CPI for P1 and P2? Which processor is faster?

For the next three problems, we will consider the addition of an L2 cache to P1 to