GLASGOW COLLEGE UESTC

Exam Paper

Digital Circuit Design (UESTC3020)

Date: Tuesday, 18th December 2018
Time: 09:30 – 11:30

Attempt all PARTS. Total 100 marks

Use one answer sheet for each of the questions in this exam.
Show all work on the answer sheet.

Make sure that your University of Glasgow and UESTC Student Identification Numbers are on all answer sheets.

An electronic calculator may be used provided that it does not allow text storage or display, or graphical display.

All graphs should be clearly labelled and sufficiently large so that all elements are easy to read.

The numbers in square brackets in the right-hand margin indicate the marks allotted to the part of the question against which the mark is shown. These marks are for guidance only.

Q1 (a) Explain the importance of *hierarchy* and *re-use* as techniques in the development of large scale digital systems, and discuss how VHDL assists designers in employing these techniques. [6]

(b) Figure Q1.b describes a simple piece of combinatorial logic in VHDL. Assume that all output(s) are driven high during asynchronous system reset, but otherwise outputs only change on the falling edge of the system clock.

(i) Describe the benefits of using synchronous design instead. [4]

(ii) Convert the system of Figure Q1.b to be synchronous, and write its VHDL entity and architecture. [8]

```
entity QUESTION_1_b is
   port (A,B,C : in std_logic;
         Z : out std_logic);
end QUESTION_1_b;

architecture SIMPLE of QUESTION_1_b is
begin
   process (A,B,C)
      Z <= (A and B) or C;
   end process;
end SIMPLE;
```

Figure Q1.b

(c) Figure Q1.c shows schematically a Configurable Logic Block (CLB) from a generic Field Programmable Gate Array. The CLB can route signals from inputs at its North, East, South and West (N_{in}, etc.) to outputs at its North, East, South and West (N_{out}, etc.). It can also perform calculations using a Universal Logic Module – shown in detail to the right of the schematic. The CLB is configured by defining the selection bits of multiplexers (A) to (I), where the multiplexer input chosen by a zero applied to the multiplexer selector is marked in each case.

(i)Which multiplexer(s) and selection bits allow routing of a signal from the CLB to the south of this one, to the CLB to the north. [3]

(ii) Which multiplexer(s) and selection bits allow inputs from CLBs to the north of this one, to be XOR-ed together and sent as output to the CLB to the south of this one. [9]

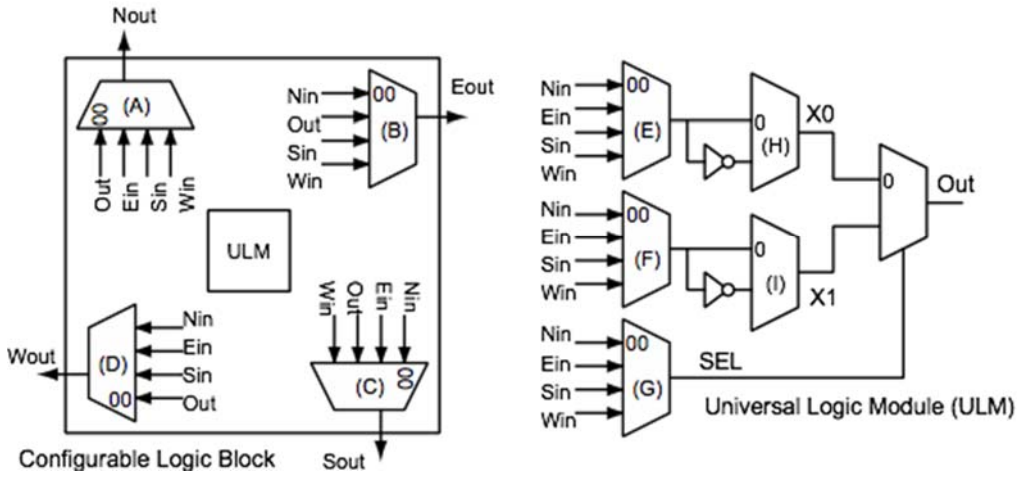

Figure Q1.c

Q2 A ten digit security keypad has two outputs: a single wire which shows a logic '1' when any key is pressed, and a four bit bus whose output is the binary representation of the number of the key pressed. The keypad is attached to a digital system whose VHDL architecture is described in Figure Q2, with the keypad single wire acting as the `KEYPRESS` signal, and the keypad bus acting as `DATA<3:0>`. In addition, the digital system has a single bit input `VERIFY`, and output `CONFIRM`.

(a) Why are case statements rather than if-then-else statements more appropriate to use in Figure Q2. [5]

(b) Draw the state diagram of the VHDL architecture described in Figure Q2. [7]

```
architecture behavior of confirm is
type state is (COUNT, RELAX, REPORT);
   signal PRESENT_STATE: state;
   signal ISUM:   UNSIGNED (3 downto 0) := ''0000'';
   signal ICOUNT: UNSIGNED (3 downto 0) := ''0000'';
begin

process (CLK, RESET)
   begin
      if RESET = '0' then
         PRESENT_STATE <= COUNT; CONFIRM <= '0';
      elsif (CLK'event and CLK='1') then

         CONFIRM <= '0';
         case state is
         when COUNT =>
            if KEYPRESS='1' then ICOUNT <= ICOUNT + 1;
                                 ISUM <= ISUM + UNSIGNED(DATA);
                                   PRESENT_STATE <= RELAX;
            elsif VERIFY='1' then PRESENT_STATE <= REPORT;
            else                  PRESENT_STATE <= COUNT;
            end if;
         when RELAX =>
            if KEYPRESS='1' then  PRESENT_STATE <= RELAX;
            else                  PRESENT_STATE <= COUNT;
            end if;
         when REPORT =>
            if (ISUM=2 and ICOUNT=4) then CONFIRM <='1';
            if VERIFY='1' then PRESENT_STATE <= REPORT;
            else               PRESENT_STATE <= COUNT;
                               ISUM <=''0000''; ICOUNT :=''0000'';
            end if;
         when others =>
            PRESENT_STATE <= WAIT; CONFIRM <= '0';

         end case;

      end if;
   end process;
end behavior;
```

Figure Q2

(c) If the keys 0, 4, 5, 9 are pushed in sequence, each for ten clock cycles, and the single bit input VERIFY is then raised, what will be the value of output CONFIRM ? Give reasons for you answer. [5]

(d) Explain what state machines are used, and why. [3]

Q3 (a) i) Sketch how a 4×4 RAM can be constructed using 1-bit memory elements and decoders.

[5]

ii) Explain the operating principle of the above 4×4 RAM.

[8]

(b) (i) Complete the ROM timing diagram of Figure Q3 (b) by adding traces for chip select (CS_L), output enable (OE_L) and data (DATA). Indicate where the output data (DATA) will be valid or not.

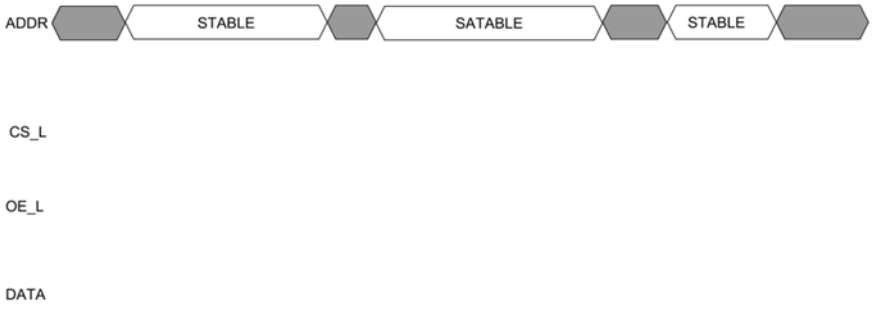

Figure Q3 (b)

[4]

ii) Annotate the following parameters on the diagram you have completed above: *access time from address, access time from chip-select, output-enable time, output-disable time and output hold time.*

[5]

c) Data stored on EEPROM can be erased and rewritten. Why cannot they be considered as a substitute for conventional read/write memories (such as RAM)?

[3]

Q4 (a) Why does synchronisation failure occur?

[3]

(b) Suppose, you are building an embedded microprocessor system with a 10MHz clock and use the circuit of figure Q4 (b) to synchronise an asynchronous input which changes 100,000 times per second. Assume that a typical flip-flop has been used for which the constants T_0 and τ are 0.4s and 1.5ns respectively. If the setup time of the flip-flop is 20ns, calculate the mean time between synchroniser failures.

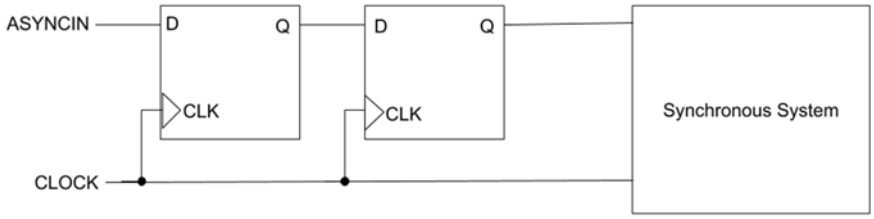

Figure Q4 (b)

[5]

(c) The circuit in Figure Q4 (c) consists of edge triggered flipflops. The second flipflop receives the clock signal through a long slow path which results in delay. Answer the following questions:

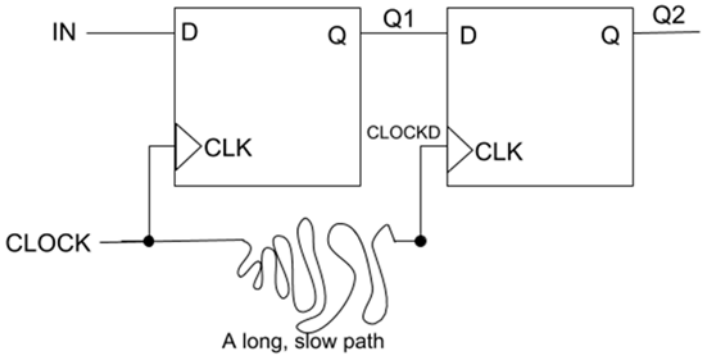

Figure Q4 (c)

i) Define clock skew.

[2]

ii) Explain with a timing diagram, what problem this circuit might face?

[6]

iii) What condition it must satisfy to ensure that the intended circuit output is not affected?

[3]

(d) For the following LFSR circuit in the Figure Q4 (d), let the initial entries of stages R_i be s_i, for $0 \leq i \leq n$. Compute the first 12 bit output for the following cases of initial entries:

	S_3	S_2	S_1	S_0
Case 1	0	1	1	0
Case 2	1	0	1	1

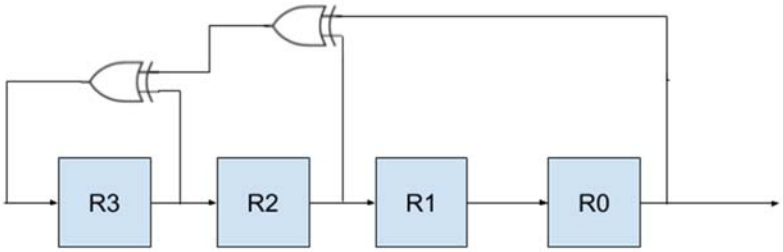

Figure Q4 (d)

[6]