

Computer Architecture Lab #2

Objectives

After this lab, the student should:

- Understand VHDL basics:
 - o Multiple Architectures for same entity
 - o Generic Entities
 - o Use of (For ... Generate)
 - Understand (When ... Else) VS (With ... Select)

Requirements

Design a n-bit ALSU that accepts two n-bit input values A and B and provides output F, the ALSU has 4 selection inputs S3, S2, S1, S0 and Cin input. The ALSU provides a total of 20 operations specified in the following table

	S₃	S ₂	S ₁	S ₀	Cin =0	Cin = 1			
Part A	0	0	0	0	F = A	F = A + 1			
	0	0	0	1	F = A + B	F = A + B + 1			
	0	0	1	0	F = A - B - 1	F = A - B			
	0	0	1	1	F= A – 1	F = 0			
Part B	0	1	0	0	Done Last Lab				
	0	1	0	1					
	0	1	1	0					
	0	1	1	1					
Part C	1	0	0	0					
	1	0	0	1					
	1	0	1	0					
	1	0	1	1					
Part D	1	1	0	0					
	1	1	0	1					
	1	1	1	0					
	1	1	1	1					

Deliverables:

- 1. Write VHDL code for part A in a separate VHDL files (don't forget to output the carry out)
- 2. You should use the full-adder given in the explanation instead of the VHDL operators (+) and (-)
- 3. Compile, your Code should be free of errors and warnings.
- 4. Simulate partA using Do file (TestCases at the end of the Document).
- 5. Bonus: optimized design (hint: you can use one full adder for part A).
- 6. **Assignment**: Integrate the 4 parts in one file And Make all of them **Generic and Simulate** them using **Do file**.

N.B. you will be graded for code neatness and understanding, Good luck

Architecture -Lab Lab #2

F0F0

F0F0

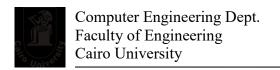
F0F0

0

E1E0

E1E1

F878



To test part A use the following table, let n = 16

Operation	A	В	Cin	Cout	F	
F = A	0F0F	-	0	0	0F0F	
F = A + B	0F0F	0001	0	0	0F10	
Γ – A + D	FFFF	0001	0	1	0000	
F = A - B - 1	FFFF	0001	0	1	FFFD	
F= A - 1	FFFF	-	0	1	FFFE	
F = A + 1	0F0E	-	1	0	0F0F	
F = A + B + 1	FFFF	0001	1	1	0001	
F = A – B	0F0F	0001	1	1	0F0E	
F = 0	-	-	1	0	0000	

To Test the Assignment

F=Logic shift right A

F=Rotate right A

F=0000

0F0F

0F0F

Operation	A B		Cin		Cout		F				
F = A	0F0F	- 0 0			0F0F						
F = A + B			0F0F		0001	0		0		0F10	
			FFFF		0001	0		1		0000	
F = A - B - 1			FFFF 0001 0 1			FFFD					
F= A – 1			FFFF		-	0		1		FFFE	
F = A + 1			0F0E		-	1		0		0F0F	
F = A + B + 1			FFFF		0001	1		1		0001	
F = A – B			0F0F	0F0F 0001 1		1		1		0F0E	
F = 0			-		-	1		0		0000	
Operation	A	В	F	Op	eration		A	1	Ci	n	F
AND	0F0F	000A	000A	F=Rotate right A with Carry			0F	0F	0		0787
OR	0F0F	000A	0F0F	F=Rotate right A with Carry			0F	0F	1		8787
XOR	0F0F	000A	0F05	F=Logic shift left A			0F	0F	_		1E1E
NOT	0F0F		F0F0	F=Rotate left A		F01	F0			E1E1	

0787

8787

0000

F=Rotate left A with Carry

F=Rotate left A with Carry

F=Arithmatic shift right A