Logo

Description automatically generated

Catedra: Calculatoare

Proiect: Interfata pentru un comutator rotativ

Student: Zilai Denis

Grupa: 30238

Indrumator proiect: Dr. Ing. Cristi Mocan

An: 2021-2022

Cuprins

[**1.** **Rezumat** 3](#_Toc92839144)

[1.1.Tema proiectului 3](#_Toc92839145)

[1.2.Problema de rezolvat si obiectivele principale ale proiectului 3](#_Toc92839146)

[1.3.Metoda de rezolvare 4](#_Toc92839147)

[1.3.1.Limbaje 4](#_Toc92839148)

[**2.** **Introducere** 5](#_Toc92839149)

[2.1.Contextul temei proiectului/tendintele tehnologice 5](#_Toc92839150)

[2.2.Domeniul de studiu 5](#_Toc92839151)

[2.3.Problema de rezolvat/obiective principale 6](#_Toc92839152)

[2.4.Solutia propusa 7](#_Toc92839153)

[**3.Fundamentare teoretica** 7](#_Toc92839154)

[3.1.Tehnologii care pot fi utilizate 7](#_Toc92839155)

[3.2.Solutia propusa 8](#_Toc92839156)

[3.3.Placa Spartan-3E 9](#_Toc92839157)

[**4.Proiectare si implementare** 10](#_Toc92839158)

[4.1.Arhitectura sistemului 10](#_Toc92839159)

[**5. Rezultate** 11](#_Toc92839160)

[**6. Concluzii** 12](#_Toc92839161)

[**Bibliografie** 12](#_Toc92839162)

[**Anexe** 13](#_Toc92839163)

[Codul implementat 13](#_Toc92839164)

# **Rezumat**

## 1.1.Tema proiectului

A close-up of a machine

Description automatically generated with low confidenceV-ati gandit vreodata ce este un comutator si care ar putea fi utilizarile sale in viata de zi cu zi? Aceste comutatoare rotative sunt folosite cu diverse scopuri in industrie, pentru automobile, in robotica precum si dispozitive rotative utilizate in detectie, montate pe nave maritime.

*Fig. 1 - Un comutator rotativ*

# 1.2.Problema de rezolvat si obiectivele principale ale proiectului

Obiectivul principal de indeplinit al proiectului a fost proiectarea si implementarea unei interfete pentru comutatorul rotativ aflat pe placuta FPGA Digilent Spartan-3E. S-a realizat o comunicatie intre comutatorul rotativ si alte componente ale placutei, in cazul de fata LCD-ul si cele 8 LED-uri.

Problema de fata este de a detecta miscarile comutatorului la stanga sau la dreapta, pentru a controla aprinderea celor 8 LED-uri aflate pe placuta FPGA. Daca comutatorul este miscat la stanga, se va aprinde cel mai apropiat LED din stanga celui aprins la momentul actual, la fel si pentru miscarea la stanga. Comutatorul poate actiona si ca si un buton, daca acesta este apasat aprinde toate LED-uri care initial erau stinse, si il stinge pe cel aprins. Daca se mentine apasat, acesta se poate si roti, putandu-se observa miscarea de stingere a celui mai apropiat LED de cel aprins, in functie de directia de rotatie.

Diagram

Description automatically generated with medium confidence

*Fig. 2 – Ilustrarea scenariilor de utilizare*

## 1.3.Metoda de rezolvare

Pentru rezolvarea problemei, a fost implementat, pentru eliminarea variatiilor rapide HIGH->LOW si invers, din timpul rotirii comutatorului, un filtru. Au fost adaugate de asemenea si doua semnale cu scopul de a identifica directia de rotatie. In continuare, LED-urile au fost inversate la apasarea comutatorului rotativ. Solutia a fost conceputa sincrona, folosind bistabilele deja integrate in placuta, astfel viteza fiind una ridicata.

## 1.3.1.Limbaje

Pentru dezvoltarea acestui proiect, s-a utilizat limbajul de descriere hardware VHDL.

1. **Introducere**

## 2.1.Contextul temei proiectului/tendintele tehnologice

Un comutator rotativ este un intrerupator care este operat prin rotatie. Acestea sunt adesea alese cand mai mult de 2 pozitii sunt necesare pentru utilizare. Aceasta componenta converteste pozitia unghiulara in semnale analogice sau digitale. Aceste comutatoare sunt de tip absolut sau incremental. Cel absolut indica pozitia curenta, pe cand cel incremental ofera informatii despre miscarea axului, acestea putand fi trimise mai departe altor componente pentru a fi utilizate referitor la pozitie, viteza respectiv distanta.

Comutatoarele au o utilizare larga, acestea fiind folosite in robotica, dispozitive uzuale precum mouse-uri, track-ball-uri, lentilele aparatelor foto, radare bazate pe rotatie.

## 2.2.Domeniul de studiu

Domeniul de studiu este cel al stiintei calculatoarelor, cu alte cuvinte, domeniul IT. Acest domeniu s-a dezvoltat exponential de la inceputul secolului XXI, aplicatiile ce rezulta din el devenind fundamentale in viata fiecarei persoane de pe planeta.

## 2.3.Problema de rezolvat/obiective principale

Problema principala de indeplinit a proiectului a fost proiectarea si implementarea unei interfete pentru comutatorul rotativ aflat pe placuta FPGA Digilent Spartan-3E. S-a realizat o comunicatie intre comutatorul rotativ si alte componente ale placutei, in cazul de fata LCD-ul si cele 8 LED-uri.

Ce este o interfata? În tehnologia calculatoarelor termenul de interfață se referă la un punct (loc) de interacțiune dintre două unități, dispozitive componente etc. ale unui sistem, care este compatibil din punct de vedere hardware și software spre ambele părți ce comunică prin el uni sau bidirecțional. Prin analogie, sensul de interfață poate fi uzual interpretat ca o față (suprafață) de margine, de graniță a unui element, care servește comunicației spre și/sau dinspre alte elemente. Interfața este o parte a unui sistem de operare care servește comunicării, facilitând aceasta.

Asadar, scopul acestui proiect este de a implementa o interfata pentru comutatorul rotativ aflat pe placuta Spartan-3E, mai exact, o legatura intre comutatorul rotativ si alte componente aflate pe placuta. Se doreste detectarea miscarii la stanga sau la dreapta a axului comutatorului, pentru a putea folosi cele 8 LED-uri disponibile pe Spartan-3E, concret, la rotirea axului, sa se aprinda LED-ul cel mai apropiat de cel aprins, la detectarea miscarii spre stanga sau spre dreapta. In plus, comutatorul poate fi si apasat precum un buton, la apasarea acestuia inversandu-se starea LED-urilor, putand fi de asemenea rotit la stanga sau la dreapta, de aceasta data miscandu-se LED-ul acum stins (inversat).

## 2.4.Solutia propusa

O solutie propusa ar fi sa se implementeze un filtru pentru eliminarea variatiilor rapide de HIGH – LOW in urma rotirii comutatorului; se implementeaza doua semnale pentru stabilirea directiei rotirii comutatorului; se inverseaza starea LED-urilor la apasarea comutatorului. Aceasta solutie este una rapida si eficienta, deoarece se utilizeaza bistabilele si componentele aflate deja pe placuta, solutia realizandu-se in stil sincron.

**3.Fundamentare teoretica**

## 3.1.Tehnologii care pot fi utilizate

Pentru realizarea acestui proiect, ca tehnologii, sunt suficiente ca resurse placa de dezvoltare FPGA Digilent Spartan-3E, aceasta avand atasata un comutator rotativ si 8 LED-uri necesare proiectului, si ca limbaj de descriere hardware, limbajul VHDL, in mediul de dezvoltare Xilinx ISE Design Suite versiunea 13.4.

Comutatorul rotativ de pe placuta poate fi rotit la stanga sau la dreapta, astfel deschizand un intrerupator inaintea celuilalt. Cand axul este stationar, se numeste pozitia detenta, in care ambele intrerupatoare sunt inchise.

Cand un intrerupator este deschis, un rezistor de tip pull-up aduce semnalul la High logic.

## 3.2.Solutia propusa

Comutatorul rotativ este o piesa hardware, care are ca scop, convertirea pozitiilor fizice, in semnale digitale sau analogice. Acesta poate fi utilizat si pentru determinarea sensului rotirii axului. Se vor observa modificarile semnalelor dedicate rotirii pentru a determina rotirea axului.

S-a propus urmatoarea logica: daca semnalul „A” are valoarea „1” logic, iar „B”, valoarea „0” logic, atunci se realizeaza o rotire la dreapta a comutatorului rotativ. Daca valorile sunt inversate, atunci se realizeaza o rotire la stanga a comutatorului rotativ.

Pentru aceasta logica pot aparea oscilatii ale semnalelor, producandu-se mai multe rotatii intr-un singura. Aceasta problema s-a remediat verificandu-se doar prima variatie a semnalelor, restul fiind ignorate pana in momentul modificarii celuilalt semnal. Bistabilele din placuta sunt utilizare pentru memorarea starilor acestor semnale.

Se utilizeaza doua semnale auxiliare „rotary\_q1” -> „A” si „rotary\_q2” -> „B”. Atunci cand „A” este pe „1” logic, de asemenea pe „rotary\_q1” se pune „1” logic; pentru B se realizeaza in acelasi mod. Cand si „A” si „B” sunt pe „0” logic, atunci „rotary\_q1” se reseteaza, atasandu-i-se „0” logic; pentru semnalul „rotary\_q2” se realizeaza in mod similar.

Diagram, schematic

Description automatically generated

*Fig. 3 – Cele doua semnale auxilaire*

Aceasta solutie este eficienta datorita faptului ca acest „filtru” pentru oscilatii functioneaza in mod sincron cu semnalul de ceas, semnalul de tact avand o frecventa ridicata de 50 MHz.

Pentru a observa aprinderea LED-urilor in fucntie de rotirea comutatorului, se utilizeaza un registru de deplasare sincron. La apasarea comutatorului, fiecare LED din registru este trecut printr-un XOR cu 8 de „1” pentru a inversa starea LED-urilor.

## 3.3.Placa Spartan-3E

Pentru realizarea acestui proiect, s-a utilizat placuta Spartan-3E, mai exact comutatorul rotativ aflat pe aceasta, respectiv cele 8 LED-uri.

Placuta Spartan-3E este optimizata logic, eficienta pentru aplicatii in care densitatile logice conteaza mai mult decat intrarile si iesirile. Este ideala pentru integrare logica, co-procesare DSP si control incorporat, necesitand procesare semnifcativa si putine interfete.

A picture containing text, electronics, circuit

Description automatically generatedAceasta placa are si un comutator rotativ, acesta avand o cama la care este conectat un ax, folosit pentru a opera doua intrerupatoare. Cele doua intrerupatoare sunt inchise atunci cand comutatorul se afla in pozitie detenta, curentul nefiind permis sa treaca. In momentul in care se roteste axul, un intrerupator se va deschide inaintea celuilalt, in functie de directia rotirii. Aceasta functionalitate se datoreaza rezistentelor de tip PULL-UP integrate in circuitul placutei Spartan-3E.

*Fig. 4 – Placa Digilent Spartan-3E*

Acest comutator mai are o functionalitate, acesta putand actiona si ca un buton, la apasarea acestuia; dispune de o rezistenta de tip PULL-DOWN. Cand se apasa comutatorul, se va inchide circuitul, permitand curentului sa treaca, obtinandu-se un „1” logic. In afara apasarii, curentului nu ii va fi permis sa treaca, astfel obtinandu-se valoarea „0” logic.

Diagram, schematic

Description automatically generated

*Diagram

Description automatically generatedFig. 5 – Schema de functionare a comutatorului rotativ*

*Fig. 6 – Diagrama semnalelor „A” si „B”*

*Diagram

Description automatically generated*

*A picture containing timeline

Description automatically generatedFig. 7 – Filtrarea semnalelor de intrare, folosing semnalele ajutatoare*

*Fig. 8 – Diagrama semnalelor „A”, „B”, „rotary\_q1” si „rotary\_q2” in urma a 4 rotiri consecutive*

Placuta Spartan-3E mai dispune si de 8 LED-uri, acestea fiind numerotate „LD0”, ... , „LD7”. Fiecare din cele 8 LED-uri este conectat la GND si la un pin al placutei, respectiv la o rezistenta.

A close-up of a circuit board

Description automatically generated with medium confidence

*Fig. 9 – Cele 8 LED-uri de pe placuta*

A diagram of a system

Description automatically generated with low confidence

*Fig. 10 – Functionarea mutarii LED-urilor*

**4.Proiectare si implementare**

# 4.1.Arhitectura sistemului

In cele ce urmeaza, se va prezenta schema bloc a proiectului. Aceasta este una relativ simpla, alcatuita din 3 4 intrari:

* rotary\_a – semnalul pentru intrerupatorul „A” al comutatorului rotativ
* rotary\_b – semnalul pentru intrerupatorul „B” al comutatorului rotativ
* rotary\_press – semnalul pentru identificarea apasarii butonului comutatorului
* clk – semnalul de ceas

si o iesire:

* led(7:0) – vectorul de LED-uri

**Diagram, schematic

Description automatically generated**

*Fig. 11 – Schema bloc*

**5. Rezultate**

S-au utilizat pentru realizarea acestui proiect de interfata pentru comutatorul rotativ, placuta Spartan-3E si mediul de dezvoltare software Xilinx ISE Design Suite, versiunea 13.4, alaturi de sistemul de operare Windows 10 pe 64 de biti. Corectitudinea realizarii proiectului s-a realizat prin testarea repetata a solutiilor incercate, nefiind utilizat un testbench, datorita dificultatii ridicate. Rezultatele obtinute nu au fost in totalitate cele asteptate, deoarece am intampinat niste dificultati la incercarea implementarii unei functionalitati, aceasta fiind ca la apasarea butonului, sa se afiseze un mesaj pe LCD, respectiv „ON”, insa incercarile au esuat si am ramas la functionalitatile descrise pana acum in proiect. In cele din urma, am reusit sa duc la bun sfarsit implementarea in ceea ce priveste controlarea LED-urilor de catre comutatorul rotativ.

**6. Concluzii**

Ca si concluzie, in urma realizarii acestui proiect, am ramas cu un bagaj destul de bogat in ceea ce priveste dezvoltarea de programe pentru placutele de tipul FPGA, datorita documentarii asidue si pertinente in depasirea problemelor intalnite pe parcurs, care au fost numeroase, deoarece nu am lucrat prea mult cu acest tip de dispozitive.

Problemele de implementare au fost intr-un final depasite, obtinandu-se rezultatele dorite in comportamentul comutatorului rotativ si al LED-urilor.

Avantajele acestui proiect sunt ca nu a necesitat componente externe, iar solutia fiind una rapida, optima si eficienta pentru realizarea task-ului dorit. Dezavantajul curent este ca pot aparea mici oscilatii ale semnalelor la apasarea butonului comutatorului rotativ, nefiind implementat un generator mono-impuls. Ca imbunatatire, pentru eliminarea oscilatiilor si un rezultat mai fidel, se poate implementa un generator mono-impuls; in plus, se poate rezolva ideea de generare de mesaj pe LCD-ul de pe placuta.

**Bibliografie**

[1] Ghid proiect, pagina domnului profesor Baruch [Site online, Decembrie 2021]

https://users.utcluj.ro/~baruch/ssc/proiect/Ghid-Proiect.pdf

[2] Rotary encoder, Wikipedia [Site online, Decembrie 2021]

<https://en.wikipedia.org/wiki/Rotary_encoder>

[3] Rotary Encoder Interface for Spartan-3E Starter Kit [Site online, Decembrie 2021]

https://www.yumpu.com/en/document/read/17057400/rotary-encoder-interface-spartan-3e-starter-kit

[4] How Rotary Encoder Works and Interface it with Arduino [Site online, Ianuarie 2022]

<https://lastminuteengineers.com/rotary-encoder-arduino-tutorial/>

[5] Spartan-3E FPGA Starter Kit Board User Guide [Site online, Decembrie 2021]

<https://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf>

[6] Sursa oferita de catre profesorul indrumator

Flux-proiectare-1.pdf

[7] Sursa oferita de catre profesorul indrumator

Flux-proiectare-2.pdf

**Anexe**

# Codul implementat

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.NUMERIC\_STD.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

library UNISIM;

use UNISIM.VComponents.all;

entity interface\_leds is

Port ( rotary\_a : in std\_logic;

rotary\_b : in std\_logic;

rotary\_press : in std\_logic;

clk : in std\_logic;

led : out std\_logic\_vector(7 downto 0);

rst : in STD\_LOGIC;

SF\_D : out STD\_LOGIC\_VECTOR (3 downto 0);

SF\_CE0 : out STD\_LOGIC;

LCD\_E : out STD\_LOGIC;

LCD\_RS : out STD\_LOGIC;

LCD\_RW : out STD\_LOGIC);

end interface\_leds;

architecture Behavioral of interface\_leds is

signal rotary\_a\_in : std\_logic;

signal rotary\_b\_in : std\_logic;

signal rotary\_press\_in : std\_logic;

signal rotary\_in : std\_logic\_vector(1 downto 0);

signal rotary\_q1 : std\_logic;

signal rotary\_q2 : std\_logic;

signal delay\_rotary\_q1 : std\_logic;

signal rotary\_event : std\_logic;

signal rotary\_left : std\_logic;

signal led\_pattern : std\_logic\_vector(7 downto 0):= "10000000";

signal led\_drive : std\_logic\_vector(7 downto 0);

signal lcd : std\_logic\_vector (63 downto 0);

--signal lcd\_delay : std\_logic\_vector (11 downto 0);

component lcd\_ctrl is

Port ( clk : in STD\_LOGIC;

rst : in STD\_LOGIC;

lcd : in STD\_LOGIC\_VECTOR (63 downto 0);

SF\_D : out STD\_LOGIC\_VECTOR (3 downto 0);

SF\_CE0 : out STD\_LOGIC;

LCD\_E : out STD\_LOGIC;

LCD\_RS : out STD\_LOGIC;

LCD\_RW : out STD\_LOGIC);

end component lcd\_ctrl;

begin

lcd\_ctrl\_i: lcd\_ctrl port map ( clk => clk,

rst => rst,

lcd => lcd,

SF\_D => SF\_D,

SF\_CE0 => SF\_CE0,

LCD\_E => LCD\_E,

LCD\_RS => LCD\_RS,

LCD\_RW => LCD\_RW);

rotary\_filter: process(clk)

begin

if clk'event and clk='1' then

rotary\_a\_in <= rotary\_a;

rotary\_b\_in <= rotary\_b;

rotary\_press\_in <= rotary\_press;

rotary\_in <= rotary\_b\_in & rotary\_a\_in;

case rotary\_in is

when "00" => rotary\_q1 <= '0';

rotary\_q2 <= rotary\_q2;

when "01" => rotary\_q1 <= rotary\_q1;

rotary\_q2 <= '0';

when "10" => rotary\_q1 <= rotary\_q1;

rotary\_q2 <= '1';

when "11" => rotary\_q1 <= '1';

rotary\_q2 <= rotary\_q2;

when others => rotary\_q1 <= rotary\_q1;

rotary\_q2 <= rotary\_q2;

end case;

end if;

end process rotary\_filter;

direction: process(clk)

begin

if clk'event and clk='1' then

delay\_rotary\_q1 <= rotary\_q1;

if rotary\_q1='1' and delay\_rotary\_q1='0' then

rotary\_event <= '1';

rotary\_left <= rotary\_q2;

else

rotary\_event <= '0';

rotary\_left <= rotary\_left;

end if;

end if;

end process direction;

led\_display: process(clk)

begin

if clk'event and clk='1' then

if rotary\_event='1' then

if rotary\_left='1' then

led\_pattern <= led\_pattern(6 downto 0) & led\_pattern(7);

else

led\_pattern <= led\_pattern(0) & led\_pattern(7 downto 1);

end if;

end if;

if rotary\_press\_in='0' then

led\_drive <= led\_pattern;

else

led\_drive <= led\_pattern xor "11111111";

lcd <= x"4F" & x"4E" & x"80" & x"80" & x"80" & x"80" & x"80" & x"80";

end if;

led <= led\_drive;

end if;

end process led\_display;

end Behavioral;