Logo

Description automatically generated

STRUCTURA SISTEMELOR DE CALCUL

**Măsurarea și afișarea accelerației pe trei direcții cu senzorul de pe placa Nexys 4 DDR**

Student: Nicoara Cristian-Cătălin

Profesor: Dragoș Florin Lișman

Universitatea Tehnica din Cluj-Napoca (UTCN)

Facultatea: Automatica și Calculatoare

Specializarea: Calculatoare și Tehnologia Informației

Anul 3, Seria B

Grupa: 30238

Data: 18.01.2023

Cuprins:

1. Rezumat

2. Introducere

* 1. Tema proiectului şi tendinţele tehnologice legate de această tema
  2. Domeniul de sudiu si tehnologia de baza
  3. Problema si obiectivele proiectului
  4. Solutia propusa si compararea acesteia cu alte solutii existente
  5. Descrierea urmatoarelor sectiuni

3. Fundamentare teoretica

* 1. Placa de dezvoltare Nexys4 DDR
  2. Accelerometrul ADXL363
  3. Interfata SPI

4. Proiectare si implementare

* 1. Schema bloc si porturile de intrare
  2. Modul de funtionare, arhitectura interna
  3. Probleme inmtampinate, incercari nereusite

5.Rezultate experimentale

* 1. Instrumente de proiectare utilizate
  2. Procedura de testare utilizata
  3. Rezultatele

6.Concluzii

7.Referinte

# REZUMAT

Proiectul care l-am avut eu este afisarea pe afisorul placii Nexys4 DDR a acceleratiei pe cele 3 axe folosind senzorul de acceleratie de pe placa. Pentru a obtine datele de la acceleromentru, era necesara utilizarea interfetei SPI, care controleaza executia slave-ului, care in acest caz este senzorul de acceleratie.

Limbajul utilizat la implementarea proiectului este limbajul de descriere hardware VHDL. L-am folosit pentru descrierea comportamentului fiecarei componente din proiect si legarea acestora intr-o entitate principala. Mediul de programare folosit este Xilinx Vivado.

Testarea nu o puteam face fara placa Nexys4, deoarece avem nevoie de datele aceleratiei trimise bit cu bit de la senzor. Doar pe o componenta era posibila testarea fara placa, componenta de afisare pe display-ul cu sapte segmente(SSD).

# INTRODUCERE

* 1. *Tema proiectului şi tendinţele tehnologice legate de această tema*

Tema proiectului consta in preluarea datelor masurate de accelerometru si reprezentarea lor pe afisorul cu 7 segmente. Intrucat placa Nexys4 DDR are incorporat un accelerometrul ADXL363 nu este necesar un modul adaugator, cum ar fi cazul daca s-ar folosi o blaca Basys3.

Accelerometrul este foarte folositor si de ajutor in zilele noastre. El ofera posibilitatea masurarii acceleratiei liniare si unghiulare, ceea ce il face util in domenii ca ingineria, biologia, industria, monitorizarea constructiilor, medicina, navigarea etc. Un exemplu de utilizarea a accelerometrului este pentru monitorizarea vulcanilor active, unde se foloseste pentru detectia miscarii lavei.

* 1. *Domeniul de sudiu si tehnologia de baza*

La baza, accelerometrul este un convertor al acceleratiei, ce masoara miscarea proprie in spatiu. El este implementat pe un obiect care vibreaza, cee ace ii permite sa transforme energia vibratiilor in semnal electric proportional cu acceleratia momentana. Majoritatea personelor cand se gandesc la masurarea acceleratiei, se gandesc la cat de repede merge o masina si nu gresesc, deoarece acesta se utilizeaza in acest domeniu. Insa nu prea se stie ca accelerometrele au un rol ridicat si in protejarea hard disk-urilor impotriva deteriorarii.

* 1. *Problema si obiectivele proiectului*

Problema principala a proiectului este crearea modulelor VHDL care va face in primul rand conexiunea cu accelerometrul, il va configura corespunzator si va prelua datele acceleratiei pentru axa X, axa Y si axa Z si le va transmite mai departe catre componenta de afisare. Este necesara configurarea, deoarece accelerometrul de pe placuta ofera mai multe funtiuni, cum ar fi si masurarea temperaturii, insa doar aflarea acceleratiei din registrele corespunzatoare este necesara.

Obiectivele proiectului sunt:

* Implementarea modului SPI – in VHDL.
* Configurarea si preluarea datelor de la accelerometru – in VHDL.
* Implementarea unui modul de afisarea a datelor preluate pe SSD – in VHDL.
* Crearea fisierului de contrangeri pentru Nexys4
* Incarcarea proiectului pe placa
  1. *Solutia propusa si compararea acesteia cu alte solutii existente*

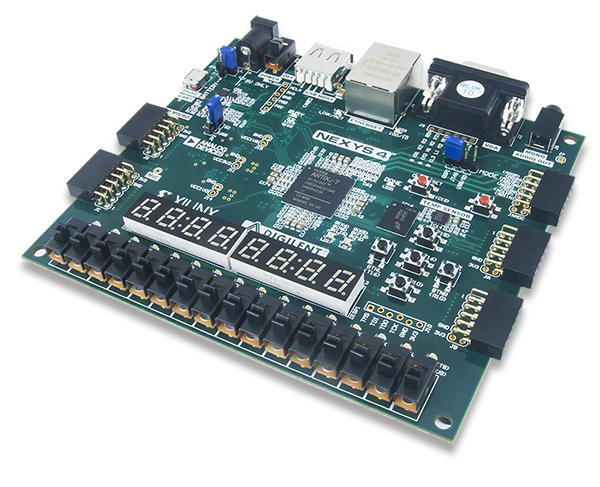
Solutia mea consta in crearea unei entitati generale pentru interfata SPI, a unei entitati pentru configurarea si preluarea datelor de la accelerometru, a unei enitati de afisare a acceleratiei pe afisorul cu 7 segmente si a unui entitati main, care leaga aceste componente. Aceasta este o adapatare simplificata mai multor solutii mai complicate gasite pe net. Una dintre aceste solutii prelua datele acceleratiei cu cate 12 biti pentru fiecare axa [1], pe cand eu iau doar 5 biti per axa, 1 bit de semn si 4 biti de date. O alta solutie la fel prelua datele in 12 biti per axa, daor ca nu le reprezenta pe afisor, ci le afisa pe un ecran cenectat cu un cablu VGA. Folosind solutia mea, chiar daca este mai simpla, se observa schimbarea valorii acceleratiei foarte usor si clar.

* 1. *Descrierea urmatoarelor sectiuni*
* **Fundamentare teoretica:** In aceasta sectiune se va prezenta amanuntit principiul de functionare al accelerometrului comunicand cu interfata SPI pentru al controla
* **Proiectare si implementare:** In aceasta sectiune se va prezenta modul si etapele de proiectare pentru rezolvarea problemei si indeplinirea obiectivelor, impreuna cu scheme bloc, pentru a se intelegea mai usor
* **Rezultate experimentale:** In aceasra sectiune se va afisa produsul final obtinut si poze cu rulare a proiectului pentru a se observa corectitudinea

# FUNDAMENTARE TEORETICA

* 1. *Placa de dezvoltare Nexys4 DDR*

Placa Nexys4 DDR este o placă de dezvoltare pentru circuite digitale proiectată pentru a fi utilizată cu FPGA-ul Xilinx Artix-7. Ea dispune de o varietate de interfețe și periferice, inclusiv un modul de memorie DDR3, un pod USB-UART, un accelerometru ADXL363, 8 afisare cu 7 segmente și diferite conectori de intrare / ieșire digitali și analogici, cum ar fi porturi USB, VGA, MicroSd etc. Este utilizată în mod obișnuit pentru proiecte educaționale și de hobby, precum și pentru prototipare și producție de scară mică a sistemelor digitale. Placa include un FPGA Xilinx Artix-7 cu o capacitate de 100 mii de porti logice, un modul DDR3 de 512MB, un ecran OLED de 2.4 inch, un modul USB-UART, o intrare audio, o intrare microfon, precum și o serie de conectori digitale și analogice pentru a permite interfațarea cu diferite senzori și actuatori.



* 1. *Accelerometrul ADXL363*

Accelerometrul ADXL363 este un accelerometru cu 3 axe, de putere redusă, produs de Analog Devices. Poate masura accelerația de-a lungul axelor X, Y și Z și poate oferi un interval de măsurare de până la ±2g, ±4g sau ±8g. Dispozitivul funcționează cu o singură sursă de tensiune și consumă foarte puțină energie (poate opera cu un voltaj de la 2V la 3.6V), ceea ce îl face foarte potrivit pentru utilizarea în aplicații portabile și alimentate cu baterii. ADXL363 are, de asemenea, un senzor de temperatură incorporat, care poate fi utilizat pentru a compensa schimbările induse de temperatură în sensibilitatea accelerometrului.

Se poate comuica cu acesta prin interfata SPI sau I2C și poate furniza date în diferite moduri, cum ar fi modul de măsurare, modul de trezire, modul de detectare a activității și inactivitate și modul de detectare a căderii libere.

Este utilizat pe scară largă în diverse aplicații, cum ar fi dispozitive portabile, jocuri, echipamente sportive și control și monitorizare industrială.

La pornire, accelerometrul este automat in modul de asteptare, de aceea primul lucru ce trebuie sal facem pentru a putea lucra cu acesta este sa-l scoatem din acest mod, si sa-l pune in modul dorit. In cazul meu, el trebuia pus in modul de masurare.

* 1. *Interfata SPI*

SPI (Serial Peripheral Interface) este o interfață de comunicare serială utilizată pentru a conecta diferite periferice la un microcontroller sau un calculator. Acesta permite transferul de date între două dispozitive prin intermediul unei linii de comandă (MOSI – Master Out Slave Input), o linie de citire (MISO – Master In Slave Out) și două linii de control (SCLK, SS – Slave Select). Este un protocol de comunicare sincron, în care cele două dispozitive se sincronizează prin intermediul unei linii de ceas (SCLK). Semnalul SS alege care dintre slavi sa fie activi.

SPI este utilizat în aplicații precum afișarea cu cristale lichide, stocarea de date, comunicarea cu senzorii și altele. Deși este similar cu interfața I2C, SPI este mai rapid și se pot conecta mai multe dispozitive la aceeași linie de comunicare.

Diagram, schematic

Description automatically generated

In imaginea de mai sus se observa ca masterul are 3 slave, insa in cazul meu am nevoie de doar un slave, accelerometrul.

SPI are patru moduri de funcționare, bazate pe doi parametri: polaritatea ceasului (CPOL) și faza ceasului (CPHA). Master-ul și slave-ul trebuie să folosească același mod pentru a comunica corect. Dacă CPOL este zero, atunci SCLK este în mod normal jos (incepe cu ‘0’), iar prima margine a ceasului este o margine ascendentă. Dacă CPOL este unu, SCLK este în mod normal inalt (incepe cu ‘1’), iar prima margine a ceasului este o margine descendentă. CPHA definește ordinea datelor. Dacă CPHA este zero, atunci primul bit de date este scris pe marginea descendentă al semnalului SS și citit pe prima margine al semnalului SCLK. Dacă CPHA este unul, datele sunt scrise pe prima margine a ceasului(SCLK) și citite pe a doua margine. La problema mea, accelerometrul ADXL363 necesita ca CPOL sa fie 0 si CPHA la fel.

Diagram

Description automatically generated

# PROIECTARE SI IMPLEMENTARE

Pentru a rezolva problema descrisa mai sus, am folosit interfata SPI pentru a configura aceelerometrul cu setarile de masurare si apoi sa preluam datele din registrele in care salveaza accelerometrul datele pentru fiecare axa. Intai am facut un modul general pentru interfata SPI, dupa care un alt modul de configurare si preluare a datelor folosind modulul SPI generalizat. Datele luate, sunt trimise la afisor, unde vor fi afisate pe SSD.

* 1. *Schema bloc si porturile de intrare*

Diagram

Description automatically generated

Schema bloc a proiectului nu este complicate. Se obbserva 2 componente, mod\_accelerometru si displ7seg. Mod\_accelerometru utilizeaza o alta entitate (SPI-ul generalizat) pentru a configura senzorul si a obtine acceleratia pe axe. Dupa cum se observa si din schema, valorile pentru fiecare axa se concatineaza si sunt ca intrare in entitatea de afisare, diplay7seg. In aceasta entitate se ia separat datele si pe fiecare le afiseaza pe cate 2 SSD-uri.

Deci porturile de intrare/iesire sunt:

* Clk – ceasul principal (al placutii) - in
* MISO – (Master In Slave Out) – in
* Rst – resetarea proiectului - in
* MOSI – (Master Out Slave In) – out
* SCLK – ceasul generat de SPI – out
* SS – (Slave Select) -out
* An - anozii pentru afisarea pe SSD
* Cat – catozii pentru afisarea pe SSD
* Sign – reprezinta catodul de semn
  1. *Modul de funtionare, arhitectura interna*

Am implementat modulul de configurare si preluare a datelor de la accelerometru folosind un finite state machine (FSM) cu 5 stari. Prima stare este start, unde se reseteaza variabilele, in caz ca s-a ajuns intr-o stare neconscuta pentru a nu crea probleme. Apoi, neconditionat, trece un urmatoarea stare. Aici se fac in functie de un parametru, care initial este 0 urmatoarele lucruri:

* Cand e 0, se incrementeaza parametrul, si se trimite intr-o variabila adresa 0x2C si in alta datele, care seteaza rata de date si lungimea de date care sa poata fi trimise
* Cand e 1, se incrementeaza parametrul si se trimite intr-o variabila adresa 0x2D si in alta valoarea “10” care va pune accelerometrul in modul de masurare
* Cand e 2, nu se mai incrementeaza, deci urmatoarele dati nu vor mai intra in celelate 2 cazuri, ceea ce e bine, pentru ca ele se fac doar la pornirea initiala a accelerometrului

In aceste tranzactii se poate ajunge doar dup 200ns de la ultima tranzactie, lucru ce tot se verifica in aceasta stare.

Din primele 2 cazuri, se trece in starea de unde se va configura accelerometrul, utilizand adresa si datele de mai sus.

Din starea 3 se trece la masurarea propriu-zisa.

La masurarea propriu-zisa, se trimite comanda de citire 0x0B, apoi se trimite adresa de unde sa inceaoa citirea datelor (0x0E). Aceasta este adresa pentru datele axei X. Deoarece aceelerometrul are o resolutie de 12 biti, sunt cate 2 registre pentru fiecare axa a cate 8 biti. 12 biti din 16 sunt de date, iar restul sunt biti de semn. Nu este nevoie sa trimitem adresa pentru fiecare registru, deoarece accelerometrul automat trece la registrul urmator, pana nu termina de citit datele. Apoi trece la ultimul pas, unde se asigneaza valorilor de iesire din componenta valorile acceleratiei axelor. Se ia un bit de semn, si restul de 4 biti sunt valorile cel mai semnificative.

Table

Description automatically generated A picture containing text, orange, close

Description automatically generated

Aceste valori ajung apoi in afisor. Acesta afiseaza datele zecimal, iar daca o valoare este negative ledul din dreapta numarului se va aprinde.

*4.3 Probleme inmtampinate, incercari nereusite*

In implementarea acestui proiect, am intampinat problema ca datele care erau citite de la accelerometru erau mereu 0. Am stat mult si nu imi dadeam seama care era problema. Apoi dup ace mam mai documentat un mic, mi-am dat seama ca nu scoteam accelerometrul din standby, stare care o are cand se deschide, si nu il puneam in modul de masurare.

# REZULTATE EXPERIMENTALE

* 1. *Instrumente de proiectare utilizate*

Limbajul de programare utilizat pentru scrierea proiectului este Xilinx Vivado (versiunea 2021.2)

* 1. *Procedura de testare utilizata*

Deoarece am lucrat pe accelerometrul de pe placa Nexys4, singura oportunitate de a testare a fost la laboratoarele de proiect. Din cauza ca am nevoie de date de la sensor pentru a testa nu puteam simula, folosind Vivado. Singura componenta care puteam sa o testez cu simularea era afisorul. Insa nu am simtit nevoia, deoarece aceasta parte a fost partea mai usora a proiectului.

* 1. *Rezultatele*

A picture containing text, electronics, circuit

Description automatically generated



Observam ca daca placa sta dreapta, coordonata x si y sunt 0



A hand holding a circuit board

Description automatically generated with low confidence



In plan inclinat, valorile sunt diferite si se schimba cu orice pozitie noua are placa

Video cu rularea proiectului:



# CONCLUZII

Scopul principal al proiectului a fost afisarea si preluarea datelor de acceleratie pentru fiecare axa. Pentru a face acest lucru am cautat mult informatie si pana la urma se afiseaza acceleratiile pe afisoarele cu 7 segmente ale placutei.

Pentru afisarea pe placa s-a facut conversia in BCD, iar apoi s-a luat fiecare cifra a acceleratiei unei axe si a fost afisata pe placuta. De asemenea pentru numerele negative, arde punctul zecimal din dreapta numarului.

Un avantaj al acestui proiect ar fi faptul ca datele se vad clar pe afisor. Un dezavantaj ar fi ca datele de acceleratie sunt doar pe 4 biti.Chiar daca s-ar folosi toate resursele placii, mai mult de 8 biti pentru o axa nu ar fi posibil.

O dezovltare ulterioara a proiectului ar fi afisarea datelor acceleratiei pe un ecran, conectat prin cablu VGA, unde nu sunt limitari sau o alta dezvoltare ar fi includerea altor date obtinute de accelerometrul ADXL363, cum ar fi temperature.

# BIBLIOGRAFIE

* <https://forum.digikey.com/t/accelerometer-adxl362-pmod-controller-vhdl/12922>
* <https://forum.digikey.com/t/spi-master-vhdl/12717>
* <https://digilent.com/reference/_media/reference/programmable-logic/nexys-4-ddr/nexys4ddr_rm.pdf>
* <https://www.analog.com/media/en/technical-documentation/data-sheets/adxl363.pdf>
* <https://github.com/Digilent/Nexys-4-DDR-OOB>
* <https://www.tme.eu/ro/news/library-articles/page/22568/Cum-functioneaza-si-la-ce-servesc-accelerometrele/>