

**NANYANG TECHNOLOGICAL UNIVERSITY**

**SEMESTER 2 EXAMINATION 2018-2019**

**CE1005/CZ1005 – DIGITAL LOGIC**

Apr/May 2019

Time Allowed: 2 hours

**INSTRUCTIONS**

1. This paper contains 4 questions and comprises 6 pages.
2. Answer **ALL** questions.
3. This is a closed-book examination.
4. All questions carry equal marks.

- 
1. (a) Convert the decimal number 214.307 to hexadecimal. Your answer must have 6 significant digits. Show the steps clearly.

(4 marks)

- (b) Write out a 3-bit Gray code sequence. Briefly describe its merit over a straight binary code sequence in repeating an 8-step process in the correct order.

(4 marks)

- (c) Simplify the following Boolean expression algebraically. Show each step clearly. Give your answer in sum-of-product (SOP) form.

$$F(a, b, c, d, e) = [(c' + b'e)(a'e + b'c)]' [(c'd')' + e']$$

(8 marks)

- (d) Sketch a logic circuit diagram to show how the following Boolean expression may be implemented with only 2-input NOR gates.

$$F = (x + y')(w' + z)$$

(4 marks)

Note: Question No. 1 continues on Page 2

- (e) A logic function is described by the following canonical sum-of-minterm expression:

$$F(A, B, C, D) = \sum m(0, 2, 5, 7, 8, 9, 10, 13, 15)$$

Using a Karnaugh map, obtain a minimum-cost product-of-sum (POS) expression for the function. Show the loops clearly.

(5 marks)

2. (a) Perform each 8-bit signed 2's complement arithmetic operation below and state whether or not there is an overflow. The values are given in hexadecimal.

- $8C + 9A$
- $3F - B6$

(4 marks)

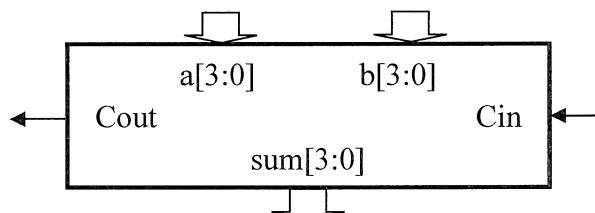
- (b) Figure Q2a shows the block diagram of a 4-bit binary adder. Design a circuit that uses two such adders to add a pair of binary-coded decimal (BCD) digits. Represent a result that exceeds 9 as two BCD digits.

- (i) Obtain a Boolean expression for  $X$ , such that  $X=1$  only when the result exceeds 9. Show your steps clearly.

(4 marks)

- (ii) Sketch a clearly-labelled diagram for the circuit. You may make use of the signal  $X$  described in Q2(b)(i).

(4 marks)



**Figure Q2a**

Note: Question No. 2 continues on Page 3

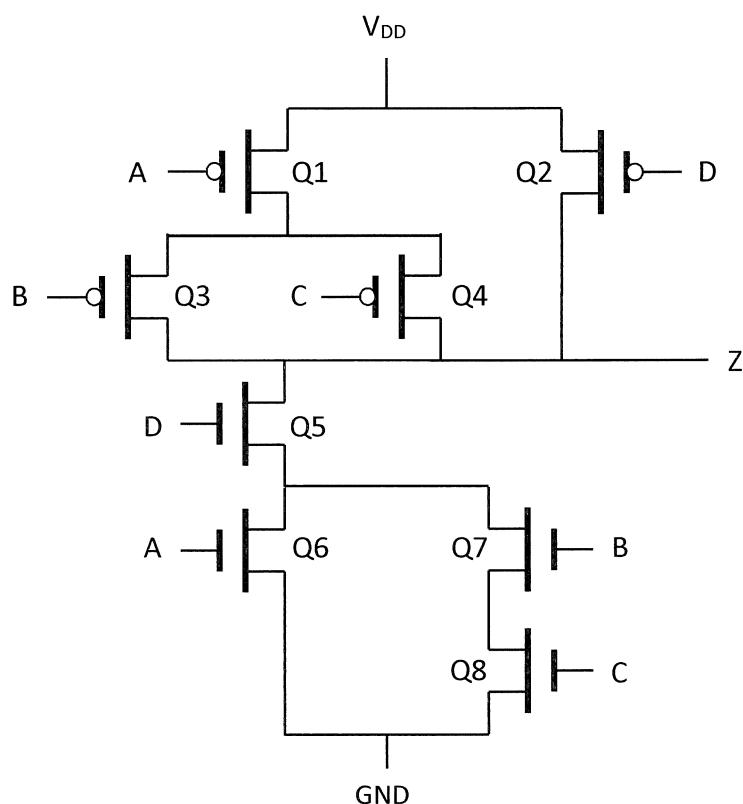
- (c) Four logic signals are asserted as specified below and are negated otherwise. Sketch their waveforms from time  $t=0$  to 40 ms. Label the signals and time scale clearly.
- Active-low reset\* signal is asserted from time  $t=0$  to 10 ms
  - Active-low CS\* is asserted from time  $t=10$  to 30 ms
  - Active-high Read signal is asserted from time  $t=10$  to 20 ms
  - Active-low Write\* signal is asserted from time  $t=20$  to 30 ms

(5 marks)

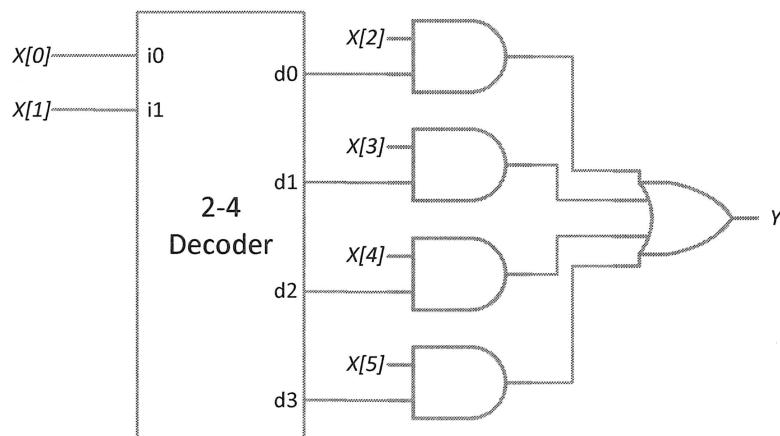
- (d) Figure Q2b shows a 4-input CMOS logic circuit.

- (i) Briefly describe how the inputs A, B, C, D may produce logic 1 and logic 0 at the circuit output Z. (6 marks)

- (ii) Hence, write the canonical product-of-maxterm Boolean expression for the output Z (A, B, C, D). (2 marks)

**Figure Q2b**

3. (a) Design a 2-bit comparator circuit that takes two inputs,  $A[1:0]$  and  $B[1:0]$ . The outputs are  $X$  and  $Y$ , where  $X = 1$  if  $A > B$ , and  $Y = 1$  if  $A \leq B$ .
- (i) Draw the truth table of the 2-bit comparator circuit. (4 marks)
- (ii) Draw the circuit of the 2-bit comparator based on the truth table in Q3a(i). Your circuit must include a 4-16 decoder. (4 marks)
- (b) Figure Q3a shows a combinational circuit with a 6-bit input bus ( $X[5:0]$ ) and an output  $Y$ .
- (i) Write the Verilog module for the 2-4 decoder. (4 marks)
- (ii) Write the Verilog module for the combinational circuit in Figure Q3a by instantiating the decoder that you have written in Q3b(i). (5 marks)
- (iii) The combinational circuit in Figure Q3a has the functionality of a commonly used circuit. What is this commonly used circuit? (4 marks)
- (iv) Based on your answer in Q3b(iii), write the behavioral Verilog module of a circuit with the same functionality as the circuit in Figure Q3a using a combinational **always** block. You are not allowed to use a decoder in your module. (4 marks)

**Figure Q3a**

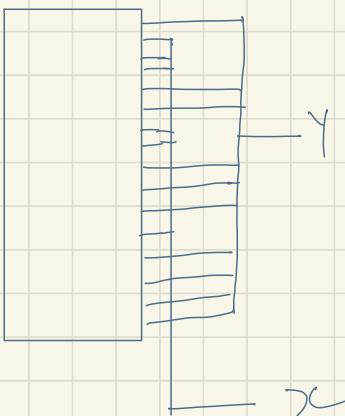
3. (a) Design a 2-bit comparator circuit that takes two inputs,  $A[1:0]$  and  $B[1:0]$ . The outputs are  $X$  and  $Y$ , where  $X = 1$  if  $A > B$ , and  $Y = 1$  if  $A \leq B$ .

- (i) Draw the truth table of the 2-bit comparator circuit. (4 marks)
- (ii) Draw the circuit of the 2-bit comparator based on the truth table in Q3a(i). Your circuit must include a 4-16 decoder.

$A[1] A[0]$	$B[1] B[0]$	$X$	$Y$
0 0	0 0	0	1
0 0	0 1	1	0
0 0	1 0	1	0
0 0	1 1	1	0
0 1	0 0	0	1
0 1	0 1	0	1
0 1	1 0	1	0
0 1	1 1	1	0
1 0	0 0	0	1
1 0	0 1	0	1
1 0	1 0	0	1
1 0	1 1	1	0
1 1	0 0	0	1
1 1	0 1	0	1
1 1	1 0	0	1
1 1	1 1	0	1

$A[1] A[0]$	$B[1] B[0]$	$X$	$Y$
00	00	0	1
00	01	1	1
00	11	1	1
00	10	1	1
01	00	0	1
01	01	0	1
01	11	0	1
01	10	0	1
10	00	0	1
10	01	0	1
10	11	0	1
10	10	0	1
11	00	0	1
11	01	0	1
11	11	0	1
11	10	0	1

$$X = B[1]A[1]' + B[0]A[1]'A[0]' + B[1]B[0]A[0]'$$



- (b) Figure Q3a shows a combinational circuit with a 6-bit input bus ( $X[5:0]$ ) and an output  $Y$ .

(i) Write the Verilog module for the 2-4 decoder.

(4 marks)

```
module decoder2to4 (input [1:0] in,
                     output [3:0] out)
  assign out[0] = in[1]' && in[0]';
  assign out[1] = in[1]' && in[0];
  assign out[2] = in[1] && in[0];
  assign out[3] = in[1] && in[0];
endmodule
```

- (ii) Write the Verilog module for the combinational circuit in Figure Q3a by instantiating the decoder that you have written in Q3b(i).

(5 marks)

00  
01  
10  
11

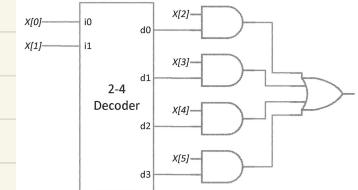


Figure Q3a

```
module decoder3a (input [5:0] X,
                  output Y)
```

```
wire [3:0] D, [5:0] Z;
decoder 2to4 m1 (.in(X[1:0]), .out(D));
assign Z[0] = D[0] && X[2];
assign Z[1] = D[1] && X[3];
assign Z[2] = D[2] && X[4];
assign Z[3] = D[3] && X[5];
assign Y = Z[0] || Z[1] || Z[2] || Z[3];
```

endmodule

4. (a) Figure Q4a shows a sequential Verilog module.

(i) Draw the circuit described by the Verilog module in Figure Q4a with D-type flip-flops and logic gates.

(5 marks)

(ii) Complete the timing diagram shown in Figure Q4b for the Verilog module in Figure Q4a by drawing the waveforms for A, B and C. Assume that  $rst = 0$ .

(8 marks)

```
module mod1 (input clk, rst, output reg A, B, C);

always@(posedge clk)
begin
    if (rst) begin
        A <= 1'b0;
        B <= 1'b0;
        C <= 1'b0;
    end
    else begin
        C <= B;
        B <= A;
        A <= A ^ ~ (B ^ C);
    end
end
endmodule
```

$A \text{ xor } !(B \text{ xor } C)$   
 $A \text{ xor } (B \text{ xnor } C)$

Figure Q4a

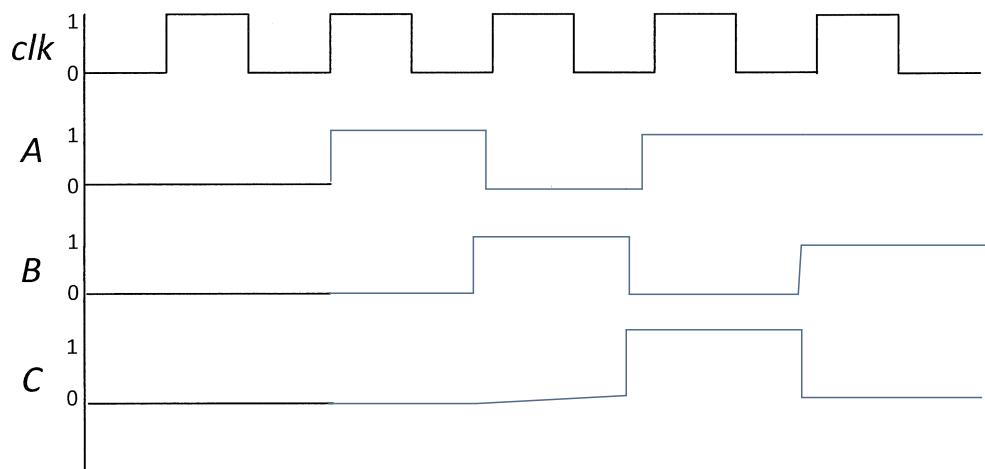
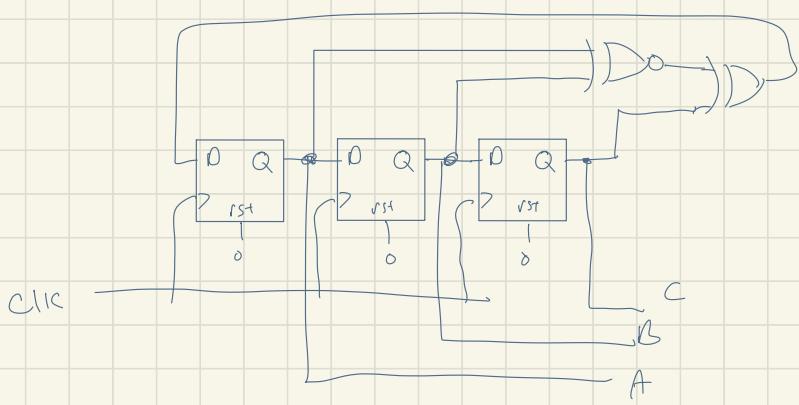


Figure Q4b

Note: Question No. 4 continues on Page 6



(b) Figure Q4c shows a state transition diagram with four states, one input  $B$  and one output  $X$ .

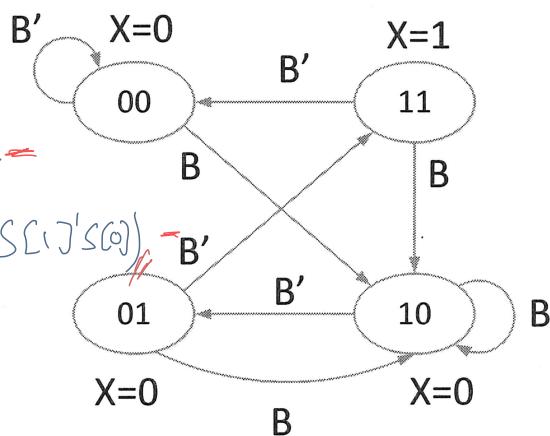
$S$	$B$	NS	$X$
00	0	00	0
00	1	10	0
01	0	11	0
01	1	10	0
10	0	01	0
10	1	10	0
11	0	00	1
11	1	10	1

$$NS = [1:0]$$

$$NS[1] = B + S[1]'S[0]$$

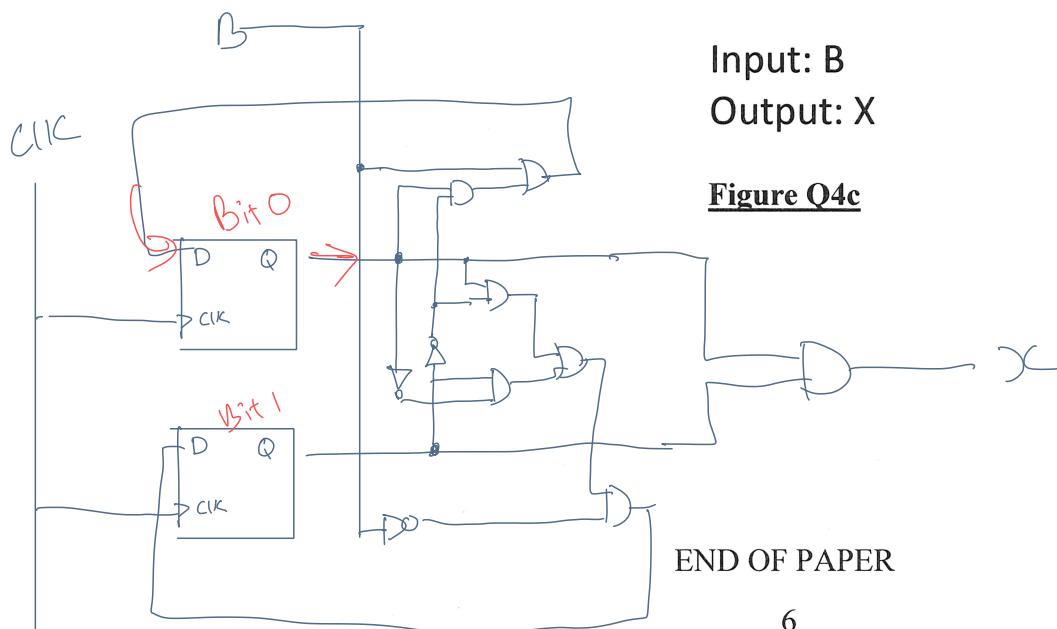
$$NS[0] = B'(S[0]'S[1] + S[1]'S[0])$$

$$X = S[1]S[0]$$



Input:  $B$   
Output:  $X$

Figure Q4c



END OF PAPER



**CE1005 DIGITAL LOGIC  
CZ1005 DIGITAL LOGIC**

Please read the following instructions carefully:

- 1. Please do not turn over the question paper until you are told to do so. Disciplinary action may be taken against you if you do so.**
2. You are not allowed to leave the examination hall unless accompanied by an invigilator. You may raise your hand if you need to communicate with the invigilator.
3. Please write your Matriculation Number on the front of the answer book.
4. Please indicate clearly in the answer book (at the appropriate place) if you are continuing the answer to a question elsewhere in the book.