A picture containing clipart

Description automatically generated **Universitatea Tehnica din Cluj-Napoca**

Student: Marinescu Roxana-Maria, Nechiti Mihai Adrian

Grupa: 30236

Profesor indrumator: Lisman Florin

Data :8 dec 2022

T-REX GAME using PmodOLED



Contents

[1. Rezumat 3](#_Toc120389260)

[2. Introducere 4](#_Toc120389261)

[2.1 Descrierea contextului temei si a tendintelor tehnologice legate de aceasta tema 4](#_Toc120389262)

[2.2 Definirea domeniului de studiu, explicarea tehnologiei de bază și explicarea importanței acestui domeniu 4](#_Toc120389263)

[2.3 Obiectivele principale ale proiectului 5](#_Toc120389264)

[2.4 Solutia propusa 5](#_Toc120389265)

[3. Fundamentare teoretică 6](#_Toc120389266)

[3.1 Interfața SPI 6](#_Toc120389267)

[3.2 VHDL și Vivado 6](#_Toc120389268)

[3.3 Resursele folosite in realizarea proiectului 7](#_Toc120389269)

[4. Proiectare și implementare 7](#_Toc120389270)

[4.1 Descrierea componentelor principale 7](#_Toc120389271)

[4.2 Descrierea librăriei tRex și construirea caracterelor 9](#_Toc120389272)

[4.3 Schema generată 10](#_Toc120389273)

[4.4 Manual de utilizare 10](#_Toc120389274)

[5. Rezultate experimentale 11](#_Toc120389275)

[5.1 Descrierea jocului 11](#_Toc120389276)

[5.2 Detalii tehnice 12](#_Toc120389277)

[6. Concluzii 12](#_Toc120389278)

[7. Bibliografie 12](#_Toc120389279)

# Rezumat

Am ales aceasta tema cu scopul de a invata cum functioneaza protocolul de comunicatie SPI si am ales sa implementam jocul dino run. Obiectivele principale au fost implementarea unui protocol de comunicatie SPI si a unei librarii cu caractere pentru a face posibila afisarea unor animatii pe Pmod OLED. Proiectul a fost implementat pe o placuta FPGA basys3, in mediul de proiectare Vivado, iar limbajul de descriere hardware ales a fost VHDL. Proiectul a fost destul de simplu de implementat dupa ce ne-am documentat corespunzator.

# Introducere

## 2.1 Descrierea contextului temei si a tendintelor tehnologice legate de aceasta tema

*Display OLED*– fiecare “pixel” al ecranului se poate aprinde și stinge individual, în funcție de imaginea pe care doresti să o afișeze. Asta înseamnă că fiecare pixel al ecranului produce lumină și de asemenea se poate stinge complet.

Display-urile OLED au în general culori mai vii si un negru mai intens ce duce la un contrast mai mare. Este eliminată problema *bleedingului* de lumină, iar ecranul poate fi mai subțire din moment ce dispar straturile de cristale lichide.

## Definirea domeniului de studiu, explicarea tehnologiei de bază și explicarea importanței acestui domeniu

Un OLED este o componentă electronică în forma de folie foarte subțire si luminoasă facuta dintr-un material semiconductor organic. Spre deosebire de un display LED care are nevoie de iluminare suplimentara a fundalului , un display OLED ofera control la nivel de pixel si astfel se pot obtine cele mai profunde culori de negru (fiindca pixelul este stins) si nivel mai inalte de contrast.

Există două tipuri de tehnologii OLED:

* matrice pasiva (**PMOLED**)
* matricea activa (**AMOLED**) - mai buna pentru afișarea negrului profund; folosită în televizoarele OLED

Un panou OLED este compus din 4 straturi primare:

* Substratul - actioneaza ca structură de bază
* Anodul - atrage electroni
* Catodul - oferă electroni
* Stratul organic - împărțit în continuare într-un strat conductiv, care oferă "găurile de electroni" cu ajutorul cărora electronii care trec prin strat se pot aprinde, producând energie în proces și un strat emisiv în care se produce de fapt lumina.

## Obiectivele principale ale proiectului

* Demonstrarea capacitatilor tehnologiei Oled
* Informarea cu privire la protocolul de comunicație SPI
* Realizarea unui controller SPI pentru comunicarea cu display-ul Pmod OLED

## Solutia propusa

Soluția propusă de noi folosește un fsm pentru alternarea unor screen-uri facandu-se trecerea prin mai multe stari (dupa o stare care afiseaza un screen se trece intr-o stare ce face un delay si tot asa). În implementarea noastra este folosita o librarie de caractere customizate numită tRex, ce conține 16 caractere folosite în joc, atat pentru dino si cactus cat si pentru mesaje de genul `game over`.

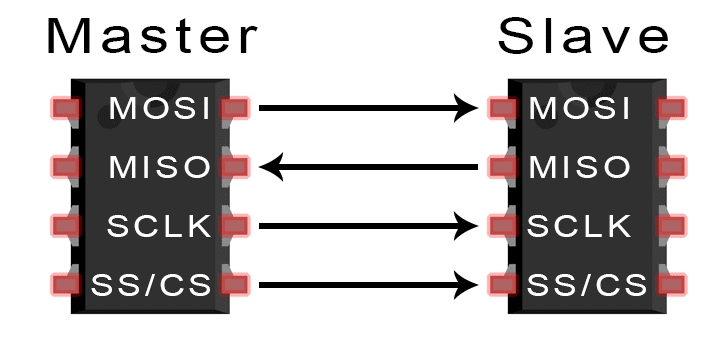
# Fundamentare teoretică

## Interfața SPI

Interfața serială periferică (SPI) este o specificație sincronă de interfață de comunicare serială folosită pentru comunicarea pe distanțe scurte, în principal în sistemele încorporate.

Dispozitivele SPI comunică folosind o arhitectură master-slave , cu un singur master. Dispozitivul principal generează cadrul pentru citire și scriere. Mai multe dispozitive slave sunt acceptate prin selectie cu selectare individuală a sclavilor (SS/CS).

SPI poate fi descris ca o interfață serială sincronă, protocol de comunicare sincronă cu patru fire. SPI este o comunicatie multi-slave cu single master.



## VHDL și Vivado

VHDL este un limbaj de descriere hardware utilizat în conceptia asistată pe calculator a circuitelor integrate sau pentru configurarea FPGA-urilor. Este cel mai popular limbaj de descriere hardware, alaturi de Verilog, și este una din principalele metode de proiectare a circuitelor integrate moderne. Vivado este un produs software realizat de Xilinx pentru sinteza și analiza proiectelor HDL.

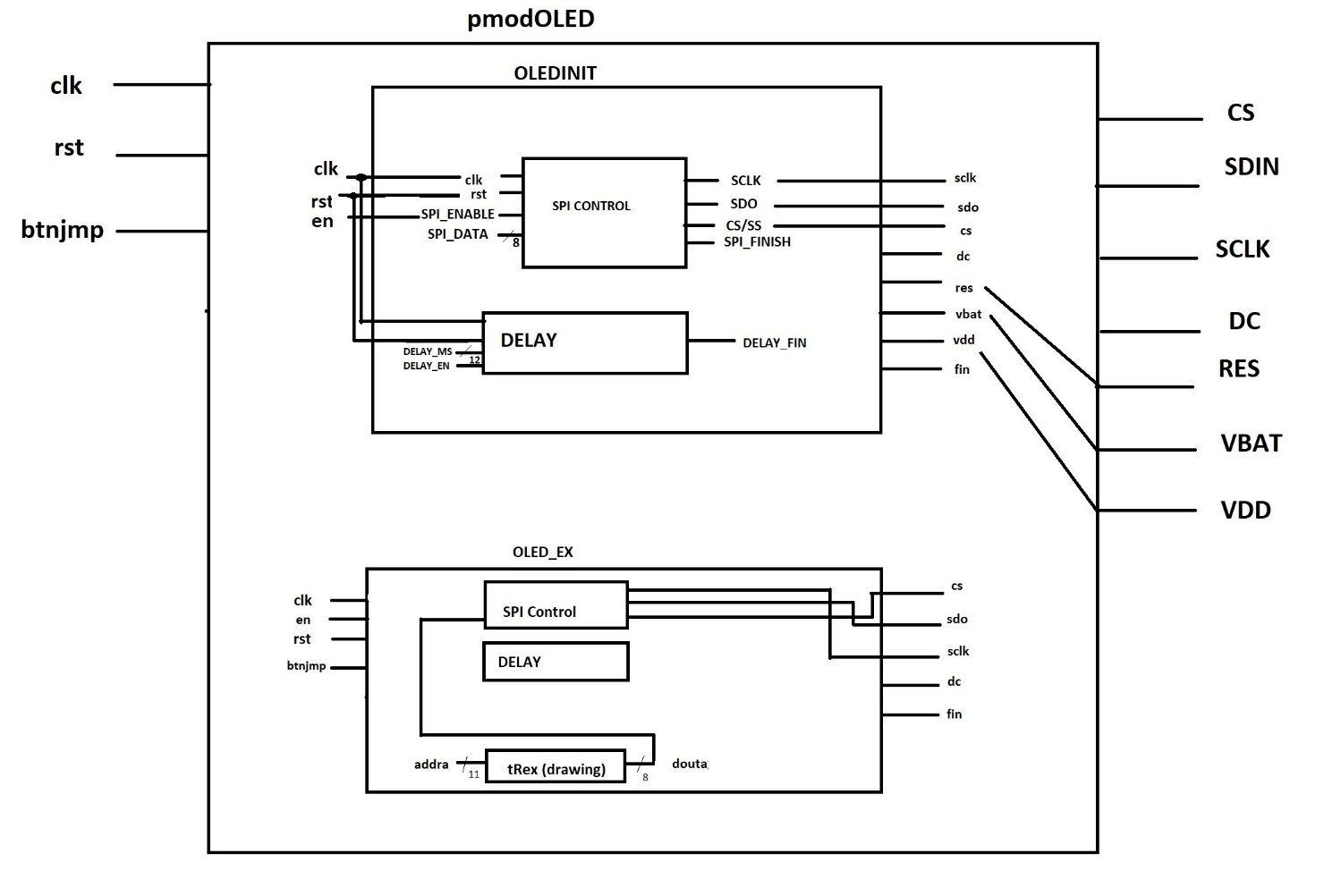
## Resursele folosite in realizarea proiectului

În realizarea acestui proiect au fost folosite:

* resursele de la laborator [[1]](http://users.utcluj.ro/~baruch/ssc/labor/Proiectare-Interfata.pdf)
* resursele oferite de Digilent [[2]](https://store.digilentinc.com/pmod-oled-128-x-32-pixel-monochromatic-oled-display/)

# Proiectare și implementare

## Descrierea componentelor principale



*Componente (ordine ierarhica):*

PMODOLED:

Acest modul este cel principal, acesta se muleaza perfect cu schema bloc, avand iesirile si intrarile aceleasi, iesirile dandu-se inspre componenta pmod prin interfata seriala SPI. In el se instantiaza doua module si se foloseste un FSM pentru controlul componentelor celula. Aceste module sunt:

=> OLED\_INIT:

Acest modul este cel care este folosit inspre a configura modulul pmod, aplicand diverse comenzi inspre acesta prin pinul de MISO, la fel cum se procedeaza cu trimiterea datelor. Are la indemana un FSM prin care, la fiecare front crescator al ceasului intern, aplica comenzile sau seteaza anumiti pini. Are ca componente modulele SPI\_CTRL si DELAY. Prima componenta este folosita pentru transmiterea comenzilor, cea de a doua este folosita inspre a face un scurt delay ca acele comenzi sa fie primite si aplicate de catre PMODOLED.

=> OLED\_EXEMPLE:

Acest modul este modulul in care se face animatia jocului. In el se instantiaza componentele SPI\_CTRL, DELAY si TREX. acesta are un FSM care controleaza componentele, construind logica jocului nostru. Astfel, Folosind un tip declarat de noi, o matrice de 4 x 16 cu elemente de 8 biti, declaram mai multe astfel de semnale care sunt de acest tip, ce reprezinta screenurile care vor fi puse pe modulul nostru. Sunt multe screenuri, vreo 13 pentru animatia cand cactusul se indreapta inspre dinozaur, 13 cand dinozaurul este in jump, pe aceasi coloana si cactusul se indreapta inspre el, cateva caractere care alcatuiesc mesaj de "Game Over" si ecranul de start care contine pe dino, cactusul si "GAME". Maparea unui element, din cadrul matricei, pe ecran folosim Starile de SENDChar1 - 8 care formeaza un cuvant pe 11 biti, in care primii cei mai semnificativi 8 sunt elementele din celula, iar ultimii 3 biti sunt folositi pentru alegerea carei colosane sa o trimitem (explicari suplimentare in modulul cu pricina).

=> SPI\_CTRL:

Acest modul este acela in care se face controlul transmiterii datelor prin interfata seriala intre master-slave. Acesta genereaza clockul folosit de cele doua placute intre care exista o interfata seriala SPI. Printr-un proces face ca pe frontul acestui clock, numit SCLK, sa transmita bit cu bit, bitii unui byte. Acestui procedeu, avand nevoie ceva timp pentru a se termina in siguranta tranferul datelor/comenzilor, se foloseste modulul de delay care ne pune la dispozitie pauza supficienta pentru infaptuirea acestui procedeu destul de interesant.

=> DELAY:

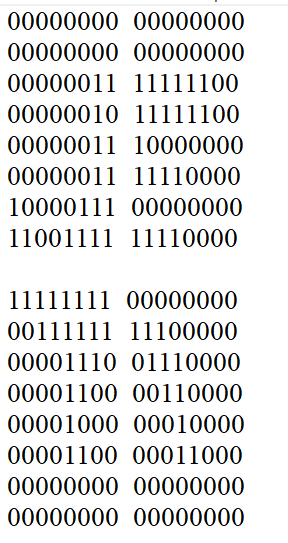
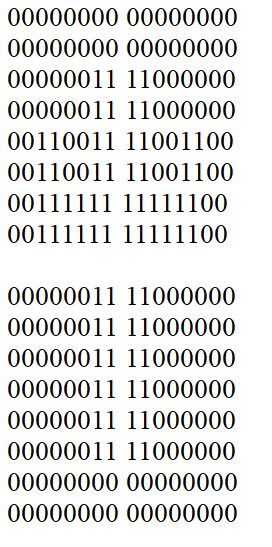
Acest modul este folosit in paralel cu SPI\_CTRL cu scopul de a oferii acestuia suficient timp cat sa isi poata face cum trebuie treaba, amount-ul de timp este masurat in ms. Mai este dotat cu un semnal de fuinish care indica ca s-a terminat, asemenea cu modulul de mai sus (tot un semnal flag de finish).

=> TREX:

In cadrul acestui modul, care are ca intrare un vector de biti de lungime 11, se scoate acei 8 biti care trebuiesc transmisi catre pmodOled prin SPI. acest lucru se face folosind un codificator care decodifica urmatoarele: primii 3 biti cei mai putin semnificativi sunt luati drept index de coloana, iar ceilalti 8 sunt luati drept adresa caracterului din care scoatem cei 8 biti o data. Pe scurt este nevoie de 8 astfel de accese ale unui caracter ca sa il putem scoate pe tot (de exemplu: daca caracterul tau are adresa 10, acele 8 accese se refera la acei 3 biti care tintesc coloana, 000 => prima coloana, 111 a opta). Explicatii mai pe amanuntite mai jos.

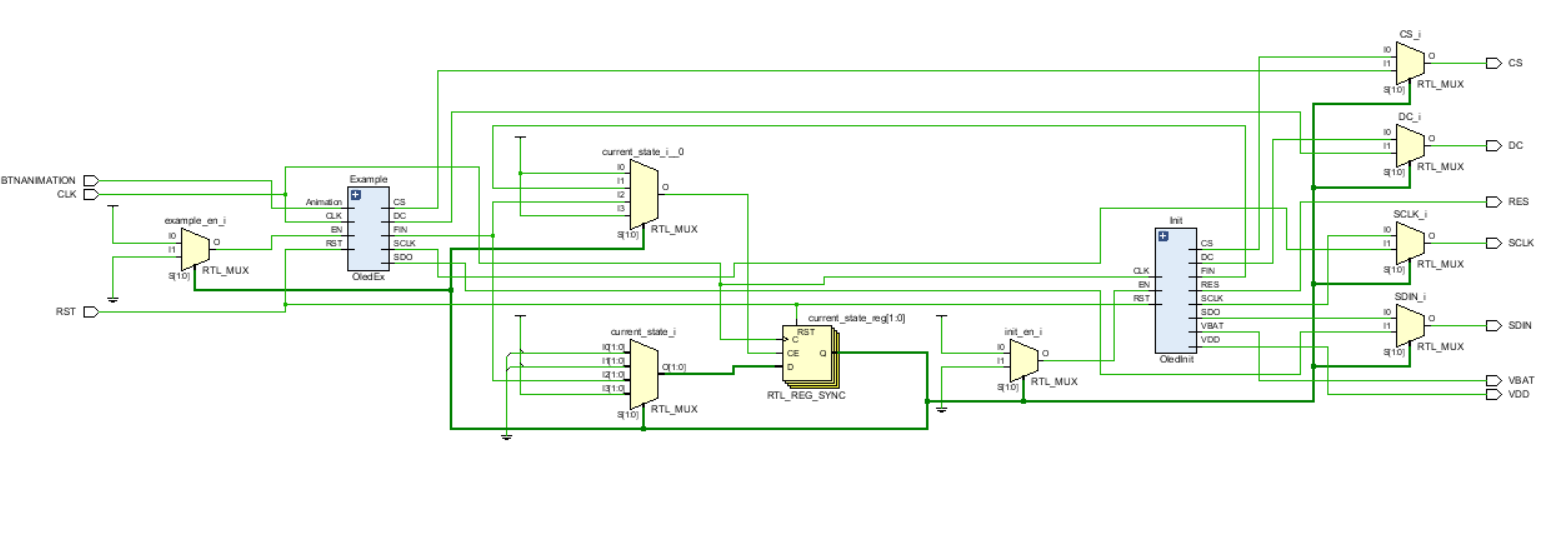
## Descrierea librăriei tRex și construirea caracterelor

Caracterele au fost desenate folosind grupuri de matrici de 8x8 biți. Trex-ul este constituit din 4 caractere customizate, un grid de 2x2 caractere fiecare având un numar de 64(8\*8) biți. La fel s-a procedat si in cazul cactusului. Mesajul de `game over` a fost realizat in acelasi fel, diferenta constand in faptul ca fiecare litera a fost scrisa folosind doar un caracter customizat pentru fiecare, adica 8 in total.

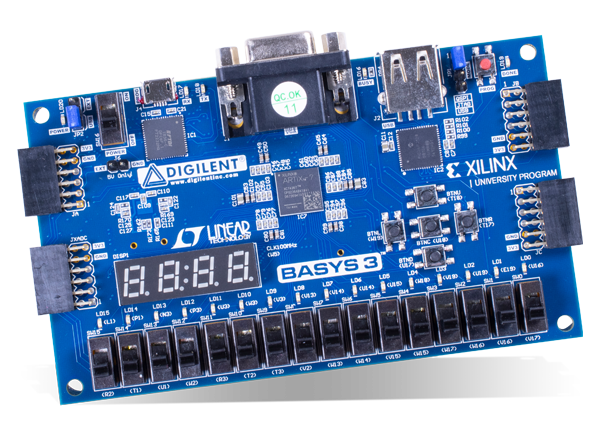
Caracterele sunt trimise către SPI de jos în sus, coloana cu coloana. Astfel este nevoie de 8 accese la aceasta librarie pentru trimiterea unui singur caracter customizat. Deci modulul tRex este o memorie care are biți stocați în ea.

## Schema generată



## Manual de utilizare

Conectati display-ul OLED în portul B pentru perifericele Pmod și programați placuta FPGA basys3. Butonul din centru este buton de reset, iar butonul de sus este folosit pentru jump.

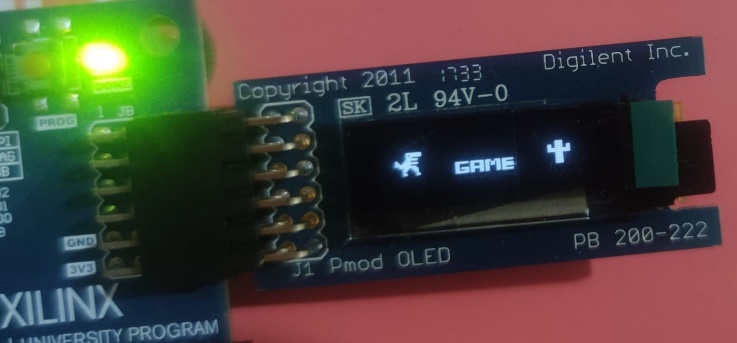


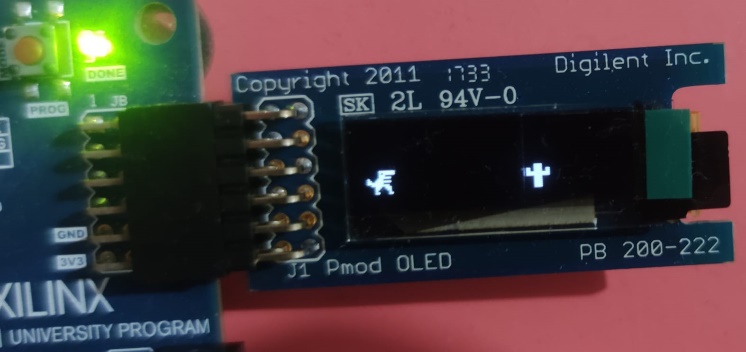
# Rezultate experimentale

## Descrierea jocului

Inițial este afisat un ecran de start, după care urmeaza un delay. Apoi ecranul este clear și apare pe ecran dino si cactusul. Sunt derulate o serie de imagini cu delay între ele. Daca este efectuat un jump la timp totul se repeta, dar daca dino loveste cactusul se va afisa un mesaj de `game over`. La sfarsit, inainte de a se afisa mesajul de joc terminat, are loc o tranziție între aceeasi stare a lui dino pentru a da senzatia de palpait cu scopul de a se observa faptul ca dino a lovit cactusul.

In urmatoarele imagini se poate observa rezultatul final al jocului:







## Detalii tehnice

S-au folosit 902 celule, 10 porturi de I/O și 1544 de nets. Procedura de testare utilizata a fost pe display-ul propriul zis, încercând sa sincronizam jocul.

# Concluzii

Intelegerea si realizarea SPI-ului au fost complicate. Realizarea librăriei in care se creeaza caracterele a necesitat multa munca si testari repetate.

In cea mai mare parte proiectul este destinat utilizarii. O posibila dezvoltare ulterioara ar putea fi adaugarea unui counter care sa retina scorul si afisarea punctajului total.

# Bibliografie

<http://users.utcluj.ro/~baruch/ssc/labor/Proiectare-Interfata.pdf>

<https://store.digilentinc.com/pmod-oled-128-x-32-pixel-monochromatic-oled-display/>