

Jacek Misiurewicz
Krzysztof Kulpa
Piotr Samczyński
Mateusz Malanowski
Piotr Krysik
Łukasz Maślikowski
Damian Gromek
Artur Gromek
Marcin K. Bączyk

Zakład Teorii Obwodów i Sygnałów
Instytut Systemów Elektronicznych
Wydział Elektroniki i Technik Informacyjnych
Politechnika Warszawska

Laboratorium Cyfrowego Przetwarzania Sygnałów

Wersja do wydruku - bez części teoretycznej

Próbkowanie, odtwarzanie, decymacja, interpolacja

Część teoretyczną w tej wersji opuszczono.

1.2. Zadania do pracy własnej studenta

Podobne zadania mogą znaleźć się na wejściówce. Nie dotyczy to zadań oznaczonych tu jako „trudne”.

1) Jakie pasmo zajmuje sygnał akustyczny (ew. skorzystaj z literatury/internetu)? Jaka z tego wynika minimalna częstotliwość próbkowania?

2) Nadajniki radia FM nadają w paśmie od 88 do 108 MHz. Rozważ realizację odbiornika FM z bezpośrednim przetwarzaniem A/C. Odbiornik taki składa się ze wzmacniacza niskoszumnego, po którym próbkuje się od razu sygnał o częstotliwości radiowej. Dalszy ciąg przetwarzania (zaczynając od filtru pasmowego wybierającego stację) odbywa się cyfrowo.

- Jaka jest minimalna częstotliwość próbkowania takiego sygnału przy próbkowaniu dolnopasmowym?
 - trudne* Dobierz możliwie małą częstotliwość próbkowania przy podpróbkowaniu (próbkowaniu środkowopasmowym).
 - Jeśli nie zrobisz zadania b), przyjmij $f_s = 80 \text{ MHz}$.* Naszkicuj widmo sygnału spróbkowanego środkowopasmowo, aby sprawdzić, że dobór jest poprawny (nie powoduje aliasingu). Ogranicz swój szkic do zakresu np. od -120 do 120 MHz
- 3) Sygnał sinusoidalny o częstotliwości 6 kHz został spróbkowany z częstotliwością próbkowania $f_s = 48 \text{ kHz}$.

- Wyznacz okres sygnału dyskretnego, unormowaną częstotliwość f_n i unormowaną częstość kołową θ .
- Jak zmieni się częstotliwość unormowana, gdy sygnał zdecymujemy dwukrotnie (tj. zredukujemy próbki ze współczynnikiem redukcji równym 2, przedtem dokonując odpowiedniej filtracji antyaliasingowej)?
- Jaki sygnał otrzymamy, gdy dokonamy ekspansji próbek (z idealną interpolacją) ze współczynnikiem ekspansji równym 2?
- trudne* Jaki sygnał otrzymamy przy ekspansji bez interpolacji (rozważ sytuację z dodaniem zer lub z powieleniem)?

4) Wartość sygnału $x_{a'}$ odtworzonego w ustalonym punkcie t jest kombinacją liniową próbek sygnału $x[n]$

$$x_{a'}(t) = \sum_{n=-\infty}^{\infty} x[n] \cdot a_n,$$

gdzie współczynniki a_n można wyliczyć ze wzoru

$$x_{a'}(t) = \sum_{n=-\infty}^{\infty} x[n] \cdot \text{Sa}\left(\frac{t - nT_s}{T_s}\right) \quad (1.1)$$

Oblicz wartości współczynników a_i dla $i = -3 \dots 3$, za pomocą których oblicza się:

- wartość $x_{a'}(t)$ w punkcie $t = 0$ i w punkcie $t = T_s$,
- wartość $x_{a'}(t)$ w punkcie $t = T_s/2$, czyli w połowie między próbką $x(0)$ i $x(1)$.

1.3. Dostępny sprzęt i oprogramowanie

1.3.1. Skrypty Matlaba

Do wykonania ćwiczenia przydatne będą opisane poniżej skrypty, które powstały specjalnie na potrzeby tego laboratorium.

- Do pobierania danych z karty dźwiękowej wykorzystuje się funkcję `LCPS_getdata(N, K, Ts)`. Funkcję tę stworzono, aby uniezależnić się od zmieniających się konwencji czytania danych w różnych wersjach Matlaba.

Jej parametry mają następujące znaczenie:

- `N` – liczba próbek do pobrania (długość bloku danych),
- `K` – liczba powtórzeń (liczba pobranych bloków) – zazwyczaj 1,
- `Ts` – okres próbkowania (w sekundach).

Należy pamiętać, że karta dźwiękowa implementuje tylko pewne stałe częstotliwości próbkowania i podane przez użytkownika wartości będą zaokrąglone. Informacje na ten temat można uzyskać, wykonując polecenie

```
help LCPS_getdata.
```

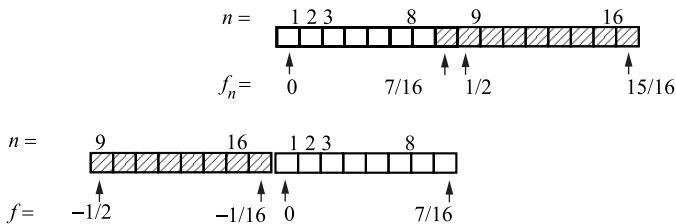
Dodatkowe, opcjonalne parametry funkcji `LCPS_getdata()` umożliwiają zmianę kanału, z którego pobierane są próbki – ale domyślnie jest to kanał karty dźwiękowej, dołączony do gniazda BNC „WE 1” w kasecie (patrz rys. 1.1).

- Wysyłanie próbek do karty w celu odtworzenia obsługuje funkcja `LCPS_putdata(N, Ts)` – przy domyślnych wartościach parametrów opcjonalnych sygnał pojawia się na gnieździe BNC „WY 1” w kasecie.
- Funkcja `Sa()` zaimplementowana jest w zestawie standardowych funkcji Matlaba pod nazwą `sinc()`.
- Standardowa funkcja `fftshift()` zamienia miejscami połówki wektora (rys. 1.2). Efekt zastosowania jej do obliczonego widma sygnału dyskretnego odpowiada zamianie widma z zakresu $(0, 2\pi)$ do zakresu $(-\pi, \pi)$. Taka możliwość wynika



Rysunek 1.1. Panel gniazd BNC dołączonych do karty dźwiękowej

z okresowości widma sygnału dyskretnego. Uwaga: jeśli argument jest macierzą, zamienione zostaną ćwiartki (lewo-prawo i góra-dół).



Rysunek 1.2. Działanie funkcji `fftshift()` na przykładzie wektora 16-elementowego

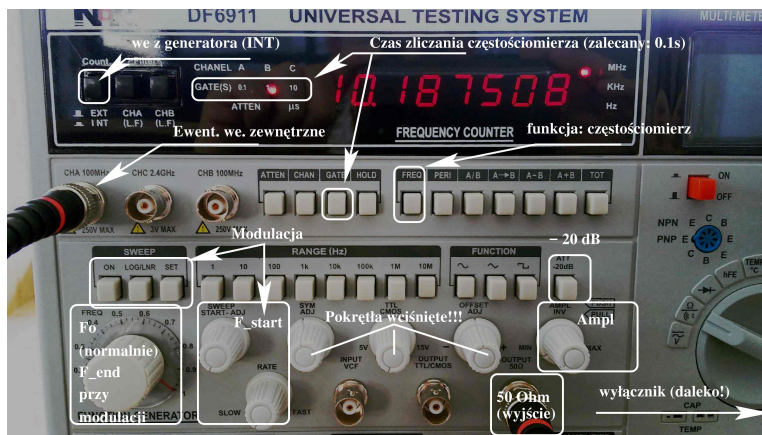
- Piękny wykres FFT w zakresie od $-\pi$ do π można uzyskać samemu, wykorzystując `fftshift()`, ale w razie pośpiechu zaleca się użyć funkcji `LCPS_dtf(x, K)`. Funkcja oblicza i wyświetla K -punktową DTF z sygnału x . **Wskazówka:** Niestety, K jest parametrem obowiązkowym. Jeśli nie wiesz, jaką wartość tu podać, podaj `length(x)`.

1.3.2. Aparatura

W ćwiczeniu student będzie posługiwać się generatorem sygnałowym jako źródłem sygnałów rzeczywistych (rys. 1.3). Najważniejsze uwagi co do obsługi generatora:

- wyłącznik jest wspólny dla zasilacza, generatora i częstotściomierza. Multimetr ma własny wyłącznik (bo jest zasilany z baterijki 9 V);
- aby mierzyć częstotliwość generatora, trzeba ustawić wejście („count”) na „int” (wcisnąć), oraz funkcję częstotściomierza na „freq”;
- nad pokrętką amplitudy jest przycisk „Att 20 dB”. Jeśli jest wyciśnięty, można uzyskać napięcia od ok. 2 do 20 V_{pp}. Po wciśnięciu jest 10× mniej;

- warto sobie ustawić czas zliczania (otwarcia bramki – „gate”) na 0,1 s (wielokrotnym wciskaniem przycisku „gate”) – pomiar będzie mniej dokładny, ale częściej odświeżany.



Rysunek 1.3. Panel czołowy generatora/częstościomierza NDN

Sygnaly będą próbkowane za pomocą dwóch różnych układów.

a) **Karty dźwiękowej komputera**, która podłączona jest przez gniazda BNC w kasecie laboratoryjnej (rys. 1.1) – karta zawiera przetworniki A/C typu sigma-delta z bardzo silnym filtrem antyaliasingowym i pozwala na łatwe wczytanie próbek do Matlaba.

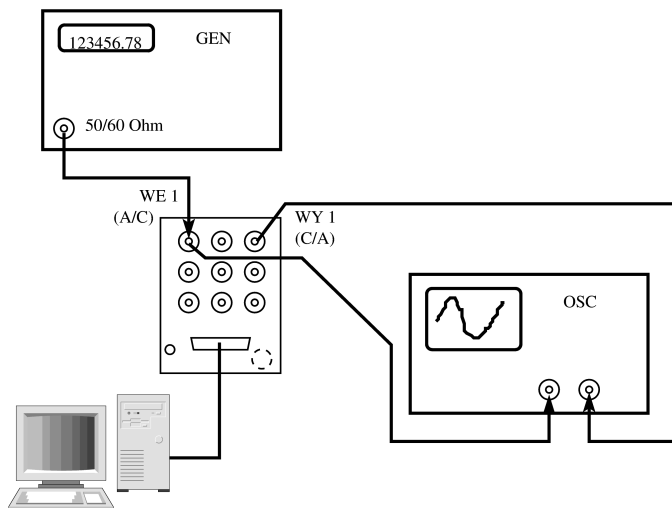
b) **Interfejsu akwizycji danych pomiarowych** firmy NI – przetworniki interfejsu nie zawierają filtrów antyaliasingowych; interfejs jest w łatwy sposób obsługiwany poprzez pakiet LabView. Na rys. 1.4 pokazano typowe podłączenie – żółty kabel podłączony jest do wyjścia analogowego AO0, a czerwony – do wejścia analogowego AI2.



Rysunek 1.4. Typowe podłączenie interfejsu NI przez modul BNC-2120

We wszystkich zadaniach zaleca się używać oscyloskopu do podglądu sygnału analogowego przed spróbkowaniem. Przykładowy sposób podłączenia aparatury z użyciem karty dźwiękowej (nadający się do większości zadań) pokazano poniżej

(rys. 1.5). Do podglądu sygnału wejściowego wykorzystuje się kanał 1 oscyloskopu, ewentualny sygnał odtworzony jest zobrazowany kanałem 2 oscyloskopu.



Rysunek 1.5. Przykładowe połączenie aparatury z przetwornikiem karty dźwiękowej

Uwaga: należy pamiętać, że **w tym ćwiczeniu** przetwarzanie odbywa się off-line (tj. po zarejestrowaniu sygnału), a więc jednocześnie porównywanie sygnału w kanałach 1 i 2 oscyloskopu nie ma sensu – nawet jeśli w czasie odtwarzania sygnału podglądamy sygnał wejściowy i nie zmieniliśmy jego parametrów, to żadne zależności fazowe pomiędzy oglądanymi sygnałami nie istnieją.

1.4. Eksperymenty do wykonania w laboratorium

Dla zróżnicowania zadań wykonywanych przez studentów, w opisie zadań będziemy się posługiwali parametrem Ns , który jest numerem stanowiska laboratoryjnego.

Wykresy sygnałów dyskretnych zaleca się rysować na ekranie za pomocą linii z zaznaczonymi położeniami próbek (podając trzeci parametr – styl linii – do funkcji rysującej: `plot(n, x, '-*')`). Można wtedy łatwo stwierdzić *gdzie* naprawdę znajdują się próbki. Poprawnie byłoby nie łączyć punktów liniami, ale linia ułatwia zauważenie kolejności próbek¹.

Odpowiedz na pytanie: Dlaczego łączenie punktów liniami nie jest poprawne formalnie? **Wskazówka:** Dysponując tylko ciągiem próbek sygnału – co wiemy o jego wartościach pomiędzy chwilami próbkowania?

Odpowiedz



¹ Poprawne jest także użycie funkcji `stem()`. Ze względu jednak na wątpliwe efekty wizualne zalecamy to tylko w uzasadnionych przypadkach.

1.4.1. Sygnał dyskretny i jego widmo

1.4.1.1. Symulowany i rzeczywisty sygnał dyskretny







-  Zanotuj  Ustaw generator tak, aby otrzymać sygnał sinusoidalny o częstotliwości f_a wybranej zgodnie z numerem stanowiska z tabeli 1.1 i amplitudzie około 5 V. Zmierz (orientacyjnie) parametry sygnału: amplitudę, częstotliwość, okres – za pomocą oscyloskopu². **Zanotuj wartości amplitudy, częstotliwości i okresu: założoną oraz zmierzoną.**


Tabela 1.1

Nr stan.	1	2	3	4	5	6	7	8	9	10	11	12	13
f_a [Hz]	2400	4800	1800	1200	600	6000	480	720	7200	1440	9600	2880	1920

-  Zanotuj  Oblicz, jakie powinny być unormowane wartości parametrów sygnału (częstotliwość f_n , okres T_n) po jego spróbkowaniu z częstotliwością próbkowania $f_s = 48$ kHz. **Zanotuj obliczone wartości.** Wartości te będą używane w następnych zadaniach.
-   Zasymuluj za pomocą Matlab'a 200 próbek sygnału o obliczonych parametrach. Narysuj na ekranie wykres sygnału. **Wskazówka:** Możesz otworzyć na ekranie tę instrukcję i kopiować fragmenty kodu z PDF'a do okna komend Matlab'a.

```
N=200;
n=0:(N-1);
xs=% tu wstaw obliczenie próbek sygnału
figure(1)
plot(n,xs,'*-')
title('Symulowany sygnał');
xlabel('Indeks czasowy probki')
ylabel('Wartość próbki')
```

W przykładowym kodzie wektor próbek sygnału jest zapisany w zmiennej xs . Nie usuwaj i nie nadpisuj tej zmiennej – w dalszym ciągu będziemy porównywali ten (idealny) sygnał z rzeczywistym.

-  Wykonaj połączenia według rys. 1.5 i zarejestruj rzeczywisty sygnał z generatora (liczba próbek jak w poprzednim punkcie, częstotliwość próbkowania jak

² Pisząc „amplituda” mamy na myśli połowę napięcia międzyszczytowego – jest to konwencja typowo przyjmowana dla sygnałów bez składowej stałej (np. dźwiękowych). Zwróć uwagę, że oscyloskop domyślnie zmierzy napięcie międzyszczytowe (*peak-to-peak*), a funkcja oscyloskopu *Amplitude* też nie odpowiada przyjętej przez nas konwencji. Uwaga: meteorolodzy pod pojęciem *amplituda temperatur* rozumieją całą wartość międzyszczytową (a my tylko pół).

założona do obliczeń, ustawiona z dokładnością ok. dwóch cyfr znaczących³). Narysuj wykres na ekranie. Zaobserwuj sygnał (wejściowy) na oscyloskopie.

```
Ts=% tu oblicz okres próbkowania
xg=LCPS_getdata(N,1,Ts);
figure(2);
plot(n,xg,'-*')
...
```

Uwaga: Wynik `LCPS_getdata()` jest unormowany do zakresu napięć przetwornika (ok. ± 10 V) – tj. amplitudzie 5 V odpowiada wartość około 0,5.

Zauważ i **zanotuj** różnice między sygnałem symulowanym i rzeczywistym. **Naszkić 1–2 okresy sygnału rzeczywistego wybrane tak, aby pokazać najwyraźniejsze nieidealności.**

†† Na podstawie wykresu oszacuj i **zanotuj** liczbę próbek w okresie sygnału symulowanego i rzeczywistego. **Odpowiedz na pytanie:** Czy można dokładnie określić okres sygnału rzeczywistego?

†† **Zadanie extra** Zmień kod rysujący wykres dla sygnału rzeczywistego, aby oś czasu była wyskalowana w jednostkach czasu fizycznego.

```
figure(3);
t=% tu wstaw obliczenie wektora czasu z numerów próbek n
...
plot(t, xg, '*-')
```

Zanotuj fragment kodu obliczania wektora czasu.

1.4.1.2. Okresowość widma sygnału dyskretnego

†† Narysuj na ekranie widma (amplitudowe) obu sygnałów (symulowanego i rzeczywistego)

```
% w poniższym kodzie użyj odpowiednio xs lub xg
x=xs;
X=fft(x);
figure(4);
Nx=length(x);
k=0:(Nx-1); %indeksy próbek widma
plot(k,abs(X);
% Matlab domyślnie indeksuje od 1
% dlatego potrzebne jest tu k, aby oś pozioma
% była indeksowana od 0

%narysujmy to samo z zerem pośrodku
```

³ Nie staraj się uzyskać zbyt idealnych ustawień – w praktyce sygnały nie są idealne, częstotliwości generatorów zmieniają się z temperaturą i czasem, na odebrane sygnały może mieć wpływ efekt Dopplera.

Zanotuj



Naszkić



Zanotuj



Odpowiedz



Zanotuj



```
figure(5)
k1=floor(-Nx/2):floor(Nx/2)-1;
plot(k1,abs(fftshift(X)));
```

Zanotuj



Zweryfikuj ze swą wiedzą położenie maksimum widma. **Zanotuj** indeks maksimum i odpowiadającą mu częstotliwość unormowaną oraz rzeczywistą.

Odpowiedz



Odpowiedz na pytania:

1) Dlaczego widmo ma dwa wierzchołki? Jakim częstotliwościom one odpowiadają?

2) Dlaczego w przykładowym kodzie użyto funkcji `abs()`?

3) Dlaczego w drugim przykładzie użyto funkcji `fftshift()`?

4) Jakiej częstotliwości odpowiada k -ty element wektora widma X (np. podaj wzór jak przeliczać k na θ lub f_n)?

Zanotuj



Uzpełnij tabelę 1.2 w sprawozdaniu, pokazując odpowiedniość indeksu częstotliwościowego k , częstości kołowej unormowanej θ i częstotliwości fizycznej f .

Tabela 1.2. Tabela indeksów do wypełnienia w sprawozdaniu

Indeks			-1	0	+1	+2		
θ	$-\pi$	$-\pi/2$						π
f [Hz]							12000	



Zadanie extra Zasygnalizuj sygnały o częstotliwościach: $2f_n$; $0,5 - f_n$; $0,5 + f_n$; $1,0 - f_n$; $1,0 + f_n$; $3,0 \pm f_n$. Narysuj na ekranie wykresy czasowe i widma tych sygnałów. **Naszkiecuj kilka wybranych. Odpowiedz na pytanie:** dlaczego w niektórych przypadkach dla różnych parametrów uzyskano takie same wykresy?

Naszkiecuj



Odpowiedz



Użyj analizatora widma sygnału zbudowanego na bazie LabView. Analizator taki dostępny jest w postaci instrumentu wirtualnego „Spectrum Analyzer”, uruchamianego z folderu LCYPSVIs (lub przez skrót na pulpicie).

Ustaw parametry:

wybór wejścia

– Dev1/ai2; jest to numer kanału wejściowego (zakładamy, że sygnał jest podłączony do wejścia AI2);

częstotliwość próbkowania – wybierz wartość (10+numer_stanowiska) kHz

liczba próbek

– 200

liczba zer

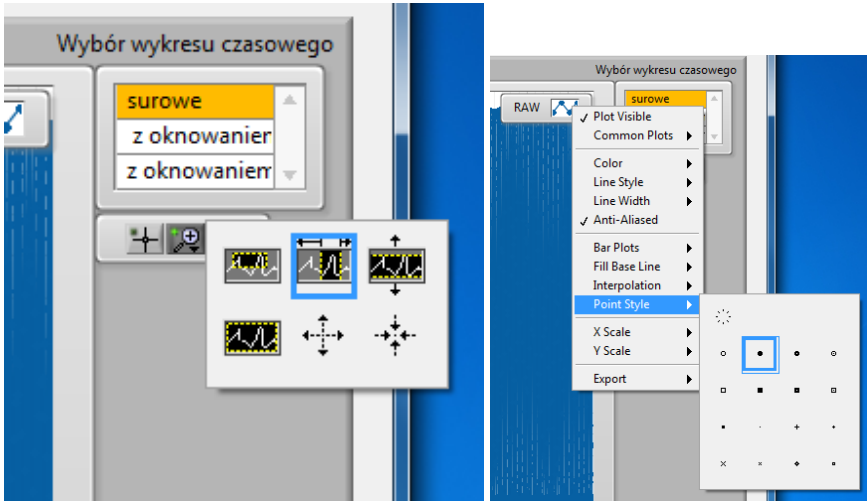
– 0 (zero)

Wciśnij przełącznik FFTShift, aby oś pozioma na wykresie wyskalowana była w wartościach częstotliwości unormowanej od $-0,5$ do $+0,5$. Użyj skali liniowej lub logarytmicznej (po przełączeniu warto nacisnąć „AutoFit”). **Wskazówka:** Bardzo przydatne są funkcje „Pauza” i możliwość rozciągnięcia wykresu w poziomie (kliknij „lupe”) oraz rysowania z zaznaczeniem punktów danych (kliknij okienko z tytułem wykresu „RAW”) – patrz rys. 1.6.

Zanotuj



Zanotuj jaka była częstotliwość próbkowania i oblicz częstotliwość Nyquista.



Rysunek 1.6. Włączanie rozciągania wykresu w poziomie oraz rysowania z zaznaczaniem punktów danych

Ustaw częstotliwość sygnału w zakresie Nyquista. Oprócz przebiegu i widma sygnału dyskretnego (na ekranie komputera), oglądaj także przebieg czasowy oraz widmo sygnału analogowego na oscyloskopie. W tym celu włącz funkcję analizatora widma na oscyloskopie; po wciśnięciu przycisku „Maths” wybierz (wciskając przyciski pod ekranem i obracając pokrętkę wyboru):

- operator: FFT,
- source: 1,
- span: 200 kHz,
- center: 0 kHz.

Stopniowo zwiększaj częstotliwość poza granicę Nyquista, aż do częstotliwości większych niż podwojona częstotliwość próbkowania.

Odpowiedz na pytanie: Co dzieje się z widmem sygnału spróbkowanego, gdy częstotliwość sygnału przekracza częstotliwość Nyquista, a co, gdy przekracza częstotliwość próbkowania? Jak wygląda sygnał w tych sytuacjach (zwróć uwagę na okres sygnału po spróbkowaniu!).

Odpowiedz



Zadanie extra Korzystając z arytmetyki zespolonej Matlaba, zasymuluj sygnał postaci $e^{jn2\pi f_n}$ przy f_n tej co zwykle. Narysuj wykres czasowy i widmo.

Odpowiedz na pytanie: Ile widzisz wierzchołków widma i dlaczego?

Odpowiedz



1.4.2. Efekty kwantowania

Zadanie extra Zasymuluj 50 próbek sygnału sinusoidalnego o częstotliwości unormowanej $3/50$.

- ✚ Skwantuj sygnał do Nb bitów. Przyjmij wartość Nb równą 4 na stanowiskach o numerze parzystym albo 5 na nieparzystym.

```
Nb = %tu ustaw liczbę bitów
n = %tu utworzenie indeksu czasowego
x = %tu generacja Twojego sygnału
figure(1)
plot(n, x, '-*');
nq = 2^Nb; %liczba poziomów kwantowania
q = 2/nq %wartość kwantu
xq = q * round(x/q);
figure(2)
plot(n, x, '-*', n, xq, '-')
```

Odpowiedz



Odpowiedz na pytanie: dlaczego w przykładowym kodzie wartość kwantu jest dwa razy większa niż odwrotność liczby poziomów? Jaki zakres wartości sygnału przyjęto przy konstrukcji tego kodu? **Wskazówka:** Naskicuj sobie poziomy kwantowania, np. przy Nb=3.

- ✚ Obejrzyj widmo sygnału na ekranie, porównaj z widmem sygnału bez kwantowania, **zannotuj zauważone różnice**.

Zannotuj

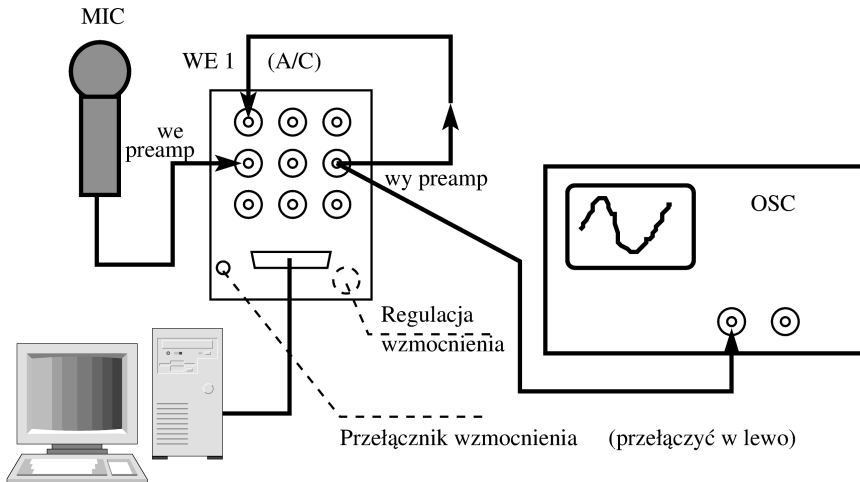


- ✚ Powtórz eksperymenty zmieniając liczbę bitów, **naskicuj** wybrane widmo, na którym widoczne są efekty kwantowania.

Naskicuj



- ✚ Zarejestruj kilka sekund sygnału mowy (połączenie mikrofonu poprzez przedwzmacniacz – patrz rysunek rys. 1.7).



Rysunek 1.7. Połączenie mikrofonu poprzez przedwzmacniacz

- ✚ **Zadanie extra** Jeśli masz czas, to możesz użyć także innego sygnału dźwiękowego – na przykład:

– odtworzyć muzykę z telefonu komórkowego (do mikrofonu – albo znaleźć w szufladach kabel, który pozwoli od razu przejść na wtyk BNC),

- ściągnąć muzykę z sieci (jeśli wiesz, jak z niej uzyskać plik .wav),
 - uruchomić odbiór radia FM za pomocą odbiornika SDR i zarejestrować plik na dysku (o sprzęt zapytaj prowadzącego, a program odbiornika nazywa się sdrsharp).
- ✚ Dla kontroli odtwórz sygnał przez głośnik:
- połącz wyjście WY1 w kasecie z wejściem głośnikowym „input” części „audio amplifier” w przyrządzie uniwersalnym (rys. 1.8);



Rysunek 1.8. Połączenie mikrofonu poprzez przedwzmacniacz

- wciśnij przełącznik „int/ext” aby włączyć wewnętrzny głośnik;
 - odtwórz próbki przez kartę dźwiękową (dobierz mnożnik, aby sygnał było dobrze słychać):
`LCPS_putdata(xg*10, 1/48000);`
- ✚ Skwantuj sygnał (uwaga – przed kwantowaniem sprawdź, jaka jest amplituda sygnału i dostosuj do niej sposób kwantowania) i odtwórz go przez głośnik po skwantowaniu.
- ✚ **Zadanie extra** zrealizuj to w LabView aby działało w czasie rzeczywistym.
- ✚ Eksperymentując z liczbą bitów kwantowania, spróbuj określić i *zanotować* minimalną liczbę bitów potrzebną:
- aby mowa była zrozumiała,
 - aby mowa brzmiała przyjemnie.

Zanotuj
✍

Pamiętaj, że istotne jest *ile bitów jest rzeczywiście wykorzystywanych* – w naszym eksperymencie prawdopodobnie sygnał nie wypełnia całej skali amplitudy. Zastanów się, jak obiektywnie obliczyć liczbę używanych bitów.

1.4.3. Redukcja i ekspansja próbek

1.4.3.1. Redukcja próbek

Przed dalszymi eksperymentami warto pozbyć się z ekranu nadmiaru wykresów.
`close all`

Nie czystymy natomiast środowiska – będziemy wykorzystywali sygnały zasymulowane i zarejestrowane w pierwszej części ćwiczenia.

Do poniższych eksperymentów możesz użyć sygnału symulowanego (powinien wciąż być w zmiennej *xs*), albo zarejestrowanego (w zmiennej *xg*). Jeśli sygnały się skasowały, wytwórz je ponownie według części ✚. Aby nie szukać daleko, **przepisz**

Zanotuj
✍

z pierwszej części protokołu częstotliwości – sygnału i próbkowania.



⚡ Oblicz i wyświetl widmo sygnału (możesz użyć `LCPS_dtf(xs, length(xs))`).

Zanotuj położenie (indeks lub częstotliwość unormowaną) maksimum widma.

⚡ Usuń z sygnału symulowanego co drugą próbkę.

```
xds=xs(1:2:end);
```



Oblicz i zanotuj jakiej częstotliwości próbkowania odpowiada teraz ciąg próbek.

Narysuj na ekranie widmo i wykres czasowy (na oddzielnych rysunkach).

Odpowiedz



Zaobserwuj indeks maksimum widma. **Odpowiedz na pytanie:** Jak redukcja wpłynęła na widmo?

⚡ Zwiększ stopień redukcji tak, aby nie były spełnione założenia twierdzenia o próbkowaniu. **Zanotuj jaka teraz jest efektywnie częstotliwość próbkowania, naszkicuj widmo.**



Naszkicuj



Odpowiedz na pytanie: Jakie niebezpieczeństwa czyhają przy redukcji próbek?

Odpowiedz



1.4.3.2. Ekspansja próbek i interpolacja

Ponownie zamknij okna wykresów. Sprawdź, czy w zmiennej `xs` nadal masz sygnał o parametrach wybranych na początku poprzedniego punktu.

⚡ Zwiększ dwukrotnie liczbę próbek w sygnale (dokonaj ekspansji) metodą dodania zer pomiędzy istniejącymi próbkami.

```
%sztuczka na dodanie zerowych próbek pomiędzy istniejącymi
N=length(xs);
U=2; % współczynnik ekspansji
%przygotuj wyzerowany wektor na wynik
xu=zeros(1,U*N); % "u" jak "upsampled"
xu(1:U:end)=xs; % w xu nieprzypisane elementy pozostaną zerowe
plot(xu, '-x');
figure; %narysuj widmo sam, ale w nowym oknie
```

Naszkicuj



Odpowiedz



Narysuj sygnał i widmo; **naszkicuj widmo** w protokole. **Odpowiedz na pytania:**

- Jakiej częstotliwości próbkowania odpowiada teraz ciąg próbek?
- Skąd wziął się prążek na częstotliwości powyżej 0.25? (uwaga, matematyka się kłania)

⚡ Dokonaj ekspansji z powieleniem próbek sąsiednich (sygnał schodkowy).

```
%sztuczka na powielenie próbek
xs=xs(:).'; %teraz xs na pewno będzie wektorem poziomym
U=2; % współczynnik ekspansji
xp=ones(U,1)*xs; % macierz U wierszy na length(xs) kolumn
xp=xp(:).'; % tworzymy wektor łącząc kolumny i transponujemy go
figure;
...
```

Narysuj sygnał, narysuj widmo i **naszkić** je w protokole. **Odpowiedz na pytanie:** Dlaczego teraz prążek na wysokich częstotliwościach ma mniejszą amplitudę?

Naszkić



Odpowiedz



- ✚ Zwiększ 4-krotnie liczbę próbek (ekspansja z dodaniem zer) i następnie oblicz wartości dodanych próbek przy założeniu dolnopasmowości oryginalnego sygnału (splot $\text{conv}(x, h)$ z jądrem sinc – utnij jądro np. do zakresu $[-16:16]$)⁴:

```
xu=%tu powtórz odpowiednio sztuczkę na dodanie zerowych próbek
sinc16=sinc([-16:16]/4);
plot([-16:16],sinc16);
xu_interp16=conv(xu,sinc16);
```

Wyświetl na ekranie sygnał z wyinterpolowanymi próbkami oraz jego widmo. **Zanotuj** wartości reszkowych prążków widma na wybranej (np. drugiej⁵) harmonicznej sygnału podstawowego. **Odpowiedz na pytania:**

Zanotuj



Odpowiedz



- Dlaczego tak właśnie dobrano skalowanie funkcji sinc w czasie? (pomóż sobie rysunkiem sinc – zauważ gdzie są przejścia przez zero)
- Dlaczego na początku i na końcu sygnał wygląda *nieszczególnie*?

Spróbuj też obliczyć widmo z pominięciem początkowych i końcowych próbek.

- ✚ **Zadanie extra** Powtórz interpolację, tym razem używając jądra sinc uciętego szerzej – np. do zakresu $[-32:32]$. **Porównaj** wartości reszkowych prążków z zanotowanymi z poprzedniego punktu.

Zanotuj



- ✚ **Zadanie extra** Wyinterpoluj próbki inaczej – poprzez splot z jądrem sinc zmodulowanym kosinusoidą o częstotliwości unormowanej $5/16$.

```
sinc32mod516=sinc([-32:32]/8).*cos(2*pi*5/16*[-32:32]);
plot([-16:16],sinc32mod516);
xu_interp32mod516=conv(xu,sinc32mod516);
```

Odpowiedz na pytania:

Odpowiedz



Ponieważ jest to zadanie *extra*, sam(a) spróbuj opisać co się stało i dlaczego tak. A jako *super extra* – uzasadnij, dlaczego takie parametry liczbowe zaproponowaliśmy.

Pomoże Ci w tym przyjrzenie się widmom wszystkich użytych sygnałów.

1.4.3.3. Ekspansja próbek i interpolacja – sygnał rzeczywisty

- ✚ Zarejestruj sygnał sinusoidalny o dowolnie wybranej częstotliwości $f_a < 4$ kHz przy $f_s = 48$ kHz i o czasie trwania ok. 4 sekund. **Zanotuj** wybrane wartości f_a i f_s . Wykonaj redukcję tak, aby efektywna częstotliwość próbkowania po redukcji wciąż spełniała twierdzenie o próbkowaniu z niewielkim marginesem (zachowaj sygnał zredukowany i oryginalny). Odsłuchaj sygnał oryginalny.


Zanotuj



⁴ Zauważ, że splot z $h[n]$ i filtracja filtrem SOI o odpowiedzi $h[n]$ jest to matematycznie ta sama operacja!

⁵ Pamiętajmy, że „pierwsza harmoniczna” to po prostu podstawowy sygnał.

LCPS_putdata(xg*10,1/48000);


 Dokonaj ekspansji z powrotem do oryginalnej liczby próbek, wypełniając dodane próbki zerami. Obejrzyj sygnał na oscyloskopie – aby uzyskać stabilny obraz wybierz źródło synchronizacji (trigger source) z kanału 2. Odsłuchaj sygnał, **zanotuj** wrażenia.


Zanotuj



Zanotuj



 Dokonaj ekspansji z powrotem do oryginalnej liczby próbek, z wypełnieniem sąsiednimi próbkami. Obejrzyj sygnał na oscyloskopie, odsłuchaj sygnał, **zanotuj** wrażenia.

 **Zadanie extra** Wyinterpoluj próbki poprzez splot z sinc (skorzystaj z kodów z zadania poprzedniego). Obejrzyj sygnał na oscyloskopie, odsłuchaj sygnał, **zanotuj** wrażenia.

Zanotuj



1.4.3.4. Zadanie projektowe: minimalna częstotliwość próbkowania przy podpróbkowaniu

1) Wygeneruj analogowo sygnał pasmowy włączając przemiatanie (liniową modulację częstotliwości) w generatorze. Ustaw szerokość pasma B i częstotliwość środkową f_c według tabeli 1.3.

Tabela 1.3

Nr stanowiska	1	2	3	4	5	6	7	8	9	10	11	12	13
f_c [kHz]	2,6	3,2	4,3	2,5	5,2	8,0	4,5	4,9	5,4	4,3	3,6	5,3	13,0
B [Hz]	300	400	200	300	400	500	200	1000	300	400	700	400	1300

Wskazówka: Najpierw musisz przeliczyć B i f_c na częstotliwość początkową i końcową, a potem dokonać kilku manipulacji w przyrządzie NDN:

- „sweep ON” załącza modulację;
- „log/lnr” powinien być wciśnięty aby modulacja była liniowa;
- wciśnięcie „set” pozwala zmierzyć częstotliwość początkową (ustawianą pokrętkiem „sweep start-adj”),
- częstotliwość końcowa przemiatania jest równa częstotliwości ustalonej przy wyłączeniu „sweep”;
- „rate” warto ustawić na „fast”.

Zanotuj



Użyj funkcji analizy widma w oscyloskopie do kontroli ustawień. **Zanotuj** ustawione wartości.

2) Zarejestruj kilka sekund sygnału przy częstotliwości próbkowania 48 kHz. Zarejestrowany sygnał zapamiętaj w dobrze nazwanej zmiennej (będzie potrzebna do późniejszych porównań).

3) Zaprojektuj schemat podpróbkowania, żeby uzyskać jak najmniej próbek na sekundę, ale wciąż odtworzyć sygnał inteligentną interpolacją do znanego pasma (przyjmij, że na etapie odtwarzania jest znana częstotliwość środkowa i stopień decymacji). Szukaj częstotliwości próbkowania wśród częstotliwości, które można uzyskać przez redukcję próbek z częstotliwości 48 kHz.

4) Zrealizuj takie podpróbki (przez redukcję próbek z nagranych sygnałów), a następnie zrekonstruuuj sygnał z powrotem do $f_s = 48$ kHz.

5) Porównaj wynik z oryginałem na oscyloskopie (w dziedzinie widma i w dziedzinie czasu) i w Matlabie oraz w głośniku.

Zadanie można zrealizować również za pomocą LabView – wtedy mamy możliwość dowolnego wyboru częstotliwości próbkowania i nie musimy „udawać” tego za pomocą redukcji próbek.

Realizacja za pomocą LabView niestety wymaga obsłużenia przetworników. Najłatwiej to będzie zrobić z pomocą „express VI”, znanych z ćwiczenia zerowego.

W protokole opisz sposób wyboru częstotliwości i ich wartości liczbowe oraz podsumuj wyniki porównania.