

**Catedra Calculatoare si Tehnologia Informatiei**

**Înregistrarea unor date audio de la microfonul plăcii Nexys 4 DDR și transmiterea lor pe un dispozitiv mobil**

**Profesor îndrumator**: **Student**:

Dragos Florin Lisman Iamnitchi Bogdan

**Data**: An 3 CTI-RO 2022

30 Noiembrie 2023 Grupa 30235

**CUPRINS**

[1. REZUMAT 4](#_Toc151216890)

[2. INTRODUCERE 5](#_Toc151216891)

[2.1 Context și Importanța Proiectului 5](#_Toc151216895)

[2.2 Definirea Problemei și Obiectivele Proiectului 5](#_Toc151216896)

[2.3 Inovația Proiectului 6](#_Toc151216897)

[3. FUNDAMENTARE TEORETICA 7](#_Toc151216898)

[3.1 Captarea și Redarea Datelor Audio - concept 7](#_Toc151216899)

[3.2 DSP - Digital Signal Processor 7](#_Toc151216900)

[3.3 Procesarea audio pe Nexys4 DDR 8](#_Toc151216901)

[3.4 Memorarea datelor audio 9](#_Toc151216902)

[3.5 Transmiterea Datelor Audio către Dispozitivul Mobil 11](#_Toc151216903)

[3.6 UART - Universal Asynchronous Receiver/Transmitter 12](#_Toc151216904)

[4. PROIECTARE SI IMPLEMENTARE 13](#_Toc151216905)

[5. REZULTATE EXPERIMENTALE 14](#_Toc151216906)

[6. CONCLUZII 15](#_Toc151216907)

Trebuie sa zic despre

* Clkgen 200mhz pt memorie
* IP catalog
* Controler memorie DDR placa

# REZUMAT

# În cadrul proiectului descris, am abordat provocarea captivării și transmiterii datelor audio într-un mod inovator și eficient, folosind microfonul plăcii Nexys 4 DDR și tehnologia Bluetooth. Problema identificată a constat in transformarea semnalelor audio in vederea transmiterii lor fără fir prin intermediul tehnologiei Bluetooth. Obiectivele principale ale proiectului au inclus optimizarea calității înregistrării audio, dezvoltarea unei metode eficiente de prelucrare a datelor și implementarea unei soluții de transmitere sigură și rapidă prin Bluetooth.

# Metoda de rezolvare a presupus utilizarea tehnologiilor avansate de procesare a semnalului audio, integrate cu placa Nexys 4 DDR, pentru a asigura o captare precisă și o prelucrare de înaltă calitate a datelor provenite de la microfon. Am avut o provocare in ceea ce consta transformarea semnalului audio. Sunetul generat de catre microfonul placi este PDM (Pulse Density Modulation) iar un telefon mobil este capabil sa redea sunet doar in formatele WAV (Waveform Audio File) si MP3 (MPEG-1 Audio Layer 3).

Din pacate nu am gasit nici un algoritm de conversie din PDM in WAV sau MP3 si transmiterea a 5 secunde inregistrate ar fi durat aproximativ 180 de secunde, am decis in urma discutie cu domnul profesor sa nu mai transmit datele pe telefonul mobil prin UART utilizand un modul bluetooth, ci sa redicrectionez sunteul la jack-ul placi de unde dupa poate fi transmis mai departe la dispozitivul mobil printr-un dispozitiv special.

Insa nici aceasta solutie nu a functionat din cauza ca dispozitivul care se ocupa cu transmisia este capabil sa trimita sunete doar la dispozitivele special concepute pentru a primi sunet, precum ar fi casti bluetooth sau boxe. De mentionat ca mai jos am analizat si cum se poate implementa acest proiect folosind un modul bluetooth deoarce asa am inceput implementarea.

Sunetele captate in urma inregistrari sunt puse pe jack-ul audio al placii Nexys4 si pot fi interceptate folosind niste casti cu mufa compatibila. Astfel desi nu am reusit sa transmit sunetele pe un dispozitiv mobil, am invatat foarte multe in ceea ce priveste teoria sunetelor si cum o pot manipula la nivel de bit.

# INTRODUCERE

Progresul tehnologic continuu a deschis calea către inovații semnificative în domeniul înregistrării și transmiterii datelor audio. În acest context, proiectul propus se concentrează asupra capturării datelor audio de la microfonul plăcii Nexys 4 DDR și transmiterii acestor date către un dispozitiv mobil. Această inițiativă reprezintă o evoluție semnificativă în domeniul tehnologiei de captare și transfer a sunetului și are potențialul de a servi într-o serie de aplicații practice.



## Context și Importanța Proiectului

Proiectul se află într-un context tehnologic în care dispozitivele mobile și plăcile de dezvoltare au devenit din ce în ce mai comune și sofisticate. Cu toate acestea, capacitatea de a înregistra și transmite date audio de la un microfon la un dispozitiv mobil rămâne o provocare în domeniu. Această inițiativă vine să abordeze această problemă și să ofere o soluție practică pentru această cerință.

Transmiterea datelor audio de la un dispozitiv de bază, cum ar fi placa Nexys 4 DDR, către un dispozitiv mobil are potențialul de a revoluționa o serie de domenii, inclusiv comunicațiile, înregistrarea de sunete, monitorizarea ambientală și multe altele. De asemenea, această tehnologie ar putea deschide noi oportunități în ceea ce privește dezvoltarea de aplicații mobile și servicii pentru utilizatorii finali.

## Definirea Problemei și Obiectivele Proiectului

Obiectivul principal al acestui proiect este de a dezvolta o soluție eficientă și practică pentru înregistrarea datelor audio de la microfonul plăcii Nexys 4 DDR și transmiterea acestora la un dispozitiv mobil. În acest sens, problemele tehnice legate de captarea și transferul de date audio trebuie să fie rezolvate cu atenție pentru a atinge performanța și calitatea dorite.

**Obiectivele specifice ale proiectului includ:**

* Proiectarea și implementarea unei interfețe de captare audio eficiente pe placa Nexys 4 DDR.
* Dezvoltarea unui mecanism de transfer de date audio la un dispozitiv mobil folosind o conexiune adecvată.
* Asigurarea calității înregistrării și a transferului de date pentru a oferi o experiență audio satisfăcătoare utilizatorilor finali.

## Inovația Proiectului

Proiectul are relevanță din mai multe motive. În primul rând, înregistrarea și transmiterea datelor audio sunt aplicabile într-o gamă largă de domenii, cum ar fi telecomunicațiile, înregistrările audio profesionale, sistemele de securitate, dispozitivele medicale și multe altele. Capacitatea de a captura și transmite date audio de la un dispozitiv de bază, cum este placa Nexys 4 DDR, către un dispozitiv mobil aduce versatilitate și potențial de inovație în aceste domenii.

Ceea ce diferențiază acest proiect de alte soluții existente este abordarea sa inovatoare de a permite captarea și transferul datelor audio între un dispozitiv de dezvoltare și un dispozitiv mobil. Acest proiect propune o soluție care combină aspecte de hardware și software pentru a obține o performanță optimă în înregistrarea și transmiterea datelor audio.

În continuare, acest raport va explora detaliat fundamentarea teoretică, metodologia, rezultatele obținute și concluziile proiectului. Această documentație oferă un cadru complet pentru înțelegerea și implementarea soluției propuse.

Această secțiune a evidențiat contextul și importanța proiectului, a definit problema și obiectivele, precum și a subliniat inovația pe care acest proiect o aduce. Urmează să explorăm fiecare aspect în detaliu în secțiunile ulterioare ale documentației.

# FUNDAMENTARE TEORETICA

În această secțiune, vom explora fundamentarea teoretică necesară pentru înțelegerea și implementarea proiectului de înregistrare a datelor audio de la microfonul plăcii Nexys 4 DDR și transmiterea lor la un dispozitiv mobil. Această secțiune se axează pe conceptele, tehnologiile și principiile de bază care stau la baza proiectului.

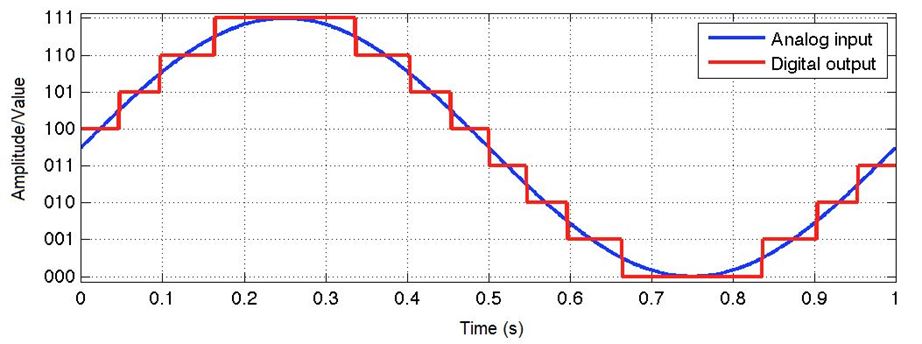
## Captarea și Redarea Datelor Audio - concept

Înregistrarea datelor audio reprezintă un proces esențial în domeniul tehnologiei audio. Pentru a realiza acest lucru, este important să înțelegem modul în care sunetele sunt captate și convertite în semnale digitale. Captarea sunetului reprezintă un proces complex, în care microfonul, în calitate de traductor, convertește undele sonore în semnale electrice. Aceste semnale, sub formă analogică, trebuie să fie digitalizate, pentru a fi prelucrate sau pentru a putea fi transmise mai departe, urmând ca mai apoi să se revină la forma lor analogică pentru a fi redare pe un dispozitiv audio de ieșire, în cazul nostru, speaker-ul de la telefonul mobil. Pentru a putea face posibil acest lucru ne vom baza pe conceptul de prelucrare digitala a semnalelor, care îl definesc mai jos.

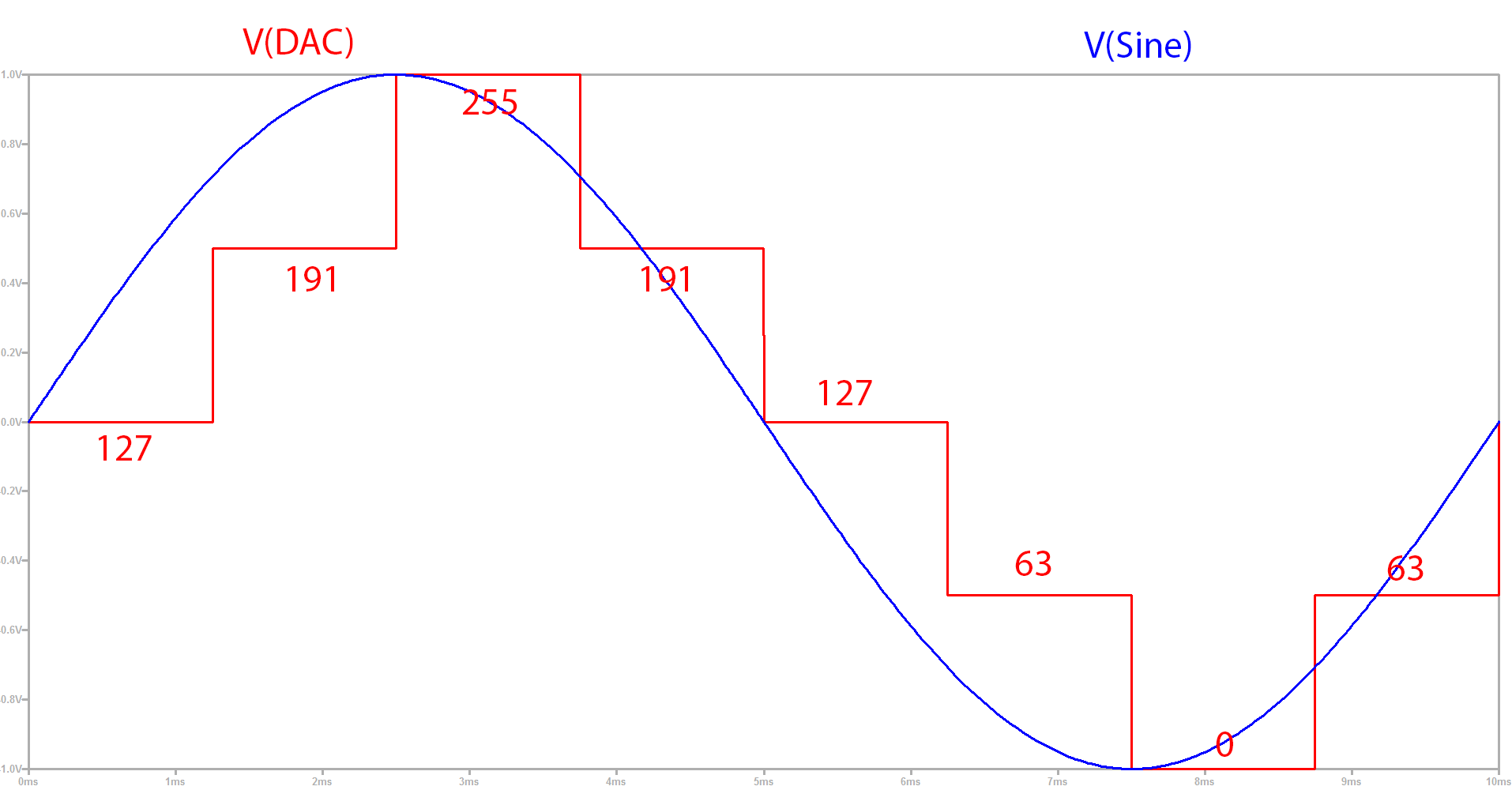
## DSP - Digital Signal Processor

DSP (Digital Signal Processor) implică în principal procesarea semnalelor prin mijloace digitale. Semnalele pot fi fie digitale, fie analogice. În cazul cel din urmă, înainte ca semnalul să poată fi procesat digital, trebuie să fie convertit în formă digitală folosind un convertor analog-digital (ADC) de un fel anume. Ulterior, semnalul este procesat înapoi în formă analogică folosind un convertor digital-analogic (DAC) după ce a fost procesat.

Există mai multe motive pentru care semnalele ar putea fi procesate digital. Un motiv este minimizarea "zgomotului electric" pe care semnalul îl poate dobândi ca rezultat al interferențelor.



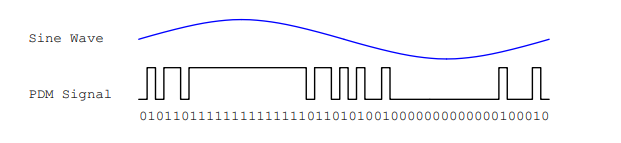
Figură 1: ADC Converter



Figură 2: DAC Converter

## Procesarea audio pe Nexys4 DDR

Placa Nexys 4 DDR este echipată cu un microfon MEMS omnidirecțional care utilizează cipul ADMP421, un dispozitiv analogic. Acest cip se remarcă printr-un raport semnal-zgomot înalt, și o sensibilitate ridicată. Datele audio captate sunt convertite într-un format de densitate a impulsurilor (PDM).



Figură 3: Reprezentarea PDM a unui semnal

Tehnologia PDM permite transmiterea a două canale de date utilizând doar două fire. Frecvența unui semnal PDM se încadrează în mod obișnuit în intervalul cuprins între 1 MHz și 3 MHz. Într-un flux de biți PDM, un "1" corespunde unui impuls pozitiv, în timp ce un "0" corespunde unui impuls negativ. O serie continuă de "1" reprezintă valoarea maximă a amplitudinii, iar o serie continuă de "0" reprezintă valoarea minimă a amplitudinii.

## Memorarea datelor audio

Datele audio înregistrate pe placa Nexys vor fi stocate în memoria DDR2 RAM a plăcii pentru a asigura o capacitate de stocare mai extinsă și o accesibilitate rapidă. Memoria DDR2 oferă o lățime de bandă mai mare și o capacitate sporită în comparație cu alte tipuri de memorie, ceea ce este esențial pentru manipularea eficientă a volumelor mari de date audio. O memorie DDR2 permite accesul rapid la datele înregistrate, facilitând astfel prelucrarea, redarea sau transferul acestora către alte dispozitive.

Nexys4 include o componentă de memorie DDR2 Micron MT47H64M16HR-25:H, 1Gb (64M X 16) care opereaza pe o interfață de 16 biți. Aceasta este o memorie din categoria Synchronous Dynamic Random Access Memory (SDRAM), utilizată în principal pentru stocarea temporară a datelor în sistemele informatice. Această memorie are o serie de caracteristici specifice care o fac potrivită pentru aplicațiile cu cerințe ridicate de transfer de date, cum ar fi în sistemele de calcul și dispozitivele embedded.

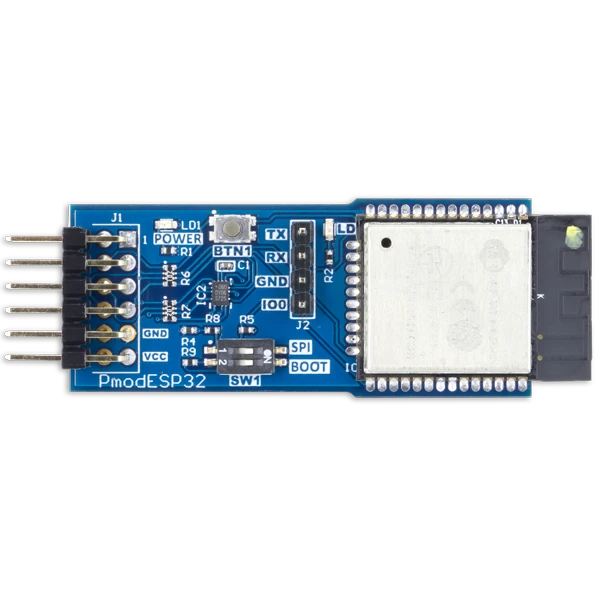
Caracteristicile cheie ale memoriei DDR2 includ:

* Arhitectură SDRAM: DDR2 utilizează o arhitectură SDRAM, care înseamnă că datele sunt stocate într-un mod static și trebuie reîmprospătate periodic pentru a evita pierderea lor. Totuși, spre deosebire de SDRAM tradiționale, DDR2 utilizează un mecanism de acces dual, permițând transferul de date la fiecare semnale de ceas, atât la frontul crescator cât și la cel descrescator al semnalului de ceas.
* Double Data Rate (DDR): DDR2 utilizează tehnologia DDR, care permite transferul de date la atât pe frontul, cât și pe coborâșul semnalului de ceas. Aceasta dublează eficiența în comparație cu tehnologiile anterioare care transmiteau date doar pe frontul semnalului de ceas.
* Capacitate și Lățimea de Bandă: Memoria specificată, Micron MT47H64M16HR-25, are o capacitate de 1 gigabit (Gb), ceea ce înseamnă că poate stoca 1 miliard de biți de informație. Structura de 64M x 16 indică faptul că memoria este organizată în 64 de milioane de cuvinte, fiecare cu o lățime de 16 biți. Acest lucru aduce o lățime totală de bandă de 16 biți x 64M cuvinte, adică 128 megabiți.
* Frecvență și Viteza: Specificația "25" indică faptul că această memorie funcționează la o viteză de 25 nanosecunde (ns), ceea ce este un indicator al timpului de acces al memoriei.

Controlul memoriei DDR2 se face prin intermediul unui controler DDR (DDR Controller), care gestionează accesul la memorie, sincronizarea cu semnalele de ceas, gestionarea comenzilor de citire și scriere și asigurarea funcționării corespunzătoare a interfeței DDR. Acest controler facilitează comunicația corectă între sistemul gazdă (de exemplu, un FPGA sau un microcontroler) și memoria DDR2.

## Transmiterea Datelor Audio către Dispozitivul Mobil

Transmiterea datelor audio către un dispozitiv mobil implică utilizarea tehnologiilor de comunicație. În cazul unui dispozitiv mobil, cum ar fi un smartphone sau o tabletă, aceste tehnologii pot include Bluetooth, Wi-Fi sau alte protocoale de comunicare. O componentă crucială a transmiterii datelor este utilizarea unui protocol de comunicație corespunzător și a unui software adecvat pentru gestionarea acestei transmiteri. Aceste elemente trebuie să fie integrate pentru a asigura o transmitere corectă și eficientă a datelor audio.

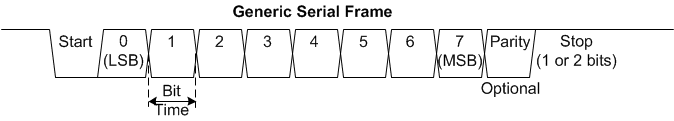
Comunicarea între placa Nexys 4 DDR și un telefon mobil, se va face utilizând un modul Bluetooth, Digilent Pmod BT. Pentru a face posibila acest lucru va trebui sa configuram modulul Bluetooth pentru a permite conexiuni și transmiterea datelor audio. Desigur si placa trebuie programată pentru a gestiona captarea datelor audio de la microfonul său și pentru a le trimite prin intermediul modulului Bluetooth către telefonul mobil. La nivelul telefonului mobil, acesta trebuie să fie echipat cu Bluetooth și să inițieze o conexiune cu placa Nexys 4 DDR, folosind o aplicatie proprie care va oferi o interfață placută utilizatorului.Odată stabilită conexiunea, datele audio captate de microfon sunt împachetate și transmise către telefonul mobil prin intermediul unui modul Bluetooth, Digilent Pmod BT, folosind o interfata de comunicare UART.

Figură 4: Digilent Pmod BT

## UART - Universal Asynchronous Receiver/Transmitter

Universal Asynchronous Receiver/Transmitter (UART) este un protocol de comunicare serială folosit pentru transmiterea și recepția datelor între dispozitive folosind biți, adesea folosit pentru conectarea și comunicația între microcontrolere și alte echipamente. Această comunicație permite transferul datelor audio de la placa Nexys 4 DDR către telefonul mobil, deschizând oportunități pentru diverse aplicații, cum ar fi înregistrări de sunete sau monitorizarea mediului înconjurător.

Comunicația serială are la bază transmiterea sau recepția datelor pe biți, cu un singur bit transmis sau recepționat la un moment dat. Deoarece în sistemele de calcul datele sunt reprezentate în octeți (sau multipli), se utilizează un port serial pentru a converti fiecare octet într-un șir de biți (0 sau 1) și invers. În timpul transmiterii unui octet, UART începe cu transmiterea bitului de START, urmat de biții de date (în mod obișnuit 8 biți, dar posibil și cu 5, 6 sau 7 biți), apoi de bitul de STOP. Acest protocol este repetat pentru fiecare octet din secvența care trebuie trimisă.



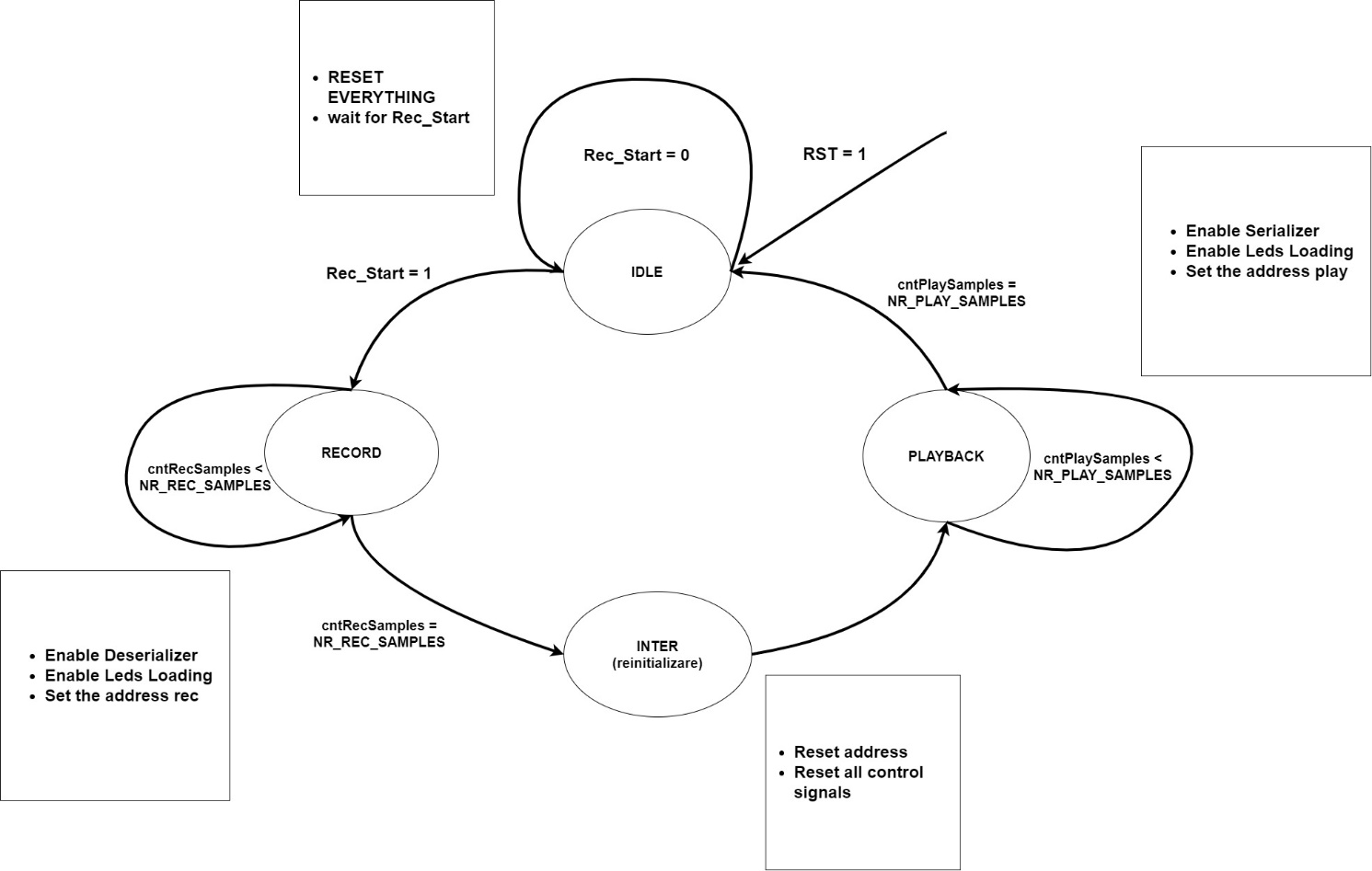
Figură 5: UART Transmision example

Transmisia serială nu necesită existența unui semnal de ceas comun între transmitător și receptor. În schimb, se utilizează o rată de eșantionare generată independent la sursă și destinație, cunoscută sub denumirea de baud rate (numărul de biți transmiși pe secundă). Valorile obișnuite pentru baud rate includ 2400, 4800, 9600 și 19200. În practică, un bit este considerat valid pe linia de transmisie pentru un interval de timp specific, echivalent cu inversul baud rate-ului. Câteva aspecte importante includ:

# PROIECTARE SI IMPLEMENTARE

Am ales sa merg pe o abordare simpla in care, initial apare un mesaj pe SSD care va indruma utilizatorul sa apese BTNU de pe placa Nexys4 dupa care se va afisa un mesaj ca inregistrarea a inceput, totodata cele 16 leduri arata trecrea celor 5 secunde de inregistrare. La terminarea celor 5 secunde pe SSD se va afisa mesajul „PLAYBACK” urmat de animarea celor 16 leduri in sens invers, totodata sunetul putand fi auzit la iesirea jack a placi folosind niste casti cu mufa compatibila.

## Etapa 1 – structurarea principalelor functionaliati

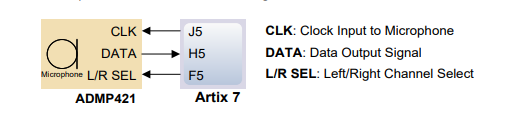
In prima etapa m-am gandit la principalele functionalitati ale proiectului si am implementat o organigrama de stare care va reprezenta principalul automat de stare al aplicatiei.

Figură 6: Main FSM

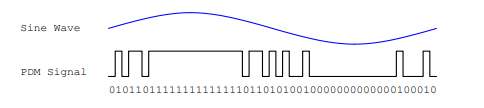
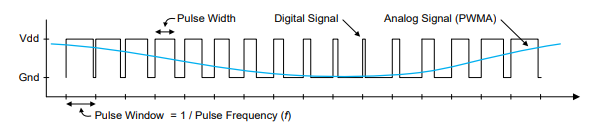
## Etapa 2 – utilizarea microfonului de pe placa

In aceasta etapa primul lucru pe care l-am facut a fost sa citesc despre microfonul de pe placa Nexys4. Am descoperit numeroase detalii manulalul de referinta al placi, (vezi [1]) si dupa am inceput sa structurez ce ar trebui sa fac. In manualul de referinta am descoperit ca exista un build in self test (demo) de la Digilent [2] unde poti testa toate functionalitatile de pe placa Nexys4 ca sa te asiguri ca totul functioneaza la standardele asteptate.

Analizand acel cod am descoperit cum ar trebui sa preiau datele de la microfon, trebuie doar sa generezi un semnal de clk cuprins intre 2Mhz si 2.4Mhz si sa setezi Left/Right Chanel. Am ales sa setez acest semnal la 0 ceea ce imi va da canalul din stanga si datele vor putea fi citite pe front crescator.



Figură 7: Microphone block diagram [1]

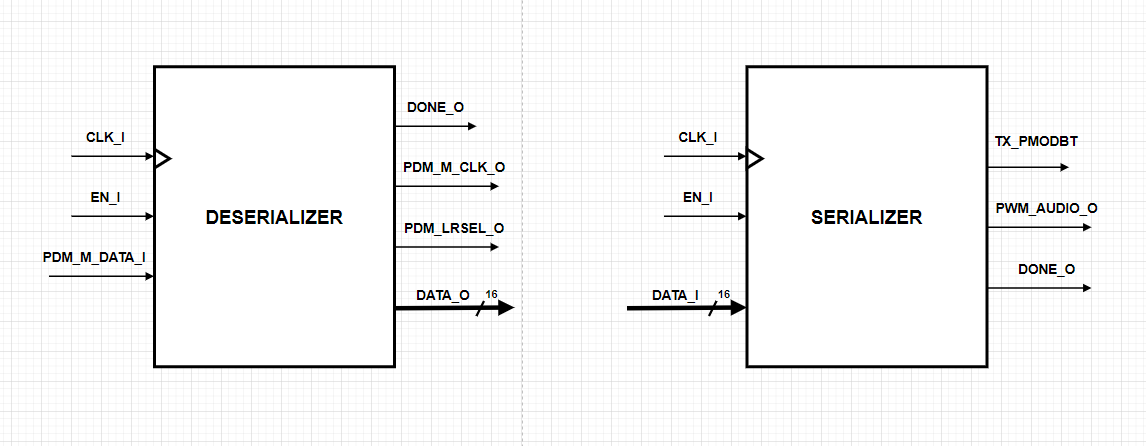
Datele citite de la microfon sunt in format PDM, iar pt a putea fi readate la jack-ul audio de pe placa trebuie convertite in PWM. Conversia se realizeaza usor in cod

Figură 8: PDM output

Figură 9: PWM output

## Etapa 3 – Manipularea datelor de la microfon

Folosind build in self test-ul de la digilent [2] am reusit sa testez ca microfonul merge corect si sa aud tot ce vorbesc in castile conectate la jack. Mai departe am avut nevoie de o modalitate de a memora tot ce inregistrez, astfel a trebuit sa ma gandesc la o metoda buna pentru a putea rupe sirul de biti continuu in cuante (sample-uri) de cate 16 biti pentru a putea maipula mai usor in ceea ce urmeaza.

In vederea acestui lucru am decis sa implementez doua componente. Prima este compomnenta Deserializer care are ca intrare, iesirea seriala de la microfon, si creeaza mai departe sample-uri ca cate 16 biti pentru a putea fi manipulati dupa. Iar a doua componenta este Serializer care serializeaza cei 16 biti si transforma in formatul PWM pentru a fi redat la iesirea jack a placii, dar mai poate si transmite bluetooth folosind o componenta UART sample-ul curent.

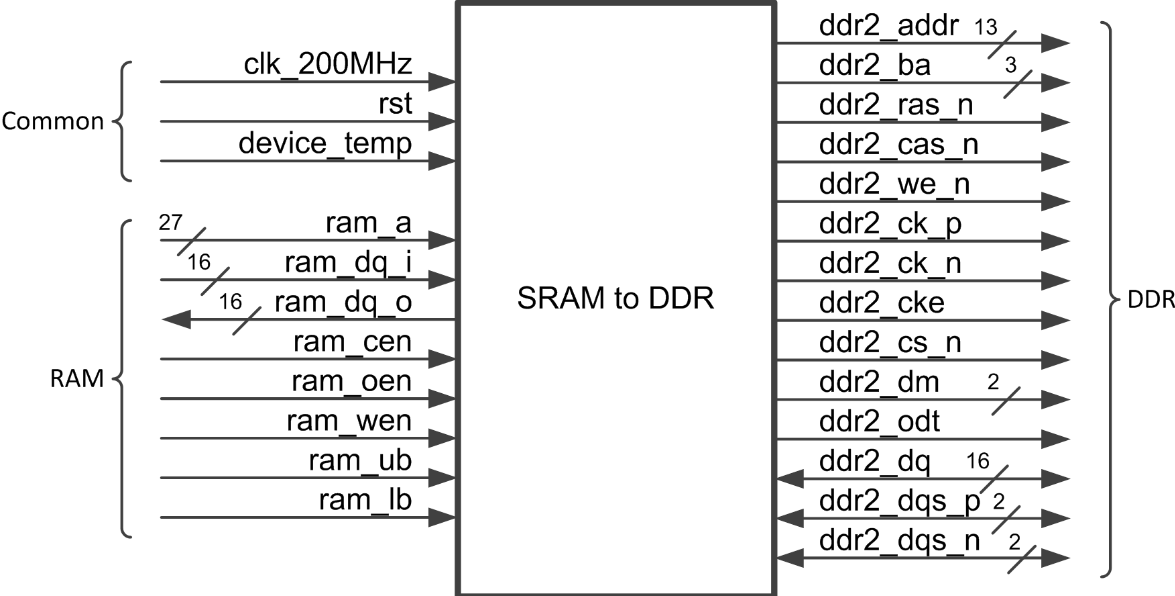
Figură 10: Componentele Deserialize si Serializer

Implementarea acestor componente a fost relativ simpla insa partea de sincronizare a fost destul de dificila insa am reusit. Odata implementate le-am adaugat in proiectul deja existent care stia sa redea sunetul inregistrat catre casti live fara a memora. Deci adaugarea acestor compomnente nu ar fi trebuit sa schimbe comportamentul daca ar functiona corect. Dupa mai multe incercari am reusit sa le implementez si sa le verific ca functioneaza corect deserializand si searilizand inapoi sunetul live.

## Etapa 4 – Memorarea sample-urilor create

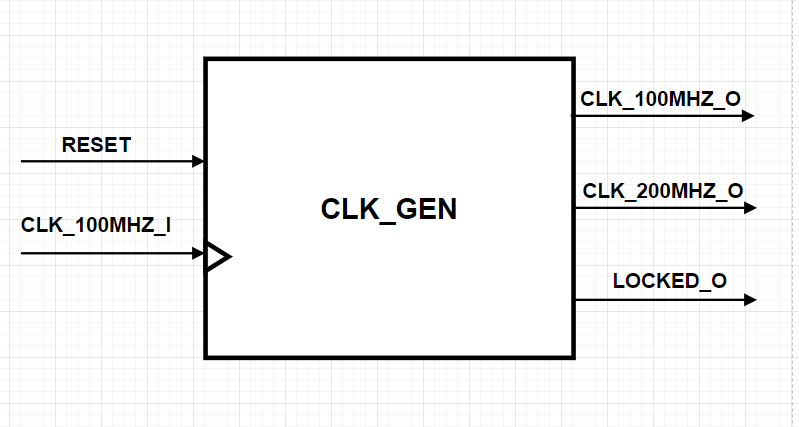
Deoarece am decis sa citesc datele de la microfon la o frecventa de 2Mhz, cantitatea de sample-uri pe care o facea deserializer-ul meu era una foarte mare undeva la 625.000 de sample-uri pt a inregistra 5 secunde.

Implementarea unei memorii pe FPGA ar fi fost foarte cositisitoare deoarce fiecare bistabil trebuia sa serveasca pt o celula de memorie logica combinationala de langa nu se folosea deloc.

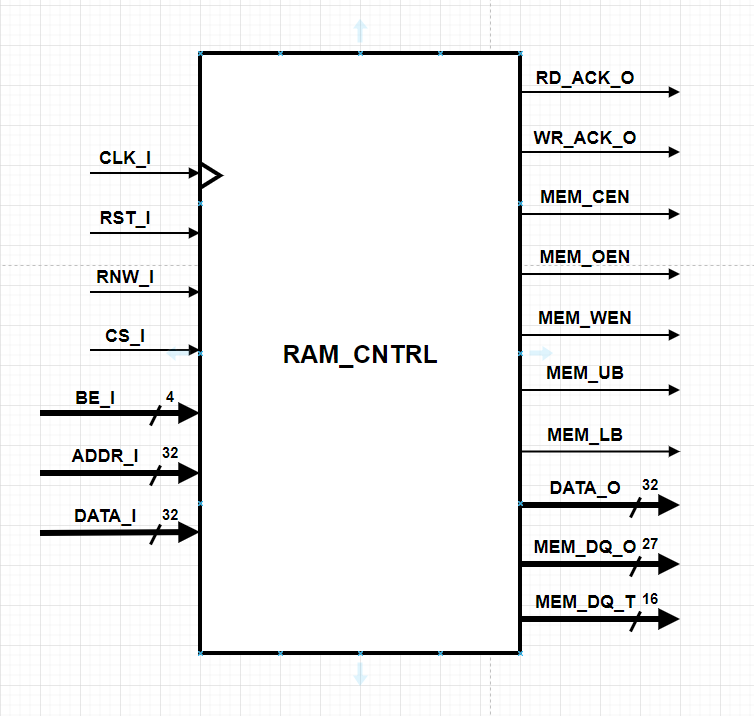
Nexys4 include o componentă de memorie DDR2 Micron MT47H64M16HR-25:H, 1Gb (64M X 16) care opereaza pe o interfață de 16 biți, care am decis sa o folosesc. Deoarece nu am abordat pana acum interfatarea unei memori reale de pe placa am decis sa ma folosesc de materialele puse la dispozitie de catre digilent [3] [4].

Figură 11: SRAM to DDR

Dupa cum se poate vede in schema aceasta componenta opreaza la un semnal ce deas ce 200Mhz ceea ce poate parea absurd, avand in vedere ca clock-ul placi este de doar 100Mhz. In vivado exista asa numitul Clock Wizard este o componenta IP cu ajutorul careia putem creea o componenta care poate creea diferite frecvente de clock atat mai mari cati si mai mici folosindu-se de PLL (phase lock loop). [5]

In urma vizionarii tutorialului [6] mi-am implemnetat si instantiat propria componenta clkGen care este capabila sa imi furnizeze un clk normal de 100Mhz si unul de 200Mhz pentru a putea controla memoria.

Figură 12: ClkGen component generated using IP Catalog

Deci pentru a comunica cu memoria DDR a trebuit implemnetat si un RAM Control care implementeaza toate functionalitatile unei memorii SRAM care dupa este translatata catre DDR de pe placa. Codul este unul standardizat si pus la dispozitie de cate digilent pt oricine vrea sa controleze memoria DDR existenta pe placa.

Figură 13:RAM Control Component

## Etapa 5 – Componenta principala

In aceasta etapa am creat componenta mare care instantiaza toate celalte componente mentionate mai sus. Aceata componenta contine si logica aditionala deoarce a trebuit sa vad cand sunt datele pe care vreau sa le citesc disponibile si sa pot sa le dau mai departe pt a fi serializate.

Totodata in aceasta componenta am implementat si automatul de stari finite conceput in etapa 1 de proiectare. Automatul are 4 stari: Idle in care nu se face nimic totul este pregatit pt o noua inregistrare, cand se apasa BTNU se trece in starea de Record unde vin datele de la microfon se deserialzeaza dupa se salveaza in memorie, cand au trecut 5 secunde se intra intr-o stare Inter (intermediara) unde se reintializeaza totul pentru a ne asigura ca totul e bine, iar dupa se trece automat in starea de Playback in care se citesc datele din memorie se serailzeaza si se transmit mai departe catre jack-ul de pe placa.

Pentru a putea sti cand au trecut 5 secunde (deci cand ar trebui sa fac tranzitiile) am decis sa precalculez numaru de sample uri care ar trebui facute. Formula este una simpla:

constant NR\_SAMPLES\_TO\_REC : integer := (((SECONDS\_TO\_RECORD\*PDM\_FREQ\_HZ)/NR\_OF\_BITS) - 1);

Atunci cand desreializam numaram cate sampleuri am deserializat, respectiv scris in memorie, si daca acesta este egal cu constanta definita mai sus se poate face tranzitia catre urmatoarea stare. Cand am serializar, respectiv transmis mai departe catre iesirea audio, numaram cate sampleuri s-au transmis, daca acest numar este egal cu constanta defintia mai sus se poate face tranzitia inapoi in idle.

## Etapa 6 – Componente aditionale

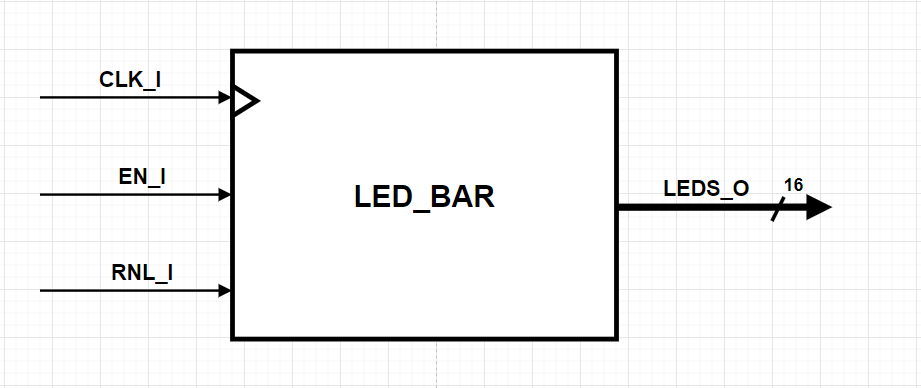
In aces pas proiectul a mers asa cum mi-am dorit insa pentru cineva care utilizeaza dispozitivul pentru prima data poate fi destul de comfuz astfel incat am decis sa implementez cateva componente care vor face interactiunea cu dizpozitivul cat mai user friendly.

1. Led Bard – este o componenta care arata dinamic utilizatorului cat timp a mai ramas din cele 5 secunde disponibile pentru a inregistra. Aceasta componenta divizeaza clock-ul dupa formula:

constant CLK\_DIV\_RATIO : integer := (((C\_SECONDS\_TO\_RECORD \* C\_SYS\_CLK\_FREQ\_MHZ \* 10^6)/16) - 1);

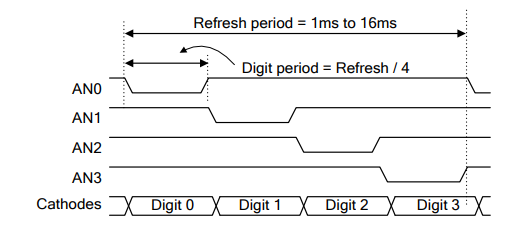
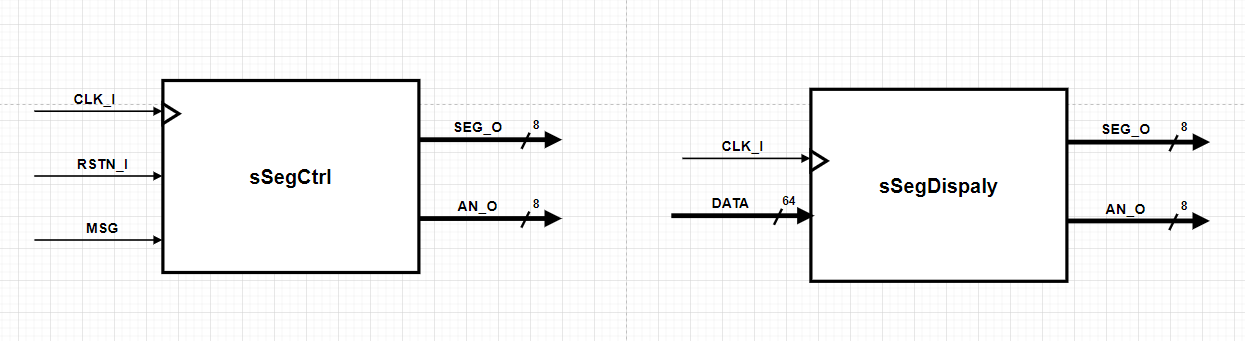
Dupa care foloseste un registru de shiftare care corespunde ledurilor care sunt aprinse, la fiecare clk\_div\_ratio se mai aprinde un led, asfel se creaza iluzia unui loading bar.

Am decis sa il fac sa functioneze in doua moduri, sa poata shifta atat la stanga cat si la dreapta. Cand inregistram se aprind ledurile de la dreapta la stanga. Iar cand suntem in starea de playback se aprind invers de la stanga la dreapta



Figură 14: Led Bar component

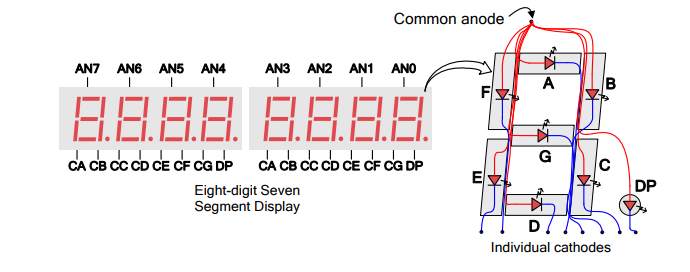
1. sSegDisplay – componenta care afiseaza pe SSD diverse mesaje pentru a putea vedea in ce stare se afla dispozitivul. Astfel incat atunci cand suntem in starea de Idle se va afisa „Press BTNU”, in starea de Record se va afisa „Rec On”, iar in starea de Playback se va afisa „Playback”.

Aceasta componenta este controlata de sSegCtrl care pregateste datele care vor fi afisate in functie de starea curenta, printr-un semnal msg venit de la componenta principala.

Figură 15: sSegCtrl and sSegDisplay components

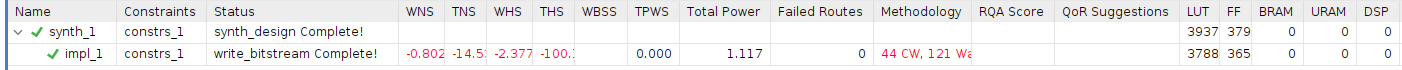
Figură 16: Common anode circuit node

Figură 17: Four digit scanning display controller timing diagram



# REZULTATE EXPERIMENTALE

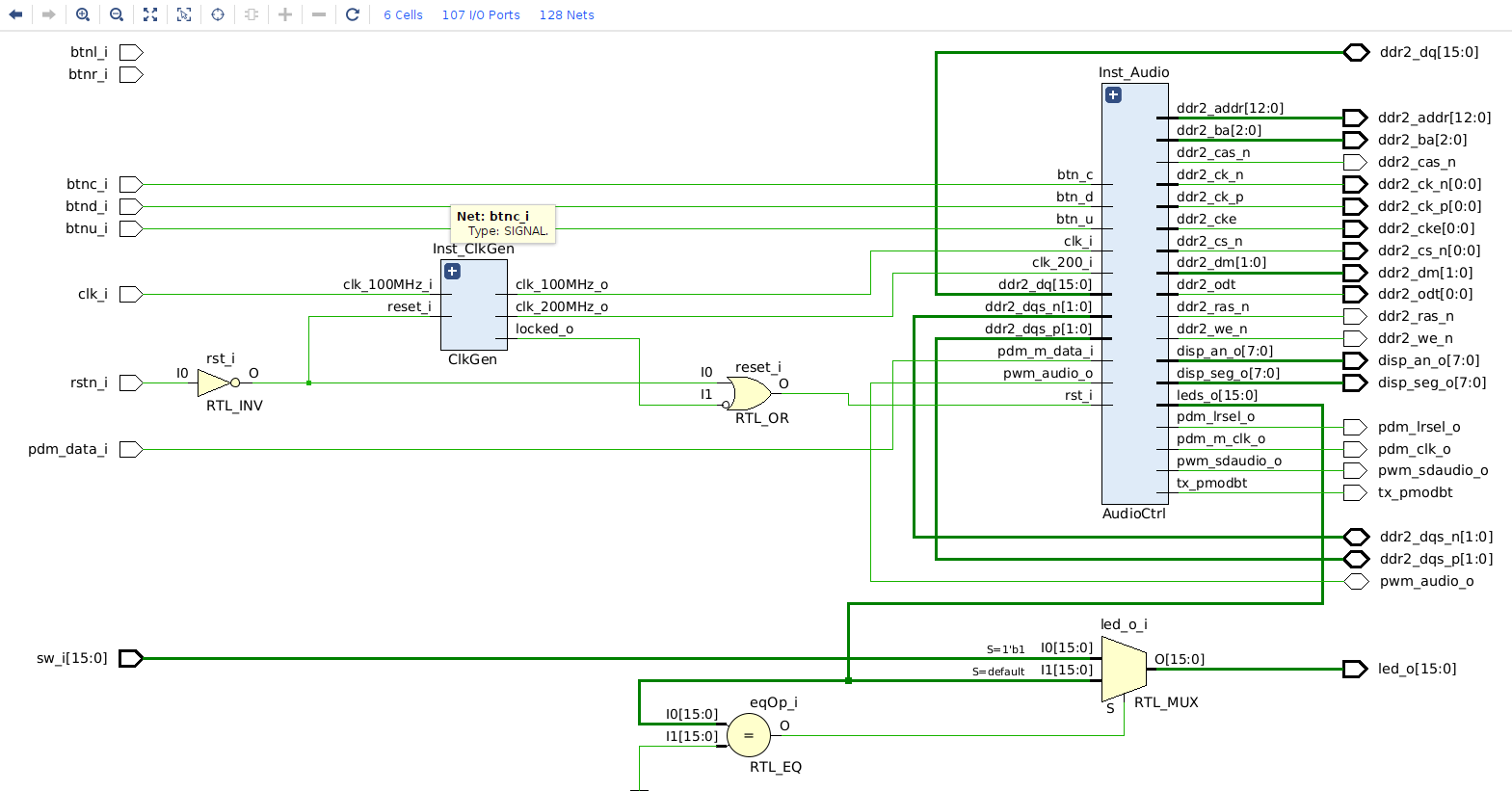
În procesul de implementare a sistemului proiectat, am folosit instrumente de proiectare care au contribuit la succesul implementării și validarea rezultatelor obținute. Limbajul de programare utilizat a fost VHDL, iar mediul software a inclus Xilinx Vivado ca sistem CAD și simulator. Platforma hardware a constat în placa Nexys 4 DDR, iar sistemul de operare folosit a fost specific pentru dispozitivul FPGA integrat.



Figură 18: Rezultatul generarii bitstream-ului

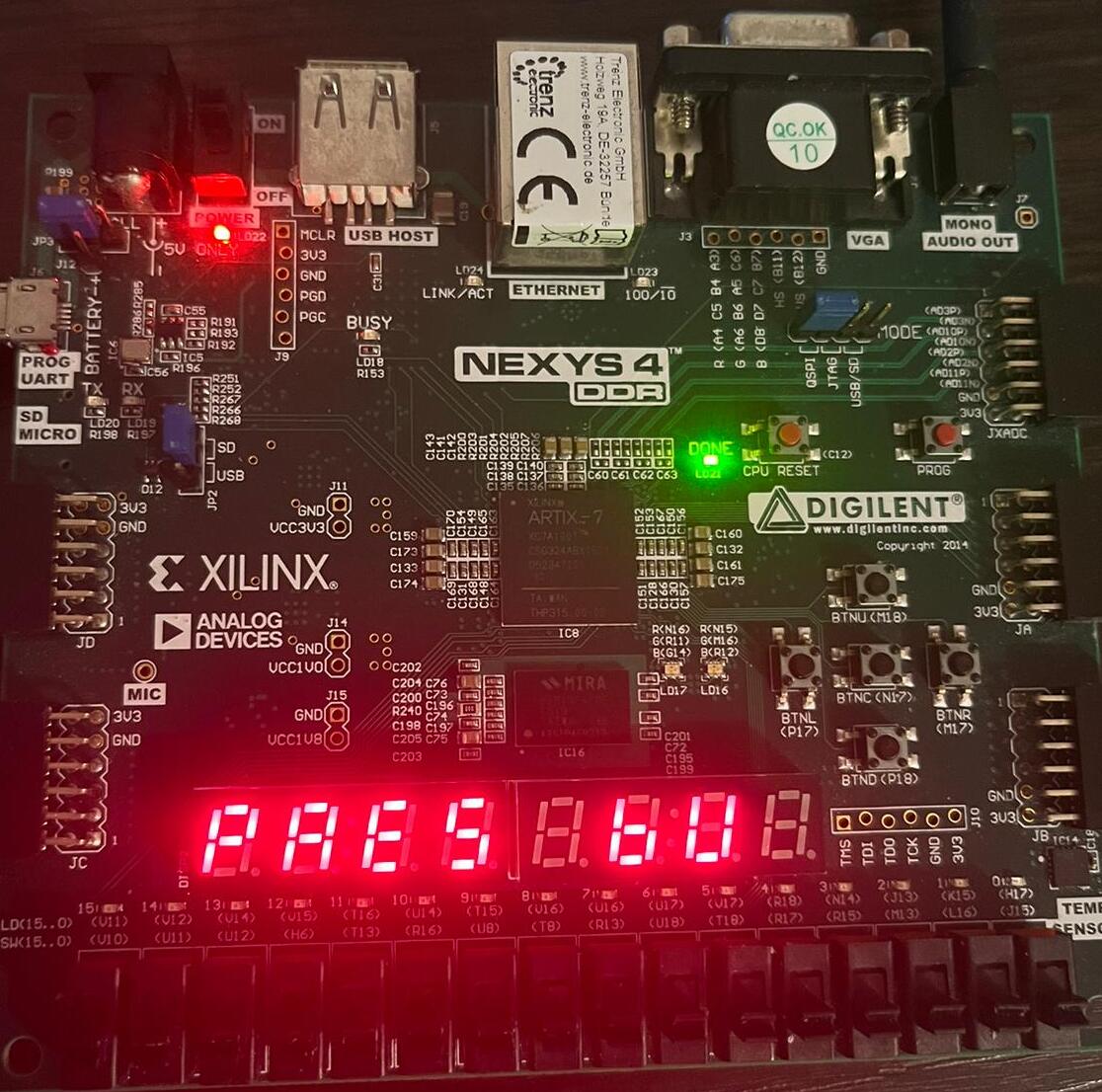
Circuitul implementat a inclus blocuri logice specifice pentru captarea semnalului audio de la microfon, prelucrarea semnalului pentru îmbunătățirea calității, și un modul dedicat pentru transmiterea datelor prin Bluetooth. Informațiile din rapoartele de implementare au fost organizate sub formă tabelară, evidențiind numărul blocurilor logice, numărul de bistabile, frecvența maximă de funcționare și alte parametri relevanți.

## **Schema RTL – pentru componenta principala**

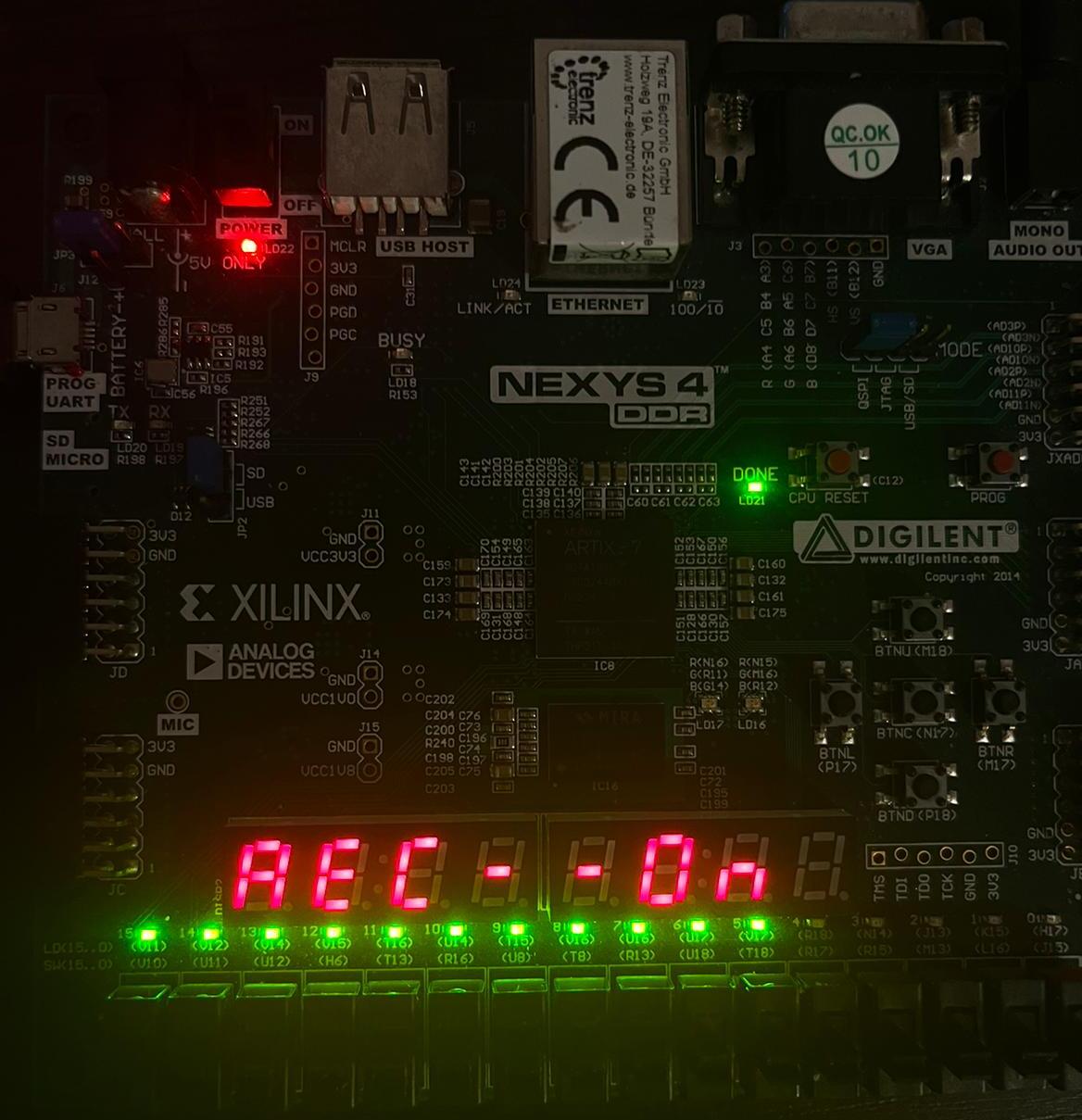
****

## **Prezentarea functionalizatilor – demo**

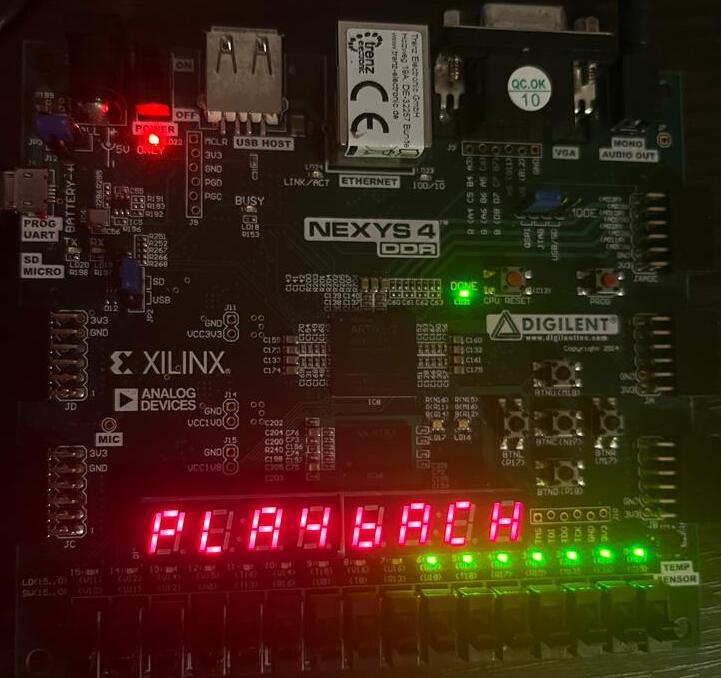
1. **Starea Idle – se așteaptă apasarea butonlui U (BTNU)**



1. **Starea Record – se observa cum ledBar-ul se încarca pe parcurs ce trece timpul și se afiseaza un nou mesaj, sunetul fiind stocat în memorie**



1. **Starea PlayBack – în aceasta stare se citește din memorie atâtea sample-uri câte s-au scris și se serializeaza, rezultatul putand fi auzit și verificat dacă conectam niște placi la jack-ul placii Nexys4**



În cadrul experimentelor realizate în contextul proiectului meu, am obținut rezultate promițătoare și semnificative care validează succesul implementării. Prin simulări detaliate și teste practice, am demonstrat că sistemul dezvoltat pe placa Nexys 4 DDR a fost capabil să înregistreze și să transmită datele audio cu precizie și eficiență. Rezultatele simulărilor au evidențiat performanța optimizată a înregistrării audio, confirmând avantajele strategiei noastre de utilizare a memoriei DDR pentru gestionarea eficientă a datelor. Testele practice au validat nu doar corectitudinea funcționalității, ci și fiabilitatea transmiterii datelor prin tehnologia Bluetooth, contribuind la atingerea obiectivelor propuse. De asemenea, am identificat și evaluat eventualele limitări și am implementat ajustări pentru a optimiza performanța generală. Aceste rezultate experimentale constituie o bază solidă pentru validarea tehnologiei propuse și evidențiază potențialul acesteia în contextul unor aplicații practice diverse.

# CONCLUZII

# Proiectul meu a reprezentat o provocare captivantă, în care am abordat cu succes problema înregistrării și transmiterii datelor audio utilizând placa Nexys 4 DDR și tehnologia Bluetooth. Obiectivul meu a fost să dezvolt o soluție inovatoare pentru captarea și transmiterea eficientă a semnalelor audio, iar rezultatele obținute confirmă îndeplinirea acestor obiective.

# În timpul implementării, am întâmpinat o provocare semnificativă legată de gestionarea volumului mare de date audio. Cu toate acestea, am reușit să găsesc o soluție ingenioasă prin utilizarea eficientă a memoriei DDR de pe placa Nexys 4 DDR. Acest aspect a contribuit semnificativ la optimizarea performanței sistemului, facilitând accesul rapid la datele înregistrate.

# În concluzie, am realizat o contribuție semnificativă la rezolvarea problemei propuse, obținând rezultate pozitive în ceea ce privește calitatea înregistrării audio insa mai putin in ceea ce priveste transmiterea eficientă prin Bluetooth. Cu toate că am depășit această provocare, recunosc existența unor aspecte care pot fi îmbunătățite, precum optimizarea generală a performanței pentru a respecta restricțiile hardware. Sunt convins că această experiență îmi va servi drept fundament pentru proiecte viitoare, iar dezvoltările ulterioare pot viza îmbunătățirea aspectelor identificate și explorarea unor aplicații extinse pentru această tehnologie inovatoare.

**BIBLIOGRAFIE**

1. NEEXYS4 Reference Manual

<https://digilent.com/reference/_media/reference/programmable-logic/nexys-4-ddr/nexys4ddr_rm.pdf>

1. NEXYS4 GPIO DEMO <https://digilent.com/reference/learn/programmable-logic/tutorials/nexys-4-basic-user-demo/start>
2. SRAM to DDR component

<https://digilent.com/reference/learn/programmable-logic/tutorials/nexys-4-ddr-sram-to-ddr-component/start>

1. FPGAs Memory Interface Solutions

<https://docs.xilinx.com/v/u/1.9-English/ug586_7Series_MIS>

1. PLL (Phase-Locked Loop)

<https://www.digikey.com/en/maker/projects/introduction-to-fpga-part-9-phaselocked-loop-pll-and-glitches/2028ce62001b4cb69335f48e127fa366>

1. Clock Wizard Tutorial

<https://www.youtube.com/watch?v=ngkpvMaNapA>