CSE 306

Computer Architecture Sessional

Assignment on **Floating Point Adder**

Lab Group : B1
Group No : 06
Group Members :
1705077
1705085
1705087
1705088
1705089

May 30, 2021

Contents

1 Introduction

In this task,we have designed a floating point adder circuit that can take floating point inputs which can be either positive or negative and will generate their sum as output.The output will be in form of a floating point too.

In order to implement this circuit,we compared the exponents of the given inputs and right shifted the smaller value to match its exponent with the larger one.Then we performed the addition operation and checked if the output result is in normalized form.We also checked overflow/underflow of the exponent.If not then we normalize the result.

We will see this process in details in the following sections.

2 Problem Specification

There will be two floating point inputs in this circuit.Input will be provided in IEEE 754 Standard format of binary representation of the floating point number (sign bit, exponent bit, fraction bits).There will be 1 bit for the sign,4 bits for the exponent and 11 bits for the fraction.

i Two 16 bit inputs A and B

The output will be in the same format too.

3 Flowchart

Flowchart of the addition algorithm is given below.

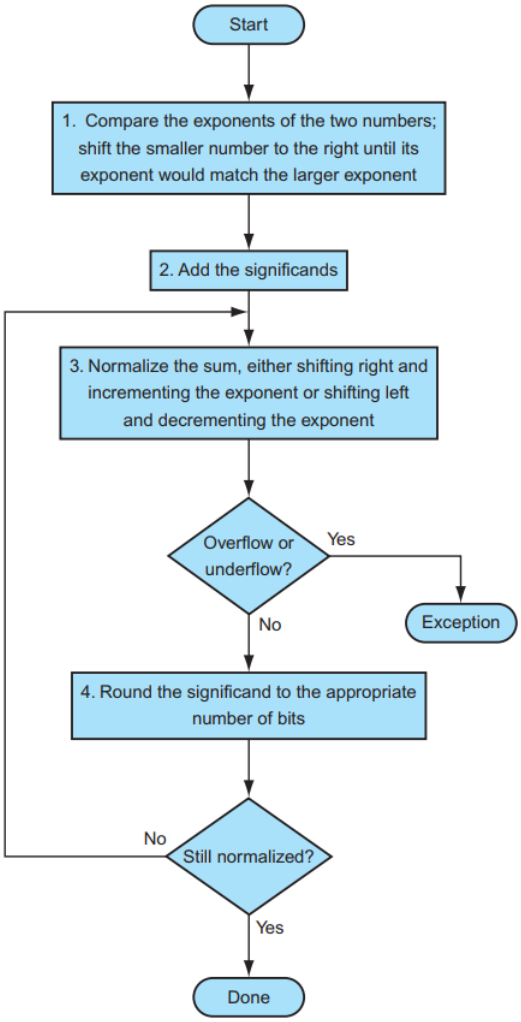

Figure 1: Floating point addition

4 High-level Block Diagram

The high level block diagram of the architecture

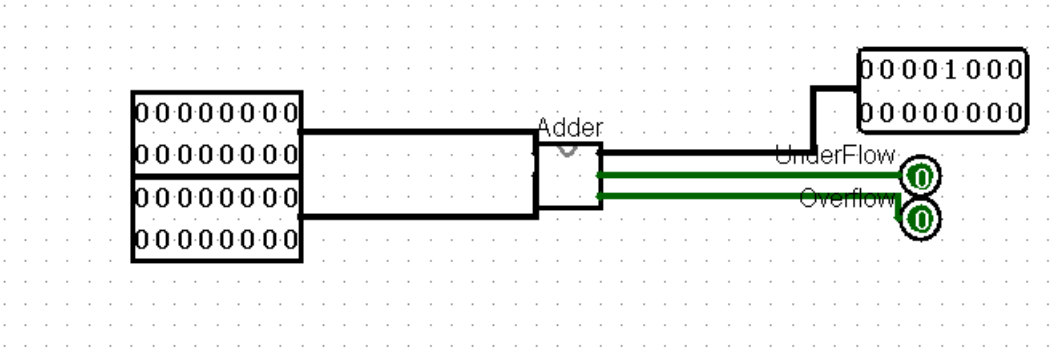

Figure 2: Block diagram of floating point adder.

5 Circuit Diagram

Detailed circuit diagram of important blocks are given below.

1. Preprocessor
    - Bit splitter

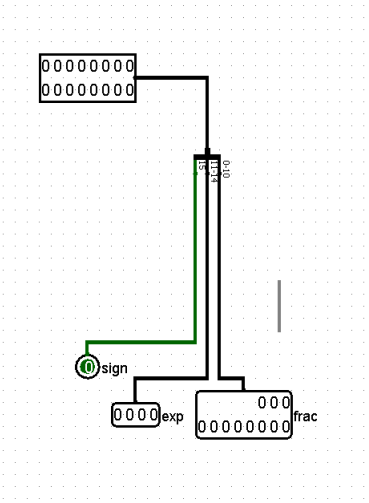

Figure 3: Diagram of bit splitter.

- Comparer

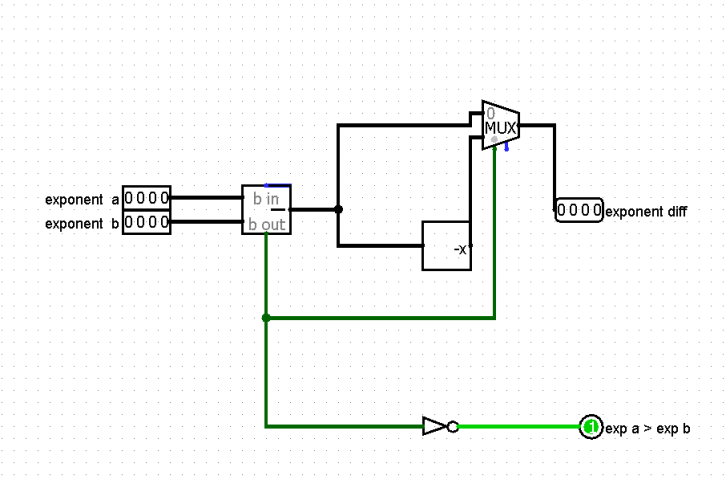

Figure 4: Diagram of comparer.

- Fraction shifter

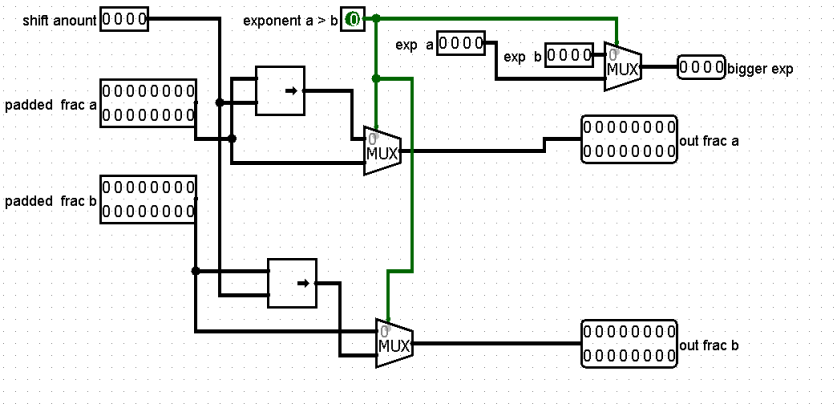

Figure 5: Diagram of fraction shifter.

- Padder

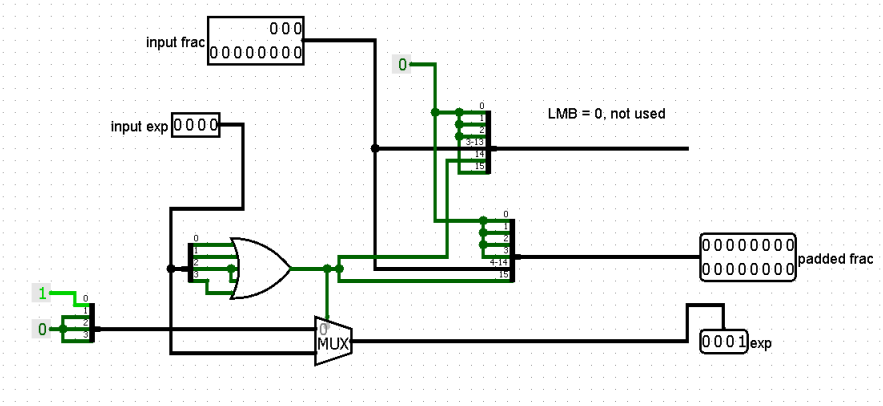

Figure 6: Diagram of padder.

- Fp preprocessor

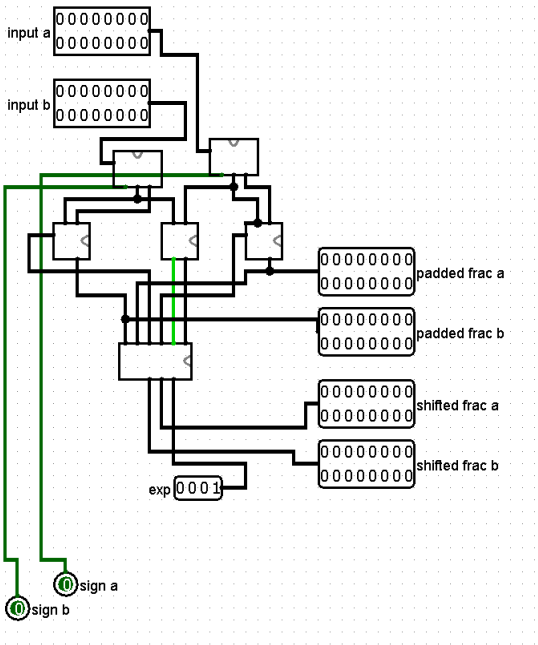

Figure 7: Diagram of fp preprocessor.

2. Adder

- 2s complement

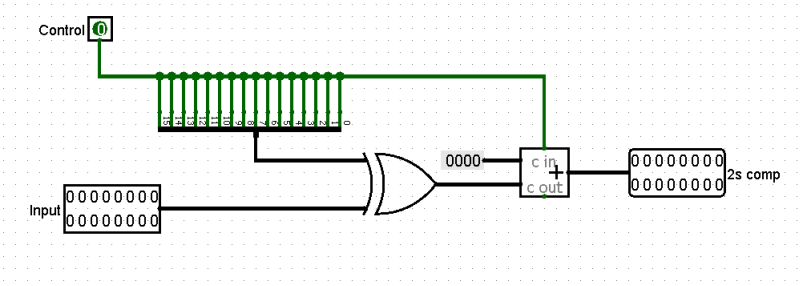

Figure 8: Diagram of 2s complement.

- IEEE adder

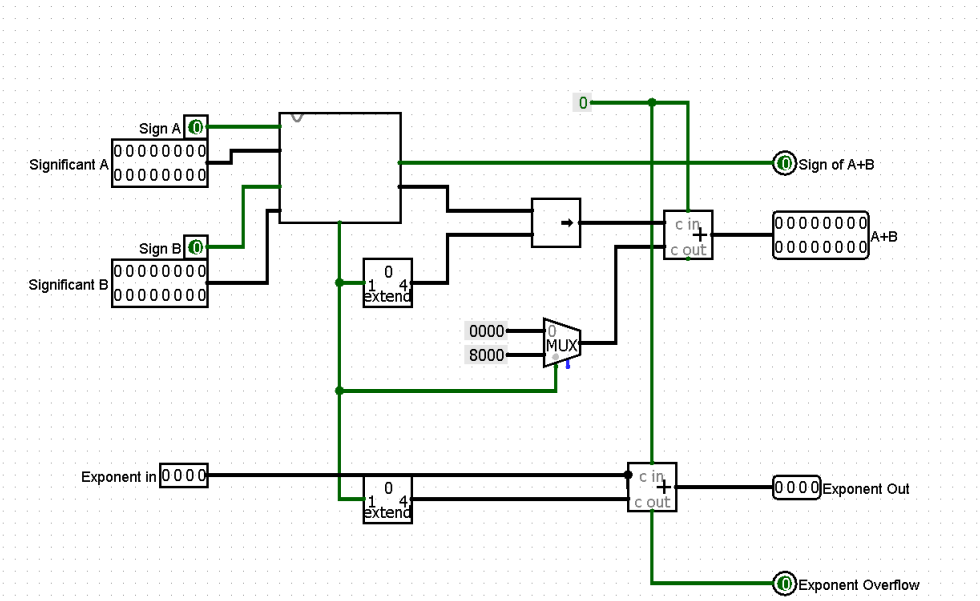

Figure 9: Diagram of IEEE adder.

- Magnitude comparator

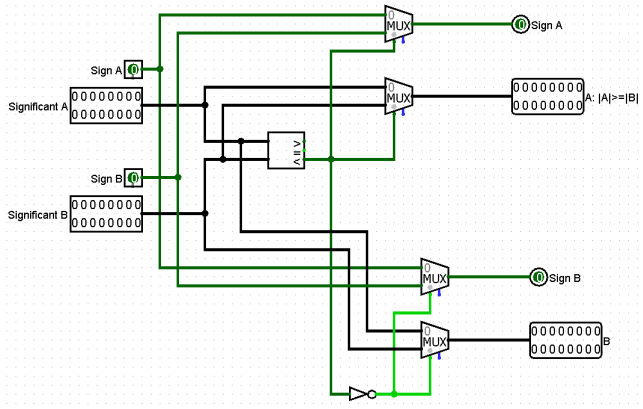

Figure 10: Diagram of magnitude comparator.

- Significant adder

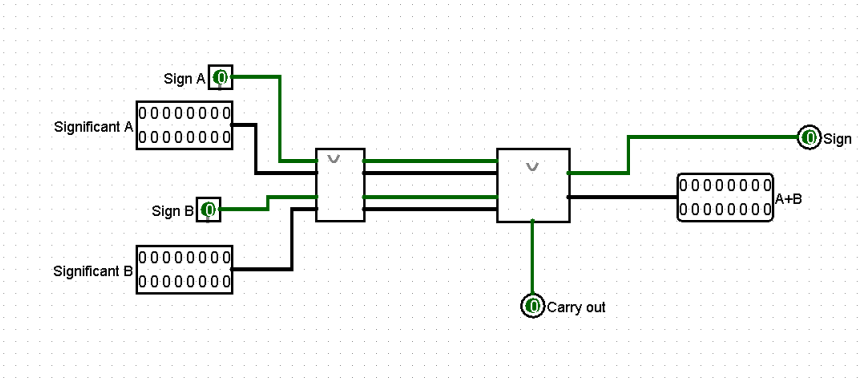

Figure 11: Diagram of significant adder.

- Special adder

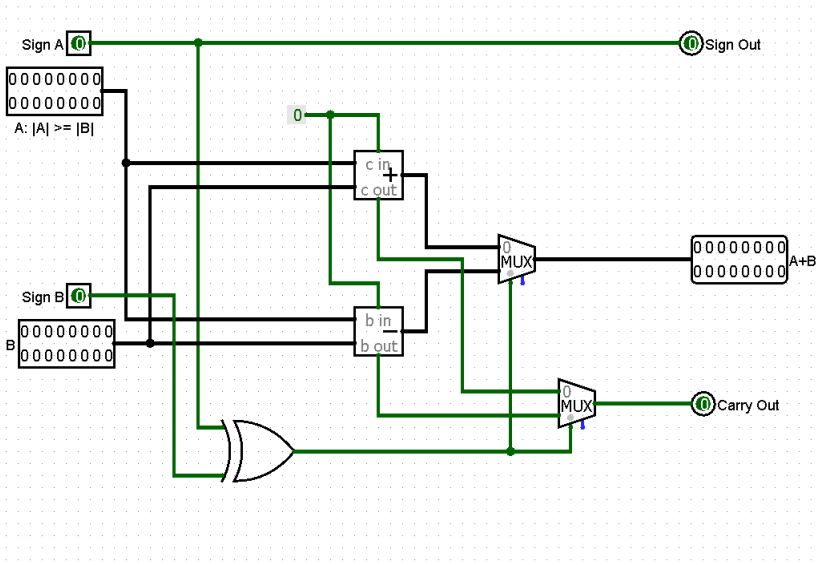

Figure 12: Diagram of special adder.

- Adder

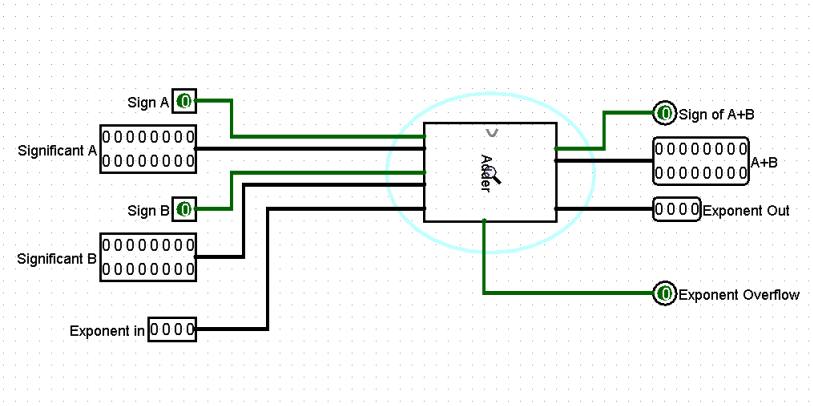

Figure 13: Diagram of adder.

3. Final circuit

- Normalizer

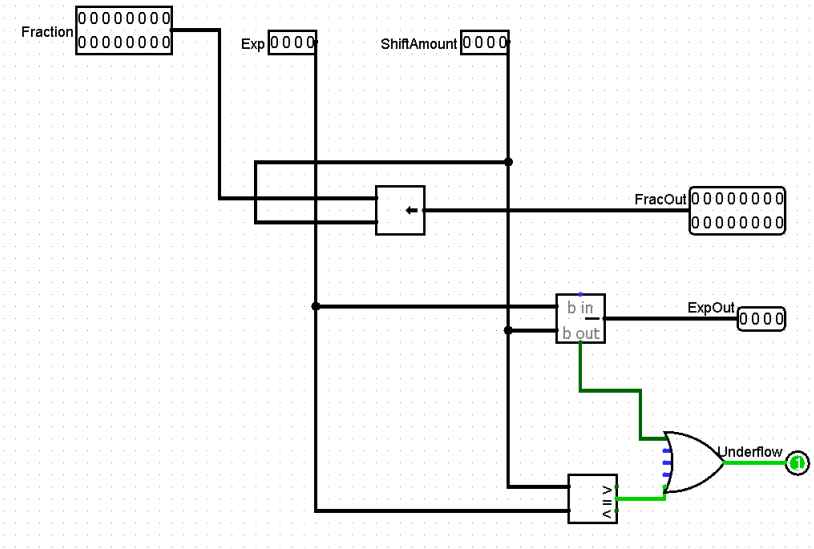

Figure 14: Diagram of normalizer.

- Shift amount

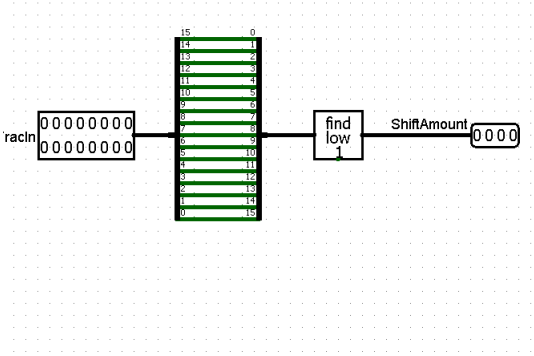

Figure 15: Diagram of shift amount.

- Truncate

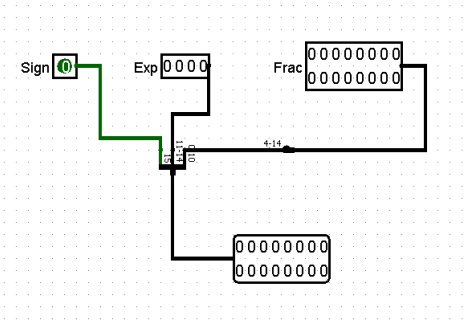

Figure 16: Diagram of truncate.

- Final adder

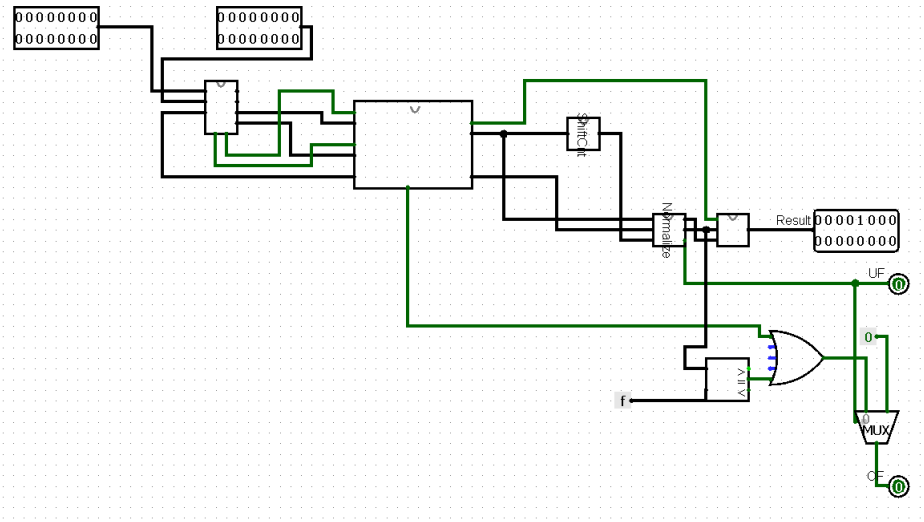

Figure 17: Diagram of final adder.

6 Used ICs

Types and counts of gates used for designing this circuit:

Operation	Special Adder	Significant Adder	2s complement	Magnitude comparator	IEEE adder	Compare r	Padder	fp preprocessor	Bit splitter	Frac shifter	Normalizer	Shift amount	Truncate	Final Adder	IC no
QUAD 2-XOR	1	1	1											1	7486
QUAD 2-OR							1	2			1			4	7432
NOT		1		1	1	1		1						1	7404
16 bit Adder	1	1	1		2									2	
4 bit Adder					1									1	7483
16 bit Subtractor	1														
4 bit Subtractor						1		1			1			2	
Quad 16:1 MUX	1	3		2	4			2		2				6	74150
Dual 4:1 MUX						1	1	2		1				2	74153
Quad 2:1 MUX	1	1		1	1									1	74157
16 to 1 bit Splitter			1				2	4				2		4	
16 to 3 bit Splitter								2	1					2	
4 to 1 bit							1	2						2	

Splitter															
3 to 1 bit Splitter													1	1	
16 bit comparator		1		1	1									1	
4 bit comparator											1			2	7485
16 bit Shifter					1			2		2	1			4	
1 to 4 bit extender					2									2	
16 bit Bit finder												1		1	
4 bit Negator						1		1						1	
														Total = 40	

7 About the simulator

The simulator used for making this design is Logisim 2.7.1. It is a free software for simulating digital logic circuits. It was downloaded from `https://sourceforge.net/projects/circuit/`.

8 Discussion

Here we checked overflow and underflow of the output.To get a normalized form,we truncated the output.To make the build up easier,We broke the problem into smaller problems and built circuit for that.And finally we combined everything to make the final circuit.