



**FUNDAMENDTAL OF DIGITAL SYSTEM FINAL PROJECT REPORT  
DEPARTMENT OF ELECTRICAL ENGINEERING  
UNIVERSITAS INDONESIA**

# **Implementasi Penjumlahan Angka Floating Point FPGA**

## **GROUP BP04**

<b>Edgrant Henderson Suryajaya</b>	<b>2206025016</b>
<b>Muhammad Nadhif Fasichul Ilmi</b>	<b>2206813416</b>
<b>Adhelia Putri Maylani</b>	<b>2206814816</b>
<b>Xavier Daniswara</b>	<b>2206030230</b>

## TABLE OF CONTENTS

<b>CHAPTER 1: INRODUCTION .....</b>	<b>1</b>
1.1 Background .....	2
1.2 Project Description.....	2
1.3 Objectives .....	2
1.4 Roles and Responsibilities .....	2
<b>CHAPTER 2: IMPLEMENTATION .....</b>	<b>4</b>
2.1 Equipment.....	5
2.2 Implementation .....	5
<b>CHAPTER 3: TESTING AND ANALYSIS .....</b>	<b>4</b>
3.1 Testing.....	5
3.2 Result .....	5
3.3 Analysis.....	5
<b>CHAPTER 4: CONCLUSION.....</b>	<b>4</b>
<b>REFERENCES.....</b>	<b>4</b>
<b>APPENDICES .....</b>	<b>4</b>
Appendix A: Project Schematic .....	5
Appendix B: Documentation.....	5

## INTRODUCTION

### 1.1 BACKGROUND

Angka floating point adalah representasi angka real dengan menggunakan biner. Representasi ini menggunakan sistem seperti notasi scientific, yaitu dengan mengelompokkan angka menjadi dua bagian, yaitu tanda dan mantissa. Tanda menunjukkan apakah angka tersebut positif atau negatif, sedangkan mantissa menunjukkan nilai sebenarnya dari angka tersebut.

Angka floating point distandarisasi oleh IEEE 754, dengan format sebagai berikut:

- 1 bit untuk tanda
- 8 bit untuk eksponen
- 23 bit (24 implisit) untuk mantissa

Mantissa selalu bersifat positif, kenegatifan mantissa diatur pada bit tanda. Exponent dapat berupa negative. Namun, negativitasnya tidak bersistem 2's complement tetapi dengan 0111(2) / 7(10) sebagai 0. Jadi 0 adalah  $2^7$  atau 127. Jika ingin dicari exponent aslinya, exponent pada float harus dikurangi 127.

Penjumlahan angka floating point dapat dilakukan dengan cara berikut:

- Dekomposisi masing-masing angka menjadi tanda, eksponen, dan mantissa.
- Sesuaikan eksponen kedua angka agar sama.
- Tambahkan mantissa kedua angka tersebut.
- Jika hasil penjumlahan mantissa lebih besar dari 1, maka normalisasi hasil dengan menggeser mantissa ke kanan dan menambah eksponen.

### 1.2 PROJECT DESCRIPTION

Proyek ini bertujuan untuk merancang dan mengimplementasikan penjumlah angka floating point berkinerja tinggi pada FPGA. Aritmatika floating point merupakan tulang punggung dari berbagai aplikasi yang membutuhkan presisi tinggi, mulai dari komputasi ilmiah hingga pemodelan keuangan dan pemrosesan gambar. FPGA dapat melakukan operasi floating-point secara cepat, yang dapat meningkatkan kecepatan operasi secara signifikan. Selain itu, FPGA dapat dikonfigurasi untuk memenuhi kebutuhan spesifik aplikasi, yang dapat meningkatkan efisiensi operasi.

### 1.3 OBJECTIVES

The objectives of this project are as follows:

1. Implementasi penjumlah Full Adder untuk penambahan floating-point efisien menggunakan gaya dataflow.
2. Implementasi decoder sinyal dan alamat menggunakan gaya behavioral.
3. Implementasi testbench untuk semua modul menggunakan gaya behavioral.
4. Implementasi penjumlah integer penuh menggunakan gaya struktural.
5. Implementasi shifting menggunakan loop untuk fleksibilitas dan pembacaan kode.
6. Implementasi shifting menggunakan fungsi untuk modularitas dan penggunaan kembali kode.
7. Implementasi FSM mewakili alur kontrol tingkat atas.
8. Implementasi microcomputing dengan format control word untuk komputasi aritmatika.

### 1.4 ROLES AND RESPONSIBILITIES

The roles and responsibilities assigned to the group members are as follows:

Responsibilities	Person
Membuat Float Adder Membuat Laporan	Edgrant Henderson S.
Membuat AdderInteger Melakukan Sintesis Quartus	M. Nadhif Fasichul Ilmi
Membuat Seluruh Dokumentasi (Readme, PPT)	Adhelia Putri Maylani
Membuat Testbench Melakukan Simulasi Modelsim	Xavier Daniswara

Table 1. Roles and Responsibilities

## **CHAPTER 2**

### **IMPLEMENTATION**

#### **2.1 EQUIPMENT**

The tools that are going to be used in this project are as follows:

- ModelSim
- Git
- GitHub
- VHDL
- Quartus
- Visual Studio Code

#### **2.2 IMPLEMENTATION**

##### **KOMPONEN**

Penjumlah (Async)

- Bertanggung jawab atas operasi penjumlahan asynchronous pada angka floating-point.
- Desain asynchronous memungkinkan proses komputasi lebih cepat.
- Membentuk pasangan mantissa yang sesuai dan menangani eksponen untuk hasil penjumlahan akurat.

Penjumlah Float (Sync)

- Inti operasi penjumlahan floating-point pada FPGA.
- Desain synchronous memungkinkan integrasi dengan arsitektur FPGA dan sinkronisasi operasi dengan clock sistem.
- Meningkatkan koordinasi dan keandalan operasi.

### Decoder (Async)

- Berfungsi sebagai decoder sinyal kontrol dan alamat, mengarahkan input ke komponen yang sesuai.
- Desain asynchronous memungkinkan respons cepat terhadap perubahan status, memastikan koordinasi yang baik antar komponen.

### Top-Level

- Kumpulan dari komponen sebelumnya, membentuk arsitektur utama penjumlah floating point pada FPGA.
- Komunikasi antar komponen diatur menggunakan sinyal kontrol yang dikelola decoder.
- Mempertimbangkan pemilihan jenis FPGA yang mendukung kecepatan dan kinerja yang diinginkan.

## **IMPLEMENTASI DALAM SETIAP MODUL**

### Dataflow:

- Digunakan dalam komponen adder untuk memanfaatkan paralelisme dan mengoptimalkan pemanfaatan sumber daya.
- Memungkinkan komputasi terus-menerus segera setelah input yang diperlukan tersedia.

### Behavioral:

- Digunakan dalam komponen Decoder Async dan Testbench untuk fokus pada fungsionalitas daripada struktur internal.
- Memberikan fleksibilitas dan kesederhanaan dalam operasi.

### Struktural:

- Digunakan dalam komponen Structural Style untuk meningkatkan modularitas dan pemeliharaan dengan mendefinisikan struktur penjumlah secara eksplisit.

### Looping:

- Digunakan dalam komponen shifting untuk fleksibilitas dan pembacaan kode yang lebih mudah. Juga digunakan dalam adder integer untuk menghitung sum.
- Mempermudah tugas berulang dan meningkatkan pemahaman kode.

#### Function:

- Digunakan dalam komponen Function untuk modularitas dan penggunaan kembali kode.
- Meningkatkan pemeliharaan dan pembacaan kode dengan mengisolasi operasi shifting ke dalam fungsi terpisah.

#### FSM (Finite State Machine):

- Digunakan dalam komponen FSM untuk mewakili alur kontrol tingkat atas.
- Memandu operasi keseluruhan dengan bertransisi di antara state yang berbeda.
- Setiap state mewakili tahap tertentu dalam proses penjumlahan floating-point, menyediakan kontrol dan koordinasi.

#### Microcomputing:

- Digunakan dalam komponen Microcomputing dengan format control word untuk komputasi aritmatika.
- Memungkinkan spesifikasi operasi dan operand secara ringkas.
- Control word memandu unit microcomputing untuk melakukan penambahan signed atau unsigned dengan operand yang ditentukan.

## CHAPTER 3

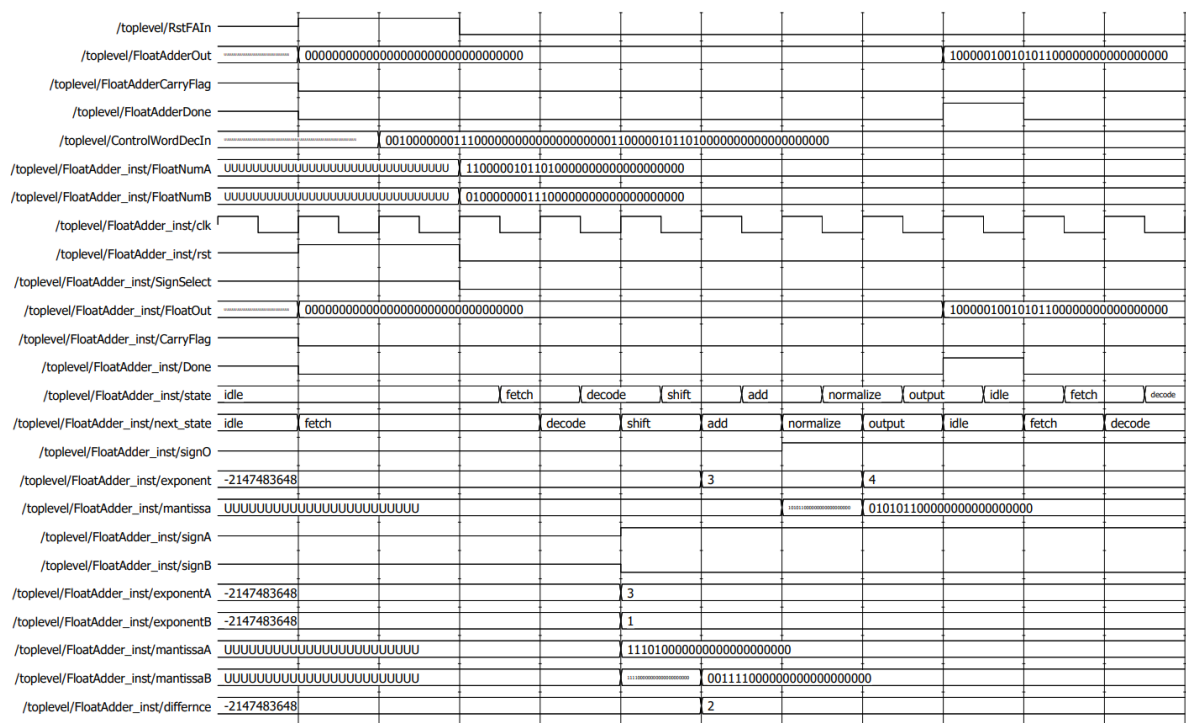
### TESTING AND ANALYSIS

#### 3.1 TESTING

Setelah proses desain dan implementasi selesai, langkah selanjutnya adalah pengujian. Pengujian dilakukan untuk memastikan bahwa proyek telah memenuhi persyaratan fungsional dan kinerja yang diinginkan.

Dalam proyek ini, pengujian dilakukan menggunakan dua metode, yaitu wave test dan synthesize test. Wave test digunakan untuk memeriksa kebenaran sinyal pada setiap tahap proses penjumlahan floating point. Synthesize test digunakan untuk memeriksa kinerja penjumlah dengan berbagai input.

#### 3.2 RESULT



Pengujian terhadap proyek FPGA Floating Point Adder menggunakan wave telah berhasil dilaksanakan. Analisis gelombang sinyal pada setiap tahap proses penjumlahan menunjukkan kebenaran behavior sesuai logika yang dirancang. Tidak ada anomali atau ketidaksesuaian yang terdeteksi dalam pengoperasian internal.



Pemasukan berbagai input dan kondisi menghasilkan nilai keluaran yang akurat dan konsisten dengan perhitungan manual. Penjumlahan bilangan floating point dengan presisi tinggi berhasil dicapai.

### **3.3 ANALYSIS**

Rangkaian sudah berjalan dengan baik dengan output dari rangkaian berhasil merupai perhitungan manual. Namun, terdapat beberapa masalah dengan rangkaian, salah satunya adalah tidak implementasinya integer adder. Pada rangkaian sekarang, tidak digunakan integer adder dan hanya menggunakan penjumlahan yang disediakan library math. Hal ini karena ketidaksesuaian output dan input oleh component integer adder yang menyebabkan bit bertipe 'X'.

Selain permasalahan integer adder, proyek FPGA Floating Point Adder telah berhasil diselesaikan dengan baik. Proyek ini telah memenuhi tujuan desain dan implementasi, yaitu mencapai fungsionalitas yang diharapkan, dapat melakukan penjumlahan bilangan floating point dengan presisi tinggi. Semua modul juga diimplementasikan dengan baik.

Secara keseluruhan, proyek FPGA Floating Point Adder ini telah berhasil diselesaikan dengan baik. Proyek ini memiliki potensi untuk diterapkan dalam berbagai aplikasi yang membutuhkan operasi penjumlahan floating point dengan presisi tinggi.

## **CHAPTER 4**

### **CONCLUSION**

Proyek FPGA Floating Point Adder sukses dibangun! Desain FPGA mencapai penjumlahan bilangan floating point presisi tinggi dengan beragam teknik seperti dataflow dan FSM. Pengujian menunjukkan kinerja baik, meski terdapat kendala pada integer adder yang memerlukan perbaikan. Secara keseluruhan, proyek ini berhasil untuk aplikasi penjumlahan floating point.

## REFERENCES

- [1] All of the Digital Laboratory Resources
- [2] GeeksforGeeks. (2020, March 16). *IEEE Standard 754 floating point numbers*.  
<https://www.geeksforgeeks.org/ieee-standard-754-floating-point-numbers/>

## APPENDICES

### Appendix A: Project Schematic

Put your final project latest schematic here

### **Appendix B: Documentation**

Put the documentation (photos) during the making of the project