

**Structura Sistemelor de Calcul**

**Unitate de calcul în virgulă mobilă: Înmulțirea**

Studenți: Velicea Andreea-Ioana și Pepelea Ioana-Adina

Profesor îndrumător: Lișman Dragoș Florin

Grupa: 30238

Facultatea de Automatică și Calculatoare

15 Ianuarie 2024

Table of Contents

[1. Introducere 3](#_Toc156217729)

[2. Fundamentare teoretică 4](#_Toc156217730)

[2.1. Tehnologii utilizate 4](#_Toc156217731)

[2.1.1. Proiectare structurală utilizând limbajul VHDL 4](#_Toc156217732)

[2.1.2. Implementarea pe o placă Basys 3 4](#_Toc156217733)

[2.1.3. Reprezentarea numerelor în virgulă mobilă, conform standardului IEEE 754 4](#_Toc156217734)

[2.1.4. Înmulțirea numerelor în virgulă mobilă 5](#_Toc156217735)

[2.1.5. Utilizarea extensiei PmodKYBD 7](#_Toc156217736)

[3. Proiectare și implementare 7](#_Toc156217737)

[3.1. Implementare 7](#_Toc156217738)

[3.2. Schemă bloc 8](#_Toc156217739)

[3.3. Diagrama de stare 9](#_Toc156217740)

[4. Rezultate experimentale 9](#_Toc156217741)

[4.1. Manual de utilizare 10](#_Toc156217742)

[5. Concluzii 10](#_Toc156217743)

[6. Bibliografie 10](#_Toc156217744)

# Introducere

Proiectul propus se concentrează asupra operațiilor de înmulțire a numerelor reprezentate conform standardului IEEE 754 pentru virgulă mobilă. Unitățile centrale de procesare (UCP) utilizează în principal două tipuri de numere: virgulă fixă și virgulă mobilă. Diversitatea acestor reprezentări este necesară pentru a gestiona informații care nu pot fi exprimate în mod complet ca numere întregi, asigurând totodată portabilitatea programelor între calculatoare.

Pentru rezolvarea problemelor asociate reprezentării în virgulă fixă, s-a propus utilizarea reprezentării în virgulă mobilă cu scalare automată. Această abordare implică includerea factorului de scală în cuvântul de calculator, iar poziția virgulei binare se ajustează automat pentru fiecare număr. O reprezentare în virgulă mobilă constă într-o mantisă (M) și un exponent (E), stocate într-un cuvânt cu trei câmpuri: semnul, mantisa și exponentul.

Există diverse modalități de implementare a reprezentării în virgulă mobilă, cu diferite abordări privind tratamentul cazurilor de excepție, alocarea numărului de biți pentru fiecare câmp, modul de rotunjire etc. Aceasta poate genera probleme de portabilitate, deoarece calculatoarele cu reprezentări diferite pot interpreta aceleași date în moduri distincte.

Pentru a asigura portabilitatea programelor, IEEE a dezvoltat un standard pentru reprezentarea numerelor în virgulă mobilă și operațiile aritmetice asociate. Standardul impune o metodă de calcul coerentă, garantând același rezultat indiferent de implementarea hardware, software sau o combinație a ambelor. Este esențial ca erorile și condițiile de eroare generate în operațiile matematice să fie raportate în mod consecvent, independent de modul de implementare.

Una dintre provocările în reprezentarea în virgulă mobilă este gestionarea depășirilor superioare și inferioare. Depășirea superioară apare atunci când exponentul depășește valoarea maximă, în timp ce depășirea inferioară apare când exponentul are o valoare sub pragul minim.

Soluția propusă pentru aceste probleme constă în implementarea corectă a reprezentării numerelor conform standardului IEEE 754 pe 32 de biți, algoritmului corespunzător înmulțirii numerelor în virgulă mobilă și aplicarea unei operații de rotunjire pentru obținerea unor rezultate precise. Scopul proiectului este de a asigura o implementare corectă și eficientă a acestor operații în conformitate cu standardele stabilite.

# 2. Fundamentare teoretică

## 2.1. Tehnologii utilizate

### 2.1.1. Proiectare structurală utilizând limbajul VHDL

VHDL reprezintă un limbaj de descriere a hardware-ului (Hardware Description Language -HDL), destinat descrierii comportamentului și arhitecturii unui "modul" electronic logic, cu alte cuvinte al unei funcțiuni logice combinatorii și secvențiale. Alături de Verilog, este cel mai utilizat limbaj de proiectare a sistemelor electronice digitale. E una din uneltele principale pentru proiectarea circuitelor integrate moderne, aplicat cu succes în câmpul microprocesoarelor, în telecomunicații, automobile și altele. Este utilizat în concepția asistată pe calculator (CAD) a circuitelor integrate sau pentru configurarea FPGA -urilor.

### 2.1.2. Implementarea pe o placă Basys 3

Placa FPGA Basys 3 reprezintă o platformă de dezvoltare hardware ce încorporează un cip FPGA Spartan-6 de la Xilinx și furnizează un mediu flexibil pentru prototiparea sistemelor digitale. Dotată cu o varietate de interfețe și componente, printre care butoane, LED-uri, porturi USB și conectori extensibili, această placă este potrivită pentru înțelegerea conceptelor de proiectare digitală și pentru dezvoltarea de prototipuri. Cu suportul software oferit de Vivado Design Suite, este adecvată pentru studenți, pasionați și profesioniști care doresc să exploreze și să învețe în domeniul electronicelor digitale.

### 2.1.3. Reprezentarea numerelor în virgulă mobilă, conform standardului IEEE 754

Formatul cu virgulă mobilă în precizie simplă este un format numeric pentru calculatoare, ocupând în mod obișnuit 32 de biți în memoria computerului. El reprezintă o gamă dinamică extinsă de valori numerice prin utilizarea unui punct zecimal flotant.

Un număr reprezentat în virgulă mobilă are două componente. Prima componentă este mantisa (M), care indică valoarea exactă a numărului într-un anumit domeniu, fiind reprezentată de obicei ca un număr fracţionar cu semn. A doua componentă este exponentul (E), care indică ordinul de mărime al numărului. Această reprezentare pe 32 de biți poate fi memorată într-un cuvânt binar cu trei câmpuri: semnul, mantisa şi exponentul, ca în figura de mai jos:



Aceasta este o reprezentare în mărime și semn, întrucât semnul are un câmp separat pentru reprezentare. Acest câmp este 0 pentru numerele pozitive și 1 pentru cele negative. Numerele reprezentate în acest format sunt normalizate, adică au bitul cel mai semnificativ al mantisei egal cu 1.

In acest format apare noțiunea de bit ascuns. Acesta se referă la fixarea unei valori pentru bitul cel mai semnificativ al mantisei prin intermediul formatului de reprezentare ,valoare ce nu poate fi modificată, motiv pentru care acest bit nu mai este reprezentat.

### 2.1.4. Înmulțirea numerelor în virgulă mobilă

În privința algoritmului de înmulțire a numerelor în virgulă mobile, acesta urmărește pașii următori:

-Se convertesc numerele x și y în formatul IEEE

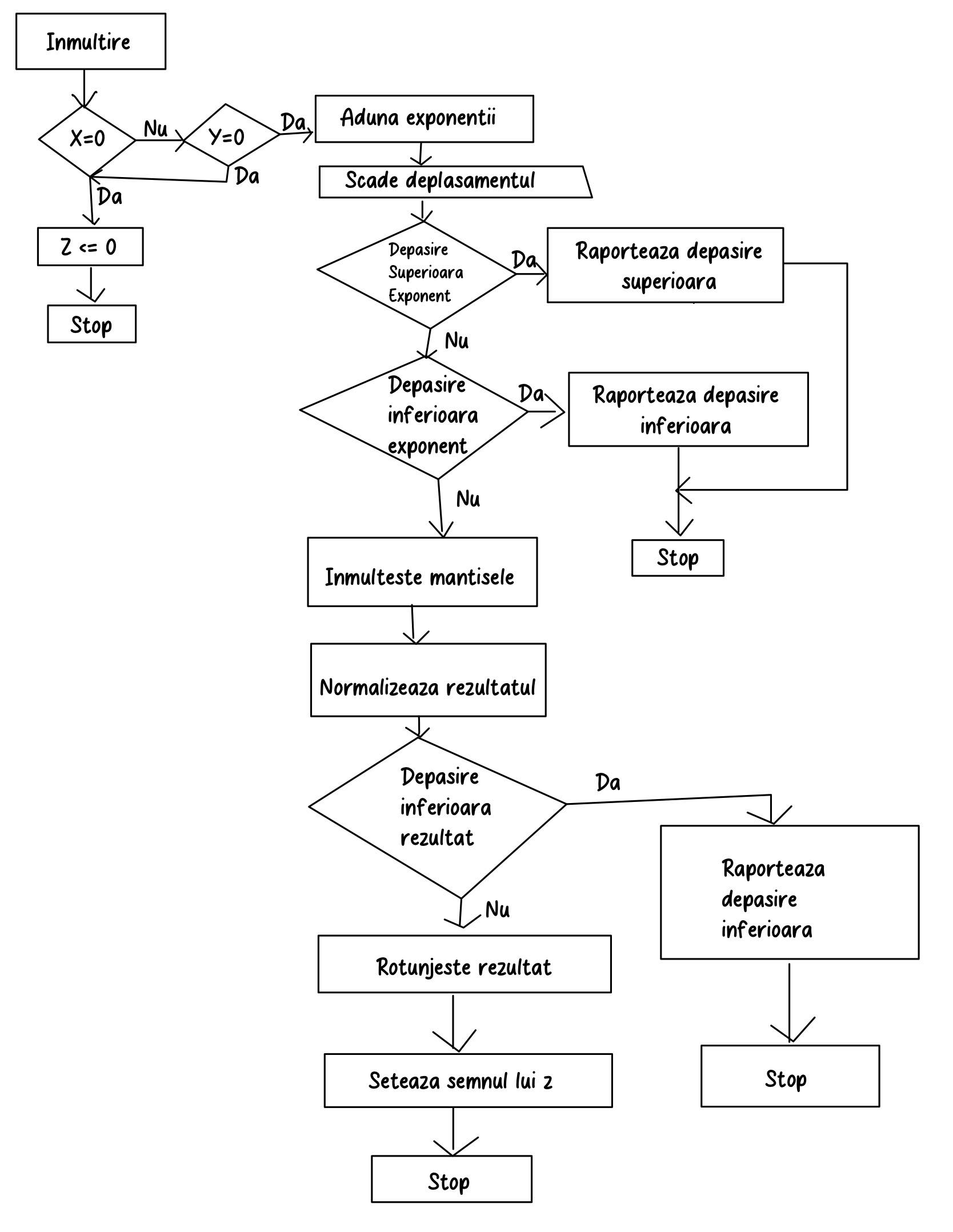
-Exponentul rezultat se calculează ca suma exponențiilor numerelor x și y: z\_exp=x\_exp+y\_exp

-Mantisa rezultată se calculează înmulțind mantisa lui x cu mantisa lui y: z\_mant=x\_mant\*y\_mant

-In cazul in care este nevoie, ajustăm exponentul

-Pentru a determina semnul lui z, facem xor ăntre semnul lui x si semnul lui y: z\_semn=x\_semn xor y\_semn

-Se convertește rezultatul înapoi.



### 2.1.5. Utilizarea extensiei PmodKYBD

PmodKYPD utilizează 4 rânduri și 4 coloane pentru a crea o matrice de 16 butoane de tip pushbutton. Am utilizat aceasta tastatură pentru ca utilizatorul să poată introduce apasând butoanele acesteia, numerele pe care vrea să le înmulțească.

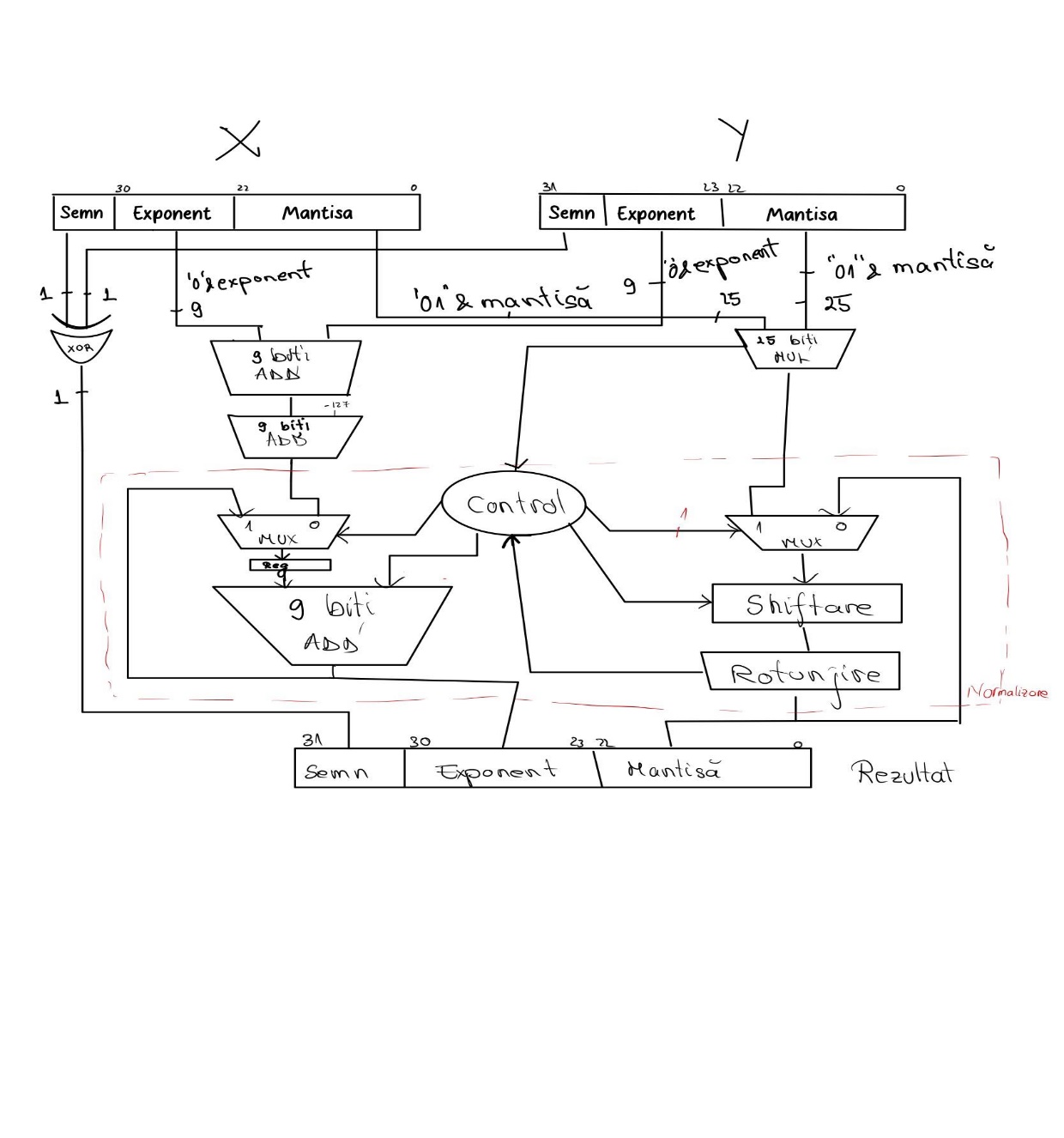
# Proiectare și implementare

## Implementare

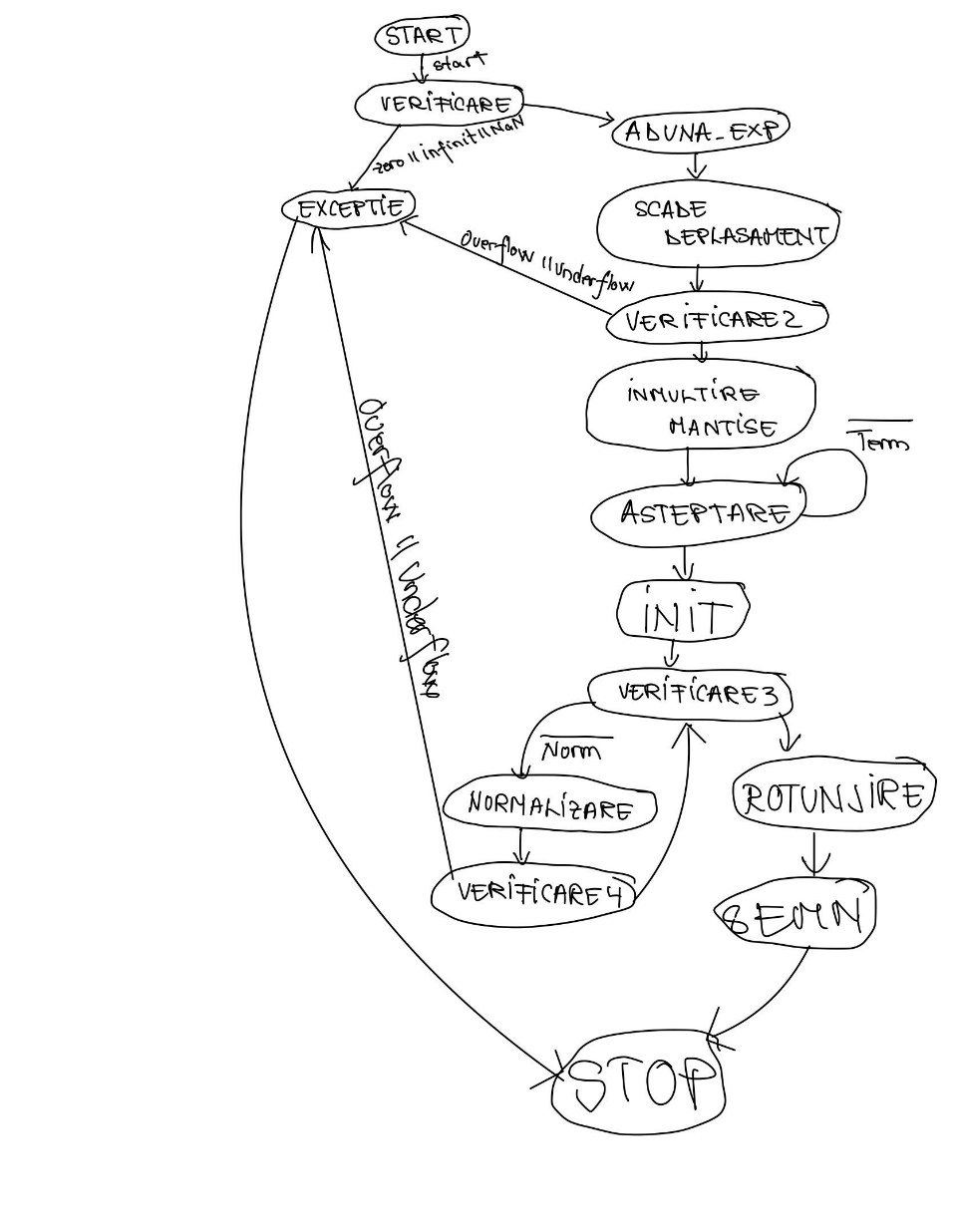
Implementarea va consta în crearea mai multor componente:

* Componenta de înmulțire care conține înmulțirea Booth pe 25 de biți, unde sunt înmulțite mantisele numerelor introduse
* Componenta de adunare, care conține partea de adunare din structura proiectului
* Registrul, pe care îl folosim pentru reținerea temporară a sumelor calculate în componenta de adunare
* Componenta de control, cuprinde stările prin care trecem
* Registrul de deplasare, componentă în care se realizează shiftarea mantisei utilizată la normalizare
* Excepții, componenta în care sunt tratate excepțiile
* Keypad decoder, componenta care ne ajută să detectăm valoarea hexazecimală apăsată pe tastatură
* Debouncer, folosit pentru butoanele de shiftare, încărcare în registru și pornirea înmulțirii
* FourBitRegister, registru utilizat la stocarea numerelor introduse de la tastatură.

## Schemă bloc

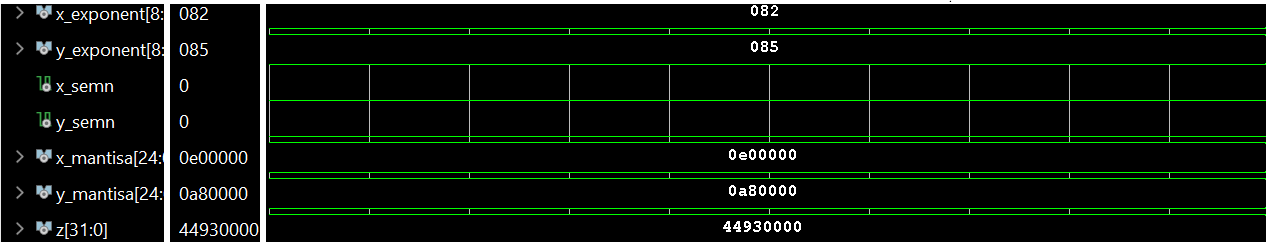
****

## Diagrama de stare

****

# Rezultate experimentale

Din punct de vedere al testării aplicației, s-au realizat teste pe plăcuța Basys 3 împreună cu tastatura PmodKYBD, dar s-au realizat și testbench-uri pentru operatia de înmulțire. Acestea din urmă au fost simulate în Vivado.



## Manual de utilizare

Tot din punctul de Vedere al testării, pentru testarea aplicației pe plăcuța Basys 3, împreună cu tastatura PmodKYBD, se vor folosi urmatoarele switch-uri, led-uri și butoane:

* Sw0 – sw2 ne ajută la vizualizarea pe afișor:
  + “000”- primele 4 valori hexazecimale din primul numar
  + “001”- ultimele 4 valori hexazecimale din primul numar
  + “010”- primele 4 valori hexazecimale din al doilea numar
  + “011”- ultimele 4 valori hexazecimale din al doilea numar
  + “100”- primele 4 valori hexazecimale din rezultat
  + “101”- ultimele 4 valori hexazecimale din rezultat
  + “110”- primele 4 valori hexazecimale din numarul current
  + “111”- ultimele 4 valori hexazecimale din numarul current
* BTNC: Butonul Start, folosit pentru pornirea operatiei
* BTNU: Load\_x, folosit pentru salvarea numarului x
* BTNL: Load\_y, folosit pentru salvarea numarului y
* BTND: Butonul Shift, folosit pentru shiftarea numarului current dupa apasarea fiecarei taste de pe tastatura PmodKYBD
* BTNR: Butonul de Reset, folosit pentru a reseta operatia.

# Concluzii

În concluzie, proiectul dedicate operațiilor de înmulțire în virgulă mobile conform standardului IEEE 754 a avut ca obiectiv implementarea eficientă și corectă a acestor operații. Utilizând limbajul VHDL și beneficiind de platforma Basys 3, am realizat o schema bloc și diagrama de stare coerente. Reprezentarea numerică în virgulă mobile, împreună cu implementarea algoritmului de înmulțire, a fost realizat cu precizie, gestionând și cazurile de depășire. Integrarea tastaturii PmodKYBD a facilitate introfucerea datelor de intrare. Proiectul a contribuit la înțelegerea detaliată a conceptelor teoretice și la dezvoltarea practică a unei soluții conforme cu standardele stabilite.

# Bibliografie

Standardul IEEE 754-32 bits:

* <https://en.wikipedia.org/wiki/Single-precision_floating-point_format>
* <https://www.mimosa.org/ieee-floating-point-format/>
* <https://www.h-schmidt.net/FloatConverter/IEEE754.html>

PmodKYBD:

* <https://digilent.com/reference/pmod/pmodkypd/reference-manual>

Operații în virgulă mobilă:

* <https://users.utcluj.ro/~baruch/media/ssc/curs/SSC-VM.pdf>
* <http://www.csit-sun.pub.ro/courses/Indrumar_CN_New.pdf>