UNIVERSITY OF DUBLIN

TRINITY COLLEGE

Faculty of Engineering, Mathematics & Science
School of Computer Science & Statistics

B.A.(Mod.) Computer Science **Trinity Term 2009**
Senior Freshman Examination

2BA4 Computer Architecture I

Tuesday 2nd June 2009 Goldsmith Hall 09:30 – 12:30

Mr. Ross Brennan, Dr. Michael Manzke

Instructions to Candidates:

- Answer FIVE questions, at least TWO from each section.
- Use separate answer books for section A and section B.

Materials permitted for this examination:

- Use of non-programmable calculators is permitted.
- Please note the make and model of your calculator on your answer book.

SECTION A

1.

a) Design a shifter unit which will efficiently implement the following micro-operations under the control of a two-bit select control S on a four-bit input A:

S	Micro-operation
00	shr A $[A_3 \leftarrow 0]$
01	shl A $[A_0 \leftarrow 0]$
10	ror A $[A_3 \leftarrow A_0]$
11	rol A $[A_0 \leftarrow A_3]$

[10 marks]

b) Provide VHDL code for a 4 bit Barrel shifter.

[10 marks]

2.

a) Describe how the status bits of Zero, Negative, Carry and oVerflow are generated in the functional unit of a datapath and explain when it is appropriate to load their values into the status register Z, N, C, V.

[7 marks]

b) For the following sequence of register transfers draw waveforms of R1 and the status bits Z, N, C, V against the clock, assuming that all registers load on the rising clock edge:

Clock	Register transfer
1	R1 ← R2 – R2
2	R1 ← R1 + 1
3	R1 ← rol R1
4	R1 ← R1 - 1

[6 marks]

c) Provide VHDL code that generates Z, N, C, V status bits in the status register.

[7 marks]

3.

a) Draft a register transfer description of a sequential circuit which, upon assertion of a synchronous GO signal, will count the number of '1' bits stored in an 8-bit register A and return the result in register CNT and set a DONE flag.

[12 marks]

b) Derive a circuit schematic from part a), clearly showing the interconnection and control of all the registers, the width of all data and control paths, and the generation of all the control signals.

[8 marks]

4.

a) Draft a schematic showing the principal components of the datapath of a processor unit, showing clearly all the control signals required for each component and any status signals issued by it.

[12 marks]

b) Assuming that the datapath ALU of part a) provides only A+B, A–B, A+1, A–1, A, NOT A, A AND B, A OR B, give a microcode register transfer sequence and the associated symbolic control signals for your datapath that will provide twos complement negation, i.e. it will execute $R_i \leftarrow -R_j$, where R_i and R_j are two processor registers.

[8 marks]

SECTION B

5.

a) Describe the conditions that can cause an address error exception to occur in the MC68008 microprocessor.

[4 marks]

b) Describe the sequence of events that occur when the MC68008 detects a bus error, after the BERRn signal has been asserted.

[6 marks]

c) What changes would be required to the VHDL code of the 2BA4 microprocessor project in order to implement a timer, in the decoder CPLD, which would assert the BERRn signal if a device timeout occurs during a bus transaction. The BERRn signal should also be asserted if the CPU attempts to write to the ROM or access regions of the address space that have not been mapped to a specific device. You can assume that the external devices have already been memory mapped into the system using the following internal active-high decoder signals: CSROM1int, CSRAM1int, CSRAM2int, CSIO1int and CSIO2int.

[10 marks]

6.

a) Discuss the purpose of a RAM vector table and briefly describe how it could be implemented.

[4 marks]

b) Discuss the differences between regular interrupt handling and interrupt auto-vectoring as implemented by the MC68008.

[6 marks]

c) Give a detailed description of the changes that would be required to the 2BA4 microprocessor project hardware and VHDL code in order to allow the CPLD to implement regular interrupt handling, by providing exception vector numbers to the CPU during a vector acquisition cycle, instead of using auto-vectoring.

[10 marks]

7.

a) Describe the sequence of events that occurs after the MC68008 has been reset by asserting both the HALTn and RESETn signals together. How does this differ from a software reset?

[6 marks]

b) What is the purpose of using pull-up and pull-down resistors and under what circumstances should they be used?

[4 marks]

c) Give a detailed description of the read and write cycle operations that allow the MC68008 to communicate with 68008 compatible memory devices. Your description should include all of the signals that are involved in the transactions and should outline the differences between the read and write cycle timings. Briefly discuss the purpose of bus-cycle wait-states.

[10 marks]

8.

a) Describe the purpose and function of the VPAn and VMAn signals as implemented in the microprocessor project.

[4 marks]

b) Design a detailed memory map that will allow the following components to be interconnected in a microprocessor system, through a shared bus.

- 1 x MC68008 CPU [20 address pins, reset @ $00000]
- 2 x 4k EEPROM
- 2 x 8k RAM
- 2 x 4-byte R6551 ACIAs

Your solution should comply with the design rules specified as part of the microprocessor project and should specify the first and last address of every device as well as the address-bus lines that are connected to each device. The first RAM device should be located at a base address of $8000.

[8 marks]

c) Given the following entity port declaration, implement your decoder design from part (b) using VHDL code.

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity decoder is
    port(
        CLK,E          : in  std_logic;
        FC0,FC1,FC2    : in  std_logic;
        A19,A18,A17    : in  std_logic;
```

```
        A16,A15,A14   : in  std_logic;
        A13,A12,A11   : in  std_logic;
        ASn,DSn       : in  std_logic;
        WEn,OEn       : in  std_logic;
        CSROM1n       : out std_logic;
        CSROM2n       : out std_logic;
        CSRAM1n       : out std_logic;
        CSRAM2n       : out std_logic;
        CSIO1         : out std_logic;
        CSIO2         : out std_logic;
        BERRn         : out std_logic;
        DTACKn        : out std_logic;
        VPAn          : out std_logic;
        VMAn          : out std_logic
    );
end decoder;
```

[8 marks]