Final-term Assignment, Fall'2021-22

Subject: Microprocessor and Embedded System

Mark: 30

Submission Deadline: **19/12/2021**

Special Instruction: Only the hand-written assignment submitted in PDF is acceptable. Each page of the assignment should contain the 'Name' and 'ID' of the author. Assignment submitted as image/word file without name and ID will get "Zero" and there will be no second chance to submit it again.

Content

1. Design a 1-bit ALU for operations listed in Table-1. **[10]**
2. Design a 4-bit shifter for the enlisted operation in Table-1. **[6]**

TABLE 1: FUNCTIONS OF CONTROL VARIABLES

Binary Code	Functions of selection variables					
	A	**B**	**D**	**F with C_{in} = 0**	**F with C_{in} = 1**	**H**
0 0 0	Input Data	Input Data	None	A-1	A	1's to the output Bus
0 0 1	R1	R1	R1	A+B	A+B+1	Shift Left with I_L=0
0 1 0	R2	R2	R2	A-B-1	A-B	No Shift
0 1 1	R3	R3	R3	A	A+1	Circulate Left with Carry
1 0 0	R4	R4	R4	$\bar{A}$	X	0's to the output Bus
1 0 1	R5	R5	R5	AX**OR** B	X	-
1 1 0	R6	R6	R6	A **AND** B	X	Circulate-Right with Carry
1 1 1	R7	R7	R7	A **OR** B	X	Shift Right with I_R=0

3. Develop the control memory outputs for the sequence presented in Table-3 using the information listed in Table-1 and 4. In order to complete the memory outputs, use the microinstructions provided below- **[14]**

TABLE 2: SYMBOLIC MICROPROGRAM FOR CONTROL MEMORY

ROM Address	Microinstruction
0	X = 1, if (q_s = 1) then (go to 1), if (q_a = 1) (then go to 2), if (q_s and q_a = 0) then (go to 0)
1	Bs ← $\bar{B}_s$
2	If (S = 1) then (go to 4)
3	A ← A + B, E ← C_{out} go to 0
4	A ← A + $\bar{B}$ + 1, E ← C_{out}
5	If (E = 1) then (go to 0), E ← 0
6	A ← $\bar{A}$
7	A ← A + 1, As ← $\bar{A}_s$, go to 0

TABLE 3: CONTROL MEMORY BIT SEQUENCE

ROM Address			ROM outputs												
			x	S_2	S_1	S_0	C_{in}	L	y	z	w	Address			Select	
			1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	0	0														
0	0	1														
0	1	0														
0	1	1														
1	0	0														
1	0	1														
1	1	0														
1	1	1														

TABLE 4: MULTIPLEXER INPUT

Rom bits		MUX select function
13	**14**	
0	0	Load input to CAR
0	1	Increment CAR
1	0	Load input to CAR if E = 1, increment CAR if E = 0
1	1	Load inputs to CAR if S = 1, increment CAR if S = 0