King Saud University
College of Computer and Information Sciences
Computer Science Department

Course Code: CSC 220
Course Title: Computer Organization

Final Exam
Semester: 2nd (2018-2019)
Duration: 3 hours

Student's Name: _____

Student's ID: _____

Section No. _____

Important Exam Regulations

1. Answer all questions
2. No calculator/electronic device is allowed

EVALUATION

	Computer Science B.Sc. Program: NCAAA: Intended Learning Outcomes (ILO) Student Outcomes ABET: Program Learning Outcomes (PLO) Student outcomes	Question No. Relevant Is Hyperlinked	Covering %	Full Mark	Obtained Mark
NCAAA	**Knowledge (NCAAA)**	**Exercise 1**	$\sum ABET\%$		
ABET	(a) Apply knowledge of computing and mathematics appropriate to the discipline;	Exercise 1	40 %	**8**	
		Exercise 2		**8**	
NCAAA	**Cognitive Skills (NCAAA)**	**Exercise 2-3**	$\sum ABET\%$		
ABET	(b) Ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.	Exercise 3	30 %	**6**	
		Exercise 4		**6**	
	(c) An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs.	Exercise 5	30 %	**6**	
		Exercise 6		**6**	
		Total	**100%**	**40**	

Feedback and Remarks:

I certify that the work contained within this assignment is all my own work and referenced where required.

Student's Signature: **Date:**

Question 1. Short Questions (8 Marks: 8 ×1)

a) Find the 8-bit signed-magnitude representation of (– 63).

Answer: 10111111

b) Write the truth table for a full adder circuit.
Answer:

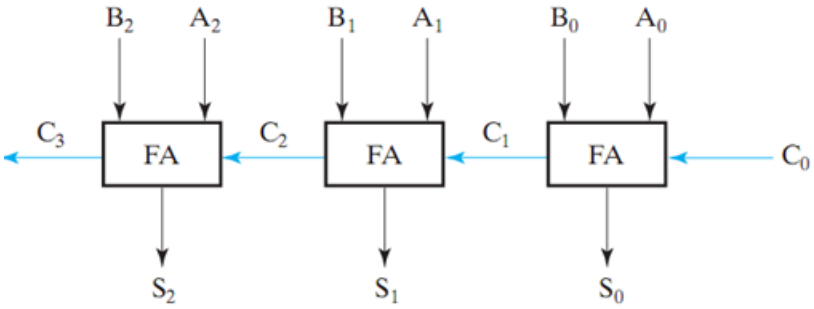
X	Y	Z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

c) Given A=1100 and B = 0100 in 2's complement form. Find A+B and A-B

Answer: A+B = 0000
A-B = 1000

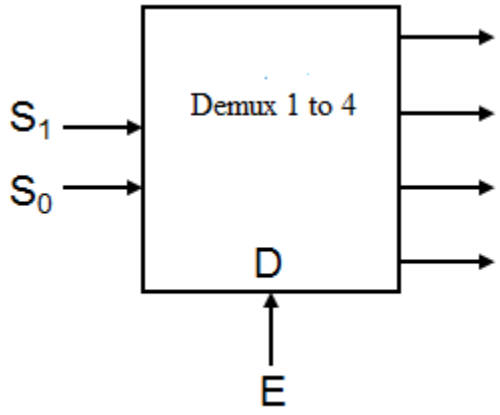
d) Show how to construct a 3-bit parallel adder using full adders?

Answer:



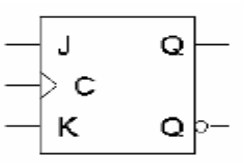
e) Show how to use a Demux 1 to 4 as a decoder 2 to 4 with enable (Hint: used block diagram of a Demux).

Answer:



f) Write the block diagram and characteristic table for JK flip flop

Answer:



C	J	K	Q_{next}
0	x	x	No change
1	0	0	No change
1	0	1	0 (reset)
1	1	0	1 (set)
1	1	1	$Q'_{current}$

g) Suppose a 4-bit register initially contains 1101. What will be the content of the register after arithmetical shift left followed by arithmetical shift right operations?

Answer:

After arithmetical shift Left : 1010

arithmetical shift right: 1101

h) Convert the 2's complement signed binary number (11011000) into decimal.

Answer: -40

Question 2. Short Questions (8 Marks: 2+ 6×1)

a) Convert the assembly instructions given in the following table into RTL instructions

Instructions	RTL
LD R5, (R1)	R5 ← M[R1]
ST (R5), 0	M[R5] ← 0
ADD R0, R5, R1	**R0 ← R5 + R1**
INC R0, R1	**R0 ← R1 + 1**

b) Choose correct answer and write them in the following table

1	2	3	4	5	6
A	D	D	C	B	C

1. Assuming registers are 8-bits width, and R1 = CF, R2 = 6F, R3 = 9F, R4 = FF, ""numbers are represented in HEX and in 2's complement" what is the contents of R1 in decimal, if the instruction ADD, R1, R1, R4 (in RTL: R1 <-- R1 + R4) is executed?

[A] -50 [B] 50 [C] 51 [D] -51

2. A digital computer has a common bus system for 8 registers of 32 bit each. The bus is constructed with multiplexers. How many multiplexers are there in the bus?

[A] 16, [B] 8, [C] 4, [D] 32

3. Assuming registers are 8-bits width, and R1 = CF, R2 = 6F, R3 = 9F, R4 = FF, ""numbers are represented in HEX and in 2's complement" what is the contents of overflow Flag "V" and Zero Flag "Z" and Sign "N" if the instruction SUB, R1, R1, R4 (in RTL: R1 <-- R1 - R4) is executed?

[A] V = 0, Z = 0, N = 1, **[B]** V = 1, Z = 0, N = 0, **[C]** V = 0, Z = 1, N= 0, **[D]**V = 0, Z = 0, N=1

4. How many JK flip flops are required to design a MOD-100 counter?

[A] 5 **[B]** 6 **[C]** 7 **[D]** 8

5. How many 128 x 8 RAM chips are needed to provide a memory capacity of 2048 bytes?

[A] 8 **[B]** 16 **[C]** 24 **[D]** 32

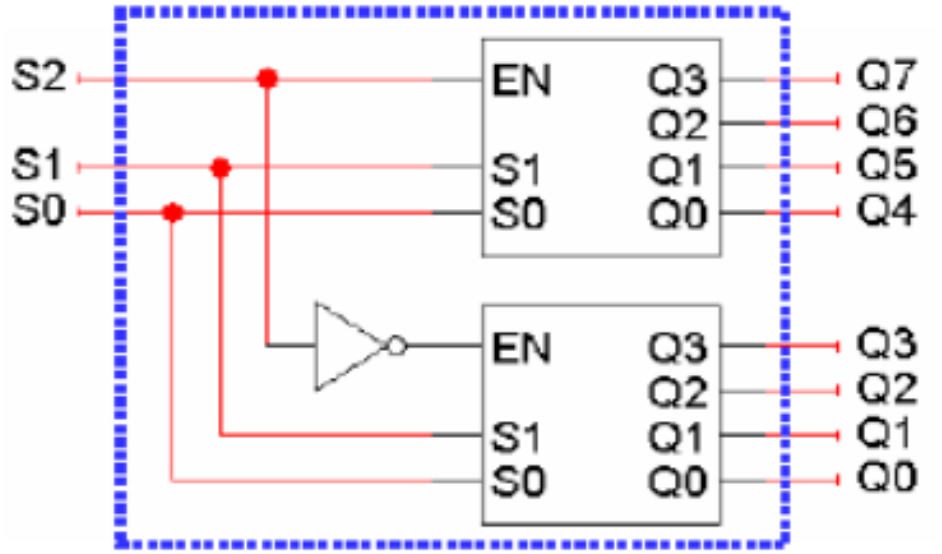
6. Which of the following registers is used to keep track of address of the memory location where the next instruction is located?

[A] Address Register, **[B]** Data Register, **[C]** Program Counter, **[D]** Instruction Register

Question 3 (6 Marks: 2+1+3)

(a) Construct a 3-to-8-line decoder with two 2-to-4 line decoders with enable.

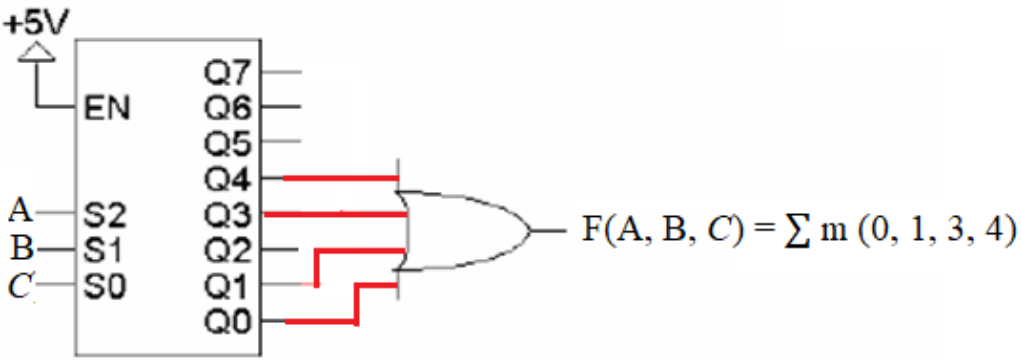
Answer of Question 3 (a)



(b) Implement the following Boolean function with a decoder.

$F(A, B, C) = \sum m\,(0, 1, 3, 4)$

Answer of Question 3 (b)



(c) Write the truth table for decimal to BCD encoder with 10 inputs (D_0, D_1, …, D_9) and 4 output lines (A_3, A_2, A_1, A_0), Write the expression function of the 4 outputs.

Answer of Question 3 (c)

Inputs										Outputs			
D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_3	A_2	A_1	A_0
0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	0	1	0	0	0	0	1	0
0	0	0	0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	1
0	0	0	1	0	0	0	0	0	0	0	1	1	0
0	0	1	0	0	0	0	0	0	0	0	1	1	1
0	1	0	0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	0	0	0	1	0	0	1

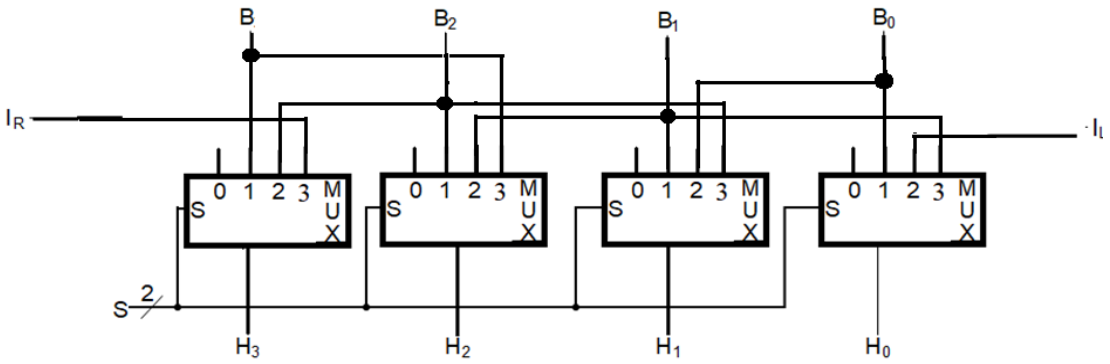
$A_0 = D_1 + D_3 + D_5 + D_7 + D_9$
$A_1 = D_2 + D_3 + D_6 + D_7$
$A_2 = D_4 + D_5 + D_6 + D_7$
$A_2 = D_4 + D_5 + D_6 + D_7$
$A_3 = D_8 + D_9$

Question 4 (6 Marks: 2 +4)

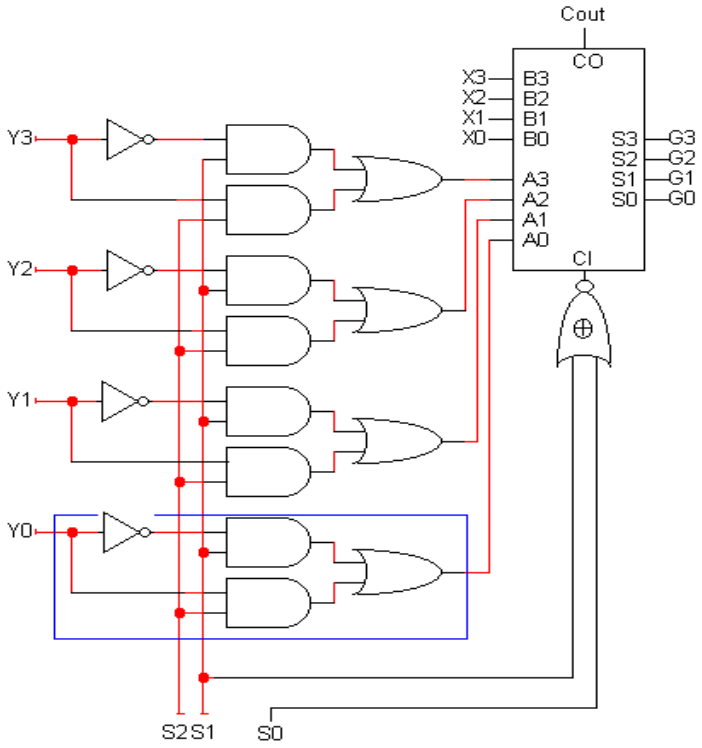
(a) Complete the diagram below to design a 4-bit combinational shifter whose operations are defined by the following table and I_R and I_L are serial inputs for right shift and left shift respectively.

S1, S0	Function
0 0	Unused
0 1	Pass B unchanged
1 0	Left shift
1 1	Right shift

Answer of Question 4 (b)



(b) The figure below shows the design of a 4-bit arithmetic unit. Fill the missing parts in the function table below.



Selection code S_2	S_1	S_0	Required adder inputs A	B	CI	Resulted arithmetic operation G (A + B + CI)	
0	0	0	0000	X	1	X + 1	(increment)
0	0	1	0000	X	0	X	(transfer)
0	**1**	**0**					
0	**1**	**1**					
1	**0**	**0**					
1	**0**	**1**					
1	**1**	**0**					
1	**1**	**1**					

Question 5 (6 Marks: 3+3)

(a) Compute the control words for the following micro-operations for the datapath below described by the table 8.5.

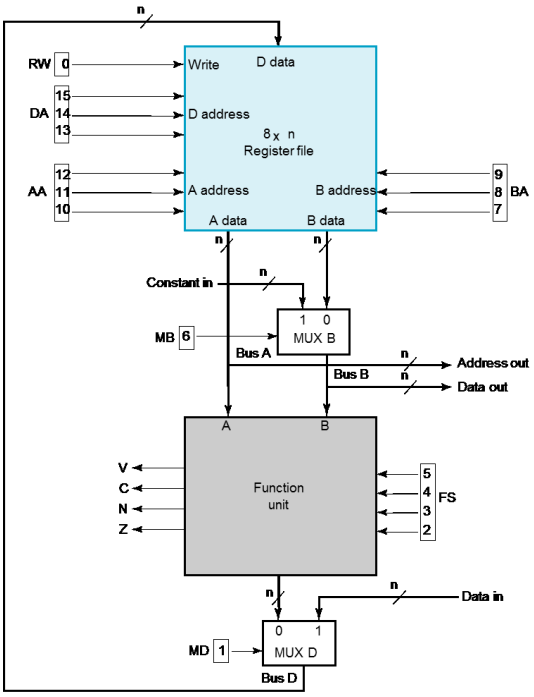
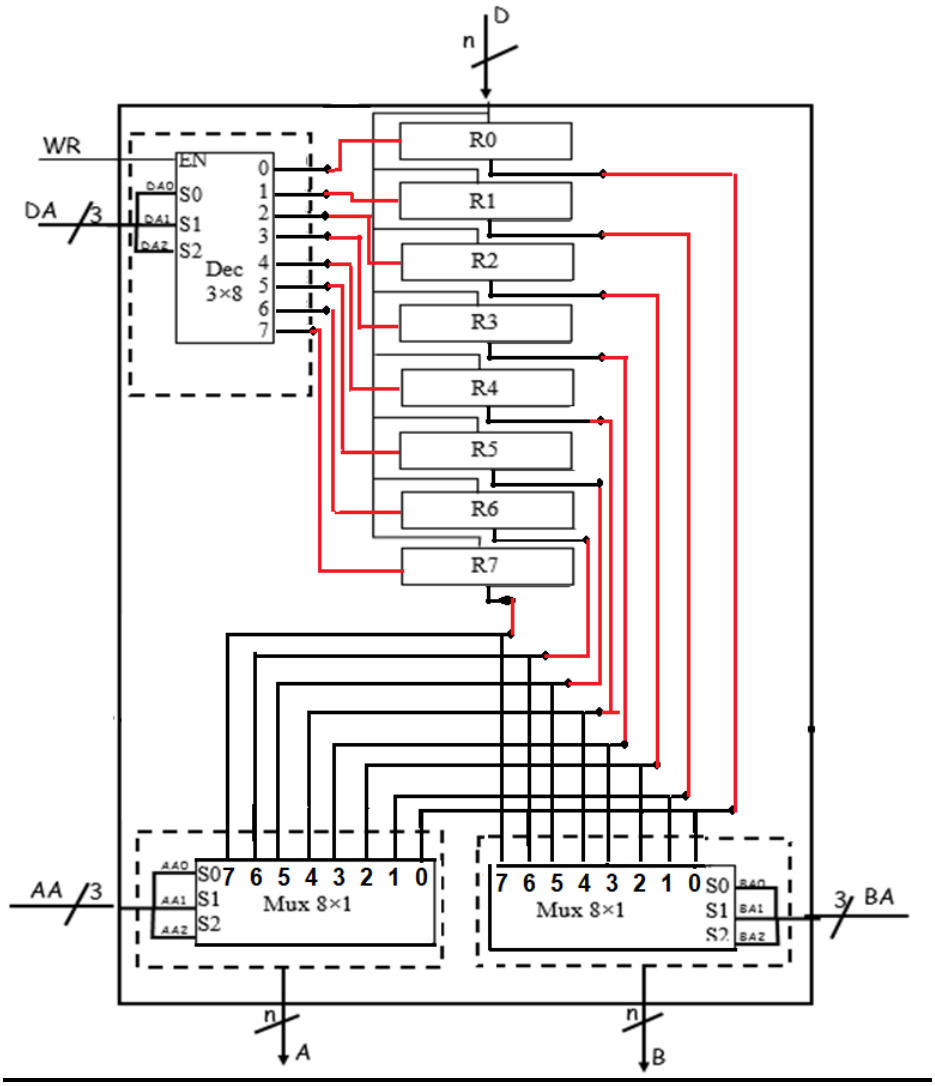


TABLE 8-5
Encoding of Control Word for the Datapath

DA, AA, BA		MB		FS		MD		RW	
Function	Code	Function	Code	Function	Code	Function	Code	Function	Code
$R0$	000	Register	0	$F = A$	0000	Function	0	No Write	0
$R1$	001	Constant	1	$F = A + 1$	0001	Data in	1	Write	1
$R2$	010			$F = A + B$	0010				
$R3$	011			$F = A + B + 1$	0011				
$R4$	100			$F = A + \overline{B}$	0100				
$R5$	101			$F = A + \overline{B} + 1$	0101				
$R6$	110			$F = A - 1$	0110				
$R7$	111			$F = A$	0111				
				$F = A \wedge B$	1000				
				$F = A \vee B$	1001				
				$F = A \oplus B$	1010				
				$F = \overline{A}$	1011				
				$F = B$	1100				
				$F = \text{sr } B$	1101				
				$F = \text{sl } B$	1110				

Micro-operations	DA (15-13)	AA (12-10)	BA (9-7)	MB (6)	FS (5-2)	MD (1)	RW (0)
R5←R4 – R1	**101**	**100**	**001**	**0**	**0101**	**0**	**1**
R5←R4 + 4	**101**	**100**	**xxx**	**1**	**0010**	**0**	**1**
R6←R0 ⊕ R7	**110**	**000**	**111**	**0**	**1010**	**0**	**1**
R6←R0 ∧ R7	**110**	**000**	**111**	**0**	**1000**	**0**	**1**
R3← R2	**011**	**010**	**xxx**	**x**	**0000**	**0**	**1**
R3←sr R2	**011**	**Xxx**	**010**	**0**	**1101**	**0**	**1**

(b) Show how to design a **8 x n** register file using necessary registers, MUXs, and decoders.



Question 6 (6 Marks: 2+4)

TABLE 8-8
Instruction Specifications for the Simple Computer

Instruction	Opcode	Mnemonic	Format	Description	Status Bits
Move A	0000000	MOVA	RD, RA	R[DR] ← R[SA]*	N, Z
Increment	0000001	INC	RD, RA	R[DR] ← R[SA] + 1*	N, Z
Add	0000010	ADD	RD, RA, RB	R[DR] ← R[SA] + R[SB]*	N, Z
Subtract	0000101	SUB	RD, RA, RB	R[DR] ← R[SA] − R[SB]*	N, Z
Decrement	0000110	DEC	RD, RA	R[DR] ← R[SA] − 1*	N, Z
AND	0001000	AND	RD, RA, RB	R[DR] ← R[SA] ∧ R[SB]*	N, Z
OR	0001001	OR	RD, RA, RB	R[DR] ← R[SA] ∨ R[SB]*	N, Z
Exclusive OR	0001010	XOR	RD, RA, RB	R[DR] ← R[SA] ⊕ R[SB]*	N, Z
NOT	0001011	NOT	RD, RA	R[DR] ← $\overline{R[SA]}$*	N, Z
Move B	0001100	MOVB	RD, RB	R[DR] ← R[SB]*	
Shift Right	0001101	SHR	RD, RB	R[DR] ← sr R[SB]*	
Shift Left	0001110	SHL	RD, RB	R[DR] ← sl R[SB]*	
Load Immediate	1001100	LDI	RD, OP	R[DR] ← zf OP*	
Add Immediate	1000010	ADI	RD, RA, OP	R[DR] ← R[SA] + zf OP*	N, Z
Load	0010000	LD	RD, RA	R[DR] ← M[SA]*	
Store	0100000	ST	RA, RB	M[SA] ← R[SB]*	
Branch on Zero	1100000	BRZ	RA, AD	if (R[SA] = 0) PC ← PC + se AD, if (R[SA] ≠ 0) PC ← PC + 1	N, Z
Branch on Negative	1100001	BRN	RA, AD	if (R[SA] < 0) PC ← PC + se AD, if (R[SA] ≥ 0) PC ← PC + 1	N, Z
Jump	1110000	JMP	RA	PC ← R[SA]*	

* For all of these instructions, PC ← PC + 1 is also executed to prepare for the next cycle.

(a) Consider table 8-8 containing instruction specification for a simple computer. Translate the following instructions into 16-bit binary machine codes (**NB:** use 0 for don't care).

Instruction	**Binary machine code**			
	Opcode	**RD/AD(left)**	**RA**	**RB/OP/AD(Right)**
ADD R5, R0, R3	**0000010**	**101**	**000**	**011**
LD R5, (R6)	**0010000**	**101**	**110**	**000**
LDI R2, 5	**1001100**	**010**	**000**	**101**
BRN R1 AD (AD = 110 001)	**110001**	**110**	**001**	**001**

(b) Assume that variable *x* is located at the address 200 in data memory contains 2, and variable *y* is located at the address 250 contains 20. Write an assembly language program to evaluate the equation $z = (x+1) - (y + 5)$ where variable *z* is located at the address 300. (**NB:** use only register R0 as the address register).

Solution:

```
LDI   R0, 200
LD    R1, (R0)
INC   R1, R1

LDI   R0, 250
LD    R2, (R0)
ADI   R2, R2, 5

SUB   R3, R1, R2

LDI   R0, 300
ST    (R0), R3
```

THE END