UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA

Unitate de calcul in virgula mobila: Inmultirea

Structura Sistemelor de Calcul

Autori: Darolti Laura si Goron Gesica
Grupa: 30235

Facultatea de Automatica
si Calculatoare

8 Ianuarie 2024

Cuprins

1 Introducere

Tema propusă pentru implementare vizează operații de înmulțire a numerelor reprezentate conform standardului IEEE 754 pentru virgulă mobilă. Majoritatea unităților centrale de procesare operează cu două tipuri de numere: virgulă fixă și virgulă mobilă. Aceasta se datorează faptului că nu toate informațiile necesare pentru operațiile specifice ale UCP-ului sunt în întregime numere întregi. Pentru a asigura portabilitatea programelor între calculatoare diferite, datele sunt reprezentate în mod intern în format virgulă mobilă, conform standardului IEEE 754, utilizat de majoritatea unităților de calcul și a coprocesoarelor matematice.

Multe aplicații necesită numere care nu sunt întregi și există diverse metode pentru reprezentarea lor. Una dintre acestea este reprezentarea în virgulă fixă, unde se poate folosi aritmetica pentru numere întregi, plasând ulterior virgula binară într-o poziție predefinită, cum ar fi după bitul de semn. Această reprezentare necesită transformarea tuturor numerelor în acest format, folosind operații de scalare sau deplasare, atașând numerelor factori de scală. Evidențierea acestor transformări necesită intervenția programului, crescând astfel timpul de calcul.

O soluție pentru problemele reprezentării numerelor în virgulă fixă este utilizarea reprezentării în virgulă mobilă cu scalare automată. În această variantă, factorul de scală devine o parte a cuvântului din calculator, iar poziția virgulei binare variază automat pentru fiecare număr. Un număr reprezentat în virgulă mobilă are două componente: mantisă (M) și exponent (E), memorate într-un cuvânt cu trei câmpuri: semnul, mantisa și exponentul.

Reprezentarea în virgulă mobilă poate fi realizată în mai multe moduri, folosind diferite metode de tratament a cazurilor de excepție, numărul de biți alocat fiecărui câmp în reprezentare, modul de rotunjire etc. Aceasta generează probleme de portabilitate a programelor, deoarece două calculatoare cu două moduri diferite de reprezentare pot interpreta aceleași date în moduri diferite.

Pentru a asigura portabilitatea programelor, Societatea Calculatoarelor a IEEE a elaborat un standard pentru reprezentarea numerelor în virgulă mobilă și pentru operații aritmetice corespunzătoare. Acest standard impune o metodă de calcul cu numere în virgulă mobilă, garantând același rezultat indiferent de implementarea în hardware, software sau o combinație a celor două. Erorile și condițiile de eroare produse în operațiile matematice trebuie raportate în mod consistent, independent de modul de implementare.

Una dintre problemele întâmpinate în această reprezentare este tratamentul depășirilor inferioare și superioare. Depășirea superioară apare atunci când un exponent depășește valoarea maximă, iar depășirea inferioară apare când exponentul are o valoare mai mică decât cea minimă.

Operația propusă pentru soluționare este înmulțirea numerelor reprezentate în virgulă mobilă, care presupune adunarea exponentului și înmulțirea mantisei. După operația de înmulțire, rezultatul trebuie rotunjit pentru a fi reprezentat pe 32 biți. Standardele permit mai multe moduri de rotunjire, cum ar fi spre 0, spre inf, spre -inf și spre cel mai apropiat număr reprezentabil.

Scopul acestui proiect constă în realizarea unei reprezentări corecte a numerelor conform standardului IEEE 754 pentru 32 biți, implementarea algoritmului de înmulțire corespunzător numerelor reprezentate în virgulă mobilă si realizarea operației de rotunjire pentru a obține rezultate precise.

2 Fundamente teoretice

2.1 VHDL

VHDL reprezintă un limbaj de descriere a hardware-ului (HDL - Hardware Description Language), utilizat pentru definirea comportamentului și arhitecturii modulelor electronice logice. În esență, acesta oferă posibilitatea de a descrie funcționalitățile logice combinatorii sau secvențiale ale unui sistem electronic. Alături de Verilog, VHDL este unul dintre cele mai frecvente limbaje utilizate în proiectarea sistemelor electronice digitale. Este considerat una dintre principalele unelte pentru proiectarea circuitelor integrate moderne și este folosit cu succes într-o varietate de domenii, inclusiv în microprocesoare, telecomunicații, industria auto și altele. Acest limbaj este esențial în procesele de proiectare asistată de calculator (CAD) pentru circuite integrate sau pentru configurarea FPGA-urilor.

2.2 Tehnologii utilizate

2.2.1 Basys 3

Placa Basys 3 FPGA este o platformă de dezvoltare hardware care integrează un cip FPGA Spartan-6 de la Xilinx și oferă un ecosistem flexibil pentru prototiparea sistemelor digitale. Cu un set variat de interfețe și componente, inclusiv butoane, LED-uri, porturi USB și conectori extensibili, această placă este ideală pentru învățarea conceptelor de proiectare digitală și pentru dezvoltarea prototipurilor. Echipată cu suita software Vivado Design Suite, este potrivită pentru studenți, hobbyiști și profesioniști care doresc să experimenteze și să învețe în domeniul electronicelor digitale.

2.2.2 Reprezentarea numerelor în virgulă mobilă, formatul IEEE 754

Reprezentarea numerică conform standardului IEEE 754 facilitează transferul programelor între diferite calculatoare, deoarece utilizează aceeași structură internă pentru numerele în virgulă mobilă. Această metodă de reprezentare poate fi stocată într-un singur cuvânt binar, structurat în trei părți: semn, mantisă și exponent.

Potrivit standardului pentru reprezentarea pe 32 de biți, dimensiunile celor trei componente ale reprezentării interne sunt definite astfel: un bit pentru semn, 8 biți pentru exponent și 23 biți pentru mantisă.

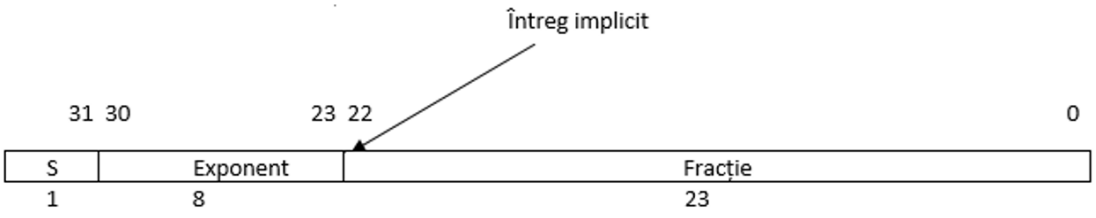

Figura 1: Formatul cu precizie simplă

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.

În acest format apare noțiunea de bit ascuns. Acesta se referă 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.

2.2.3 Inmultirea numerelor în virgulă mobilă

În ceea ce privește algoritmul de înmulțire al numerelor în virgulă flotantă, acesta are următorii pași:

- Conversia numerelor (x și y) în formatul IEEE
- Fie a exponentul lui x și b exponentul lui y
- Fie c = a + b exponentul rezultat
- Se înmulțește mantisa lui x cu mantisa lui y, iar rezultatul este m
- Se ajustează exponentul dacă este cazul
- Pentru a vedea care este semnul înmulțirii, se adună biții de semn si se face mod 2. Rezultatul este semnul
- Se convertesc numerele înapoi și se trunchiază dacă este cazul

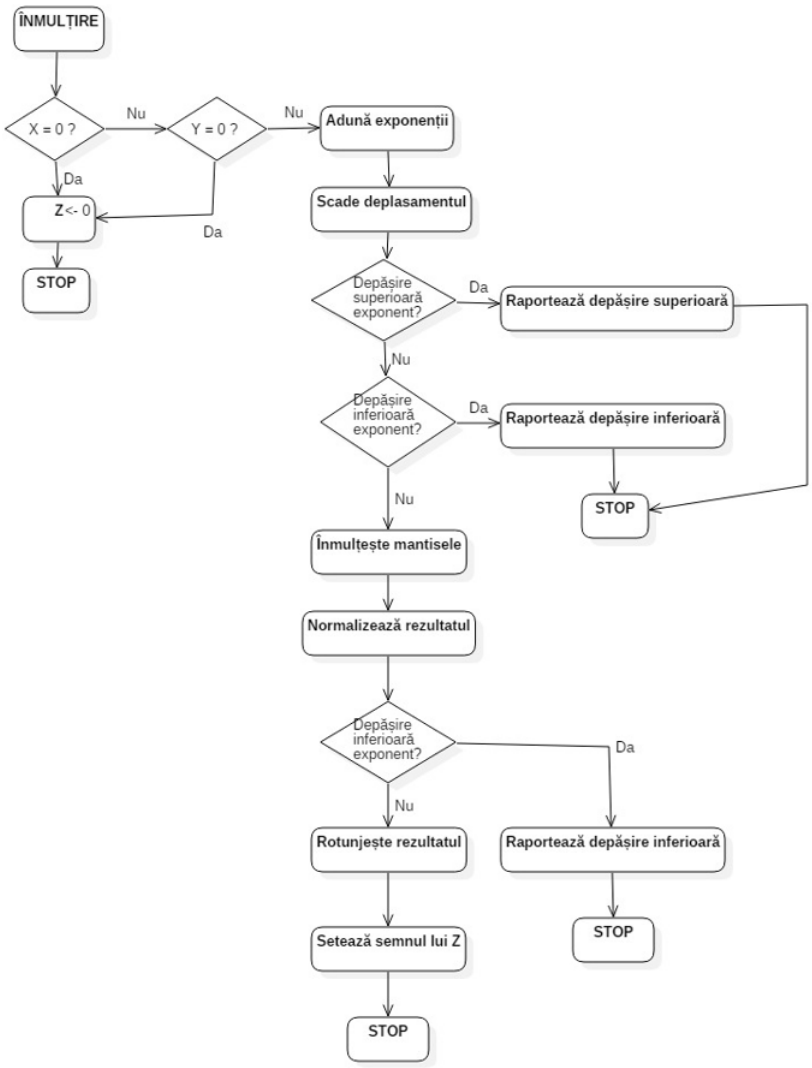

Figura 2: Algoritmul de inmultire

3 Proiectare şi implementare

3.1 Schemă înmulțitor

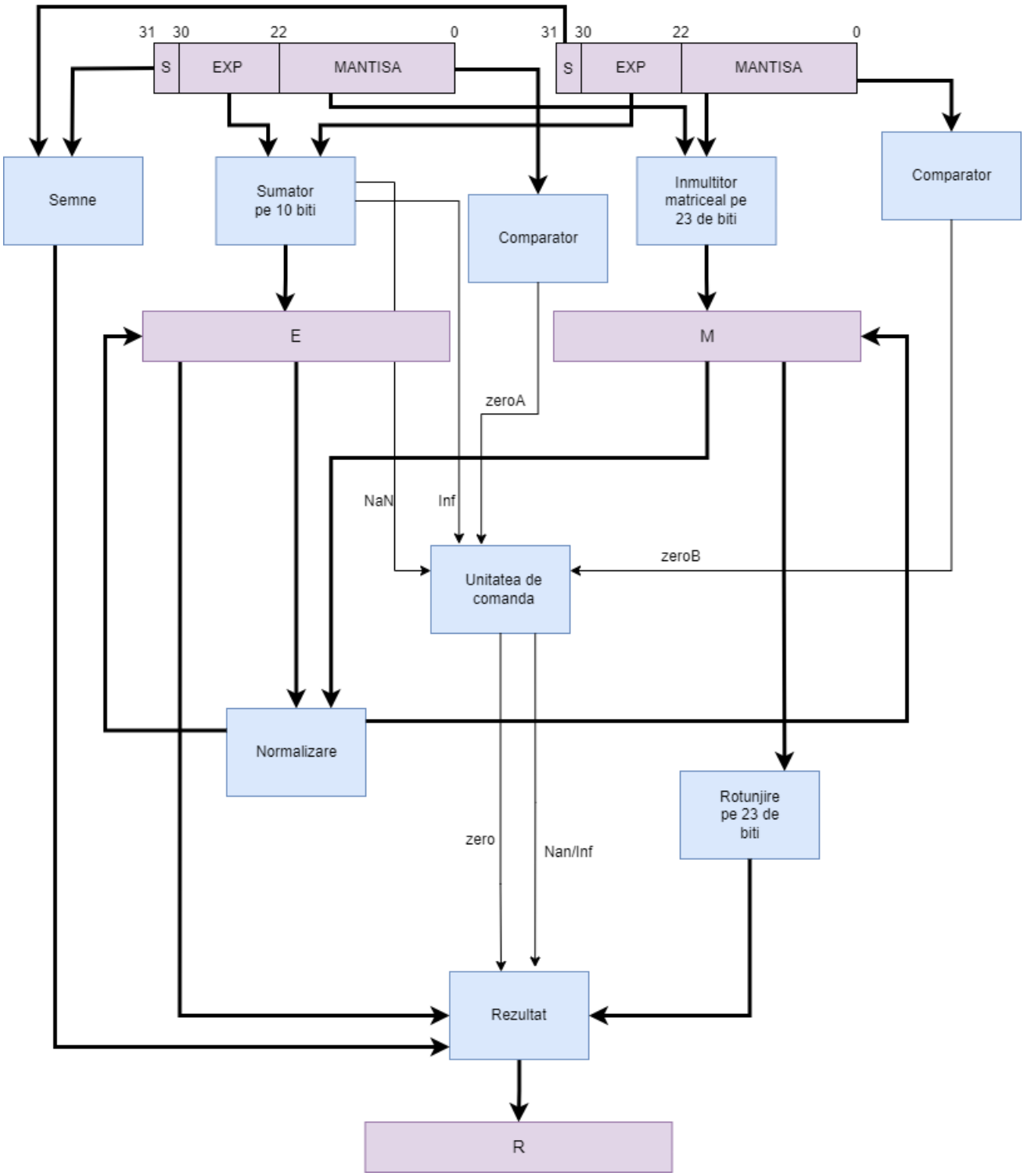

Figura 3: Schema bloc înmulțitor pe 32 biți în virgulă mobilă

3.2 Implementare

Implementarea va consta în crearea unei componente care să îi permită user-ului să introducă numerele dorite, cifră cu cifră, dar și să aleagă poziția virgulei din cadrul numărului. Apoi va fi nevoie de o altă componentă pentru a transforma input–ul user–ului în format binar, apoi în formatul standard IEEE.

Alta componenta va fi cea de înmulțire, care va avea la rândul ei alte componente auxiliare ajutătoare integrate.

După efectuarea operațiilor de înmulțire în formatul standard IEEE, o altă componentă va facilita conversia din IEEE înapoi în BCD pentru a putea fi afișate utilizatorului.

Pe lângă elementele descrise mai sus, se vor mai folosi debouncere pentru butoane, dar și un afișor pe șapte segmente.

3.3 Diagrama de stare

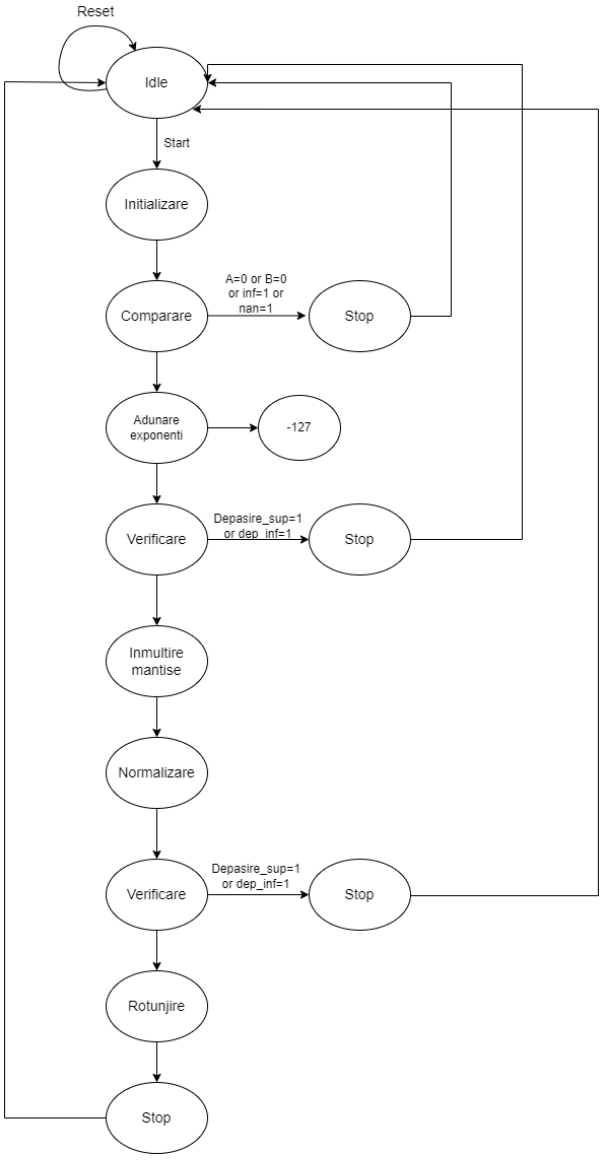

Figura 4: Diagrama de stare a inmultitorului

4 Testare

Din punct de vedere al testării aplicației, s-au realizat teste pe plăcuța FPGA Basys3, dar s-au realizat și testbench-uri pentru componentele de înmulțire .Acestea din urmă au fost simulate în Vivado.

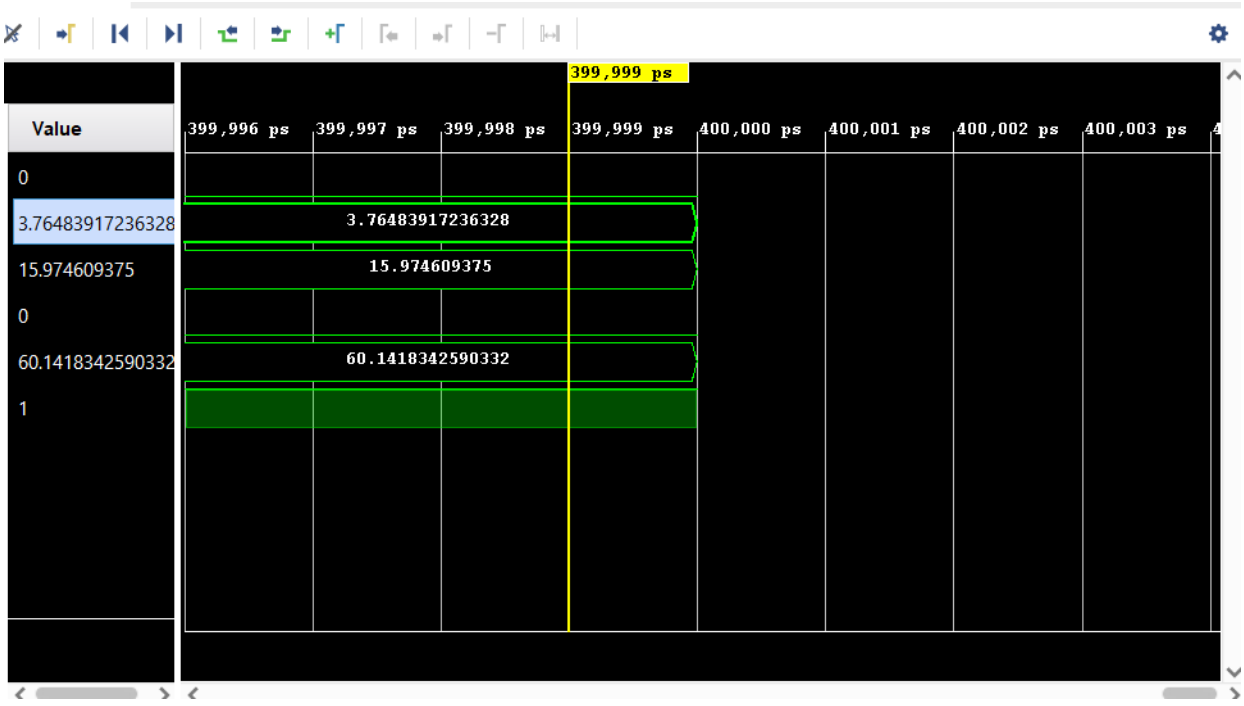

Figura 5: Simularea operatiei de inmultire in Vivado

4.1 Manual de utilizare

Tot din punct de vedere al testării, pentru testarea aplicației pe plăcuța Basys3, se vor folosi următoarele switch-uri/led-uri/butoane:

- sw0 – controlează numărarea crescătoare/descrescătoare
- sw2 – semnul de la al doilea număr
- sw3 – semnul de la primul număr
- sw4 – selectează numărul (1 primul număr, 0 al doilea număr)
- sw15 – load (1 încarcă, 0 afișează rezultatul)
- ld0 – done
- ld10 – cifra curentă 0
- ld11 – cifra curentă 1
- ld12 – cifra curentă 2
- ld13 – cifra curentă 3
- ld15 – semnul rezultatului
- btnc – numărare incrementare/decrementare
- btnr – selectare următoarea cifră
- btnd – schimbarea locației punctului

5 Concluzii

Scopul acestui proiect a fost proiectarea și implementarea unei aplicații ce suportă înmulțirea numerelor în virgulă flotantă pe 32 de biți.

Din pacate, nu am reusit sa integram modulul PmodKYPD in proiectul nostru, am reustit doar sa il facem sa mearga separat, afisand cate o cifra la apasarea unei taste.

Concluzionând, acest proiect a fost o oportunitate de a studia mai bine operația de inmultire în virgulă flotantă, conversiile IEEE-BCD și BCD-IEEE, dar și reprezentarea numerelor în formatul specific.

6 Bibliografie

- https://users.utcluj.ro/ baruch/media/ssc/curs/SSC-VM.pdf
- http://elf.cs.pub.ro/asm/wiki/laboratoare/laborator-12
- http://www.123seminarsonly.com/Seminar-Reports/2013-02/119072061-AnEfficientImplementationofFlo pdf.pdf
- https://digitalsystemdesign.in/floating-point-multiplication/
- https://hardwaredescriptions.com/floating-point-in-vhdl/
- https://vhdlguru.blogspot.com/2015/04/vhdl-code-for-bcd-to-binary-conversion.html
- https://electronics.stackexchange.com/questions/85132/converting-bcd-to-ieee-754