

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)
 - o 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

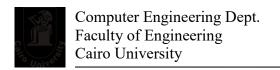
| | S₃ | S ₂ | S ₁ | S ₀ | Cin =0 | Cin = 1 | |
|--------|----|----------------|----------------|----------------|-----------------|---------------|--|
| Part A | 0 | 0 | 0 | 0 | F = A F = A + 1 | | |
| | 0 | 0 | 0 | 1 | F = A + B | F = A + B + 1 | |
| | 0 | 0 | 1 | 0 | F = A - B - 1 | F = A - B | |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

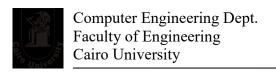


To test part A use the following table, let n = 8

| Operation | A | В | Cin | F | Cout |
|---------------|----|------|-----|------|------|
| F = A | 0F | - | 0 | 0F | 0 |
| F = A + B | 0F | 0001 | 0 | 10 | 0 |
| Γ – A + D | FF | 0001 | 0 | 00 | 1 |
| F = A - B - 1 | FF | 0001 | 0 | FD | 1 |
| F= A – 1 | FF | - | 0 | FE | 1 |
| F = A + 1 | 0E | - | 1 | 0F | 0 |
| F = A + B + 1 | FF | 0001 | 1 | 01 | 1 |
| F = A – B | 0F | 0001 | 1 | 0E | 1 |
| F = 0 | F0 | - | 1 | 0000 | 0 |

To Test the Assignment

| Operation | A | В | Cin | F | Cout |
|--|----|------|-----|--------------|------|
| F = A | 0F | - | 0 | 0F | 0 |
| $\mathbf{F} = \mathbf{A} + \mathbf{B}$ | 0F | 0001 | 0 | 10 | 0 |
| $\mathbf{F} = \mathbf{A} + \mathbf{B}$ | FF | 0001 | 0 | 00 | 1 |
| $\mathbf{F} = \mathbf{A} - \mathbf{B} - 1$ | FF | 0001 | 0 | FD | 1 |
| F=A-1 | FF | - | 0 | FE | 1 |
| $\mathbf{F} = \mathbf{A} + 1$ | 0E | - | 1 | 0F | 0 |
| $\mathbf{F} = \mathbf{A} + \mathbf{B} + 1$ | FF | 0001 | 1 | 01 | 1 |
| $\mathbf{F} = \mathbf{A} - \mathbf{B}$ | 0F | 0001 | 1 | 0E | 1 |
| $\mathbf{F} = 0$ | F0 | - | 1 | 00 | 0 |
| Operation | A | В | Cin | \mathbf{F} | Cout |
| AND | F5 | AA | - | A0 | |
| OR | F5 | AA | - | FF | |
| NOR | F5 | AA | - | 00 | |
| NOT | F5 | - | - | 0A | |
| F=Logic shift right A | F5 | _ | - | 7A | 1 |
| F=Rotate right A | F5 | _ | - | FA | 1 |
| F=Rotate right A with Carry | F5 | | 0 | 7A | 1 |
| F=Rotate right A with Carry | F5 | | 1 | FA | 1 |
| F=Arithmetic shift right A | F5 | | _ | FA | 1 |
| F=Logic shift left A | F5 | | _ | EA | 1 |
| F=Rotate left A | F5 | | _ | EB | 1 |
| F=Rotate left A with Carry | F5 | | 0 | EA | 1 |
| F=Rotate left A with Carry | F5 | | 1 | EB | 1 |
| F=0000 | F5 | | _ | 00 | 0 |
| F=Rotate right A | 7A | | _ | 3D | 0 |



Architecture -Lab Lab #2 3