CS2100 Computer Organization

AY2024/25 Semester 2

Assignment 3

(Deadline: 7 April 2025, Monday, 1pm)

ANSWERS

Instructions

1. There are **9** questions in this assignment, totaling TWENTY (20) marks.

2. This assignment is due on **Monday, 7 April 2025, 1 pm.** The late submissions will incur penalties: a 10% penalty for submissions up to one day late (e.g., submitting at 10.20pm for a 10pm deadline), a 20% penalty for submissions up to two days late, and no grades will be given for assignments submitted more than two days late.

3. Answer these questions on **Canvas > Assignment 3**.

4. Please read and follow the instructions on how to format your answers. This is important as your answers will be autograded, so any answer that departs from the specified format will be graded as incorrect.

5. You should do these assignments on your own. Do not discuss the assignment questions with others.

6. Please post on QnA "Assignments" topic if you have any queries on this assignment.

Note that unless otherwise stated, complemented literals are not available. Constants 1 (true) and 0 (false) are always available. This instruction applies to the final exam and it may not be repeated there.

Important Instructions (don't skip this!):

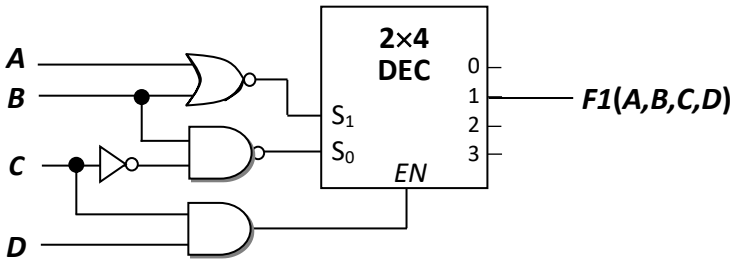
For Questions 1 – 5, when listing out the minterms/maxterms, you are to write them in increasing order, enclosed in a pair of parentheses, separated by comma, with no space and no other punctuation. For example, if the answer is $\boldsymbol{\Sigma m(1,2,3,5)}$, enter **(1,2,3,5)**. Answers such as (1, 2, 3, 5) (with additional spaces), (1,5,2,3) (not in increasing order), and (1,2,3,3,5) (duplicate numbers) will all be graded as wrong with no partial credit given. Pay attention to the order of the variables in the given function.

All the questions in this assignment are autograded. There will be no partial credit for wrong answers.

Question 1. (2 marks)

The diagram below shows a 2×4 decoder with one-enable and active-high outputs. Fill in the list of minterms for *F1*(*A,B,C,D*) in the **$\boldsymbol{\Sigma m}$ notation** (sum-of-minterms notation), i.e. *F1*(*A,B,C,D*) = $\Sigma m(\ldots)$.

(Read the "Important Instructions" above.)



Answer:

F1(A,B,C,D) = (*C*·*D*)·((*A*+B)')'·(*B*·*C*')' = (*C*·*D*)·(*A*+*B*)·(*B*'+*C*) = (*C*·*D*)·(*A*·*B*' + *A*·*C* + *B*·*B*' + *B*·*C*)

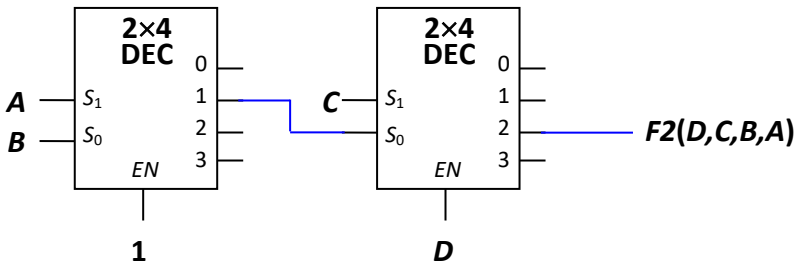
= (*C*·*D*)·(*A*·*B*' + *A*·*C* + *B*·*C*) = *A*·*B*'·*C*·*D* + *A*·*C*·*D* + *B*·*C*·*D* = *A*·*C*·*D* + *B*·*C*·*D* = (*m*11 + *m*15) + (*m*7 + *m*15)

= $\Sigma m(\mathbf{7,11,15})$.

Question 2. (2 marks)

The diagram below shows two 2×4 decoders with one-enable and active-high outputs. Fill in the list of minterms for *F2*(*D,C,B,A*) in the **$\boldsymbol{\Sigma m}$ notation** (sum-of-minterms), i.e. *F2*(*D,C,B,A*) = $\Sigma m(\ldots)$.

(Read the "Important Instructions" on page 1.)

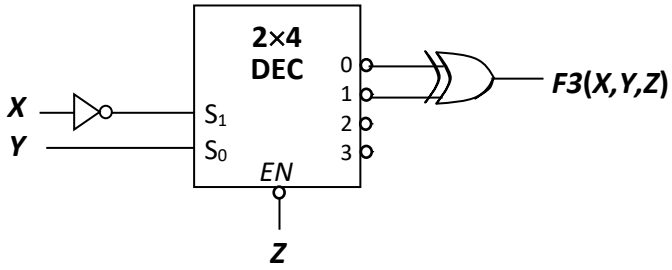


Answer: *F2(D,C,B,A)* = *D*·*C*·(*A*'·*B*)' = *D*·*C*·(*A*+*B*') = *D*·*C*·*A* + *D*·*C*·*B*' = (*m*13 + *m*15) + (*m*12 + *m*13)

= $\Sigma m(\mathbf{12,13,15})$.

Question 3. (2 marks)
The diagram below shows a 2×4 decoder with zero-enable and active-low outputs. Fill in the list of minterms for *F3*(*X,Y,Z*) in the **Σm notation** (sum-of-minterms), i.e. *F3*(*X,Y,Z*) = $\Sigma m(\dots)$.

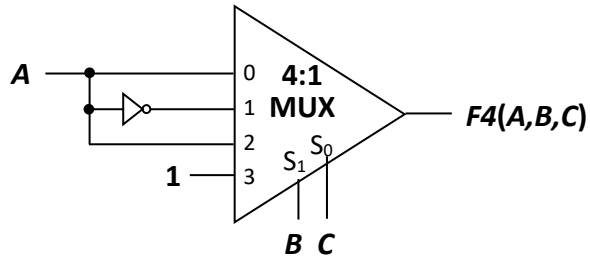
(Read the "Important Instructions" on page 2.)



Answer: *F3(X,Y,Z)* = (*Z*+*X*'+*Y*)'·(*Z*+*X*'+*Y*') + (*Z*+*X*'+*Y*)·(*Z*+*X*'+*Y*')' = *Z*'·*X*·*Y*' + *Z*'·*X*·*Y* = *X*·*Y*'·*Z*' + *X*·*Y*·*Z*' = $\Sigma m(\mathbf{4},\mathbf{6})$.

Question 4. (2 marks)
The diagram below shows a 4:1 multiplexer. Fill in the list of maxterms for *F4*(*A,B,C*) in the **ΠM notation** (product-of-maxterms), i.e. *F4*(*A,B,C*) = $\Pi M(\dots)$.

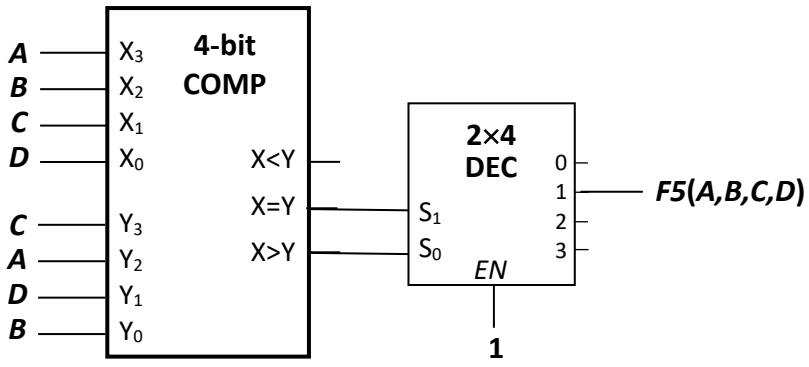
(Read the "Important Instructions" on page 2.)



Answer: *F4(A,B,C)* = *A*·(*B*'·*C*') + *A*'·(*B*'·*C*) + *A*·(*B*·*C*') + 1·(*B*·*C*) = *m*4 + *m*1+ *m*6 + *m*3 + *m*7
= $\Sigma m(4,1,6,3,7)$ = $\Pi M(\mathbf{0},\mathbf{2},\mathbf{5})$.

Question 5. (2 marks)
The diagram below shows a 4-bit magnitude comparator and a 2×4 decoder with 1-enable and active-high outputs. What is *F5*(*A,B,C,D*) in **Σm notation** (sum-of-minterms)? List out the minterms.

(Read the "Important Instructions" on page 2.)



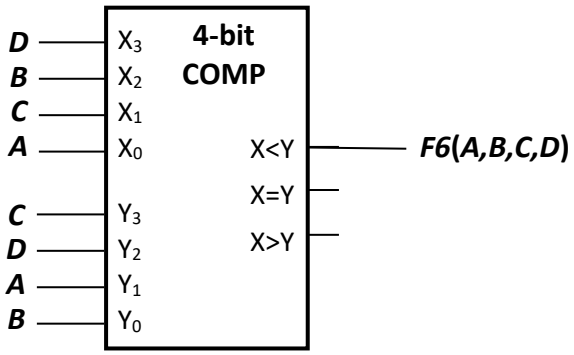
ABCD	*CADB*	*X=Y*	*X>Y*	*F5*
0000	0000	1	0	**0**
0001	0010	0	0	**0**
0010	1000	0	0	**0**
0011	1010	0	0	**0**
0100	0001	0	1	**1**
0101	0011	0	1	**1**
0110	1001	0	0	**0**
0111	1011	0	0	**0**
1000	0100	0	1	**1**
1001	0110	0	1	**1**
1010	1100	0	0	**0**
1011	1110	0	0	**0**
1100	0101	0	1	**1**
1101	0111	0	1	**1**
1110	1101	0	1	**1**
1111	1111	1	0	**0**

Answer: *F5(A,B,C,D)* = $\Sigma m(\mathbf{4},\mathbf{5},\mathbf{8},\mathbf{9},\mathbf{12},\mathbf{13},\mathbf{14})$

Question 6. (2 marks)
The diagram below shows a 4-bit magnitude comparator. What is the simplified SOP expression of *F6*(*A,B,C,D*)?

Shorter product terms must be written before longer product terms. For product terms with the same length, they can be written in any order. If a product term contains more than one literal, you must have the dot (AND) symbol, or your answer will be considered wrong. Within a product term, the literals should be in alphabetical order. Literals should be in upper-case letters, and there should be no space, no parenthesis, and no symbols other than the dot, plus and single quote in your answer.

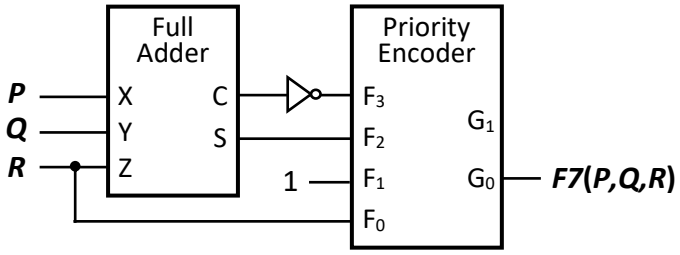
Important: Do NOT paste your answer from Word document into Canvas as the smart quotation mark in Word will not be recognized by Canvas, hence resulting in your answer being graded as wrong.



ABCD	DBCA	CDAB	F6
0000	0000	0000	0
0001	1000	0100	0
0010	0010	1000	1
0011	1010	1100	1
0100	0100	0001	0
0101	1100	0101	0
0110	0110	1001	1
0111	1110	1101	0
1000	0001	0010	1
1001	1001	0110	0
1010	0011	1010	1
1011	1011	1110	1
1100	0101	0011	0
1101	1101	0111	0
1110	0111	1011	1
1111	1111	1111	0

Answer: *F6(A,B,C,D)* = $\Sigma m(2,3,6,8,10,11,14)$ = ***B'·C+C·D'+A·B'·D'*** or ***C·D'+B'·C+A·B'·D'***

Question 7. (2 marks)
A Boolean function *F7(P,Q,R)* is implemented using a full adder, an inverter, and a 4-to-2 priority encoder as shown below. The function tables of the full adder and the priority encoder are also shown below. What is the simplified SOP expression of *F7(P,Q,R)*?



Full adder

X	Y	Z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Priority encoder

F_3	F_2	F_1	F_0	G_1	G_0
0	0	0	0	X	X
0	0	0	1	0	0
0	0	1	X	0	1
0	1	X	X	1	0
1	X	X	X	1	1

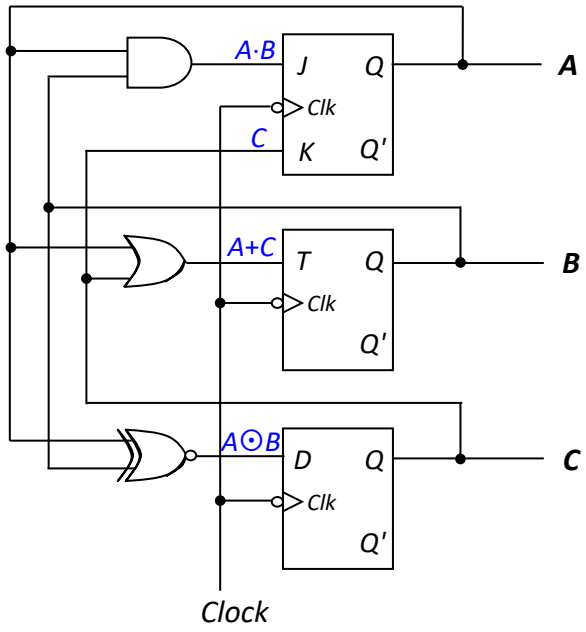
Shorter product terms must be written before longer product terms. For product terms with the same length, they can be written in any order. If a product term contains more than one literal, you must have the dot (AND) symbol, or your answer will be considered wrong. Within a product term, the literals should be in alphabetical order. Literals should be in upper-case letters, and there should be no space, no parenthesis, and no symbols other than the dot, plus and single quote in your answer.

Answer: *F7*(*P*,*Q*,*R*) = ***P'+Q'+R'*** or any permutation of *P'*, *Q'*, and *R'*. Note that (*P*·*Q*·*R*)' is incorrect as it is not an SOP expression.

Working:

P	*Q*	*R*	C	S	F_3=C'	F_2=S	F_1=1	F_0=*R*	G_1	G_0
0	0	0	0	0	1	0	1	0	1	1
0	0	1	0	1	1	1	1	1	1	1
0	1	0	0	1	1	1	1	0	1	1
0	1	1	1	0	0	0	1	1	0	1
1	0	0	0	1	1	1	1	0	1	1
1	0	1	1	0	0	0	1	1	0	1
1	1	0	1	0	0	0	1	0	0	1
1	1	1	1	1	0	1	1	1	1	0

Question 8. (Total: 3 marks)
A sequential circuit with state *ABC* is shown below.



All answers should be in decimal, not binary. Answers in binary will be graded as incorrect. Refer to tutorial 9 for the definition of a sink.

(a) If the initial state is 7 (*ABC* = 111), what state (in decimal) is the circuit in after 5 clock cycles? [1 mark]

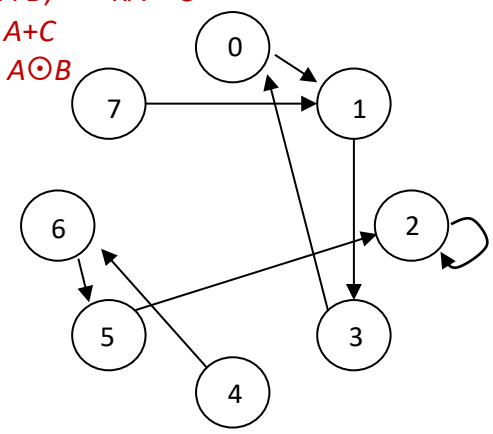
(b) If the initial state is 4 (*ABC* = 100), what state (in decimal) is the circuit in after 5 clock cycles? [1 mark]

(c) Identify all the sink states (in decimal value) in this circuit. If there are more than one sink state, write them in ascending order, separated by comma, without any space. There should be no punctuation other than comma. If there are no sink states, write **none**. [1 mark]

Answers: (a) 3 (7 → 1 → 3 → 0 → 1 → 3), (b) 2 (4 → 6 → 5 → 2 → 2 → 2), (c) 2

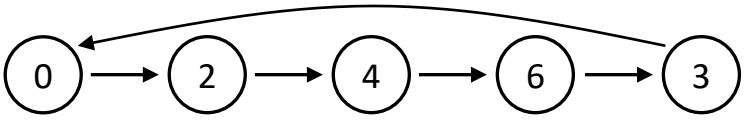
Working:

JA = A·B; *KA = C*
TB = A+C
DC = A⊙B



Present state			Flip-flop inputs				Next state		
A	***B***	***C***	***JA =A·B***	***KA =C***	***TB =A+C***	***DC =A⊙B***	***A^+***	***B^+***	***C^+***
0	0	0	0	0	0	1	0	0	1
0	0	1	0	1	1	1	0	1	1
0	1	0	0	0	0	0	0	1	0
0	1	1	0	1	1	0	0	0	0
1	0	0	0	0	1	0	1	1	0
1	0	1	0	1	1	0	0	1	0
1	1	0	1	0	1	1	1	0	1
1	1	1	1	1	1	1	0	0	1

Question 9. (Total: 3 marks)
A sequential circuit goes through the following states, whose state values are shown in decimal:



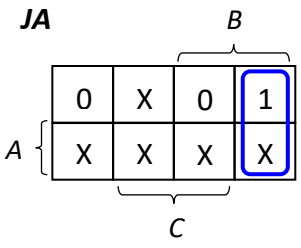
The states are represented by 3-bit values *ABC*. Implement the sequential circuit using a *JK* flip-flop for *A*, a *T* flip-flop for *B*, and a *D* flip-flop for *C*.

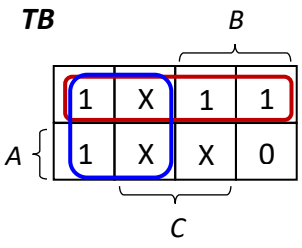
Write the simplified SOP expression for each of the following flip-flop inputs. For each product term in your SOP expression, write the literals in alphabetical order. Literals should be in upper-case letters, and there should be no space, no parenthesis, and no symbols other than the dot, plus and single quote in your answer.

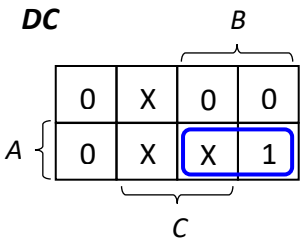
(a) Flip-flop input *JA* [1 mark]; (b) Flip-flop input *TB* [1 mark]; (c) Flip-flop input *DC* [1 mark]

Answers:
(a) *JA* = ***B·C'***,
(b) *TB* = ***A'+B'*** or ***B'+A'***
(c) *DC* = ***A·B***

A	*B*	*C*	*A^+*	*B^+*	*C^+*	*JA*	*KA*	*TB*	*DC*
0	0	0	0	1	0	0	X	1	0
0	0	1	X	X	X	X	X	X	X
0	1	0	1	0	0	1	X	1	0
0	1	1	0	0	0	0	X	1	0
1	0	0	1	1	0	X	0	1	0
1	0	1	X	X	X	X	X	X	X
1	1	0	0	1	1	X	1	0	1
1	1	1	X	X	X	X	X	X	X







=== END OF PAPER ===