2017 EXAMINATIONS

Lancaster University

Part I (First Year)

SCHOOL OF COMPUTING AND COMMUNICATIONS

SCC.150 Digital Systems (1 hour & 30 Minutes)

- **Answer any <u>THREE</u> out of the four questions.**
- **Use a <u>separate</u> answer book for <u>each</u> question.**

Question 1

1.a Draw an empty template of a 3-variable Karnaugh map, ensuring that you label the axes correctly.

[3 marks]

1.b i. Write the full Boolean expression of the 3-input truth table below (where A, B, C are inputs and Y is the output):

[3 marks]

A	**B**	**C**	***Y***
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

ii. Using the Karnaugh map template from part (a) try to simplify the Boolean expression you derived earlier in part (b-i). Show all working and give your answer as a (simplified) sum-of-products Boolean expression.

[4 marks]

iii. Using de Morgan's law, manipulate your sum-of-products expression from above to generate an equivalent product-of-products expression that is ready to map to a logic circuit implementation that uses only NAND gates.

[4 marks]

iv. Finally, draw the corresponding circuit using 2-input NAND gates.

[4 marks]

Question 1 continues on the next page…

1.c i. Write down the truth table for the OR function in terms of two inputs, A and B and an output Y.

[2 marks]

ii. Write down the corresponding sum-of-products Boolean function for the OR function.

[2 marks]

iii. Draw the corresponding and most simplified circuit representation of the OR function in terms of 2-input NOR gates.

[3 marks]

[Total 25 marks]

Question 2

2.a Describe the difference between Reduced Instruction Set Computing and Complex Instruction Set Computing.

[3 marks]

2.b Give a MIPS instruction that can be used to set register $s0 to the value 5?

[2 marks]

2.c Convert the instruction below to its MIPS machine language representation (32 bits). Show the value for each of the 4 fields. Give the full 32 bit representation of the instruction in hexadecimal format.

[8 marks]

ori $s0, $zero, 32

OR immediate

`ori rt, rs, imm`

0xd	rs	rt	imm
6	5	5	16

Put the logical OR of register `rs` and the zero-extended immediate into register `rt`.

Name	Register number
$zero	0
$v0–$v1	2–3
$a0–$a3	4–7
$t0–$t7	8–15
$s0–$s7	16–23
$t8–$t9	24–25
$gp	28
$sp	29
$fp	30
$ra	31

2.d The address 0x10000008 should be loaded into register $s2. Write the MIPS assembler code for this task without using MIPS pseudo-instructions. Please comment the code you provide.

[5 marks]

Question 2 continues on the next page…

2.e Write a MIPS assembler program which carries out the following calculation: x = a+b+c

The result x should be available in register $s0 at program completion. Please explain your register mapping. You can assume that a, b and c are present in registers (no loading from memory is required).

[7 marks]

[Total 25 marks]

Question 3

3.a i. Describe the main goal of a program linker during the compilation of an executable program from source code.

[3 marks]

ii. Define the differences between static and dynamic linking. What are the main trade-offs in terms of disk space usage and portability during the compilation of a program when the different linking techniques are used?

[4 marks]

3.b Please explain the difference between unions and structs. Two structs (s1 and s2) and one union (u1) are shown below. How large (in bytes) are the s1, s2 and u1 on a 32bit processor architecture? Explain in detail how you calculate the size.

```
struct s1 {
     char *a;
     char b;
     int   c;
 };

 struct s2 {
    char a[16];
    char   b;
    int c;
}__attribute__((packed));

union u1 {
    char a[16];
    char   b;
    int c;
 };
```

[8 marks]

Question 3 continues on the next page…

3.c A function atoi is required which accepts as input a c string terminated with a NULL character ('\0'). The **`atoi`** function should convert the initial part of the string in s to an integer value. The conversion process should terminate either when a non-numeric or NULL character is found and the function should return the integer value that has been converted up to that character (In case a string doesn't have a numeric character in the beginning, a value of zero should be returned). Provide the implementation of **`atoi`**.

```
int atoi(const char* s) {
    << YOUR CODE >>>
}
```

(Hint: the characters '0'-'9' are represented with the hex values 0x30-0x39 respectively).

[10 marks]

[Total 25 marks]

Question 4

4.a i. Describe the two main limitations of the S-R Flip Flop when we want to implement a bit in a given memory register.

[2 marks]

ii. Describe what the control unit is and what is its main functionality.

[2 marks]

iii. Name the special register in which the currently-executing instruction is held.

[1 mark]

iv. Name and describe the role of the two special registers in the control unit that enable it to control memory.

[2 marks]

4.b Write a MIPS assembler program which has equivalent functionality to the following C code. At program termination, the result should be available in $s0.

```
int a=0;
while (a<10){
   a=a+1;
}
```

[10 marks]

4.c What will the SWAP macro invocation in the following program be expanded to after the source file is parsed by the preprocessor? What will the output of the program be?

```
#include<stdio.h>
#define SWAP(a, b, t) do {\
         t tmp;  \
         tmp=a; \
         a=b;      \
         b=tmp;  \
        } while (0)

int main()
{
    int x=10, y=20;
    SWAP(x, y, int);
    printf("x=%d, y=%d\n", x, y);
    return 0;
}
```

[8 marks]

[Total 25 Marks]

---End of Paper---