****

**Facultatea de Automatica și Calculatoare**

**Slot Machine pe afisor SSD**

Îndrumător Studenți: Alexe Rares-Nicusor

Grupa: 30215

27.05.2022

**Cuprins**

[**1.Rezumat** 3](#_Toc104673577)

[**2. Introducere** 4](#_Toc104673578)

[**2.1. FPGA și VHDL** 4](#_Toc104673579)

[**2.1. Obiectul proiectului** 4](#_Toc104673580)

[**3. Fundamentare teoretică** 5](#_Toc104673581)

[**3.1. Limbajul de descriere hardware VHDL** 5](#_Toc104673582)

[**3.2. Divizor de frecvență** 5](#_Toc104673583)

[**3.3. Afișaj cu Șapte Segmente** 5](#_Toc104673584)

[**4. Proiectare și implementare** 7](#_Toc104673585)

[**4.1.1. Automat Finit** 7](#_Toc104673586)

[**4.2.2. Proiectarea unui automat finit în VHDL** 7](#_Toc104673587)

[**5. Concluzii** 10](#_Toc104673588)

[**Bibliografie** 11](#_Toc104673589)

# **1.Rezumat**

FPGA-ul (Field-programmable gate array) este un circuit programabil, capabil să implementeze un anumit circuit definit de utilizator și este format dintr-o matrice de blocuri programabile, interconectate între ele printr-o serie de conexiuni programabile. VHDL este un limbaj de descriere a sistemelor electronice hardware, necesar realizării programării FPGA-ul oferit de Digilent Nexys4.

Proiectul s-a realizat folosind divizor de frecvență pentru un numărător cu timp constant, afișaje șapte segmente pentru animatie, afisarea mizei și afisarea creditului, monopulse generator pentru exactitatea apăsării butoanelor și unitate de comandă și control pentru implementarea organigramei.

# **2. Introducere**

# **2.1. FPGA și VHDL**

FPGA-ul (Field-programmable gate array) este un circuit programabil, capabil să implementeze un anumit circuit definit de utilizator și este format dintr-o matrice de blocuri programabile, interconectate între ele printr-o serie de conexiuni programabile.

VHDL este acronimul folosit pentru Very High Speed Integrated Circuit (VHSIC) Hardware Description Language(HDL). Acesta este un limbaj de descriere a sistemelor electronice hardware. Standardul, pentru VHDL, a apărut în 1987, denumit IEEE (Institute of Electrical and Electronics Engineers) 1076. Acesta a fost extins în 1993, urmând variantele din 2004, 2008 și actual este varianta 1076/2019.

Structura programului este bazată pe proiectere ierarhică, modelul VHDL fiind format din perechea entitate-arhitectură. Entitatea este zona de declarație a intrărilor și ieșirilor modulului. Arhitectura reprezintă zona de descriere detaliată a structurii modulului sau descrierea detaliată a funcționării modului.

# **2.1. Obiectul proiectului**

Obiectivul proiectului este realizarea unui joc folosind VHDL și implementarea acestuia prin intermediul FPGA-ul de pe o placă Digilent Nexys4. Jocul consta intr-un slot machine in care jucatorul isi alege balanta prin intermediul switch-urilor, la fel si miza, iar apoi apasa pe un buton prin care se afiseaza pe ecran rezultatul unei animatii ce imita animatia unu slot machine, iar in functie de combinatiile pe care le obtine jucatorul este premiat sau nu. Jucatorul are apoi posibilitatea sa isi retraga banii si astfel jocul s-ar opri, sau poate sa continue si sa-si aleaga din nou miza.

# **3. Fundamentare teoretică**

# **3.1. Limbajul de descriere hardware VHDL**

Sistemele hardware sunt în mod natural concurente, însă modelarea se face atât în domeniul concurent, cât și în domeniul secvențial. Obiectele limbajului sunt: constantele, variabilele și semnalele. Semnalele sunt specifice sistemelor hardware, acestea modelează informația care tranzitează între componente și există în tot timpul simulării.

Descrierea VHDL poate fi structurală, comportamentală, de tip ”flux de date” și o combinație a acestor trei tipuri de bază. Descrierea structurală constă în descrierea modelul prin structura sa. Descrierea de nivel comportamental urmărește să descrie funcționarea unui model fără a se preocupa de o eventuală împărțire în blocuri. O descriere de tip ”flux de date” este o formă prescurtată a unei descrieri comportamentale.

Procesul este unitatea de bază pentru descrierea de tip comportamental, fiind o serie de operații secvențiale care în timpul simulării constituie o singură acțiune. Un proces nu se termină niciodata, el execută în buclă lista de instrucțiuni secvențiale. Suspendarea și reactivarea se realizează cu instrucțiunea wait. VHDL ofera o construcție numită listă de sensibilitate a unui proces și ea este identică cu wait on listă\_de\_sensibilitate, scrisă la sfârșitul procesului. Un proces cu listă de sensibilitate nu poate să conțină nicio altă instrucțiune wait explicită.

Instrucțiunile din domeniul secvențial necesare realizării acestui proiect sunt: ,,if” și ,,case”. Instrucțiunea ,,if” permite executarea condiționată a unor secvențe de instrucțiuni. Pentru înlănțuirea condițiilor, fără a crește nivelul de imbricare, se foloseste ramura ,,elsif”. Instrucțiunea ,,case” permite selectarea, în funcție de valoare unei expresii, a unei secvențe de instrucțiuni dintre mai multe alternative.

# **3.2. Divizor de frecvență**

Divizorul de frecvență are ca scop reducerea frecventei de bază. Acesta este implementat folosind un factor de scalare și un numărător. Factorul de scalare reprezintă relația dintre frecvența de bază și frecvența dorita, fiind egal cu câtul împărțirii dintre frecvența de intrare și cea dorită.

# **3.3. Afișaj cu Șapte Segmente**

Un afișaj cu Șapte Segmente este un dispozitiv electronic care permite afișarea numerelor hexazecimale printr-o combinație de șapte segmente independente. Fiecare dintre segmente are două stări posibile pentru afișare, activate, sau neactivate. Numerele afișate sunt trimise ca modelul lor binar echivalent, pe patru biți, acestea trecând printr-un codificator în Șapte Segmente.

Aprinderea individuală a fiecărui segment este realizată prin aplicarea unui potențial negativ pentru pinul corespuzător segmentului. Iluzia schimbării numerelor se realizează pornind fiecare afișaj din cele patru folosite la un anumit interval.

Prin convenție, cele șapte segmente sunt notate cu litere de la a la g. Primul segment, notat cu litera a, este cel din partea de sus, apoi se face notarea în sensul acelor de ceasornic, segmentul central notându-se cu g.

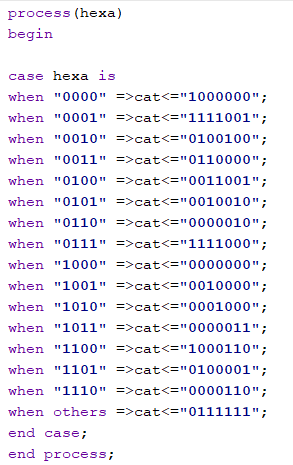


Figura 1.Codificarea în Șapte Segmente (logică negativa

**3.4. Monopulse Generator**

Monopulse Generator este o componentă necesară pentru a filtra semnalul apăsării pe buton. Această componentă funcționează prin luarea semnalului de la buton la un anumit timp (cu ajutorul unui numărător) și trecerea acestuia prin trei bistabile D (Delay). Output-ul este rezultatul operației logice ,,Și” între output-ul celui de al doilea bistabil D și output-ul negat celui de al treilea bistabil D.

**Chart, box and whisker chart

Description automatically generated**

Figura 2. Monopulse Generator

# **4. Proiectare și implementare**

# **4.1.1. Automat Finit**

Un automat finit este un model de comportament compus din stări, tranziții și acțiuni. Stările stochează informatii, o tranziție indică o schimbare de stare și este descrisă de o condiție care este nevoie să fie îndeplinită pentru a declanșa tranziția. Acțiunile pot fi de intrare, de ieșire, de intrare de date și de tranziție.

# **4.2.2. Proiectarea unui automat finit în VHDL**

Proiectarea automatul finit din proiect s-a realizat printr-o unitate de comandă și control. Aceasta, în principal conține trei procese: un proces pentru actualizarea stării actuale, un proces pentru a determina starea urmatoare și un proces pentru determinarea semnalelor de ieșire.

In starea idle apasam butonul btn\_power\_on\_off pentru a porni slot machine\_ul, dupa care in starea Start alegem soldul, iar apoi apasam butonul btn\_confirmare pentru a putea ajunge in starea Pariu unde alegem miza si apasam btn\_confirmare ca sa o selectam. Dupa aceea vom ajunge in starea scadere unde se scade din sold miza pe care am selectat-o si fara nici o conditie se trece la prima stare din cele 4. Acestea fac aceleasi lucruri, pe un SSD alocat pentru fiecare ruleaza niste numere, iar atunci cand apasam pe btn\_confirmarea se salveaza numarul pe care l-ai prin pe acel SSD. Dupa a patra animatie intram in starea evaluare in care daca rezultatele de pe SSD sunt egale intri in starea adunare care ne adauga la sold valoarea corespunzatoare, fiind alocate diferite sume pentru diferite combinatii de numere. Intr-un final, daca conditia de dinainte nu este indeplinita sau dupa ce am fost in starea Adunare intram in starea Retragere in care avem doua optiuni: fie apasam pe btn\_power\_on\_off care executa retragerea soldului, resetarea mizei si reintoarcerea in starea Start, fie apasam pe btn\_confirmare si continuam jocul cu soldul ramas reintorcandu-ne in starea pariu in care trebuie sa realegem miza(putem sa o alegem pe aceeasi).

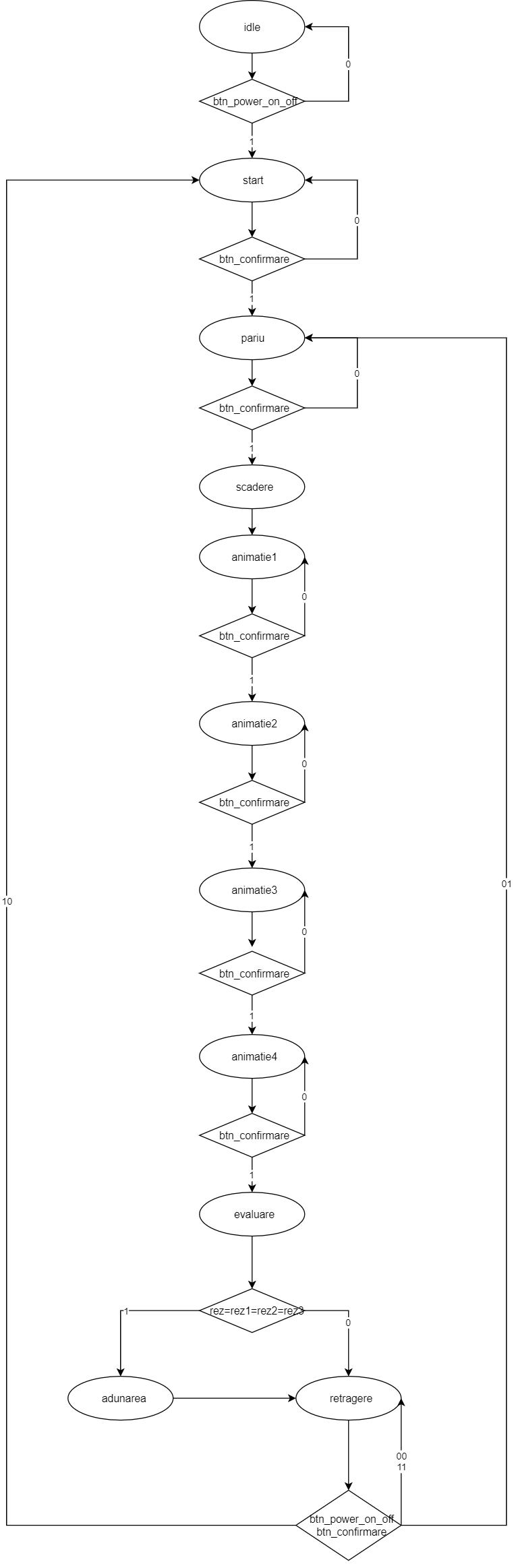


Figura 3. Organigrama

# **5. Concluzii**

Limbajul de proiectare hardware VHDL, împreună cu Digilent Nexys4 cu FPGA, switch-uri, led-uri, afișaje șapte segmente reprezintă un mediu propice dezvoltării unui automat finit care reprezintă un slot machine.

# **Bibliografie**

[1]. 1076-2002 – IEEE Standard VHDL Language Reference Manual. 2002.

[2]. Volnei A. Pedroni, Circuit Design With VHDL, Third Edition, The MIT PRESS, 2008

[3]. Volnei A. Pedroni, Finite State Machine in Hardware. Theory and Design (with VHDL and SystemVerilog), The MIT Press, 2013

[4]. Peter J. Ashenden, "The Designer's Guide to VHDL, Third Edition (Systems on Silicon)", 2008