

26 MAJA 2023, JELENIA GÓRA

SZCZEPAŃSKI MICHAŁ 257821

GUZIK IZABELA 257841

ROK III

SEMESTR VI

SEMESTR AKADEMICKI: LETNI 2022/2023

KURS: IPV190031L

KOD GRUPY: F00-03B

GODZINY ZAJĘĆ KURSU: 11:50-14:20

**PROCESORY SYGNAŁOWE
NR ĆWICZENIA - 4**

FILTRY O SKOŃCZONEJ ODPOWIEDZI IMPULSOWEJ

PROWADZĄCY: DR INŻ. KRZYSZTOF KARDACH

CEL ĆWICZENIA

Celami czwartego zadania laboratoryjnego jest zapoznanie się z filtrami, rodzajami i efektami ich działania na sygnałe audio, opanowanie zdolności uruchomienia i operowania na filtrach FIR na płytce TMS320C5515 USB Stick, podgląd oraz ocena działania wskazanych filtrów FIR na podstawie rejestracji sygnałów i ich widma.

PRZEBIEG ĆWICZENIA

A. Utworzenie <tekst>

UŻYTY SPRZĘT

- Środowisko Code Composer Studio 3.7
- 1 x smartphone
- 1 x przewód typu Jack
- 1 x przewód typu USB
- 1 x słuchawki douszne
- 1 x moduł TMS320C5515 USB Stick

ANALIZA PROGRAMU - BIBLIOTEKI

```
62 #include "stdio.h"
63 #include "usbstk5505.h"
64 #include "aic3204.h"
65 #include "PLL.h"
66 #include "FIR_Filters_asm.h"
67 #include "LEDFlasher.h"
68 #include "hamming.h"
69 #include "hanning.h"
70 #include "kaiser.h"
71 #include "rectangular.h"
72 #include "stereo.h"
```

stdio.h, usbstk5505.h, aic3204.h oraz PLL.h były używane oraz opisywane na wcześniejszych zajęciach laboratoryjnych.

FIR_Filters_asm.h - biblioteka nagłówkowa odpowiedzialna za filtr z odpowiedzią na impuls skończoną w czasie.

LEDFlasher.h - biblioteka nagłówkowa dotycząca LED-ów na płycie.

hamming.h - biblioteka nagłówkowa filtru Hamminga.

hanning.h - biblioteka nagłówkowa filtru Hanninga.

kaiser.h - biblioteka nagłówkowa filtru Kaisera.

rectangular.h - biblioteka nagłówkowa filtru prostokątnego.

ANALIZA PROGRAMU - ZMIENNE GLOBALNE

```
74 Int16 left_input;
75 Int16 right_input;
76 Int16 left_output;
77 Int16 right_output;
78 Int16 mono_input;
79 Int16 idx;
80 Int16 left[480];
81 Int16 right[480];
```

Int16 left_input, right_input, left_output, right_output to zmienne globalne zadeklarowane dla wejść/wyjść kanałów prawego i lewego.

Int16 mono_input to deklaracja wejściowego sygnału mono.

Int16 idx to deklaracja indeksu, a Int16 left[480] i right[480] to deklaracja częstotliwości 480Hz dla kanałów.

```
83 #define SAMPLES_PER_SECOND 48000
84 #define GAIN_IN_dB 30
85
86 unsigned long int i = 0;
87 unsigned int step = 0;
```

#define SAMPLES_PER_SECOND - deklaracja zmiennej próbek na sekundę o wartości 48kHz.
#define GAIN_IN_DB - deklaracja zmiennej globalnej wzmocnienia o wartości 30 dB.

ANALIZA PROGRAMU - MODYFIKACJA

```
74 Int16 left_input;
75 Int16 right_input;
76 Int16 left_output;
77 Int16 right_output;
78 Int16 mono_input;
79 Int16 idx;
80 Int16 left0[1024];
81 Int16 right0[1024];
82 Int16 left500[1024];
83 Int16 right500[1024];
84 Int16 left1000[1024];
85 Int16 right1000[1024];
86 Int16 left2000[1024];
87 Int16 right2000[1024];
88 Int16 left3000[1024];
89 Int16 right3000[1024];
```

Do ćwiczenia zadeklarowaliśmy również tablice pomocnicze w celu wygenerowania 10 wykresów na raz.

Int16 left0 oraz right0 przechowują wartości bez filtru.

ANALIZA PROGRAMU - MAIN - FLASHES

```
puts("1 Flash    = Straight through, no processing");
puts("2 Flashes = Hamming filter 1000 Hz");
puts("3 Flashes = Hanning filter 1000 Hz");
puts("4 Flashes = Kaiser filter 1000 Hz");
puts("5 Flashes = Rectangular filter 1000 Hz");
```

ANALIZA PROGRAMU - MAIN - SWITCH CASE

```
step = LEDFlasher(5); // 5 Different filter settings.

switch (step)
{
    case 1:
        left_output = mono_input; // Straight through, no processing.
        right_output = mono_input;
        left0[idx] = left_output;
        right0[idx] = right_output;
        idx=(1+idx)%1024;
        break;

    case 2:
        left_output = FIR_filter_asm(&Hamming_Low_Pass_Filter_1000Hz[0],
                                     mono_input);
        right_output = FIR_filter_asm_2(&Hamming_High_Pass_Filter_1000Hz[0],
                                       mono_input);
        left500[idx] = left_output;
        right500[idx] = right_output;
        idx=(1+idx)%1024;
        break;

    case 3:
        left_output = FIR_filter_asm(&Hanning_Low_Pass_Filter_1000Hz[0],
                                     mono_input);
        right_output = FIR_filter_asm_2(&Hanning_High_Pass_Filter_1000Hz[0],
                                       mono_input);
        left1000[idx] = left_output;
        right1000[idx] = right_output;
        idx=(1+idx)%1024;
        break;
```



```

case 4:
    left_output = FIR_filter_asm(&Kaiser_Low_Pass_Filter_1000Hz[0],
        mono_input);
    right_output = FIR_filter_asm_2(&Kaiser_High_Pass_Filter_1000Hz[0],
        mono_input);
    left2000[idx] = left_output;
    right2000[idx] = right_output;
    idx=(1+idx)%1024;
break;

case 5:
    left_output = FIR_filter_asm(&Rectangular_Low_Pass_Filter_1000Hz[0],
        mono_input);
    right_output = FIR_filter_asm_2(&Rectangular_High_Pass_Filter_1000Hz[0],
        mono_input);
    left3000[idx] = left_output;
    right3000[idx] = right_output;
    idx=(1+idx)%1024;
break;
}

aic3204_codec_write(left_output, right_output);
}

```

<tekst>

SYGNAŁ PRAWY I LEWY DLA CASE 2

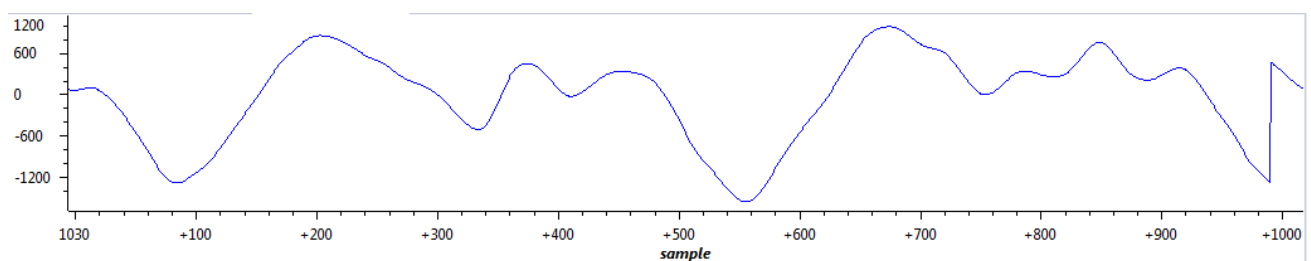
Case 2 code

```

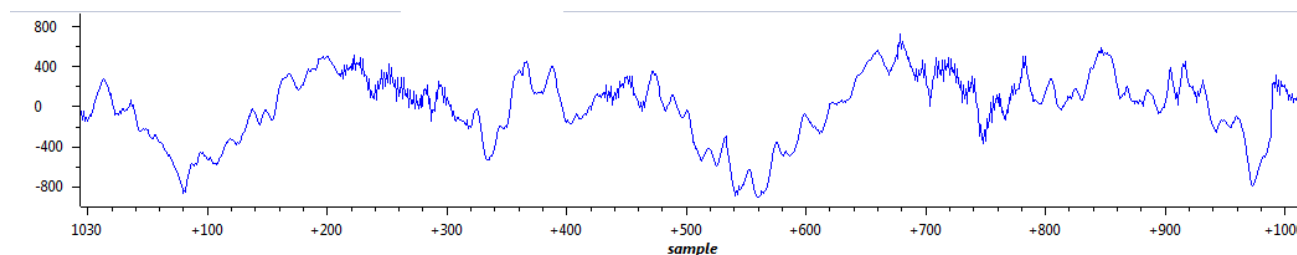
case 2:
    left_output = FIR_filter_asm(&Hamming_Low_Pass_Filter_500Hz[0], mono_input);
    right_output = FIR_filter_asm_2(&Hamming_High_Pass_Filter_500Hz[0], mono_input);
    left[idx] = left_output;
    right[idx] = right_output;
    idx=(1+idx)%1024;
break;

```

Sygnał lewy

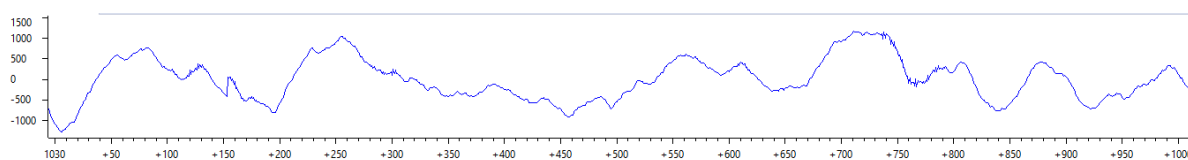


Sygnal prawy

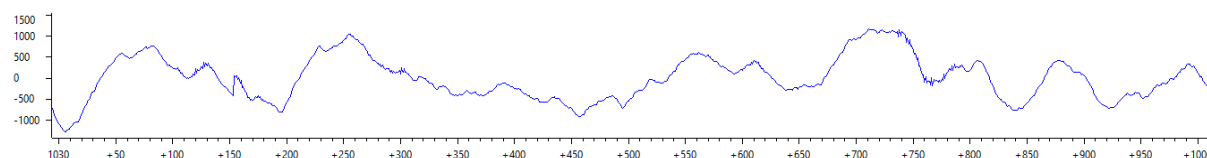


TESTOWANIE SYGNAŁU SZUMU BIAŁEGO

Sygnal lewy 1

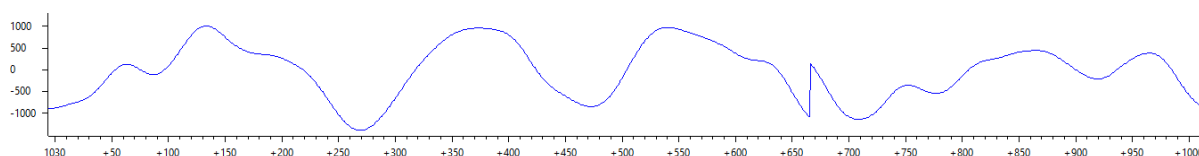


Sygnal prawy 1



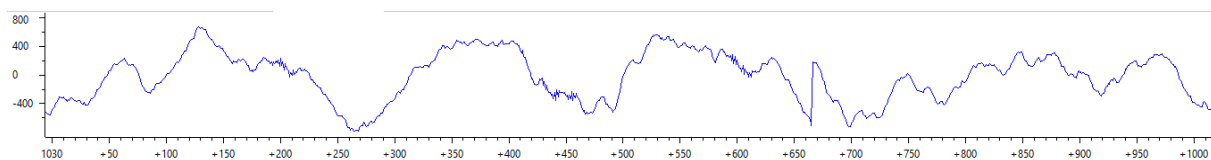
Spostrzeżenie: wykresy są identyczne, ponieważ w tym module nie występuje żaden filtr. Dźwięk w słuchawkach ma tę samą głośność.

Sygnal lewy 2



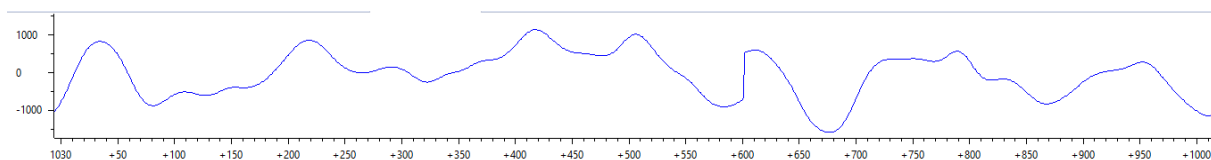
Sygnal prawy 2

FILTRY O SKOŃCZONEJ ODPOWIEDZI IMPULSOWEJ

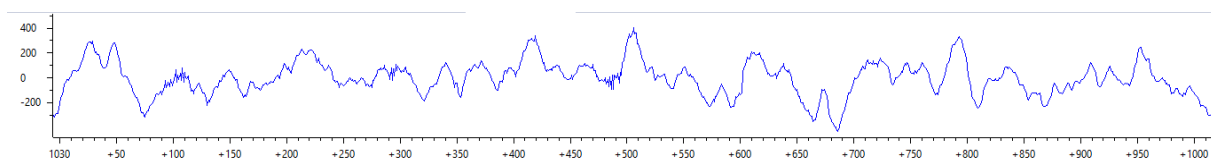


Spostrzeżenie: prawa słuchawka jest cichsza.

Sygnal lewy 3

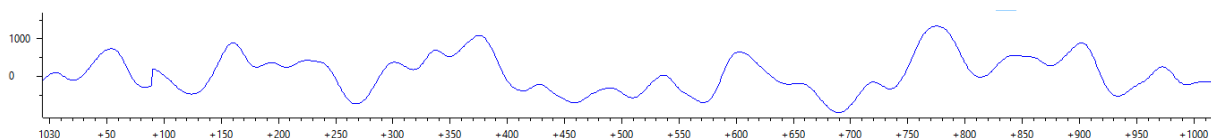


Sygnal prawy 3



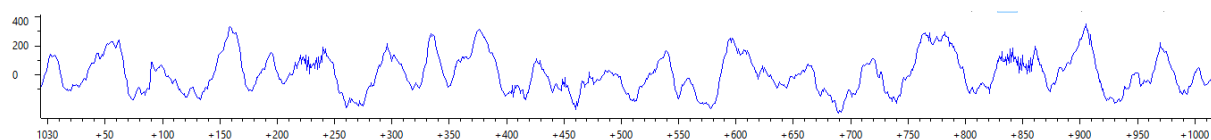
Spostrzeżenie: lewa słuchawka ma mocniejszy dźwięk.

Sygnal lewy 4



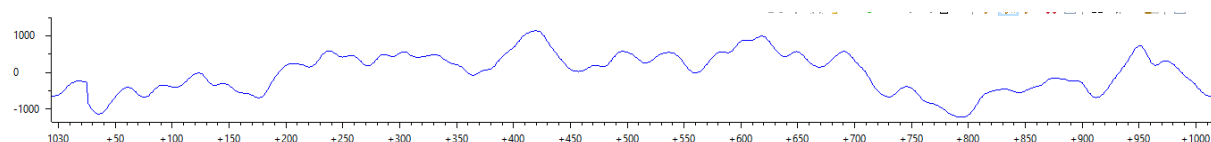
Sygnal prawy 4

FILTRY O SKOŃCZONEJ ODPOWIEDZI IMPULSOWEJ

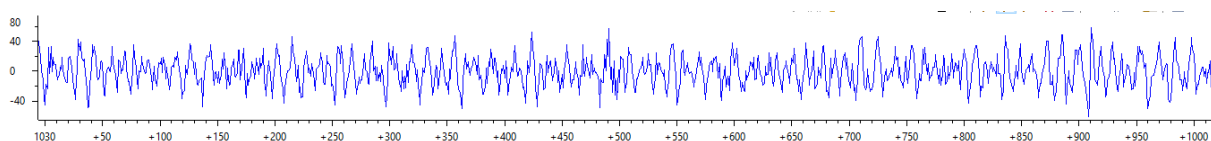


Spostrzeżenie: Lewa słuchawka jest cichsza/dźwięk jest delikatniejszy.

Sygnal lewy 5



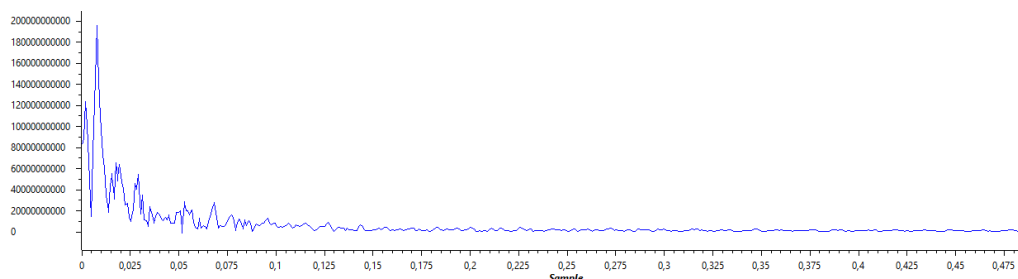
Sygnal prawy 5



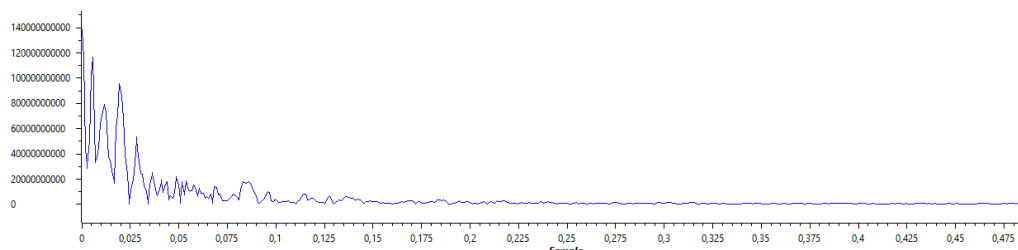
Spostrzeżenie: Dźwięk w prawej słuchawce brzmi bardzo głośno.

TRANSFORMATA FOURIERA - SZUM BIAŁY - FILTR 1000 HZ

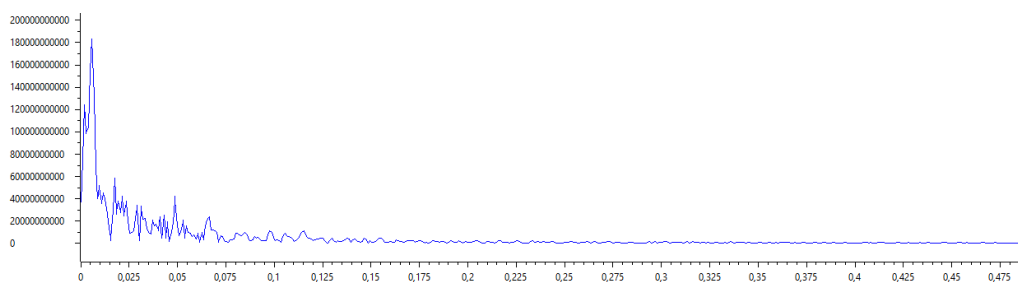
Okno Hamminga - wykres lewy



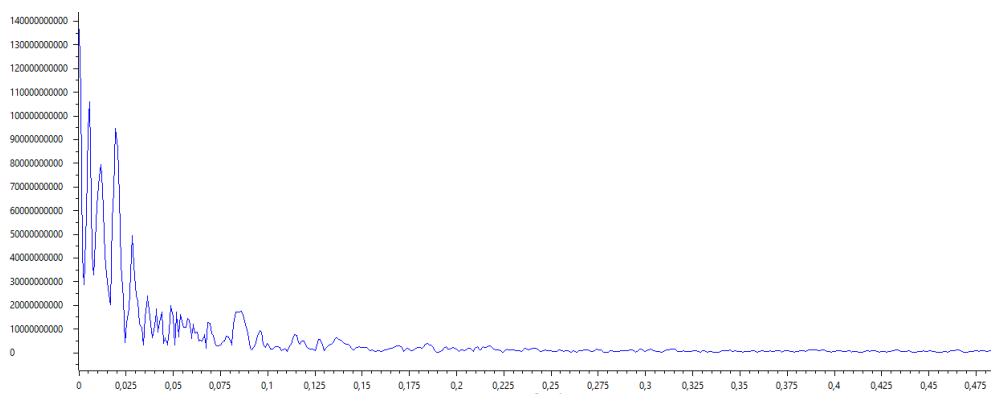
Okno Hamminga - wykres prawy



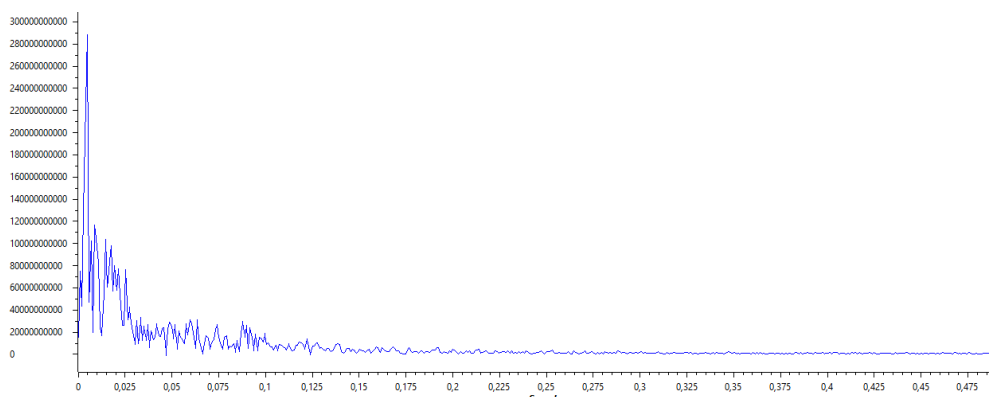
Okno Hanninga - wykres lewy



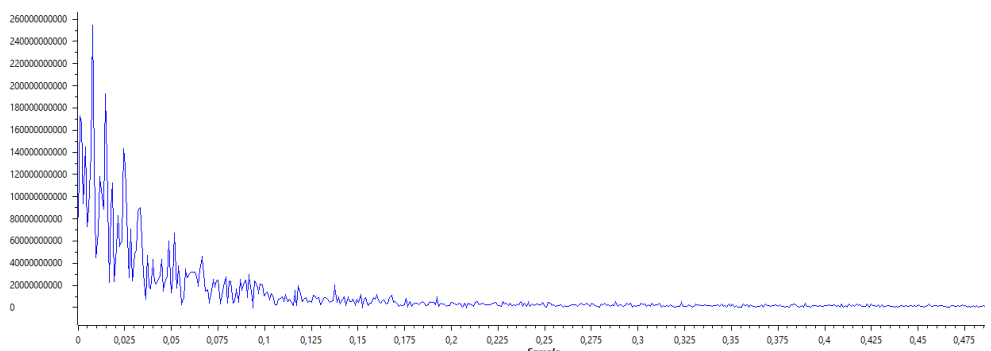
Okno Hanninga - wykres prawy



Okno Rectangle - wykres lewy



Okno Rectangle - wykres prawy



ODPOWIEDZI NA PYTANIA

1. Omawiane filtry nazywane są filtrami o skończonej odpowiedzi impulsowej. Jaka jest motywacja słowa „skończona”?

- Motywacja słowa „skończona” wynika z charakterystyki tych filtrów. FIR to typ filtrów cyfrowych, które charakteryzuje skończona w czasie odpowiedź na impuls. Czyli po nałożeniu impulsu na filtr FIR, jego odpowiedź będzie miała określoną długość w czasie i po pewnym czasie zniknie. Słowo „skończona” oznacza, że wejście (impuls) rozprzestrzeni się przez filtr FIR przez ograniczoną ilość czasu do momentu gdy zniknie. Przeciwnie do pracy filtrów o nieskończonej odpowiedzi impulsowej IIR mające skomplikowane charakterystyki czasowe FIR mają odpowiedź impulsową ograniczoną do określonej ilości próbek.

2. Czym jest funkcja okna stosowana i dlaczego jest stosowana w projektowaniu filtrów cyfrowych?

- Funkcja okna to technika używana w projektowaniu filtrów cyfrowych, szczególnie w dziedzinie przetwarzania sygnałów. Polega na mnożeniu próbek sygnału wejściowego przez pewną funkcję okna, która ma na celu zmniejszenie efektów ubocznych filtracji, takich jak przecieki częstotliwościowe i artefakty wynikające z charakterystyki filtru. Głównym powodem stosowania funkcji okna w projektowaniu filtrów cyfrowych jest efekt niepożądanych artefaktów, które mogą pojawić się w wyniku filtracji. Filtry cyfrowe, zwłaszcza te o skończonej odpowiedzi impulsowej (FIR), wprowadzają zjawisko znane jako "zjawisko przecieku częstotliwościowego". Oznacza to, że część energii sygnału wejściowego o wysokich częstotliwościach "przecieka" na niższe częstotliwości, powodując zniekształcenia w wyniku filtracji. Funkcje okna są stosowane w celu ograniczenia tego zjawiska przecieku częstotliwościowego i innych artefaktów wynikających z charakterystyki filtru. Funkcje okna są zazwyczaj symetrycznymi, dodatnimi i skończonymi funkcjami, które są mnożone przez próbki sygnału przed nałożeniem filtra. Mnożenie próbek sygnału przez funkcję okna powoduje stopniowe zmniejszanie amplitudy sygnału na brzegach próbek, co prowadzi do redukcji efektów ubocznych.

Istnieje wiele różnych funkcji okna, takich jak okno prostokątne, okno Hamminga, okno Blackmana, okno Hanninga, okno Kaiser-Bessela i wiele innych. Każde okno ma inne charakterystyki, które wpływają na zakres redukcji artefaktów i kompromis między szerokością pasma a tłumieniem. Wybór konkretnej funkcji okna zależy od konkretnych wymagań projektu i żądanej charakterystyki filtru.

3. W jaki sposób okno można użyć do polepszenia jakości filtru?

- Funkcje okna są używane w celu polepszenia jakości filtru cyfrowego poprzez zmniejszenie efektów ubocznych, takich jak przecieki częstotliwościowe i artefakty wynikające z charakterystyki filtru. Oto kilka sposobów, w jaki okno może przyczynić się do poprawy jakości filtracji:

a) Redukcja przecieków częstotliwościowych: Filtry cyfrowe, zwłaszcza te o skończonej odpowiedzi impulsowej (FIR), mają charakterystykę idealną, która jest niewykonalna w praktyce. Zjawisko przecieku częstotliwościowego powoduje, że część energii sygnału wejściowego o wysokich częstotliwościach "przecieka" na niższe częstotliwości. Funkcje okna stopniowo zmniejszają amplitudę sygnału na brzegach próbek, co pomaga ograniczyć przecieki częstotliwościowe i poprawić dokładność filtracji.

b) Tłumienie boczne: Charakterystyka filtru cyfrowego może powodować pojawienie się tłumienia bocznego w pasmach przenoszenia i tłumienia. Funkcje okna mogą być tak dobrane, aby zapewnić odpowiednie tłumienie boczne, minimalizując niepożądane efekty w tych obszarach i poprawiając jakość filtracji.

c) Ograniczenie efektów brzegowych: Filtry cyfrowe, szczególnie te o skończonej odpowiedzi impulsowej, mogą wprowadzać artefakty na brzegach sygnału wynikające z niepełnego opóźnienia. Funkcje okna stopniowo zmniejszają amplitudę sygnału na brzegach, co zmniejsza wpływ tych artefaktów na sygnał wynikowy.

d) Minimalizacja przejścia: W przypadku filtrów o konkretnych wymaganiach projektowych, funkcje okna mogą być używane do minimalizacji szerokości pasma przejściowego między pasmem przenoszenia a pasmem tłumienia. Poprzez odpowiedni dobór funkcji okna można zmniejszyć szerokość pasma przejściowego, co przekłada się na lepszą selektywność filtru.

e) Poprawa charakterystyki amplitudowo-częstotliwościowej: Funkcje okna mogą być dostosowane w taki sposób, aby zmieniać charakterystykę amplitudowo-częstotliwościową filtru. Na przykład, funkcje okna Kaiser-Bessela są często stosowane do projektowania filtrów o precyzyjnie określonej charakterystyce tłumienia i szerokości pasma przejściowego.

4. Omów co najmniej dwie metody obliczania współczynników filtru.

- **Metoda FIR** z wykorzystaniem odpowiedzi impulsowej, która opiera się na bezpośrednim obliczaniu odpowiedzi impulsowej filtru cyfrowego, a następnie przeliczaniu jej na współczynniki filtru. Oto kroki tej metody:

- a) Określenie pożądanej charakterystyki filtru w dziedzinie częstotliwości (np. dolnoprzepustowy, górnoprzepustowy, pasmowoprzepustowy).
- b) Przeliczenie charakterystyki filtru na odpowiedź impulsową filtru w dziedzinie czasu.
- c) Dyskretyzacja odpowiedzi impulsowej, czyli próbkowanie jej w odpowiednich punktach czasowych.
- d) Przeliczenie dyskretnych próbek odpowiedzi impulsowej na współczynniki filtru.

Ta metoda umożliwia precyzyjne określenie charakterystyki filtru, ponieważ opiera się na bezpośrednich pomiarach odpowiedzi impulsowej. Jednak jest bardziej pracochłonna i kosztowna obliczeniowo niż inne metody.

- **Metoda okienkowa** (np. metoda okna Hamminga), która polega na projektowaniu filtru poprzez odpowiednie okienkowanie charakterystyki idealnej filtru cyfrowego. Oto kroki tej metody:

- a) Określenie pożądanej charakterystyki filtru w dziedzinie częstotliwości.
- b) Obliczenie charakterystyki idealnej filtru, która spełnia żądane wymagania.
- c) Wybór funkcji okna, np. okno Hamminga, które zostanie zastosowane do charakterystyki idealnej filtru.

d) Pomnożenie charakterystyki idealnej filtru przez funkcję okna, aby otrzymać ostateczną charakterystykę filtru.

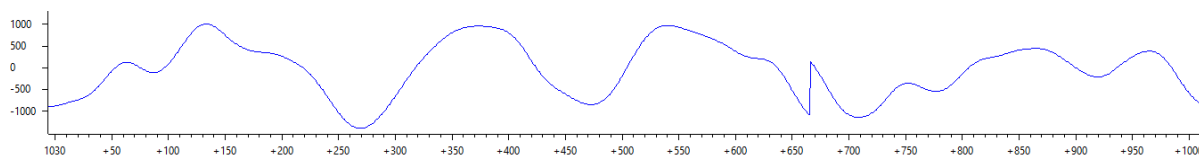
e) Przeliczenie charakterystyki filtru na współczynniki filtru, np. poprzez zastosowanie odwrotnej transformacji Fouriera.

Metoda okienkowa jest bardziej elastyczna i łatwiejsza do zastosowania niż metoda oparta na odpowiedzi impulsowej. Funkcje okna pozwalają kontrolować kompromis między szerokością pasma przejściowego a tłumieniem bocznym, co daje większą elastyczność projektowania filtrów cyfrowych.

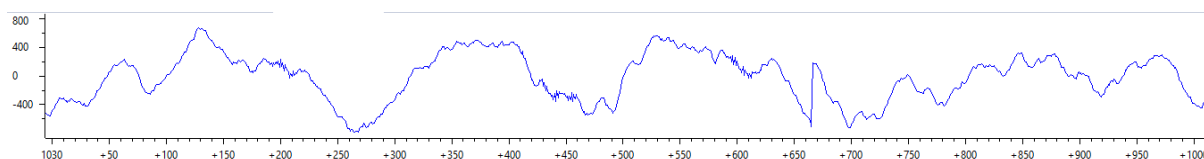
Obie te metody mają swoje zalety i wady, a wybór odpowiedniej metody zależy od konkretnych wymagań projektowych, charakterystyki filtru i dostępnych narzędzi obliczeniowych.

PORÓWNIANIE CZĘSTOTLIWOŚCIOWYCH CHARAKTERYSTYK FILTRA 500HZ DLA CZĘSTOTLIWOŚCI PRÓBKOWANIA 48000 HZ ORĄZ 12000 HZ

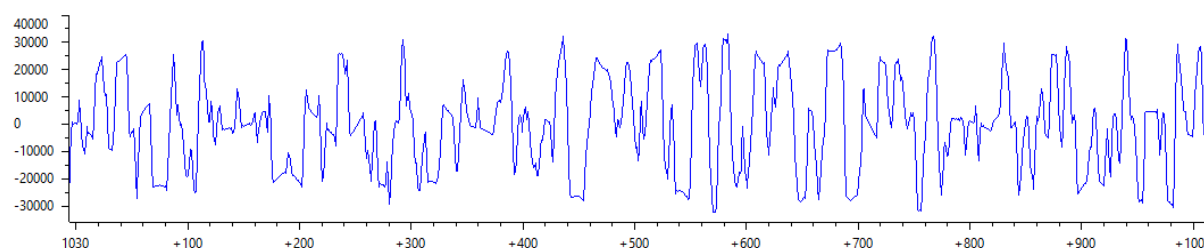
Wykres lewy - 48000 Hz



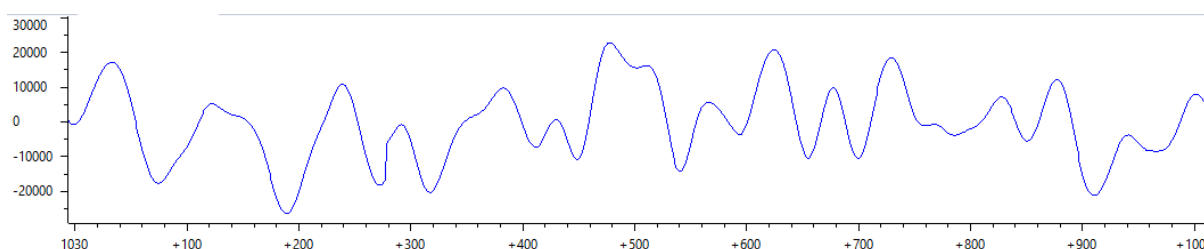
Wykres prawy - 48000 Hz



Wykres lewy - 12000 Hz



Wykres prawy - 12000 Hz



Zmniejszenie częstotliwości próbkowania może mieć wpływ na częstotliwościowe charakterystyki filtra FIR. Oto kilka aspektów, które warto wziąć pod uwagę:

1) Zmiana pasma przejściowego: Częstotliwość próbkowania jest ściśle związana z

maksymalną częstotliwością, jaką filtr FIR może skutecznie przetwarzać. Zmniejszenie częstotliwości próbkowania spowoduje zmniejszenie maksymalnej możliwej częstotliwości, która może być poprawnie filtrowana przez filtr FIR. W rezultacie pasmo przejściowe filtra może się skompresować lub przesunąć w niższe częstotliwości.

2) Zmiana pasma zaporowego: Zmniejszenie częstotliwości próbkowania może również wpływać na pasmo zaporowe filtra FIR. Pasmo zaporowe jest obszarem, w którym filtr powinien tłumić sygnały. Zmniejszenie częstotliwości próbkowania może spowodować, że pasmo zaporowe będzie musiało być poszerzone, aby uwzględnić niższe częstotliwości.

3) Aliasing: Zmniejszenie częstotliwości próbkowania może zwiększyć ryzyko wystąpienia zjawiska aliasingu. Aliasing to zniekształcenie sygnału spowodowane niewłaściwym próbkowaniem, które powoduje, że wyższe częstotliwości są błędnie reprezentowane jako niższe. Aby uniknąć aliasingu, należy stosować odpowiednie filtry antyaliasingowe przed próbkowaniem lub zwiększyć częstotliwość próbkowania.

Ważne jest, aby uwzględnić zmiany częstotliwości próbkowania i ich wpływ na charakterystyki filtra FIR podczas projektowania i analizy. Zmniejszenie częstotliwości próbkowania ma bezpośredni wpływ na możliwości filtracji i ograniczenia w zakresie częstotliwościowe. Należy odpowiednio dostosować parametry filtra FIR, takie jak długość filtra i częstotliwość odcięcia, aby osiągnąć pożądane efekty filtrowania w kontekście zmniejszonej częstotliwości próbkowania.