Universiti Tun Hussein Onn Malaysia

UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FINAL EXAMINATION
SEMESTER I
SESSION 2022/2023

COURSE NAME	:	DIGITAL DESIGN
COURSE CODE	:	BEJ 30503 / BEC 30503
PROGRAMME CODE	:	BEJ
EXAMINATION DATE	:	FEBRUARY 2023
DURATION	:	3 HOURS
INSTRUCTION	:	1. ANSWER **ALL QUESTIONS** IN **PART A** AND **ONE (1)** QUESTION IN **PART B**. 2. THIS FINAL EXAMINATION IS CONDUCTED VIA **CLOSED BOOK**. 3. STUDENTS ARE **PROHIBITED** TO CONSULT THEIR OWN MATERIAL OR ANY EXTERNAL RESOURCES DURING THE EXAMINATION CONDUCTED VIA CLOSED BOOK.

THIS QUESTION PAPER CONSISTS OF **EIGHT (8)** PAGES

PART A

Q1 (a) List four principles in digital design to achieve the design objectives effectively and quickly.

(4 marks)

(b) Compare the digital system with the analog system. Thus, explain one advantage of digital signal.

(5 marks)

(c) Explain what digital design is, how it differs from designing an application by using a general-purpose processor?

(4 marks)

(d) The Verilog program in **Figure Q1(d)** is written in behavioral modelling, name the circuit that is described by the codes and rewrite the program by using the dataflow modelling.

(7 marks)

Q2 (a) **Figure Q2(a)** shows a 2-to-4 decoder and its truth table.

(i) Write a Verilog program to describe the given decoder.

(4 marks)

(ii) Design a 3-to-8 decoder by using the decoder in **Figure Q2(a)**. *Hint*: Derive the 3-to-8 decoder truth table based on the given 2-to-4 truth table, then draw the block diagram showing the interconnection of the designed module.

(6 marks)

(b) A combinational logic module, *Sub_Core1* is described by the Verilog program in **Figure Q2(b)**. It is made up of logic component, *Core1_slice*.

(i) Sketch the logic circuit of *Core1_slice* component.

(3 marks)

(ii) Draw the functional block diagram (FBD) of the *Sub_Core1* module, showing how the *Core1_slice* components are interconnected.

(5 marks)

(iii) If the input vector, (*in*3, *in*2, *in*1, *in*0) is 0101, what is the value of the output vector, (*flag*, *out*3, *out*2, *out*1, *out*0)?

(2 marks)

Q3 (a) Convert the decimal numbers -100 into a signed 8-bit number in the following representations:

(i) Sign and magnitude

(1 mark)

(ii) 1's complement

(1 mark)

(iii) 2's complement

(1 mark)

(b) Explain why the 2's complement representation is used to represent a signed number in a computer system instead of sign and magnitude or 1's complement representations.

(4 marks)

(c) **Figure Q3(c)** shows a functional block diagram (FBD) of a 4-bit arithmetic comparator. Based on the diagram, answer the following questions:

(i) What is the logic that should be applied to *Cin*? Explain your answer.

(3 marks)

(ii) Write a full Verilog program to model the given circuit by using the FA module given in **Figure Q3(c)(ii)**.

(7 marks)

(iii) If the input $x = -4$, $y = 5$ and $Cin = 1$, determine the values of b, N, Z, V and $Cout$.

(3 marks)

Q4 (a) Derive the circuit that is synthesized from the Verilog program in **Figure Q4(a)**.

(6 marks)

(b) Explain the differences between the blocking assignment and the non-blocking assignment in describing the sequential circuits.

(4 marks)

(c) A sequence detector to detect a sequence 101 is to be designed using a Finite State Machine (FSM).

(i) Sketch the state diagram for the FSM by using the Moore model.

(6 marks)

(ii) Draw the FBD of the FSM.

(4 marks)

PART B

Q5 A digital system is modelled by the register transfer level (RTL) codes in **Figure Q5**. Given that the registers are positive-edge-triggered and $*a$ denotes a bitwise inverted. The adder-subtractor unit is performing an add operation when $f = 0$, or a subtraction operation when $f = 1$.

(a) From the RTL codes, derive the datapath unit of the digital system. Given A and B are registers and Acc is a load-shift register.

(7 marks)

(b) Design an FBD of the control unit for the digital system.

(6 marks)

(c) Derive the RTL Control Sequence (RTL-CS) table, applying the control vector format *LdA*, *LdB*, *LdAcc*, *f*, *S* and *done*.

(7 marks)

Q6 The data flow graph (DFG) given in **Figure Q6** 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, while x and y are the outputs.

(a) If the design is constrained to one multiplier and one adder. Construct the schedule of this DFG applying "as late as possible" (ALAP) scheduling.

(7 marks)

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

(6 marks)

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

(7 marks)

-END OF QUESTIONS-

FINAL EXAMMINATION

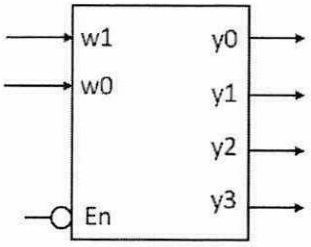
SEMESTER / SESSION	: SEM I 2022/2023	PROGRAMME CODE	: BEJ
COURSE NAME	: DIGITAL DESIGN	COURSE CODE	: BEJ 30503 / BEC 30503

```
module CircuitQ1d(a, b, c, d, S, X);
    input a, b, c, d;
    input [1:0] S;
    output X;
    reg X;

    always@ (a or b or c or d or S)
        if (S == 0) X = a;
        else if (S == 1) X = b;
        else if (S == 2) X = c;
        else X = d;

endmodule
```

Figure Q1(d)



(i)

En	w1	w0	y0	y1	y2	y3
0	0	0	1	0	0	0
0	0	1	0	1	0	0
0	1	0	0	0	1	0
0	1	1	0	0	0	1
1	X	X	0	0	0	0

(ii)

Figure Q2(a)

FINAL EXAMMINATION

SEMESTER / SESSION : SEM I 2022/2023
COURSE NAME : DIGITAL DESIGN
PROGRAMME CODE : BEJ
COURSE CODE : BEJ 30503 / BEC 30503

```
module Sub_Core1(in3, in2, in1, in0, flag, out3, out2, out1, out0);
    input in3, in2, in1, in0;
    output flag, out3, out2, out1, out0;
    wire co0, co1;

    Core1_slice slice0(in3, in0, in3, out0, co0);
    Core1_slice slice1(in3, in1, co0, out1, co1);
    Core1_slice`slice2(in3, in2, co1, out2, flag);
    assign out3 = in3;
endmodule


module Core1_slice(in3, ini, bi, s, co);
    input in3, ini, bi;
    output s, co;
    wire ai;

    assign ai = in3 ^ ini;
    assign s = ai ^ bi;
    assign co = ai & bi;
endmodule
```

Figure Q2(b)

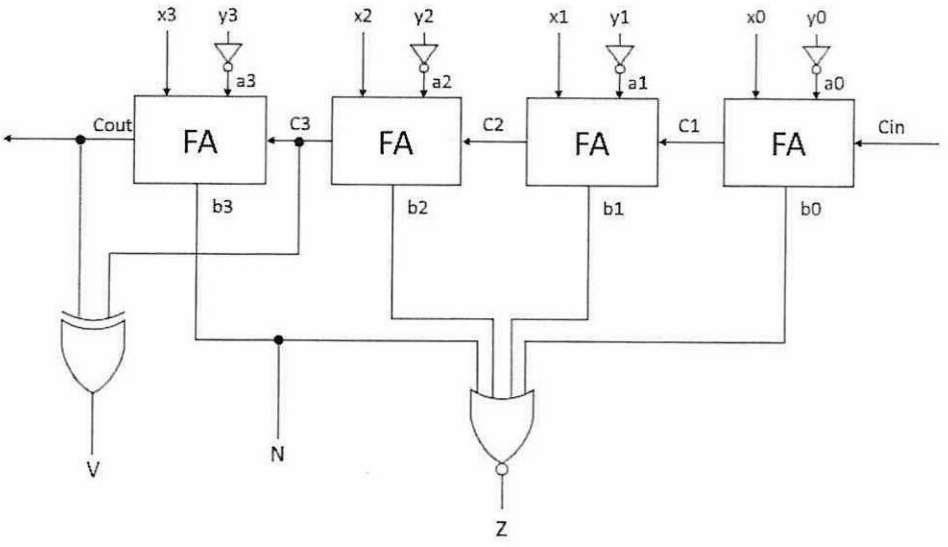


Figure Q3(c)

FINAL EXAMMINATION

SEMESTER / SESSION	: SEM I 2022/2023	PROGRAMME CODE	: BEJ
COURSE NAME	: DIGITAL DESIGN	COURSE CODE	: BEJ 30503 / BEC 30503

```
module FA (x, y, cin, cout, s);
        input x, y, cin;
        output cout, s;

        assign s = x ^ y ^ cin;
        assign cout = x & y | x & cin | y & cin;
endmodule
```

Figure Q3(c)(ii)

```
module CircuitQ4(a, b, c, f, g);
    input a, b, c;
    output reg f, g;
    always@ (posedge Clk, posedge Rst)
        if (Rst) begin f = 0; g = 0; end
        else
            begin
                f = a & b;
                g = f ^ c;
            end
endmodule
```

Figure Q4(a)

FINAL EXAMMINATION

SEMESTER / SESSION : SEM I 2022/2023
COURSE NAME : DIGITAL DESIGN
PROGRAMME CODE : BEJ
COURSE CODE : BEJ 30503 / BEC 30503

```
S0: A ← In
S1: (*a) Acc ← A + B;
    (a) Acc ← A – B;
S2: (*a) Acc ← B + B;
    (a) Acc ← B – B;
S3: B ← (Acc >> 1);
    done;
```

Figure Q5

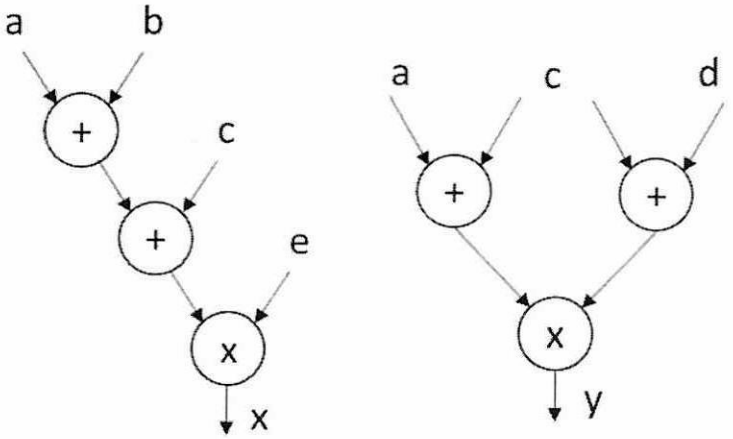


Figure Q6