1b10, Computer Architecture

Answer any three questions., 2.5 hours.

Syntax errors in code will not lose any marks.

Question 1.

VLIW, Addressing Modes

a) A computer designer decides to design a new type of CPU using the idea of Very Long Instruction Words. Instead of the current trend towards regular, reduced instruction set sizes, the idea is to have a small number of very complex instructions that can carry out a lot of operations in one go.

The proposed instruction design has up to three operands which can be memory or register, and has two opcodes, which can be any combination of the logical or arithmetic instructions **and/or/add/sub/div/mul**, and a further operand which is the address of the next instruction to be fetched and executed.

    i) Assuming a 32bit byte addressable memory architecture, how many bits might the typical instruction take? **[8 Marks]**

    ii) What are the problems with this approach, and why is RISC preferred nowadays? **[10 Marks]**

b) The MIPS Processor has three addressing modes – explain, with simple examples, how these cover all the common cases needed. **[15 Marks]**

[Turn Over]

Question 2.

Memory, subroutines

a) The following code fragment is compiled to MIPS assembler.

Factorial(n) {
    If (n == 1)
        Return 1;
    Else
        Return n * factorial(n-1);
}

Write down what the assembler version of this, and describe and justify the subroutine linkage mechanisms that you have used. **[20 Marks]**

b) Modern Computers typically have a three level memory system comprising a cache, main memory, and virtual memory – Explain how a direct mapped cache is accessed, and how a page table is accessed. **[10 Marks]**

c) What are the properties of many programs that mean that caches make sense? **[3 Marks]**.

[Turn Over]

Question 3.

Pipelining

a) A RISC CPU Is designed with 8 instructions:
   lw, sw, add,sub,and,or,slt,beq

The first two instructions move data from memory to/from registers. The next fourcarry out operations on registers. The last two are used to affect the next instruction to be fetched and executed. The instruction fetch and execute cycle may involve up to five smaller steps: memory access to get the instruction, register access, the ALU, data movement and finally register access again.

i)   Explain (with the help of diagrams) the two main types of hazards to pipelining of memory access and branch instruction

                                                                                [8 marks]

ii)   Outline (again, with the help of diagrams) the two common solutions to such hazards.

                                                                                 [8 marks]

a)   Re-write the following code to *minimise* the performance on the data path – i. e. to take the largest number of clock cycles to get the same answer!

Loop: Lw $3, 0($5)
          Lw $4, 0($5)
          Add $7,$7,$3
          Add $8,$8,$4
          Add $10,$7,$8
          Sw $6,0($5)
          Beq $10,$11,Loop

                                                                                [17 marks]

Question 4.

Input/Output, Exceptions

a) A serial port on a computer is implemented as a set of registers that are mapped at memory addresses 0xffff0000 onwards. The locations:
+0 keyboard control
+4 keyboard data
+8 display control
+12 display data

The control registers have 1 bit to indicate that data is ready.
Explain the following program by commenting each instruction:

```
        .text
main:   li,$8,0xfffff0000

mloop:
in:     lw $7,0$8)
        andi $7,$7, 1
        blez $7,in

        lw $6,4($8)

out:     lw $7,8($8)
        andi $7,$7,1
        blez $7,out

        sw $6, 12($8)

        b mloop
```

[17 Marks]

b)
i) Outline how a dispatch table is used in a system call handler routine to map traps and interrupts to the appropriate handler code. **[9 Marks]**
iii) Explain the term DMA, and how it helps improve the program and data throughput of a system.

[7 Marks]

[Turn Over]

Question 5.

Logic

a) Explain what is meant by 5 of the following terms: [10 marks]

    1. Two's complement representation
    2. Floating point representation
    3. Binary Coded Decimal representation
    4. Truth table
    5. Karnaugh map
    6. Static hazard
    7. "Don't care" conditions

b) Explain how the mantissa of a floating point number is *normalised* to obtain maximum precision for a given number of bits (e.g. in IEEE Floating-Point representation).   [5 marks]

c) A computer application has to deal with text in a variety of fonts from a set of up to about 250. There are less than 250 characters to each font. Suggest two possible ways of encoding a document that contains characters from a mixture of fonts, indicating the advantages and disadvantages of each way.

    [5 marks]

d) You are asked to design the logic for a convertor from Binary Coded Decimal to Gray Code. Gray Code is a binary representation in which only one bit changes between one digit and its successor, i.e.

0	0000	5	0111
1	0001	6	0101
2	0011	7	0100
3	0010	8	1100
4	0110	9	1101

    1. Draw the truth table for this function.   [2 marks]
    2. Draw the Karnaugh maps (indicating any don't cares)   [5 marks]
    3. Extract the minimised logic equations for the least significant two output bits   [4 marks]
    3. Draw the corresponding circuits for these two outputs.   [2 marks]

[End of Paper]