

UNIVERSITATEA TEHNICĂ „GHEORGHE ASACHI” IAȘI
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
SPECIALIZAREA CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

ACHIZIȚIA ȘI PRELUCRAREA DATELOR- PROIECT

Student: Martin Maria

Grupa: 1309A

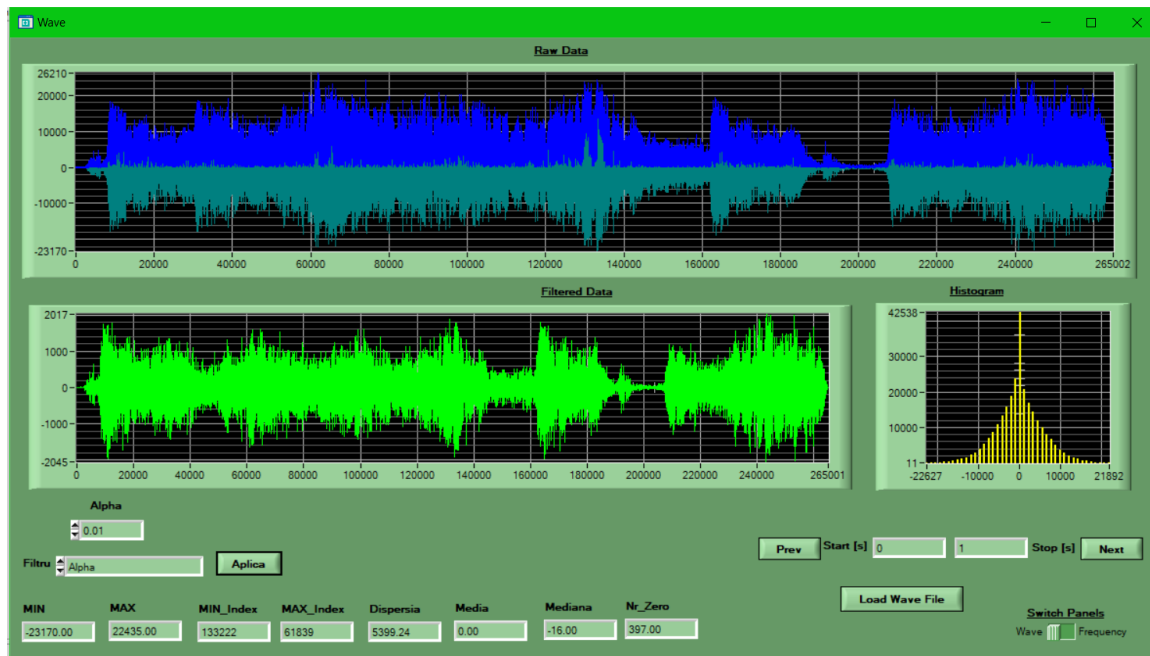
Rezumatul Proiectului:

Implementarea proiectului a fost realizată în mediul de dezvoltare LabWindows CVI, a fost realizată o interfață grafică prin intermediul căreia utilizatorul interacționează cu aplicația și a fost citit un semnal pe care l-am prelucrat. Semnalul citit a fost un fișier audio (wav) pe care l-am analizat în domeniul timp și în frecvență. Pentru analiza în domeniul timp am filtrat semnalul prin mediere și aplicând elementul de ordinul I. În domeniul frecvență am aplicat două fereștruri ce mi-au fost asignate, iar apoi am realizat filtrarea semnalului aplicând filtre cu răspuns finit și filtre cu răspuns infinit la impuls.

LabWindows/CVI este o platformă pentru dezvoltare de software cu orientare pe aplicații de instrumentație. Limbajul acceptat este ANSI C, cu extensii specifice. Mediul CVI ne pune la dispoziție un mediu interactiv de dezvoltare a unor aplicații pentru sistemul de operare Windows. Acesta îmbină avantajele programării de tip vizual cu avantajele simplității și flexibilității limbajului C.

Scopul aplicației este prelucrarea semnalelor în timp și frecvență.

Etapă I:



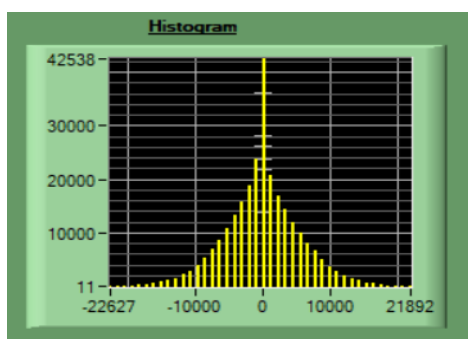
În prima etapă a proiectului a fost realizată interfața, în mediul de dezvoltare CVI, conform cerințelor, au fost citite datele din fișierul de test, am reprezentat grafic semnalul, iar apoi acesta a fost filtrat în domeniul timp.

Realizarea interfeței în LabWindows/CVI a fost necesară pentru încărcarea fișierului wav și afișarea semnalului citit pe un control de tip Graph. Fișierul wav pe care l-am folosit este numărul 41.

Pe baza acestui semnal am calculat valorile de minim și maxim din grafic, dispersia, media, mediana și numărul de treceri prin zero, iar apoi, valorile rezultate au fost afișate în interfață pe un control de tip Numeric și, cu ajutorul rezultatelor obținute, am reprezentat și histograma pe un control de tip Graph.

MIN	MAX	MIN_Index	MAX_Index	Dispersia	Media	Mediana	Nr_Zero
-23170.00	22435.00	133222	61839	5399.24	0.00	-16.00	397.00

(valorile obținute din semnalul citit)



(Histograma semnalului citit)

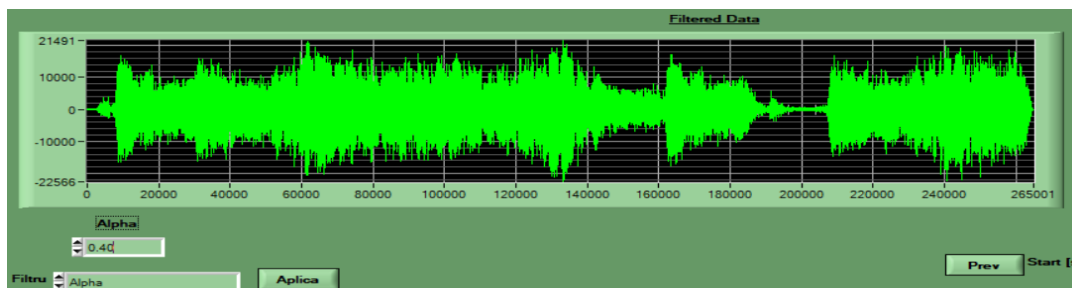
Următorul pas îl constituie filtrarea semnalului în domeniul timp, care a fost realizată prin două metode:

- prin mediere (pe 16 sau 32 de elemente)
- cu un element de ordin I, aplicând relația:

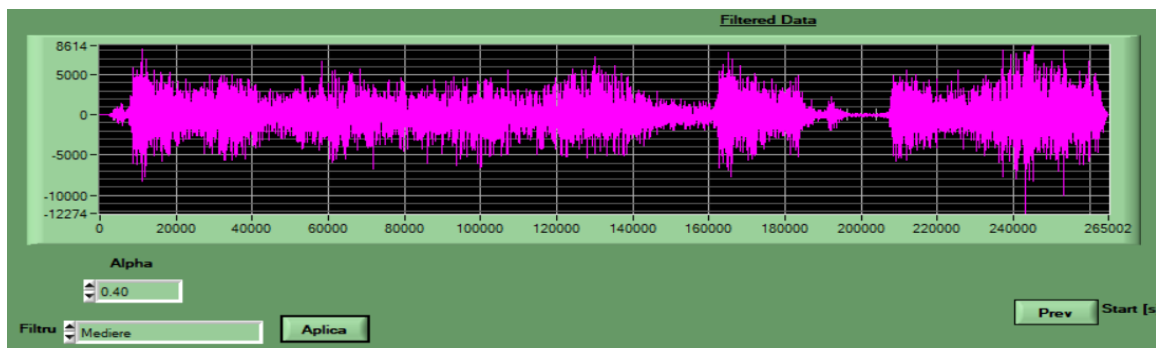
$$\text{filtru}[i] = (1 - \alpha) * \text{filtru}[i-1] + \alpha * \text{array}[i];$$

unde $\text{array}[i]$ reprezintă vectorul care conține valorile semnalului audio și $\text{filtru}[i]$ este vectorul valorilor filtrate.

Pe interfață a fost creat un control de tip Ring pentru selectarea tipului de filtru pe care dorim să-l aplicăm, iar pentru filtrul de ordin I valoarea parametrului α se va stabili prin intermediul unui control de tip Ring, valorile încadrându-se în intervalul numeric (0,1).

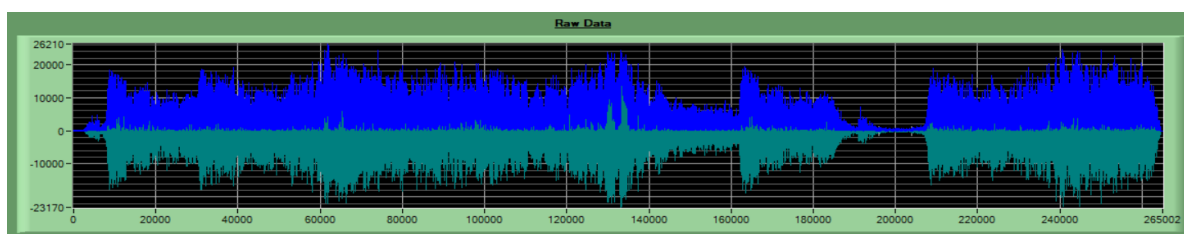


(filtru de ordin I cu $\alpha=0.40$)



(filtru mediere)

După realizarea filtrelor, am obținut li anvelopa semnalului cu ajutorul transformatei Hilbert.



(semnalul inițial și anvelopa sa)

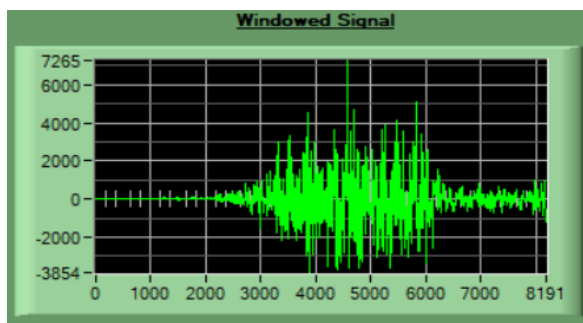
Pentru fiecare interval de timp au fost salvate imaginile obținute pentru semnalul dat și cel filtrat.

Etapă a II-a:

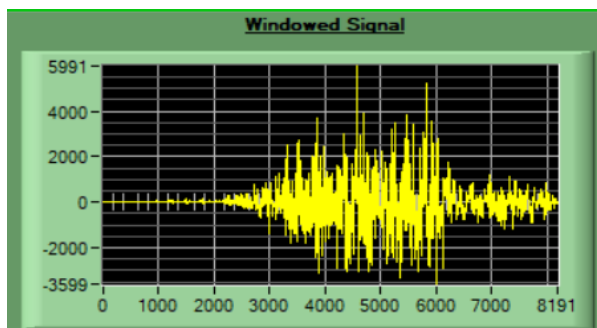
În cea de a doua etapă am extins interfața corespunzător noilor cerințe și am analizat în frecvență semnalele de test. Acestea au fost filtrate și au reprezentate spectrele lor.

Pentru început, am inclus în aplicație un nou panou pe care îl accesăm cu ajutorul unui control de tip Switch.

Pentru semnalul achiziționat am implementat două tipuri de ferestruire: *Hamming* și *Welch*. Ferestruirea unui semnal este necesară deoarece unui semnal eșantionat a cărei perioadă nu se cunoaște i se aplică o fereastră cu scopul de a îmbunătăți forma semnalului la capetele intervalului de eșantionare.

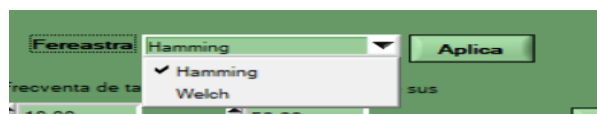


(ferestruire Hamming pe 8192 puncte)

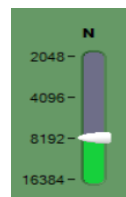


(ferestruire Welch pe 8192 puncte)

Tipul de ferestruire pe care dorim să-l aplicăm va fi selectat în interfață prin intermediul unui control de tip Ring.

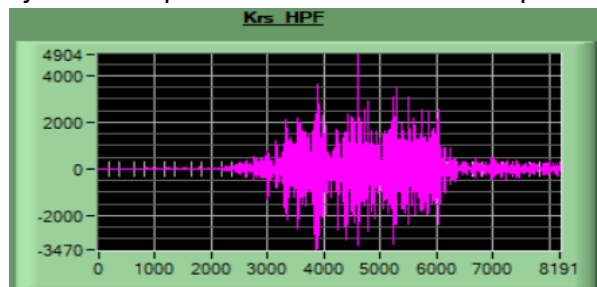


Numărul de puncte se va selecta dintr-un control de tip Ring Slide.

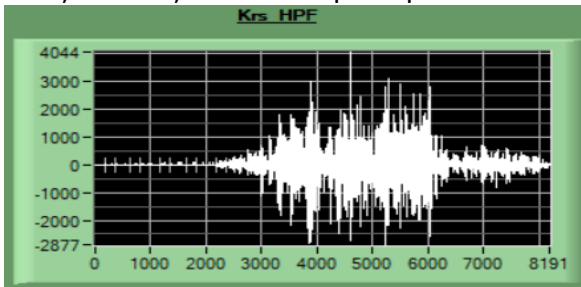


Pentru procesarea semnalului au fost aplicate două tipuri de filtre: FIR Kaiser trece sus și Butterworth trece sus.

Filtrul FIR Kaiser oferă rezultatele cele mai bune deoarece prin parametrul beta permite ajustarea raportului dintre atenuarea riplului și creșterea lățimii lobului principal.

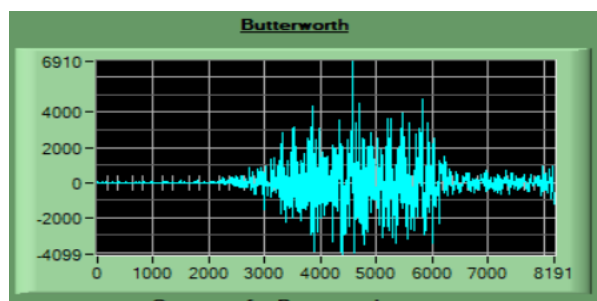


(filtru Krs_HPF Hamming pe 8192 puncte)

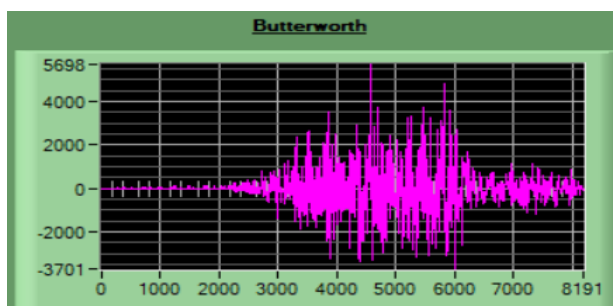


(filtru Krs_HPF Welch pe 8192 puncte)

Filtrul IIR Butterworth este unul dintre cele mai simple filtre electronice . Scopul său este să mențină modulul răspunsului de frecvență în banda de trecere cât mai plat posibil.



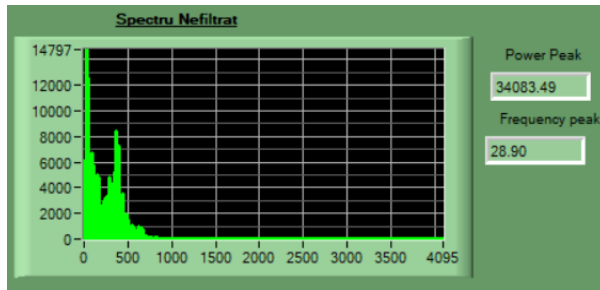
(filtru Butterworth trece sus Hamming pe 8192 puncte)



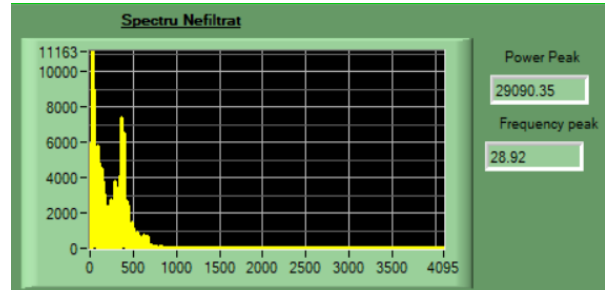
(filtru Butterworth trece sus Welch pe 8192 puncte)

Pentru a obține spectrul de putere se vor utiliza mai multe funcții CVI.

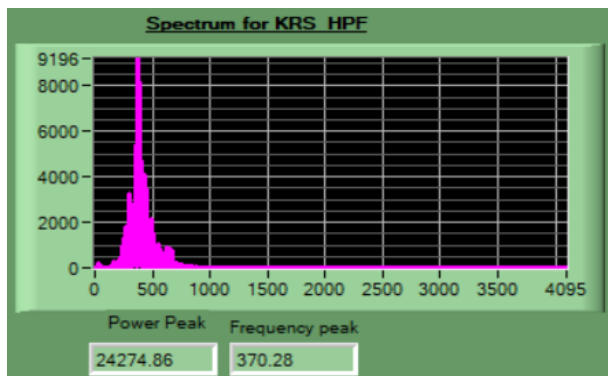
- *ScaledWindowEx*: realizează ferestruirea semnalului;
- *AutoPowerSpectrul*: calculează partea pozitivă a spectrului scalat de putere pentru un semnal eșantionat;
- *PowerFrequencyEstimate*: funcția returnează frecvența estimată pentru spectrul de putere maxim din vectorul generat de funcția precedentă (vectorul autoSpectrum) și valoarea maximă din spectrul de putere.
- *SpectrumUnitConversion*: funcția convertește spectrul de intrare în formate alternative ce permit o reprezentare grafică mai convenabilă.



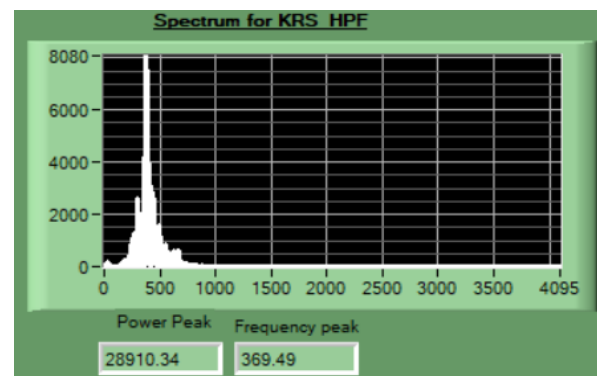
(spectru nefiltrat Hamming pe 8192 puncte)



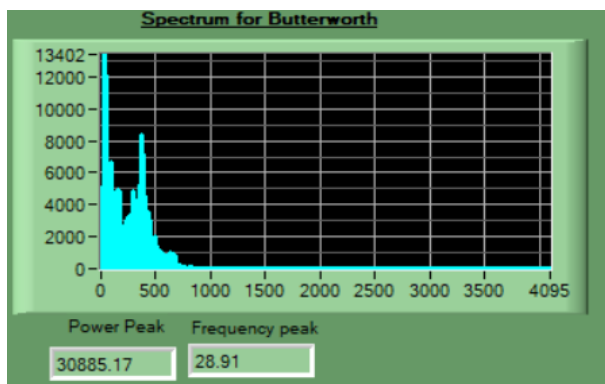
(spectru nefiltrat Welch pe 8192 puncte)



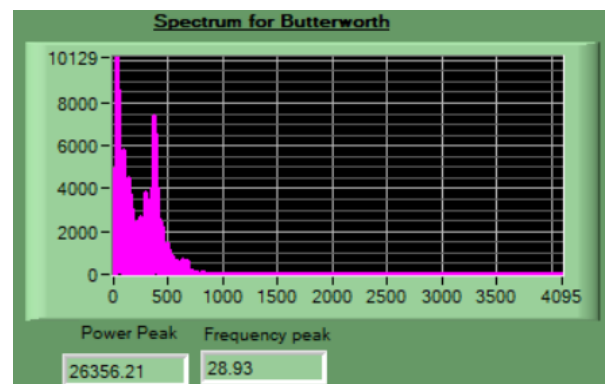
(spectru Krs_HPF Hamming pe 8192 puncte)



(spectru Krs_HPF Welch pe 8192 puncte)



(spectru Butterworth Hamming pe 8192 puncte)



(spectru Butterworth Welch pe 8192 puncte)