UTHM
Universiti Tun Hussein Onn Malaysia

UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FINAL EXAMINATION
(TAKE HOME)
SEMESTER II
SESSION 2019/2020

COURSE NAME	:	DIGITAL DESIGN
COURSE CODE	:	BEC30503
PROGRAMME CODE	:	BEJ
EXAMINATION DATE	:	JULY 2020
DURATION	:	5 HOURS
INSTRUCTION	:	ANSWER **ONE QUESTION** IN **SECTION A** AND **ALL QUESTIONS** IN **SECTION B. OPEN BOOK EXAMINATION**

THIS QUESTION PAPER CONSISTS OF **SIX (6)** PAGES

SECTION A

Q1 The data flow graph (DFG) given in **Figure Q1** shows the operations and data dependencies of a digital system to be designed. ***a***, ***b***, *c*, ***d*** and ***e*** are all 8-bit data inputs that are registered in the initial state and ***f*** is the output.

(a) If the design is constrained to three arithmetic units, where each arithmetic unit contains a multiplier and an adder. Construct the schedule of this DFG applying "as soon as possible" (ASAP) scheduling.

(8 marks)

(b) Derive the corresponding RTL code for your design.

(6 marks)

(c) Obtain the fbd of the datapath unit of this digital system.

(8 marks)

(d) Obtain the fbd of the control unit showing all the control signals.

(8 marks)

(e) It is given that the propagation delay of the components are as follows: the adder is 30 ns, multiplier is 130 ns and register is 10 ns. Calculate the maximum operating frequency of your design.

(5 marks)

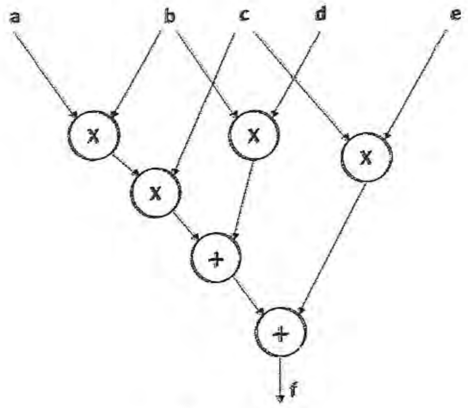

Figure Q1

Q2 A digital system is modelled by the RTL code in **Listing Q2**. Assume that *a*, *b* and *c* are external 8-bit inputs, and the registers are positive-edge triggered. By using the ALU with the functions given in **Table Q2**, answer the following questions. Note that *m** denotes *m* bitwise inverted.

```
S0:   ( )    / R1 ← b;
      ( )    / R2 ← c;
S1:   ( )    / R1 ← R1 * R2;
S2:   (m)    / R2 ← 2R2;
      (m*)   / R2 ← R1 – R2;
      ( )    / done = 1;
      ( )    / goto S0;
```

Listing Q2: RTL code

Table Q2: ALU operation

f_1f_0	output	function
00	X * Y	Multiply
01	X + Y	ADD
10	X - Y	SUB
11	Y	PASS Y

(a) Derive the functional block diagram (fbd) of the datapath unit (DU) for the digital system in **Q2**.

(8 marks)

(b) Write the Verilog code to model the datapath in **Q2(a)**.

(9 marks)

(c) Derive the fbd of the control unit (CU) showing the state registers, next state block, output block and all the control signals. Note: Group the control signals as a vector formatted as follows: *sel*1, *sel*2, *selY*, *ld*1, *ld*2, *f*1, *f*0, *done*.

(6 marks)

(d) Write the Verilog code to model the control unit in **Q2(c)**. The control vector must be formatted as in **Q2(c)**.

(9 marks)

(e) If the multiply function is not available in the ALU, suggest an option to implement the operation of R2 ← R1 * R2 in state S1 if Y input is limited to 2, 4 and 8.

(3 marks)

SECTION B

Q3 (a) A PLD is a general-purpose chip for implementing logic circuits. It contains a collection of logic circuit elements that can be customized in different ways. Discuss the advantage of FPGA as compared to PLA/PAL/CPLD.

(3 marks)

(b) The Verilog code in **Listing Q3(b)** is written using dataflow modelling style. Sketch the respective circuit and rewrite the full Verilog code to describe the circuit by using structural modelling style.

```
module Q3(a, b, c, d, f, g);
    input [1:0] a, b, c, d;
    output f, g;

    assign f = (a & b) | (c & d);
    assign g = (a | c) & (b | d);
endmodule
```

Listing Q3(b)

(7 marks)

(c) Derive the fbd that is described by the Verilog code in **Listing Q3(c)**.

```
module Q3(a, b, m, n, z);
    input [1:0] a, b;
    input m, n;
    output reg [0:3]z;
    reg [1:0] w;

    always@ (a, b, m)
        if (m == 0)
            w = a + b;
        else
            w = a - b;

    always@ (w, n)
        if (n == 0)
            z = 0;
        else
            case(w)
              2'b00: z = 4'h8;
              2'b01: z = 4'h4;
              2'b10: z = 4'h2;
              2'b11: z = 4'h1;
            endcase
endmodule
```

Listing Q3(c)

(10 marks)

Q4 (a) Referring to the Verilog code fragment in **Listing Q4(a)**, derive the fbd of the module M_Q4a. Assuming that A, B, C and D are all 4-bit input signals, complete the input/output declaration for module M_Q4a.

(10 marks)

```
module M_Q4a (K, L, M, A, B, C, D, R);
      .......
      .......
      always@ (negedge K)
      begin
           if (L == 0)
                case (M)
                     2'b00: R <= A;
                     2'b01: R <= B;
                     2'b10: R <= C;
                     2'b11: R <= D;
                endcase
           else R <= R;
      end
endmodule
```

Listing Q4(a)

(b) By referring to the Verilog code in **Listing Q4(b)**, answer the following questions:

```
module M_Q4b (R, L, w, Clock, Q);
      parameter n = 4;
      input [n -1:0] R;
      input L, w, Clock;
      output reg [n -1:0] Q;
      integer k;
always @(posedge Clock)
      if (L)
           Q <= R;
      else
           begin
               for (k = 0; k < n -1; k = k+1)
                  Q[k] < = Q[k+1];
                  Q[n -1] <= w;
           end
endmodule
```

Listing Q4(b)

(i) Determine the type of operation perform by the circuit of module M_Q4b.

(2 marks)

(ii) Verify that the code in **Listing Q4(b)** implements the operation in **Q4(b)(i).**

(8 marks)

Q5 (a) Controller is the sequential circuit that implement Finite State Machine (FSM). The design process of a controller begins with capturing the FSM's behavior and then converting the captured behavior into a circuit. Briefly discuss all substeps required to design a controller.

(5 marks)

(b) Consider the FSM of **Figure Q5(b)** that has an input *m* and an output *n*. Design a controller to implement the **Figure Q5(b)** using a state register and logic gates. Clearly show all the design steps.

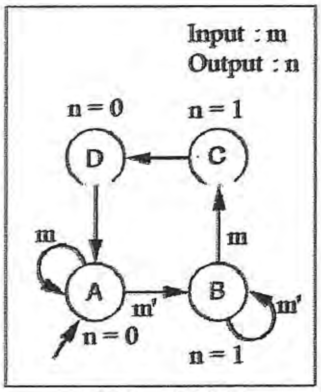

Figure Q5(b)

(10 marks)

(c) **Figure Q5(c)** shows a block diagram of a serial adder. Explain the operation of the circuit by using a suitable example. Assume that the operation is 4-bit.

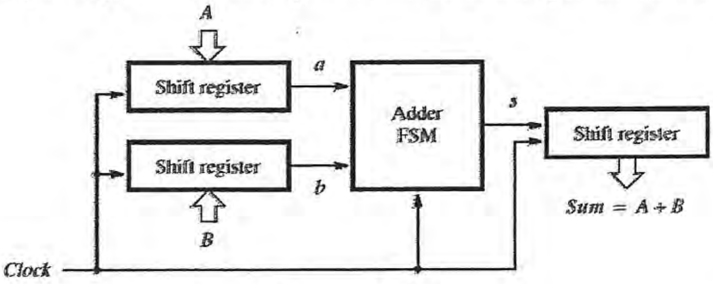

Figure Q5(c)

(5 marks)

-END OF QUESTIONS –