



Cyfrowe przetwarzanie sygnałów

Dr inż. Janusz Baran



1. Wstęp

LITERATURA

- Zieliński T.: *„Cyfrowe przetwarzanie sygnałów. Od teorii do zastosowań”*, WKiŁ, 2005 (*„Od teorii do cyfrowego przetwarzania sygnałów”*, Wydział EAIiE AGH, 2002).
- Smith S.W.: *„Cyfrowe przetwarzanie sygnałów. Praktyczny poradnik dla inżynierów i naukowców”*, Wyd. BTC, 2007.
- Lyons R.G.: *„Wprowadzenie do cyfrowego przetwarzania sygnałów”*, WKiŁ, 1999.
- Stranneby D.: *„Cyfrowe przetwarzanie sygnałów. Metody, algorytmy, zastosowania”*, Wyd. BTC, 2004.
- Praca zbiorowa pod red. A. Dąbrowskiego: *„Przetwarzanie sygnałów przy użyciu procesorów sygnałowych”*, Wyd. Politechniki Poznańskiej, 1998.
- Oppenheim A.V., Schaffer R.W.: *„Cyfrowe przetwarzanie sygnałów”*, WKiŁ, 1979.
- Marven C., Ewers G.: *„Zarys cyfrowego przetwarzania sygnałów”*, WKiŁ, 1999.
- Praca zbior. pod red. A. Wojtkiewicza: *„Cyfrowe przetwarzanie sygnałów”*, skrypt lab., Oficyna Wyd. Pol. Warszawskiej, 2002.

LITERATURA – cd.

- Zbiór artykułów pod red. A.V. Oppenheima: „*Sygnały cyfrowe. Przetwarzanie i zastosowania*”, WNT, 1982.
- Rutkowski L.: „*Filtry adaptacyjne i adaptacyjne przetwarzanie sygnałów*”, WNT, 1994.
- Papoulis A.: „*Obwody i układy*”, WKiŁ, 1988.
- Baksakow W.: „*Układy i sygnały radiotechniczne*”, PWN, 1991.
- Izydorczyk J., Płonka G., Tyma G.: „*Teoria sygnałów. Wstęp (teoria, przykłady zadania)*”, Wyd. Helion, Gliwice, 1999.
- Szabatin J.: „*Podstawy teorii sygnałów*”, WKiŁ, 2003.
- Basztura Cz.: „*Źródła, sygnały i obrazy akustyczne*”, WKiŁ, 1988
- Drozdek A.: „*Wprowadzenie do kompresji danych*”, WNT, 1999.
- Sayood K.: „*Kompresja danych – wprowadzenie*”, Wyd.RM, 2002
- Praca zbior. pod red. Wł. Skarbka: „*Multimedia. Algorytmy i standardy kompresji*”, Akadem. Oficyna Wyd. PLJ, 1998.
- Malina W., Ablameyko S., Pawlak W.: „*Podstawy cyfrowego przetwarzania obrazów*”, Akadem. Oficyna Wyd. EXIT, 2002.
- Białasiewicz J.: „*Falki i aproksymacje*”, WNT, 2000.
- Wesołowski K.: „*Podstawy cyfrowych systemów telekomunikacyjnych*”, WKiŁ, 2003.
- Haykin S.: „*Systemy telekomunikacyjne*”, tom 1,2, WKiŁ, 1998.
- Strony internetowe Texas Instruments, Analog Devices, Motorola itp.

Cyfrowe przetwarzanie sygnałów

elektronika analogowa
(czujniki i akulatory)

elektronika cyfrowa
(mikroprocesory,
przetworniki a/c i c/a)

analiza numeryczna

teoria sygnałów

układy dynamiczne

prawdopodobieństwo i
statystyka

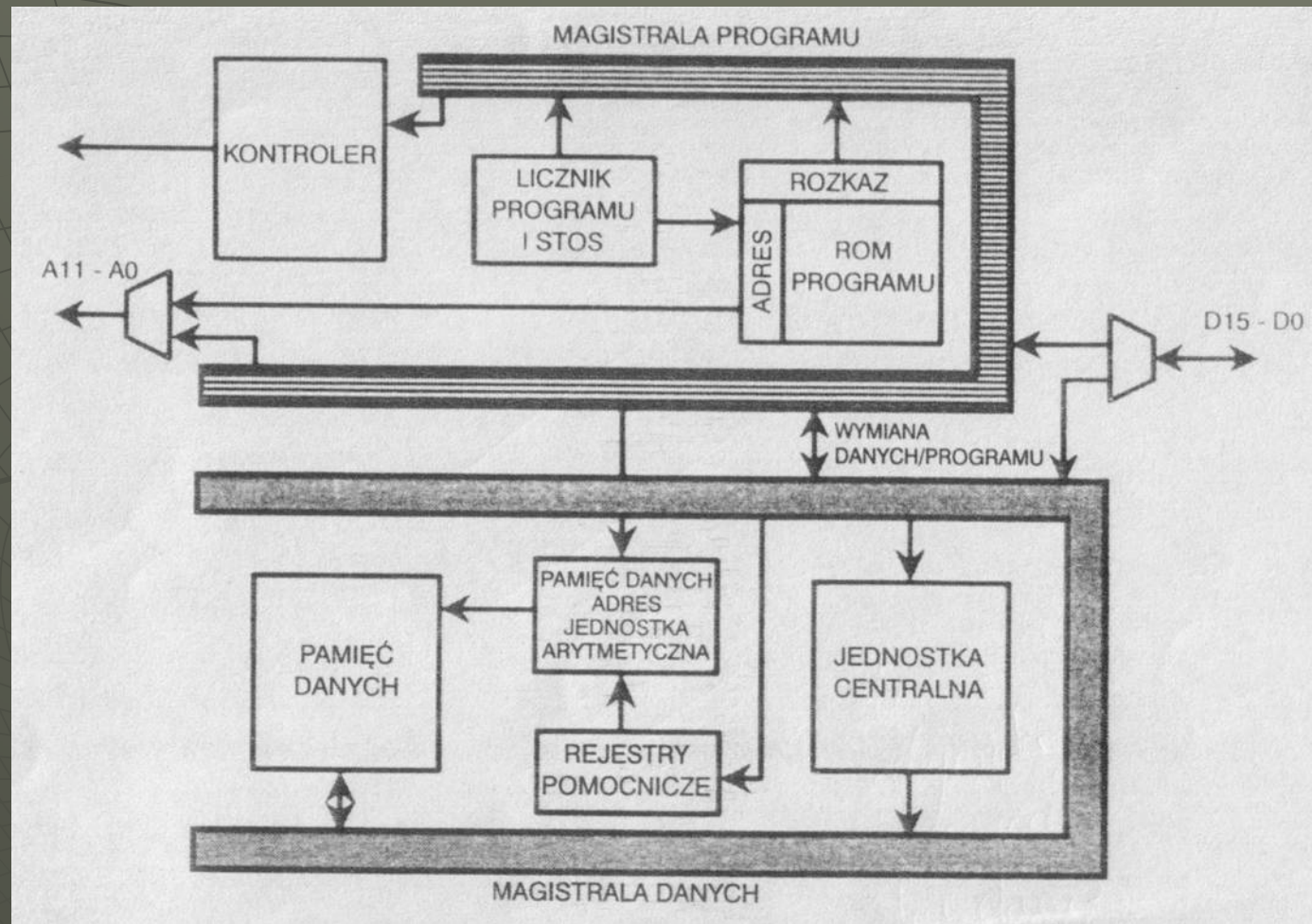
teoria informacji

optymalizacja

Cyfrowe przetwarzanie
sygnałów jako dziedzina
interdyscyplinarna

Wybrane daty i wydarzenia z historii DSP

- XIX wiek - sformułowanie przekształceń Fouriera i Laplace'a
- koniec lat 30 XX w. - opracowanie architektury z rozdzielonymi szynami danych i programu (Howard Aiken, Uniwersytet Harvard)
- 1945 - pierwszy komputer elektroniczny ENIAC (Uniwersytet Pensylwania), architektura Harvard
- 1947 - opracowanie rekurencyjnego algorytmu Levinsona rozwiązywania równań normalnych w analizie korelacyjnej (modyfikacja Durбина 1959)
- 1965 - opracowanie algorytmu FFT (Cooley, Tukey: „An algorithm for the machine computation of complex Fourier series”)
- 1971 - komputer realizujący mnożenie w jednym cyklu zegarowym (Lincoln FDP, 600ns, zbudowany z szybkich elementów logicznych w technologii ECL)
- lata 80 XX w. - scalone mikroprocesory DSP, m.in. NEC mPD7720, Texas Instruments TMS32010 (1982) (technologie NMOS, 3mm)
- przełom XXI w. - mikroprocesory DSP o mocy ponad 10 GFlops, przetwarzanie równoległe i potokowe, wszystkie rozkazy (w tym MAC) w jednym cyklu, wbudowana pamięć i układy we/wy

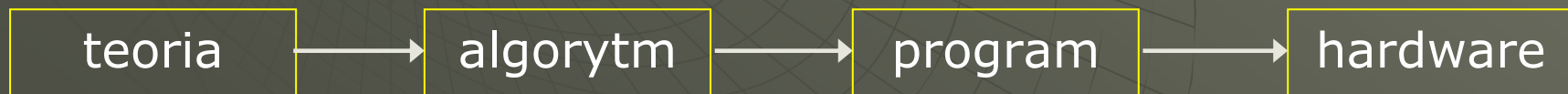


Rys. Uproszczony schemat blokowy architektury Harvard mikroprocesora TMS32010

Zalety technologii cyfrowego przetwarzania sygnałów

- **elastyczność**: umożliwia realizację szerokiego zakresu funkcji przy użyciu niewielu elementów i odpowiedniego oprogramowania
- **odporność na zakłócenia i straty informacji** przy przesyłaniu, przechowywaniu i kopiowaniu
- **możliwość kompresji i kodowania** informacji
- **powtarzalność**: nie występują problemy tolerancji lub zmiany parametrów typowe dla układów analogowych
- **możliwość pracy wielokanałowej** z podziałem czasu i zwiększenia precyzji przez proste zwiększenie długości słowa danych
- **dostępność tanich elementów** (układy VLSI, przetworniki a/c i c/a) o coraz większej szybkości działania i mocy obliczeniowej mogące sprostać większym wymaganiom

Projektowanie układów DSP



Typowe obszary zastosowań cyfrowego przetwarzania sygnałów

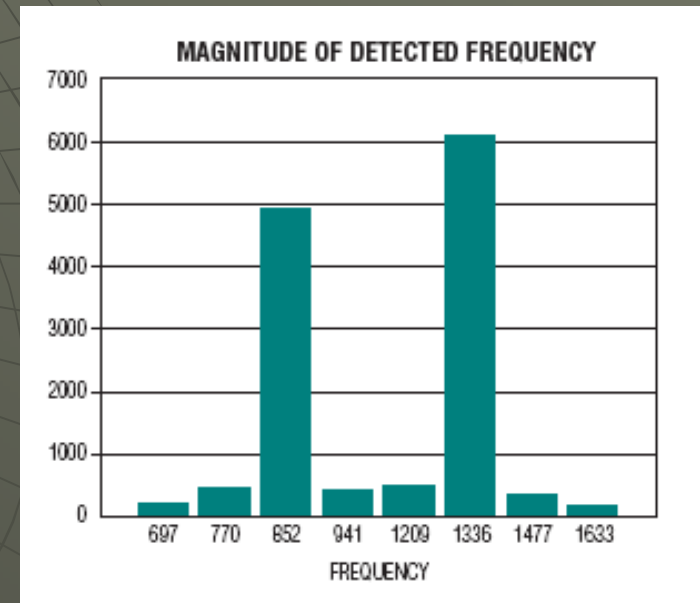
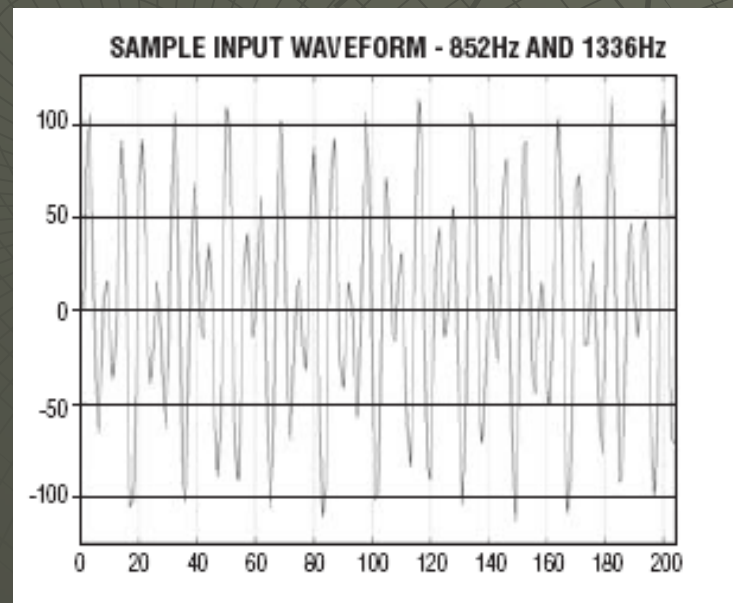
Telekomunikacja cyfrowa:

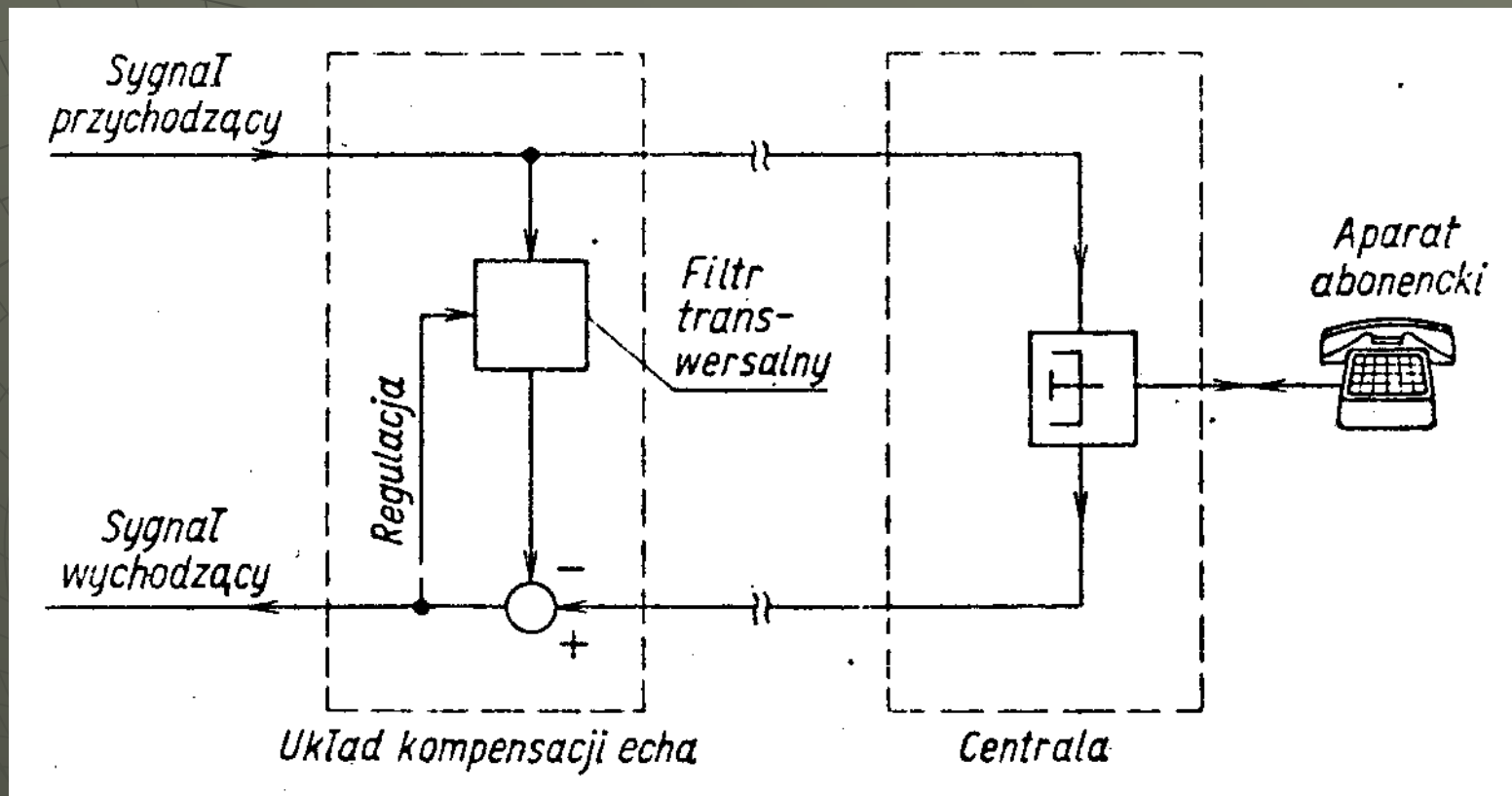
- ✓ transmisja cyfrowa (wielokanałowa multipleksowana np. z modulacją logPCM)
- ✓ kompresja danych (obrazu, dźwięku, plików)
- ✓ bezpieczna komunikacja (kodowanie informacji)
- ✓ modemy cyfrowe ADPCM, generowanie i detekcja sygnałów tonowych DTMF
- ✓ komutacja cyfrowa (małe matryce łączą sygnały rozdzielone w czasie, sterowanie funkcjami, 200 razy mniej przewodów)
- ✓ filtracja cyfrowa, np. tłumienie szumów, tłumienie echa

Column >	1209 Hz	1336 Hz	1477 Hz	1633 Hz
Row v				
697 Hz	1	2	3	a
770 Hz	4	5	6	b
852 Hz	7	8	9	c
941 Hz	*	0	#	d

Rys. Tabela częstotliwości DTMF (*Dual Tone MultiFrequency*)

Rys. Przebieg czasowy i amplituda widma DTMF dla cyfry „8”

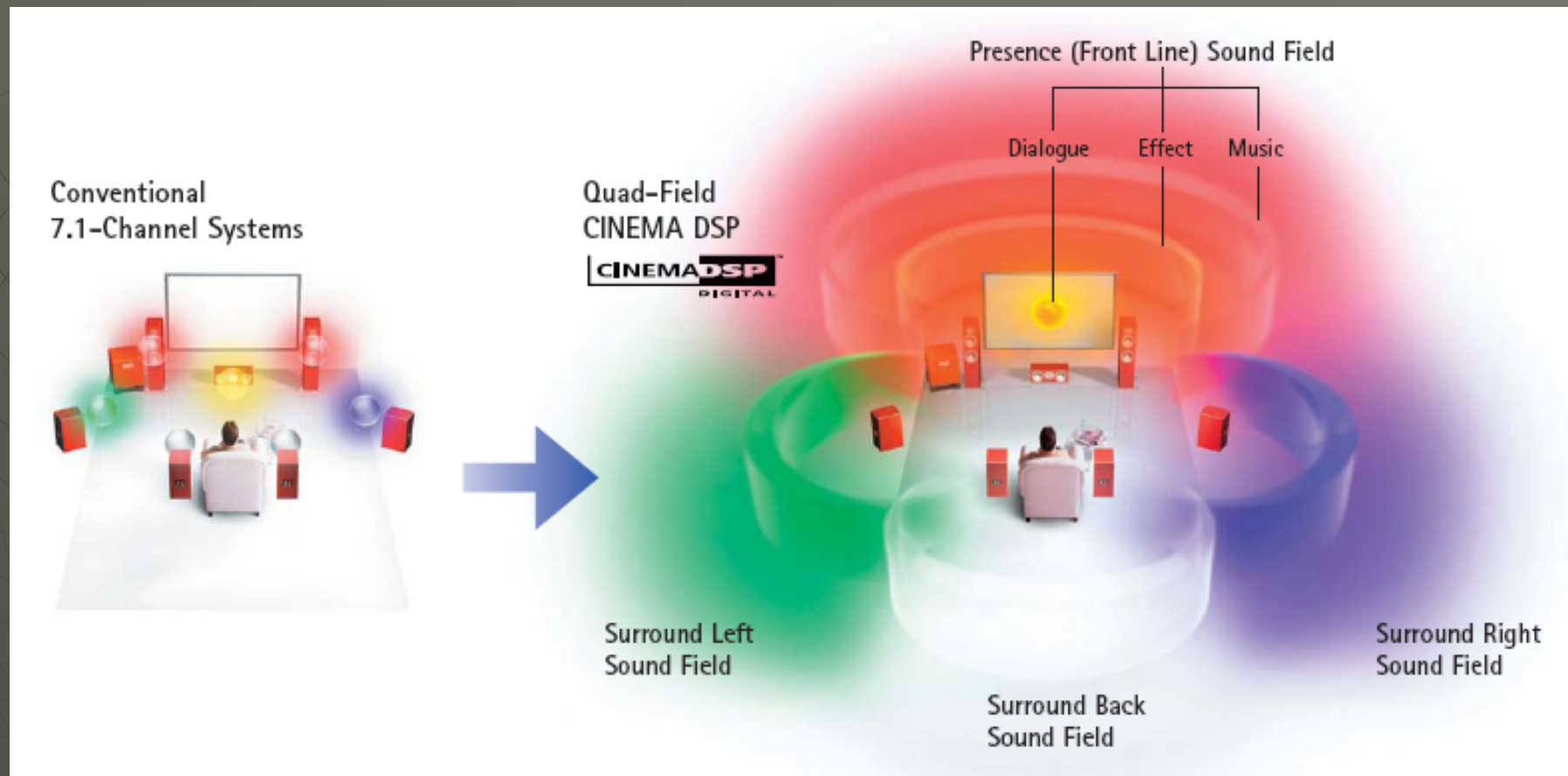




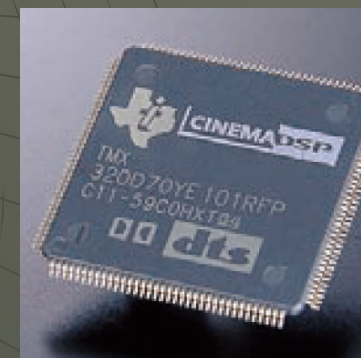
Rys. Tłumienie echa

Przetwarzanie sygnałów akustycznych:

- ✓ cyfrowy zapis, kompresja (np. MP3) i przechowywanie sygnałów, przesyłanie i kopiowanie bez utraty jakości
- ✓ cyfrowe miksowanie i edycja (filtracja) sygnałów
- ✓ efekty dźwiękowe (sztuczny pogłos, *surround*)
- ✓ synteza (np. LPC) i rozpoznawanie mowy (ekstrakcja i dopasowanie cech)
- ✓ aktywne tłumienie szumu



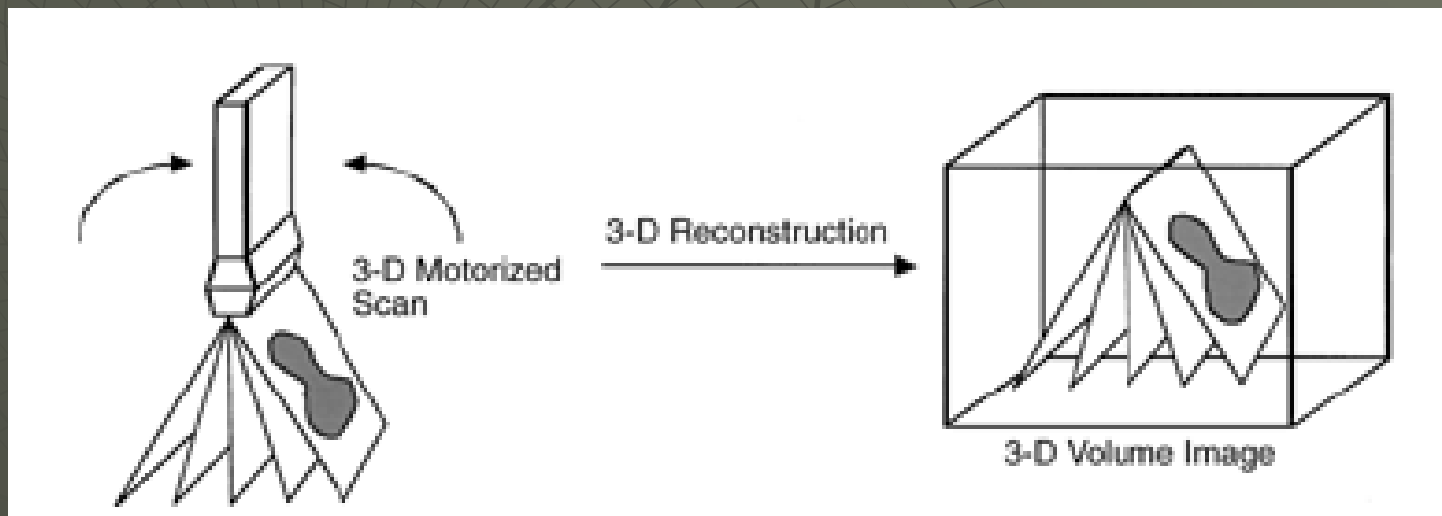
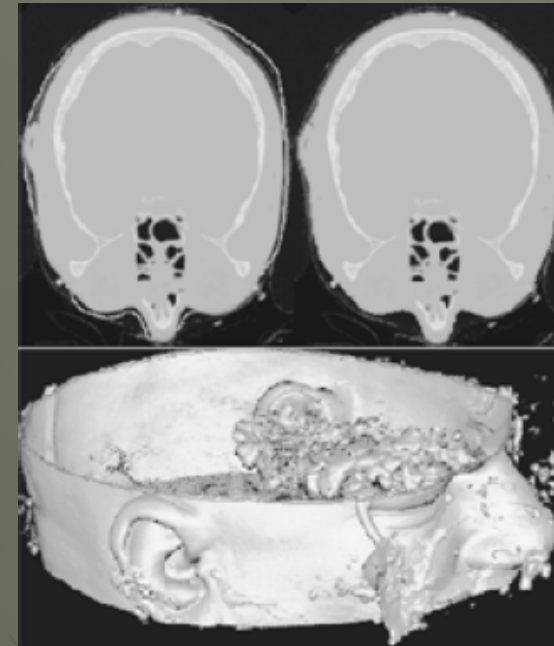
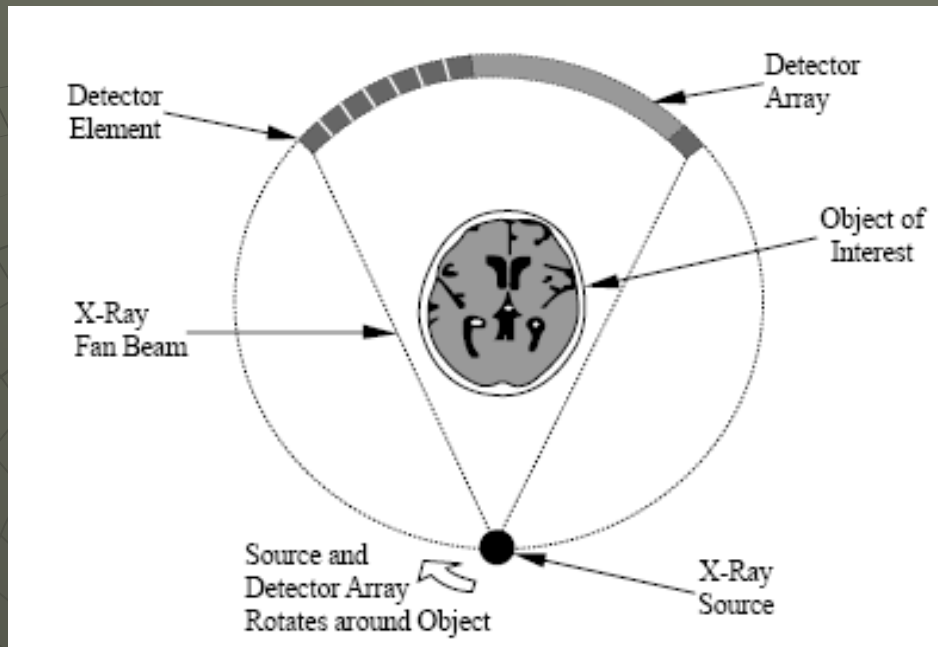
Rys. Kino domowe i procesor sygnałowy TMS320070 realizujący efekty dźwiękowe



New CINEMA DSP LSI

Przetwarzanie obrazów:

- ✓ kompresja obrazów cyfrowych (2D DCT, MPEG4)
- ✓ rekonstrukcja i poprawa jakości (np. zdjęcia z teleskopów, lotnicze)
- ✓ zastosowania biomedyczne (obrazowanie USG, tomografia CT, rezonans magnetyczny MRI)
- ✓ rozpoznawanie obrazów (pismo odręczne, OCR, odciski palców)

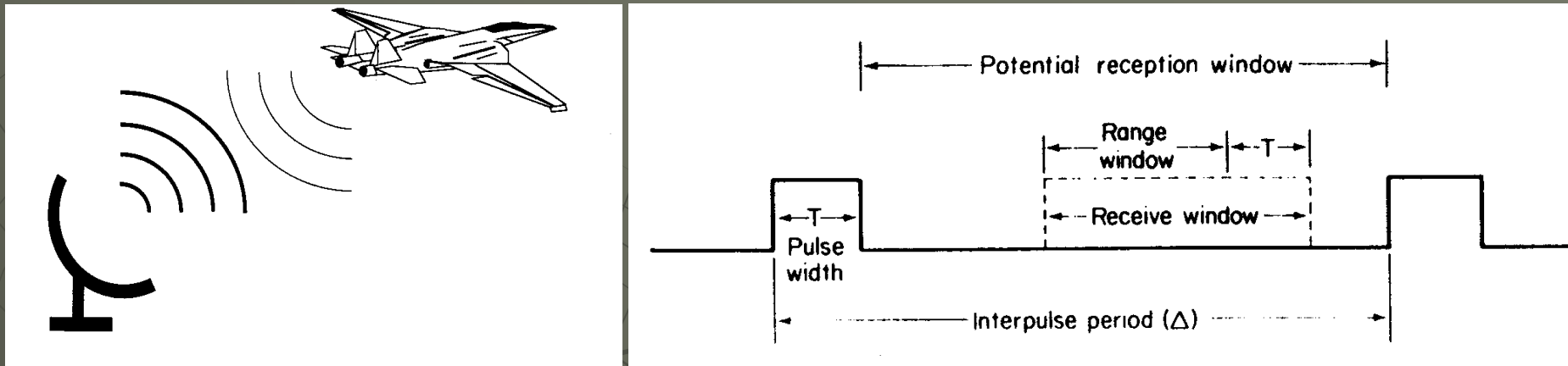


Rys. Zasada działania tomografu komputerowego i tworzenie obrazu 3D

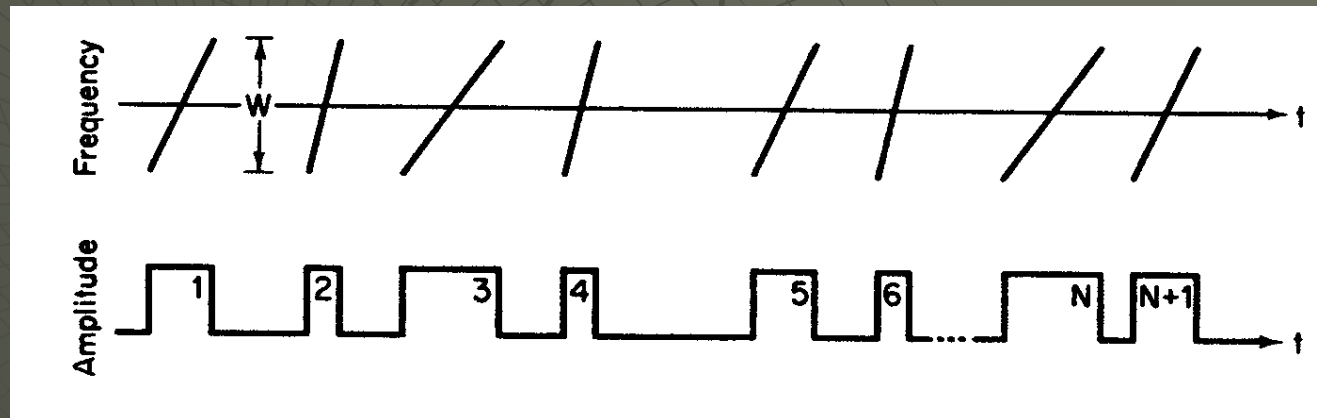
Radary:

Główny problem: częstotliwość przetwarzania (do GHz)

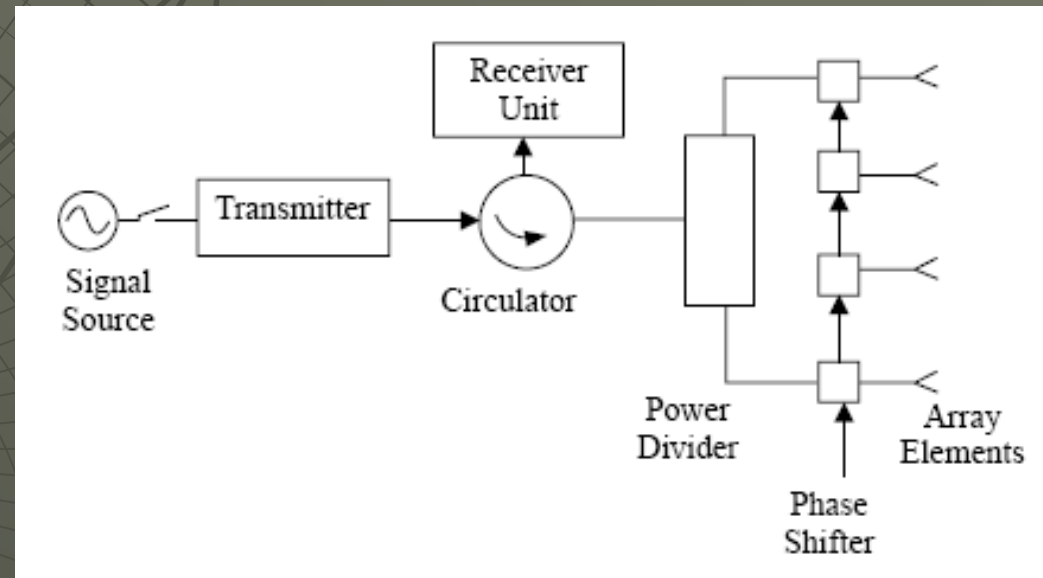
- ✓ wykrywanie i śledzenie obiektów
- ✓ pomiar położenia i prędkości (efekt Dopplera)
- ✓ nawigacja morska i lotnicza
- ✓ obserwacja i prognozowanie pogody
- ✓ geodezja i kartografia



Rys. Zależności czasowe w radarze impulsowym



Rys. Seria niejednorodnych impulsów *LFM* (*Linear Frequency Modulation*) o takim samym paśmie częstotliwości W

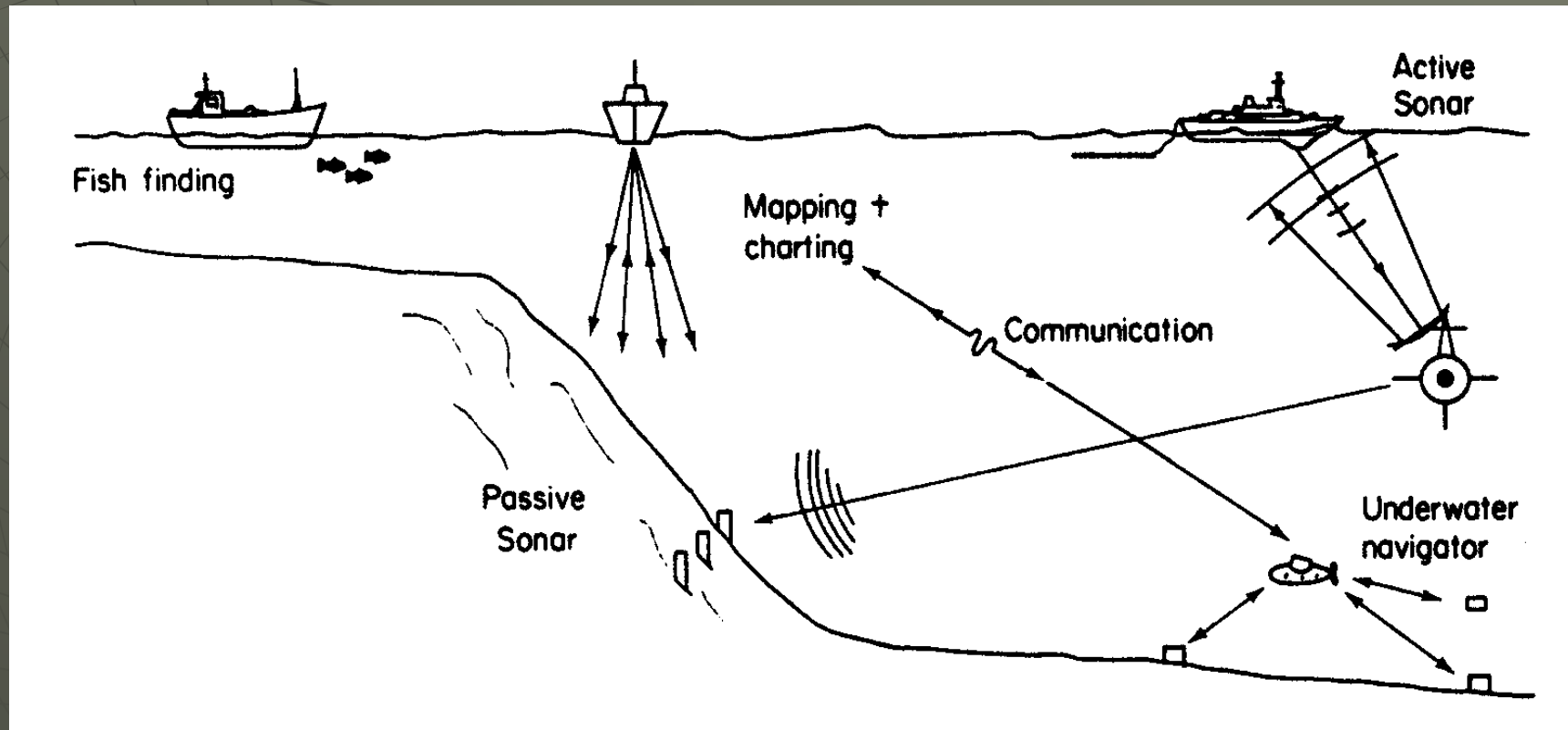


Rys. *Phased Array Radar* systemu AEGIS i zasada jego działania

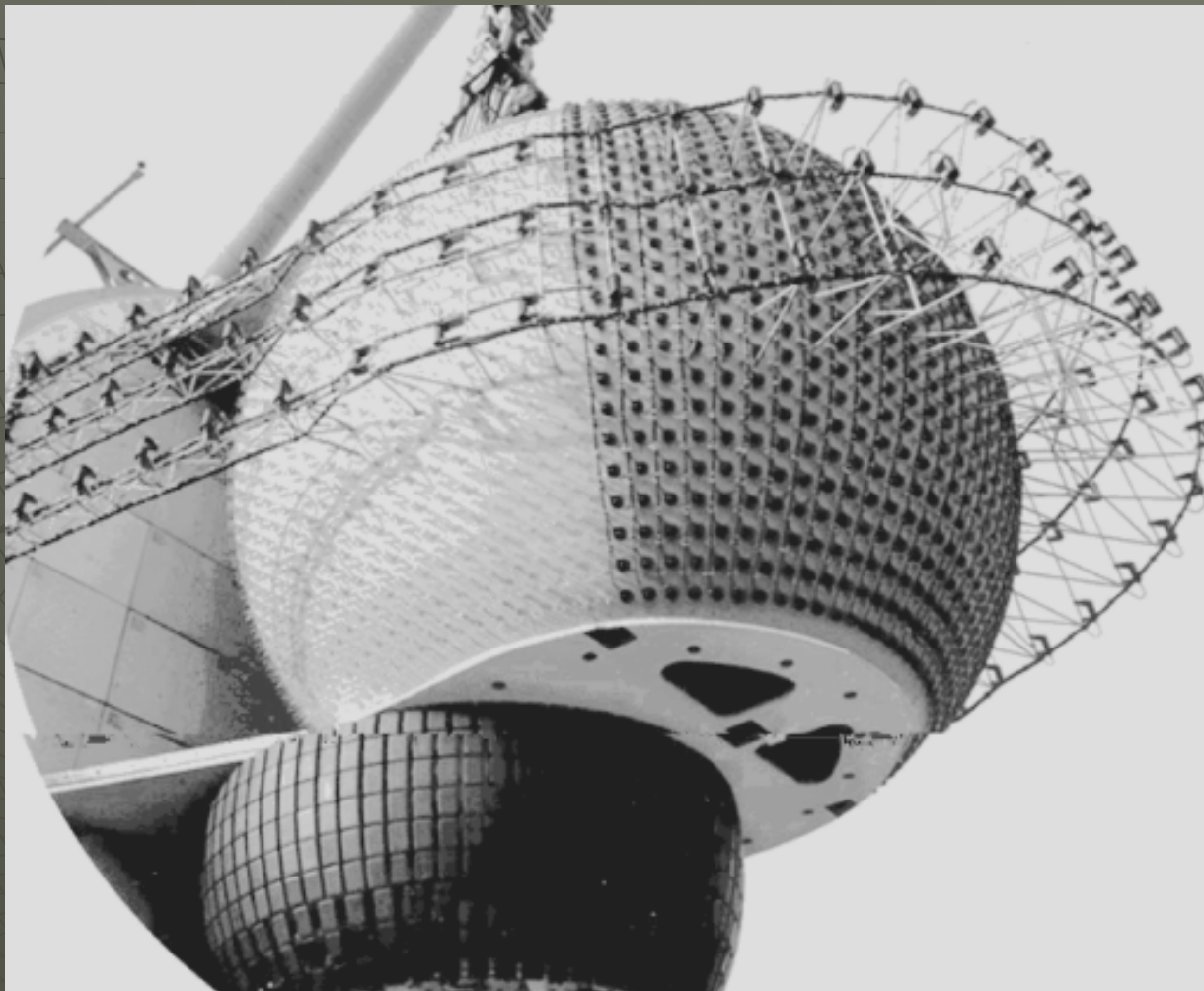
Sonary:

Główny problem: niejednorodność środowiska, macierze elementów (sygnały wielowymiarowe)

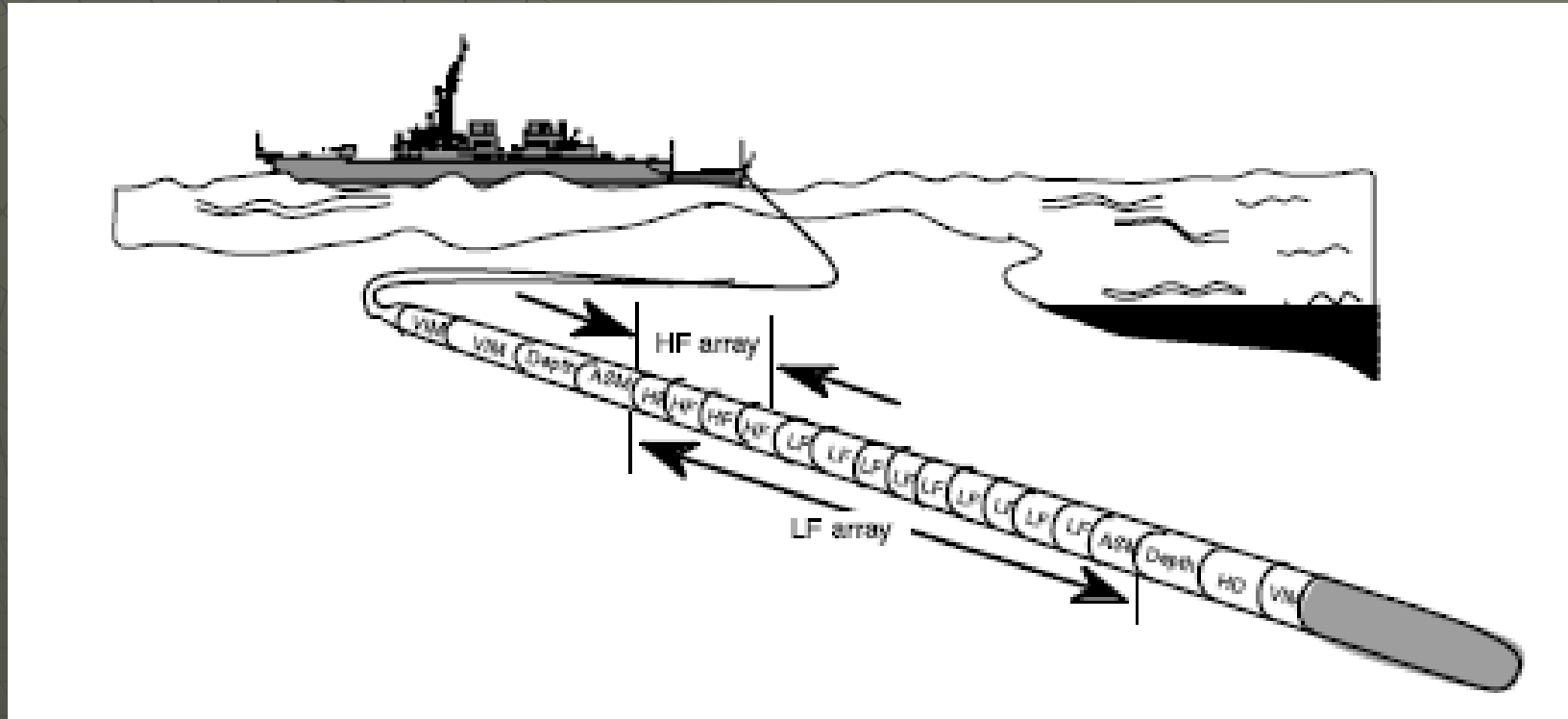
- ✓ wykrywanie i śledzenie obiektów
- ✓ sonary aktywne – synteza impulsów (*beamforming*) i analiza echa (2-40KHz)
- ✓ sonary pasywne – analiza pola echa z hydrofonów
- ✓ badanie zasobów morskich, wykrywanie ławic ryb



Rys. Przykłady zastosowań układów sonarowych



Rys. Matryca elementów sonaru dziobowego na okręcie podwodnym *Seawolf*



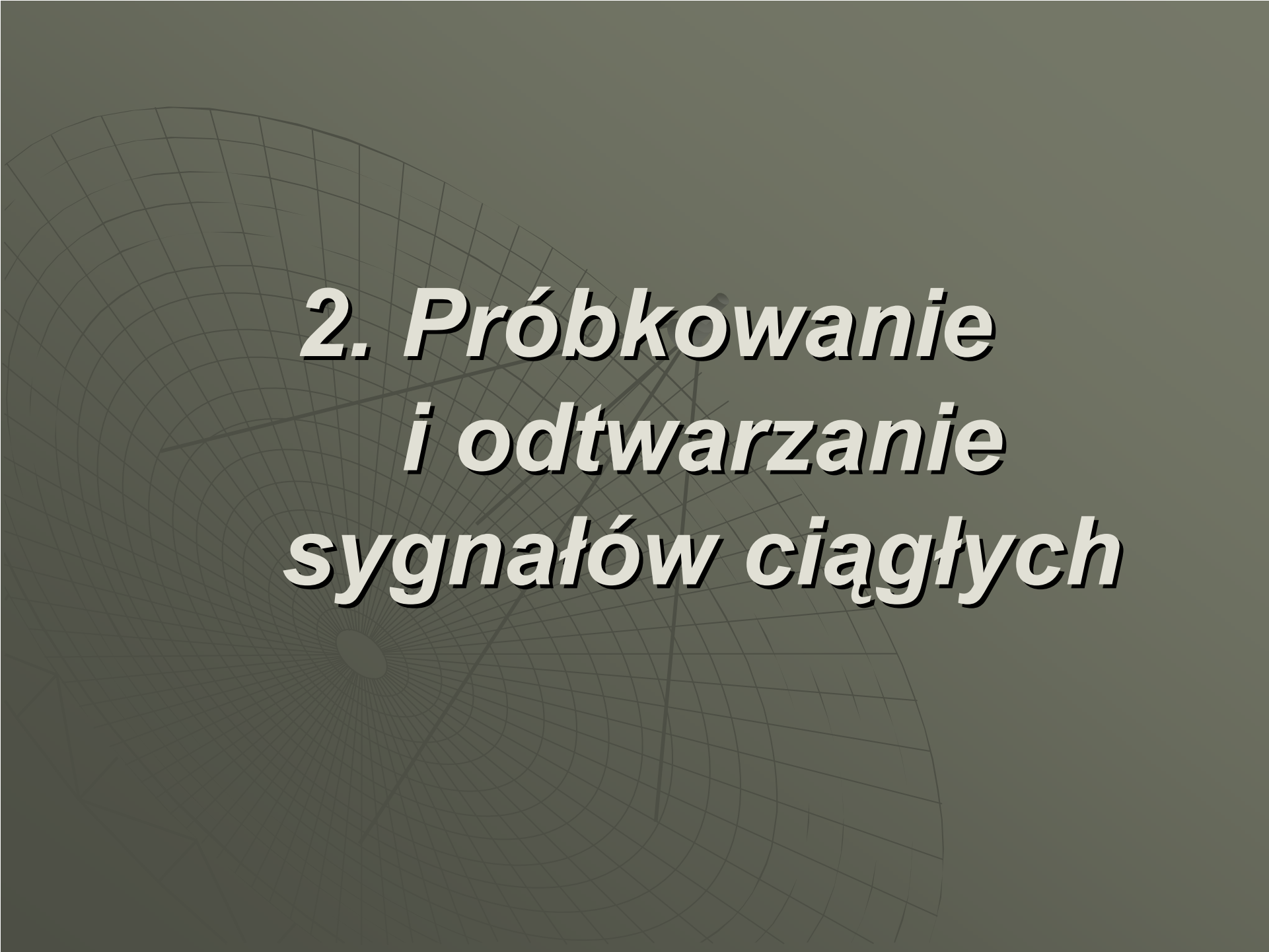
Rys. Liniowy sonar holowany zawierający elementy o różnym przeznaczeniu

Przetwarzanie sygnałów geofizycznych i sejsmicznych

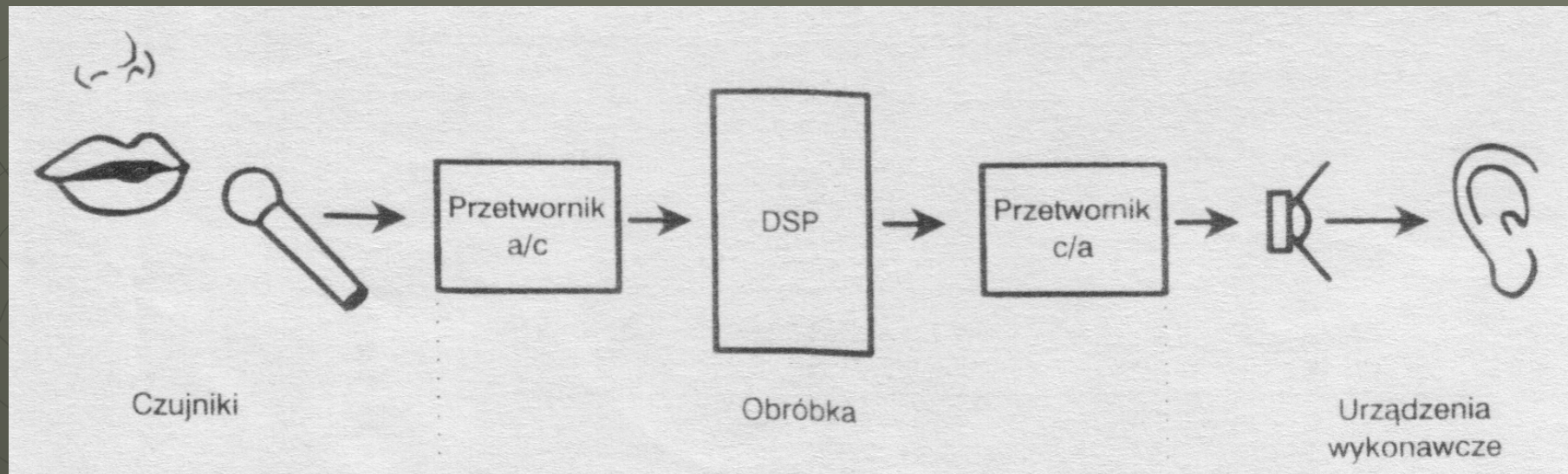
- ✓ seismologia odbiciowa, wykrywanie zasobów naturalnych (np. ropy naftowej)

Badania i wykorzystanie kosmosu

- ✓ komunikacja satelitarna
- ✓ przetwarzanie sygnałów z sond kosmicznych
- ✓ nawigacja satelitarna (GPS, Galileo)

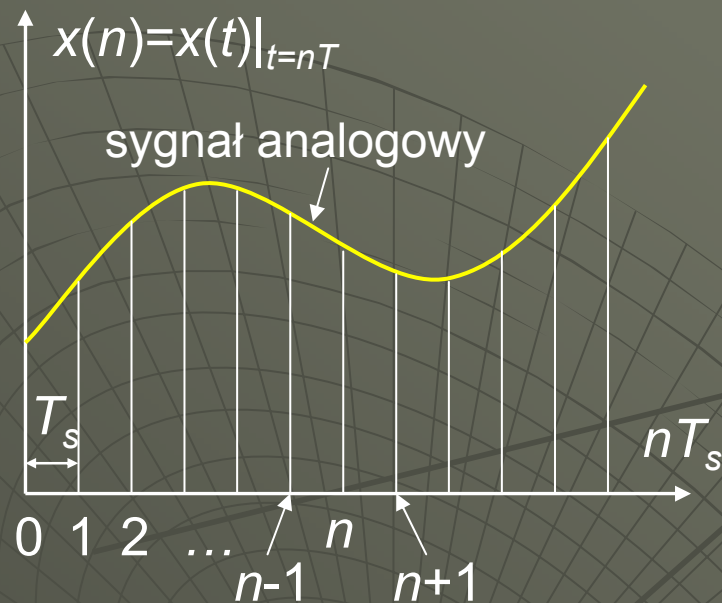


2. Próbkowanie i odtwarzanie sygnałów ciągłych

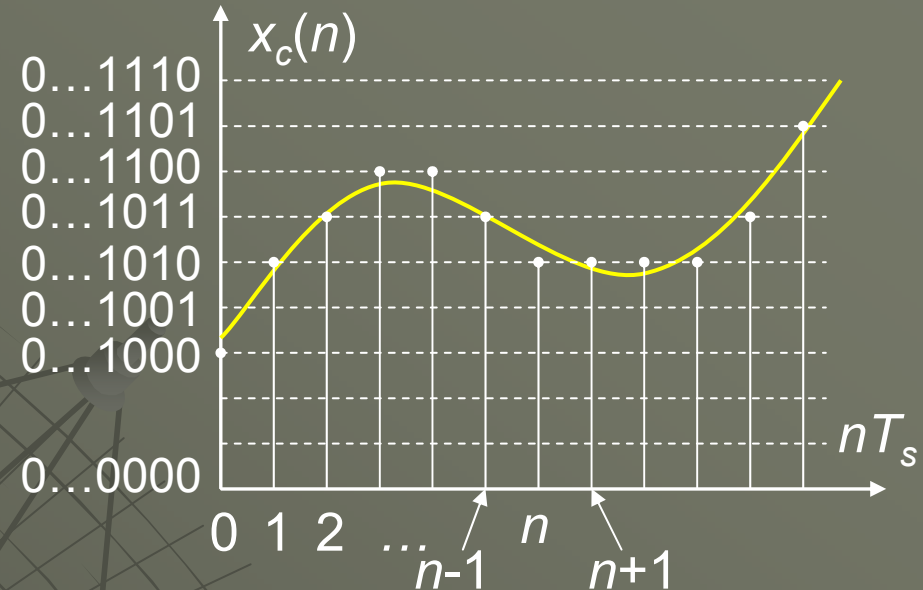


Rys. Przykład toru cyfrowego przetwarzania sygnału dźwiękowego

Próbkowanie i kwantowanie

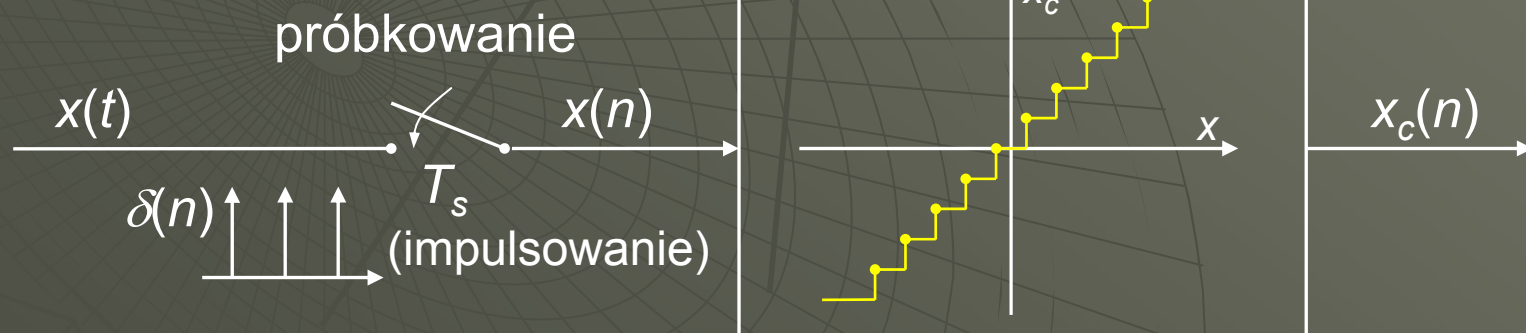


próbkowanie → sygnał dyskretny w czasie



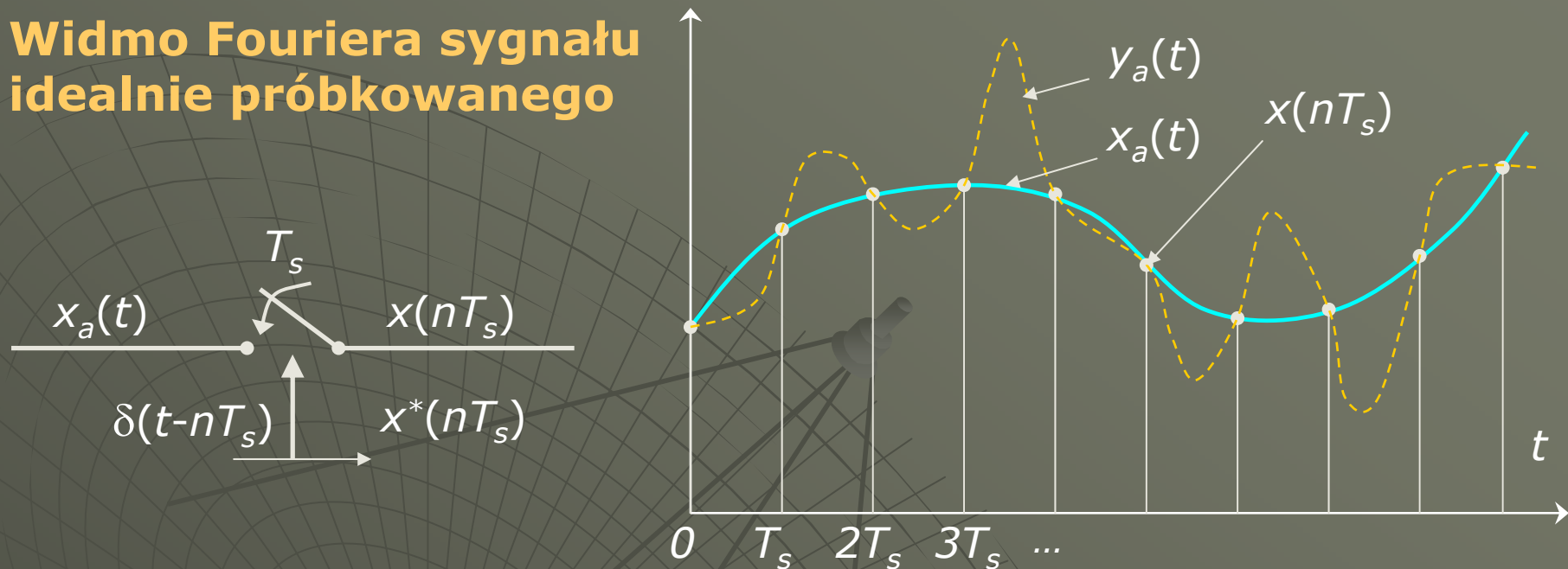
kwantowanie → sygnał cyfrowy

kwantowanie



Rys. Sygnał analogowy, dyskretny w czasie i cyfrowy (przetwarzanie A/C)

Widmo Fouriera sygnału idealnie próbkowanego



W wyniku próbkowania *impulsowego* funkcji analogowej (ciągłej w czasie) $x_a(t)$ otrzymuje się funkcję impulsową:

$$x^*(t) = x_a(t) \cdot \sum_{n=-\infty}^{\infty} \delta(t - nT_s) = x_a(t) \cdot f^*(t)$$

Wartości funkcji dyskretnej otrzymuje z wartości impulsowych:

$$x(nT_s) = x(n) = \int_{-\infty}^{\infty} x^*(nT_s) dt = \int_{-\infty}^{\infty} x_a(t) \delta(t - nT_s) dt$$

Ciąg impulsów stanowi funkcję okresową, którą reprezentuje zespolony szereg Fouriera:

$$f^*(t) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} e^{jk\omega_s t} \quad \text{gdzie: } \omega_s = 2\pi f_s = \frac{2\pi}{T_s}$$

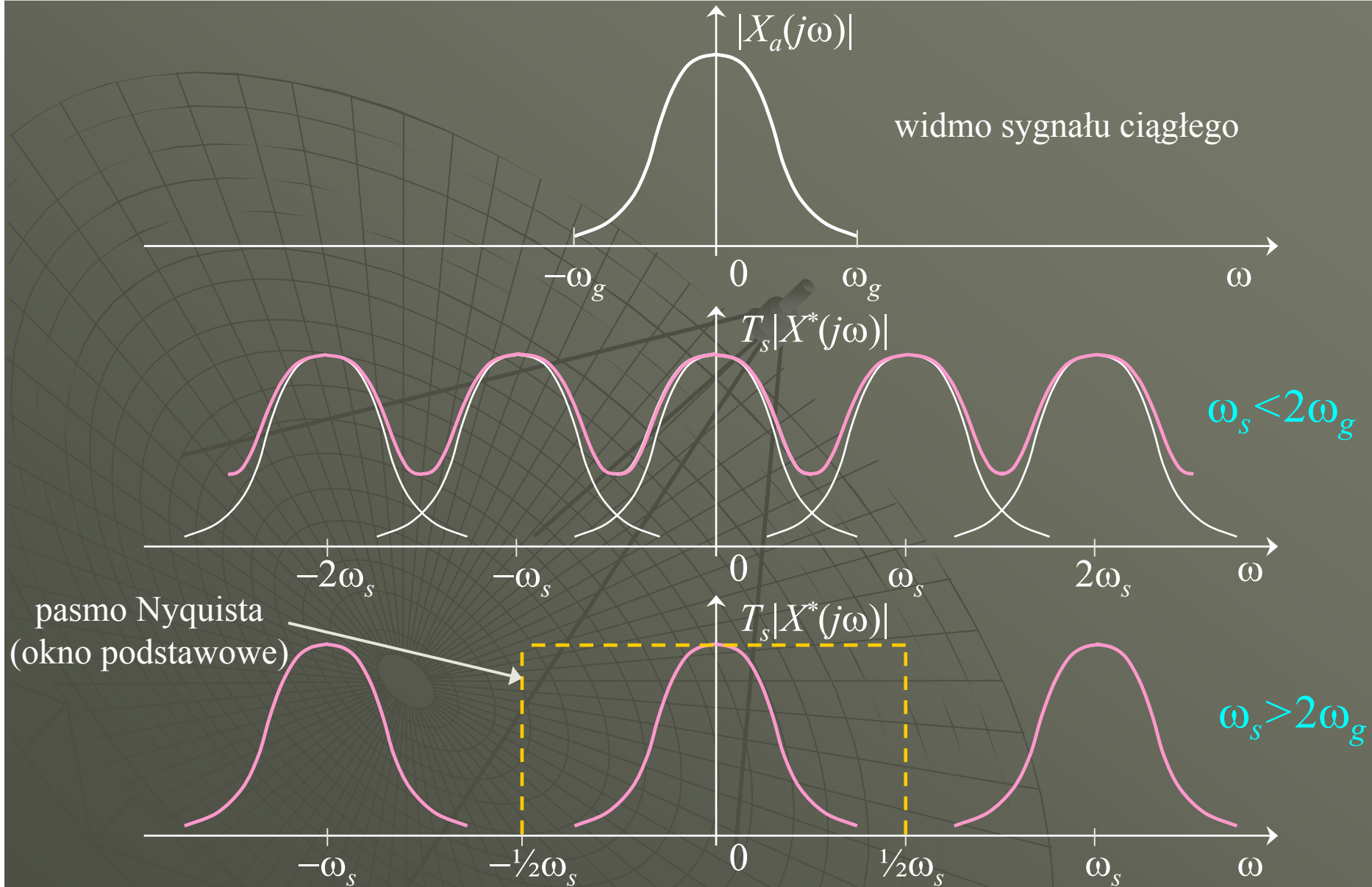
Stąd:

$$x^*(t) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} x_a(t) e^{jk\omega_s t}$$

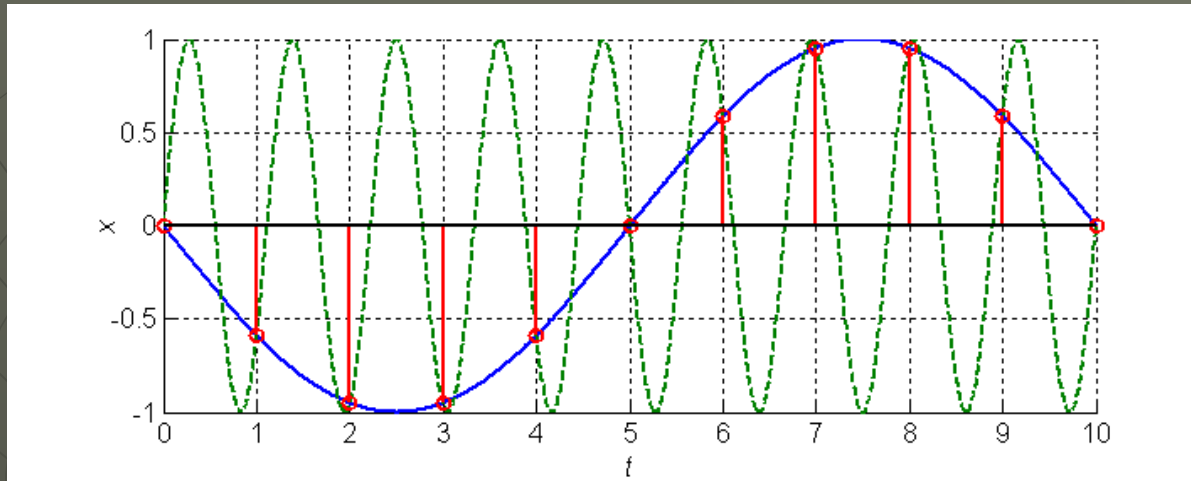
Transformata Fouriera funkcji impulsowej (próbkowanej):

$$X^*(j\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X_a(j\omega - jk\omega_s)$$

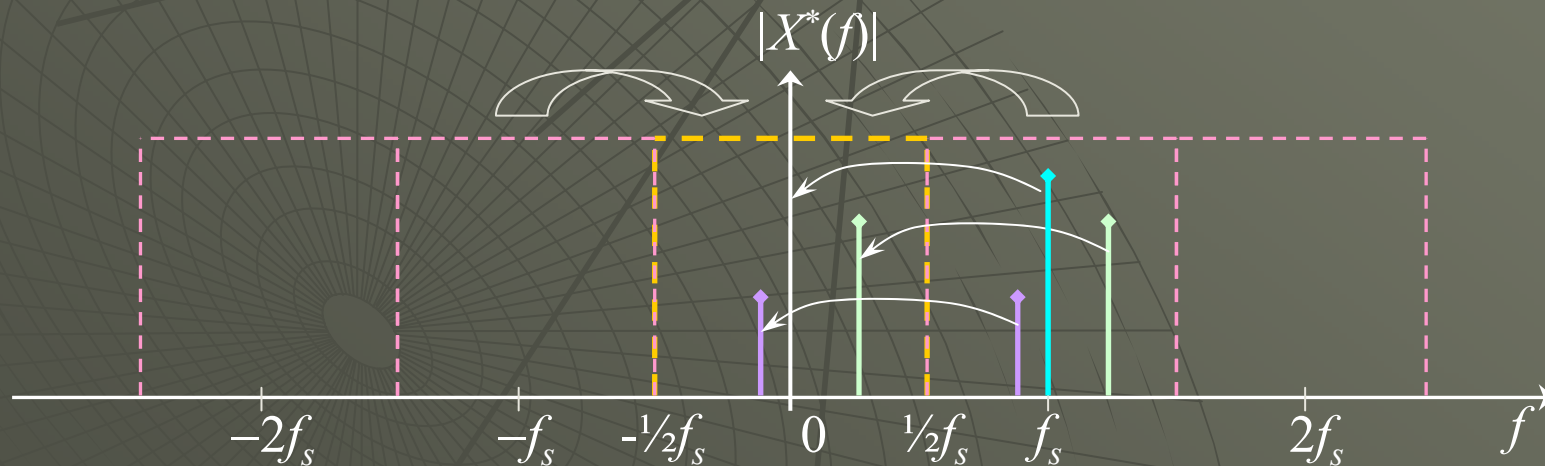
gdzie: $X_a(j\omega) = \int_{-\infty}^{\infty} x_a(t) e^{-j\omega t} dt$ jest transformatą funkcji ciągłej.



Rys. Moduł widma Fouriera sygnału ciągłego $x_a(t)$ oraz powielone widma sygnału impulsowego $x^*(nT_s)$. W praktyce $\omega_s = 5 \div 10 \omega_g$



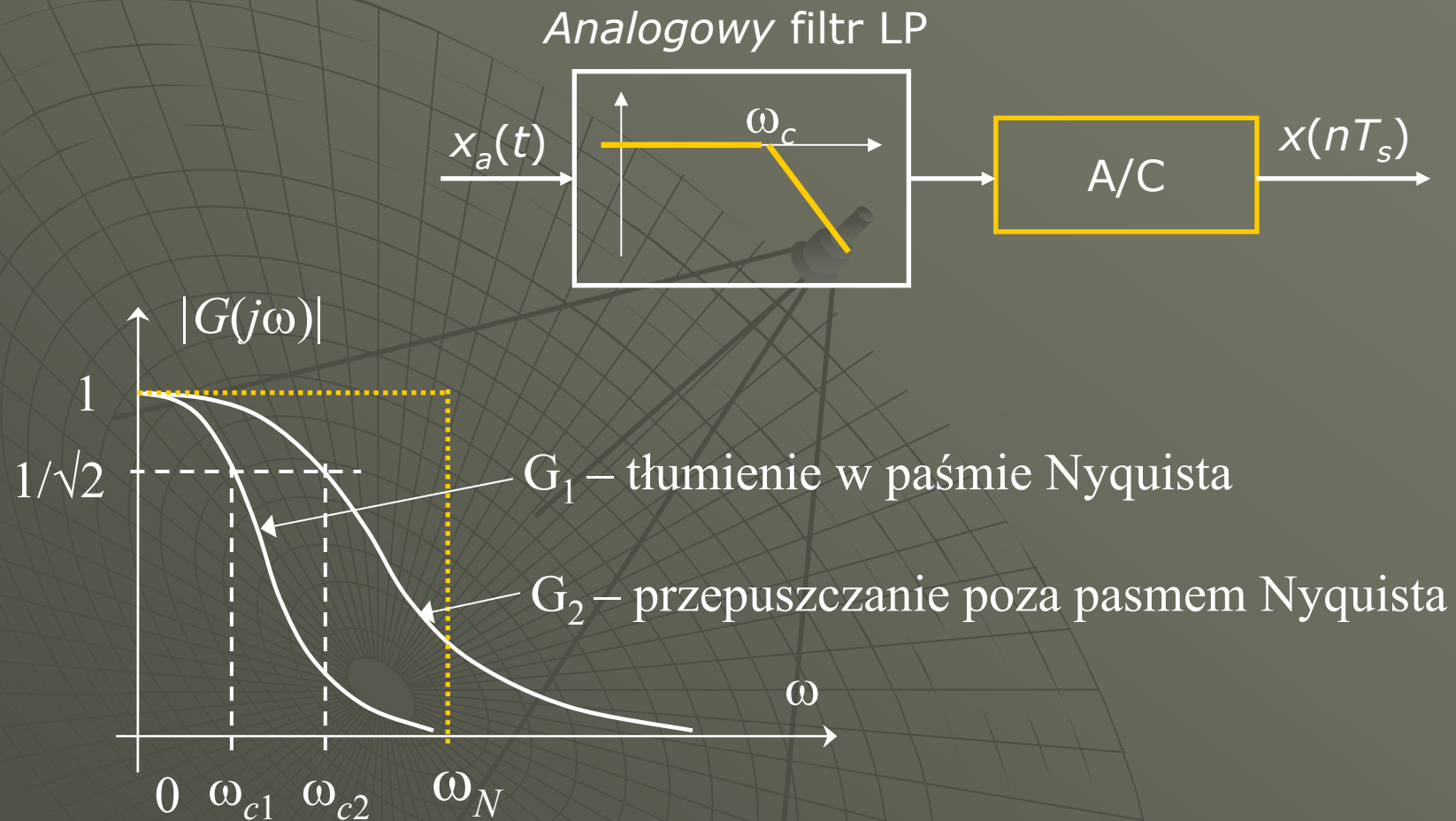
Rys. *Aliasing* w dziedzinie czasu: dwa sygnały o częstotliwościach 0.9Hz i 0.1Hz mogą mieć takie same wartości w chwilach próbkowania, $f_s=1\text{Hz}$



Rys. *Aliasing* w dziedzinie częstotliwości: nakładanie się okien wokół wielokrotności f_s na pasmo podstawowe

$\frac{1}{2}f_s = f_N$ – częstotliwość Nyquista

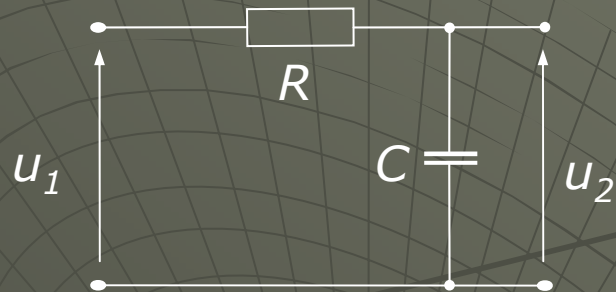
Filtr antyaliasingowy



Rys. Problem wyboru szerokości pasma filtra antyaliasingowego
 Zwykle filtr projektuje się tak, żeby

$$0.001 \leq |G(j\omega_N)| \leq 0.5$$

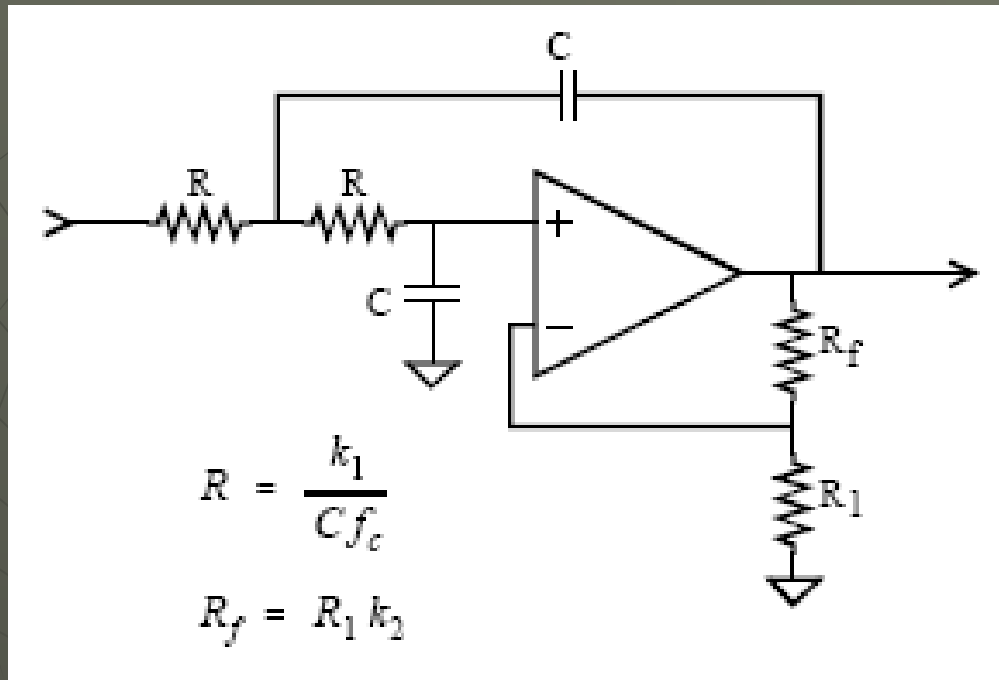
Najprostszy filtr RC ma małą stromość ch-ki amplitudowej poza pasmem przepuszczania.



$$G(s) = \frac{1}{Ts + 1}, \quad T = \frac{1}{\omega_c} = RC$$

Możliwości:

- 1) zwiększenie stromości opadania ch-ki filtra poza pasmem przepuszczania (rzędu filtra) → bardziej skomplikowana część analogowa układu
- 2) zwiększenie częstotliwości próbkowania (szerokości okna Nyquista) → potrzebny szybszy procesor (większa częstość obliczeń) i przetwornik A/C

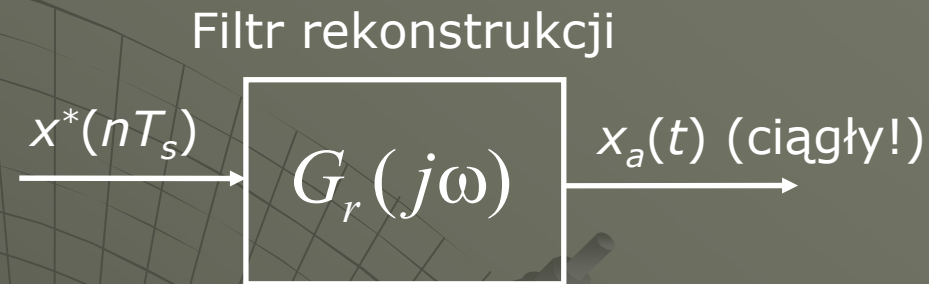


Rys. Aktywny filtr dolnoprzepustowy (LP) 2-go rzędu. Współczynniki k_1 i k_2 dobiera się z tabeli, R_1 i C można wybrać dowolnie (np. $10\text{k}\Omega$ i $0.01\mu\text{F}$), a następnie z podanych równań oblicza się R i R_f . Parametr f_c jest częstotliwością odcięcia (cutoff) filtra w Hz. Filtry wyższych rzędów można uzyskać przez szeregowe łączenie sekcji 2-go rzędu.

$$G(s) = \frac{1}{\left(\frac{s}{\omega_c}\right)^2 + 2\zeta \frac{s}{\omega_c} + 1}, \quad \omega_c = 2\pi f_c$$

Rząd filtra	Nr sekcji	Bessel		Butterworth		Czebyszew (6% ripple)	
		k_1	k_2	k_1	k_2	k_1	k_2
2	1	0.1251	0.268	0.1592	0.586	0.1293	0.842
4	1	0.1111	0.084	0.1592	0.152	0.2666	0.582
	2	0.0991	0.759	0.1592	1.235	0.1544	1.660

Odtwarzanie sygnału ciągłego z próbek

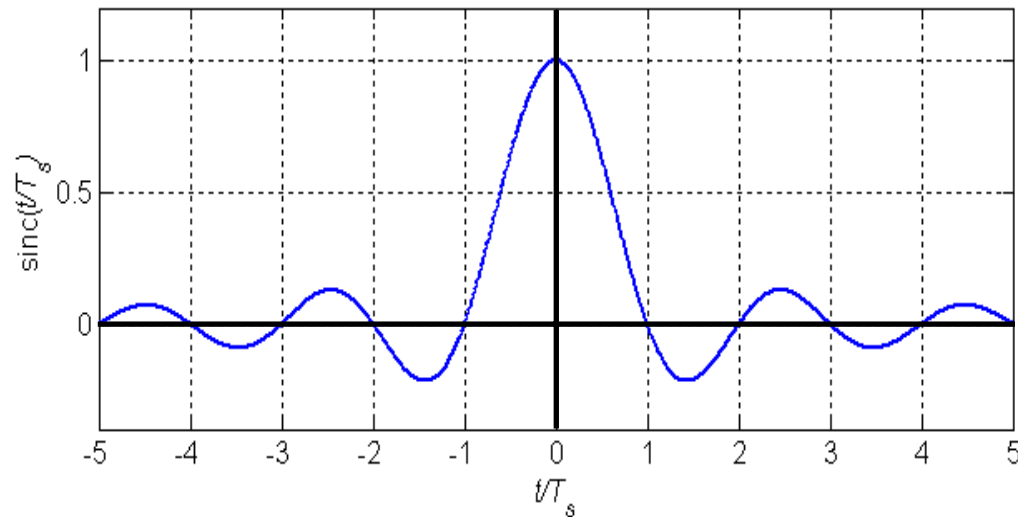


Tw. Shannona (o rekonstrukcji): Jeżeli sygnał ciągły $x_a(t)$ o ograniczonym widmie był próbkowany z częstotliwością $\omega_c > 2\omega_g$, to można go odtworzyć z próbek przy pomocy interpolacji:

$$x_a(t) = \sum_{n=-\infty}^{\infty} x(nT_s) \operatorname{sinc} \frac{t - nT_s}{T_s}$$

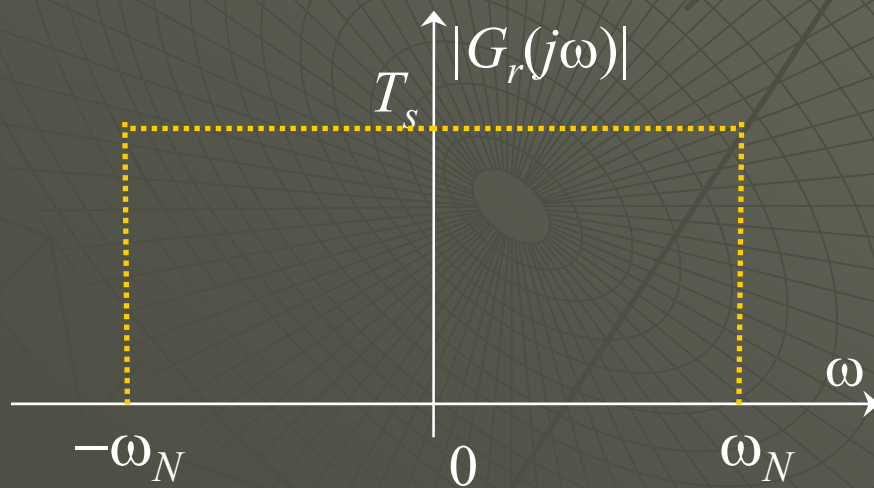
gdzie: $\operatorname{sinc} \frac{t}{T_s} = \frac{\sin \omega_N t}{\omega_N t}$ $\omega_N = \frac{\omega_s}{2} = \frac{\pi}{T_s}$

jest funkcją wag próbek.



Rys. Funkcja wagowa

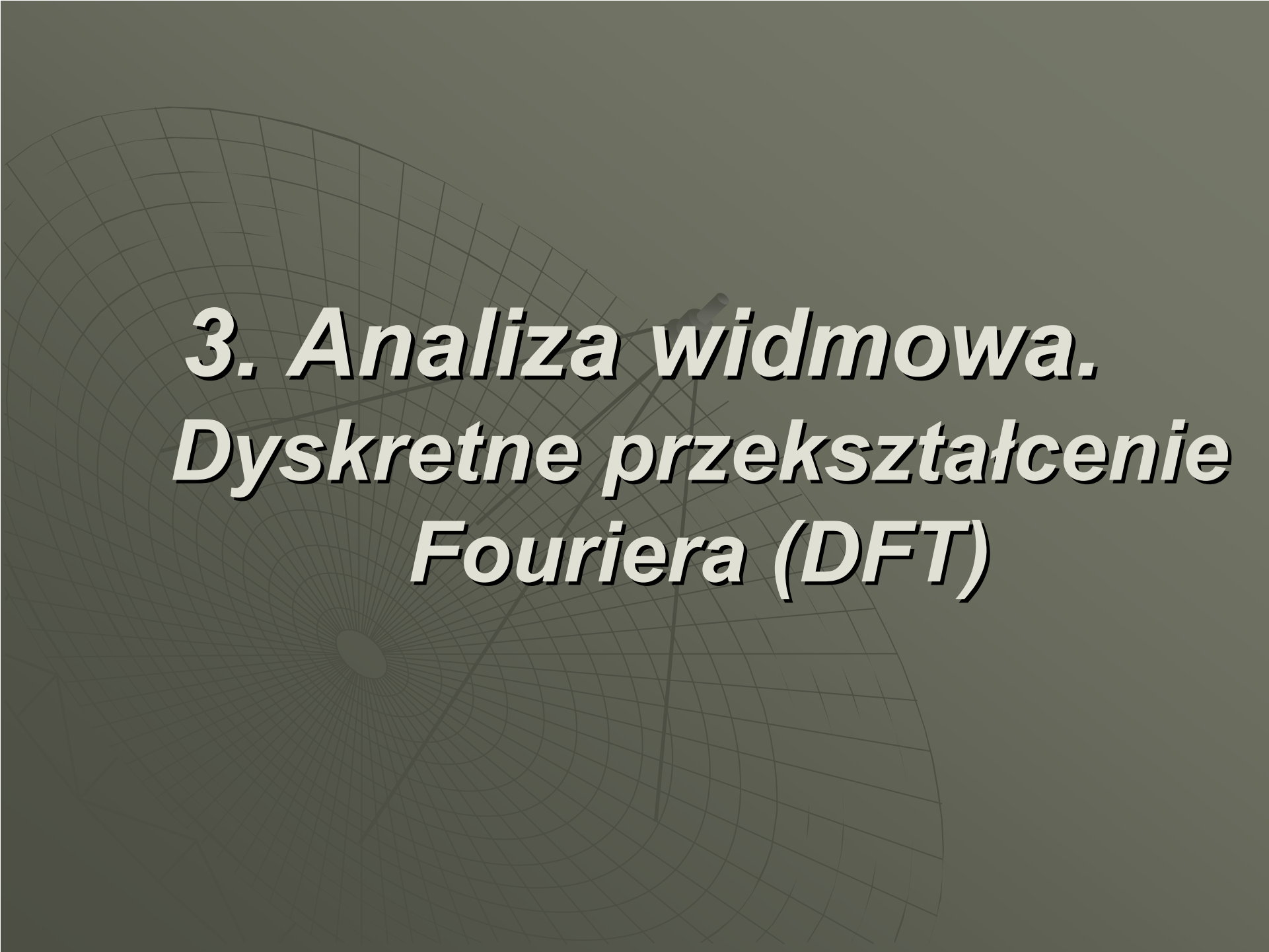
$$g_r(t) = \text{sinc} \frac{t}{T_s}$$



Rekonstrukcja polega na przepuszczeniu sygnału impulsowego $x^*(nT_s)$ przez filtr analogowy (nieprzyczynowy) $G_r(j\omega)$ o prostokątnej charakterystyce amplitudowej, zerowym przesunięciu fazowym i odpowiedzi impulsowej $g_r(t)$.



Rys. Rekonstrukcja przyczynowa (uciąglenie) sygnału z próbek przez przetwornik C/A (zero order hold) – przebieg schodkowy



***3. Analiza widmowa.
Dyskretne przekształcenie
Fouriera (DFT)***

Transformatą Fouriera (widmem Fourierowskim *DTFT* – *Discrete Time Fourier Transform*) sygnału dyskretnego w czasie $x(n)$ i o ograniczonej energii, tzn. spełniającego warunek

$$\sum_{n=-\infty}^{\infty} |x(n)|^2 < \infty$$

nazywamy funkcję

$$DTFT[x(n)] = X(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\Omega n}$$

gdzie

$$\Omega = \omega T_s = 2\pi \frac{f}{f_s}$$

jest pulsacją unormowaną względem częstotliwości próbkowania.

Rozwijając ją w szereg Fouriera otrzymamy współczynniki określające kolejne próbki sygnału dyskretnego:

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\Omega}) e^{jn\Omega} d\Omega$$

Właściwości *DTFT*:

- jest funkcją *okresową kąta* Ω o okresie 2π ,
- jest funkcją *ciągłego argumentu* Ω ,
- jest obliczana na podstawie *nieskończonego ciągu próbek*,
- dla sygnału rzeczywistego amplituda transformaty jest funkcją parzystą, a faza – funkcją nieparzystą.

Z właściwości tych wynika, że widmo sygnału dyskretnego wystarczy przedstawić w zakresie kątów $0 \leq \Omega \leq \pi$ lub częstotliwości $0 \leq f \leq f_s/2$. Pulsację $\Omega_N = \pi$ (częstotliwość $f_N = f_s/2$) nazywa się pulsacją Nyquista. Jest to największa częstotliwość okna pasma podstawowego sygnału dyskretnego.

Podstawowe właściwości DTFT

Jeżeli $DTFT[x(n)] = X(e^{j\Omega})$, to:

1) widmo sygnału modulowanego $x(n)e^{jn\Omega_0}$ ma postać $X(e^{j(\Omega-\Omega_0)})$
(przesunięcie skali pulsacji),

2) widmo sygnału przesuniętego w czasie $x(n-d)$ ma postać
 $e^{-jd\cdot\Omega} X(e^{j\Omega})$ (przesunięcie w fazie),

3) widmo splotu sygnałów $x(n) * y(n) = \sum_{k=-\infty}^{\infty} x(k)y(n-k)$
jest iloczynem widm $X(e^{j\Omega}) \cdot Y(e^{j\Omega})$

4) widmo iloczynu sygnałów $x(n) \cdot y(n)$ jest splotem ich widm

$$X(e^{j\Omega}) * Y(e^{j\Omega}) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\theta}) Y(e^{j(\Omega-\theta)}) d\theta$$

Dyskretne przekształcenie Fouriera

Widmo wycinka sygnału (o skończonej liczbie N próbek):

$$x_0(n) = \begin{cases} x(n) & \text{dla } 0 \leq n \leq N-1 \\ 0 & \text{dla } n \geq N \end{cases}$$

$$x_0(n) = x(n) \cdot w(n), \quad \text{gdzie } w(n) = 1 \text{ dla } 0 \leq n \leq N-1 \text{ i } 0 \text{ dla } n \geq N$$

$$DTFT[x_0(n)] = X_0(e^{j\Omega}) = X_0(\Omega) = \sum_{n=0}^{N-1} x_0(n) e^{-j\Omega n}$$

Stosując cyfrowe metody obliczeń transformatę Fouriera można obliczyć tylko dla skończonego zbioru dyskretnych wartości Ω_m . Obliczmy N wartości widma (tyle ile jest próbek sygnału) dla *równomiernie rozłożonych* w zakresie $[0, 2\pi]$ pulsacji unormowanych

$$\Omega_m = \frac{2\pi m}{N}, \quad m = 0, 1, \dots, N-1$$

Otrzymujemy w ten sposób *próbki* ciągłego widma $DTFT[x_0(n)]$.

Jest to N -punktowe dyskretne przekształcenie Fouriera DFT (*Discrete Fourier Transform*). Przyporządkowuje ono skończonemu ciągowi N próbek $x_0(n)$ zespolone współczynniki:

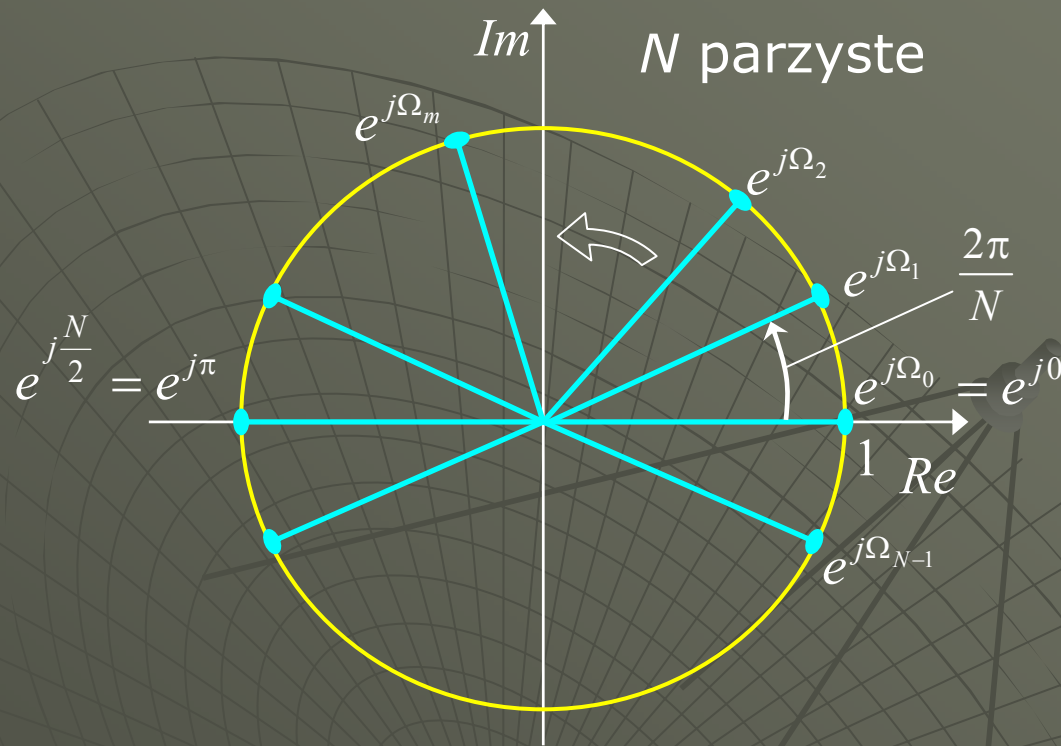
$$DFT[x_0(n)] = X_0(\Omega_m) = X_0(m) = \sum_{n=0}^{N-1} x_0(n) e^{-jn\Omega_m} \quad m = 0, 1, \dots, N-1$$

Ważne właściwości DFT :

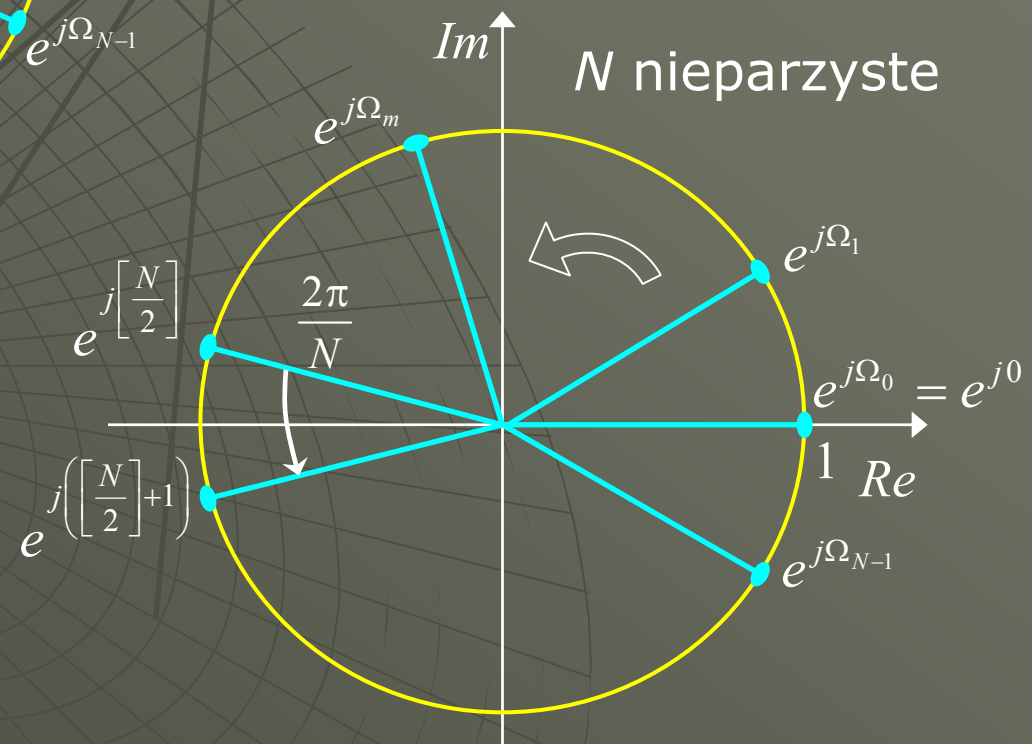
- współczynniki DFT nie zmieniają wartości bezwzględnych po *cyklicznym* przestawieniu próbek (np. $x_0(0) \rightarrow x_0(1)$, $x_0(1) \rightarrow x_0(2)$, ..., $x_0(N-1) \rightarrow x_0(0)$), a ich fazy zmieniają się o $2\pi/N$,
- współczynniki DFT sygnału rzeczywistego mają wartości *parami zespolone sprzężone*: $X_0(N-1) = X_0^*(1)$, $X_0(N-2) = X_0^*(2)$ itd., dlatego istotnie różne są współczynniki w zakresie $0 \leq m \leq [N/2]$.

Rzeczywiste częstotliwości f_m próbek w Hz otrzymuje się nadając skalę zgodnie z częstotliwością próbkowania f_s :

$$f_m = \frac{\Omega_m}{2\pi} \cdot f_s = \frac{m}{N} \cdot f_s \quad [\text{Hz}], \quad \text{np. } f_1 = \frac{f_s}{N}$$



Rys. Graficzna interpretacja współczynników fazowych DFT na płaszczyźnie zespolonej dla różnych Ω_m , $m=0, 1, \dots, N-1$



Obliczanie DFT zamiast widma ciągłego $DTFT$ wnosi zniekształcenia wynikające z:

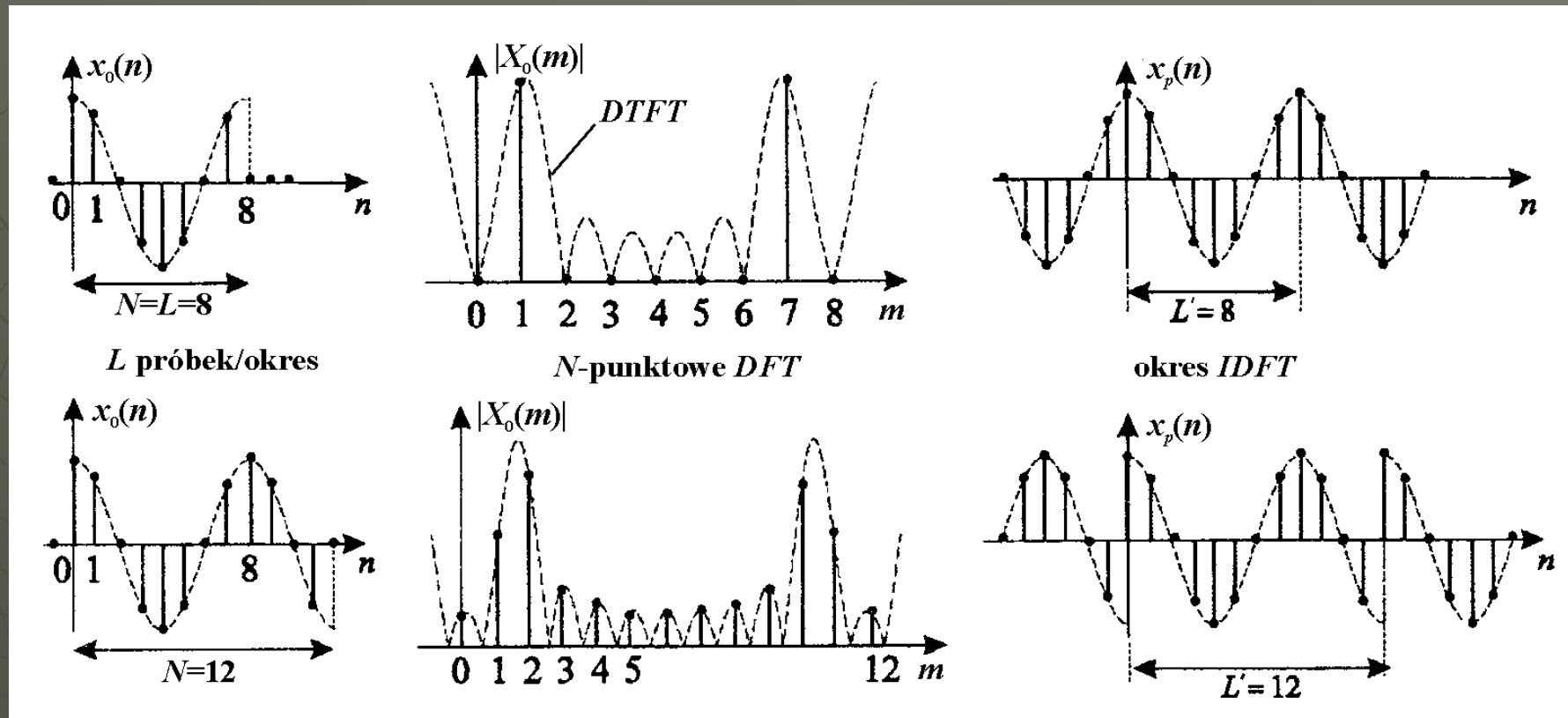
- 1) brania do obliczeń jedynie skończonej liczby próbek (nakładania okna czasowego),
- 2) obliczania dyskretnego zbioru wartości transformaty (próbkowania widma ciągłego).

Przekształcenie odwrotne $IDFT$ (*Inverse DFT*) umożliwia bezpośrednio odtworzenie ciągu próbek $x_0(n)$:

$$IDFT[X_0(m)] = x_0(n) = \frac{1}{N} \sum_{m=0}^{N-1} X_0(m) e^{jn\Omega_m}, \quad n = 0, 1, \dots, N-1$$

Próbkowanie widma w częstotliwości powoduje, że operacja $IDFT[X_0(m)]$ formalnie daje *okresowe powielenie* ciągu próbek, tzn. sygnał okresowy $x_p(n)$ o okresie N :

$$IDFT[X_0(m)] = x_p(n) = \sum_{k=-\infty}^{\infty} x_0(n + kN)$$



Rys. Moduł N -punktowej DFT wycinka $x_0(n)$ sygnału okresowego, w którym przypada $L=8$ próbek/okres i sygnały okresowe o okresie L' próbek odtworzone na podstawie transformaty odwrotnej $IDFT[X_0]$. Jeżeli w okresie obserwacji nie mieści się całkowita liczba okresów ($N \neq kL$) składowej harmonicznnej o pulsacji Ω_0 , to pulsacja ta wypada pomiędzy pulsacjami Ω_m prążków DFT i część widma obrazująca tę składową ulega rozmyciu (*leakage*)

Wniosek: $DFT[x_0(n)]$ jest równe $DTFT$ jednego okresu sygnału okresowego, którego wycinek okresowy jest równy $x_0(n)$, $n=0,1,\dots,N-1$.

Rozdzielczość DFT

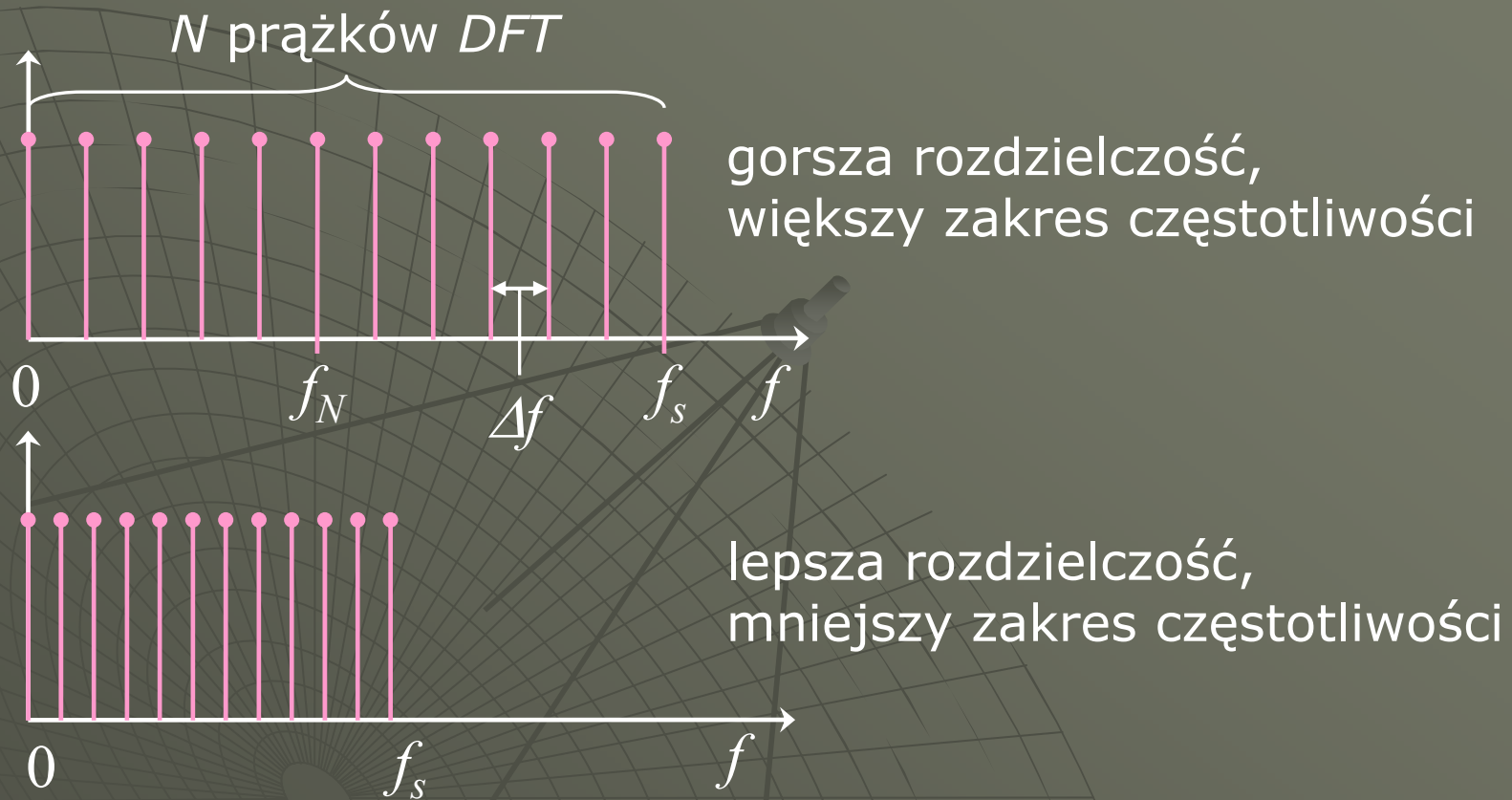
Rozdzielczość częstotliwościowa w dziedzinie częstotliwości jest określana jako odległość między kolejnymi prążkami DFT (jest to zdolność rozróżnienia w sygnale dwóch składowych o mało różniących się częstotliwościach):

$$\Delta f_N = \frac{1}{NT_s} = \frac{1}{T_0}$$

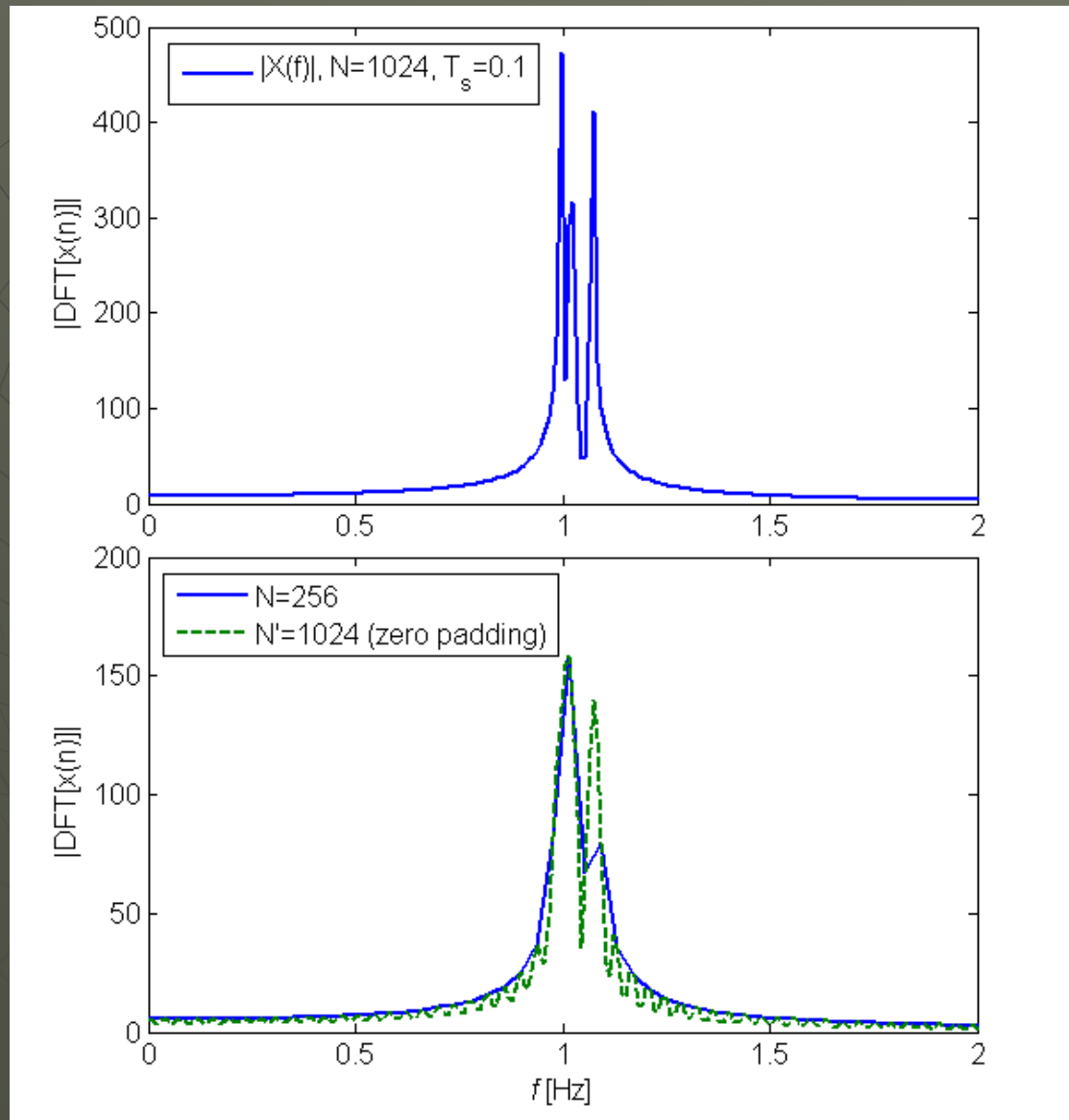
gdzie $T_0 = NT_s$ jest czasem obserwacji (długością wycinka sygnału w czasie).

Liczbę prążków DFT można zwiększyć poprzez uzupełnienie zarejestrowanego ciągu N_1 próbkami zerowymi (*zero-padding*, sztuczne wydłużenie czasu obserwacji). Nowe prążki powstają jako interpolacja oryginalnych. Teoretyczna nowa rozdzielczość

$$\Delta f' = 1 / [(N + N_1)T_s]$$



Rys. Zmiana rozdzielczości częstotliwościowej i zakresu widma *DFT* przy zmianie częstotliwości próbkowania, $N=\text{const}$



Rys. Widma DFT sygnału $x_a(t)$ próbkowanego z okresem $T_s=0.1$ s
 a) rozdzielczość

$$\Delta f = \frac{1}{NT_s} \approx 0.01 \text{ Hz}$$

b) rozdzielczość

$$\Delta f \approx 0.04 \text{ Hz}$$

po uzupełnieniu zerami

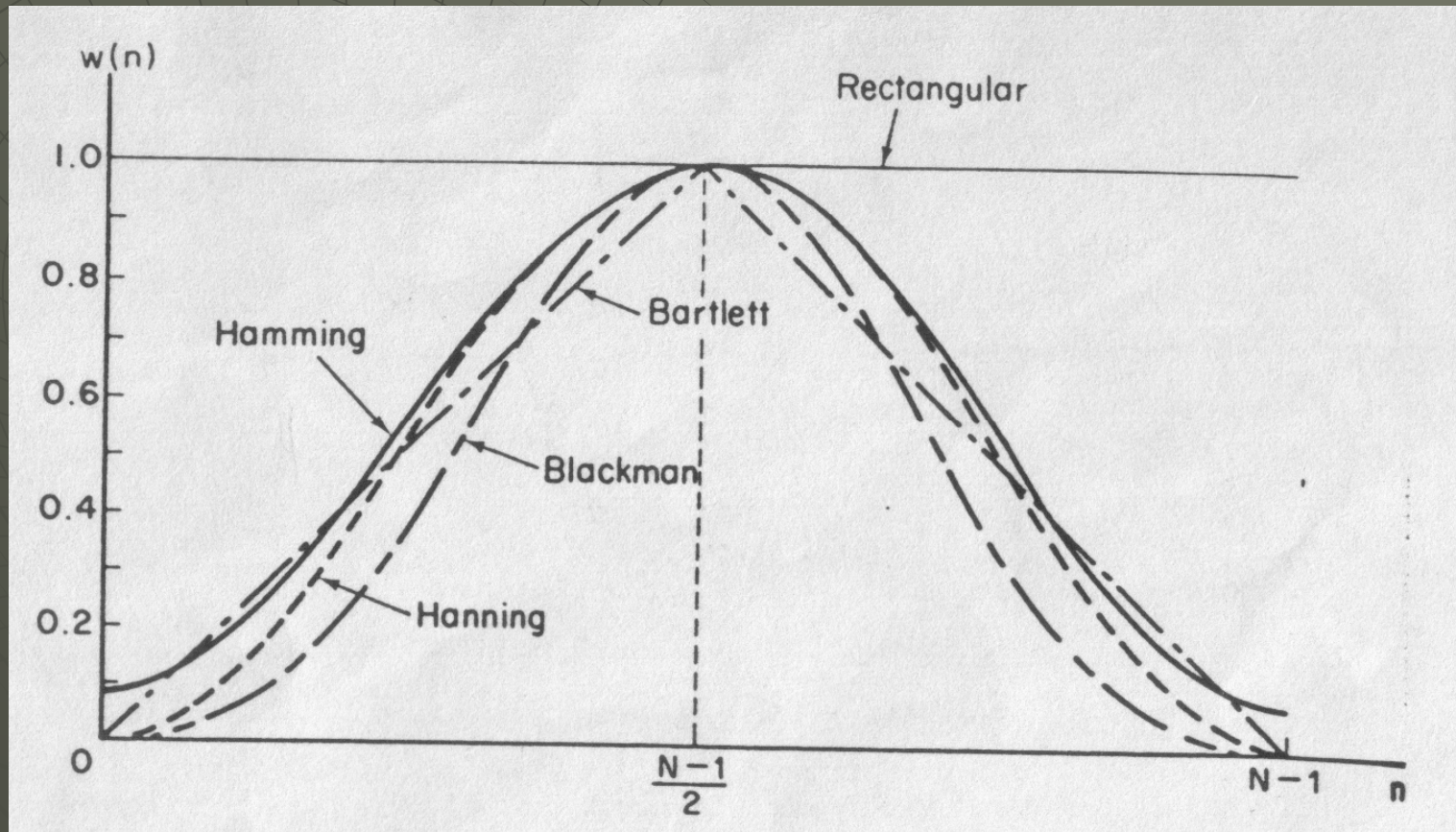
$$\Delta f' \approx 0.01 \text{ Hz}$$

$$x_a(t) = \sin 2\pi f_1 t + \sin 2\pi f_2 t + \sin 2\pi f_3 t, \quad f_1 = 1, f_2 = 1.02, f_3 = 1.07 \text{ Hz}$$

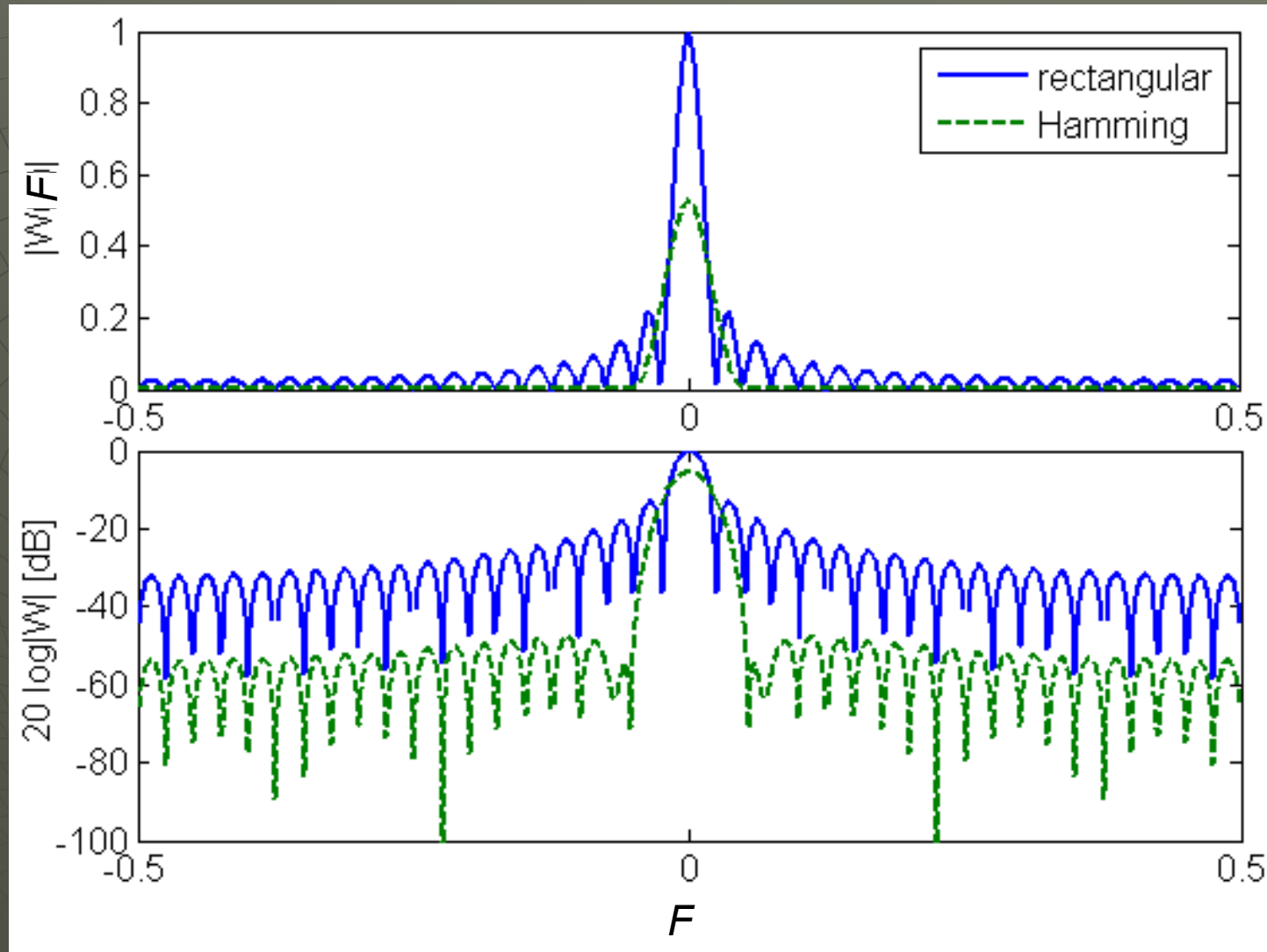
Okna czasowe w analizie widmowej

Ponieważ $x_0(n) = x(n)w(n)$, więc w dziedzinie częstotliwości:

$$X_0(e^{j\Omega}) = X(e^{j\Omega}) * W(e^{j\Omega}) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\theta}) W(e^{j(\Omega-\theta)}) d\theta$$

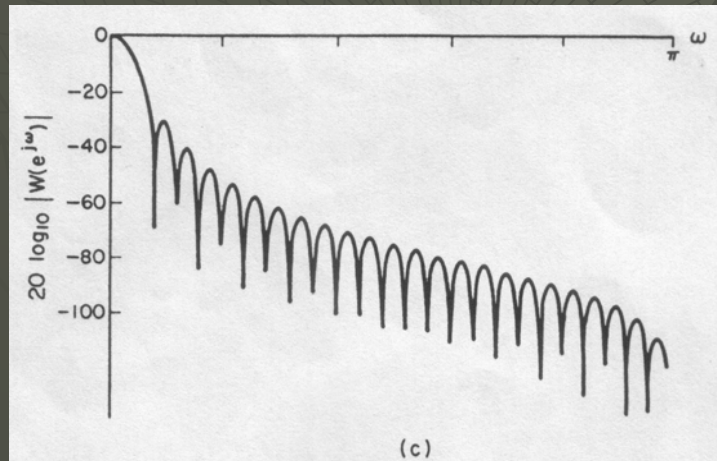
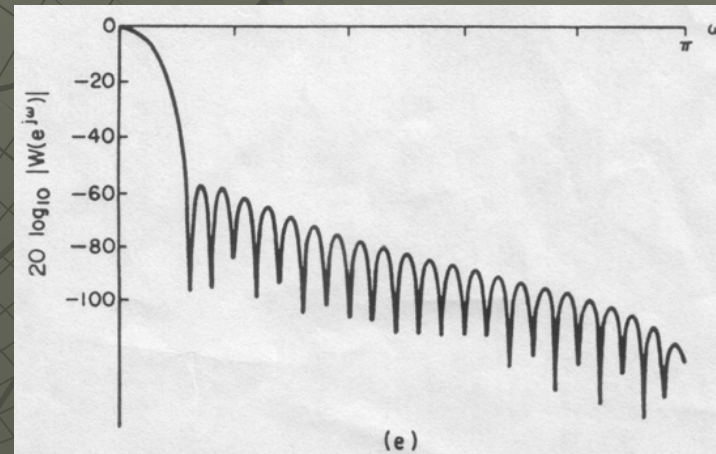
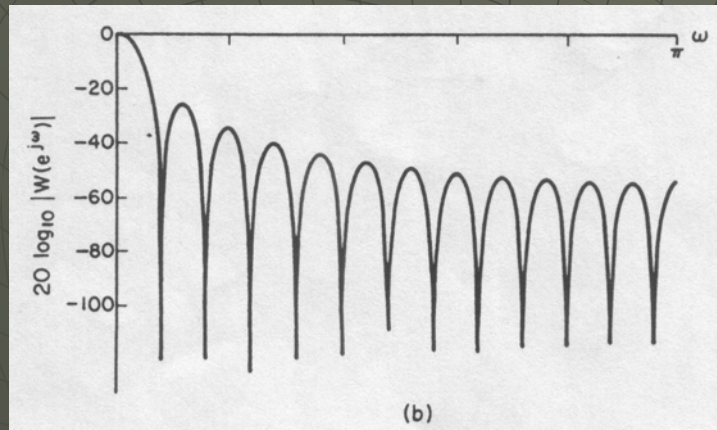
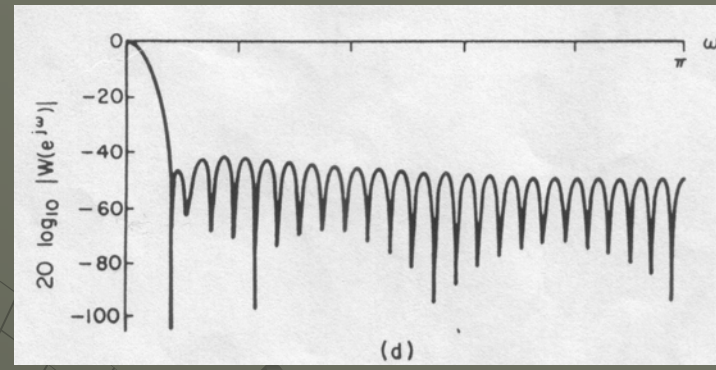
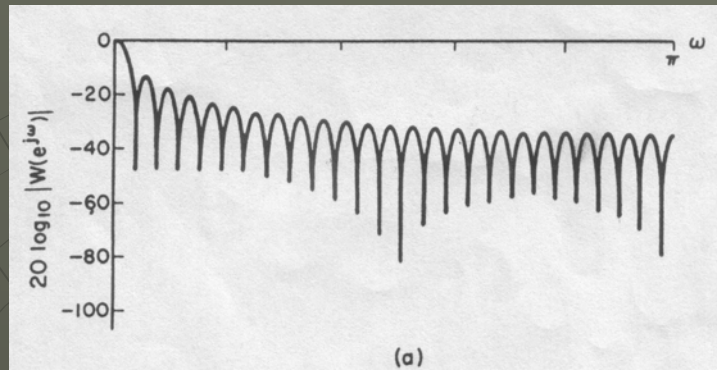


Rys. Przebiegi niektórych często stosowanych okien czasowych



Rys. Porównanie widm Fouriera okna prostokątnego i Hamminga, częstotliwość unormowana $F = f / f_s$

Okno	Równanie	Szerokość listka głównego	Maks. wys. listków bocznych [dB]
Prostokątne	$w(n)=1, 0 \leq n \leq N-1$	$2\pi/N$	-13
Bartletta (trójkątne)	$w(n) = \begin{cases} 2n/(N-1), & 0 \leq n \leq (N-1)/2 \\ 2-2n/(N-1), & (N-1)/2 \leq n \leq N-1 \end{cases}$	$4\pi/N$	-25
Hanninga	$w(n) = 0.5[1 - \cos(2\pi n/(N-1))], 0 \leq n \leq N-1$	$4\pi/N$	-31
Hamminga	$w(n) = 0.54 - 0.46 \cos(2\pi n/(N-1)), 0 \leq n \leq N-1$	$4\pi/N$	-41
Blackmana	$w(n) = 0.42 - 0.5 \cos(2\pi n/(N-1)) + 0.08 \cos(4\pi n/(N-1)), 0 \leq n \leq N-1$	$6\pi/N$	-57
Kaisera	$w(n) = \frac{I_0[\beta \sqrt{[(N-1)/2]^2 - [n - (N-1)/2]^2}]}{I_0[\beta(N-1)/2]}$ <p>$I_0(\cdot)$ – zmodyfikowana f. Bessela I rodzaju 0 rzędu, typowo $4 < \beta(N-1)/2 < 10, 0 \leq n \leq N-1$</p>		-46 do -82



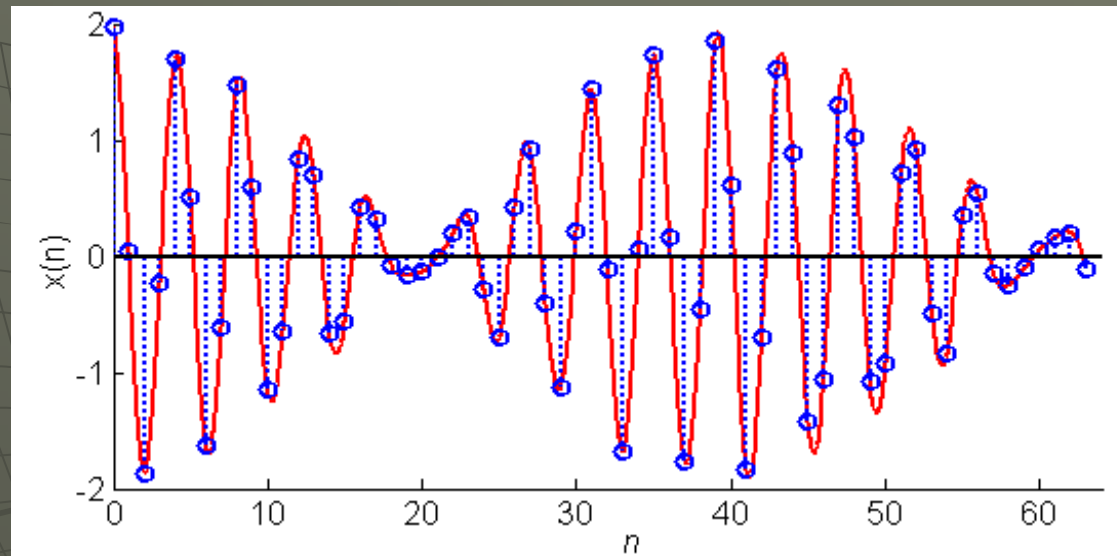
Rys. Widma Fouriera okien czasowych (skala logarytmiczna):
a) prostokątne, b) Bartletta (trójkątne), c) Hanninga,
d) Hamminga, e) Blackmana

Wpływ nałożenia okna czasowego na zniekształcenie widma

Szerokość listka głównego widma okna wpływa na rozróżnialność częstotliwościową *DFT* (jeżeli różnica częstotliwości dwóch składowych jest mniejsza od szerokości listka głównego, to odpowiadające im prążki zleją się w jeden wskutek rozmycia widma.

Wysokość listków bocznych widma okna wpływa na rozróżnialność *amplitudową* *DFT* (jeżeli w sygnale występuje składowa o amplitudzie porównywalnej z amplitudą listków bocznych, to „utonie” ona w pofalowaniach widma.

W celu poprawienia rozróżnialności amplitudy stosuje się okna o kształcie innym niż prostokątny. Okna takie mają niższe listki boczne kosztem poszerzenia listka głównego, który zawiera więcej energii w porównaniu z oknem prostokątnym.

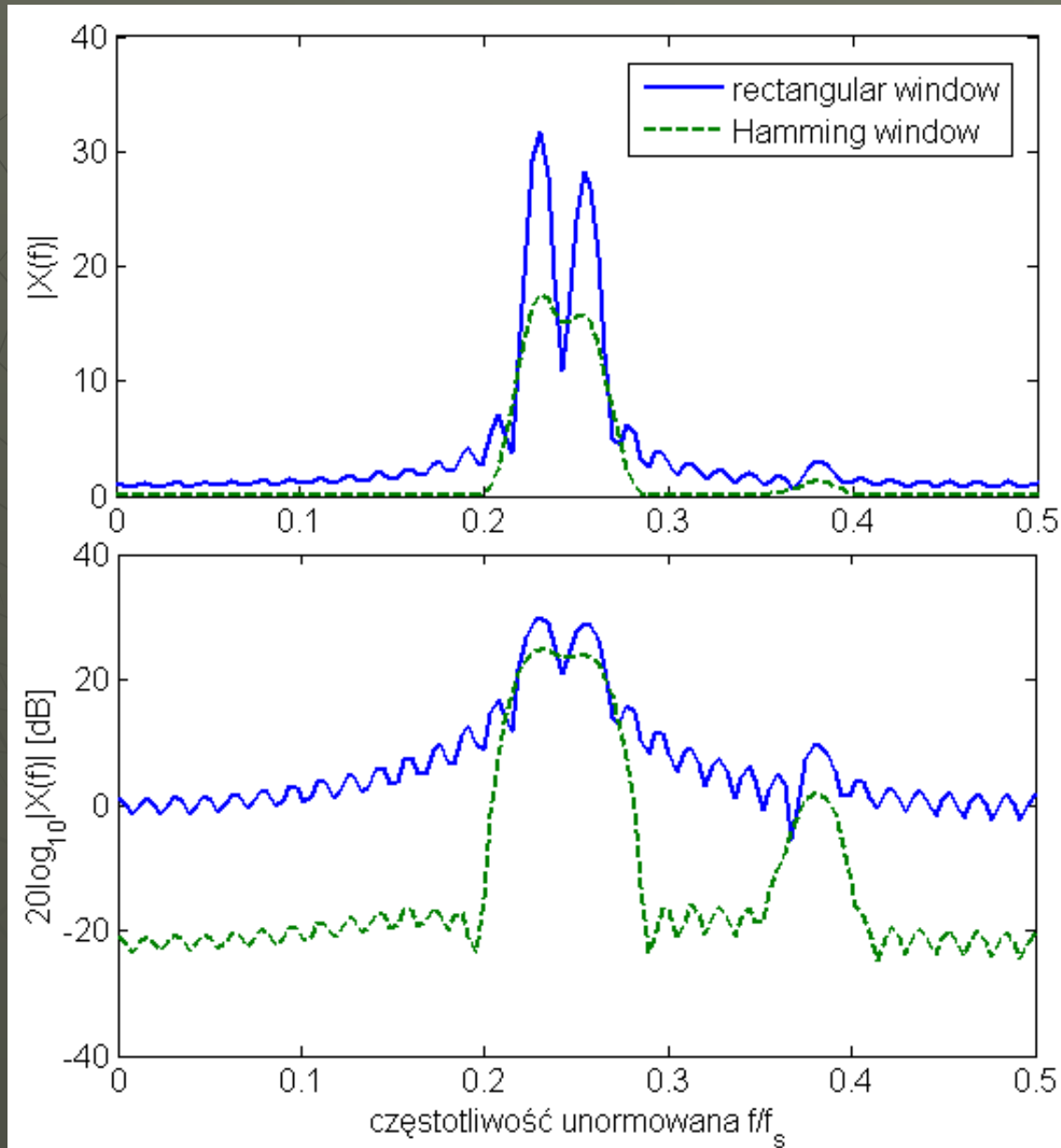


Rys. Sygnał dyskretny złożony z 3 składowych harmoniczných (N=64 próbki). Częstotliwości unormowane względem częstotliwości próbkowania $f_n = f / f_s$

$$x(n) = A_1 \cos 2\pi f_{n1} n + A_2 \cos 2\pi f_{n2} n + A_3 \cos 2\pi f_{n3} n,$$

$$f_{n1} = 0.23, \quad f_{n2} = 0.255, \quad f_{n3} = 0.38$$

$$A_1 = 1.0, \quad A_2 = 0.9, \quad A_3 = 0.03$$



Rys. 256-punktowe $DFT[x(n)]$ (uzupełnione zerami) dla okna prostokątnego i okna Hamminga. Skala liniowa i logarytmiczna (dB). Skala OX unormowana $F=f/f_s$ (częstotliwość Nyquista $F_N=0.5$)

Związek DFT z szeregiem Fouriera

Jeżeli dyskretny sygnał okresowy $x_p(n)$ ma skończone rozwinięcie w zespolony szereg Fouriera

$$x_p(n) = \sum_{m=0}^{M-1} \bar{c}_m e^{jn\Omega_m}$$

to współczynniki szeregu są związane ze współczynnikami N -punktowej $DFT[x_p(n)]$ zależnością:

$$c_m = \bar{c}_m = \frac{X_p(m)}{N}$$

o ile tylko $N \geq M$.

Jeżeli rozwinięcie jest nieskończone ($M = \infty$) lub $N < M$, to we współczynnikach c_m wyznaczonych na podstawie DFT występuje błąd nałożenia (*aliasing*), bo:

$$c_m = \sum_{k=-\infty}^{\infty} \bar{c}_{m+kN}$$

gdzie \bar{c}_m są prawdziwymi współczynnikami rozwinięcia w szereg.

Odtworzenie *DTFT* sygnału skończonego lub okresowego na podstawie *DFT*

Jest to rekonstrukcja widma ciągłego z próbek *DFT*:

$$X_0(e^{j\Omega}) = \sum_{m=0}^{M-1} \frac{X_0(m)}{M} \cdot \frac{e^{-j\frac{1}{2}\Omega(M-1)} \sin \frac{\Omega}{2} M}{e^{-j\frac{1}{2}\Omega_m} \sin \frac{\Omega - \Omega_m}{2}}$$

Nakład obliczeniowy DFT

$$X(m) = X(\Omega_m) = \sum_{k=0}^{M-1} x(k) w_M^{mk}, \quad m = 0, 1, \dots, M-1$$

gdzie $w_M = e^{-j(2\pi/M)}$ - zespolony pierwiastek M -tego stopnia z 1.

$$w_M^{(m+1M)k} = w_M^{mk}$$

$$X(m) = \sum_{k=0}^{M-1} \left[\left(\operatorname{Re} x(k) \operatorname{Re} w_M^{mk} - \operatorname{Im} x(k) \operatorname{Im} w_M^{mk} \right) + j \left(\operatorname{Re} x(k) \operatorname{Im} w_M^{mk} + \operatorname{Im} x(k) \operatorname{Re} w_M^{mk} \right) \right]$$

Klasyczne <i>DFT</i>	m	k	Liczba operacji
Mnożenia	M	$4M$	$4M^2$ (M^2 zespolonych)

Idea szybkiego przekształcenia FFT (Fast Fourier Transform)

Podział ciągu M-punktowego na 2 ciągi (M/2)-punktowe (*decimation-in-time*) daje oszczędność obliczeń:

$$M^2 \rightarrow 2(M/2)^2 = M^2/2 \text{ mnożeń}$$

Po podziale na 4 ciągi (M/4)-punktowe:

$$(M/2)^2 \rightarrow 2(M/4)^2 = M^2/8 \text{ mnożeń itd.}$$

Podział na 2 ciągi: próbek parzystych i nieparzystych

$$g(k) = x(2k)$$

$$h(k) = x(2k+1), \quad k = 0, 1, \dots, M/2 - 1$$

$$X(m) = \sum_{k=0}^{M/2-1} x(2k)w_M^{2km} + \sum_{k=0}^{M/2-1} x(2k+1)w_M^{(2k+1)m}$$

$$X(m) = \sum_{k=0}^{M/2-1} g(k)w_{M/2}^{km} + w_M^m \sum_{k=0}^{M/2-1} h(k)w_{M/2}^{km}$$

ponieważ $w_M^2 = w_{M/2}$

$$X(m) = G(m) + w_M^m H(m)$$

(suma $M/2$ -punktowych DFT z próbek $x(k)$ o parzystych i nieparzystych indeksach)

Ponieważ:

$$w_M^{m+M/2} = -w_M^m, \quad G(m + M/2) = G(m), \quad H(m + M/2) = H(m)$$

Można napisać:

$$X(m) = \begin{cases} G(m) + w_M^m H(m) & 0 \leq m \leq M/2 - 1 \\ G(m - \frac{M}{2}) - w_M^m H(m - \frac{M}{2}) & M/2 \leq m \leq M - 1 \end{cases}$$

Jeśli $M=2^n$, to proces dekompozycji można kontynuować aż do 2-punktowych DFT ($\log_2 M$ dekompozycji). Na każdym etapie otrzymuje się $M/2$ tzw. *schematów motylkowych*. W każdym z nich wykonywane jest jedno mnożenie zespolone.

Nakład obliczeniowy M -punktowej FFT:

$$(M/2) \log_2 M \quad \text{mnożeń zespolonych}$$

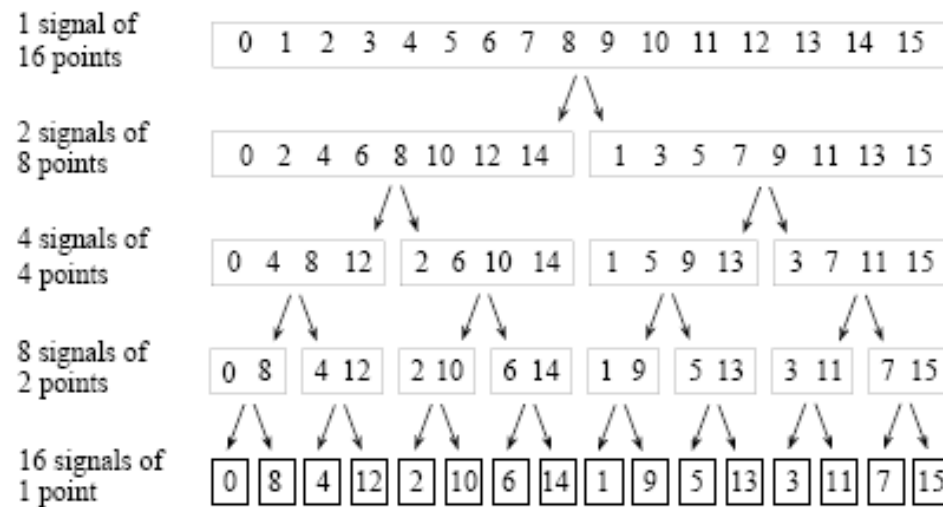


FIGURE 12-2
The FFT decomposition. An N point signal is decomposed into N signals each containing 1 point. Each stage uses an *interlace decomposition*, separating the even and odd numbered samples.

Sample numbers in normal order		Sample numbers after bit reversal	
Decimal	Binary	Decimal	Binary
0	0000	0	0000
1	0001	8	1000
2	0010	4	0100
3	0011	12	1100
4	0100	2	0010
5	0101	10	1010
6	0110	6	0100
7	0111	14	1110
8	1000	1	0001
9	1001	9	1001
10	1010	5	0101
11	1011	13	1101
12	1100	3	0011
13	1101	11	1011
14	1110	7	0111
15	1111	15	1111

FIGURE 12-3
The FFT bit reversal sorting. The FFT time domain decomposition can be implemented by sorting the samples according to bit reversed order.

FIGURE 12-5
FFT synthesis flow diagram. This shows the method of combining two 4 point frequency spectra into a single 8 point frequency spectrum. The $\times S$ operation means that the signal is multiplied by a sinusoid with an appropriately selected frequency.

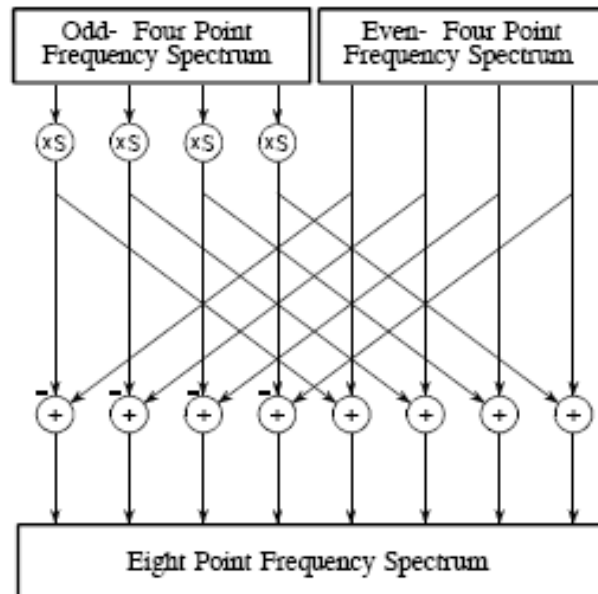
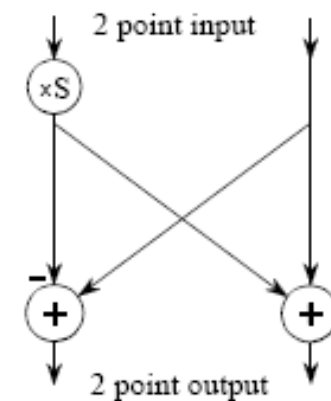


FIGURE 12-6
The FFT butterfly. This is the basic calculation element in the FFT, taking two complex points and converting them into two other complex points.





4. Dyskretne układy liniowe

Transformata Z

Odpowiednikiem transformaty Laplace'a sygnałów ciągłych $x_a(t)$ dla sygnałów $x(n)$ dyskretnych w czasie jest transformata Z.

Dla sygnałów przyczynowych ($x(n)=0$ dla wszystkich $n<0$) oblicza się transformatę *jednostronną*:

$$Z[x(n)] = X(z) \triangleq \sum_{n=0}^{\infty} x(n)z^{-n} = x(0) + x(1)z^{-1} + x(2)z^{-2} + \dots$$

Transformata jest określona w obszarze zbieżności nie zawierającym biegunów $X(z)$.

Odwrotna transformata Z (C jest konturem obejmującym początek układu i nie zawierającym biegunów $X(z)$):

$$Z^{-1}[X(z)] = x(n) \triangleq \frac{1}{2\pi j} \oint_C X(z)z^{n-1} dz$$

Związek transformaty Z funkcji dyskretnej $x(n)=x(nT_s)$ z transformatą Laplace'a funkcji ciągłej $x_a(t)$ próbkowanej impulsowo

$$x^*(t) = \sum_{n=0}^{\infty} x_a(t)\delta(t - nT_s)$$

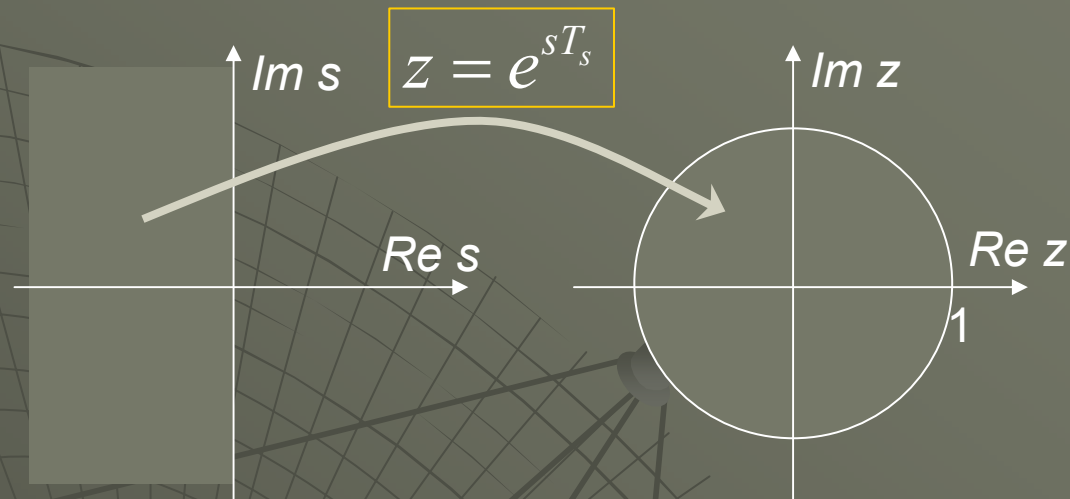
określa zależność:

$$X(z) \Big|_{z=e^{sT_s}} = X^*(s)$$

gdzie: $X(z) = Z[x(n)]$

$$X^*(s) = L[x^*(t)] = \int_0^{\infty} \sum_{n=0}^{\infty} x_a(t)\delta(t - nT_s) \cdot e^{-st} dt =$$

$$\sum_{n=0}^{\infty} \int_0^{\infty} x_a(t)\delta(t - nT_s) e^{-st} dt = \sum_{n=0}^{\infty} x(nT_s) e^{-snT_s}$$



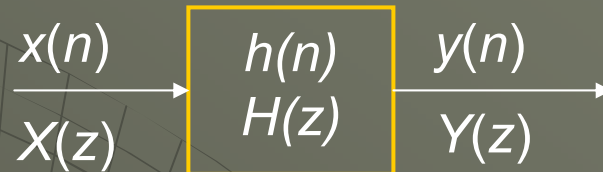
Rys. Przekształcenie lewej półpłaszczyzny zmiennej zespolonej s (zanikające sygnały ciągłe) na wnętrze okręgu jednostkowego na płaszczyźnie zmiennej zespolonej z (zanikające sygnały dyskretne)

Właściwości przekształcenia Z

	Funkcja dyskretna	Transformata
Definicja transformaty jednostronnej ($x(n)=0$ dla $n<0$)	$x(n)$	$X(z) = \sum_{n=0}^{\infty} x(n)z^{-n}$
Liniowość	$a_1 x_1(n) + a_2 x_2(n)$	$a_1 X_1(z) + a_2 X_2(z)$
Zmiana skali częstotliwości zespolonej	$e^{-an} x(n)$	$X(ze^a)$
Przesunięcie w czasie	$x(n-k)$ $x(n+1)$	$z^{-k} \left[X(z) + \sum_{j=1}^k x(-j)z^j \right]$ $zX(z) - x(0)$
Różniczkowanie transformaty	$n \cdot x(n)$	$-z \frac{dX(z)}{dz}$
Transformata splotu	$x_1(n) * x_2(n) = \sum_{k=0}^n x_1(k) \cdot x_2(n-k)$	$X_1(z) \cdot X_2(z)$
Transformata funkcji okresowej	$x(n) = \sum_{k=0}^{\infty} x_M(n+kM)$	$\frac{z}{z-1} X_M(z)$
Wartość początkowa		$x(0) = \lim_{z \rightarrow \infty} X(z)$
Wartość końcowa		$x(\infty) = \lim_{z \rightarrow 1} (z-1)X(z)$

Transformaty Z podstawowych funkcji

Funkcja dyskretna $x(n), n \geq 0$	Transformata $X(z)$
$a\delta(n)$	a
$a \cdot \mathbf{1}(n)$	$\frac{az}{z-1}$
$a \cdot n$	$\frac{az}{(z-1)^2}$
a^n	$\frac{z}{z-a}$
na^n	$\frac{az}{(z-a)^2}$
$\sin(\Omega n)$	$\frac{z \sin \Omega}{z^2 - 2z \cos \Omega + 1}$
$\cos(\Omega n)$	$\frac{z(z - \cos \Omega)}{z^2 - 2z \cos \Omega + 1}$
$a^n \cos(\Omega n)$	$\frac{z(z - a \cos \Omega)}{z^2 - 2az \cos \Omega + a^2}$

Stacjonarny układ liniowy LTI (Linear Time Invariant)

Układ LTI w pełni charakteryzuje jego odpowiedź impulsowa $h(n)$. Jest to odpowiedź $y(n)$ na impuls $x(n)=\delta(n)$ ($\delta(n)=1$ dla $n=0$ i $=0$ dla $n\neq 0$). Dla dowolnego wymuszenia $x(n)$ wyjście jest *splotem*:

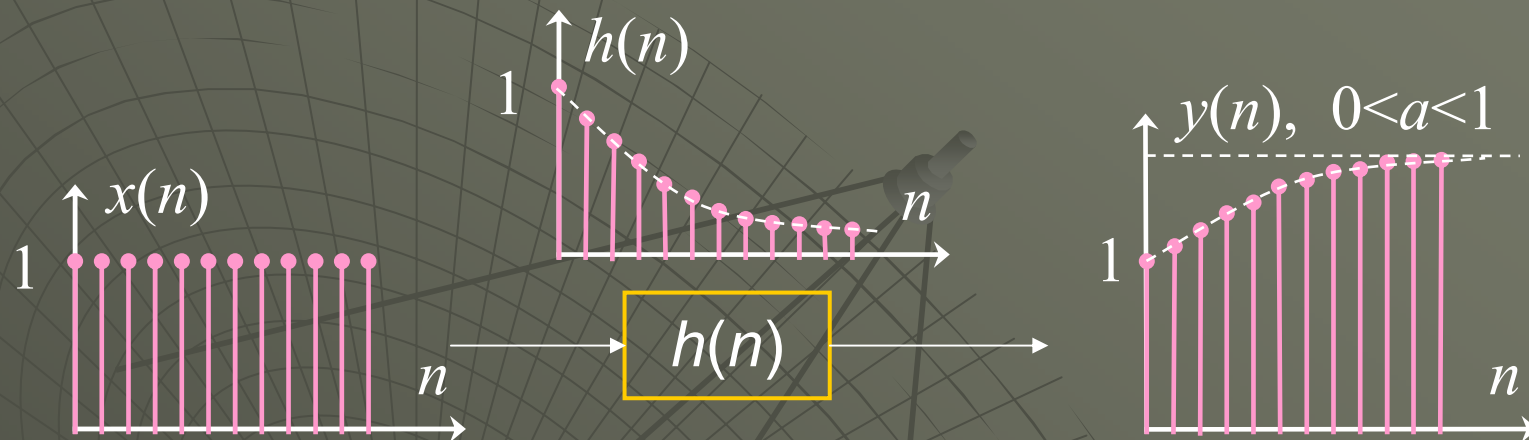
$$y(n) = h(n) * x(n) = \sum_{k=0}^{\infty} h(k)x(n-k) = \sum_{k=0}^{\infty} h(n-k)x(k)$$

Układ LTI jest stabilny w sensie BIBO (*bounded input – bounded output*), jeżeli jego odpowiedź impulsowa spełnia warunek bezwzględnej sumowalności:

$$\sum_{k=-\infty}^{\infty} |h(k)| < \infty$$

Jeżeli $h(n)\neq 0$ dla pewnego $n < 0$, to układ nazywa się **układem nieprzyczynowym** (odpowiedź wyprzedza wymuszenie!).

Przykład: $h(n)=a^n$, $|a|<1$, wymuszenie $x(n)=\mathbf{1}(n)$



$$y(n) = h(n) * x(n) = \sum_{k=0}^{\infty} h(k) \cdot \mathbf{1}(n-k) = \sum_{k=0}^n h(k) = \sum_{k=0}^n a^k,$$

$$y(0) = 1, \quad y(1) = 1 + a, \quad y(2) = 1 + a + a^2, \dots, \lim_{n \rightarrow \infty} y(n) = \frac{1}{1-a}$$

Dalej będziemy zajmować się układami LTI opisanymi *liniowymi równaniami różnicowymi o stałych współczynnikach*:

$$a_0 y(n) + a_1 y(n-1) + \dots + a_{N_a} y(n-N_a) = b_0 x(n) + b_1 x(n-1) + \dots + b_{N_b} x(n-N_b)$$

$$\sum_{i=0}^{N_a} a_i y(n-i) = \sum_{i=0}^{N_b} b_i x(n-i)$$

Często przyjmuje się $N_a = N_b = N$, gdzie $N = \max(N_a, N_b)$ (rzęd układu), oraz $a_0 = 1$. Wtedy równanie różnicowe w formie rekurencyjnej:

$$y(n) = -a_1 y(n-1) - \dots - a_N y(n-N) + b_0 x(n) + b_1 x(n-1) + \dots + b_N x(n-N)$$

Dokonując przekształcenia Z równania różnicowego wyraz po wyrazie dla zerowych warunków początkowych

$$y(-1) = y(-2) = \dots = y(-N) = 0, \quad x(-1) = \dots = x(-N) = 0$$

otrzymujemy:

$$a_0 Y(z) + a_1 z^{-1} Y(z) + \dots + a_N z^{-N} Y(z) = b_0 X(z) + b_1 z^{-1} X(z) + \dots + b_N z^{-N} X(z)$$

$$\text{skąd } Y(z)(a_0 + a_1 z^{-1} + \dots + a_N z^{-N}) = X(z)(b_0 + b_1 z^{-1} + \dots + b_N z^{-N})$$

Transmitancję (funkcję przejścia) układu dyskretnego definiuje się jako:

$$H(z) = \left. \frac{Y(z)}{X(z)} \right|_{WP=0} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_N z^{-N}} = \frac{B(z)}{A(z)}$$

Dla sygnału wejściowego o transformacie $X(z)$ transformata wyjścia układu opisanego transmitancją $H(z)$ jest równa:

$$Y(z) = H(z) \cdot X(z)$$

Transmitancja układu jest transformatą Z jego odpowiedzi impulsowej $h(n)$:

$$H(z) = Z[h(n)]$$

Przykład (poprzedni): W dziedzinie transformat Z:

$$H(z) = Z[a^n] = \frac{z}{z-a}, \quad X(z) = Z[1(n)] = \frac{z}{z-1}$$

$$Y(z) = H(z) \cdot X(z) = \frac{z^2}{(z-1)(z-a)} = \frac{1}{1-a} \left(\frac{z}{z-1} - \frac{az}{z-a} \right) \rightarrow y(n) = \frac{1-a^{n+1}}{1-a}, n \geq 0$$

Po pomnożeniu licznika i mianownika przez z^N i rozłożeniu na czynniki $H(z)$ można przedstawić w formie:

$$H(z) = \frac{b_0(z - z_1)(z - z_2) \cdots (z - z_N)}{a_0(z - p_1)(z - p_2) \cdots (z - p_N)}$$

z_1, z_2, \dots, z_N - zera transmitancji

p_1, p_2, \dots, p_N - bieguny transmitancji

Układ LTI jest *stabilny* (jego odpowiedź impulsowa zanika do zera) wtedy i tylko wtedy, kiedy *wszystkie bieguny jego transmitancji leżą wewnątrz okręgu jednostkowego*, tj.:

$$|p_i| < 1 \quad \text{dla } i = 1, 2, \dots, N$$

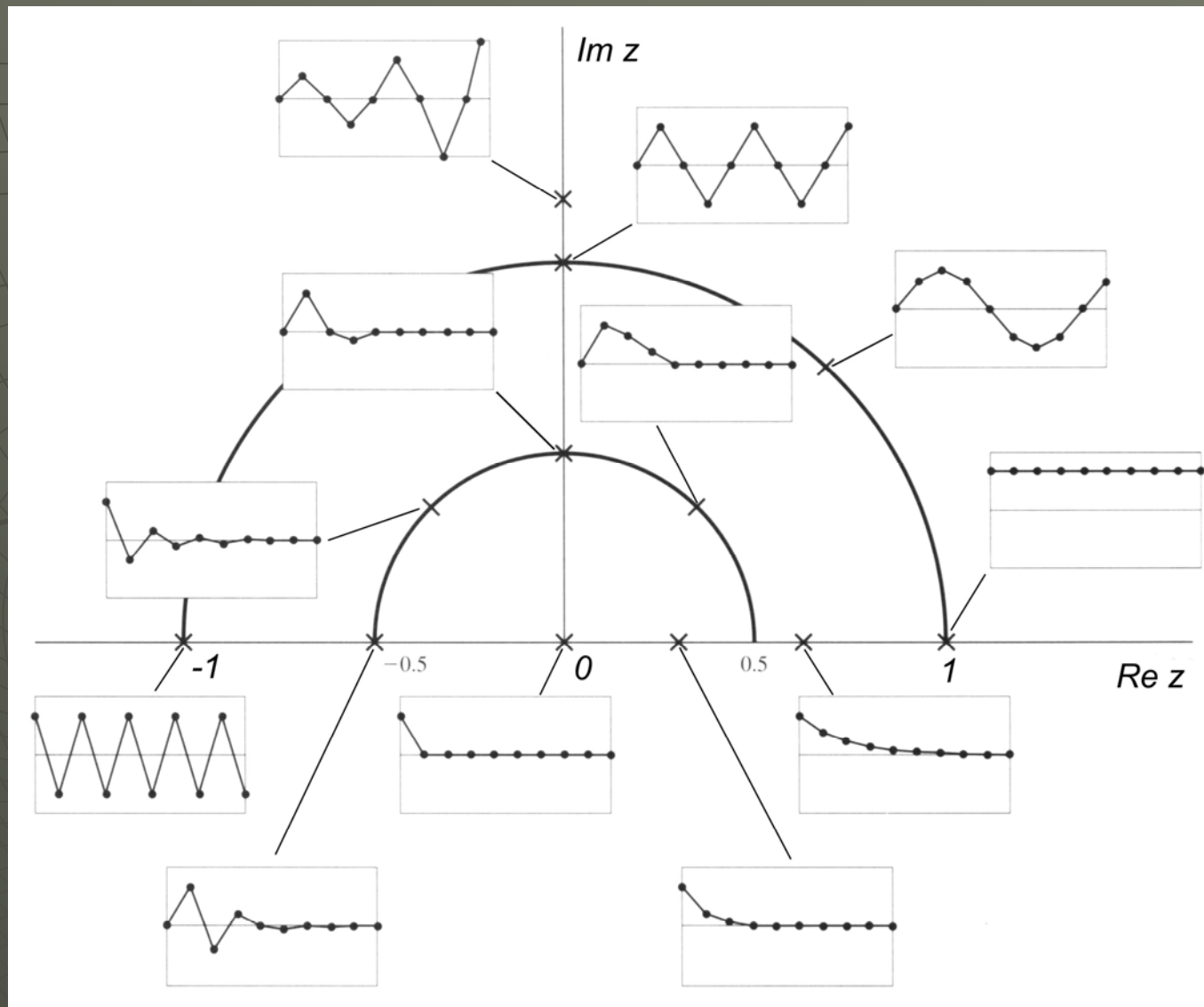
Z jednokrotnym biegunem rzeczywistym $p_i = a$ związana jest składowa odpowiedzi impulsowej postaci

$$\text{const} \cdot a^n,$$

natomiast z parą biegunów zespolonych sprzężonych

$p_{i,i+1} = |p_i| \cdot \exp(\pm j\Omega_i)$, składowa odpowiedzi impulsowej postaci

$$\text{const} \cdot |p_i|^n \cdot \cos(n\Omega_i + \varphi_0).$$



Rys. Mapa przebiegów czasowych związanych z położeniem bieguna transformaty Z na płaszczyźnie zespolonej.

Transmitancja w pełnej formie opisuje układ (filtr) o nieskończonej odpowiedzi impulsowej NOI (lub IIR – *Infinite Impulse Response*):

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_N z^{-N}}$$

Jeżeli $A(z)=1$ to układ nazywa się układem o skończonej odpowiedzi impulsowej SOI (lub FIR – *Finite Impulse Response*)

$$H(z) = B(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}$$

$$H(z) = \frac{b_0 z^N + b_1 z^{N-1} + b_2 z^{N-2} + \dots + b_0}{z^N}$$

Wszystkie bieguny układu SOI leżą w początku układu: $p_i=0$, $i=1,2,\dots,N$ (układ SOI jest zawsze stabilny).

Przykład: Wyznaczyć równania różnicowe oraz kilka początkowych wartości odpowiedzi impulsowych układów o transmitancjach:

$$H_1(z) = \frac{1 + 0.7z^{-1}}{1 - 0.5z^{-1}} \quad H_2(z) = 1 + 0.7z^{-1} - 0.5z^{-2}$$



5. Projektowanie filtrów cyfrowych

Cele filtracji

- 1) usunięcie z sygnału wejściowego zakłóceń i wyodrębnienie informacji użytecznej,
- 2) celowe przekształcenie danego sygnału w inny (np. filtry różniczkujące, filtry Hilberta lub banki filtrów do dekompozycji sygnału).

Etapy projektowania

- 1) specyfikacja wymaganych właściwości (np. częstotliwościowych),
- 2) aproksymacja specyfikacji określoną transmitancją (typ, rząd, parametry filtra),
- 3) implementacja algorytmu filtra w określonej arytmetyce i/lub sprzęcie

Ogólna postać transmitancji filtra

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + \dots + b_N z^{-N_b}}{a_0 + a_1 z^{-1} + \dots + a_N z^{-N_a}}, \quad a_0 = 1, \quad N_a = N_b = N$$

Równanie różnicowe odpowiadające $H(z)$ (N – rząd filtra)

$$y(n) + a_1 y(n-1) + \dots + a_N y(n-N) = b_0 x(n) + b_1 x(n-1) + \dots + b_N x(n-N)$$

$$y(n) = -a_1 y(n-1) - \dots - a_N y(n-N) + b_0 x(n) + b_1 x(n-1) + \dots + b_N x(n-N)$$

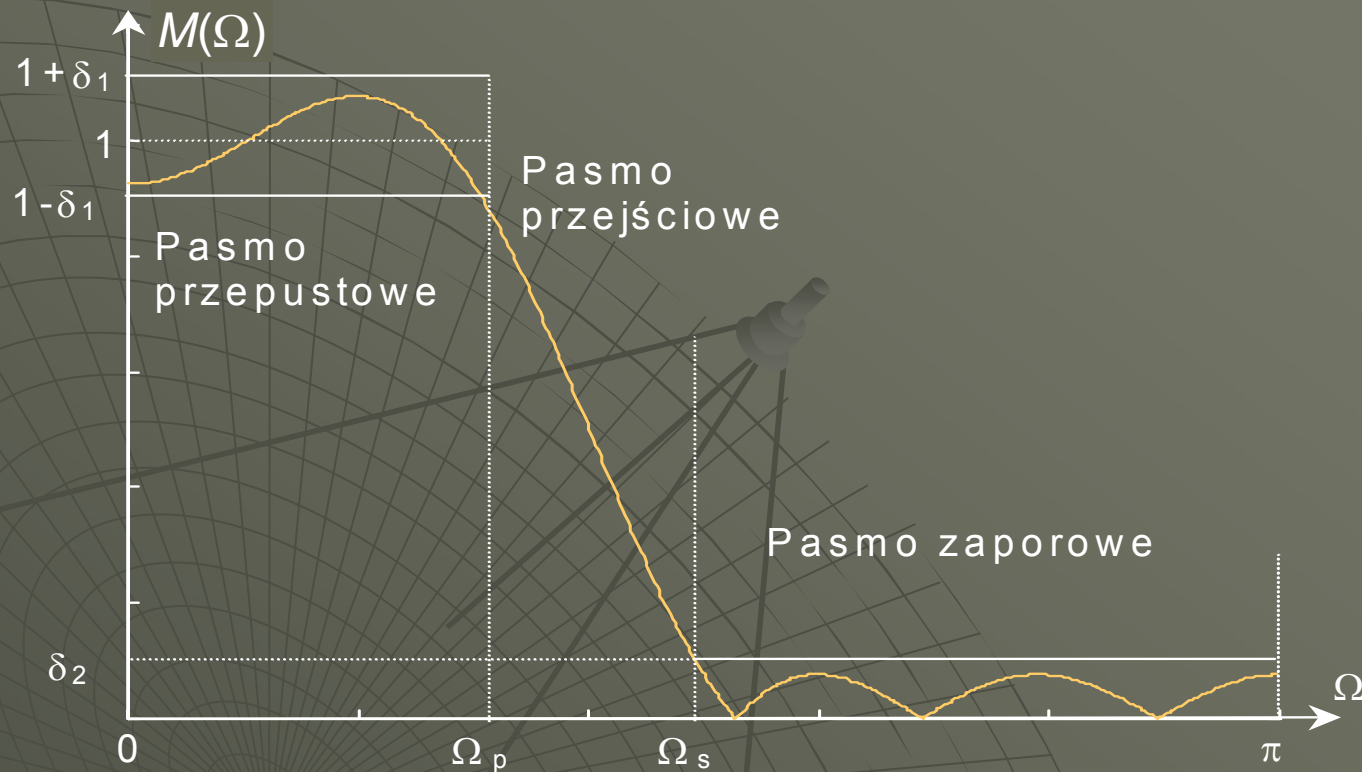
Charakterystyki częstotliwościowe: amplitudowa (*magnitude*)
i fazowa

$$M(\Omega) = |H(z)|_{z=e^{j\Omega}}$$

$$\varphi(\Omega) = \arg H(z) |_{z=e^{j\Omega}}$$

gdzie: $\Omega = \omega T_s = 2\pi f / f_s$

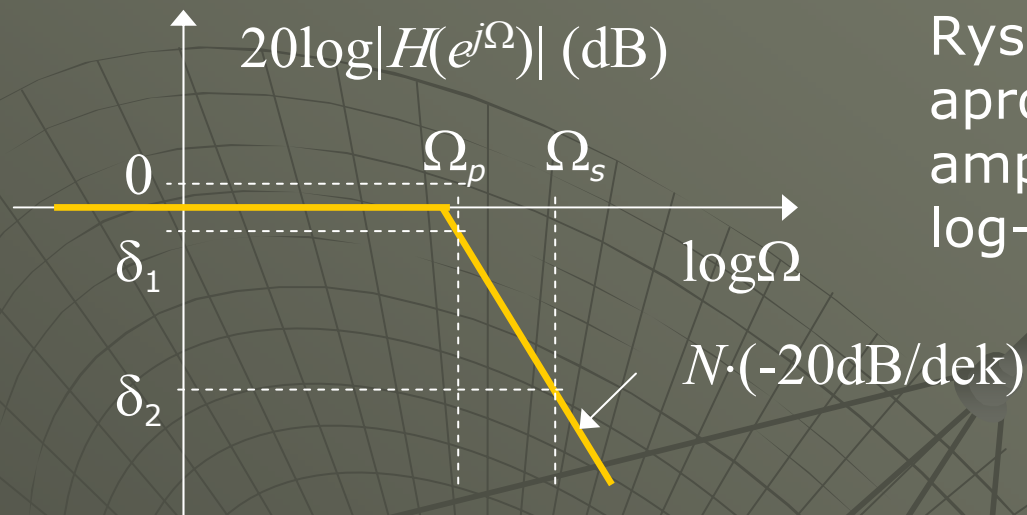
Specyfikacje w dziedzinie częstotliwości



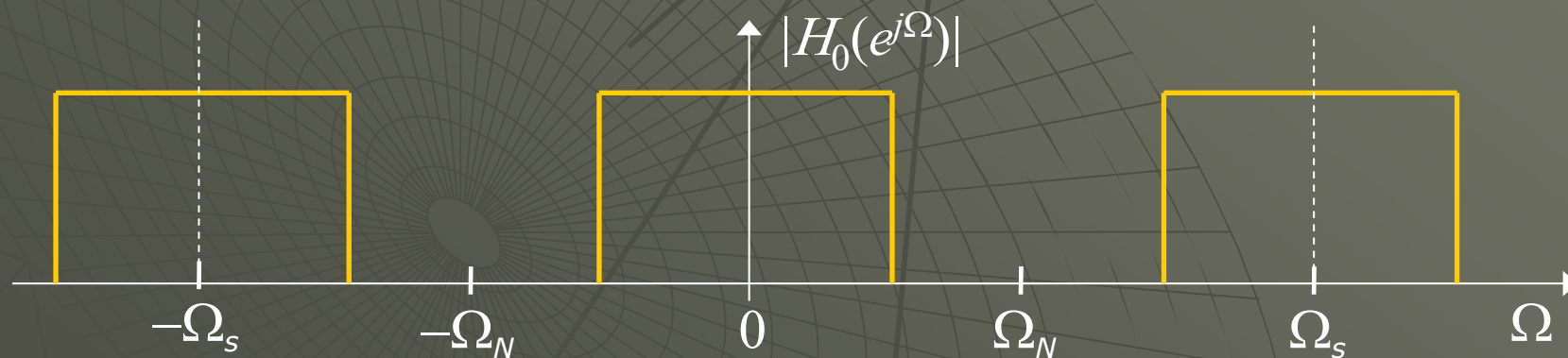
Rys. Tolerancje specyfikacji i ch-ka amplitudowa filtra LP. Skala liniowa

$1 - \delta_1 \leq M(\Omega) \leq 1 + \delta_1$ - dopuszczalne odchylenie od $M(\Omega)=1$ w paśmie przepustowym (*passband rippling* - r_p , dB)

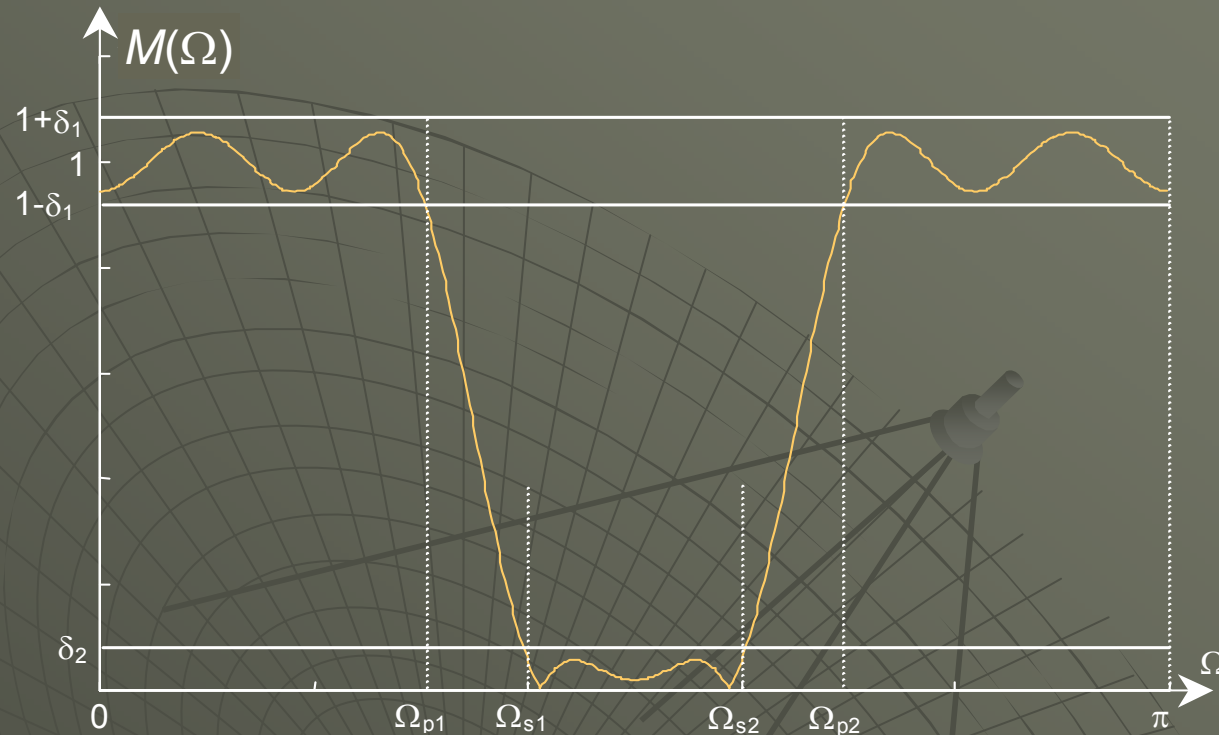
$M(\Omega) \leq \delta_2$ - minimalne tłumienie w paśmie zaporowym (*stopband rippling* - r_s , dB)



Rys. Asymptotyczna aproksymacja charakterystyki amplitudowej filtra LP w skali log-log, N – rząd filtra



Rys. Powielenie charakterystyki filtra dyskretnego LP w dziedzinie częstotliwości



Rys. Tolerancje specyfikacji i ch-ka amplitudowa filtra BS (pasmowozaporowego)

Charakterystyka *opóźnienia grupowego*:

$$\tau(\Omega) = -\frac{d\varphi(\Omega)}{d\Omega}$$

Stała wartość opóźnienia grupowego filtra oznacza liniowy przebieg fazy $\varphi(\Omega)$, co jest często wymagane (np. w przypadku przetwarzania sygnałów zmodulowanych fazowo), bo powoduje *zachowanie kształtu sygnału* filtrowanego w czasie.

Filtry minimalnofazowe

Czasem wymaga się, aby nie tylko bieguny, ale i zera transmitancji leżały wewnątrz okręgu jednostkowego. Filtry spełniające taki dodatkowy warunek nazywamy układami *minimalnofazowymi*. Charakteryzują się one wzajemną zależnością pomiędzy charakterystyką amplitudową i fazową. Zachodzi wtedy związek:

$$\hat{H}(e^{j\Omega}) = \ln H(e^{j\Omega}) = \ln M(\Omega) + j\varphi(\Omega)$$

Projektowanie ch-ki amplitudowej $M(\Omega)$ filtra minimalnofazowego implikuje wynikający z niej przebieg ch-ki fazowej $\varphi(\Omega)$. Jeżeli zrezygnuje się z warunku minimalnofazowości - wprowadzając do transmitancji zera $|z| > 1$ - można nakładać *niezależne* wymagania przebieg amplitudy i fazy. Ze względu na możliwość niezależnego kształtowania $\varphi(\Omega)$ filtry nieminimalnofazowe stosuje się jako *korektory fazy*.

Filtry wszechprzepustowe

Charakterystyka amplitudowa filtra wszechprzepustowego (*all-pass filter*) $M(\Omega)=1$ w całym zakresie częstotliwości. Wielomian licznika transmitancji jest wielomianem odwrotnym do wielomianu mianownika, tzn.:

$$A(z) = 1 + a_1 z^{-1} + \dots + a_{N-1} z^{-(N-1)} + a_N z^{-N}$$

$$B(z) = z^{-N} + a_1 z^{-(N-1)} + \dots + a_{N-1} z^{-1} + a_N = z^{-N} A(z^{-1})$$

$$H(z) = \frac{z^{-N} A(z^{-1})}{A(z)}$$

Każdemu (stabilnemu) biegunowi $z_1 = r_1 \exp(j\Omega_1)$ transmitancji leżącemu wewnątrz okręgu $|z|=1$ towarzyszy niestabilne zero $z_0 = 1/z_1$ leżące na zewnątrz okręgu („zwierciadlane odbicie” względem okręgu jednostkowego).

Charakterystykę opóźnienia grupowego $\tau(\Omega)$ takiego układu można odpowiednio ukształtować przez odpowiedni wybór wielomianu $A(z)$.

Projektowanie filtrów NOI (*Infinite Impulse Response – IIR*)

Zadanie projektowe: Wyznaczyć współczynniki wielomianów $A(z)$ i $B(z)$ transmitancji $H(z)$ tak, aby jej ch-ka widmowa była aproksymacją zadanej charakterystyki $H_0(e^{j\Omega})$.

1. Dyskretyzacja prototypów analogowych

Metoda polega na projektowaniu najpierw *filtra analogowego* o transmitancji ciągłej $H_a(s)$ spełniającego postawione wymagania (prototypu), a następnie przekształceniu $H_a(s)$ do docelowej postaci dyskretnej $H(z)$ (dyskretyzacja).

Zalety: proste, bezpośrednie formuły projektowe.

Wady: projektowanie filtrów o standardowych charakterystykach LP, HP, BS i BP - brak możliwości projektowania filtrów o dowolnym przebiegu charakterystyki $M(\Omega)$.

Jako prototypy analogowe wykorzystuje się kilka typów filtrów:

- **filtr Butterwortha** (o maksymalnie płaskiej charakterystyce $M_a(\Omega)$ w paśmie przepustowym),
- **filtr Czebyszewa I** rodzaju (o charakterystyce $M_a(\Omega)$ równomiernie pofalowanej w paśmie przepustowym i monotonicznej w paśmie zaporowym),
- **filtr Czebyszewa II** rodzaju (o charakterystyce równomiernie pofalowanej w paśmie zaporowym i monotonicznej w paśmie przepustowym),
- **filtr eliptyczny** (Cauera, o charakterystyce $M_a(\Omega)$ równomiernie pofalowanej w obu pasmach i najwęższym (najbardziej stromym dla określonego rzędu N) paśmie przejściowym,

Stosuje się też filtr Bessela, którego wersja analogowa ma prawie stałe opóźnienie grupowe $\tau(\Omega)$ w paśmie przepustowym i monotoniczną charakterystykę amplitudową.

Filtr eliptyczny jest najbardziej selektywny, ale ma najbardziej nieliniową charakterystykę fazową.

Nieliniowości ch-tyk fazowych filtrów są największe wokół pasm przejściowych ch-tyk amplitudowych.

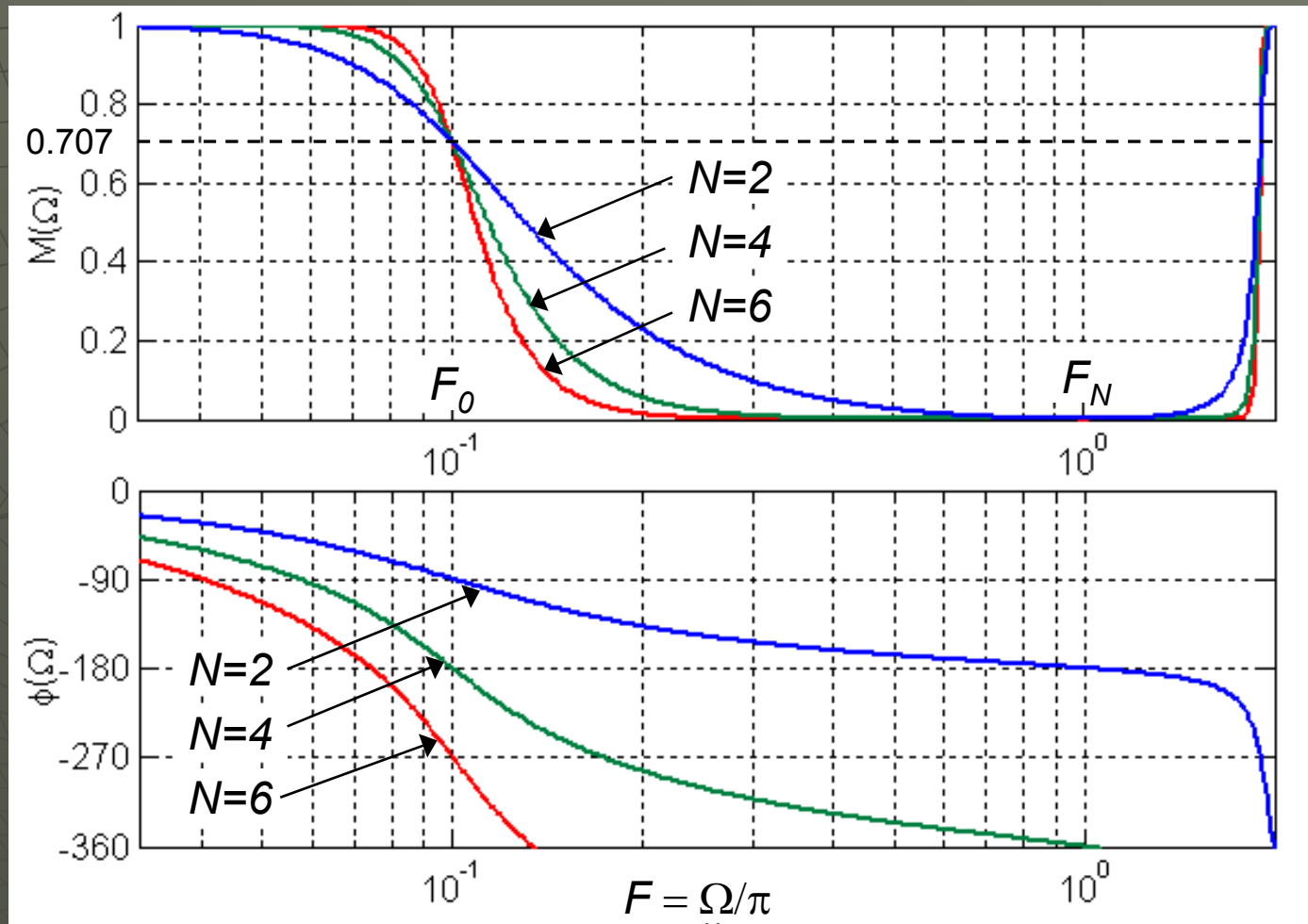
Kwadrat ch-ki amplitudowej prototypów:

$$M_a^2(\omega_a) = H_a(j\omega_a)H_a^*(j\omega_a) = \frac{1}{1 + \varepsilon^2 F_N^2\left(\frac{\omega_a}{\omega_{a0}}\right)}$$

Dla filtra Butterwortha: $F_N\left(\frac{\omega_a}{\omega_{a0}}\right) = \left(\frac{\omega_a}{\omega_{a0}}\right)^N$ $\varepsilon = 1$

Dla filtra Czebyszewa I: $F_N\left(\frac{\omega_a}{\omega_{a0}}\right) = V_N\left(\frac{\omega_a}{\omega_{a0}}\right)$ - wielomian Czebyszewa rzędu N
 $\varepsilon = \delta_1$ ($\varepsilon = \delta_2$ dla f. Czebyszewa II)

Dla eliptycznego: $F_N\left(\frac{\omega_a}{\omega_{a0}}\right) = U_N\left(\frac{\omega_a}{\omega_{a0}}\right)$ - funkcja eliptyczna Jacobiego
 $\varepsilon = \delta_1 = \delta_2$

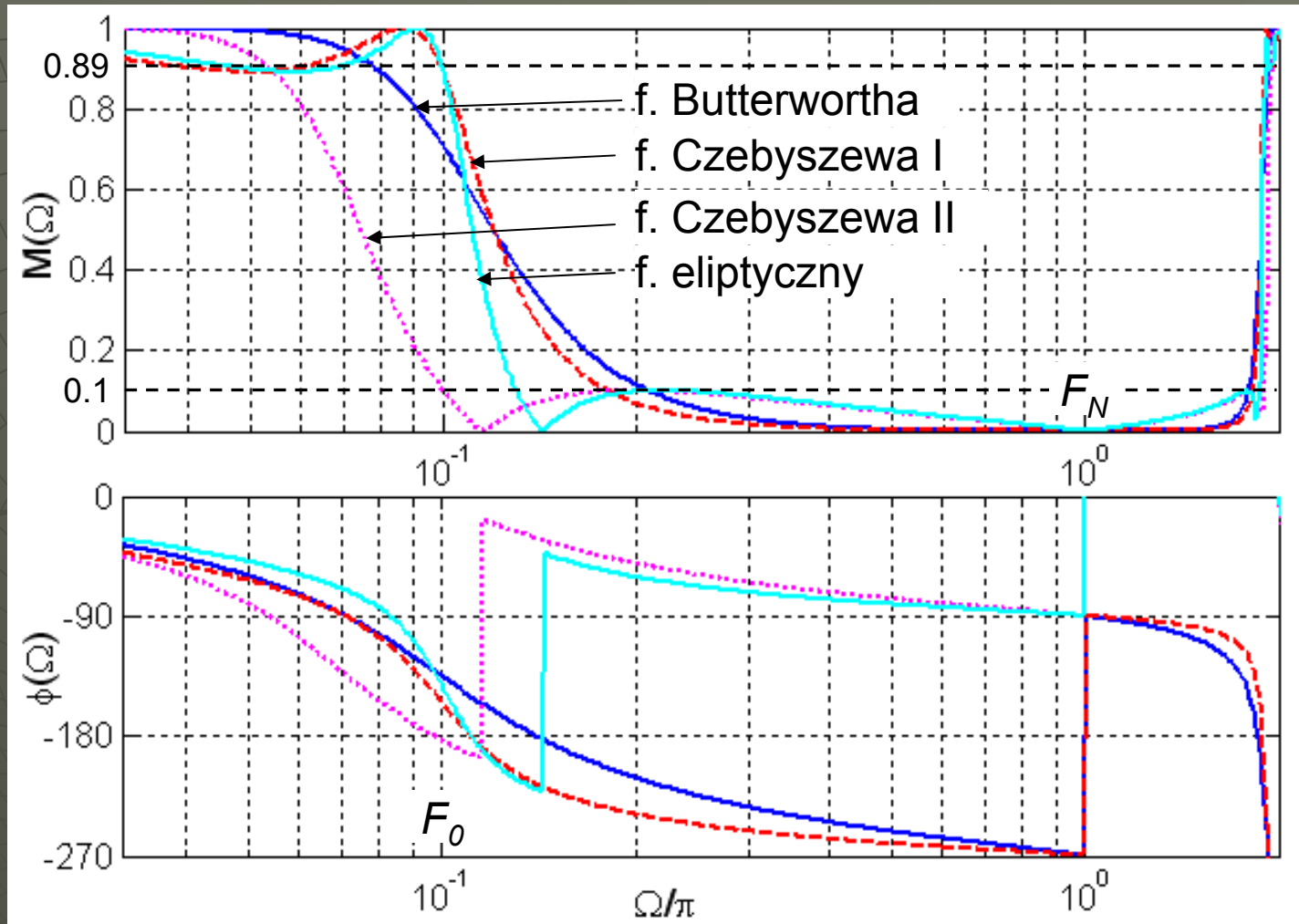


Rys. Charakterystyki filtra Butterwortha rzędu $N = 2, 4, 6$.
 Częstotliwość odcięcia $F_0 = 0.1F_N$, $M(F_0) = -3\text{dB}$

Mianownik prototypu Butterwortha:

$$M_B(s) = s^N + a_{N-1}s^{N-1} + \dots + a_1s + 1$$

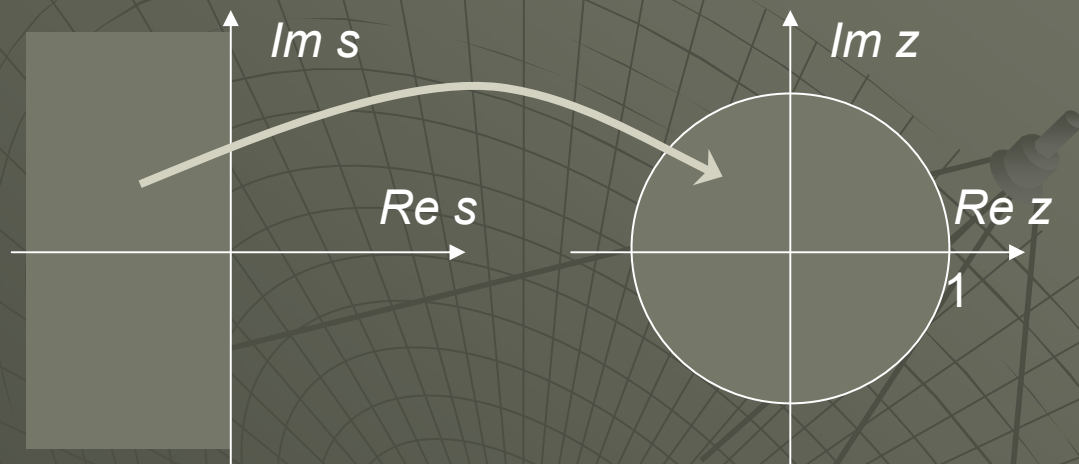
$$a_n = \prod_{i=1}^n \frac{\cos \frac{\pi(i-1)}{2N}}{\sin \frac{\pi i}{2N}}, \quad n = 1, 2, \dots, N-1$$



Rys. Porównanie charakterystyk filtrów LP różnego typu, tego samego rzędu $N=3$. Parametry $F_0=0.1$, $r_p=1\text{dB}$ (Czebyszew I rodzaju, eliptyczny), $r_s=20\text{dB}$ (Czebyszew II rodzaju, eliptyczny)

Metody dyskretyzacji

1. Metoda *odwzorowania blińowego* płaszczyzny 's' na 'z'



$$H(z) = H_a(s) \Big|_{s = \frac{2}{T_s} \cdot \frac{1-z^{-1}}{1+z^{-1}}}$$

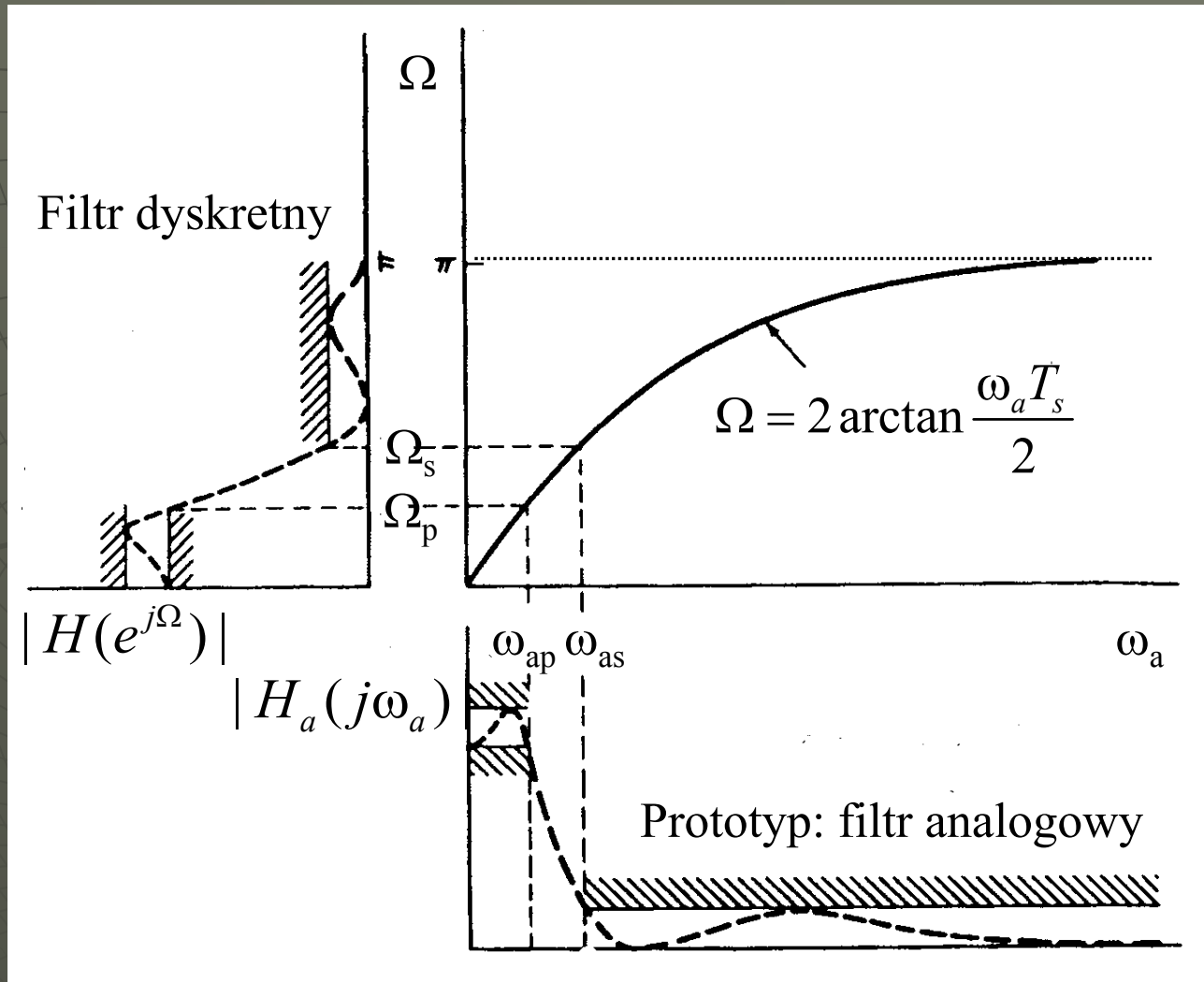
Zalety: odwzorowanie wzajemnie jednoznaczne, zachowuje stabilność/niestabilność układu.

Wada: *nieliniowe przekształcenie skali częstotliwości*

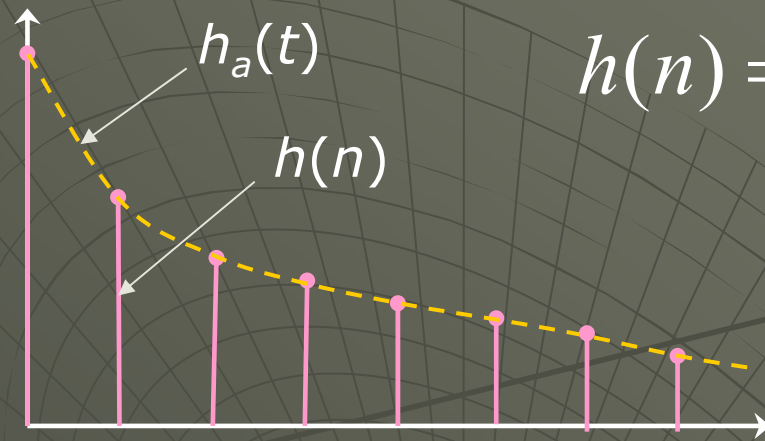
(ω_a – pulsacja analogowa, ω_d – pulsacja dyskretna, rad/s)

$$\Omega = 2 \cdot \arctan \frac{\omega_a T_s}{2}$$

$$\Omega = \omega_d T_s$$



Rys. Wpływ nieliniowego przekształcenia częstotliwości odwzorowania biliniowego na różnice granic pasm prototypu analogowego i dyskretnego odpowiednika

2. Metoda *niezmienniczej odpowiedzi impulsowej* (Impulse Invariant)

$$h(n) = h_a(t) \Big|_{t=nT_s}$$

Rozkład $H_a(s)$ na ułamki proste:

$$H_a(s) = K + \sum_{k=1}^N \frac{R_k}{s - s_k}$$

Przekształcenie biegunów analogowych na dyskretne według

zależności $z = e^{sT_s}$

$$H(z) = K + \sum_{k=1}^N \frac{R_k}{1 - e^{s_k T_s} z^{-1}}$$

Zalety: liniowe przekształcenie skali częstotliwości, zachowuje stabilność/niestabilność układu.

Wada: *niejednoznaczność przekształcenie skali częstotliwości*, występuje efekt sumowania się okresowo poprzesuwanym charakterystyk prototypu

$$H(e^{j\Omega}) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} H_a \left(j \frac{\Omega}{T_s} + jk\omega_s \right)$$

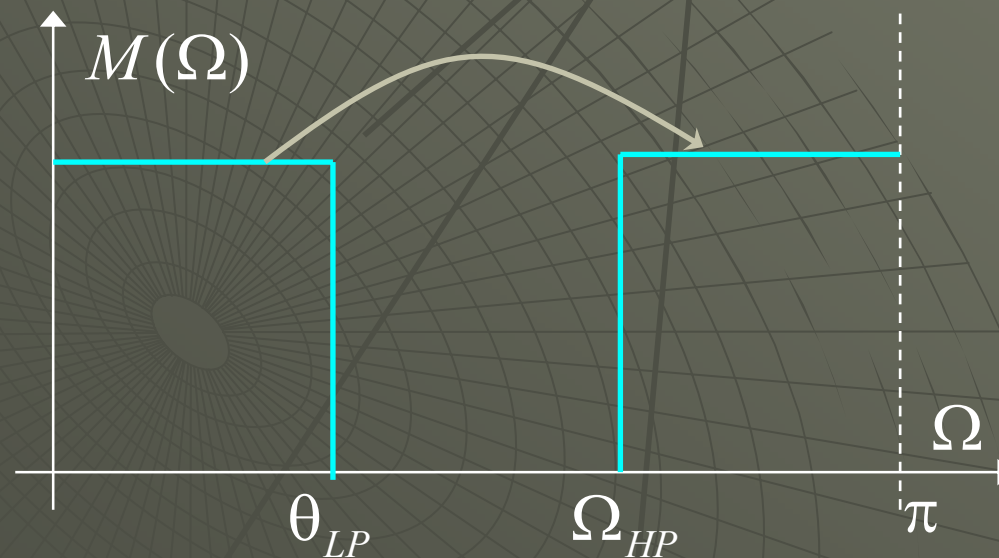
Ważne: Z tego względu metodą tą nie można dyskretyzować filtrów analogowych, które w zakresie wysokich częstotliwości mają pasmo przepustowe (typu HP i BS), ponieważ wynikowa ch-ka filtra dyskretnego byłaby zniekształcona przez aliasing.

3. Metoda *odwzorowania zer i biegunów* (Zero-Pole Matching)

Bieguny i zera skończone transmitancji analogowej s_k przekształca się na dyskretne wg zależności $z_k = e^{s_k T_s}$

TRANSFORMACJE Z PROTOTYPU FILTRA DOLNOPRZEPUSTOWEGO
O CZĘSTOTLIWOŚCI ODCIĘCIA θ_p NA FILTR INNEGO TYPU
(wg Oppenheim, Schaffer: *Cyfrowe przetwarzanie sygnałów*)

Przykład: transformacja LP na HP dla $\alpha=0$, czyli $\Omega_{HP} = \pi - \theta_{LP}$



Typ filtra	Przekształcenie	Wzory projektowe
dolnoprzepustowy	$Z^{-1} = \frac{Z^{-1} - \alpha}{1 - \alpha Z^{-1}}$	Ω_p - wymagana pulsacja odcięcia $\alpha = \frac{\sin\left(\frac{\theta_p - \Omega_p}{2}\right)}{\sin\left(\frac{\theta_p + \Omega_p}{2}\right)}$
górnoprzepustowy	$Z^{-1} = -\frac{Z^{-1} + \alpha}{1 + \alpha Z^{-1}}$	Ω_p - wymagana pulsacja odcięcia $\alpha = -\frac{\cos\left(\frac{\Omega_p + \theta_p}{2}\right)}{\cos\left(\frac{\Omega_p - \theta_p}{2}\right)}$
pasmowoprzepustowy	$Z^{-1} = -\frac{Z^{-2} - \frac{2\alpha k}{k+1} Z^{-1} + \frac{k-1}{k+1}}{\frac{k-1}{k+1} Z^{-2} - \frac{2\alpha k}{k+1} Z^{-1} + 1}$	Ω_2, Ω_1 - wymagane pulsacje odcięcia: górna i dolna $\alpha = \frac{\cos\left(\frac{\Omega_2 + \Omega_1}{2}\right)}{\cos\left(\frac{\Omega_2 - \Omega_1}{2}\right)}$ $k = \operatorname{ctg}\left(\frac{\Omega_2 - \Omega_1}{2}\right) \cdot \operatorname{tg}\left(\frac{\theta_p}{2}\right)$
pasmowozaporowy	$Z^{-1} = -\frac{Z^{-2} - \frac{2\alpha}{1+k} Z^{-1} + \frac{1-k}{1+k}}{\frac{1-k}{1+k} Z^{-2} - \frac{2\alpha}{1+k} Z^{-1} + 1}$	Ω_2, Ω_1 - wymagane pulsacje odcięcia: górna i dolna $\alpha = \frac{\cos\left(\frac{\Omega_2 + \Omega_1}{2}\right)}{\cos\left(\frac{\Omega_2 - \Omega_1}{2}\right)}$ $k = \operatorname{tg}\left(\frac{\Omega_2 - \Omega_1}{2}\right) \cdot \operatorname{tg}\left(\frac{\theta_p}{2}\right)$

2. Metody optymalizacyjne

Projektowanie filtrów NOI o *dowolnym* przebiegu charakterystyki jest możliwe z wykorzystaniem iteracyjnych metod optymalizacji zadanego wskaźnika jakości (metod numerycznych).

Metoda (uogólnionego) błędu najmniejszych kwadratów

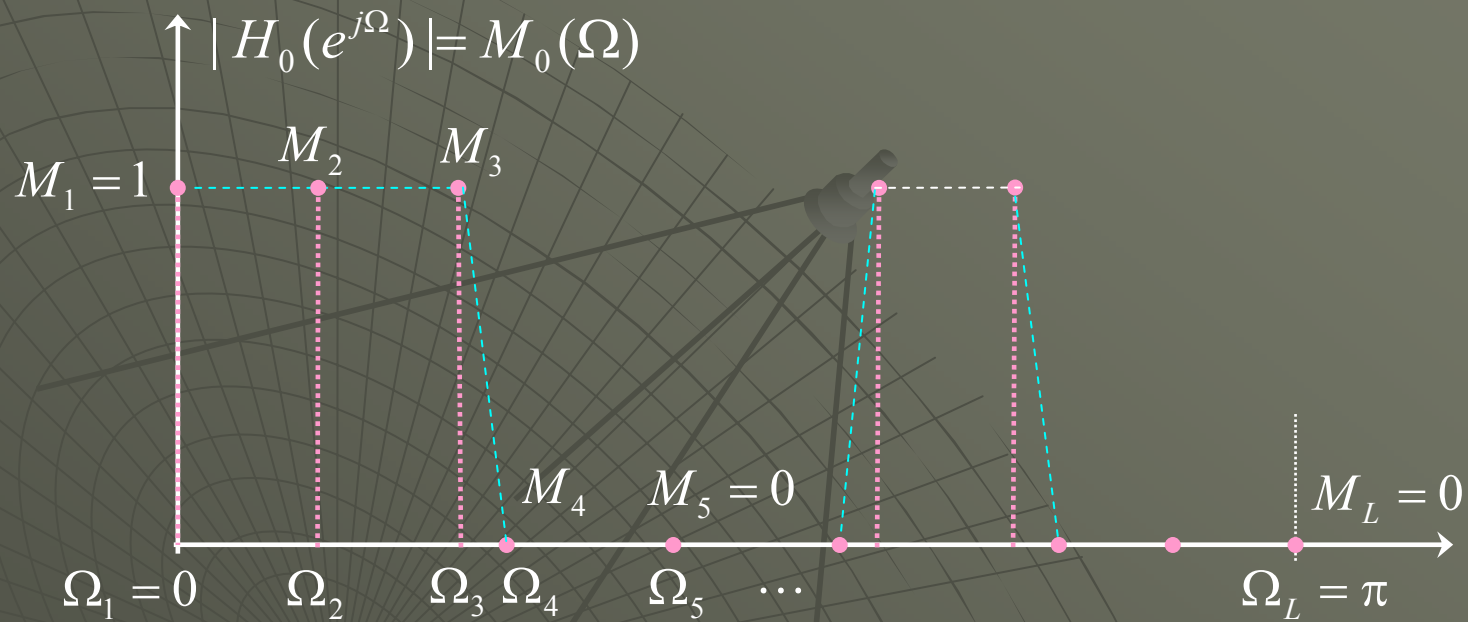
Charakterystykę projektowaną zadaje się za pomocą dyskretnego zbioru punktów charakterystycznych:

$$\{M_i, \Omega_i : i = 1, 2, \dots, L\} \quad \text{gdzie} \quad M_i = |H_0(e^{j\Omega_i})|$$

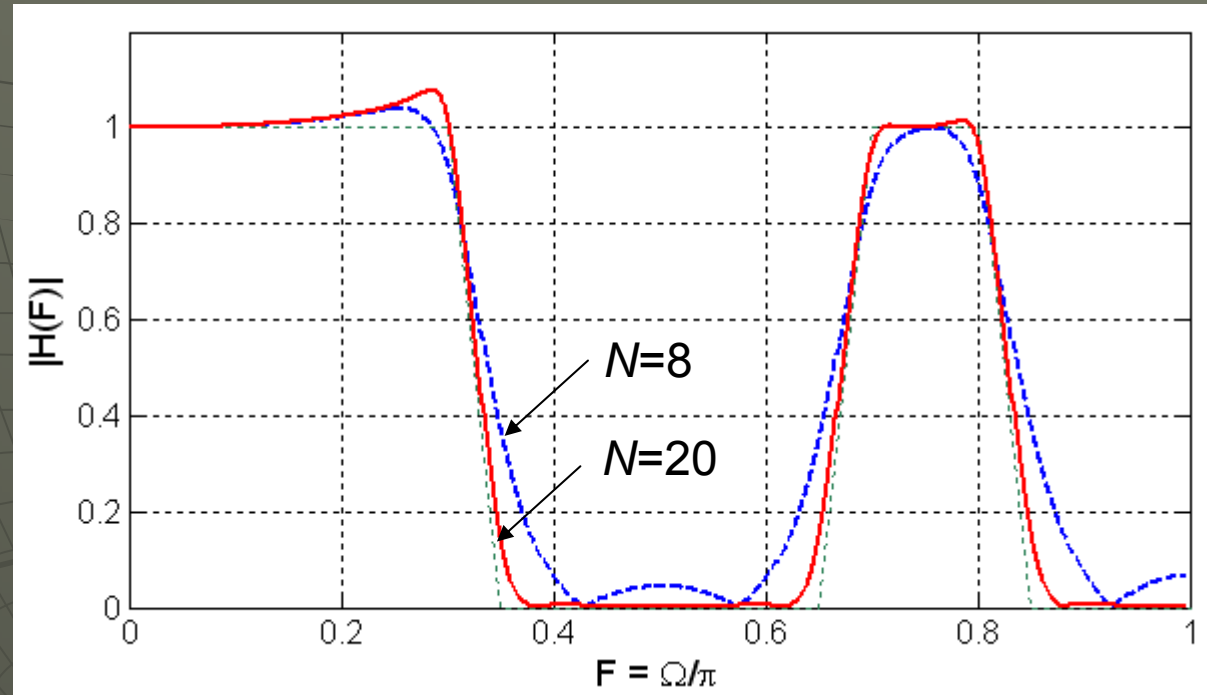
Optymalne współczynniki transmitancji $H(e^{j\Omega})$ filtra otrzymuje się *minimalizując wskaźnik jakości*:

$$E_k(\Omega_i) = \sum_{i=1}^L W(\Omega_i) \left[|H_0(e^{j\Omega_i})| - |H(e^{j\Omega_i})| \right]^{2p}$$

Metoda nie daje informacji o maksymalnym błędzie aproksymacji.



Rys. Wymagana ch-ka amplitudowa filtra z dwoma pasmami przepustowymi zadana za pomocą siatki punktów



Rys. Projekt filtra NOI otrzymany metodą Yule-Walkera

W *Signal Processing Toolbox* Matlab:

$[B, A] = \text{yulewalk}(N, F, M)$

gdzie: N – rząd filtra,

F – wektor unormowanych częstotliwości $F_i = \Omega_i/\pi$ (z zakresu $[0, 1]$),
 M – wektor magnitud M_i ; F i M są mapowane na gęstą siatkę pktów

$F = [0 \ 0.3 \ 0.35 \ 0.65 \ 0.7 \ 0.8 \ 0.85 \ 1];$

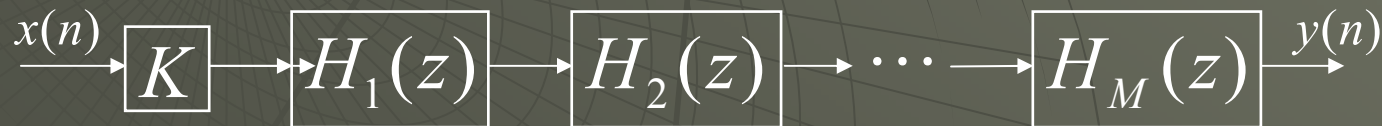
$M = [1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0];$

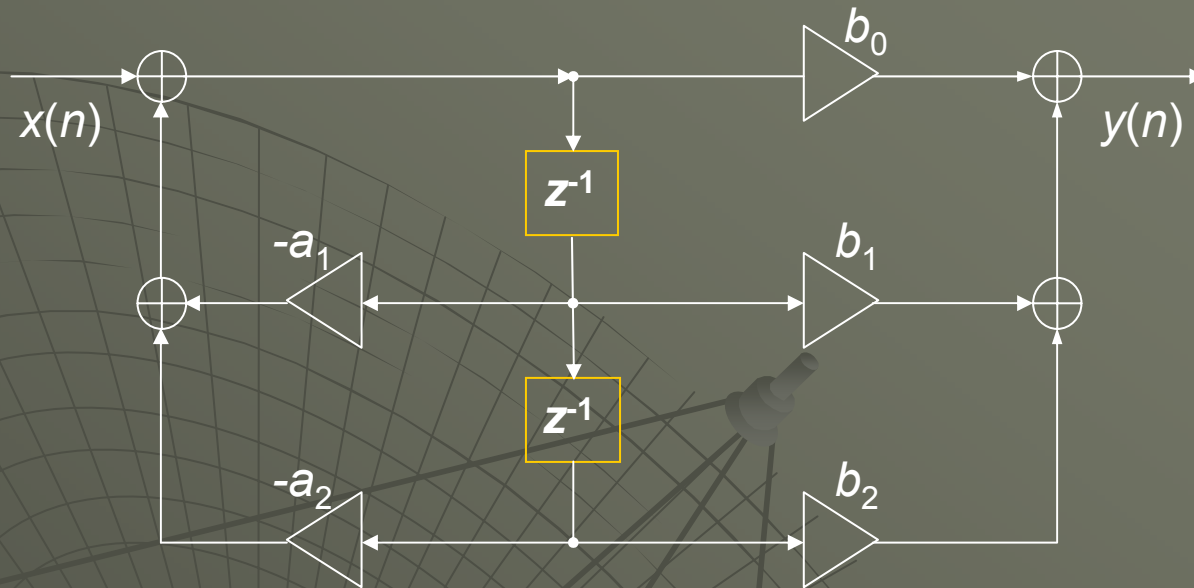
Filtry NOI wyższych rzędów projektuje się w formie szeregowego połączenia elementów 2-go rzędu, tzw. *sekcji bikwadratowych*, o transmitancji:

$$H_{sb}(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

Zespolone bieguny sekcji położone w punktach $z_{1,2} = r \cdot e^{\pm j\Omega}$ wyznacza się z zależności $a_1 = -2r \cos \Omega$, $a_2 = r^2$

$$H(z) = K \cdot H_1(z) \cdot H_2(z) \cdot \dots \cdot H_M(z) = K \prod_{i=1}^M \frac{b_{0i} + b_{1i} z^{-1} + b_{2i} z^{-2}}{1 + a_{1i} z^{-1} + a_{2i} z^{-2}}$$



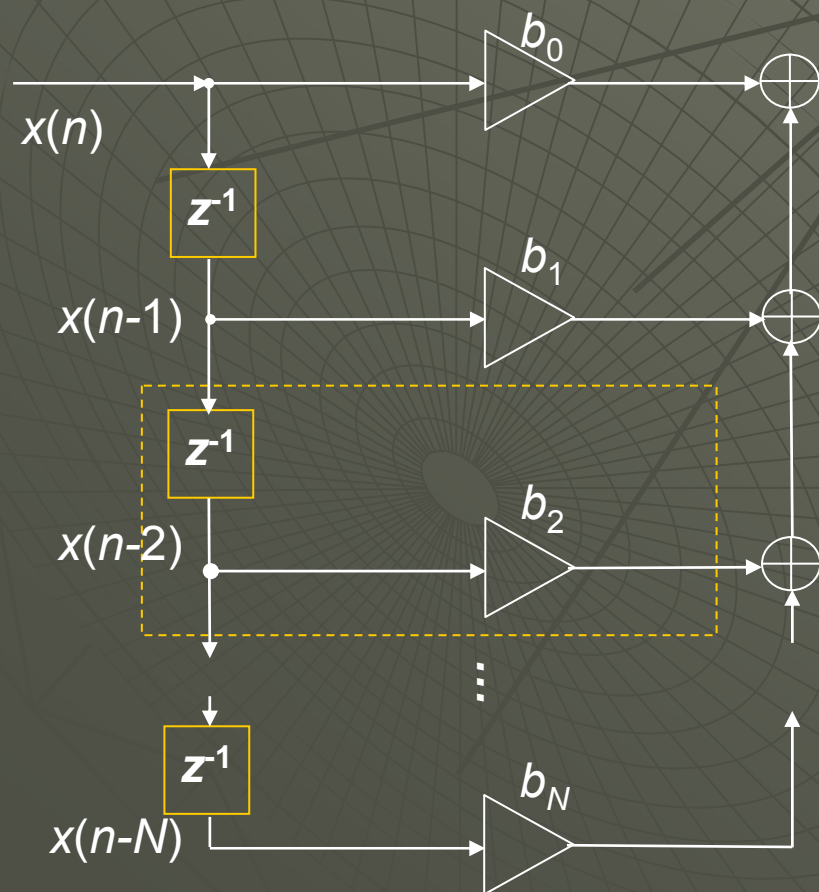


Rys. Schemat sekcji bikwadratowej
Filtry NOI są nazywane rekursywnymi ze względu na obecność w strukturze sprzężenia zwrotnego (wielomian $A(z)$)

Projektowanie filtrów SOI (*Finite Impulse Response – FIR*)

Filtry o skończonej odpowiedzi impulsowej (nierekursywne) mają transmitancję postaci:

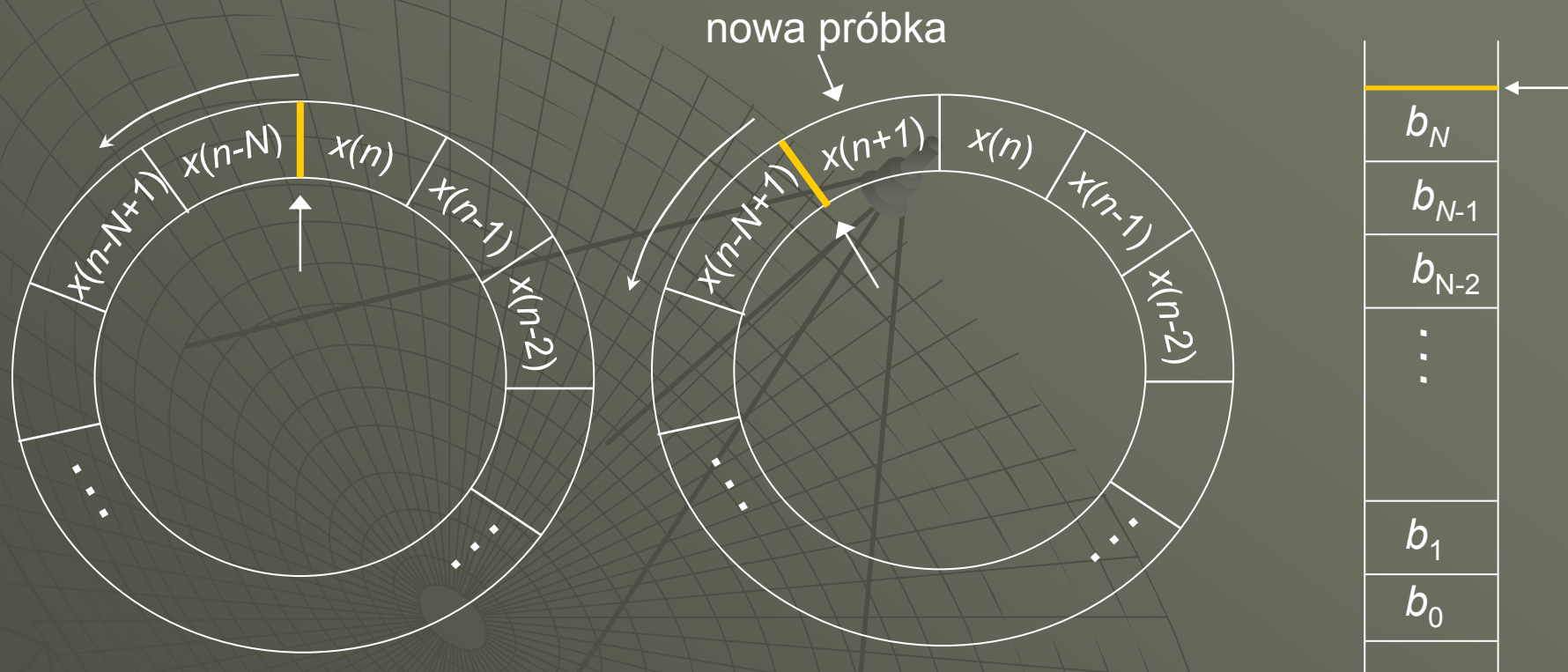
$$H(z) = B(z) = b_0 + b_1 z^{-1} + \dots + b_N z^{-N}, \quad A(z) = 1$$



Rys. Struktura bezpośrednia (transwersalna) filtra SOI rzędu N (linia opóźniająca z odczepami, zaznaczony 1 *tap*)

Implementacja realizacji bezpośredniej filtra SOI

$$y(n) = b_0 x(n) + b_1 x(n-1) + \dots + b_{N-1} x(n-N+1) + b_N x(n-N)$$



Rys. Bufor kołowy (*circular buffer*) próbek, bufor liniowy współczynników i jeden cykl działania jednostki MAC procesora

$$\begin{cases} y_i := y_{i-1} + a_i * x_i, & x_i = x(n - N + i), & a_i = b_{N-i} \\ x_i := x_{i+1}, & i = 0, 1, \dots, N, & y_{-1} = 0 \end{cases}$$

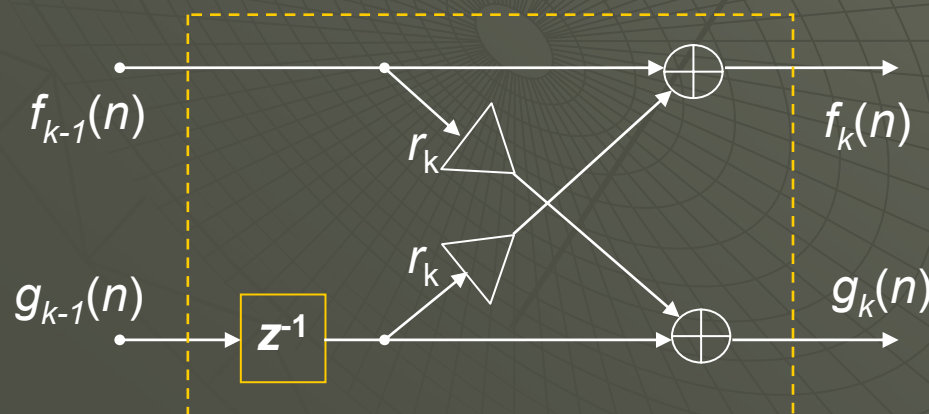
Forma kratownicowa filtra SOI (*lattice filter*)

Realizacja kratownicowa filtrów SOI ($H(z)=B(z)$ z $b_0=1$) i NOI z samymi biegunami ($H(z)=1/A(z)$, *all-pole IIR*) jest popularna w wielu aplikacjach (np. filtry wybielające rozkładające sygnał na niezależne składowe ortogonalne) ze względu na modułarną strukturę, małą wrażliwość na błędy kwantyzacji parametrów i proste kryterium stabilności.

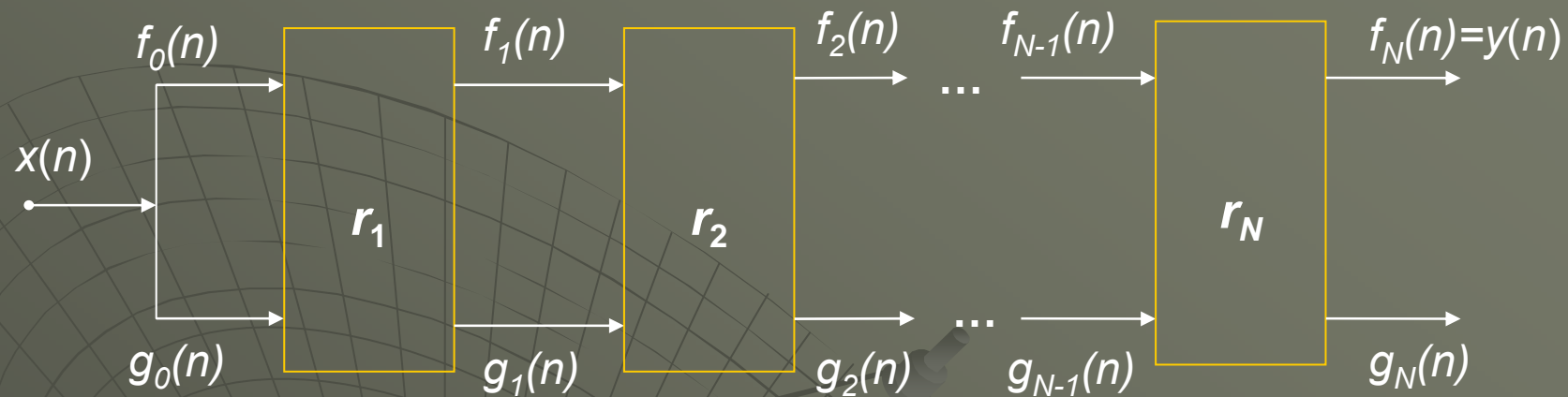
Filtr kratownicowy jest połączeniem kaskadowym sekcji określonych przez współczynnik odbicia r_k (*reflection coefficient*) i opisanych równaniami:

$$f_k(n) = f_{k-1}(n) + r_k g_{k-1}(n-1)$$

$$g_k(n) = g_{k-1}(n-1) + r_k f_{k-1}(n)$$



Rys. Sekcja kratownicowa filtra SOI



Rys. Połączenie kaskadowe N sekcji kratownicowych filtra SOI

Transmitancja od $x(n)$ do $f_k(n)$: $F_k(z) = B_k(z)X(z)$

Korzystając z równań sekcji otrzymujemy zależność rekurencyjną:

$$B_k(z) = B_{k-1}(z) + r_k z^{-k} B_{k-1}(z^{-1})$$

z warunkiem początkowym $B_0(z) = 1$ (bo $b_0 = 1$ – ważne!).

Transformacja odwrotna daje (rekursja Levinsona):

$$b_k(0) = 1,$$

$$b_k(n) = b_{k-1}(n) + r_k b_{k-1}(k-n), \quad n = 1, 2, \dots, k-1$$

$$b_k(k) = r_k.$$

Procedura obliczeń współczynników odbicia w języku Matlab:

```
b=[1 b1 b2 ... bn];  
h=b;  
p=length(h)-1;  
r=zeros(p,1); % inicjalizacja  
for k=p:-1:1  
    r(k)=h(k+1); % ostatni współczynnik  
    h=(h-r(k)*flip1r(h))/(1-r(k)^2);  
    h=h(1:k); % obcięcie ostatniego zera  
end
```

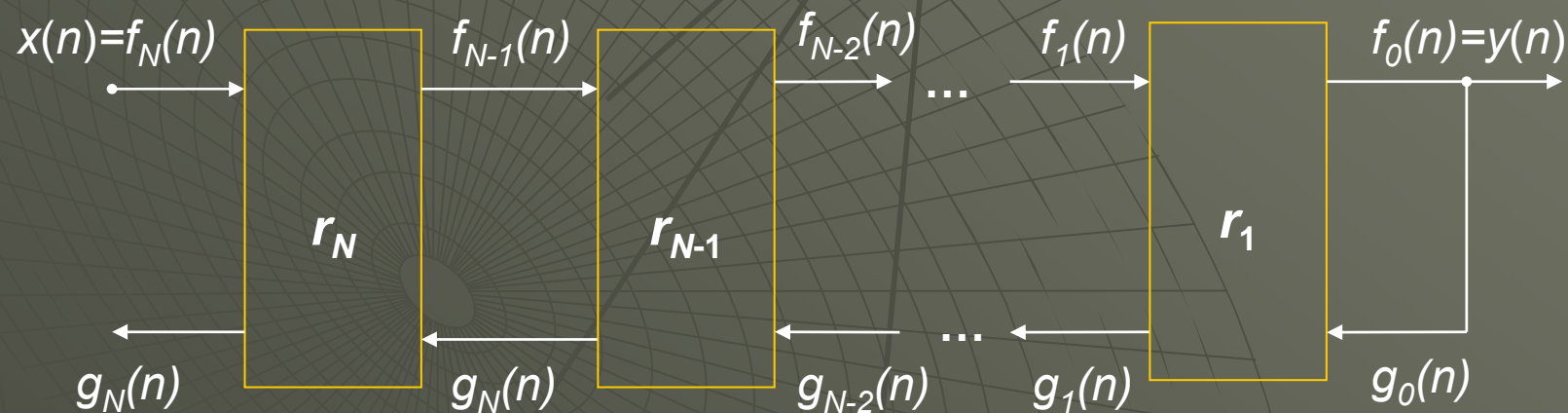
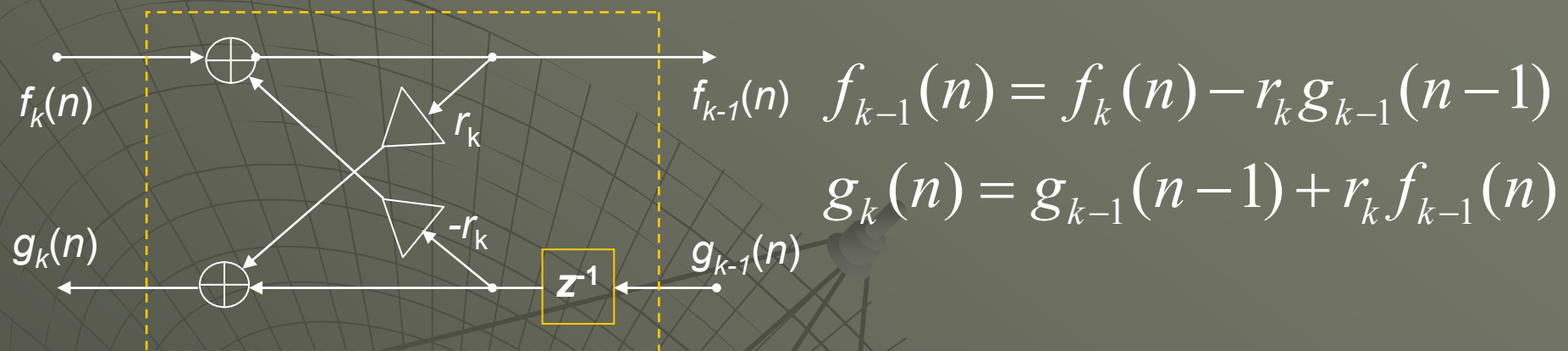
W *Signal Processing Toolbox* jest funkcja:

`r = poly2rc(b)`

Ważne właściwości:

1. Jeżeli wszystkie współczynniki odbicia filtra kratownicowego spełniają warunek $|r_k| < 1$, $k=1,2,\dots,N$, to filtr jest stabilny (prosty test stabilności).
2. Zmiana rzędu filtra, tzn. dodanie lub usunięcie sekcji kratownicowych, nie zmienia pozostałych współczynników (ortogonalność).

Forma kratownicowa filtra *all-pole* NOI ($H(z)=1/A(z)$, $a_0=1$)



Połączenie struktury kratownicowej z transwersalną umożliwia realizację filtrów NOI z N biegunami i N zerami.

Charakterystyka impulsowa filtra SOI:
$$h(n) = \begin{cases} b_n & \text{dla } n = 0, 1, \dots, N \\ 0 & \text{dla } n > N \end{cases}$$

$$y(n) = \sum_{k=0}^N h(k) \cdot x(n-k), \quad h(k) = b_k$$

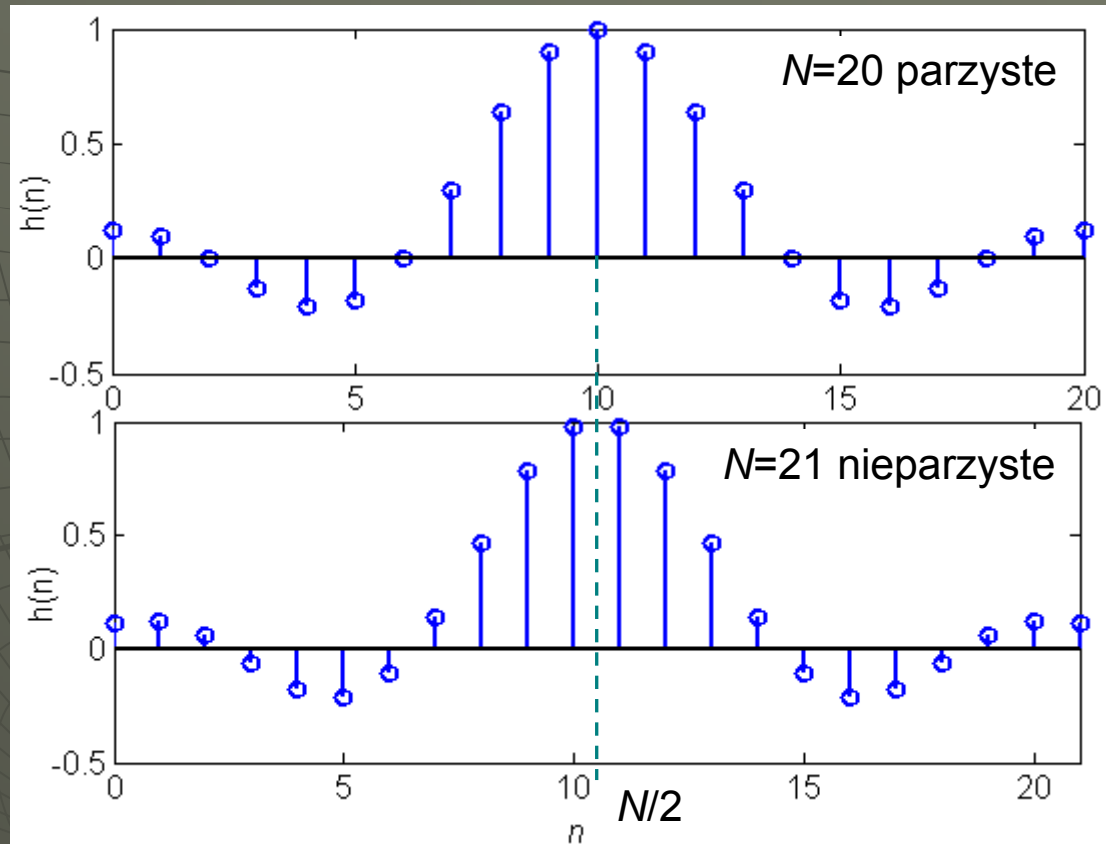
Stany przejściowe w filtrach SOI są skończone w czasie i trwają dokładnie N okresów próbkowania (tyle ile wynosi rząd filtra)

Ważną cechą filtrów SOI jest to, że **mogą one mieć dokładnie liniową charakterystykę fazową** (stałe opóźnienie grupowe).

Warunkiem uzyskania filtra o liniowej fazie jest *symetria* jego współczynników (próbek odpowiedzi impulsowej):

$$(a) \quad h(n) = h(N-n) \quad \text{lub} \quad (b) \quad h(n) = -h(N-n)$$

Symetria ta oznacza, że filtry SOI o liniowej fazie *nie są* minimalnofazowe, bo każde zero z_i filtra ma zwierciadlane odbicie $1/z_i^*$. Ch-kę amplitudową można projektować niezależnie.



Rys. Przykładowe charakterystyki impulsowe (współczynniki) filtrów SOI o liniowej fazie

Filtry SOI o liniowej fazie i symetrii parzystej (a) mają następujące charakterystyki widmowe:

$$H(e^{j\Omega}) = \begin{cases} e^{-j\Omega\frac{N}{2}} \cdot \left[h\left(\frac{N}{2}\right) + \sum_{n=0}^{\frac{N-1}{2}} 2h(n) \cdot \cos\Omega\left(n - \frac{N}{2}\right) \right] & N \text{ parzyste} \\ e^{-j\Omega\frac{N}{2}} \cdot \left[\sum_{n=0}^{\frac{N-1}{2}} 2h(n) \cdot \cos\Omega\left(n - \frac{N}{2}\right) \right] & N \text{ nieparzyste} \end{cases}$$

Dla nieparzystego N zachodzi ograniczenie: $H(e^{j\pi}) = 0$.

1. Metoda okienkowa (skończonego rozwinięcia w szereg Fouriera)

Charakterystyka idealnego filtra selektywnego jest nieprzyczynowa i ma nieskończone rozwinięcie w szereg:

$$H_0(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} h_0(n)e^{-j\Omega n}, \quad \text{gdzie } h_0(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_0(e^{j\Omega})e^{j\Omega n} d\Omega$$

W rozwinięciu zostawiamy skończoną liczbę wyrazów nakładając okno czasowe:

$$h(n) = h_0(n) \cdot w(n)$$

Symetryczne obcięcie daje filtr nieprzyczynowy o *liniowej* fazie:

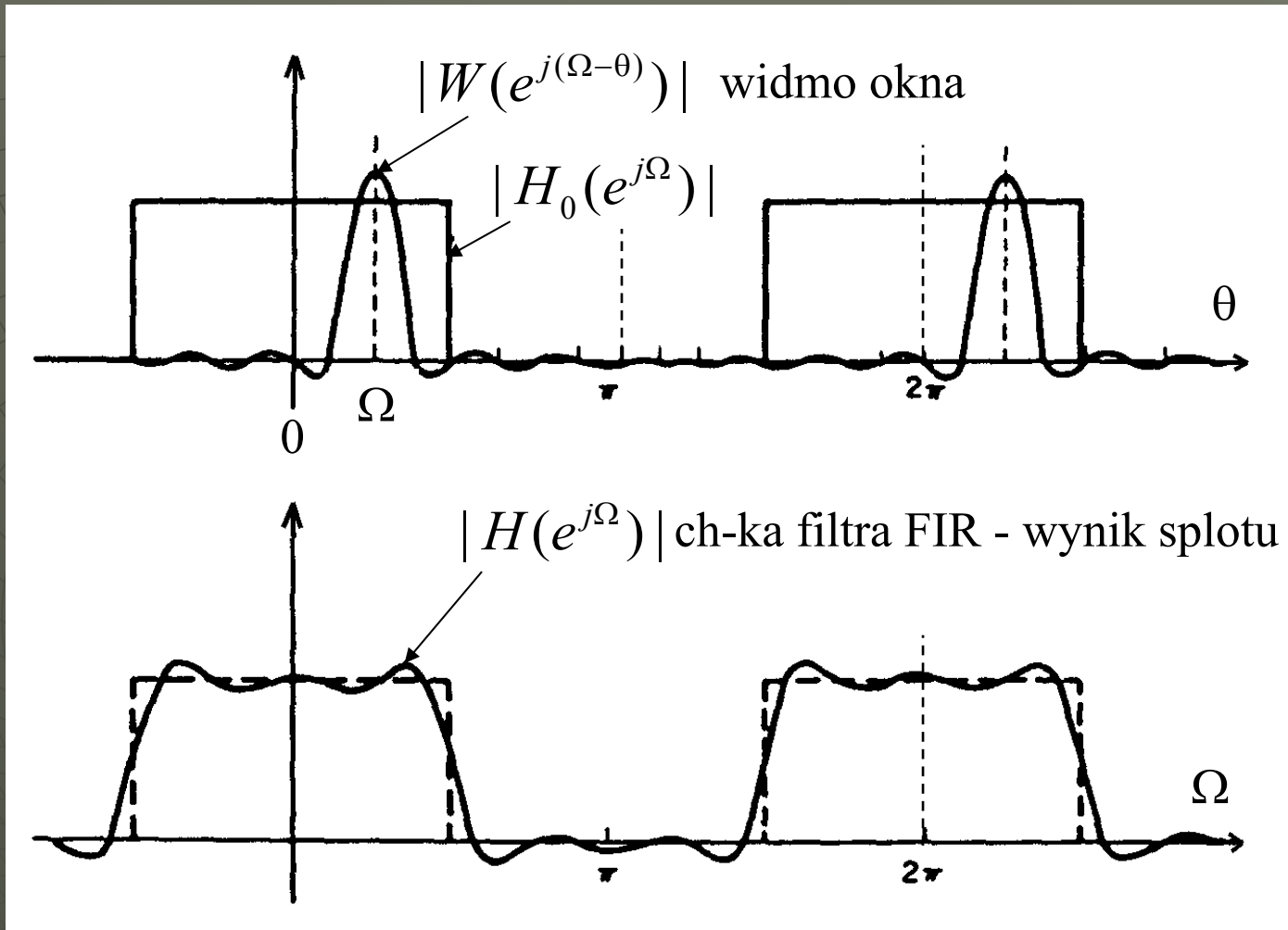
$$H_L(e^{j\Omega}) = \sum_{n=-L}^L h_L(n)e^{-j\Omega n} \Rightarrow H_L(z) = \sum_{n=-L}^L h_L(n)z^{-n}$$

Transmitancję SOI uzyskujemy opóźniając $H_L(z)$ o L okresów:

$$H(z) = z^{-L} H_L(z) = \sum_{n=0}^N h(n) z^{-n}$$

$$N = 2L, \quad h(n) = h_L(n - L), \quad n = 0, 1, \dots, N$$

Ch-ka widmowa otrzymanego filtra jest splotem ch-ki idealnej i widma nałożonego okna. Stosując okna inne niż prostokątne można zmniejszyć pofalowanie charakterystyki.



Rys. Charakterystyka filtra SOI LP jako wynik splotu w dziedzinie częstotliwości

Przykład: Projektowanie filtra LP:

$$H_0(e^{j\Omega}) = \begin{cases} 1 \cdot e^{-j\Omega\alpha} & \text{dla } |\Omega| \leq \Omega_0 \\ 0 & \text{dla pozostałych } \Omega \end{cases}$$

$$h_0(n) = \text{IDTFT}[H_0(e^{j\Omega})] =$$

$$= \frac{1}{2\pi} \int_{-\Omega_0}^{\Omega_0} e^{-j\Omega\alpha} \cdot e^{j\Omega n} d\Omega = \begin{cases} \frac{\sin \Omega_0 (n - \alpha)}{\pi(n - \alpha)} & \text{dla } n \neq \alpha \\ \frac{\Omega_0}{\pi} & \text{dla } n = \alpha \end{cases}$$

$$\alpha = \frac{N}{2} = L \quad \text{- opóźnienie grupowe}$$

2. Metody optymalizacyjne

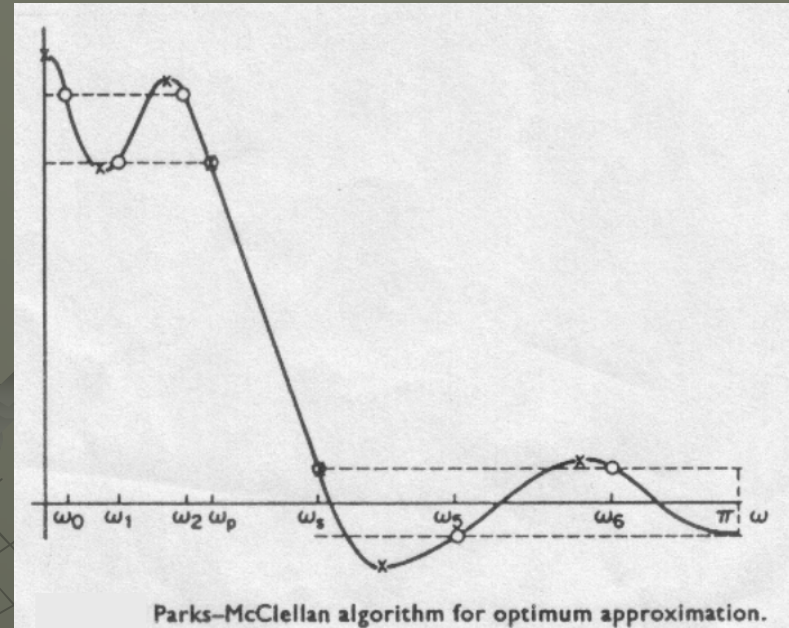
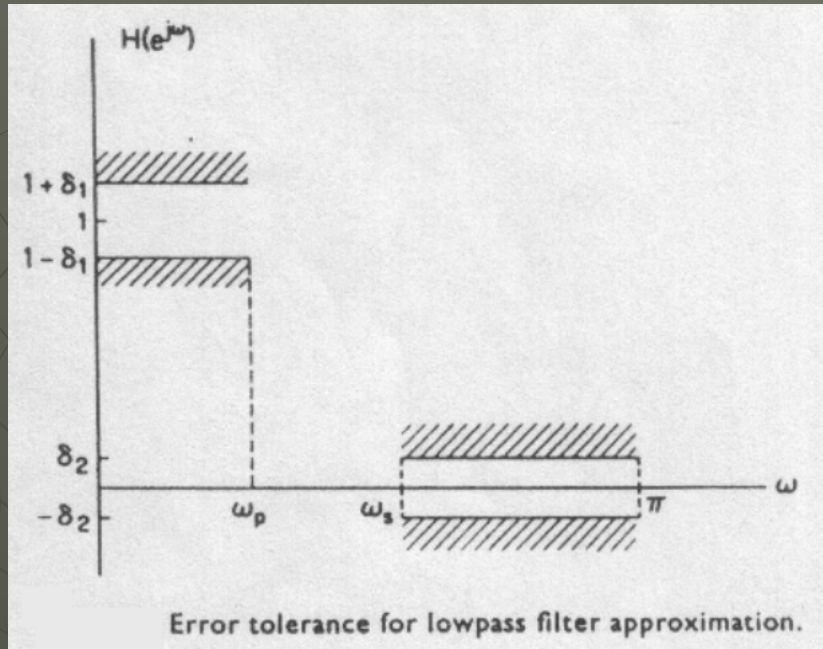
Metody te umożliwiają projektowanie filtrów o dowolnym (kawałkami liniowym) kształcie wymaganej charakterystyki amplitudowej $|H_0(e^{j\Omega})|$ metodą najmniejszych kwadratów lub, co stosowane jest częściej, tzw. **metody minimaksowe minimalizujące maksymalną odchyłkę aproksymacji** postaci:

$$\max_{\Omega_i} E(\Omega_i) = W(\Omega_i) \left(|H_0(e^{j\Omega_i})| - |H(e^{j\Omega_i})| \right)$$

gdzie $W(\Omega_i)$ – funkcja wagowa.

Aproksymacja minimaksowa (wielomianem trygonometrycznym, Czebyszewa) daje charakterystykę amplitudową **równomiernie pofalowaną (equiripple) z liniową fazą**. Daje kontrolę nad utrzymaniem charakterystyki filtra w zadanych granicach tolerancji (znana jest maksymalna odchyłka).

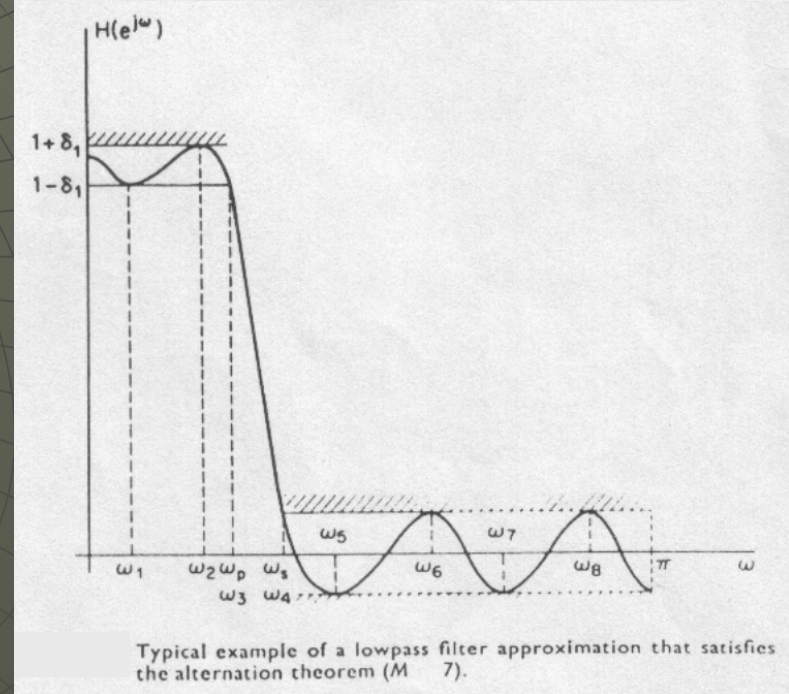
Do obliczeń wykorzystuje się iteracyjny algorytm wymiany Remeza.

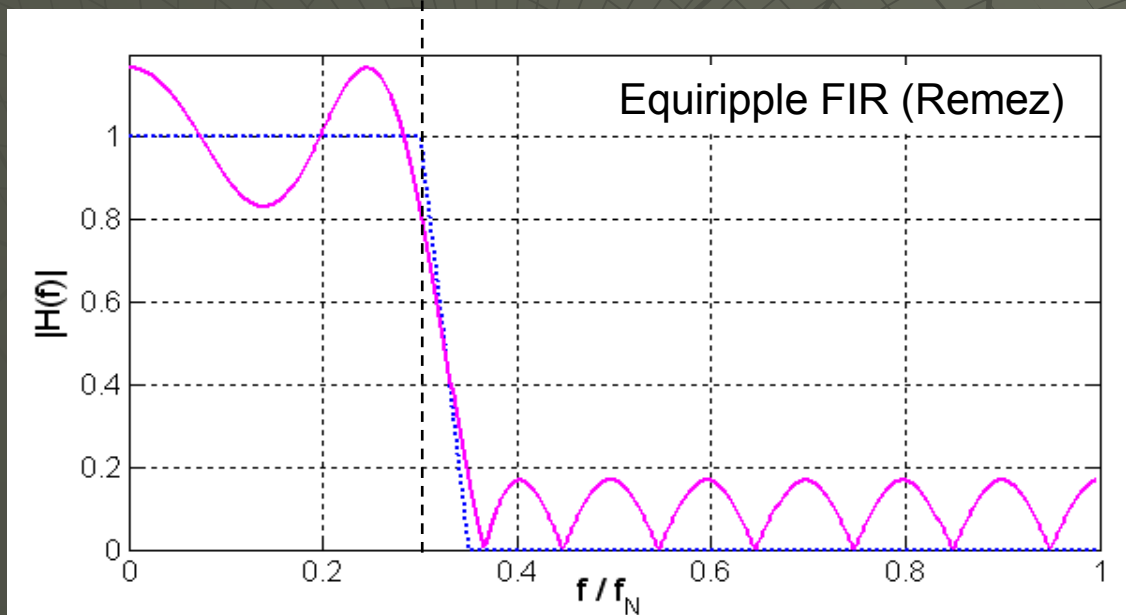
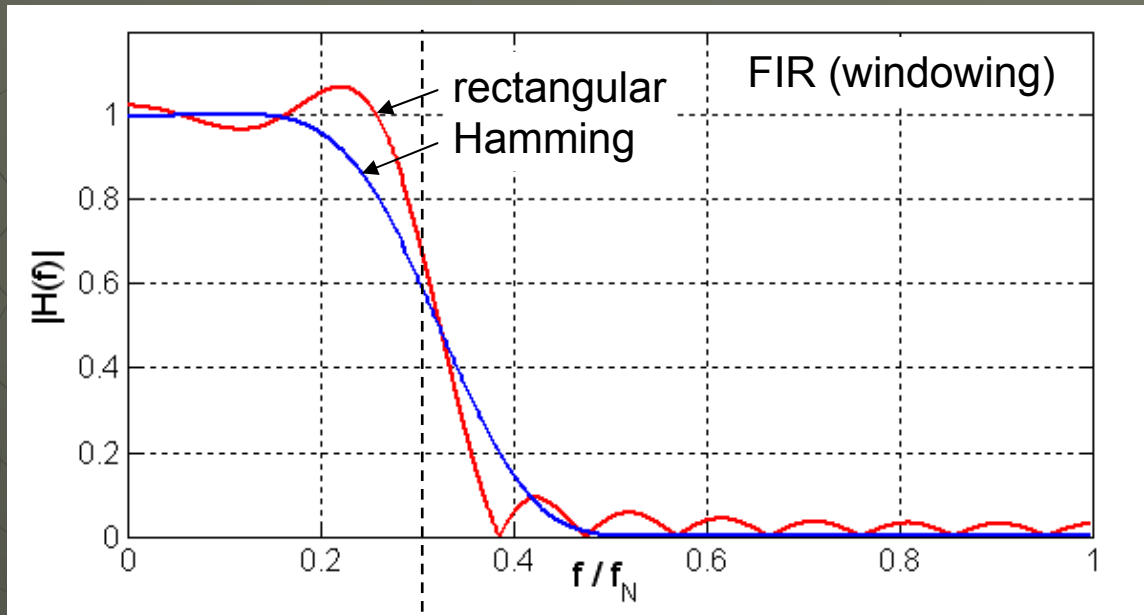


$$H(e^{j\Omega}) = h(0) + \sum_{n=1}^M 2h(n) \cos \Omega n =$$

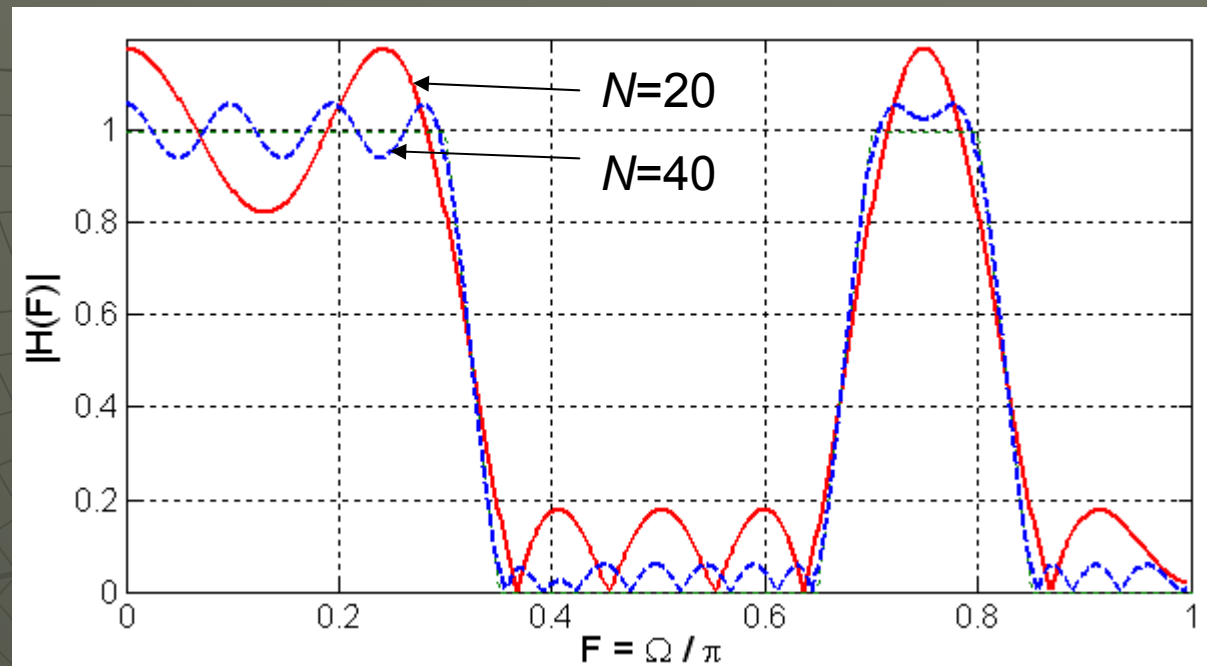
$$= c_0 + \sum_{n=1}^M c_n \cos \Omega n$$

Rys. Specyfikacja, algorytm Parksa-McClellana optymalnej aproksymacji i wynikowa charakterystyka LP





Rys. Porównanie ch-tyk filtrów SOI LP projektowanych metodami okienkowymi i metodą Remeza, rząd $N=20$



Rys. Projekt filtra SOI otrzymany metodą Remeza

W *Signal Processing Toolbox* Matlab:

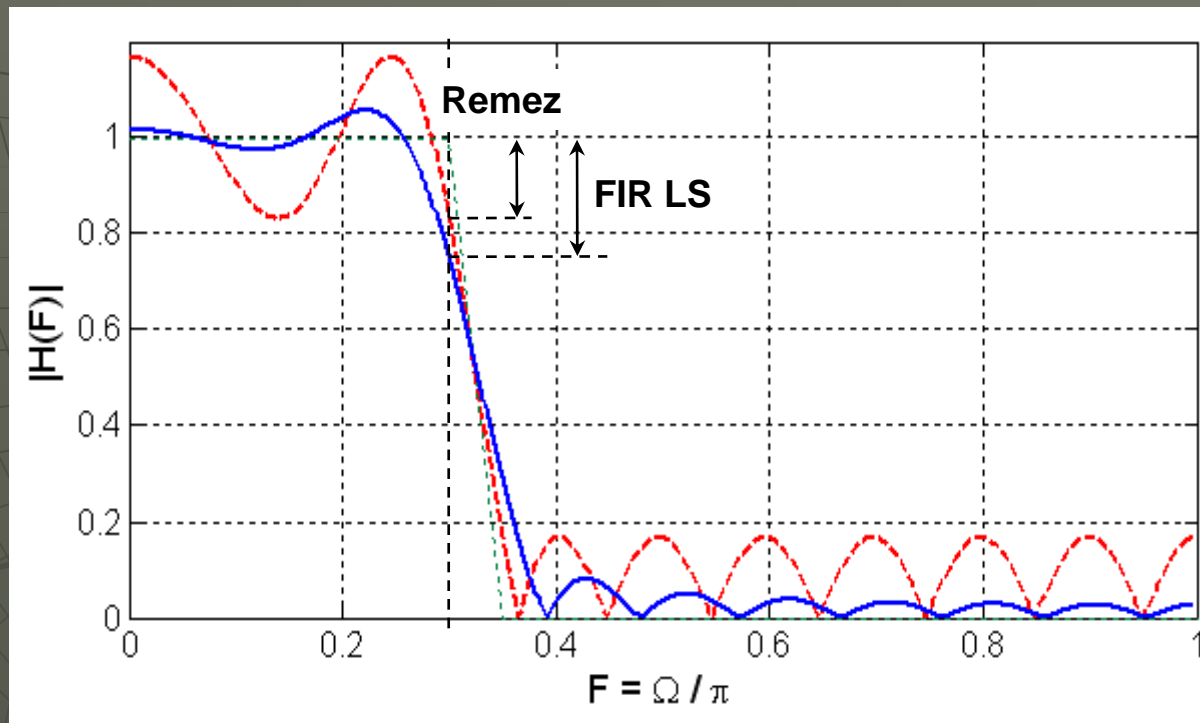
$B = \text{remez}(N, F, M)$

gdzie: N – rząd filtra,

F – wektor unormowanych częstotliwości $F_i = \Omega_i / \pi$ (z zakresu $[0, 1]$),
 M – wektor magnitud M_i ; F i M są mapowane na gęstą siatkę pktów

$F = [0 \ 0.3 \ 0.35 \ 0.65 \ 0.7 \ 0.8 \ 0.85 \ 1];$

$M = [1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0];$



Rys. Porównanie aproksymacji SOI najmniejszych kwadratów *LS* i Remeza (równomierne pofalowanie), zaznaczone odchyłki maksymalne na granicy pasm. $N=20$, pasmo przepustowe do $F_p=0.3$

Przykład w Matlabie: filtrowanie sygnału dźwiękowego z szumem w górnym paśmie od $F=0.5$. Zwrócić uwagę na efekt bliskości składowej sygnału od granicy pasm dla filtra NOI (zniekształcenie fazowe). (Porównać $F_p=0.3$ i 0.4)

```
F=[0 0.3 0.35 1]; M=[1 1 0 0];
```

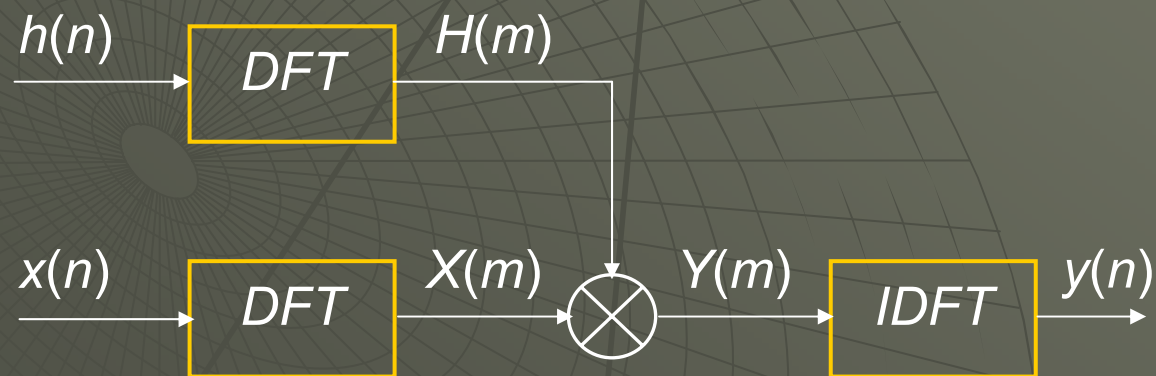
```
dspc3_filt(F,M,typ);
```

```
% typ=1 > Yule-Walker-20, typ=2 > Remez-40 order
```

Filtracja SOI przez mnożenie transformat w dziedzinie częstotliwości

W przypadku przetwarzania *blokowego*, tzn. obliczania odpowiedzi na *ciągi próbek* sygnału wejściowego $x(n)$ (ramki czasowe), stosuje się bardziej efektywny algorytm mnożenia transformat FFT.

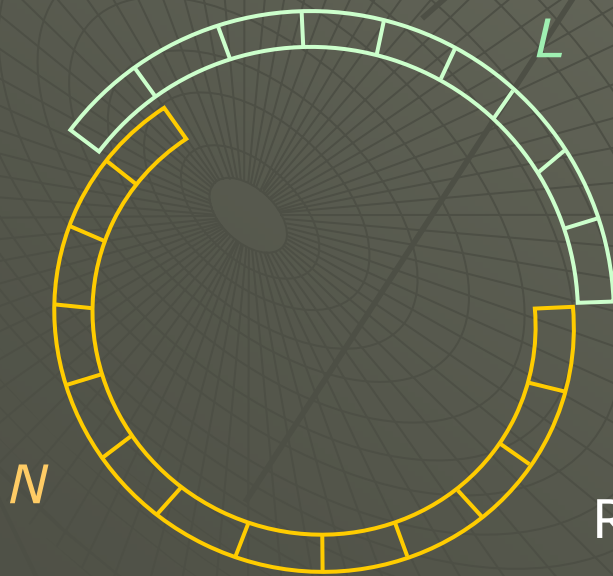
Oblicza się dwie M -punktowe DFT: ciągu próbek $x(n)$ $n=0,1,\dots,N-1$ oraz ciągu współczynników filtra $h(n)$, $n=0,1,\dots,L-1$ (z ew. uzupełnieniem zerami, aby obie DFT miały jednakowe długości), a następnie mnoży transformaty:



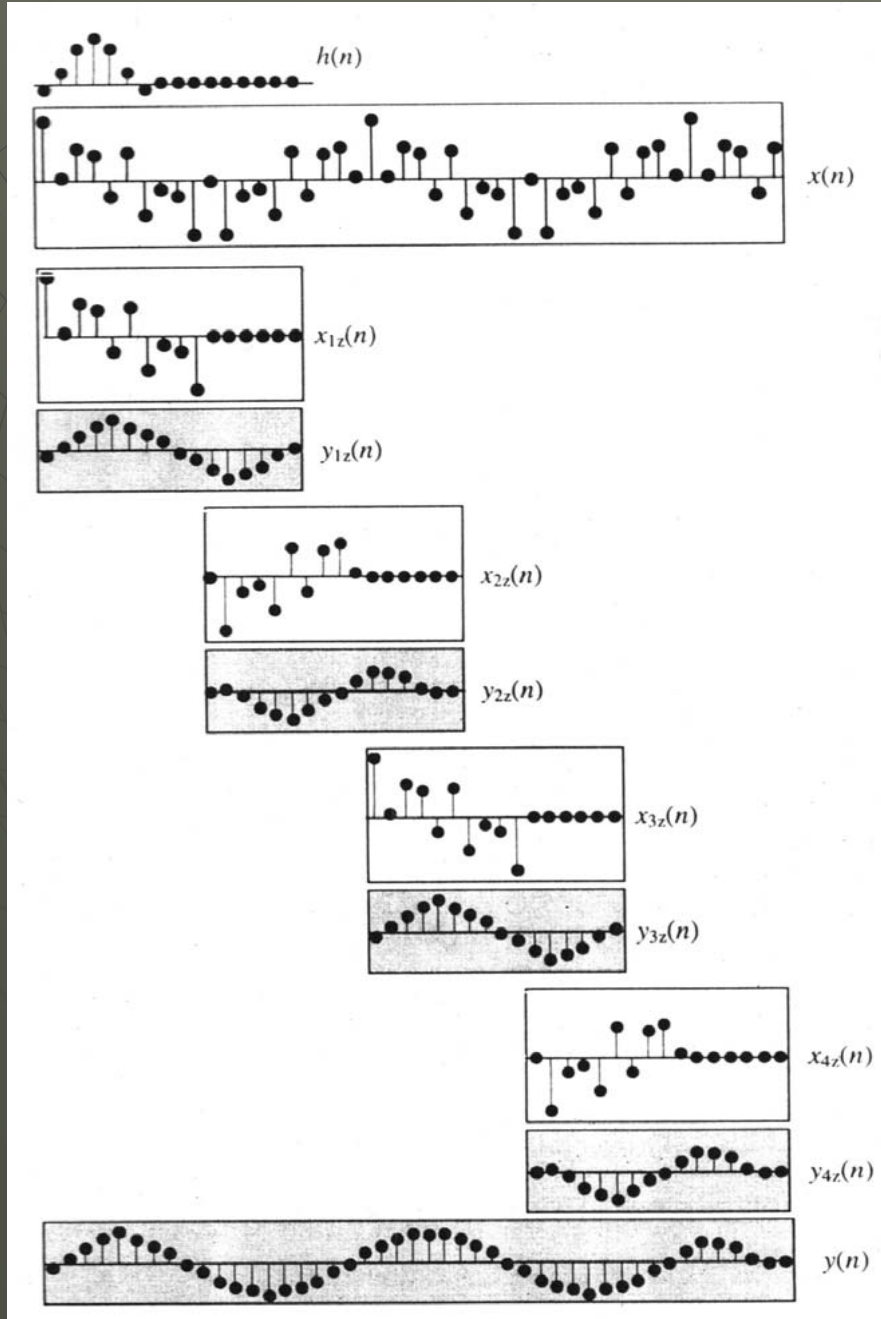
Przekształcenie odwrotne daje blok wartości sygnału wyjściowego:

$$y_p(n) = IDFT[H(m)X(m)]$$

Uwaga: operacja mnożenia transformat odpowiada splotowi cyklicznemu, który daje sygnał okresowy $y_p(n)$ o okresie M . Aby uniknąć aliasingu w czasie i uzyskać równość $y_p(n) = y(n)$ dla $n=0,1,\dots,M-1$, należy przyjmować $M \geq N+L-1$, czyli stosować *uzupełnianie zerami* bloków danych wejściowych do DFT.



Rys. Ilustracja splotu cyklicznego



Rys. Segmentacja szybkiego splotu metodą *overlap-add* (filtracja blok po bloku i sumowanie zachodzących na siebie segmentów wyjściowych): L – długość filtra (7), N_s – długość segmentu sygnału wejściowego (10), $M \geq N_s + L - 1$ – długość *FFT* i segmentu sygnału wyjściowego po *IFFT* (16). Odpowiedź impulsową filtra uzupełnia się $M - L$ zerami, a segment wejściowy – $M - N_s$ zerami. Uwaga: w przypadku filtracji przyczynowej $L - 1$ początkowych i końcowych próbek $y(n)$ jest obarczonych „błędami brzegowymi”. Aby wykorzystać końcowe próbki $x(n)$ należy go odpowiednio uzupełnić zerami, aby utworzyć jeszcze jeden segment .

Dyskretny filtr Hilberta

Jest to dyskretna aproksymacja filtra realizującego ciągłą transformację Hilberta:

$$H_a(j\omega) = \begin{cases} -j & \text{dla } \omega > 0 \\ 0 & \text{dla } \omega = 0 \\ +j & \text{dla } \omega < 0 \end{cases}$$

(filtr przepustowy dla wszystkich $\omega \neq 0$, przesuwnik fazy o $\pm\pi/2$, charakterystyka impulsowa $h_a(t) = 1/(\pi t)$).

Dyskretna aproksymacja o nieskończonej ch-ce impulsowej:

$$h_H(n) = \begin{cases} \sin^2(n\pi/2) & \text{dla } n \neq 0 \\ 0 & \text{dla } n = 0 \end{cases}$$

Ch-kę SOI uzyskuje się przez nałożenie okna czasowego:

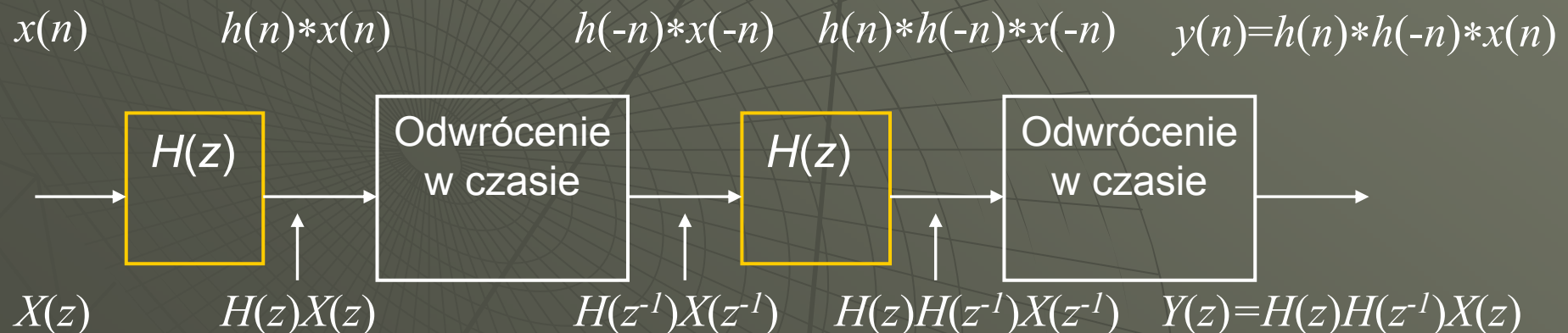
$$h(n) = h_H(n) \cdot w(n)$$

Filtracja bez przesunięcia fazowego (zerowa ch-ka fazowa)

Możliwe jest zrealizowanie *nieprzyczynowego*, który ma transmitancję $H(z)H(z^{-1})$ i nie wprowadza przesunięcia fazowego, ponieważ *charakterystyka widmowa jest rzeczywista* i określona wzorem

$$|H(e^{j\Omega})|^2 \cdot e^{j0}$$

Filtr wykorzystuje informacje z próbek leżących w czasie zarówno przed jak i po punkcie bieżącym.



Porównanie filtrów typu NOI i SOI

Filtry NOI mają nieliniowe, szczególnie na brzegach pasm, charakterystyki fazowe.

Przy takim samym rzędzie filtra zapewniają znacznie bardziej strome przejścia charakterystyki amplitudowej między pasmami, czyli lepszą selektywność.

Pożądany przebieg amplitudy można osiągnąć przy (nawet kilkakrotnie) mniejszym nakładzie obliczeń (niższy rząd filtra).

Filtry SOI są preferowane w zastosowaniach, w których podstawowym wymaganiem jest liniowość charakterystyki fazowej.

Nie zawierają sprzężenia zwrotnego z wyjścia, co upraszcza strukturę.

Są z założenia stabilne i mają krótkie przebiegi przejściowe. Optymalizacyjne i adaptacyjne procedury projektowania są prostsze.

Problemy związane z arytmetyką stałoprzecinkową

W tanich stałoprzecinkowych, zwykle 16-bitowych, procesorach DSP dominuje *ułamkowa reprezentacja liczb w kodzie uzupełnień do 2* (U2) (nazywa się ją arytmetyką 1.15 (15 oznacza liczbę bitów po kropce):

$$x = b_0.b_1b_2 \dots b_{M-1}b_M, \quad M = 15$$

$$(x)_{10} = -b_0 + \sum_{m=1}^M b_m 2^{-m}$$

$$x_{\min} = 1\ 000\ 0000\ 0000\ 0000 = (-1)_{10}$$

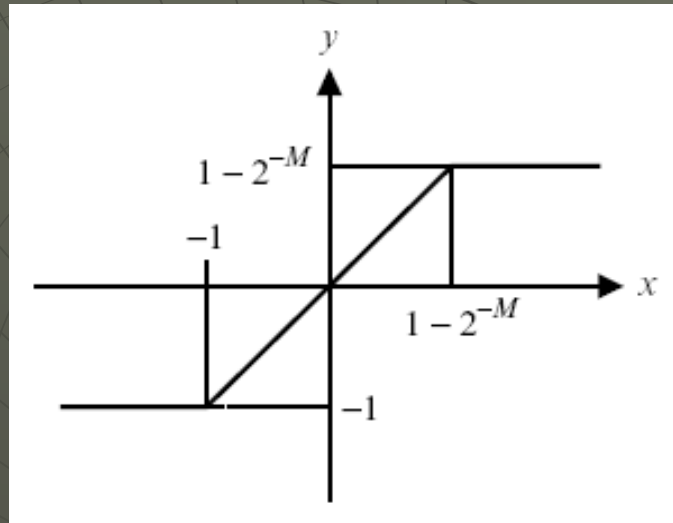
$$x_{\max} = 0\ 111\ 1111\ 1111\ 1111 = (1 - 2^{-15})_{10} \cong (0.999969)_{10}$$

$$-1 \leq x \leq 1 - 2^{-M}$$

$$\text{np. } x = 0101\ 0000\ 0000\ 0000 = \begin{cases} 2^{14} + 2^{12} = 20480 & (16.0) \\ 2^{-1} + 2^{-3} = 5/8 = 0.625 & (1.15) \end{cases}$$

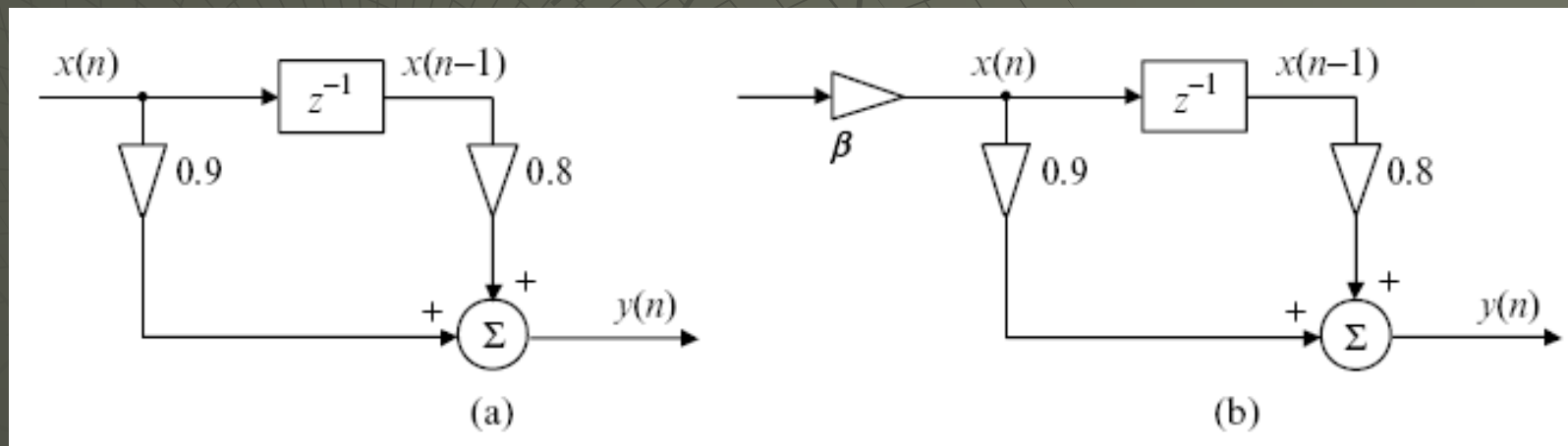
Formaty 16.0 i 1.15 są wspomagane sprzętowo w konstrukcji ALU i MAC.

Większość procesorów ma zabezpieczenie przed przepełnieniem (*overflow*) przy dodawaniu – arytmetykę z nasyceniem.



Rys. Charakterystyka arytmetyki z nasyceniem

Efektywną techniką zabezpieczenia przed przepełnieniem przy obliczeniach DSP (np. filtracji) jest *skalowanie sygnałów*.



Rys. Prosty filtr SOI: a) bez skalowania, b) ze skalowaniem sygnału przez β . W arytmetyce ułamkowej $|x(n)| \leq 1$

Dla filtra SOI o współczynnikach b_k , $k=0,1,\dots,N$, oszacowanie współczynnika skalującego jest następujące:

$$|y(n)| < 1 \quad \text{jesli} \quad \beta \leq \frac{1}{M_x \sum_{k=0}^N |b_k|} \quad \text{albo słabiej:} \quad \beta \leq \frac{1}{M_x \sqrt{\sum_{k=0}^N b_k^2}}$$

gdzie M_x jest wartością szczytową $x(n)$.

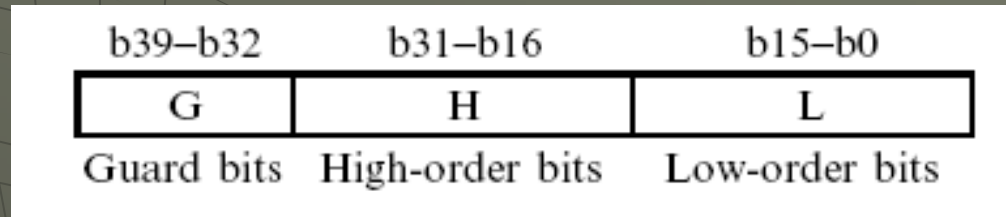
Filtry ze współczynnikami większymi od 1.0 implementuje się *skalując współczynniki*, np. przez przesunięcie kropki dziesiętnej w prawo - *shift* do formatu 2.14:

$$x = b_0 b_1 \cdot b_2 \dots b_{M-1} b_M, \quad M = 15$$

$$(x)_{10} = -b_0 \cdot 2 + \sum_{m=1}^M b_m 2^{-m+1}, \quad -2 \leq x \leq 2 - 2^{-M+1}$$

W każdym przypadku po wykonaniu obliczeń należy dokonać odpowiedniego *reskalowania* powrotnego.

Dodatkowe bity „strażnicy” w akumulatorze zabezpieczają przed przepełnieniem, np. przy długich iteracyjnych obliczeniach SOI.

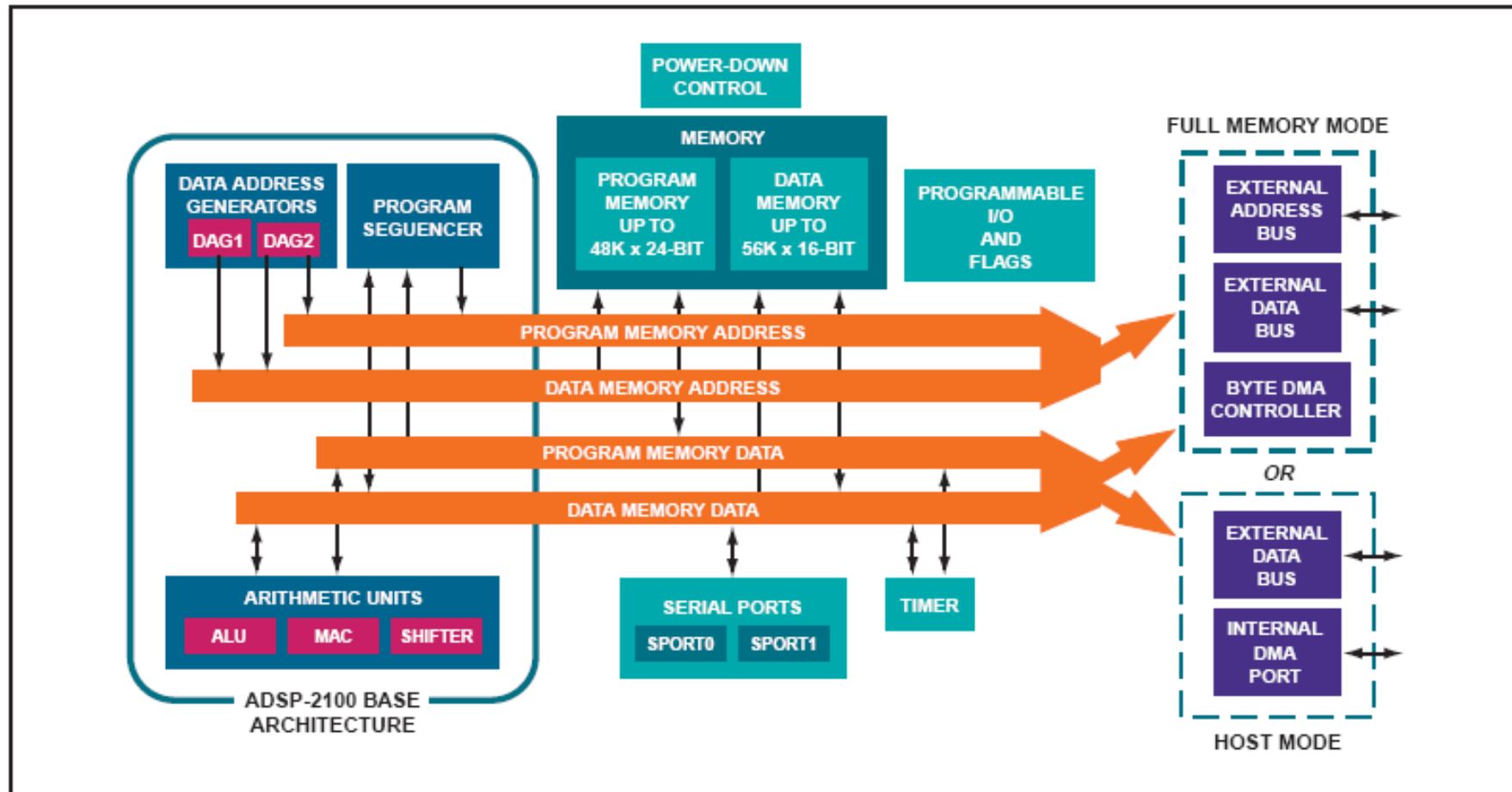


Rys. Konfiguracja 40-bitowego akumulatora MAC procesora TMS320C55x.

Mnożenie liczb w formacie M.N i P.Q daje wynik (M+P).(N+Q). Wynik 2.30 może być automatycznie przeskalowany do formatu 1.31. Bit znaku jest tracony po 256 przepełnieniach.

W procesorach ADSP-21xx: L=MR0, H=MR1, G=MR2. Mnożniki w rejestrach MX (2x16: MX0 i MX1) i MY (2x16: MY0 i MY1)

ADSP-218x Architecture Block Diagram



Rys. Architektura procesorów sygnałowych rodziny ADSP-218x firmy Analog Devices

FEATURES

PERFORMANCE

25 ns Instruction Cycle Time from 20 MHz Crystal @ 5.0 Volts
40 MIPS Sustained Performance
Single-Cycle Instruction Execution
Single-Cycle Context Switch
3-Bus Architecture Allows Dual Operand Fetches in Every Instruction Cycle
Multifunction Instructions
Power-Down Mode Featuring Low CMOS Standby Power Dissipation with 100 Cycle Recovery from Power-Down Condition
Low Power Dissipation in Idle Mode

INTEGRATION

ADSP-2100 Family Code Compatible, with Instruction Set Extensions
80K Bytes of On-Chip RAM, Configured as
16K Words On-Chip Program Memory RAM
16K Words On-Chip Data Memory RAM
Dual Purpose Program Memory for Both Instruction and Data Storage
Independent ALU, Multiplier/Accumulator, and Barrel Shifter Computational Units
Two Independent Data Address Generators
Powerful Program Sequencer Provides
Zero Overhead Looping
Conditional Instruction Execution
Programmable 16-Bit Interval Timer with Prescaler
128-Lead TQFP/128-Lead PQFP

SYSTEM INTERFACE

16-Bit Internal DMA Port for High Speed Access to On-Chip Memory
4 MByte Memory Interface for Storage of Data Tables and Program Overlays
8-Bit DMA to Byte Memory for Transparent Program and Data Memory Transfers
I/O Memory Interface with 2048 Locations Supports Parallel Peripherals
Programmable Memory Strobe and Separate I/O Memory Space Permits "Glueless" System Design
Programmable Wait State Generation
Two Double-Buffered Serial Ports with Companding Hardware and Automatic Data Buffering
Automatic Booting of On-Chip Program Memory from Byte-Wide External Memory, e.g., EPROM, or Through Internal DMA Port
Six External Interrupts
13 Programmable Flag Pins Provide Flexible System Signaling
ICE-Port™ Emulator Interface Supports Debugging in Final Systems

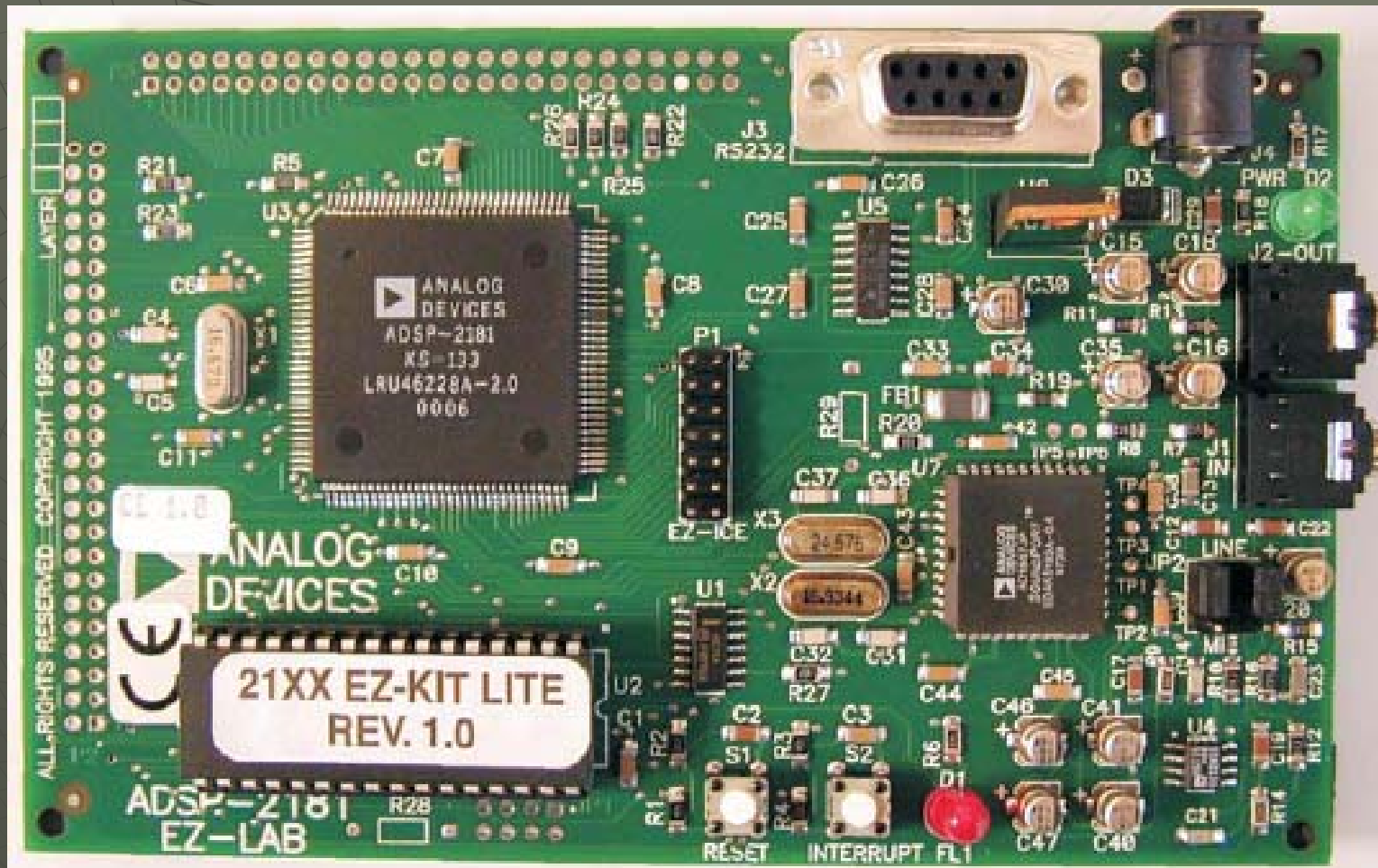
**Rys. Właściwości mikrokontrolera
ADSP-2181**

ADSP-218x DSP FAMILY SELECTION GUIDE

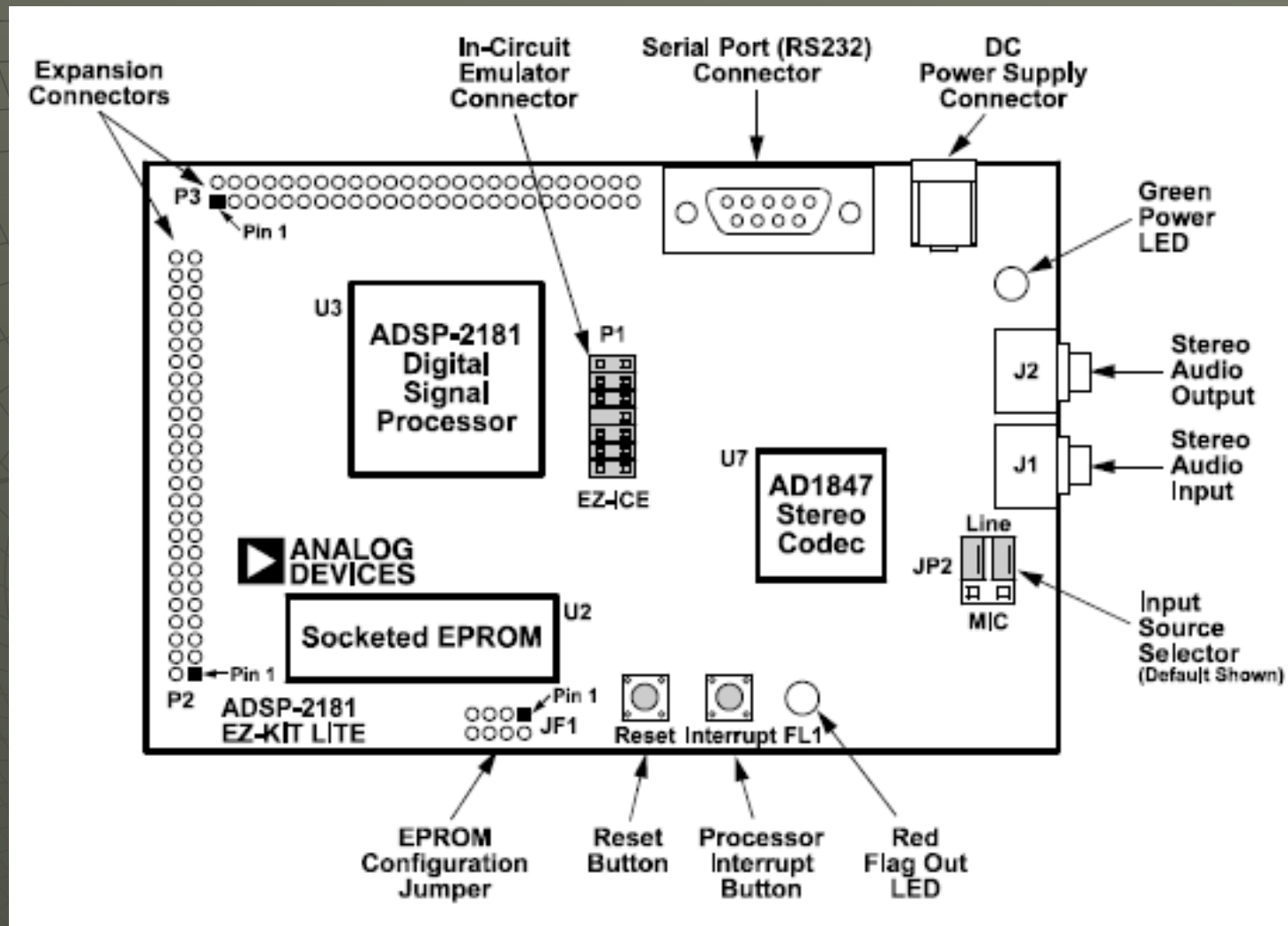
Product	Package Options	MAX MIPS/MHz	Program RAM	Data RAM	Serial Ports	Core V _{DD}	I/O V _{DD}
ADSP-2181	128-LQFP, 128-MQFP	40	16 K	16 K	2	5 V	5 V
ADSP-2183	128-LQFP, 144-MBGA	52	16 K	16 K	2	3.3 V	3.3 V
ADSP-2184	100-LQFP	40	4 K	4 K	2	5 V	5 V
ADSP-2184L	100-LQFP	40	4 K	4 K	2	3.3 V	3.3 V
ADSP-2185	100-LQFP	33	16 K	16 K	2	5 V	5 V
ADSP-2185L	100-LQFP, 144-MBGA	52	16 K	16 K	2	3.3 V	3.3 V
ADSP-2185M	100-LQFP, 144-MBGA	75	16 K	16 K	2	2.5 V	2.5 V–3.3 V
ADSP-2185N	100-LQFP, 144-MBGA	80	16 K	16 K	2	1.8 V	1.8 V–3.3 V
ADSP-2186	100-LQFP, 144-MBGA	40	8 K	8 K	2	5 V	5 V
ADSP-2186L	100-LQFP, 144-MBGA	40	8 K	8 K	2	3.3 V	3.3 V
ADSP-2186M	100-LQFP, 144-MBGA	75	8 K	8 K	2	2.5 V	2.5 V–3.3 V
ADSP-2186N	100-LQFP, 144-MBGA	80	8 K	8 K	2	1.8 V	1.8 V–3.3 V
ADSP-2187L	100-LQFP	52	32 K	32 K	2	3.3 V	3.3 V
ADSP-2187N	100-LQFP, 144-MBGA	80	32 K	32 K	2	1.8 V	1.8 V–3.3 V
ADSP-2188M	100-LQFP, 144-MBGA	75	48 K	56 K	2	2.5 V	2.5 V–3.3 V
ADSP-2188N	100-LQFP, 144-MBGA	80	48 K	56 K	2	1.8 V	1.8 V–3.3 V
ADSP-2189M	100-LQFP, 144-MBGA	75	32 K	48 K	2	2.5 V	2.5 V–3.3 V
ADSP-2189N	100-LQFP, 144-MBGA	80	32 K	48 K	2	1.8 V	1.8 V–3.3 V

Algorithm	Approximate MIPS Required	Approximate RAM Required
V.32bis or V.32terbo FAX/Modem	16	16K words PM, 16K words DM
20 Voice Music Synthesis	20	6K words PM, 4K words DM
Full Duplex Speaker Phone	13	2K words PM, 2K words DM
Digital Answering Machine	8	5K words PM, 1K word DM
GSM, 13 kbps Compression/Decompression	4	2K words PM, 1K word DM
CELP, 4.3 or 7.5 kbps Compression/Decompression	12.5	1.5K words PM, 1.5K words DM
MPEG Layer2, 64 kbps Compression	16	5K words PM, 4K words DM
MPEG Layer2, 64 kbps Decompression	10	1K words PM, 8.5K words DM
Dolby AC-2, 117 kbps Compression/Decompression	13	4K words PM, 8.5K words DM

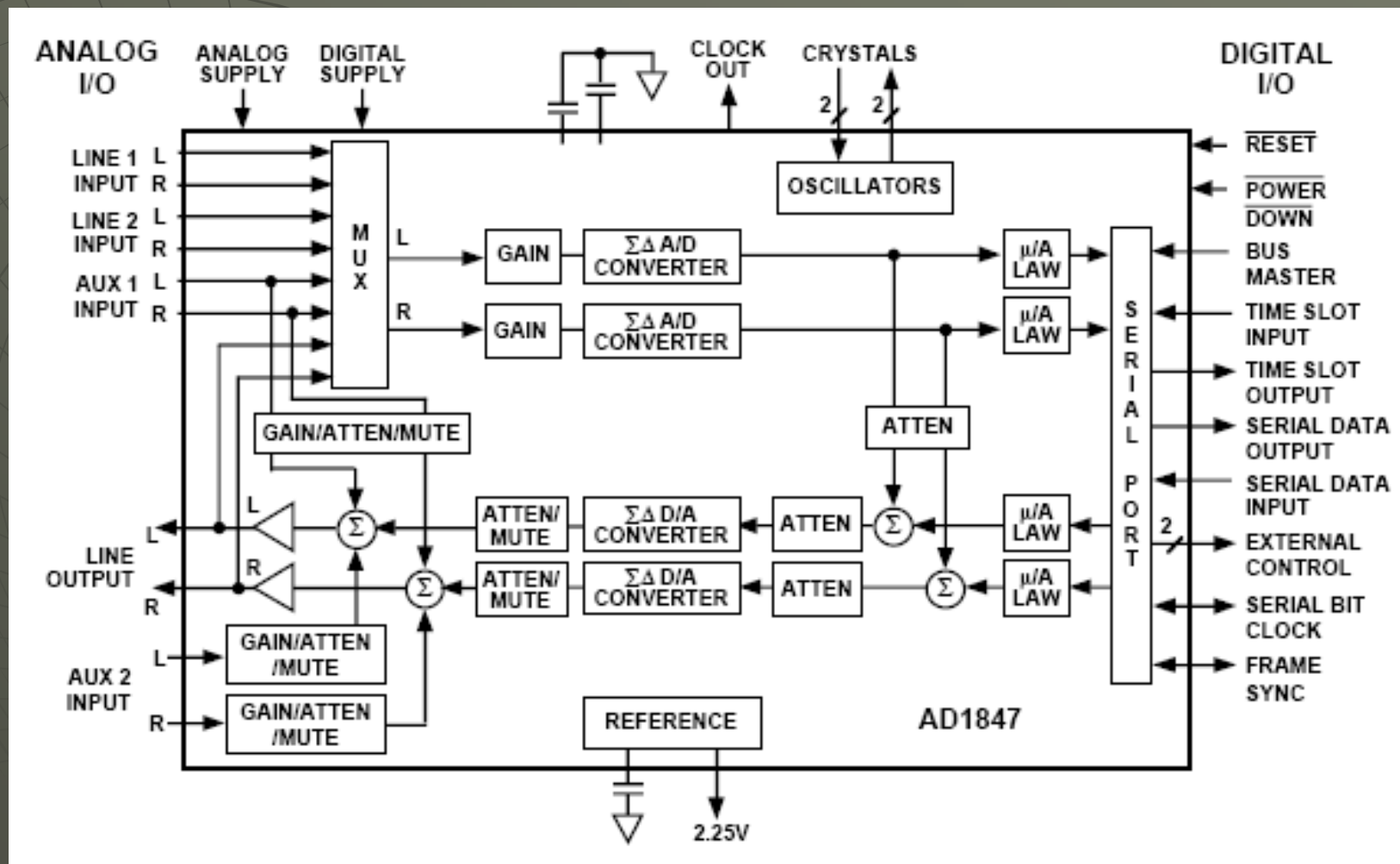
Tabela. Typowe algorytmy DSP z orientacyjnymi wymaganiami dla procesorów stałoprzecinkowych dotyczącymi zapotrzebowania na moc obliczeniową (w MIPS) i pamięć (programu PM i danych DM)



Rys. Płytką uruchomieniową EZ-KIT Lite z procesorem ADSP-2181



Rys. Rozkład elementów na płycie ADSP-2181 EZ-KIT Lite



Rys. Schemat funkcjonalny kodeka stereo AD1847

FEATURES

Single-Chip Integrated $\Sigma\Delta$ Digital Audio Stereo Codec

Supports the Microsoft Windows Sound System*

Multiple Channels of Stereo Input

Analog and Digital Signal Mixing

Programmable Gain and Attenuation

On-Chip Signal Filters

Digital Interpolation and Decimation

Analog Output Low-Pass

Sample Rates from 5.5 kHz to 48 kHz

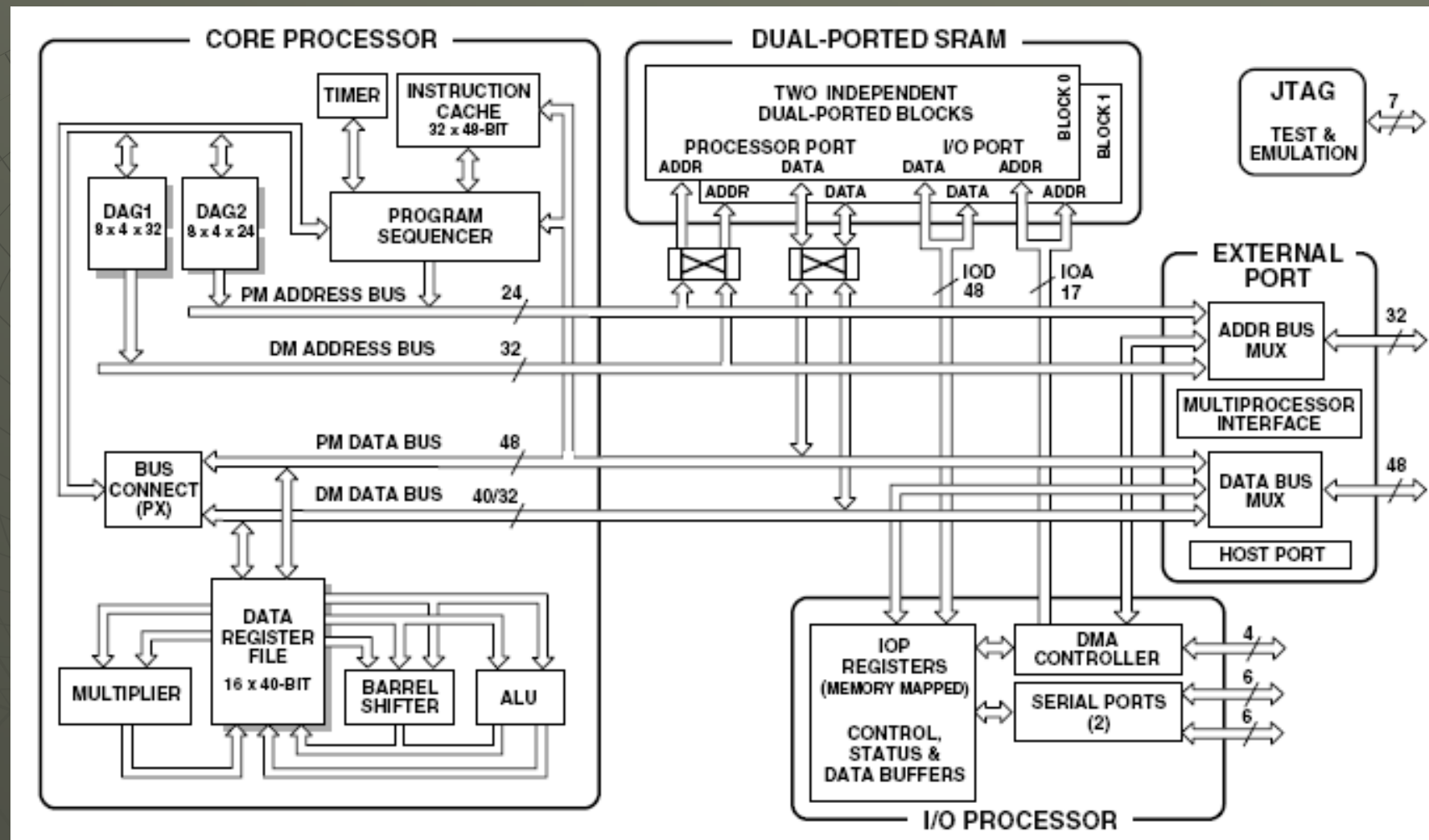
44-Lead PLCC and TQFP Packages

Operation from +5 V Supplies

Serial Digital Interface Compatible with ADSP-21xx

Fixed-Point DSP

Rys. Właściwości kodeka AD1847



Rys. Architektura zmiennoprzecinkowych procesorów sygnałowych ADSP-2106x SHARC firmy Analog Devices

SUMMARY

High Performance Signal Computer for Speech, Sound, Graphics and Imaging Applications

Super Harvard Architecture Computer (SHARC)–

Four Independent Buses for Dual Data, Instructions, and I/O

32-Bit IEEE Floating-Point Computation Units–

Multiplier, ALU and Shifter

1 Megabit On-Chip SRAM Memory and Integrated I/O

Peripherals – A Complete System-On-A-Chip

Integrated Multiprocessing Features

KEY FEATURES

50 MIPS, 20 ns Instruction Rate, Single-Cycle Instruction Execution

120 MFLOPS Peak, 80 MFLOPS Sustained Performance

Dual Data Address Generators with Modulo and Bit-Reverse Addressing

Efficient Program Sequencing with Zero-Overhead

Looping: Single-Cycle Loop Setup

IEEE JTAG Standard 1149.1 Test Access Port and

On-Chip Emulation

240-Lead MQFP Package

225-Ball Plastic Ball Grid Array (PBGA)

Pin-Compatible with ADSP-21060 (4 Mbit) and ADSP-21062 (2 Mbit)

Flexible Data Formats and 40-Bit Extended Precision

32-Bit Single-Precision and 40-Bit Extended-Precision

IEEE Floating-Point Data Formats

32-Bit Fixed-Point Data Format, Integer and Fractional, with 80-Bit Accumulators

Parallel Computations

Single-Cycle Multiply and ALU Operations in Parallel with

Dual Memory Read/Writes and Instruction Fetch

Multiply with Add and Subtract for Accelerated FFT

Butterfly Computation

1024-Point Complex FFT Benchmark: 0.37 ms (18,221 Cycles)

1 Megabit Configurable On-Chip SRAM

Dual-Ported for Independent Access by Core Processor and DMA

Configurable as 32K Words Data Memory (32-Bit), 16K

Words Program Memory (48-Bit) or Combinations of

Both Up to 1 Mbit

Off-Chip Memory Interfacing

4-Gigawords Addressable (32-Bit Address)

Programmable Wait State Generation, Page-Mode DRAM Support

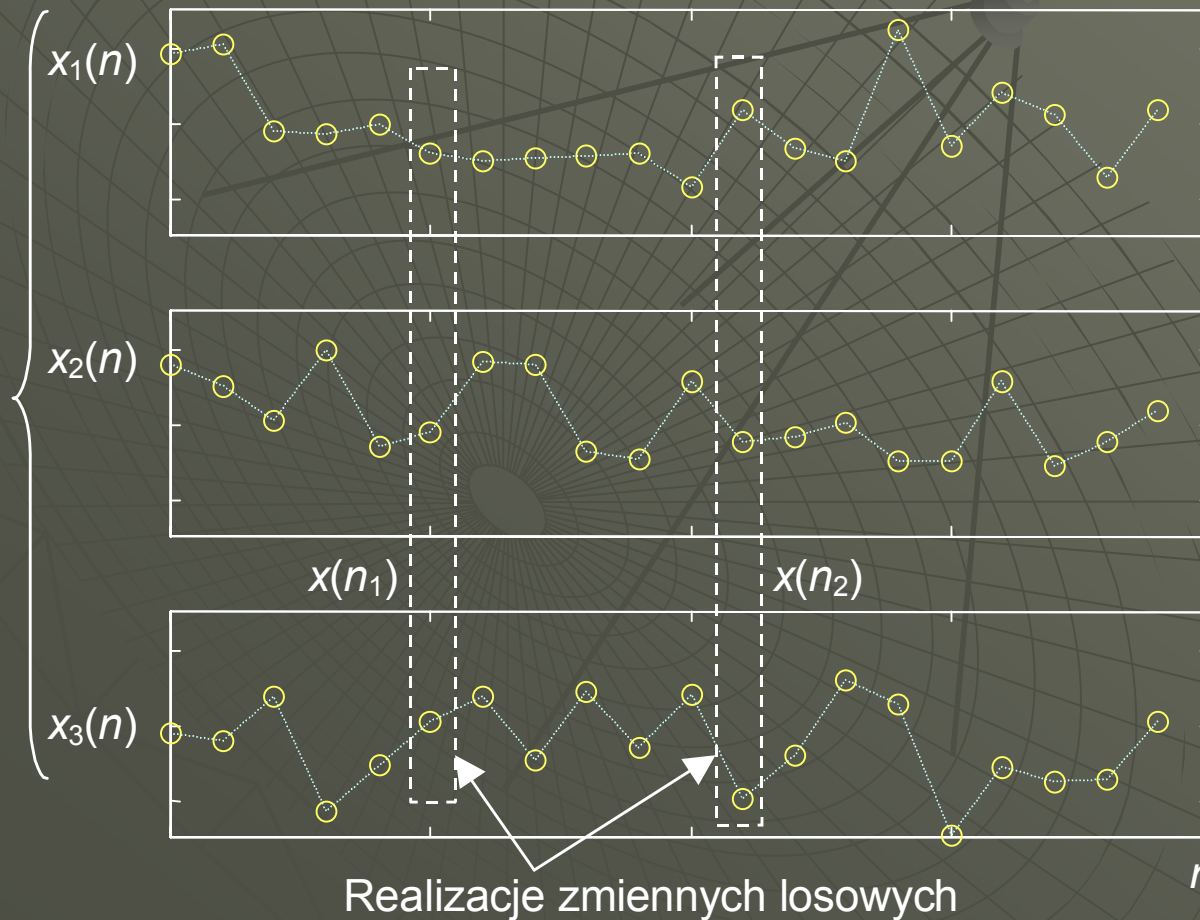
Rys. Właściwości mikrokontrolera ADSP-21061



6. *Sygnały losowe*

Statystyka – zajmuje się interpretowaniem *danych numerycznych*, np. zarejestrowanych sygnałów (tj. realizacji procesów losowych).
Prawdopodobieństwo – w DSP jest stosowane do *teoretycznego opisu* i zrozumienia procesów losowych generujących sygnały.

Realizacje procesu $X(n)$



Rys. Realizacje dyskretnego procesu losowego (stochastycznego) i zmiennych losowych w chwilach n_1 i n_2

Statystyki sygnału losowego w dziedzinie czasu

Najważniejsze wielkości charakteryzujące dyskretny proces losowy $x(n)$ o wartościach rzeczywistych. Wzory są podane dla procesów o wartościach ciągłych x oraz dyskretnych (skwantowanych) x_i , $i=1,2,\dots,M$; $p_x(x,n)$ – gęstość prawdopodobieństwa (*Probability Density Function – PDF*) w chwili n .

- **Wartość średnia** (oczekiwana) sygnału:

$$m_x(n) = E[x(n)] = \int_{-\infty}^{\infty} x p_x(x, n) dx$$

$$m_x(n) = E[x(n)] = \sum_i x_i p_x(x_i, n)$$

- **Średnia kwadratowa** (moc średnia) sygnału:

$$P_x(n) = E[x^2(n)] = \int_{-\infty}^{\infty} x^2 p_x(x, n) dx$$

$$P_x(n) = E[x^2(n)] = \sum_i x_i^2 p_x(x_i, n)$$

- **Wariancja** (moc składowej zmiennej):

$$\sigma_x^2(n) = \text{var } x(n) = E\{[x(n) - m_x(n)]^2\} = E[x^2(n)] - m_x^2(n)$$

- **Funkcja autokorelacji** zmiennych losowych $x(n_1)$ i $x(n_2)$ sygnału:

$$R_{xx}(n_1, n_2) = E[x(n_1)x(n_2)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(n_1)x(n_2) p_{xx}(x, n_1, n_2) dx(n_1) dx(n_2)$$

$$R_{xx}(n_1, n_2) = E[x(n_1)x(n_2)] = \sum_i \sum_j x_i x_j p_{xx}[x_i(n_1), x_j(n_2)]$$

(jest to prawdopodobieństwo łączne: $\text{Prob}[x(n_1)=x_i \wedge x(n_2)=x_j]$).

- **Funkcja autokowariancji** zmiennych losowych $x(n_1)$ i $x(n_2)$ sygnału:

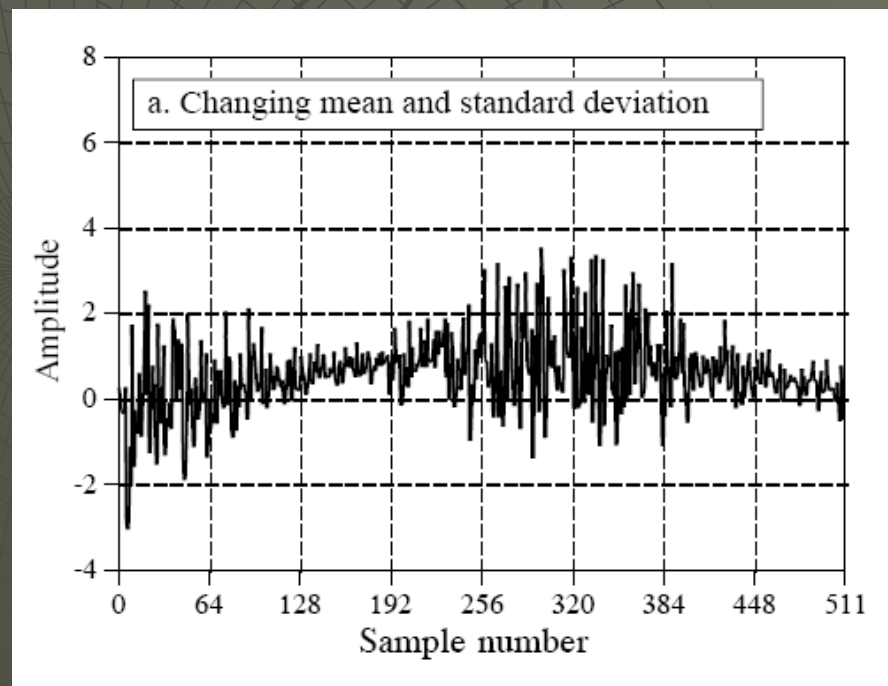
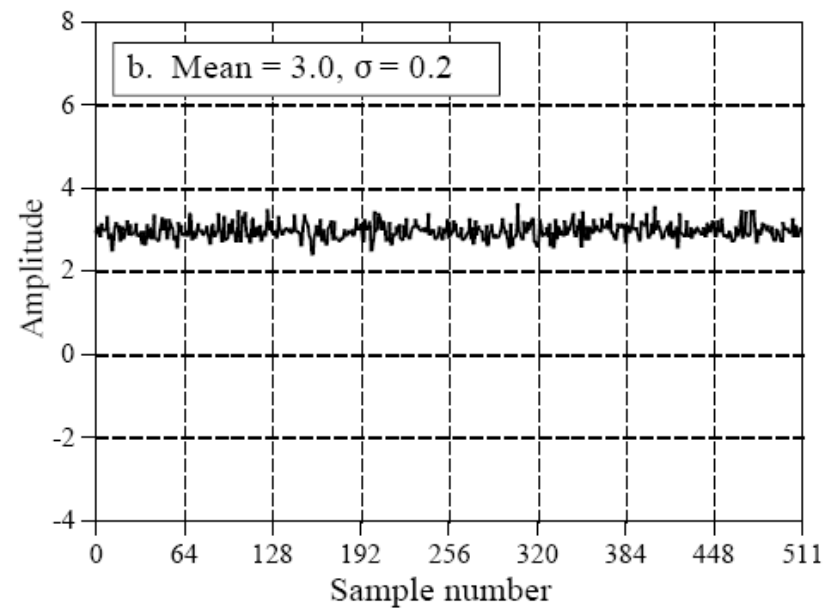
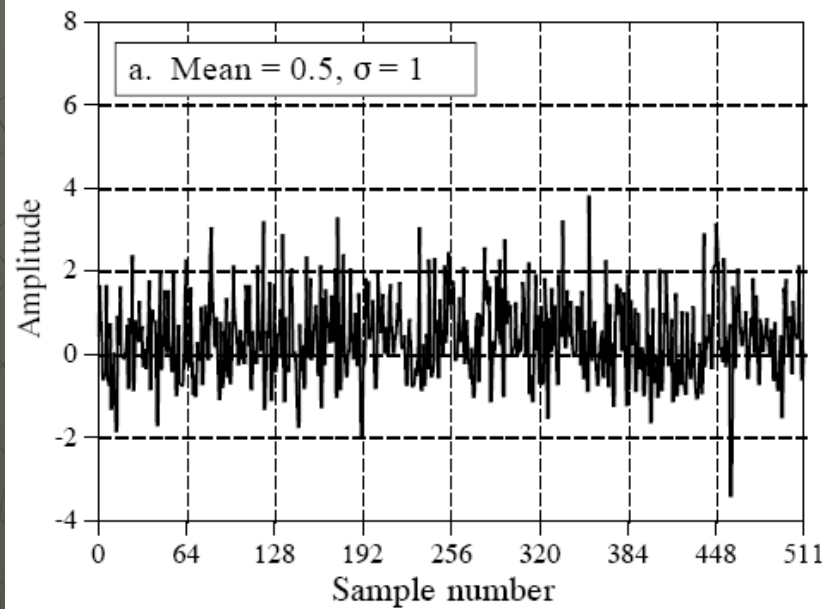
$$C_{xx}(n_1, n_2) = E\{[x(n_1) - m_x(n_1)][x(n_2) - m_x(n_2)]\}$$

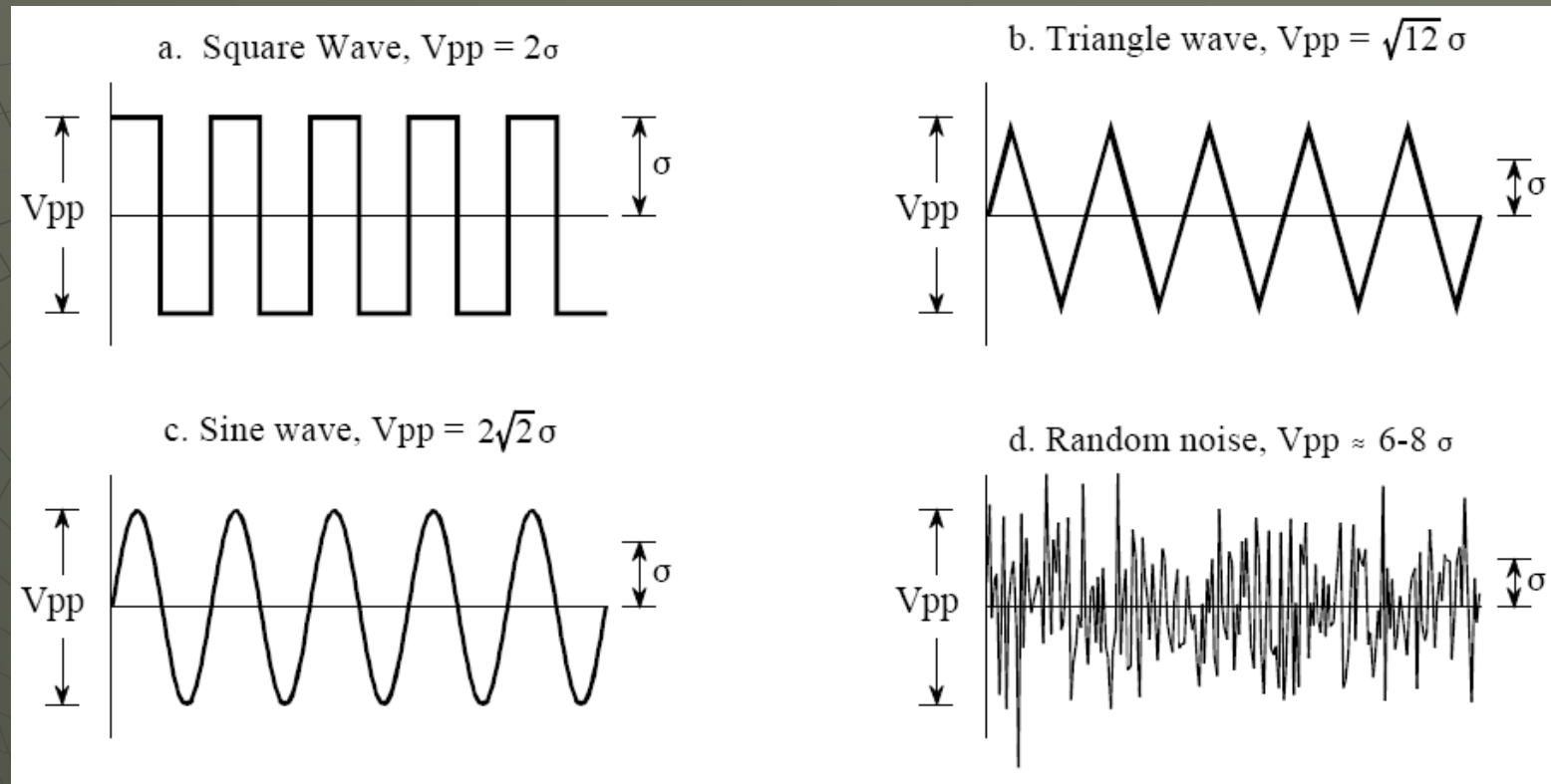
W szczególności kowariancja $C_{xx}(n, n) = \sigma_x^2(n)$

W analogiczny sposób definiuje się *funkcje korelacji i kowariancji wzajemnej* dwóch sygnałów losowych $x(n)$ i $y(n)$, np. kowariancja wzajemna:

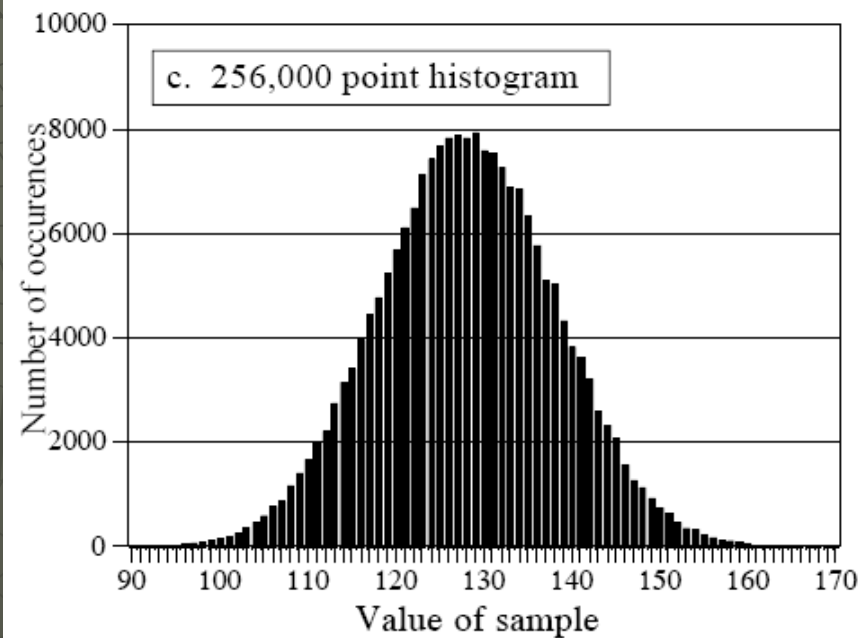
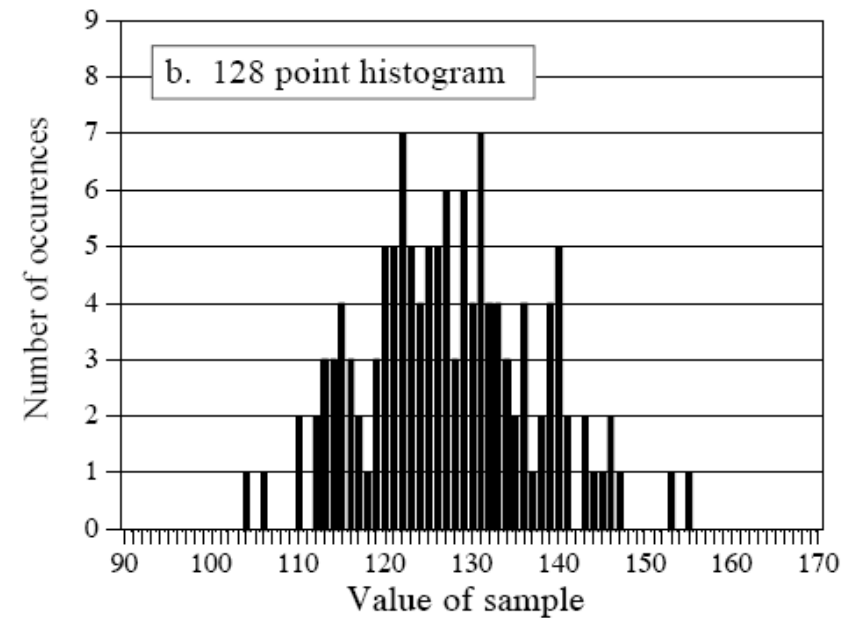
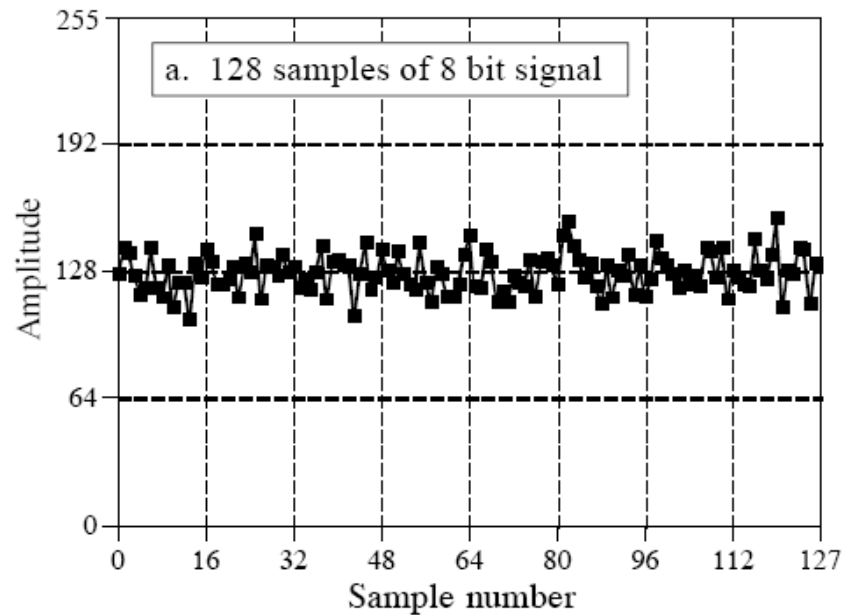
$$C_{xy}(n_1, n_2) = E\{[x(n_1) - m_x(n_1)][y(n_2) - m_y(n_2)]\}$$

Autokorelacja (korelacja wzajemna dwóch sygnałów) jest funkcją *deterministyczną* i najważniejszą wielkością charakteryzującą sygnały stochastyczne w *d dziedzinie czasu*.

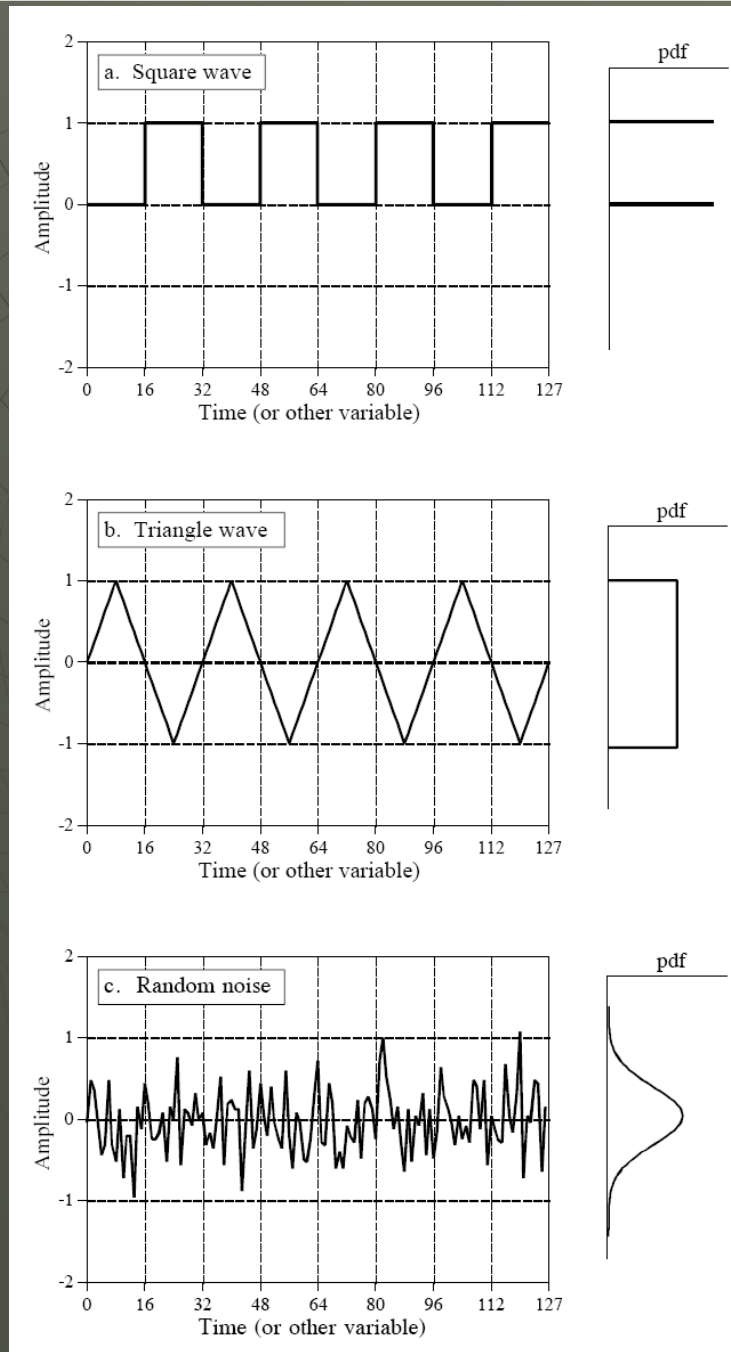




Rys. Zależność wartości międzyszczytowej (*peak-to-peak*) od odchylenia standardowego (pierwiastka z wariancji) dla typowych sygnałów. Sygnał losowy nie ma dokładnej wartości V_{pp}



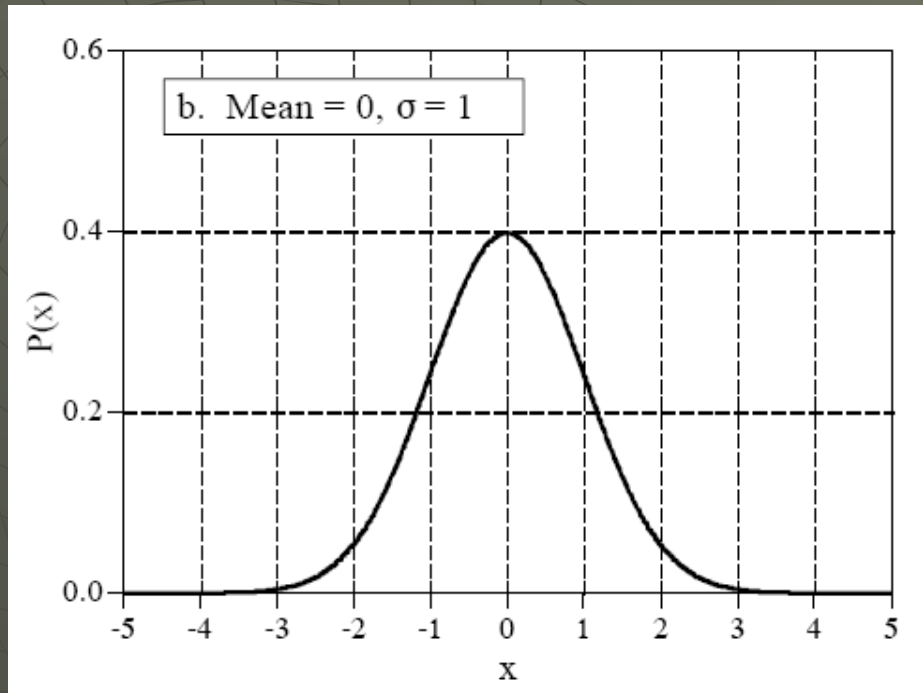
Rys. Seria próbek sygnału losowego i jej histogramy dla dwóch długości serii



Rys. Przykładowe sygnały i ich funkcje gęstości prawdopodobieństwa

Rozkład normalny (gaussowski)

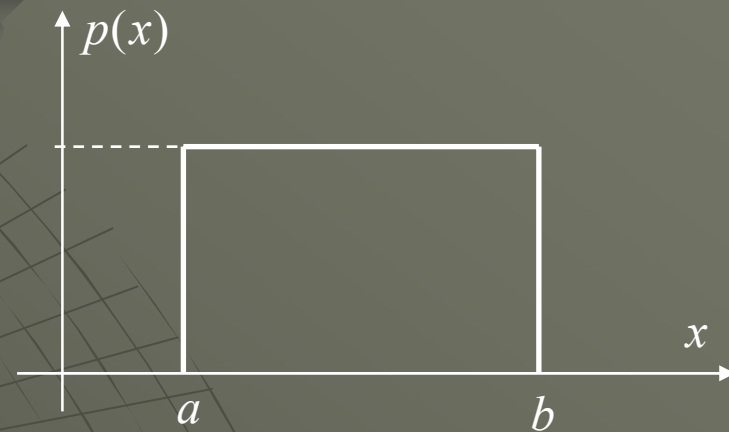
$$N(m_x, \sigma_x^2)$$



$$p(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp \frac{-(x - m_x)^2}{2\sigma_x^2}$$

Rozkład jednostajny

$$U(a, b)$$



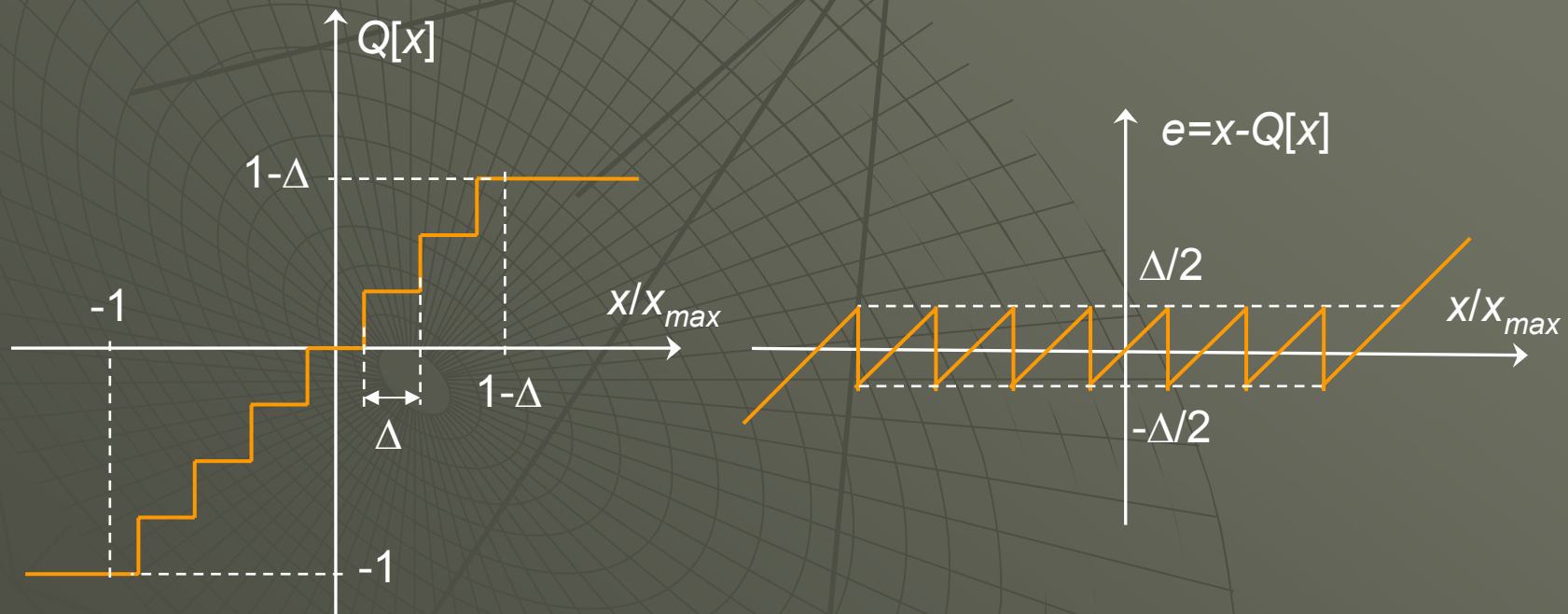
$$p(x) = \frac{1}{b-a},$$

$$m_x = \frac{b+a}{2}, \quad \text{var } x = \frac{(b-a)^2}{12}$$

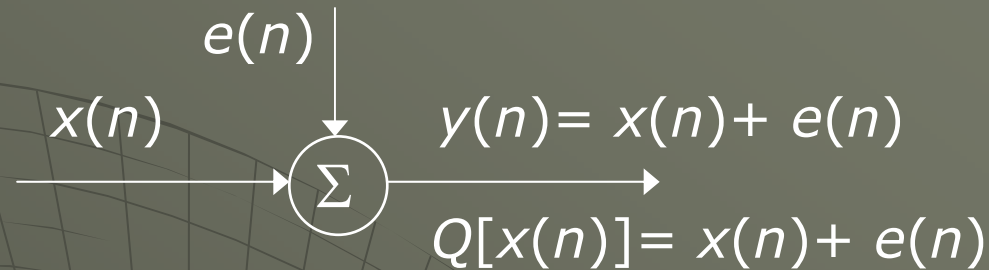
Przykład: Błędy kwantowania

Dla kwantyzatora B -bitowego z unormowanym sygnałem wejściowym o zakresie $-1 \leq x/x_{max} < 1$, krok kwantowania:

$$\Delta = \frac{\text{pełny zakres}}{\text{liczba poziomów kwantowania}} = \frac{2}{2^B} = 2^{-B+1}$$



Rys. Charakterystyka kwantyzatora $B=3$ -bitowego (8 poziomów) i charakterystyka błędu kwantowania



Rys. Modelowanie nieliniowego procesu kwantowania jako addytywnego szumu $e(n)$ o rozkładzie jednostajnym $U(-\Delta/2, \Delta/2)$

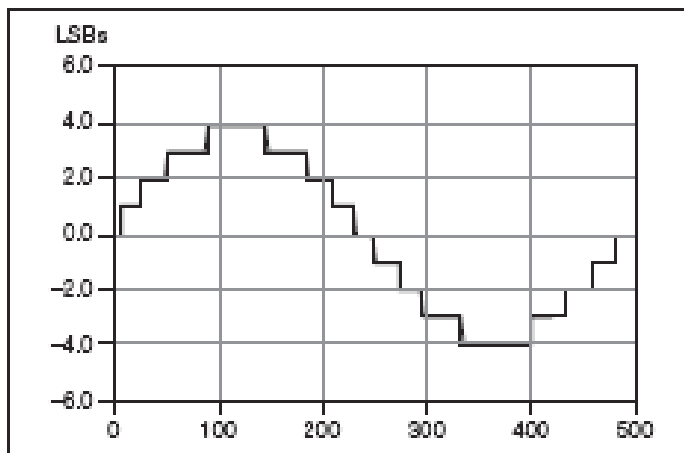
Model taki jest właściwy dla „drobnego” kwantowania (dużej liczby bitów B), kiedy można założyć, że błąd kwantowania *nie jest skorelowany* z sygnałem x . Parametry szumu:

$$m_e = 0, \quad \sigma_e^2 = \frac{\Delta^2}{12} = \frac{2^{-2B}}{3}$$

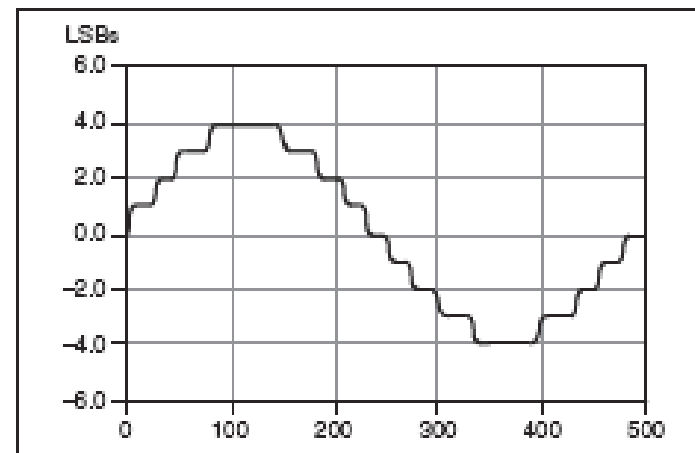
Stosunek sygnału do szumu kwantowania (*Signal-to-Noise Ratio*):

$$SNR = \frac{\sigma_x^2}{\sigma_e^2}, \quad \text{gdzie } \sigma_x^2 \leq 1 \text{ – wariacja sygnału } x(n).$$

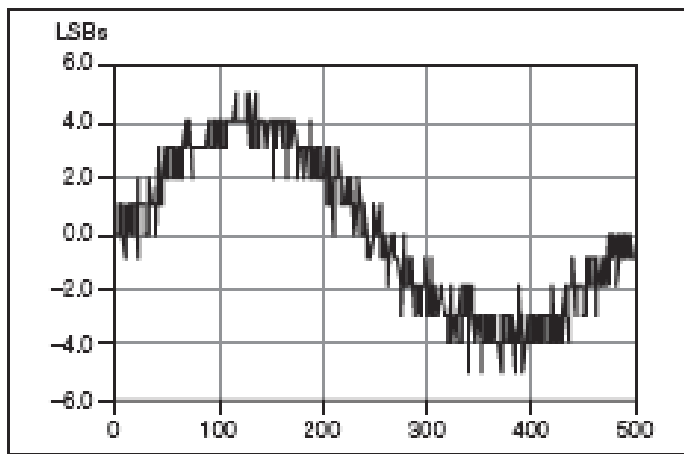
W decybelach: $SNR = 10 \log(\sigma_x^2 / \sigma_e^2) \cong 4.77 + 6.02B + 10 \log \sigma_x^2$ [dB]



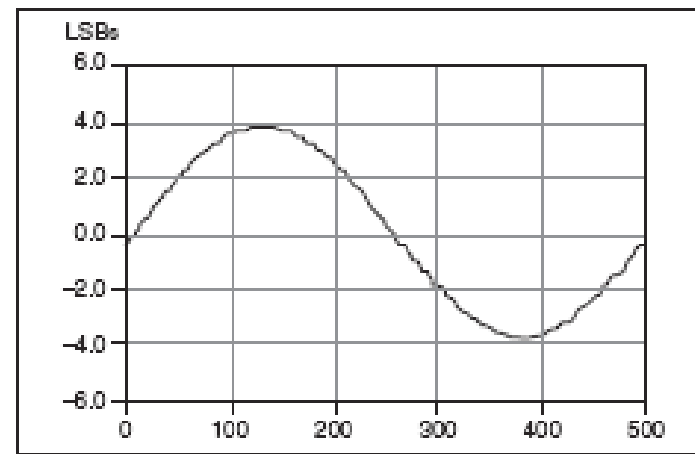
a. Dither Disabled; No Averaging



b. Dither Disabled; Average of 50 Acquisitions



c. Dither Enabled; No Averaging



d. Dither Enabled; Average of 50 Acquisitions

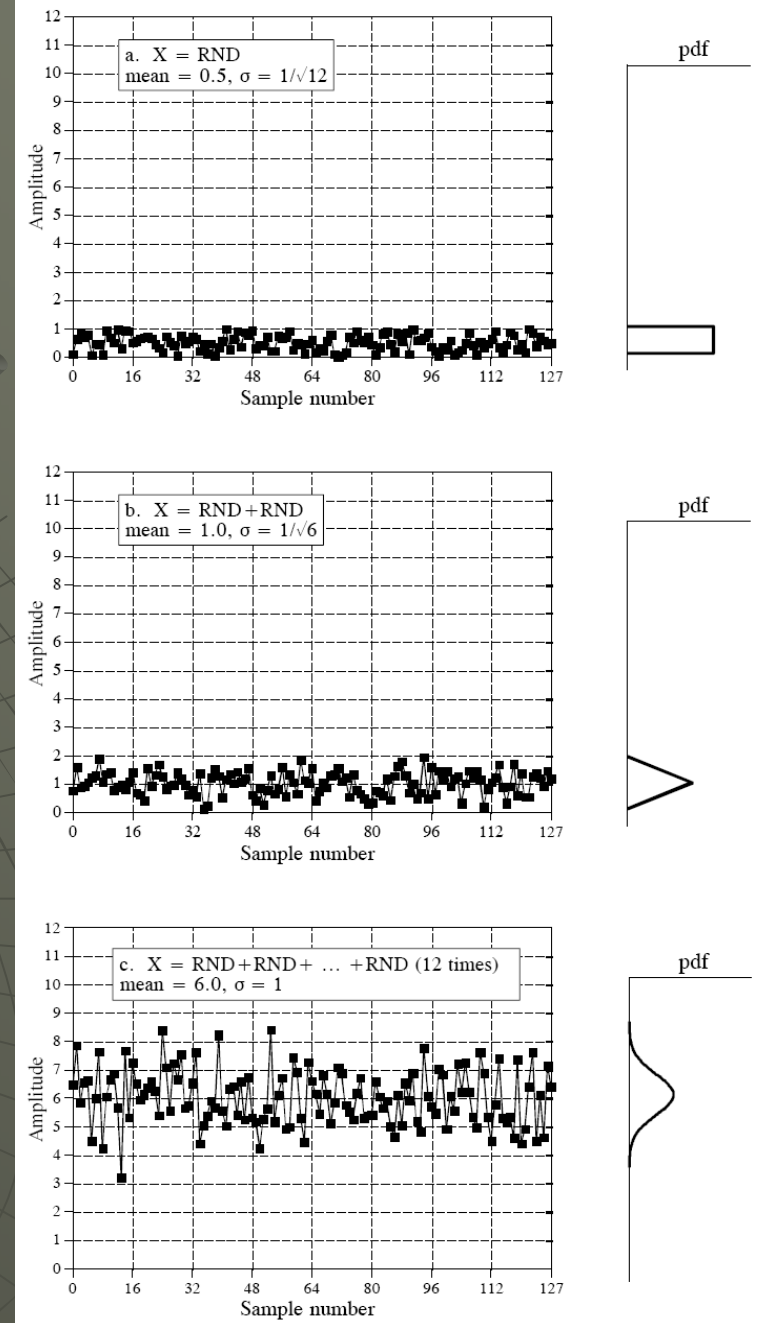
Rys. *Dithering* – rozpraszanie błędów kwantowania przez dodanie do sygnału analogowego, przetwarzanego przez przetwornik A/C, białego szumu gaussowskiego o odchyleniu standardowym $\sigma_n = 1/2 \text{LSB}_{rms}$ oraz powtarzanie i uśrednianie pomiarów.

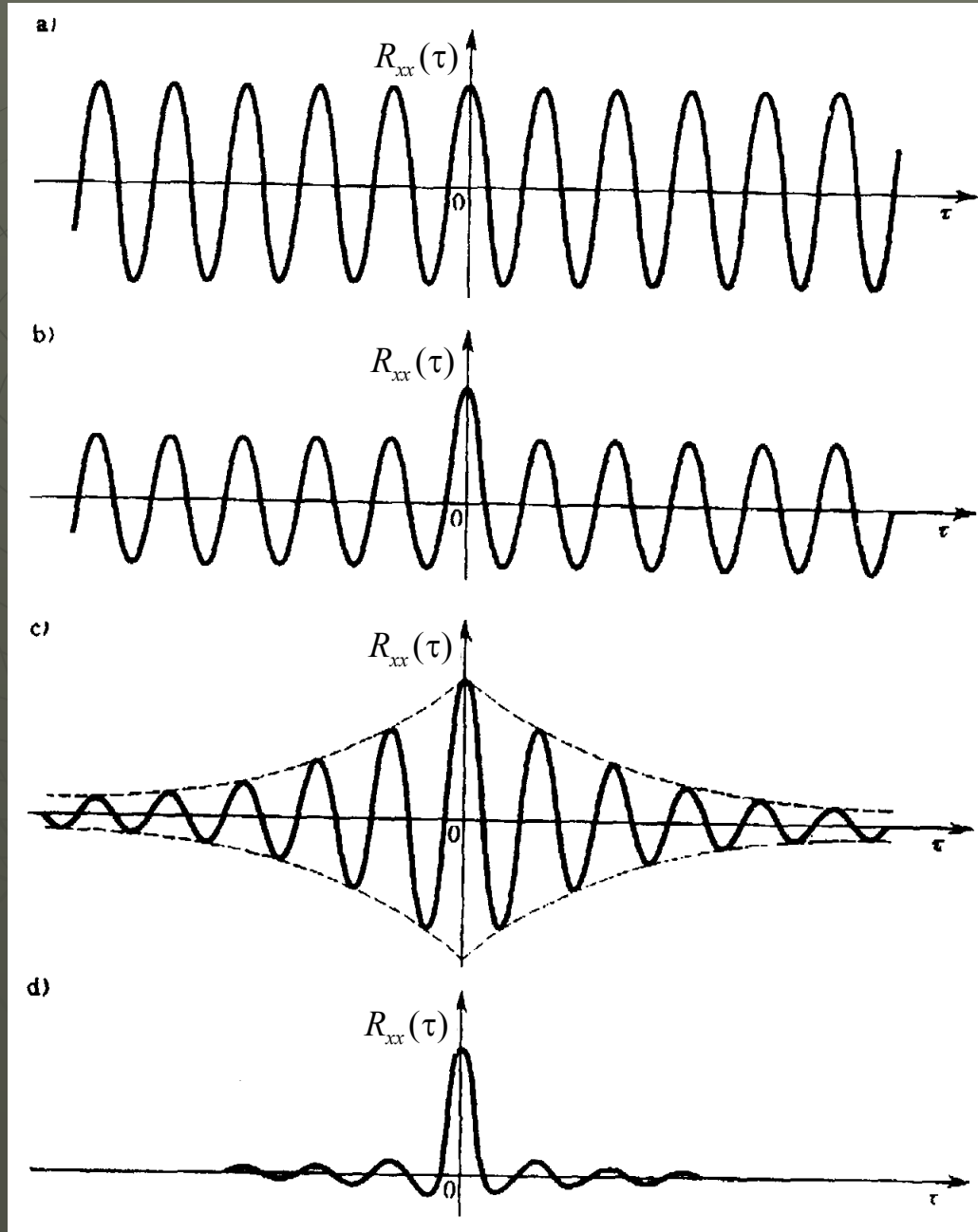
Centralne twierdzenie graniczne: Zmienna losowa

$$y = \sum_{i=1}^N \frac{x_i - m_x}{\sqrt{N\sigma_x^2}} \rightarrow \text{pdf}(y) = N(0,1)$$

gdzie zmienne losowe x_i podlegają rozkładowi o średniej m_x i wariancji σ_x^2 , ma rozkład normalny o zerowej średniej i wariancji równej 1.

Rys. Generowanie liczb o rozkładzie normalnym jako sumy liczb losowych o rozkładzie jednostajnym





Rys. Przebiegi (teoretyczne) funkcji autokorelacji sygnałów:
 a) sinusoidalnego,
 b) sinusoidalnego z szumem losowym,
 c) szumu wąskopasmowego,
 d) szumu szerokopasmowego,
 $\tau = t_2 - t_1$ - przesunięcie (*lag*)

Sygnały stacjonarne i ergodyczne

Jeżeli PDF procesu losowego nie zmieniają się w czasie, tzn.

$$p_x(x, n_1) = p_x(x, n_2) = p_x(x),$$

to proces nazywamy stacjonarnym.

Procesy stacjonarne w szerszym sensie to procesy, których prawdopodobieństwa mogą się zmieniać w czasie, ale spełnione są następujące warunki:

- wartość średnia i wariancja są *stałe*:

$$m_x(n) = m_x \quad \sigma_x^2(n) = \sigma_x^2$$

- autokorelacja *zależy tylko od odległości w czasie* $m = n_2 - n_1$ (ang. *lag*) *między zmiennymi losowymi* $x(n_1)$ i $x(n_2)$:

$$R_{xx}(n_1, n_2) = R_{xx}(n_2 - n_1) = R_{xx}(m)$$

Dla sygnału o zerowej średniej $m_x = 0$ wariancja $\sigma_x^2 = R_{xx}(0)$

Dla każdego $m \neq 0$: $|R_{xx}(m)| \leq R_{xx}(0)$ oraz $R_{xx}(m) = R_{xx}(-m)$.

W praktyce DSP zamiast wyznaczania podanych zespołowych statystyk, co wymaga *uśredniania po realizacjach* procesu $\{x_k(n)\}$, $k=1,2,\dots$, o właściwościach procesu wnioskuje się na podstawie statystyk czasowych, tzn. stosuje się *uśrednianie po czasie* dla pojedynczej zaobserwowanej realizacji procesu. Wymaga to przyjęcia założenia, że proces jest ergodyczny, tzn. taki, że *średnie po realizacjach* są dla niego równe *średnim czasowym*.

Średnie czasowe $\langle \cdot \rangle$ stacjonarnego sygnału losowego $x(n)$:

$$\langle x(n) \rangle = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N x(n)$$

$$\langle R_{xx}(m) \rangle = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N x(n)x(n+m)$$

$$\langle x(n) \rangle = m_x = \text{const}$$

$$\langle R_{xx}(m) \rangle = R_{xx}(m), \quad m = n_2 - n_1$$

W obliczeniach DSP oblicza się średnie czasowe *skończonego ciągu próbek* zarejestrowanej realizacji jako *estymaty* teoretycznych zespołowych statystyk sygnału.

Widmo mocy sygnału losowego (reprezentacja sygnału losowego w dziedzinie częstotliwości)

Odległe w czasie zmienne losowe sygnału dyskretnego $x(n)$ stają się od siebie niezależne i kowariancja sygnału zanika : $\lim_{m \rightarrow \infty} C_{xx}(m) = 0$

Można wtedy wyznaczyć jej transformatę Z, której obszar zbieżności zawiera okrąg jednostkowy $|z|=1$ i istnieje odpowiednia transformata Fouriera dla $z=e^{j\Omega}$.

Widmowa gęstość mocy (widmo mocy) $P_{xx}(\Omega)$ jest transformatą Fouriera kowariancji $C_{xx}(m)$ sygnału:

$$P_{xx}(\Omega) = \sum_{m=-\infty}^{\infty} C_{xx}(m) e^{-jm\Omega}, \quad \Omega = \frac{\omega}{f_s}$$

Widmo mocy jest najważniejszą wielkością opisującą sygnał losowy w dziedzinie częstotliwości.

Gęstość mocy $P_{xx}(\Omega)$ określa *moc sygnału przypadającą na jednostkę pulsacji* Ω .

Właściwości widma mocy:

- $P_{xx}(\Omega)$ jest funkcją okresową o okresie 2π ,
- jeżeli $x(n)$ jest sygnałem o wartościach rzeczywistych, to widmo mocy jest funkcją *nieujemną i parzystą*: $P_{xx}(\Omega) = P_{xx}(-\Omega)$,
- całka oznaczona z widma mocy sygnału o zerowej średniej ($m_x = 0$):

$$\int_{\Omega_1}^{\Omega_2} P_{xx}(\Omega) d\Omega$$

określa moc średnią moc sygnału w zakresie pulsacji $\Omega_1 \leq \Omega \leq \Omega_2$

Całka obliczona w całym zakresie częstotliwości określa całkowitą moc sygnału.

Znając widmo mocy można określić wartości autokowariancji jako współczynniki rozwinięcia widma w szereg Fouriera:

$$C_{xx}(m) = \frac{1}{2\pi} \int_{-\pi}^{\pi} P_{xx}(\Omega) e^{jm\Omega} d\Omega$$

W szczególności: $\sigma_x^2 = C_{xx}(0) = \frac{1}{2\pi} \int_{-\pi}^{\pi} P_{xx}(\Omega) d\Omega$

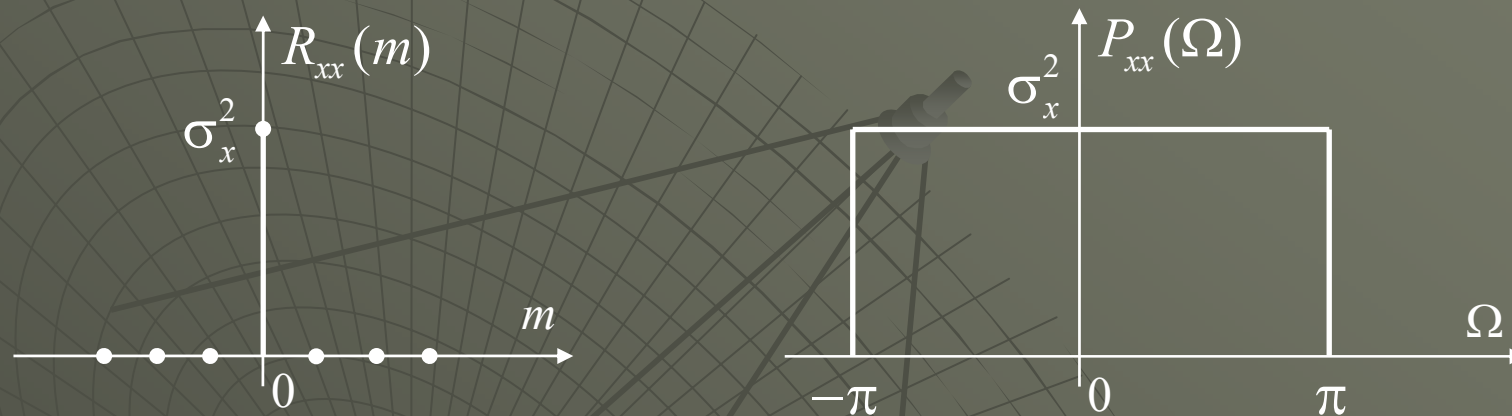
Widmo mocy wzajemnej dwóch sygnałów:

$$P_{xy}(\Omega) = \sum_{m=-\infty}^{\infty} C_{xy}(m) e^{-jm\Omega}$$

Dla sygnałów o zerowej wartości średniej $m_x=0$ ($m_y=0$) autokowariancję $C_{xx}(m)$ (kowariancję wzajemną $C_{xy}(m)$) można zastąpić równoważną autokorelacją $R_{xx}(m)$ (korelacją wzajemną $R_{xy}(m)$).

Szum biały

Szum biały jest wyidealizowanym modelem sygnału losowego, którego próbki są nie są ze sobą skorelowane. Widmo mocy szumu białego ma w całym zakresie częstotliwości wartość stałą.



$$R_{xx}(m) = \begin{cases} \sigma_x^2 & \text{dla } m = 0 \\ 0 & \text{dla } m \neq 0 \end{cases} \quad P_{xx}(\Omega) = \sigma_x^2 \quad \text{dla } -\pi \leq \Omega \leq \pi$$

Szum biały odgrywa szczególną rolę jako sygnał wejściowy ujawniający dynamikę układu w całym zakresie częstotliwości (podobnie jak impuls jednostkowy w przypadku sygnałów deterministycznych).

Najczęściej stosuje się szum biały o rozkładzie jednostajnym lub normalnym.

Wskaźniki jakości estymatorów

Przypuśćmy, że pewien (nieznany) parametr α procesu losowego oceniamy wyznaczając jego estymator $\hat{\alpha}$.

Jakość estymatora ocenia się na podstawie oszacowania jego:

- **obciążenia** (ang. *bias*): $B(\hat{\alpha}) = \alpha - E[\hat{\alpha}]$
- **wariancji**: $\text{var } \hat{\alpha} = \sigma_{\hat{\alpha}}^2 = E[(\hat{\alpha} - E[\hat{\alpha}])^2]$

Obciążenie jest odchyleniem wartości oczekiwanej estymatora od rzeczywistej wartości parametru, a wariancja - miarą skupienia wartości estymatora wokół przeciętnej $E[\hat{\alpha}]$.

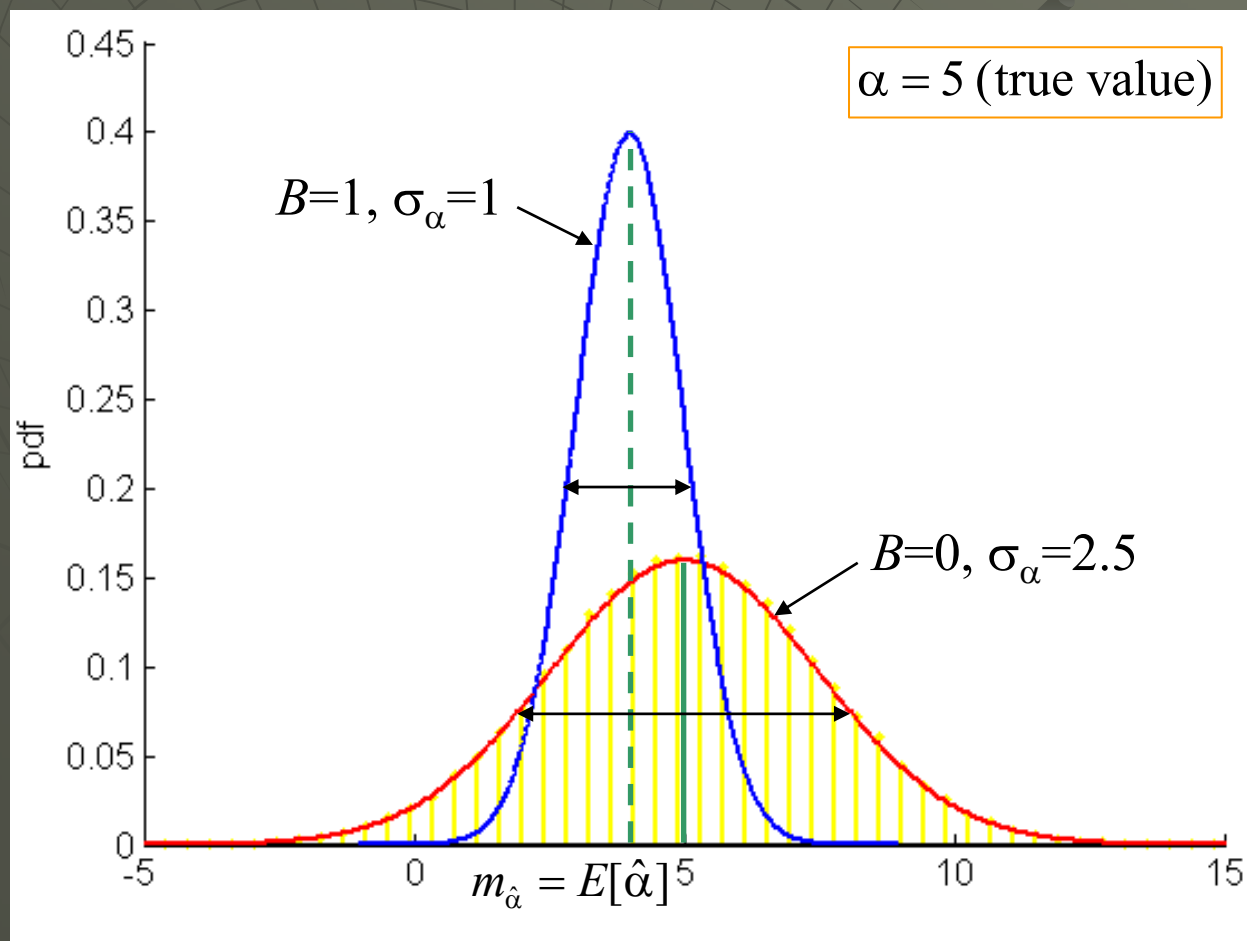
Estymator powinien mieć jak najmniejsze obciążenie i wariancję.

Często estymator o małym obciążeniu ma dużą wariancję lub na odwrót. Z tego powodu jakość estymatora ocenia się podając **błąd średniokwadratowy** obejmujący sumę tych wskaźników:

$$E[(\hat{\alpha} - \alpha)^2] = B^2(\hat{\alpha}) + \sigma_{\hat{\alpha}}^2$$

Wymaga się, aby estymator był *zgodny* (*consistent*), tzn. zbieżny do wartości rzeczywistej estymowanego parametru przy liczbie próbek $N \rightarrow \infty$ (obciążenie i wariancja dążące do 0):

$$\lim_{N \rightarrow \infty} Prob(\hat{\alpha}_N = \alpha) = 1$$



Rys. Rozkład prawdopodobieństwa z próbek dla dwóch estymatorów α :

- 1) nieobciążonego o dużej wariancji,
- 2) obciążonego o małej wariancji

Estymatory statystyk sygnału losowego z próbek

W praktyce właściwości procesu losowego $X(n)$ trzeba oceniać na podstawie obserwacji tylko jednej jego realizacji $x(n)$ w skończonym przedziale czasu $n=0,1,\dots,N-1$, czyli na podstawie ciągu N próbek. Wyznamy estymatory statystyk sygnału, zakładając, że analizowany sygnał jest stacjonarny (w szerszym sensie) i ergodyczny.

- *estymator wartości średniej* (największej wiarygodności):

$$\hat{m}_x = \frac{1}{N} \sum_{n=0}^{N-1} x(n)$$

Estymator ten jest nieobciążony i zgodny o wariancji:

$$B(\hat{m}_x) = 0 \quad \text{var } \hat{m}_x = \frac{\sigma_x^2}{N}$$

- *estymator* (największej wiarygodności) *wariancji* sygnału o nieznannej wartości średniej:

$$\hat{\sigma}_x^2 = \frac{1}{N} \sum_{n=0}^{N-1} [x(n) - \hat{m}_x]^2$$

Estymator ten jest obciążony i zgodny:

$$B(\hat{\sigma}_x^2) = \frac{N-1}{N} \sigma_x^2 \quad \text{var } \hat{\sigma}_x^2 \sim \frac{1}{N}$$

Nieobciążony estymator (o większej wariancji):

$$\hat{\sigma}_x^2 = \frac{1}{N-1} \sum_{n=0}^{N-1} [x(n) - \hat{m}_x]^2$$

- estymator funkcji autokorelacji (również autokowariancji w przypadku sygnału o zerowej wartości średniej, $m_x=0$):

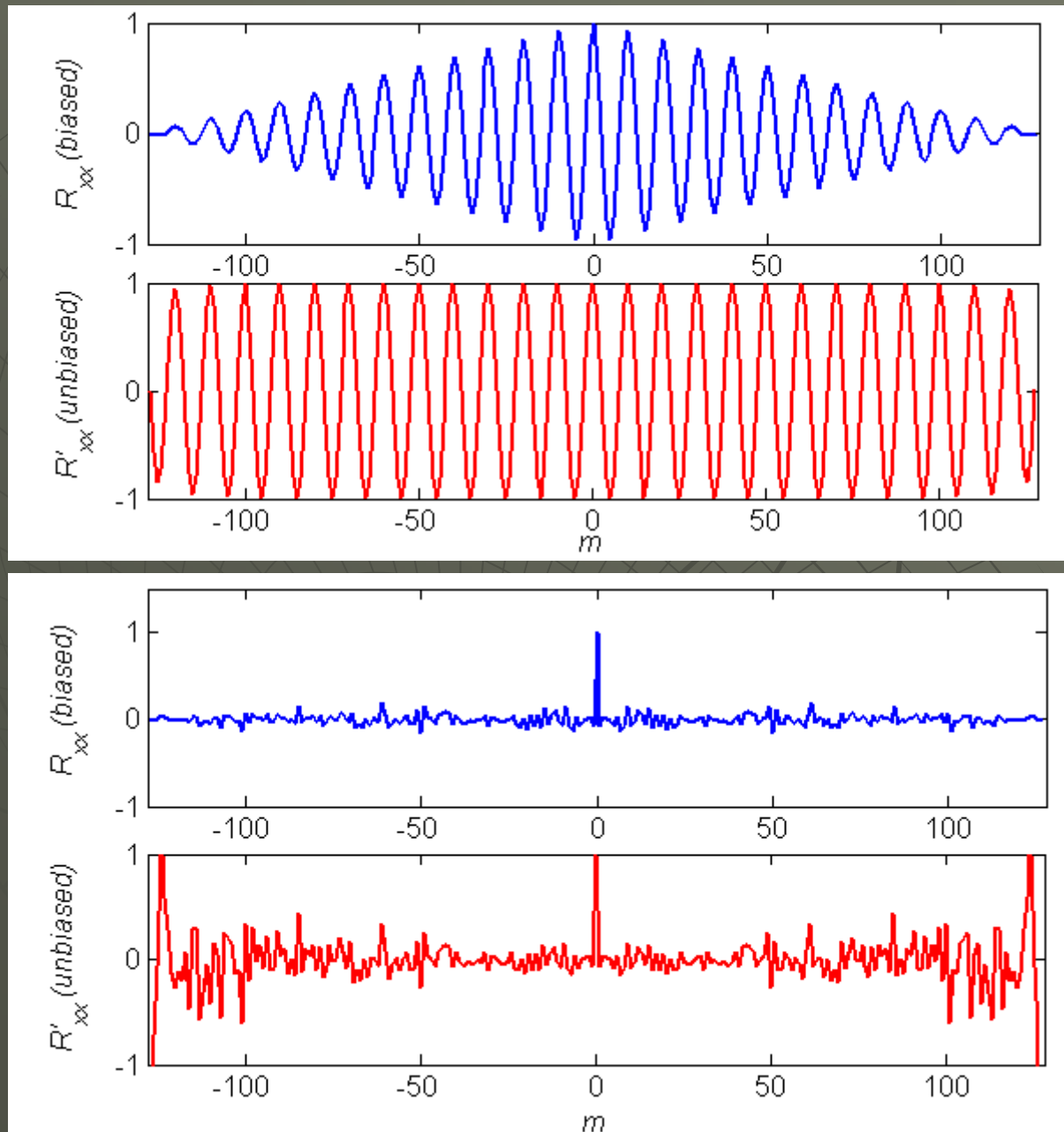
$$\hat{R}_{xx}(m) = \frac{1}{N} \sum_{n=0}^{N-|m|-1} x(n)x(n+m), \quad -(N-1) \leq m \leq N-1$$

Estymator ten jest obciążony i zgodny:

$$B[\hat{R}_{xx}(m)] = \frac{|m|}{N} R_{xx}(m) \quad \text{var } \hat{R}_{xx}(m) \sim \frac{R_{xx}^2(m)}{N}$$

Nieobciążony estymator autokorelacji (kompensujący błędy na krańcach rekordu danych, ale o większym błędzie kwadratowym):

$$\hat{R}'_{xx}(m) = \frac{1}{N-|m|} \sum_{n=0}^{N-|m|-1} x(n)x(n+m),$$



Rys. Estymaty autokorelacji:
 1) sygnału harmonicznego,
 2) szumu białego.
 Zwrócić uwagę na zafałszowany wzrost wariacji estymaty R'_{xx} na krańcach rekordu.

W Matlabie:

```
Rxy=xcorr(x,y,'opt')
```

```
Rxx=xcorr(x,'opt')
```

```
opt = biased
```

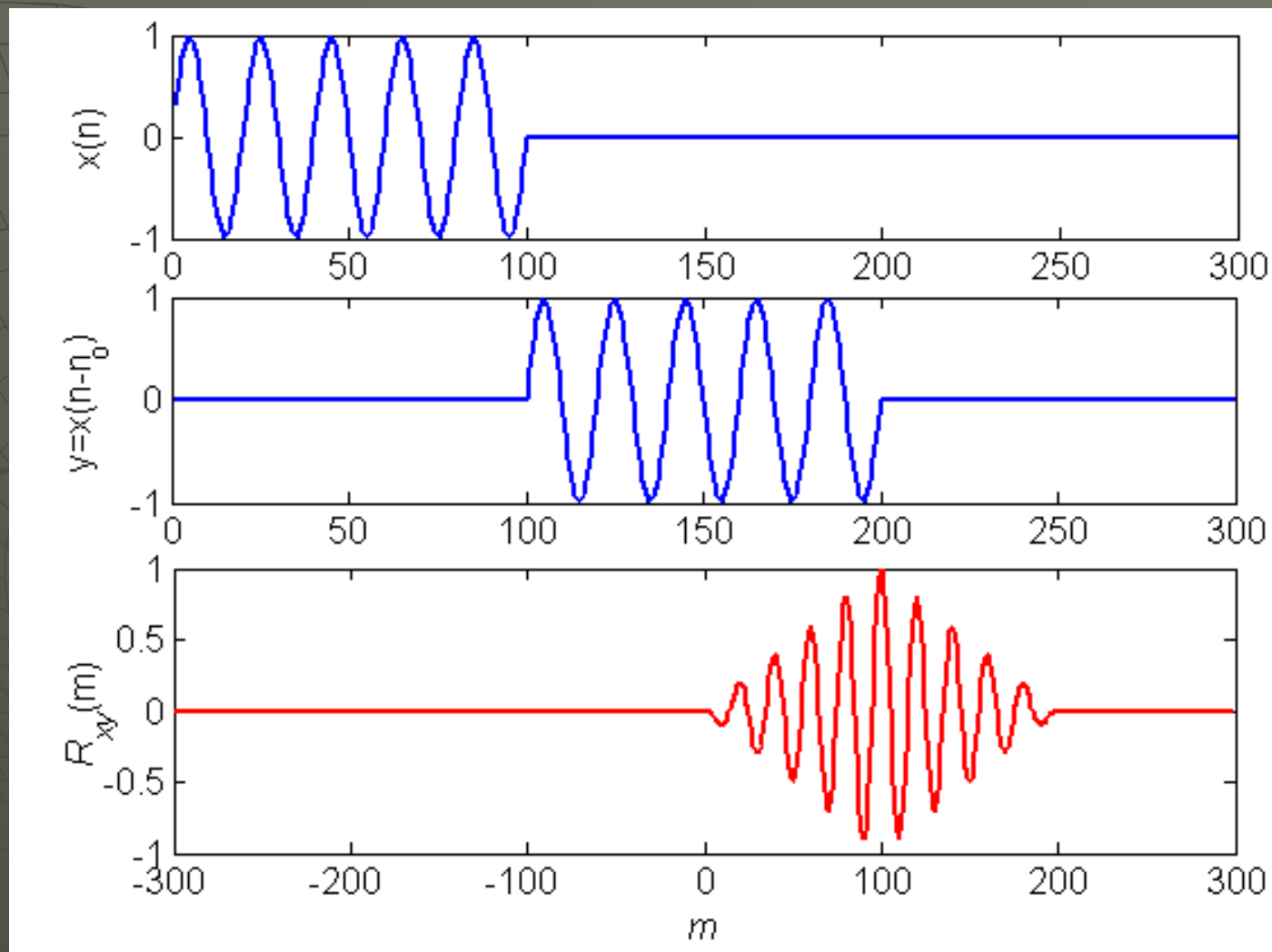
```
unbiased
```

```
coeff
```

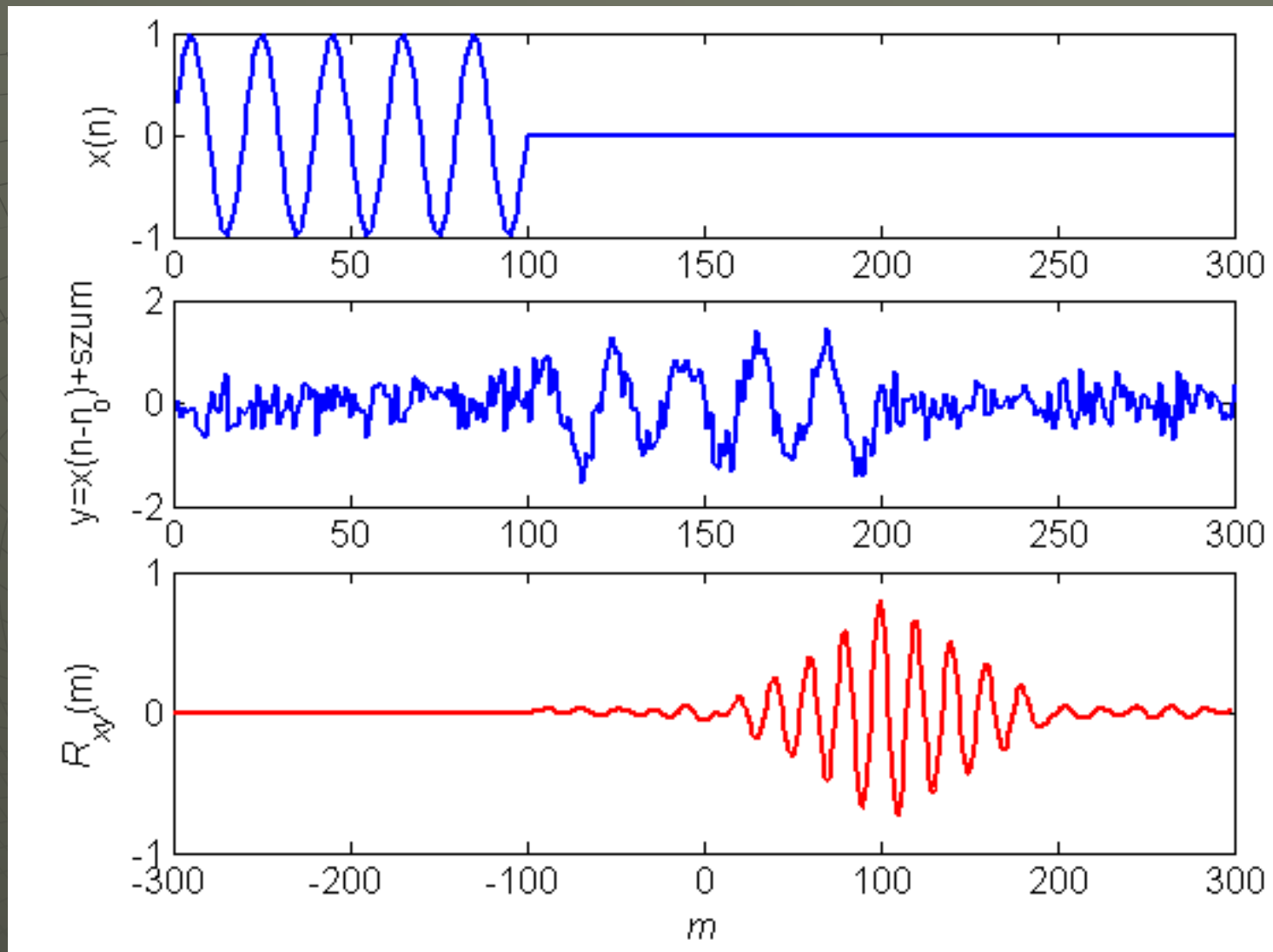
```
none
```

```
Cxy=xcov(x,y,'opt')
```

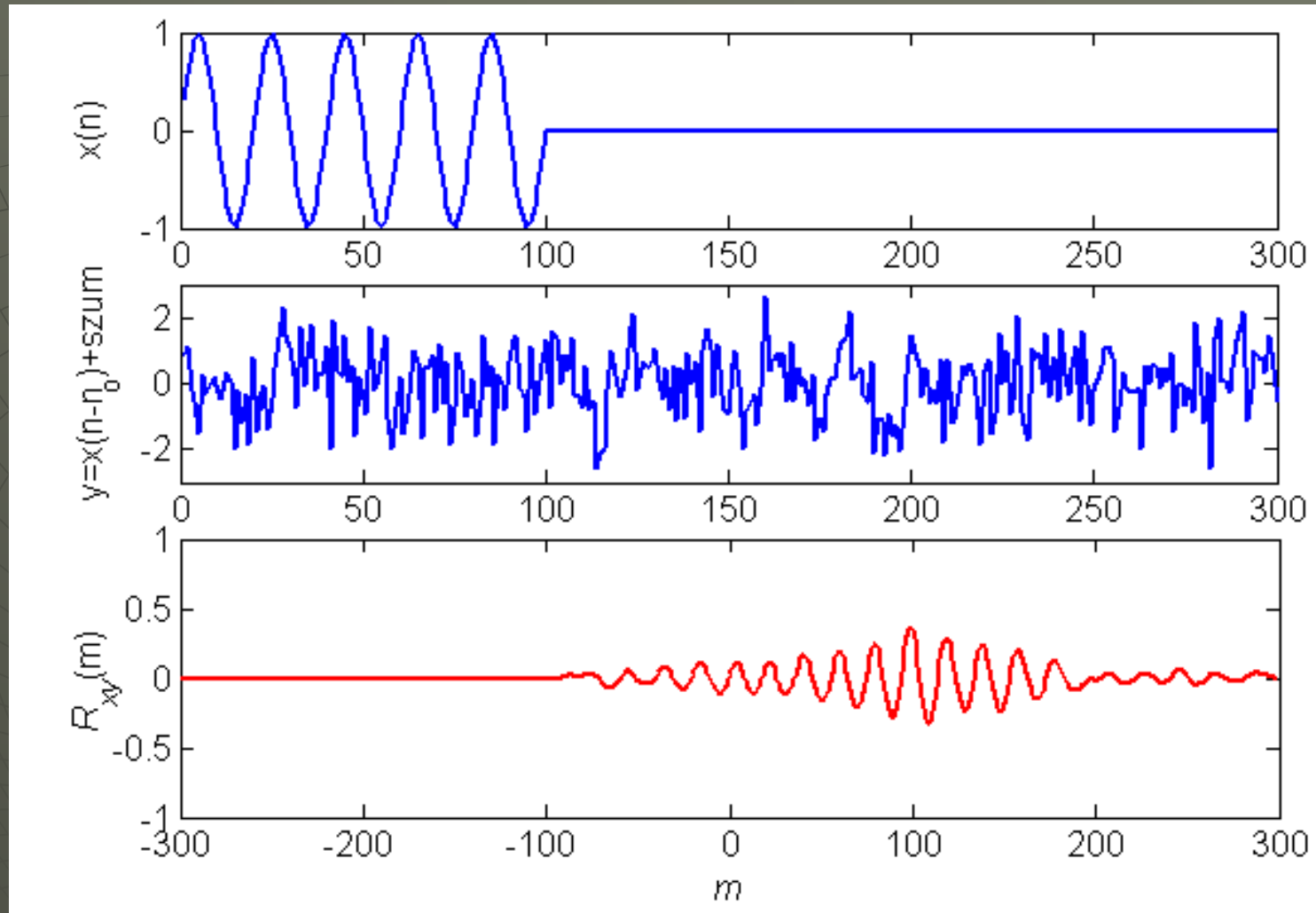
Przykład: Korelacja dwóch sygnałów



Rys. Korelacja wzajemna impulsowego sygnału $x(n)$ o długości N i jego opóźnionej wersji $y(n) = x(n - n_0)$. Opóźnienie $n_0 = 100$ próbek, $R_{xy}(m)$ unormowane przez R_{xy_max} . Maksimum korelacji dla $m = n_0$



Rys. Korelacja wzajemna $R_{xy}(m)$ sygnałów $x(n)$ i $y(n) = x(n - n_0) + \text{szum biały}$. Wariancja szumu $\sigma_x^2 = 0.1$, $m_x = 0$



Rys. Korelacja wzajemna sygnałów $x(n)$

i $y(n) = x(n-n_0) + \text{szum biały}$. Wariancja szumu $\sigma_x^2 = 1$.

Wniosek ogólny: korelacja wzajemna ujawnia składowe sygnału $y(n)$ skorelowane z sygnałem $x(n)$

- *estymator widma mocy PSD* (wyliczany na podstawie estymatora korelacji w przypadku $m_x=0$):

$$\hat{P}_{xx}(\Omega) = \frac{1}{N} \sum_{m=-(N-1)}^{N-1} \hat{R}_{xx}(m) e^{-jm\Omega}$$

Estymator ten jest obciążony i *nie jest zgodny*:

$$\text{var}[\hat{P}_{xx}(\Omega)] = P_{xx}^2(\Omega) \left[1 + \left(\frac{\sin N\Omega}{N \sin \Omega} \right)^2 \right]$$

W praktyce stosuje się metody, które pomijają etap estymacji $\hat{R}_{xx}(m)$ i umożliwiają estymację *PSD* bezpośrednio z próbek $x(n)$ poddawanych dyskretnemu przekształceniu Fouriera DFT (*met. korelacyjna*). Obliczanym w ten sposób estymatorem jest *periodogram*:

$$I_N(\Omega_m) = \hat{P}_{xx}(\Omega_m) = \frac{1}{N} \left| X(e^{j\Omega_m}) \right|^2 = \frac{1}{N} \left| \text{DFT}[x(n)] \right|^2$$

Periodogram jest estymatorem PSD niezgodnym i o dużej wariancji:

$$\text{var}[I_N(\Omega_m)] \cong P_{xx}^2(\Omega) \quad \text{IDFT}[I_N(\Omega_m)] \text{ daje estymatę } \hat{R}_{xx}(m)$$

Uśrednianie periodogramów – estymacja z wygładzaniem

W celu uzyskania zgodnego estymatora PSD (o malejącej wariancji przy wzroście liczby próbek N) stosuje się *uśrednianie pewnej liczby niezależnych periodogramów* (metoda Bartletta).

1) Ciąg danych $\{x(n): n=0,1,\dots,N-1\}$ dzieli się na K segmentów $x^{(i)}(n)$, $i=1,2,\dots,K$ o długości M każdy ($N=KM$).

2) Oblicza się K krótszych periodogramów $I_M^{(i)}(\Omega_m) = \frac{1}{M} |DFT[x^{(i)}(n)]|^2$

3) Wyznacza się estymatę Bartletta PSD w postaci:

$$\hat{P}_{xx}^B(\Omega_m) = \frac{1}{K} \sum_{i=1}^K I_M^{(i)}(\Omega_m)$$

Kiedy periodogramy są od siebie niezależne, wariancja estymaty:

$$\text{var}[\hat{P}_{xx}^B(\Omega)] = \frac{1}{K} \text{var}[I_M(\Omega)] \approx \frac{1}{K} P_{xx}^2(\Omega)$$

Obciążenie jest większe niż dla jednego długiego periodogramu, ale estymata jest *zgodna*, co oznacza *wygładzenie widma* przy wzroście K . Ze względu na krótsze segmenty *pogarsza się rozdzielczość częstotliwościowa*.

Metoda Welch nieparametrycznej estymacji PSD

Ulepszona metoda uśredniania periodogramów polega na nałożeniu na segmenty danych *przed* obliczaniem periodogramów okna czasowego różnego od prostokątnego (zwykle Hanninga).

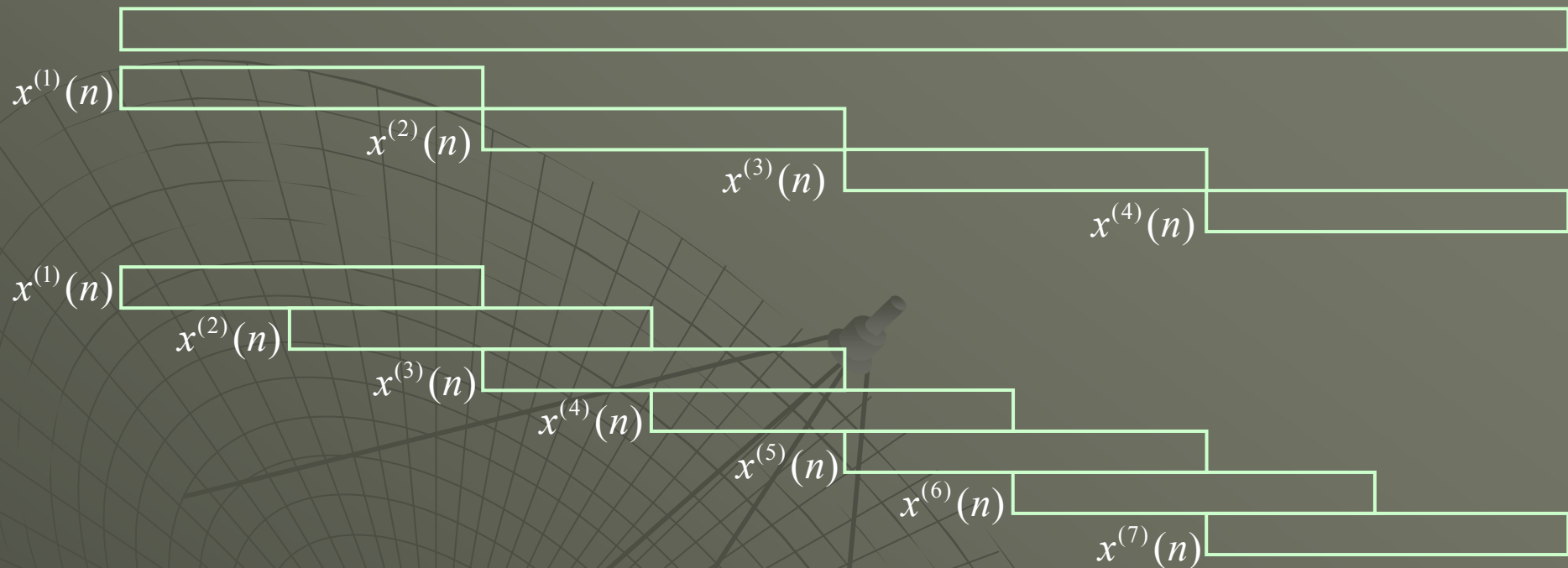
Zmodyfikowane periodogramy mają postać

$$I_M^{(i)W}(\Omega_m) = \frac{1}{MF} |DFT[x^{(i)}(n) \cdot w(n)]|^2$$

gdzie $w(n)$ jest funkcją okna czasowego, a F jest czynnikiem normującym energię (*energia sygnału okna*):

$$F = \frac{1}{M} \sum_{n=0}^{M-1} w^2(n)$$

W celu zwiększenia liczby segmentów K (zmniejszenia wariacji estymacji) bez zmniejszania długości M (tzn. rozdzielczości widmowej) stosuje się podział na segmenty nakładające się na siebie (*overlapping*). Zależność periodogramów segmentów zachodzących na siebie zmniejsza się przez nałożenie (przed obliczeniem periodogramów) okna nieprostokątnego. Metoda Welch z nakładaniem się segmentów daje estymatę o mniejszej wariacji w porównaniu z metodą Bartletta.



Rys. Podział rekordu próbek o długości N na $K=4$ segmenty o długości $M=N/4$ i na $K=7$ segmentów zachodzących na siebie

Estymacja widma mocy metodą Welch w Matlabie:

```
[Pxx, f] = psd(x, ndft, fs, win, overlap)
```

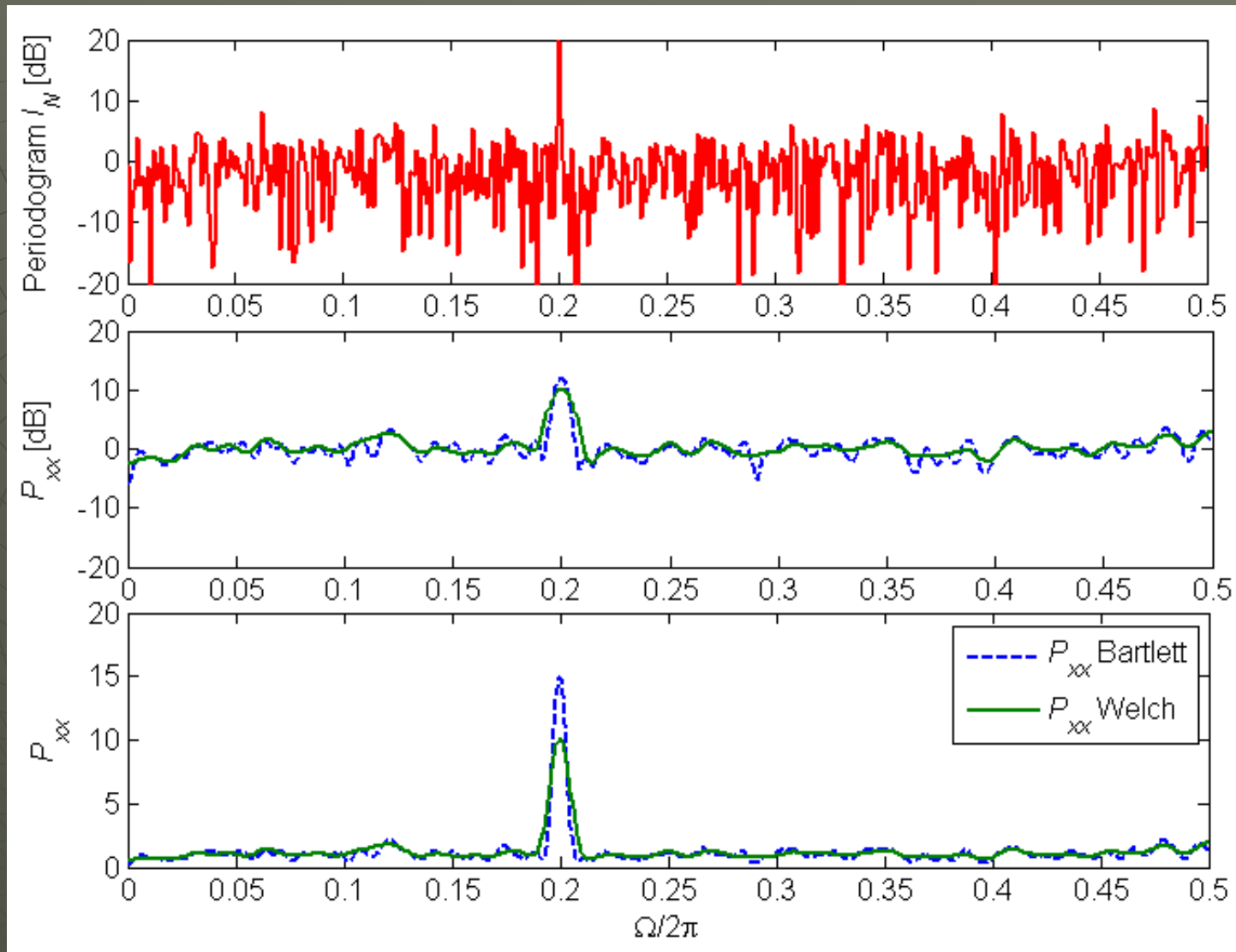
x - wektor próbek sygnału

$ndft$ - liczba punktów DFT

fs - częstotliwość próbkowania (określa skalę częstotliwości)

win - typ i długość okna czasowego (segmentu, np. `hanning(M)`)

$overlap$ - długość obszaru nakładania się segmentów



Rys. Estymacja widma mocy szumu białego $N(0,1)$ i składowej harmonicznej o $F=0.2$. Liczba próbek $N=1024$, długość segmentów $M=128$, $overlap=64$. Estymata Welcha z oknem Hanninga

Przetwarzanie sygnału losowego przez liniowy układ dyskretny

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1 z^{-1} + \dots + b_N z^{-N}}{1 + a_1 z^{-1} + \dots + a_N z^{-N}} = Z[h(n)]$$

Sygnał wyjściowy układu jest *splotem* sygnału wejściowego i charakterystyki impulsowej:

$$y(n) = h(n) * x(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k)$$

Jeżeli sygnał wejściowy $x(n)$ jest sygnałem losowym stacjonarnym w szerszym sensie o znanych m_x , $\text{var}(x)$ i $R_{xx}(m)$, to (w *stanie ustalonym* po zaniku procesów przejściowych) sygnał wyjściowy $y(n)$ jest również sygnałem losowym o statystykach zależnych od parametrów sygnału wejściowego i od charakterystyki widmowej (lub impulsowej) filtra.

Zależności pomiędzy statystykami sygnału wejściowego i wyjściowego w stanie ustalonym:

- wartość średnia sygnału wyjściowego: $m_y = H(e^{j0})m_x$
- funkcja autokorelacji sygnału wyjściowego:

$$R_{yy}(m) = h(m) * h(-m) * R_{xx}(m)$$

gdzie:
$$h(m) * h(-m) = \sum_{k=-\infty}^{\infty} h(k)h(n+k)$$

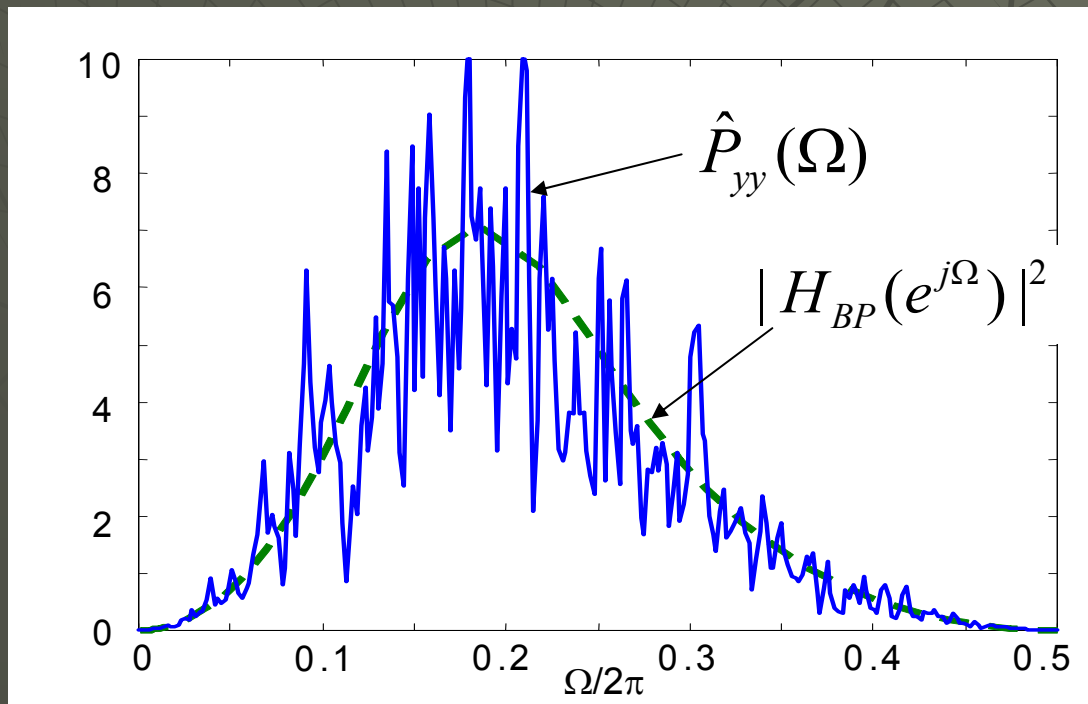
- jeżeli sygnał wejściowy ma zerową wartość średnią $m_x=0$, a transformaty Z autokorelacji sygnałów oznaczymy jako $S_{xx}(z)=Z[R_{xx}(m)]$, $S_{yy}(z)=Z[R_{yy}(m)]$, to:

$$S_{yy}(z) = H(z) \cdot H(z^{-1}) \cdot S_{xx}(z)$$

Widmo mocy sygnału wyjściowego: $P_{yy}(\Omega) = S_{yy}(z) \Big|_{z=e^{j\Omega}}$

$$P_{yy}(\Omega) = S_{yy}(z) \Big|_{z=e^{j\Omega}} = |H(e^{j\Omega})|^2 P_{xx}(\Omega)$$

Powyższa zależność stanowi podstawę do *modelowania sygnałów losowych o określonym widmie mocy* (szumów „kolorowych”) za pomocą szumu białego przepuszczanego przez stabilny układ liniowy o odpowiedniej transmitancji.



Rys. Periodogram szumu na wyjściu filtra środkowoprzepustowego 2-go rzędu o transmitancji $H_{BP}(z)$ pobudzanego szumem białym o wariancji $\sigma_x^2=1$, $P_{xx}(\Omega)=1=\text{const}$ (koloryzowanie szumu)

- funkcja *korelacji wzajemnej* sygnałów wejściowego i wyjściowego:

$$R_{xy}(m) = h(m) * R_{xx}(m)$$

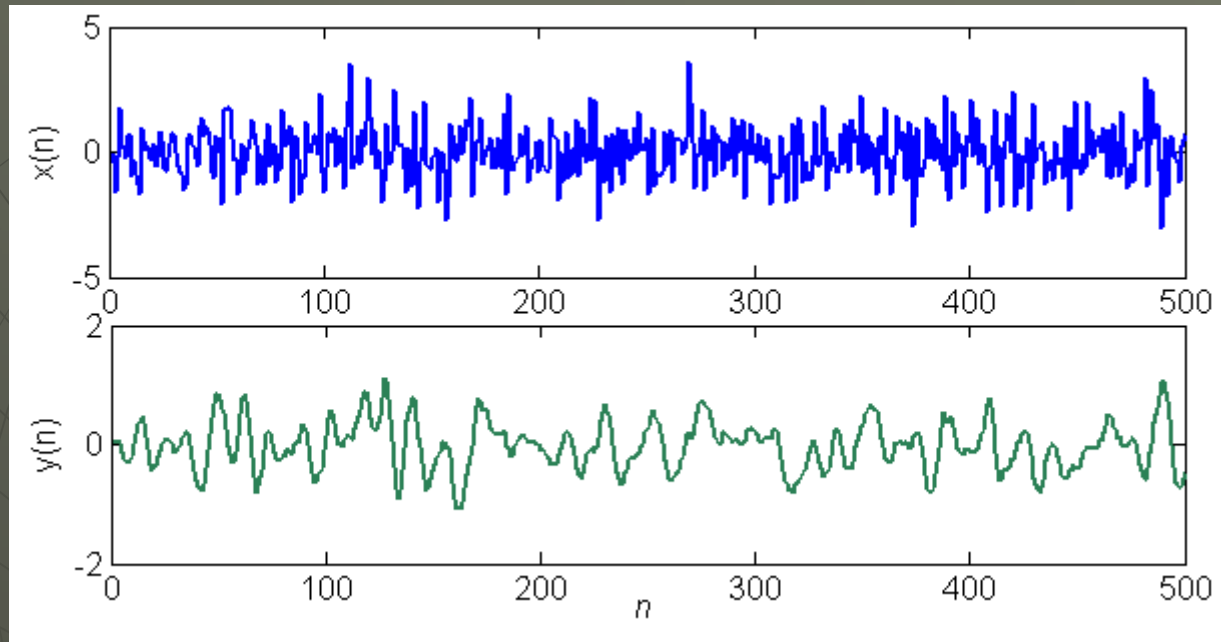
Wynika z tego, że funkcja widma mocy wzajemnej (w ogólności zespolona):

$$S_{xy}(z) = H(z)S_{xx}(z) \Big|_{z=e^{j\Omega}} \rightarrow P_{xy}(\Omega) = H(e^{j\Omega})P_{xx}(\Omega)$$

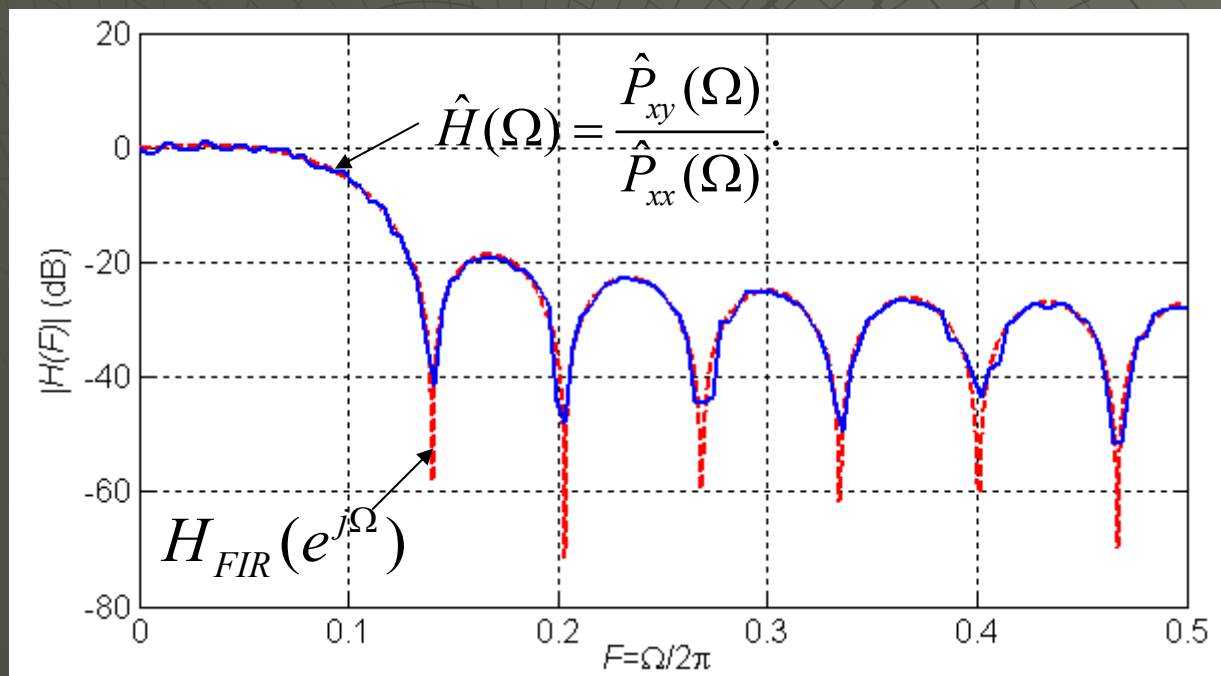
Powyższa zależność pomiędzy widmami mocy umożliwia *identyfikację* charakterystyki widmowej nieznanego układu liniowego na podstawie obserwacji realizacji sygnałów losowych na jego wejściu i wyjściu:

$$H(e^{j\Omega}) = \frac{P_{xy}(\Omega)}{P_{xx}(\Omega)}$$

Sygnał wejściowy musi być **szerokopasmowy**, żeby pobudzać wszystkie składowe dynamiki układu (najlepszym pobudzeniem jest szum biały obejmujący całe pasmo).



Rys. Sygnał wejściowy $x(n)$ (szum biały) i wyjściowy $y(n)$ dolnoprzepustowego filtra FIR



Rys. Rzeczywista i estymowana charakterystyka amplitudowa filtra

Detekcja sygnału w szumie – filtr dopasowany (*matched filter*)

Filtry dopasowane są filtrami optymalnymi z punktu widzenia zdolności do wykrywania impulsu znanego sygnału użytecznego $s(n)$ w szumie (stanowią jedną z podstaw działania radarów).

Charakterystyką impulsową *filtra dopasowanego do sygnału* deterministycznego $s(n)$ o skończonym czasie trwania równym N jest ten sygnał odwrócony w czasie (w przypadku zespolonych sygnałów modulowanych sprzężenie zespolone $s^*(-n)$) [North, poufny raport 1943]

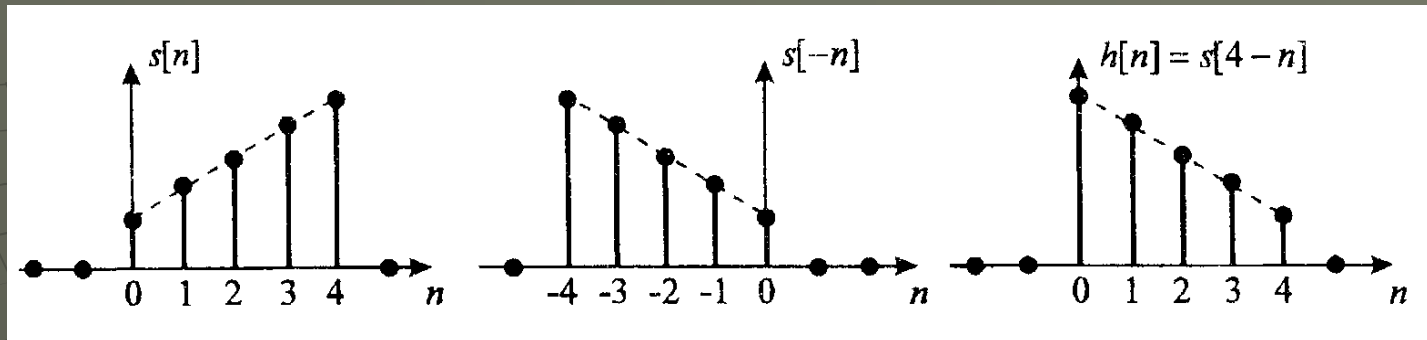
$$h(n) = s(-n), \quad n = 0, 1, \dots, N$$

Aby otrzymać filtr przyczynowy charakterystykę tę należy opóźnić o N okresów:

$$h(n) = s(N - n)$$

Sygnał wyjściowy filtra pobudzanego sygnałem, do którego jest on dopasowany:

$$y(n) = \sum_{k=0}^n h(k)s(n-k) = \sum_{k=0}^n s(N-k)s(n-k)$$



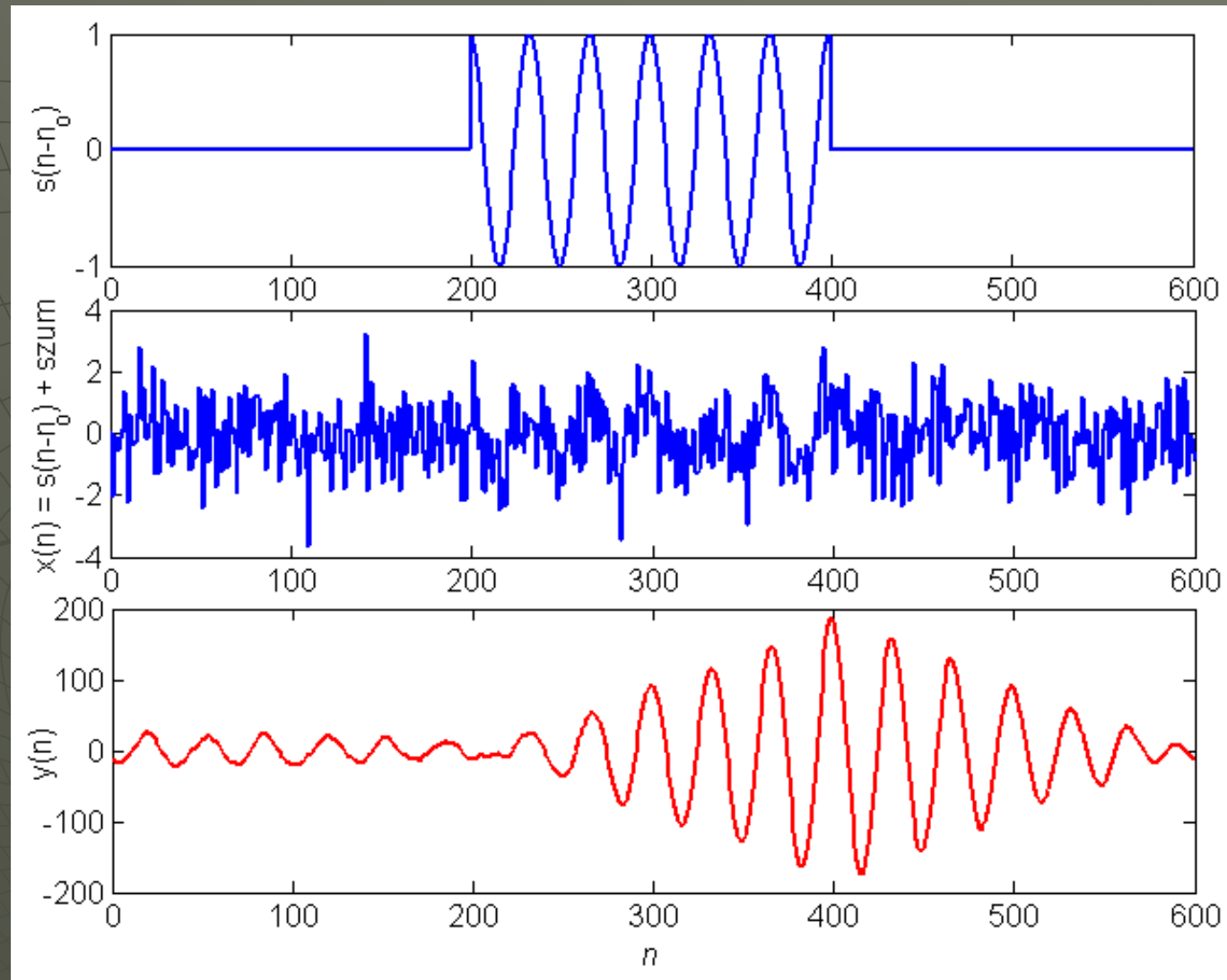
Rys. Sygnał $s(n)$ o skończonej długości ($N=4$) i odpowiedź impulsowa $h(n)=s(N-n)$ przyczynowego filtra FIR dopasowanego do sygnału $s(n)$

Dla $n=N$ otrzymujemy na wyjściu filtra:

$$y(N) = \sum_{k=0}^N s(N-k)s(N-k) = \sum_{m=0}^N s^2(m)$$

Zatem w chwili $n=N$ wartość sygnału wyjściowego reprezentuje całą energię sygnału użytecznego (maksymalną wartość funkcji korelacji sygnału dopasowanego i wejściowego) i wtedy odpowiedź filtra osiąga maksimum, *niezależnie od kształtu $s(n)$* .

W układach radarowych wysyłane impulsy stanowią sygnał $s(n)$, który wraca do anteny w formie opóźnionej $s(n-n_0)$ +szum. Wyjście filtra dopasowanego osiąga wtedy maksimum dla $n=N+n_0$.



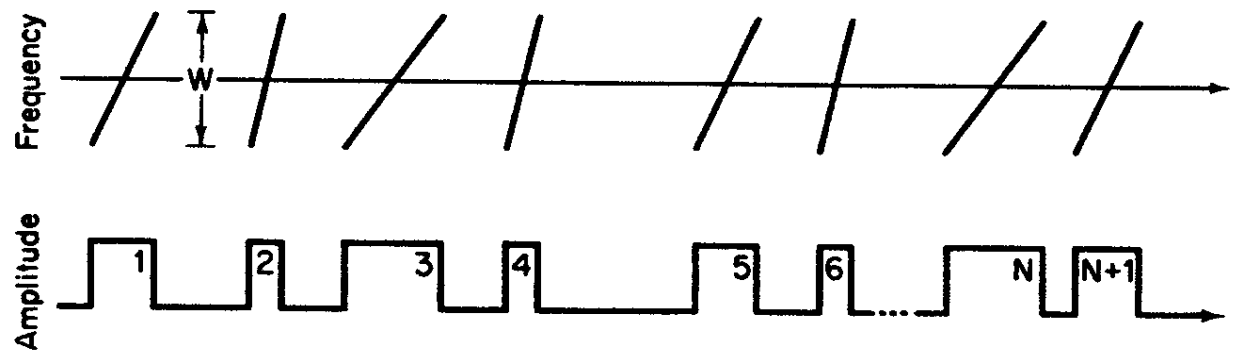
Rys. 1) Opóźniona o $n_0=200$ próbek wersja sygnału impulsowego $s(n)$ o długości $N=200$ i amplitudzie $A=1$, 2) sygnał $x(n)=s(n-n_0)+$ szum biały ($m_x=0, \sigma_x^2=1$) na wejściu filtra dopasowanego o ch-ce $h(n)=s(-n+N)$, 3) sygnał wyjściowy filtra $y(n)$, maksimum dla $n=N+n_0$

Największą efektywność detekcji uzyskuje się dla impulsów $s(n)$, których autokorelacja bardzo szybko zanika do zera. Filtr dopasowany reaguje wtedy skupionym impulsem wyjściowym. Właściwość taką ma stosowany w układach radarowych sygnał harmoniczny z liniowo modulowaną częstotliwością (*LFM*), którego ciągła wersja ma postać:

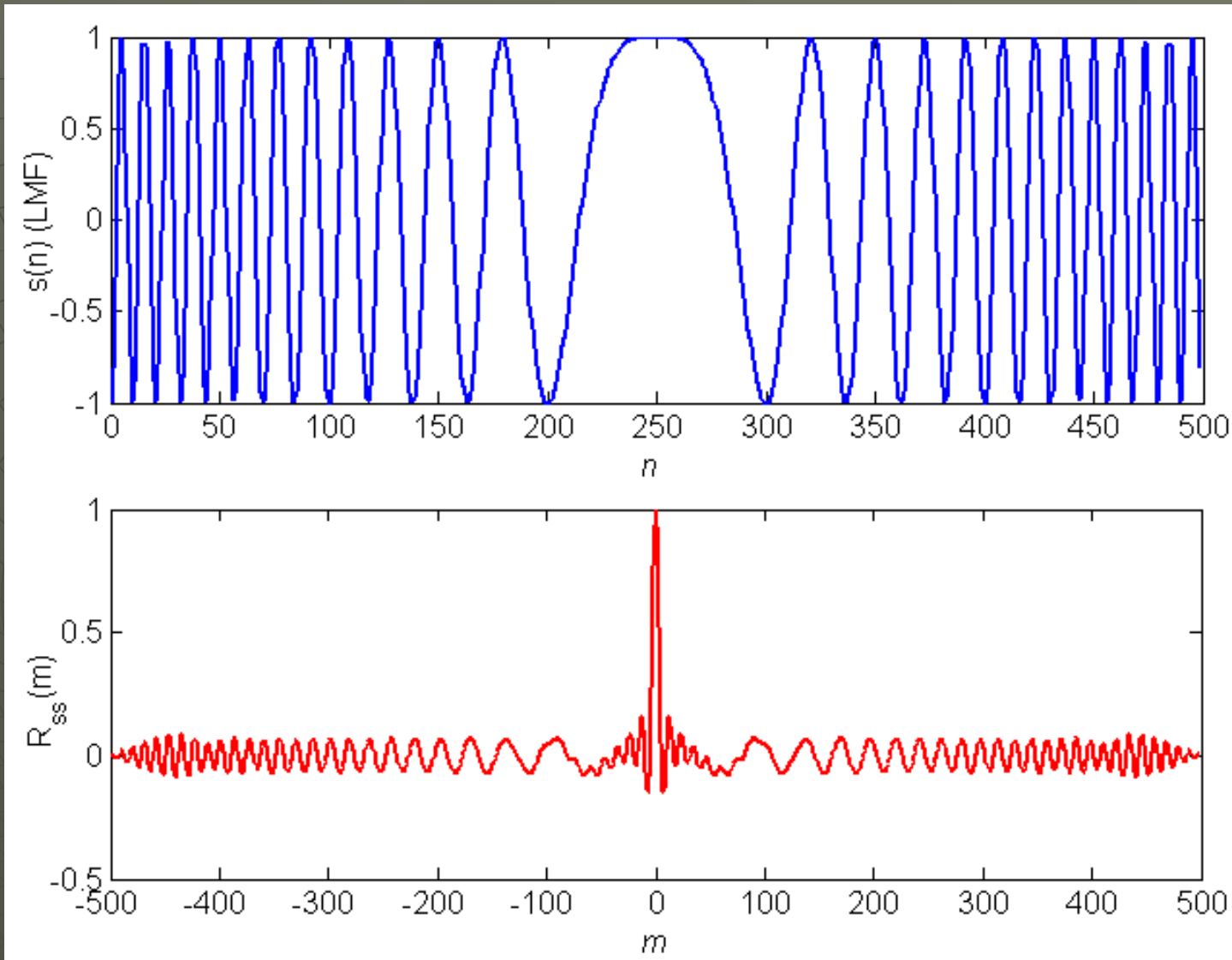
$$s(t) = p(t)e^{j\pi\frac{W}{T}t^2}, \quad p(t) = \begin{cases} 1 & \text{dla } -T \leq t \leq T \\ 0 & \text{dla pozostałych } t \end{cases}$$

gdzie $2T$ – czas trwania impulsu, W – szerokość przemiatanego zakresu częstotliwości.

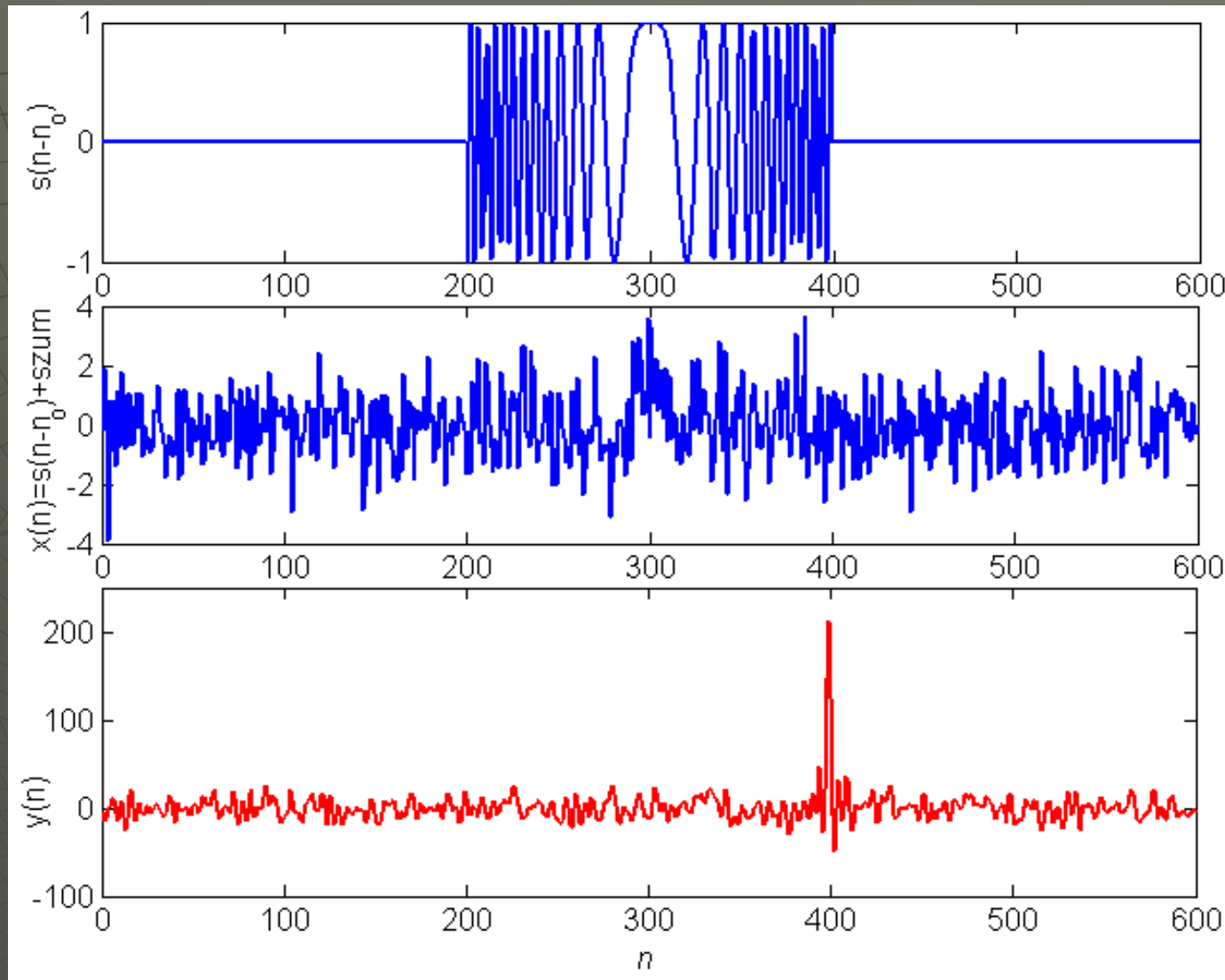
Filtr dopasowany działa w trybie blokowym *frame-by-frame* dla rekordu z całego okna odbiorczego radaru korzystając z *FFT*.



Rys. Seria niejednorodnych impulsów *LFM* o takim samym paśmie częstotliwości W



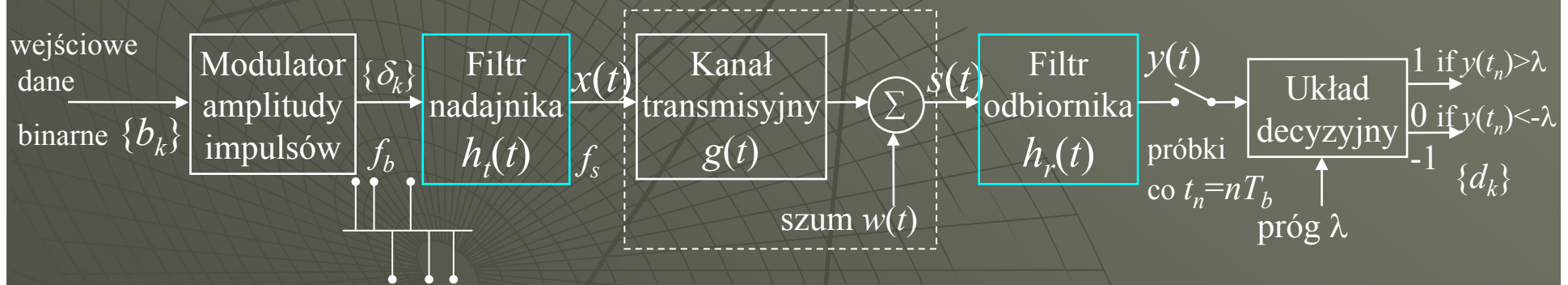
Rys. Sygnał $s(n)$ o liniowo modulowanej częstotliwości (*Linear Frequency Modulation LFM* – „chirp”) i jego unormowana funkcja autokorelacji $R_{ss}(m)$



Rys. Detekcja impulsowego sygnału *LFM* przez filtr dopasowany. Impuls zakłócony szumem białym $m_x=0$, $\sigma_x^2=1$. Zwrócić uwagę na skupienie (kompresję) odpowiedzi filtra

Filtracja dopasowana w transmisji w paśmie podstawowym

Pasmo podstawowe (*baseband*) to pasmo wokół częstotliwości 0, w którym sygnał jest transmitowany bez modulacji (np. sygnał video, sygnały cyfrowe w sieciach LAN). Pasmo podstawowe jest kanałem dyspersyjnym, przesyłane impulsy są zakłócone przez impulsy sąsiednie (*interferencja międzysymbolowa ISI - Intersymbol Interference*) oraz przez szумы.



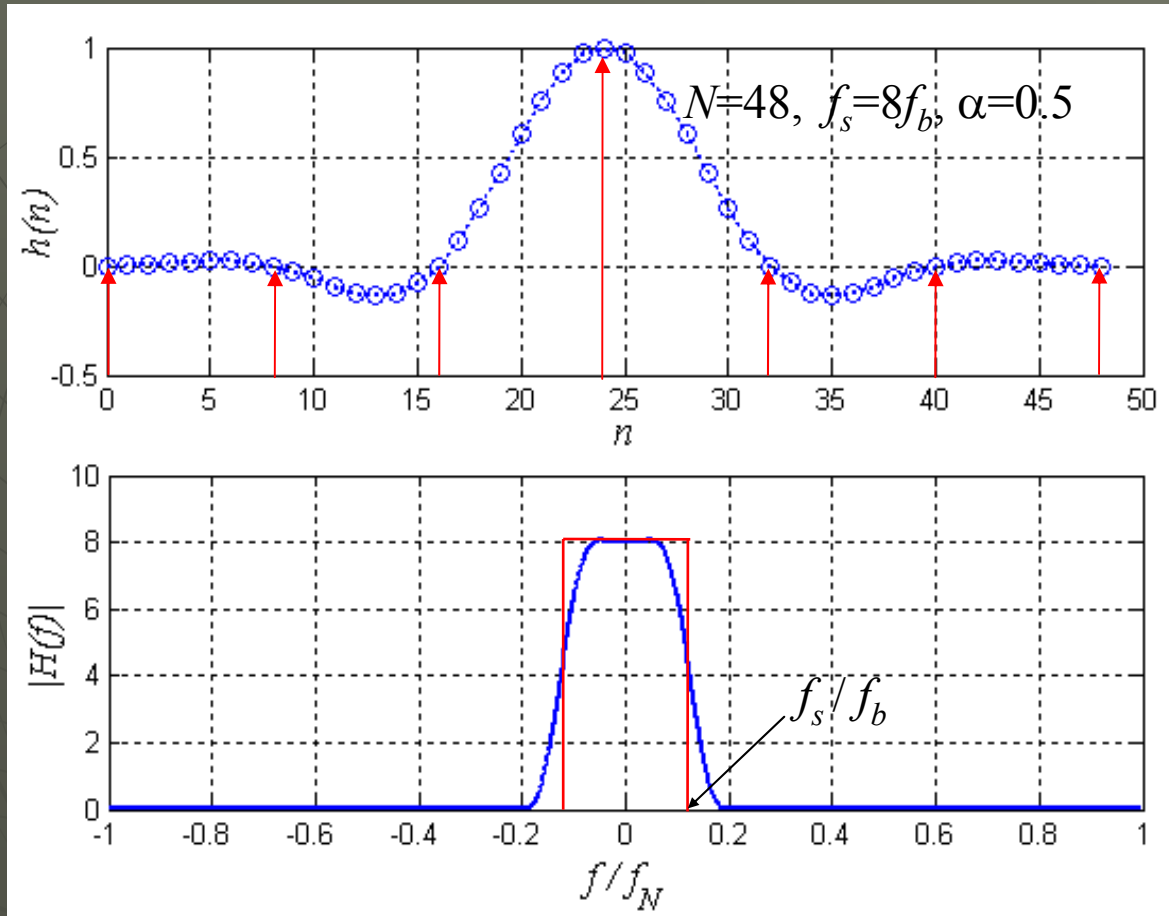
Rys. Schemat transmisji danych binarnych w paśmie podstawowym. $f_b = 1/T_b$ – częstotliwość bitowa, $f_s > f_b$ – częstotliwość pracy filtrów. Odbiornik zawiera filtr dopasowany do impulsów nadawanych $x(t)$, tj. $h_r(n) = h_t(N-n)$

Pożądanne właściwości ze względu na ISI i możliwość realizacji mają filtry o widmie podniesionego cosinusa (*raised cosine filter*)

$$H(f) = \begin{cases} \frac{1}{f_b} = \text{const}, & 0 \leq |f| < f_1 \\ \frac{1}{2f_b} \left\{ 1 - \sin \left[\frac{\pi(|f| - \frac{f_b}{2})}{f_b - 2f_1} \right] \right\}, & f_1 \leq |f| < f_b - f_1 \\ 0, & |f| \geq f_b - f_1 \end{cases}$$

Pasmo transmisji: $B_T = (1 + \alpha) \frac{f_b}{2}$

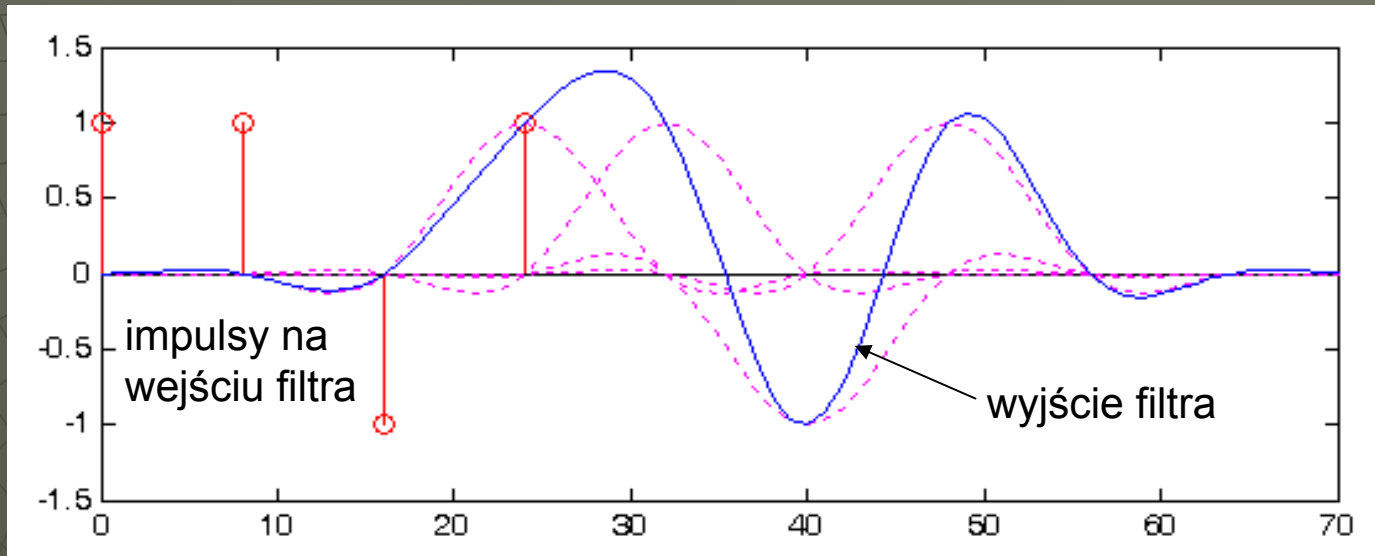
$\alpha = 1 - 2 \frac{f_1}{f_b}$ - indeks spadku (*rolloff factor*), nadmiar pasma (zwykle używany parametr projektowy)



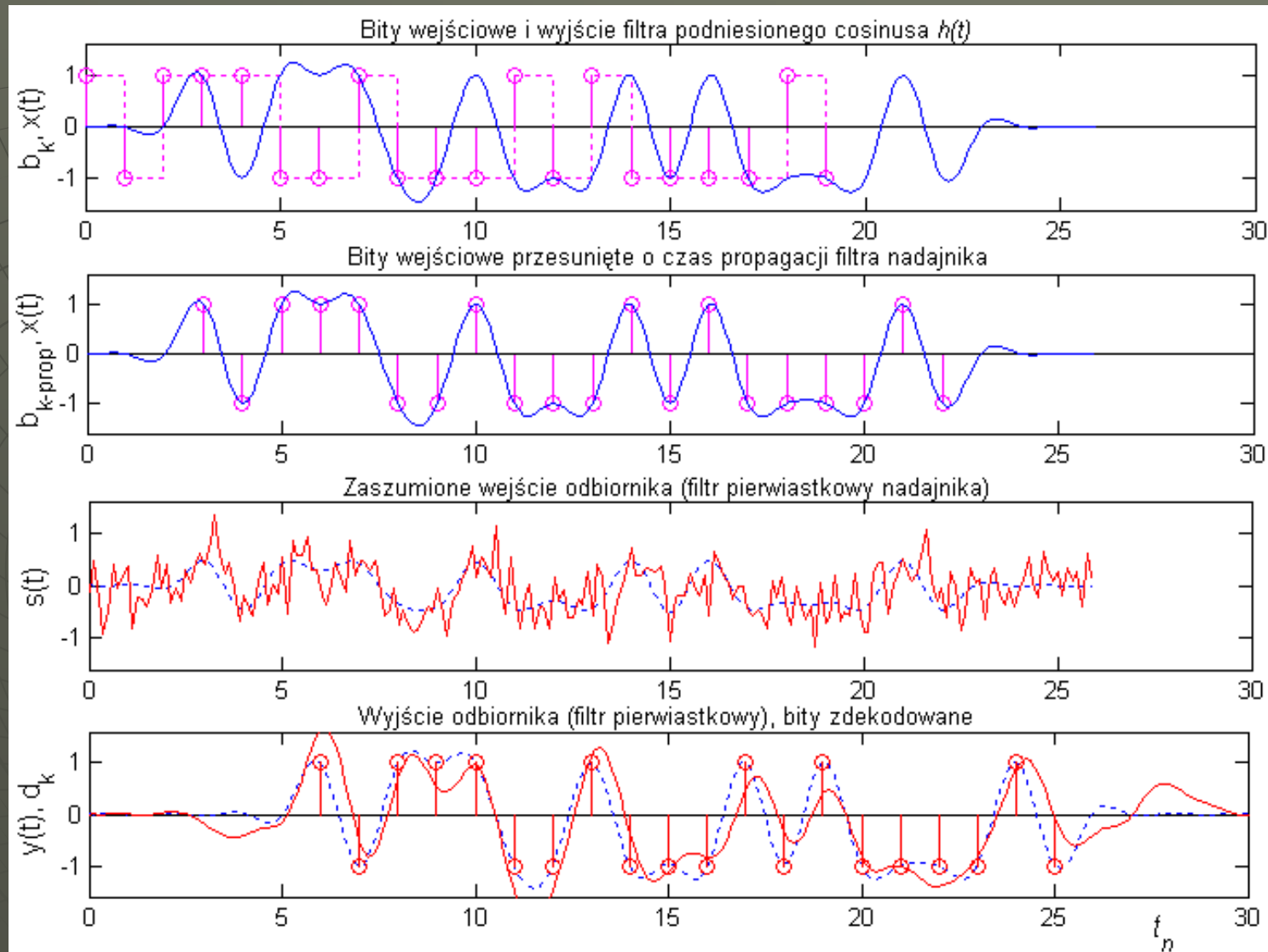
Rys. Charakterystyka impulsowa i widmowa amplitudowa filtra SOI podniesionego cosinusa (prostokątna dla $\alpha=0$). Odpowiedź impulsowa ma miejsca zerowe co okres bitowy T_b , co zapobiega ISI, a w porównaniu z funkcją *sinc* szybciej zanika na krańcach.

W nadajniku i odbiorniku umieszcza się dwa filtry o charakterystykach pierwiastkowych:

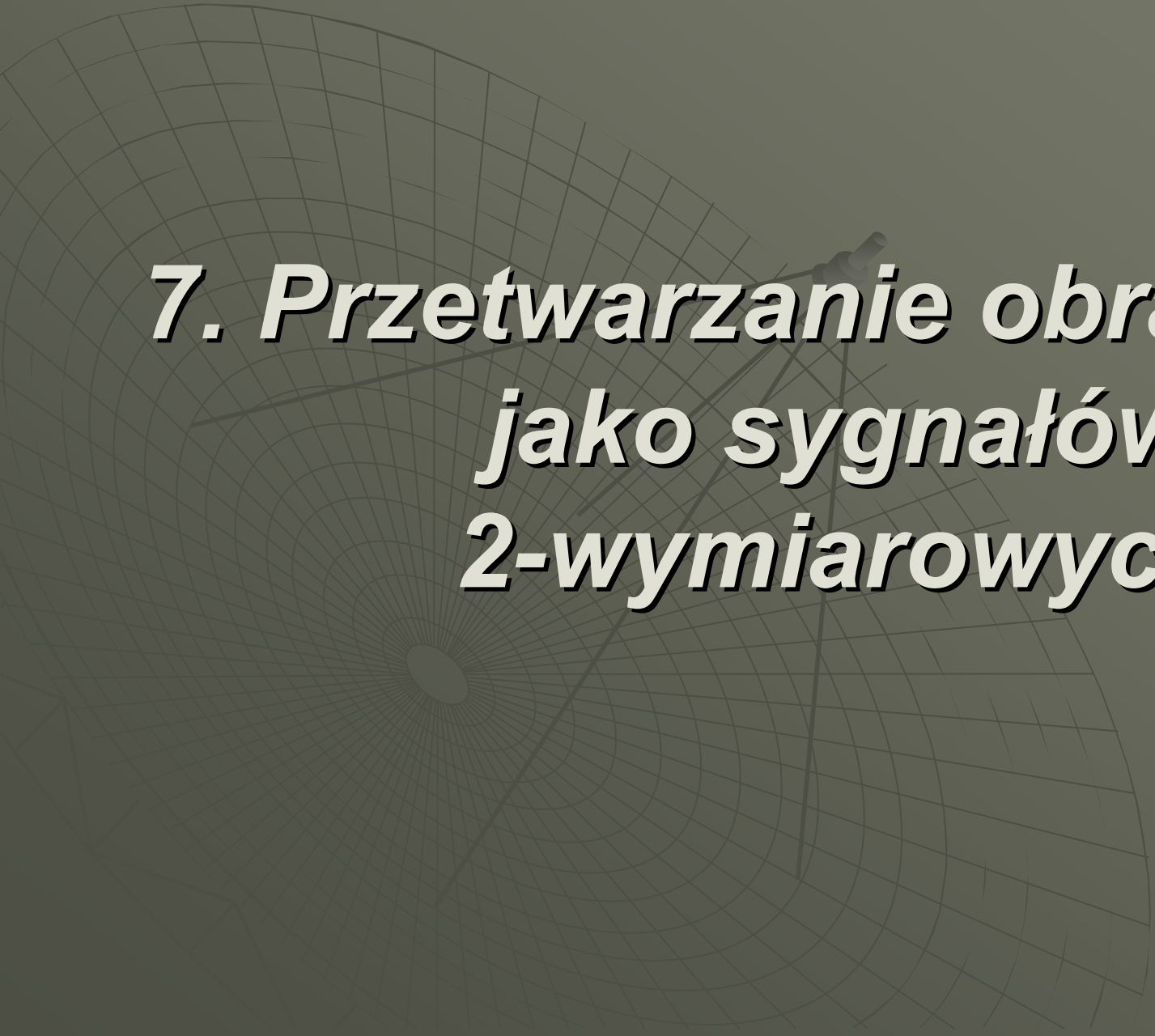
$$H_t(z) = H_r(z) = \sqrt{H(z)}, \quad h_t(n) * h_r(n) = h(n)$$



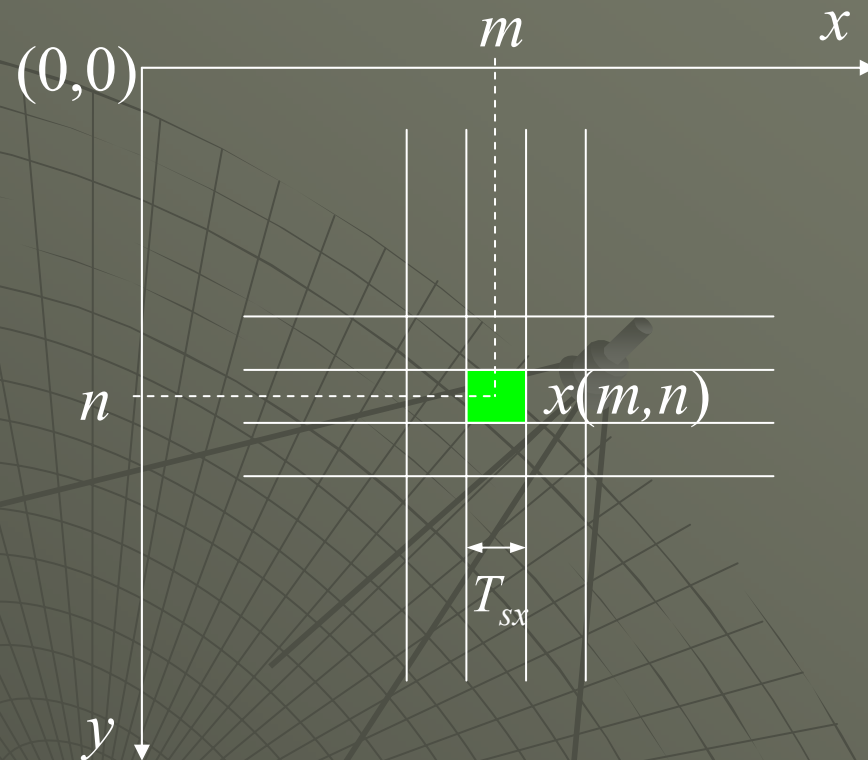
Rys. Sygnał wyjściowy filtra nadajnika jako suma przesuniętych w czasie funkcji podniesionego cosinusa - odpowiedzi na impulsy jednostkowe reprezentujące ciąg bitów [1 1 -1 1]. Zwrócić uwagę na opóźnienie grupowe filtra (równe 3 okresy bitowe T_b).



Rys. Detekcja danych binarnych ± 1 po transmisji w paśmie podstawowym przez zaszumiony kanał. Sygnał nadawany jest sumą poprzesuwanymi o okres bitowy $T_b=1$ odpowiedzi impulsowych podniesionego cosinusa. Filtr odbiornika stanowi filtr dopasowany zapewniający optymalną detekcję bitów.

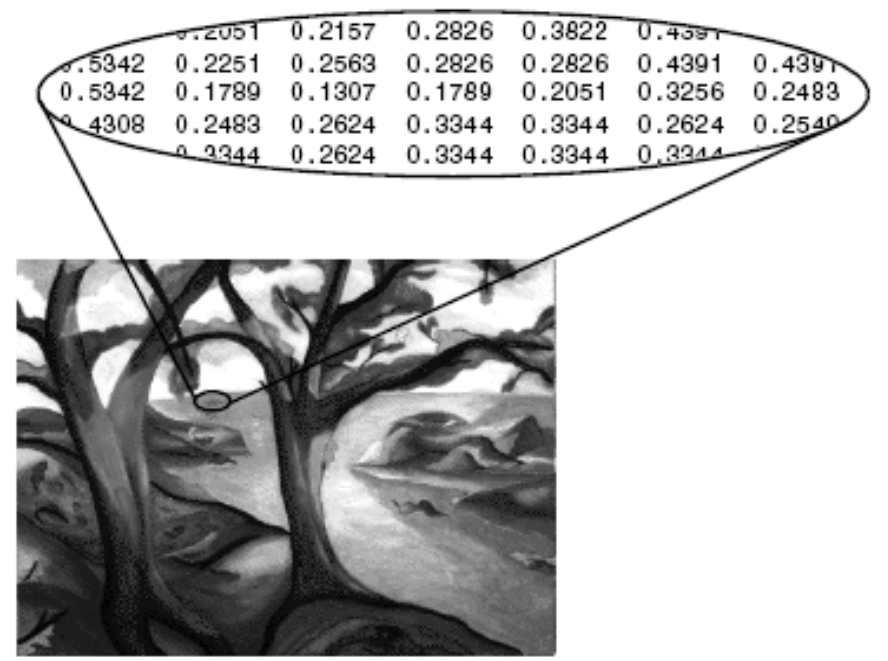
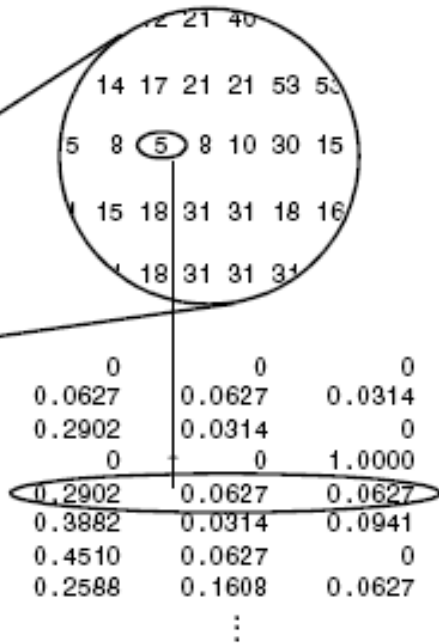
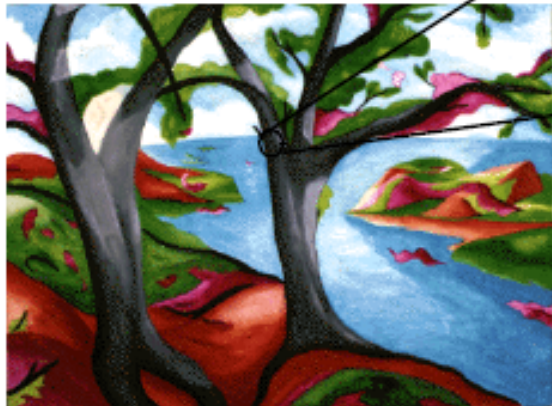


7. Przetwarzanie obrazów jako sygnałów 2-wymiarowych



Rys. Indeksowanie pikseli obrazu traktowanego jako 2-wymiarowy sygnał $x(m,n)$. Okres próbkowania T_s ma wymiar długości.

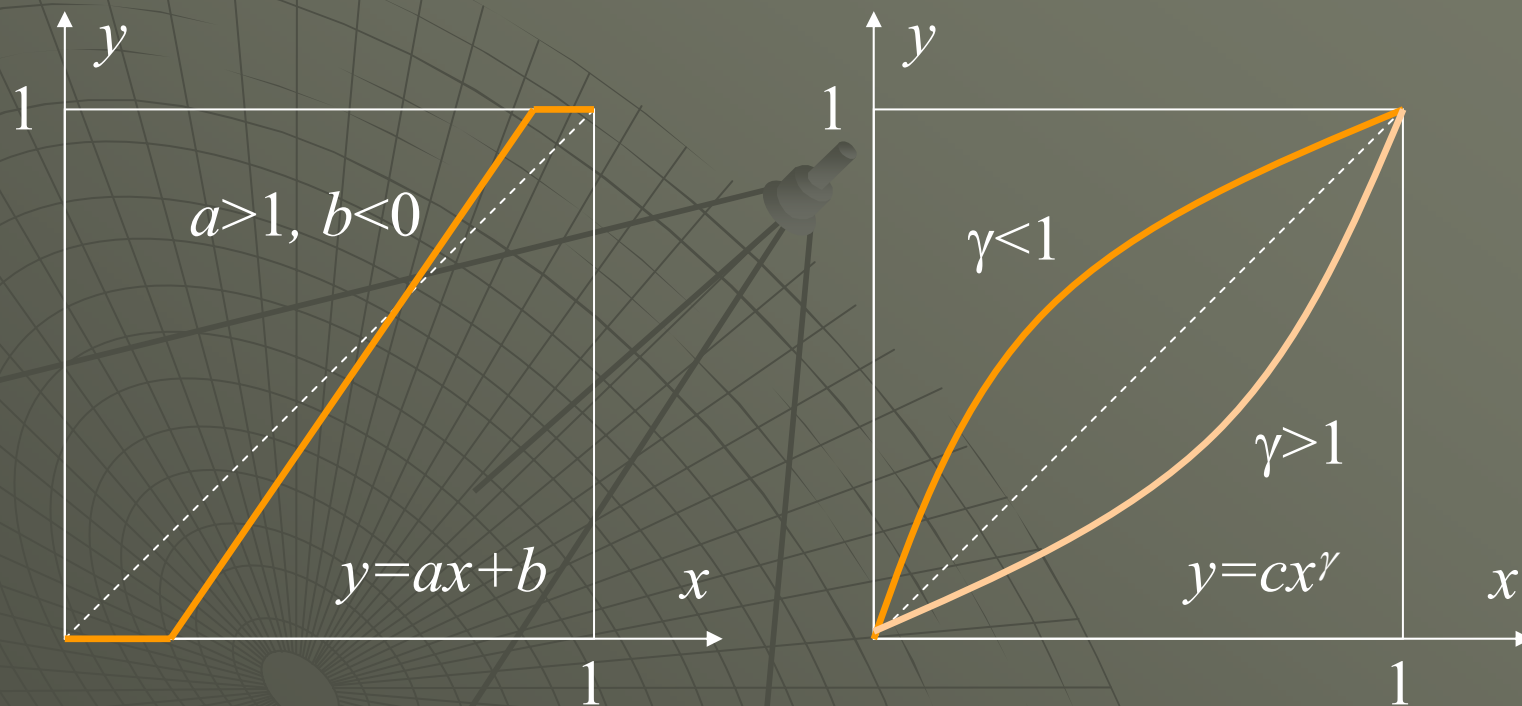
Częstotliwość próbkowania: $f_s = \frac{1}{T_s}$ (np. dpi)



Rys. Obraz indeksowany i obraz intensywności (przeskalowane do zakresu $0 \leq x \leq 1$). Kolorowe obrazy RGB to obrazy intensywności dla 3 kolorów podstawowych

Globalne operacje arytmetyczne na pikselach

Zmiana jasności i kontrastu, korekcja gamma



Rys. Liniowe przekształcenie $y=ax+b$ (zmiana jasności i kontrastu) z nasyceniem. Wartości pikseli unormowane do zakresu $0 \leq x/x_{max} \leq 1$. Zakres wartości: $0 < a < \infty$, $-1 < b < 1$.
 $a > 1$ – zwiększenie kontrastu, $b > 0$ – zwiększenie jasności (intensywności)

Rozciąganie i wyrównywanie histogramu (*histogram equalization*)

Poziomy (wysokości słupków) histogramu (L skwantowanych poziomów intensywności $Q[r_k]$, n pikseli w obrazie, n_k pikseli o wartości r_k):

$$p(r_k) = \frac{n_k}{n}, \quad 0 \leq r_k \leq 1, \quad k = 0, 1, \dots, L-1$$

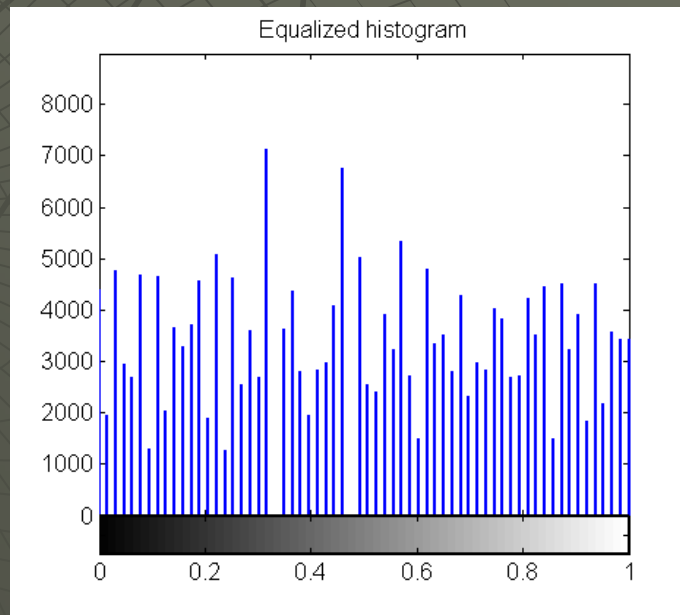
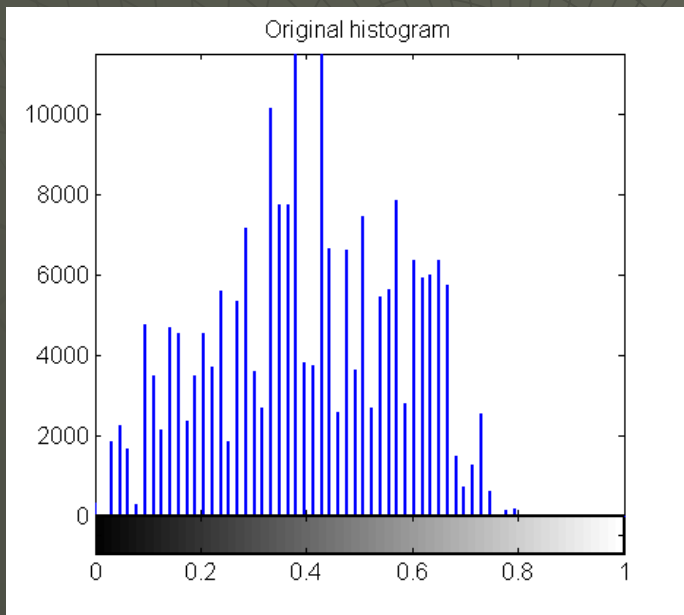
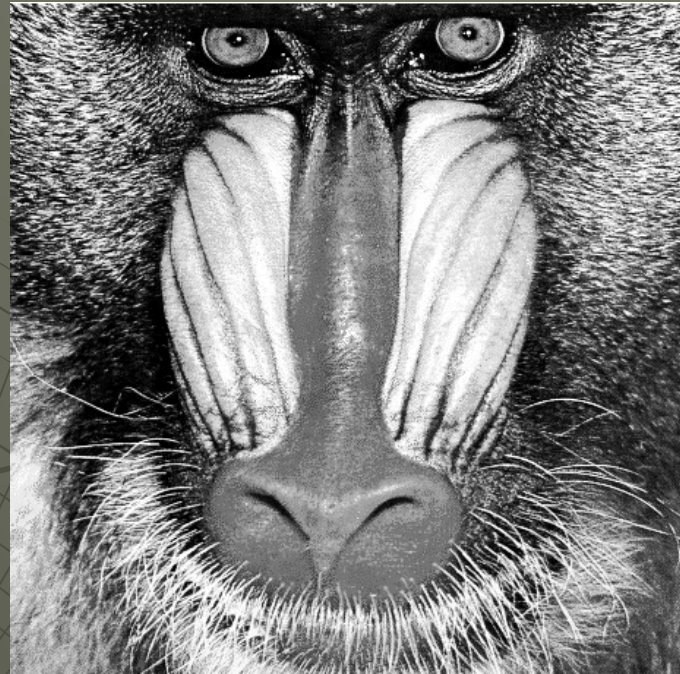
Po przekształceniu wszystkie piksele o jasności r_k uzyskują nową wartość:

$$s_k = T(r_k) = \sum_{j=0}^k p(r_j), \quad k = 0, 1, \dots, L-1$$

Przykład: obraz 64×64 , $n=4096$ pikseli, $L=8$ poziomów jasności (3 bity)

$Q[r_k]$	n_k	$p(r_k) = n_k/n$
$r_0 = 0$	790	0,19
$r_1 = 1/7$	1023	0,25
$r_2 = 2/7$	850	0,21
$r_3 = 3/7$	656	0,16
$r_4 = 4/7$	329	0,08
$r_5 = 5/7$	245	0,06
$r_6 = 6/7$	122	0,03
$r_7 = 1$	81	0,02

s_k	$Q[s_k]$	n'_k	$p(s_k)$
$s_0 = T(r_0) = 0.19$	1/7	790	0.19
$s_1 = T(r_1) = 0.19 + 0.25 = 0.44$	3/7	1023	0.25
$s_2 = T(r_2) = 0.65$	5/7	850	0.21
$s_3 = T(r_3) = 0.81$	6/7	985	0.24
$s_4 = T(r_4) = 0.89$	6/7	448	0.11
$s_5 = T(r_5) = 0.95$	1		
$s_6 = T(r_6) = 0.98$	1		
$s_7 = T(r_7) = 1$	1		



Rys. Efekt wyrównania (i rozciągnięcia) histogramu obrazu

Widmo Fouriera 2-wymiarowego sygnału dyskretnego

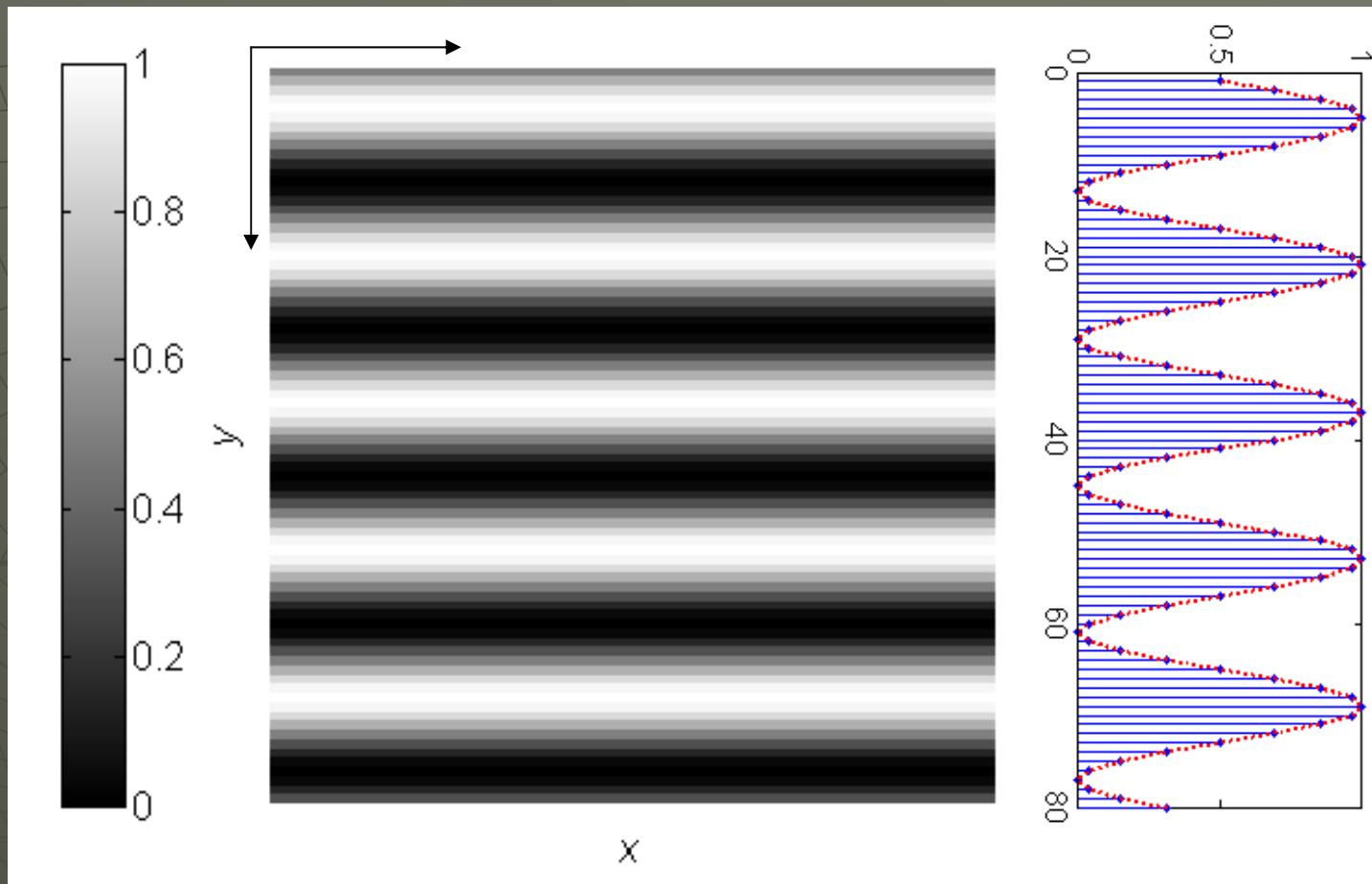
$$X(e^{j\Omega_x}, e^{j\Omega_y}) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x(m, n) \cdot e^{-jm\Omega_x} e^{-jn\Omega_y}, \quad \Omega_x = \frac{\omega}{f_{sx}}, \quad \Omega_y = \frac{\omega}{f_{sy}}$$

Częstotliwość przestrzenna $f = \omega/2\pi$ jest liczbą zmian intensywności na jednostkowym odcinku obrazu.

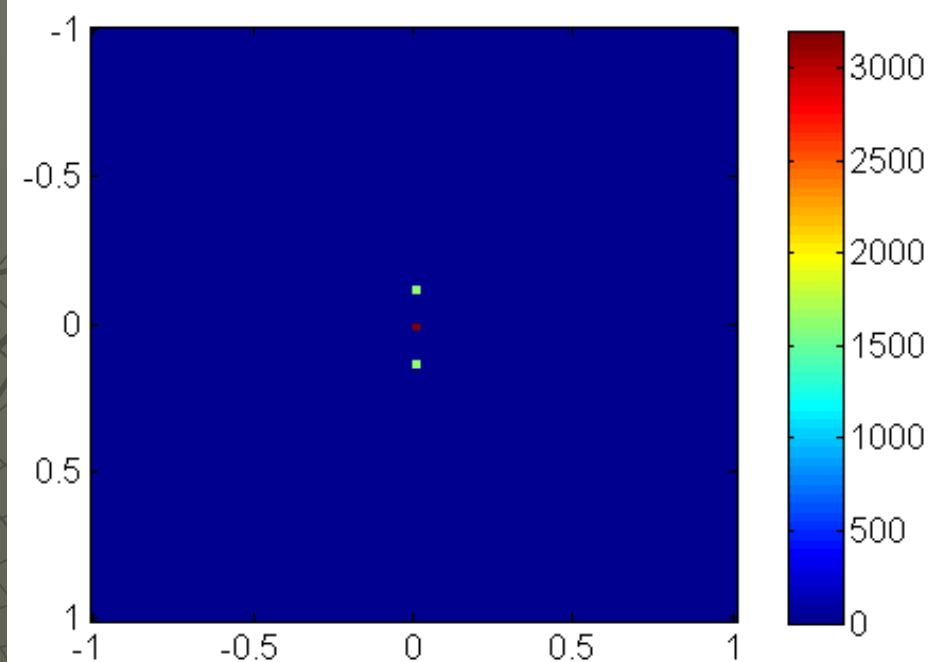
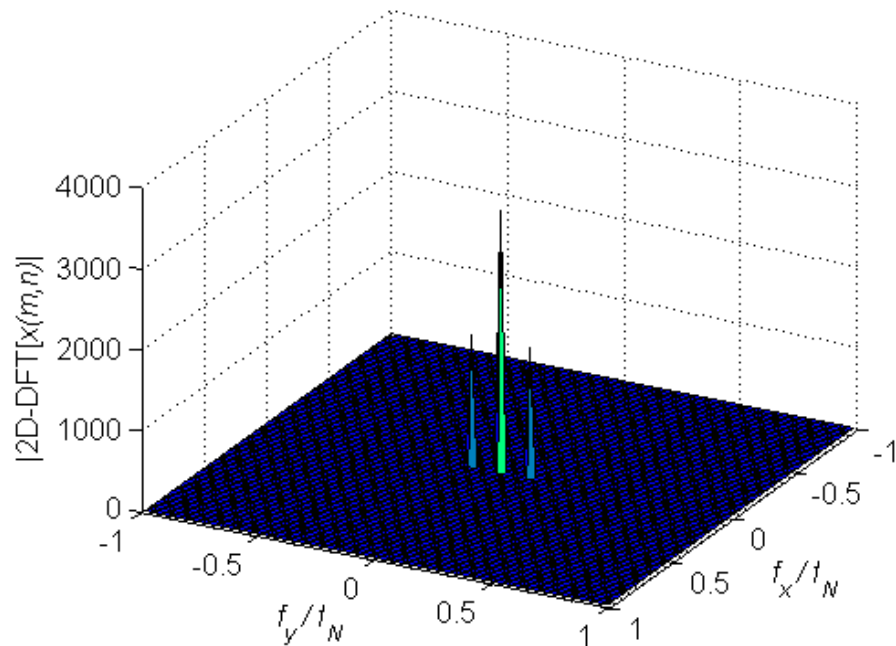
Widmo przestrzenne jest niezmiennicze względem ustawienia (obrotu) obrazu, co m.in. ułatwia rozpoznawanie obrazów w dziedzinie częstotliwości. Jednakowe częstotliwości przestrzenne obrazów tworzą na wykresie współśrodkowe okręgi o środkach w punkcie odpowiadającym częstotliwości zero.

Dwuwymiarowe dyskretne $M \times N$ -punktowe przekształcenie Fouriera (DFT2) obrazu $x(m, n)$ definiuje się jako wynik próbkowania widma ciągłego w punktach $\Omega_{xk} = 2\pi k/M$, $k=0, 1, \dots, M-1$ i $\Omega_{yl} = 2\pi l/N$, $l=0, 1, \dots, N-1$:

$$X(k, l) = DFT[x(m, n)] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m, n) \cdot e^{-jm\Omega_{xk}} \cdot e^{-jn\Omega_{yl}}$$



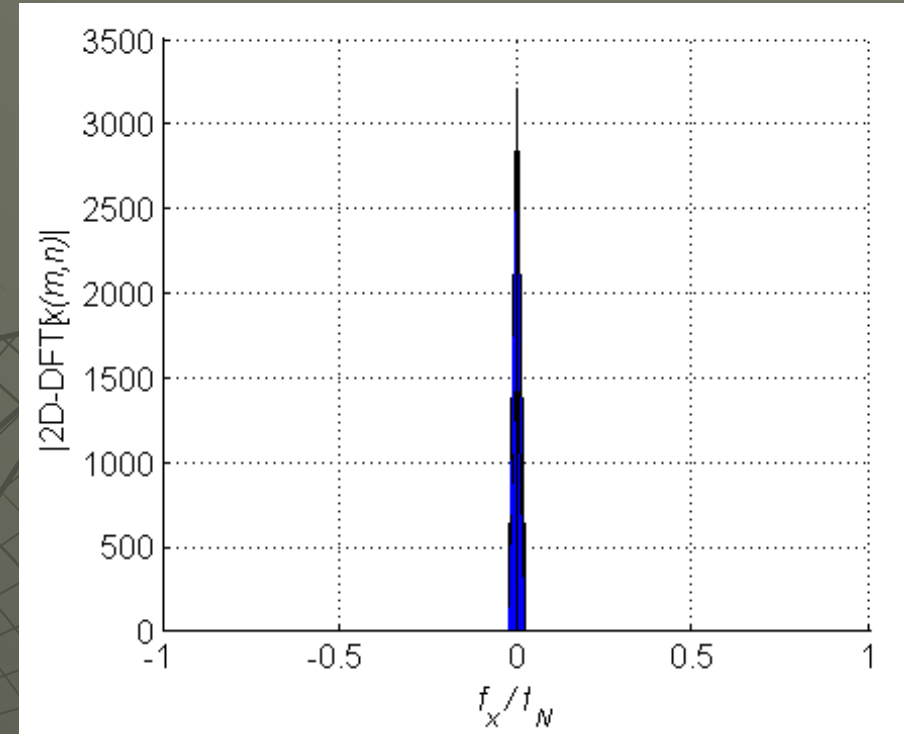
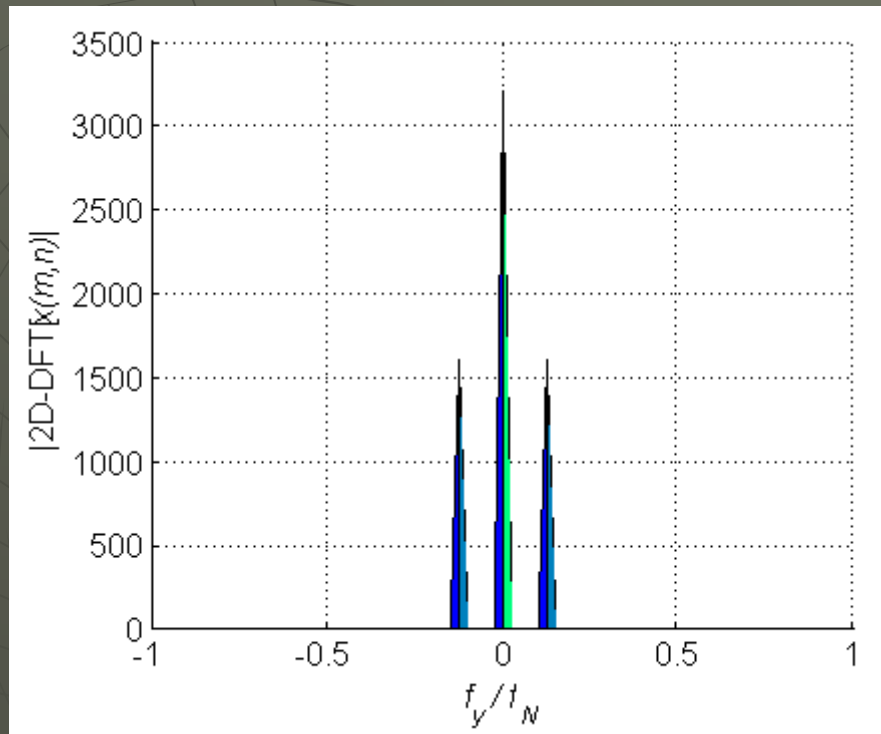
Rys. Obraz poziomych pasów (80×80 pikseli) z płynną zmianą wartości pikseli od 0 (czarny) do 1 (biały), mapa odcieni szarości i przebieg zmiany intensywności wzdłuż osi OY . Taki sygnał obrazu ma zawsze składową stałą; 5 okresów zmian (16 pikseli/okres) odpowiada $1/8$ częstotliwości Nyquista (2 piksele/okres)



wykres 3D - $|DFT|$ na osi OZ

mapa kolorów (*jet*)

Rys. Dwuwymiarowe widmo DFT2 obrazu poziomych pasów. Wartości częstotliwości unormowane do częstotliwości Nyquista f_N . Moduł $|X(k, l)|$ stanowi *dyskretne widmo amplitudowe obrazu*. Punkt centralny reprezentuje składową stałą sygnału obrazu (częstotliwość 0 w centrum widma)



Rys. Rzuty widma DFT2 na oś OX i na oś OY . Boczne prążki częstotliwość $f_y = 1/8$ reprezentują częstotliwość poziomych pasów. Widmo wykazuje brak zmian sygnału wzdłuż osi OX .

Odwrotne dwuwymiarowe dyskretne przekształcenie Fouriera umożliwia jednoznaczne odtworzenie pikseli obrazu na podstawie współczynników widma dyskretnego:

$$x(m, n) = IDFT[X(k, l)] = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X(k, l) \cdot e^{jm\Omega_{xk}} e^{jn\Omega_{yl}}$$

Liniowa filtracja obrazu

Liniowa filtracja sygnału dwuwymiarowego $x(m,n)$ to *splot sygnału z dwuwymiarową charakterystyką impulsową* $h(m,n)$ filtra:

$$y(m,n) = h(m,n) * x(m,n) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} h(k,l) \cdot x(m-k, n-l)$$

Charakterystykę impulsową filtra to jego odpowiedź na pobudzenie dwuwymiarową funkcją delta Kroneckera (przy zerowych warunkach początkowych):

$$\delta(m,n) = \begin{cases} 1 & \text{dla } m = n = 0 \\ 0 & \text{dla pozostałych par } (m,n) \end{cases}$$

Ważne: W przypadku obrazów możliwa jest realizacja filtracji *nieprzyczynowej*, tzn. za pomocą filtrów, których $h(m,n)$ ma wartości niezerowe dla ujemnych m i/lub n . Wartość piksela po filtracji zależy wtedy od wartości pikseli sąsiednich, zarówno o indeksach mniejszych jak i większych od przetwarzanego.

W praktyce stosuje wyłącznie filtry o skończonej odpowiedzi impulsowej SOI (FIR) i charakterystyce fazowej $\varphi(\Omega)=0$.

Charakterystykę impulsową $h(m,n)$ (macierz o skończonych wymiarach) nazywa się *maską* albo *jądrem* filtra.

Jeżeli $x(m,n)$ jest obrazem o wymiarach $M \times N$, a $h(i,j)$ jest maską filtra o wymiarach $I \times J$ ($I=2a+1$ – liczba wierszy, $J=2b+1$ – liczba kolumn), to wartość splotu obrazu z maską w punkcie (m,n) wyznacza się z zależności:

$$y(m,n) = \sum_{i=-a}^a \sum_{j=-b}^b h(i,j) \cdot x(m+i,n+j)$$

Obraz

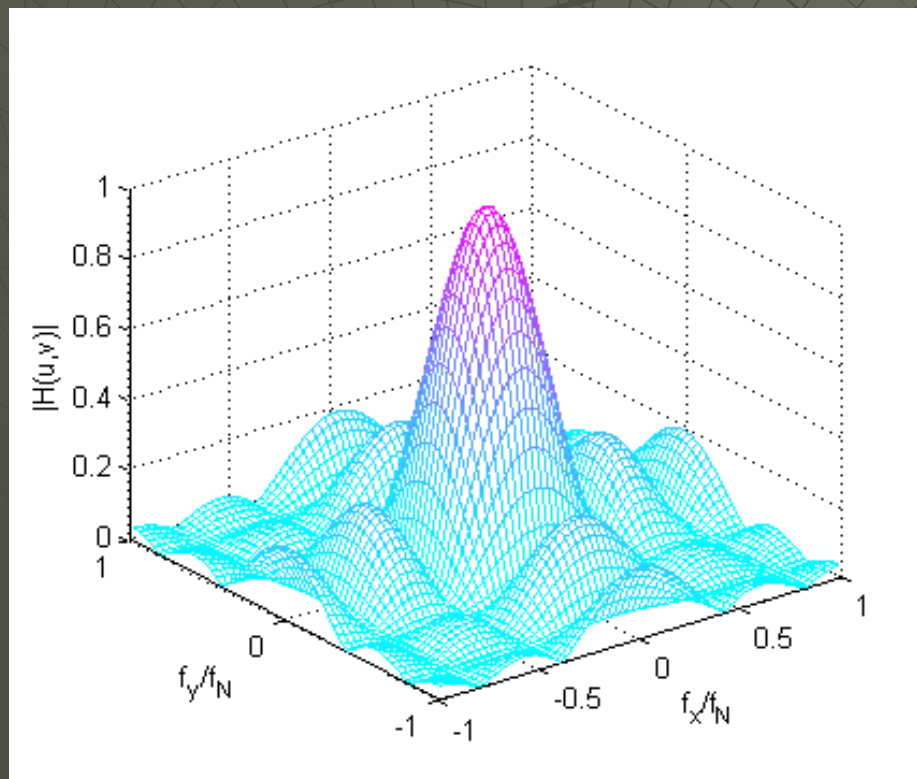
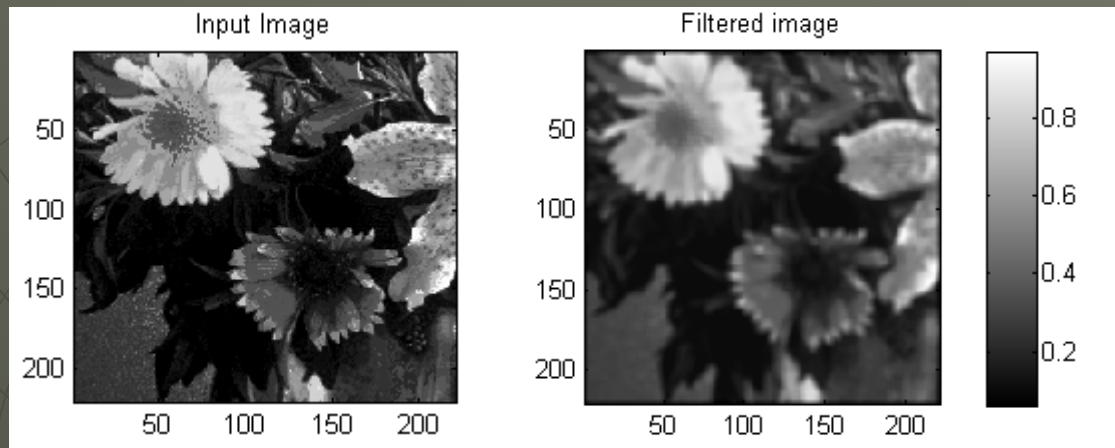
3	7	6	12	...
8	7	10	6	...
14	13	8	5	...
1	12	5	2	...
2	8	2	6	...
...

Maska

h =

$h(-1,-1) = -1$	$h(0,-1) = -1$	$h(1,-1) = -1$
$h(-1,0) = -1$	$h(0,0) = 8$	$h(1,0) = -1$
$h(-1,1) = -1$	$h(0,1) = -1$	$h(1,1) = -1$

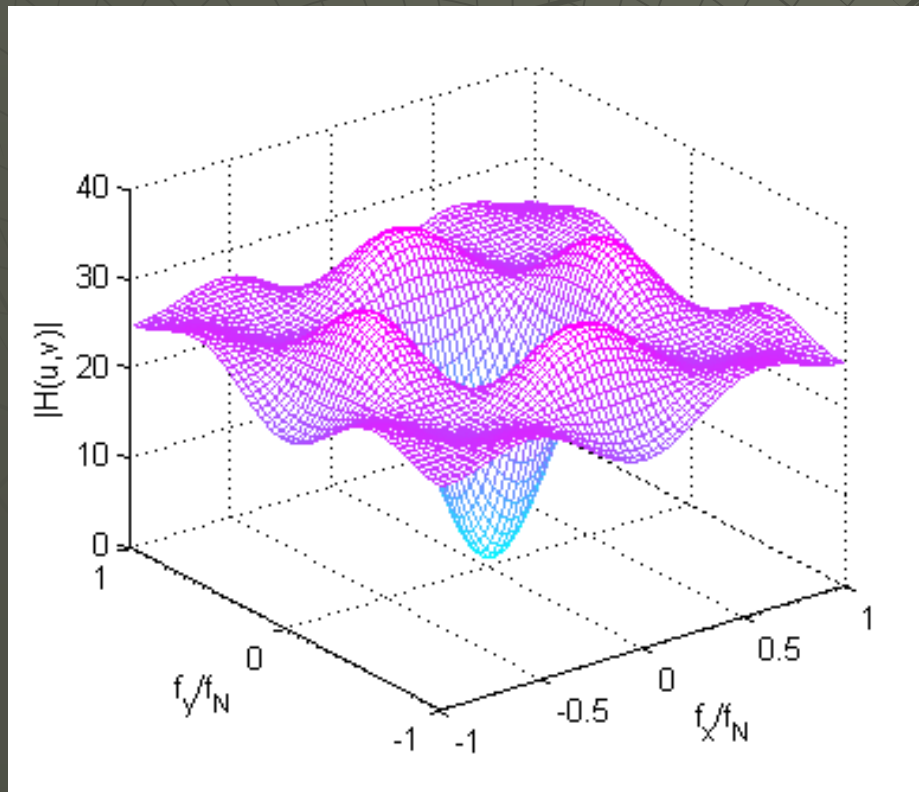
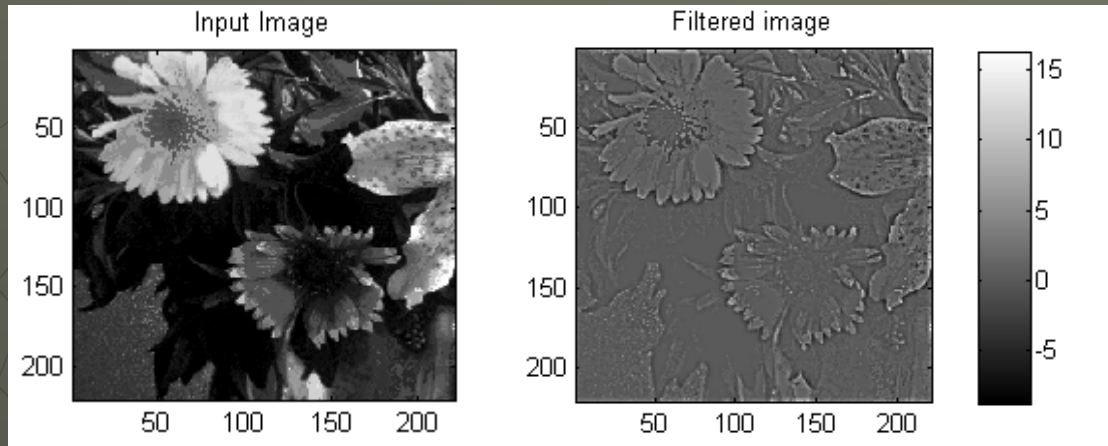
$$y(2,3) = 8 \cdot (-1) + 7 \cdot (-1) + 10 \cdot (-1) + 14 \cdot (-1) + 13 \cdot 8 + 8 \cdot (-1) + 1 \cdot (-1) + 12 \cdot (-1) + 5 \cdot (-1) = 39$$



$\mathbf{h} =$

0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04

Rys. Filtracja dolnoprzepustowa 2D: obraz oryginalny i po filtracji, maska filtra, charakterystyka amplitudowa filtra



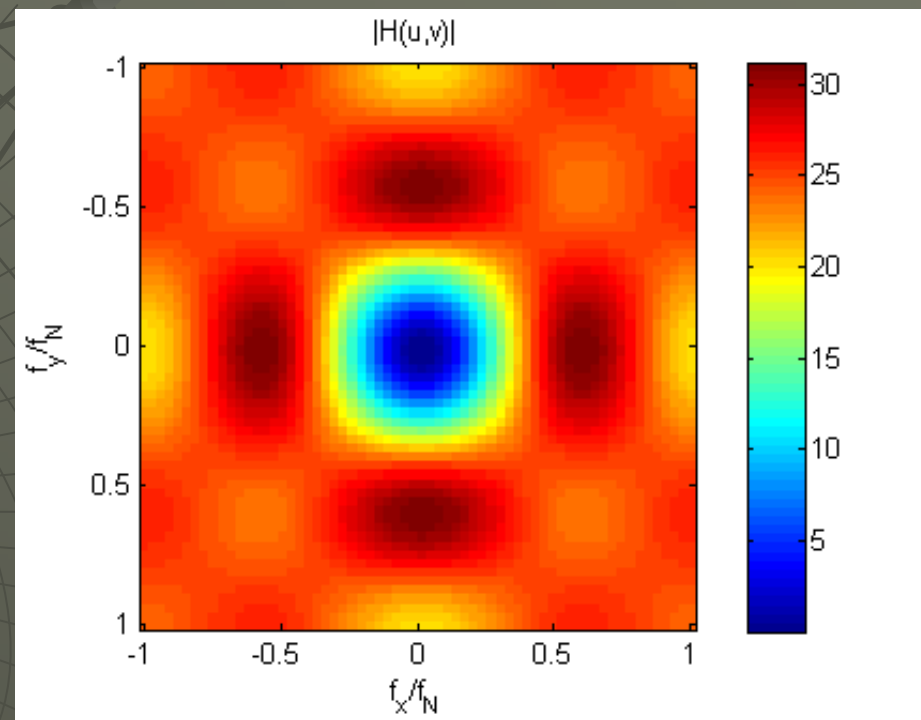
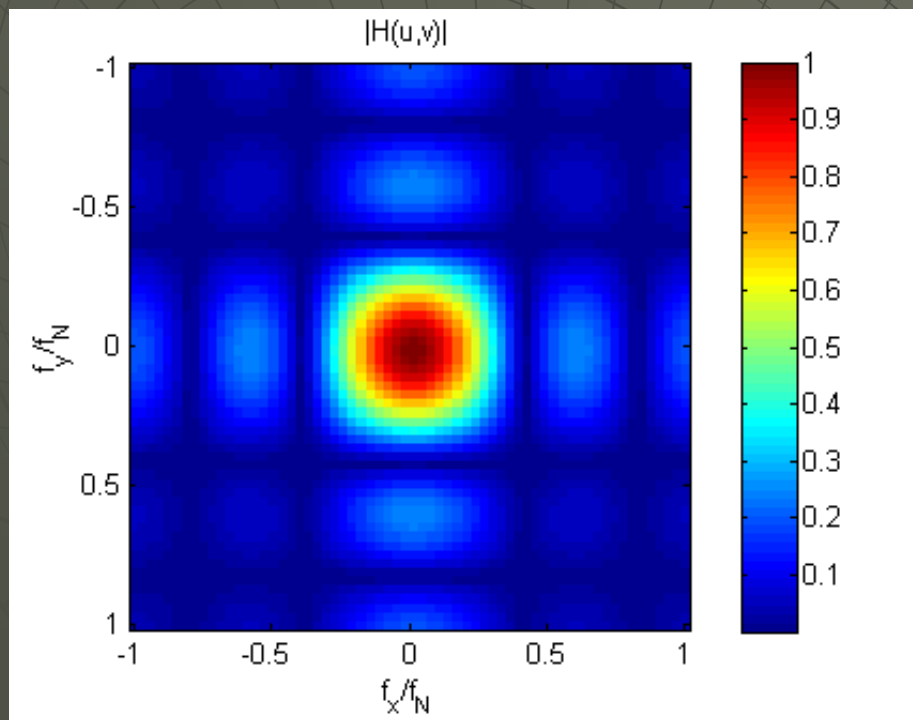
$\mathbf{h} =$

-1	-1	-1	-1	-1
-1	-1	-1	-1	-1
-1	-1	24	-1	-1
-1	-1	-1	-1	-1
-1	-1	-1	-1	-1

Rys. Filtracja górnoprzepustowa 2D: obraz oryginalny i po filtracji, maska filtra, charakterystyka amplitudowa filtra

0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04

-1	-1	-1	-1	-1
-1	-1	-1	-1	-1
-1	-1	24	-1	-1
-1	-1	-1	-1	-1
-1	-1	-1	-1	-1



Rys. Charakterystyki amplitudowe filtra LP i HP jako mapy barwne

Przykłady masek filtrów

$$y(m, n) = \frac{1}{w} h(m, n) * x(m, n)$$

- filtry LP

1	1	1
1	a	1
1	1	1

$$w = a + 8$$

$$a = 0, 1, 2, 4$$

1	a	1
a	a^2	a
1	a	1

$$w = (a + 2)^2$$

$$a = 0, 1, 2, 4$$

- filtry HP

1	-2	1
-2	4(5)	-2
1	-2	1

$$w = 1$$

1	-1	1
-1	4(5)	-1
1	-1	1

$$w = 1$$

operatory Laplace'a (wyostrzające)

$$L[J(x, y)] = \frac{\partial^2 J}{\partial x^2} + \frac{\partial^2 J}{\partial y^2}, \quad \frac{\partial^2 J}{\partial x^2} \approx [J(x+1, y) - J(x, y)] - [J(x, y) - J(x-1, y)] = J(x-1, y) - 2J(x, y) + J(x+1, y)$$

0	0	0
-1	2	-1
0	0	0

+

0	-1	0
0	2	0
0	-1	0

=

0	-1	0
-1	4	-1
0	-1	0

operatory Prewitta

1	1	1
0	0	0
-1	-1	-1

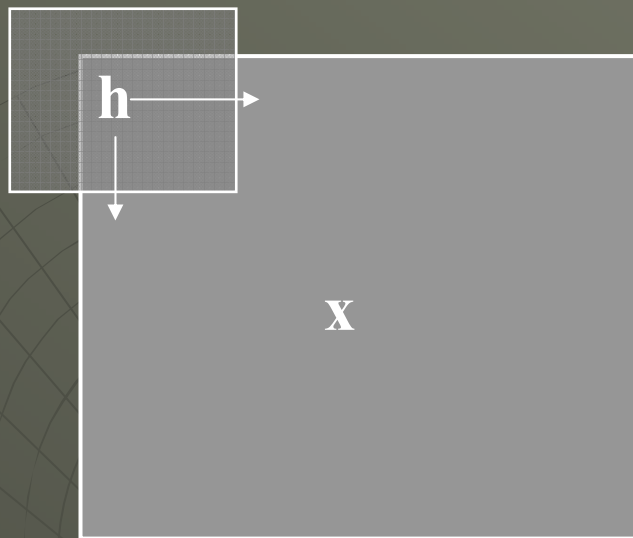
1	0	-1
1	0	-1
1	0	-1

5 × 5 Laplacian kernel

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

7 × 7 Mexican hat kernel

0	0	-1	-1	-1	0	0
0	-1	-3	-3	-3	-1	0
-1	-3	0	7	0	-3	-1
-1	-3	7	24	7	-3	-1
-1	-3	0	7	0	-3	-1
0	-1	-3	-3	-3	-1	0
0	0	-1	-1	-1	0	0



Rys. Wystawanie maski filtra poza obraz powodujące ewentualne efekty brzegowe filtracji.

Ponieważ w punktach przy krawędziach obrazu maska filtra wystaje poza obraz, przyjmuje się jedną z następujących możliwości wyznaczania splotu:

- wartości pikseli poza obrazem są równe zero. Obraz będący wynikiem splotu ma te same wymiary co oryginał, ale na brzegach powstają zafałszowania,
- splot oblicza się tylko w tych punktach, w których maska nie wystaje poza obraz. Obraz po filtracji jest mniejszy od oryginalnego,
- uzupełnia się brakujące wartości, np. metodą lustrzanego odbicia względem krawędzi obrazu, żeby uniknąć nieciągłości. Obraz po filtracji ma wymiary oryginału, a zafałszowania są mniejsze niż w pierwszym przypadku.

Filtracja liniowa w dziedzinie częstotliwości zespolonej

Transmitancja filtra liniowego to *dwuwymiarowa dyskretna transformata* Z jego odpowiedzi impulsowej (u, v – zmienne zespolone):

$$H(u, v) = Z[h(m, n)] = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} h(m, n) \cdot u^{-m} \cdot v^{-n}$$

W dziedzinie zespolonej splotowi odpowiada mnożenie transformat:

$$Y(u, v) = H(u, v) \cdot X(u, v)$$

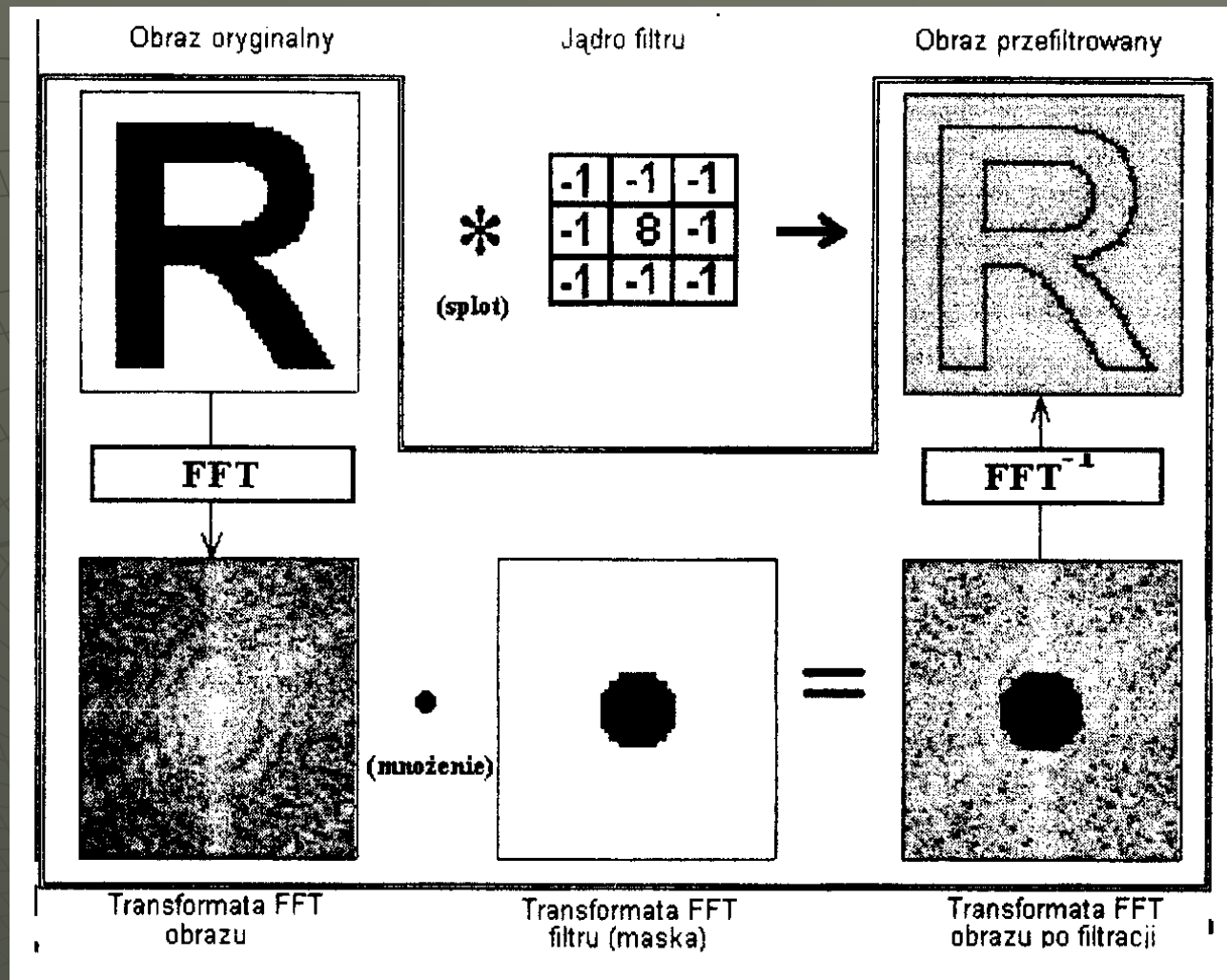
gdzie $Y(u, v)$ – transformata sygnału (obrazu) wyjściowego.
Analogiczna zależność zachodzi dla dyskretnych transformat Fouriera

$$Y(k, l) = H(k, l) \cdot X(k, l)$$

gdzie $H(k, l) = DFT[h(i, j)]$, $X(k, l) = DFT[x(i, j)]$ są $K \times L$ -punktowymi transformatami DFT2, tzn. $k=0, 1, \dots, K-1$, $l=0, 1, \dots, L-1$.

Obraz po filtracji uzyskuje się przez wyznaczenie odwrotnej transformaty Fouriera:

$$y(m, n) = IDFT[Y(k, l)]$$



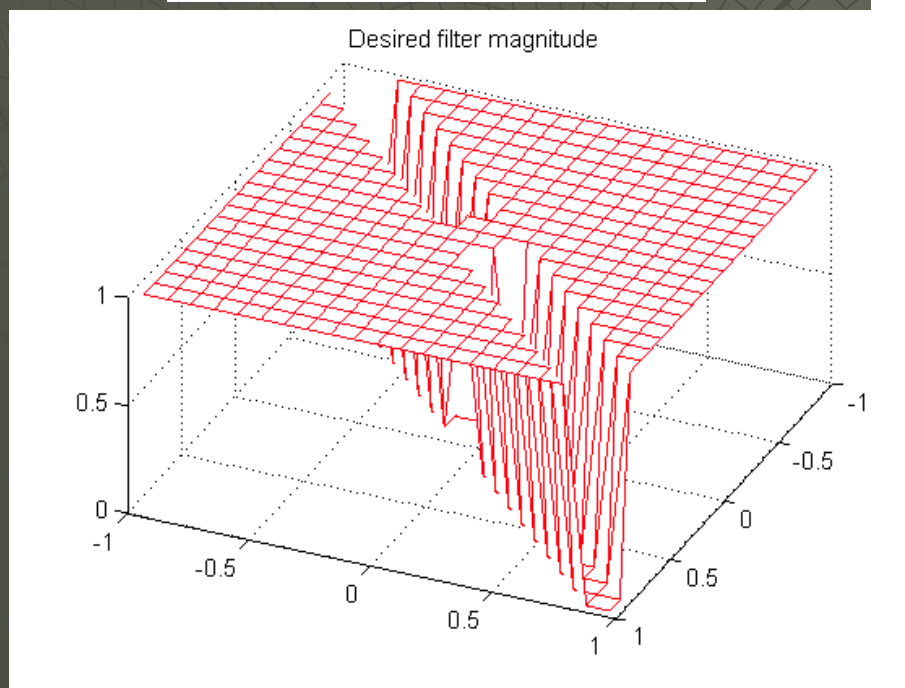
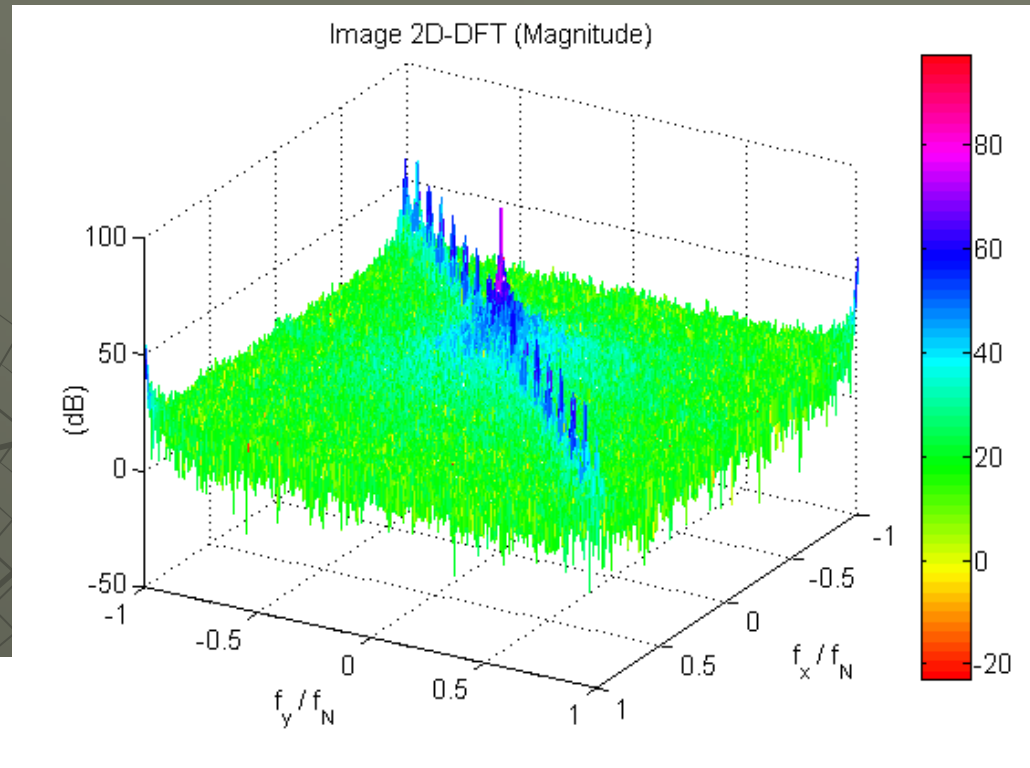
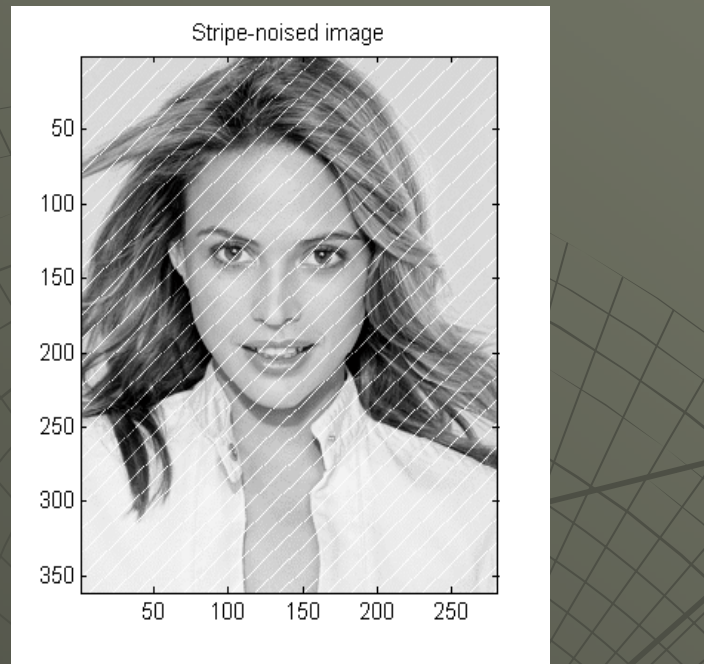
Rys. Filtracja w dziedzinie obrazu (splot $h(m,n)*x(m,n)$) i częstotliwości (mnożenie transformat $IDFT[H(k,l)X(k,l)]$ – operacje objęte ramką)

Maska, równanie różnicowe i transmitancja filtra liniowegoPrzykład:

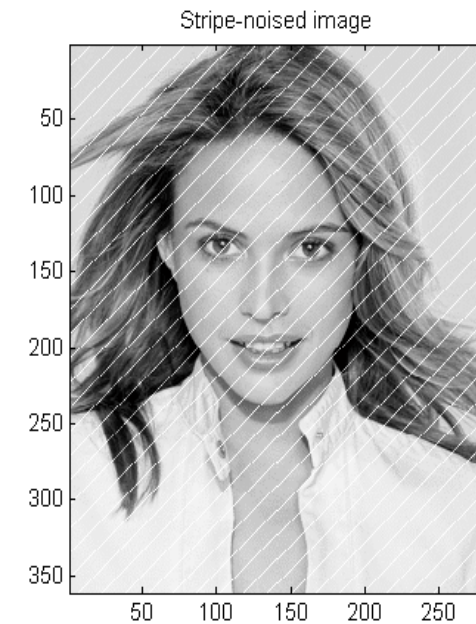
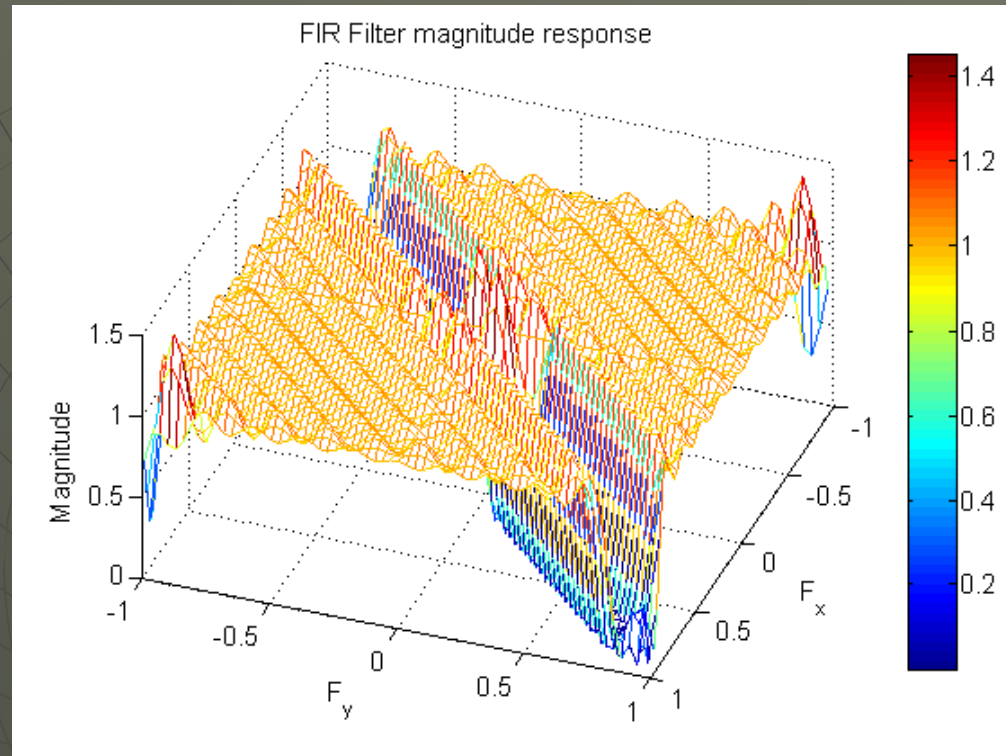
$$\mathbf{h} = \begin{array}{|c|c|c|} \hline 1 & -2 & 1 \\ \hline -2 & 4 & -2 \\ \hline 1 & -2 & 1 \\ \hline \end{array}$$

$$y(m, n) = x(m-1, n-1) - 2x(m, n-1) + x(m+1, n-1) - \\ - 2x(m-1, n) + 4x(m, n) - 2x(m+1, n) + \\ + x(m-1, n+1) - 2x(m, n+1) + x(m+1, n+1)$$

$$H(u, v) = u^{-1}v^{-1} - 2v^{-1} + uv^{-1} - 2u^{-1} + 4 - 2u + u^{-1}v - 2v + uv$$



Przykład: Projektowanie filtra pasmowozaporowego do usuwania zakłóceń okresowych w formie ukośnych pasów: obraz i jego widmo, pożądana ch-ka $|H_0(u,v)|$ filtra



Rys. Charakterystyka amplitudowa zaprojektowanego filtra FIR (maska 21×21), obraz przed i po filtracji

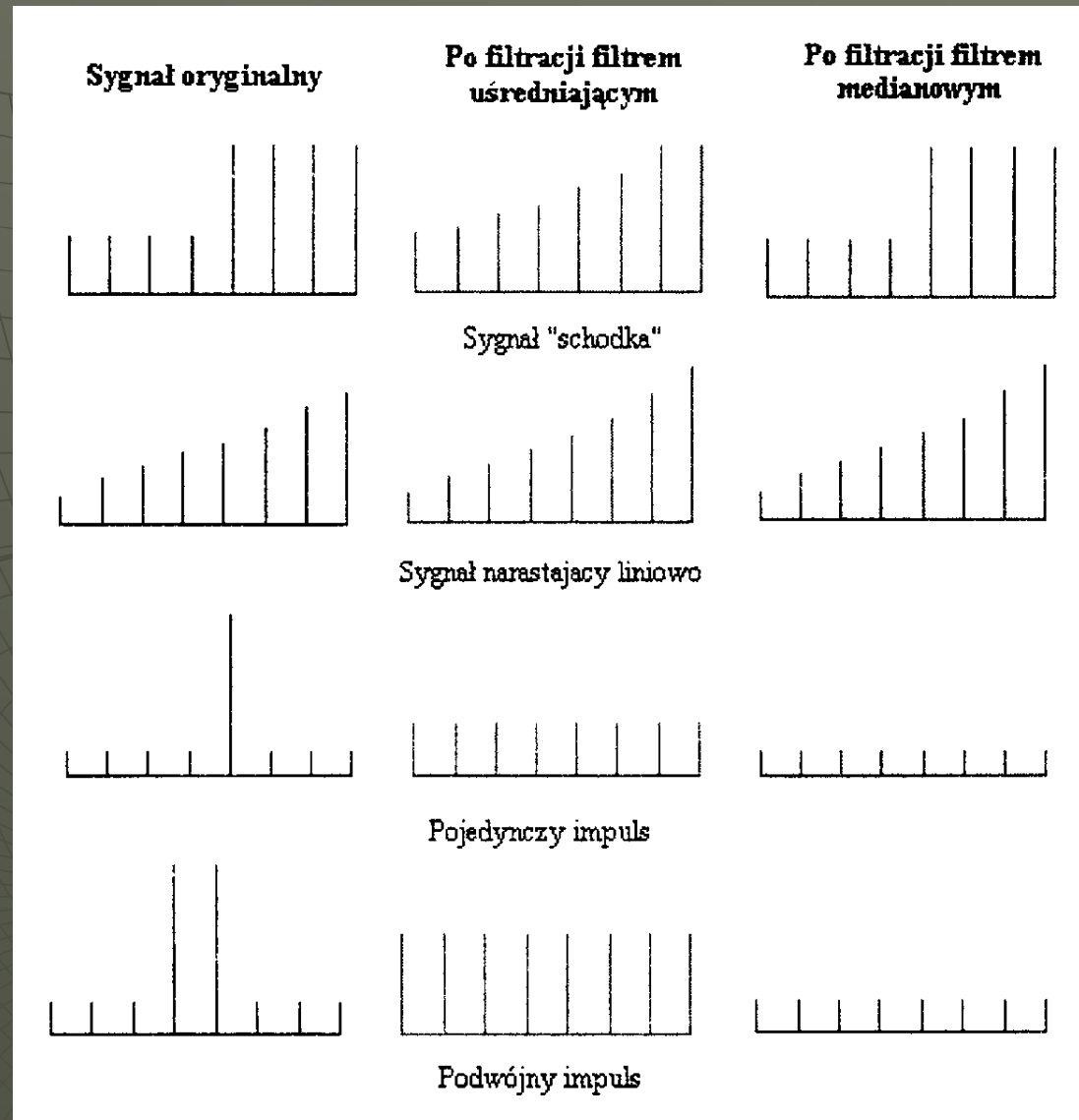
Nieliniowa filtracja medianowa

Medianą dla wycinka sygnału o długości $N=2L+1$ jest $L+1$ (środkowa) próbka w ciągu powstałym po uporządkowaniu ich w porządku rosnącym wartości (*tyle samo próbek jest mniejszych lub równych i większych lub równych od mediany*; w przypadku parzystej liczby próbek jako medianę można wziąć średnią z dwóch środkowych).

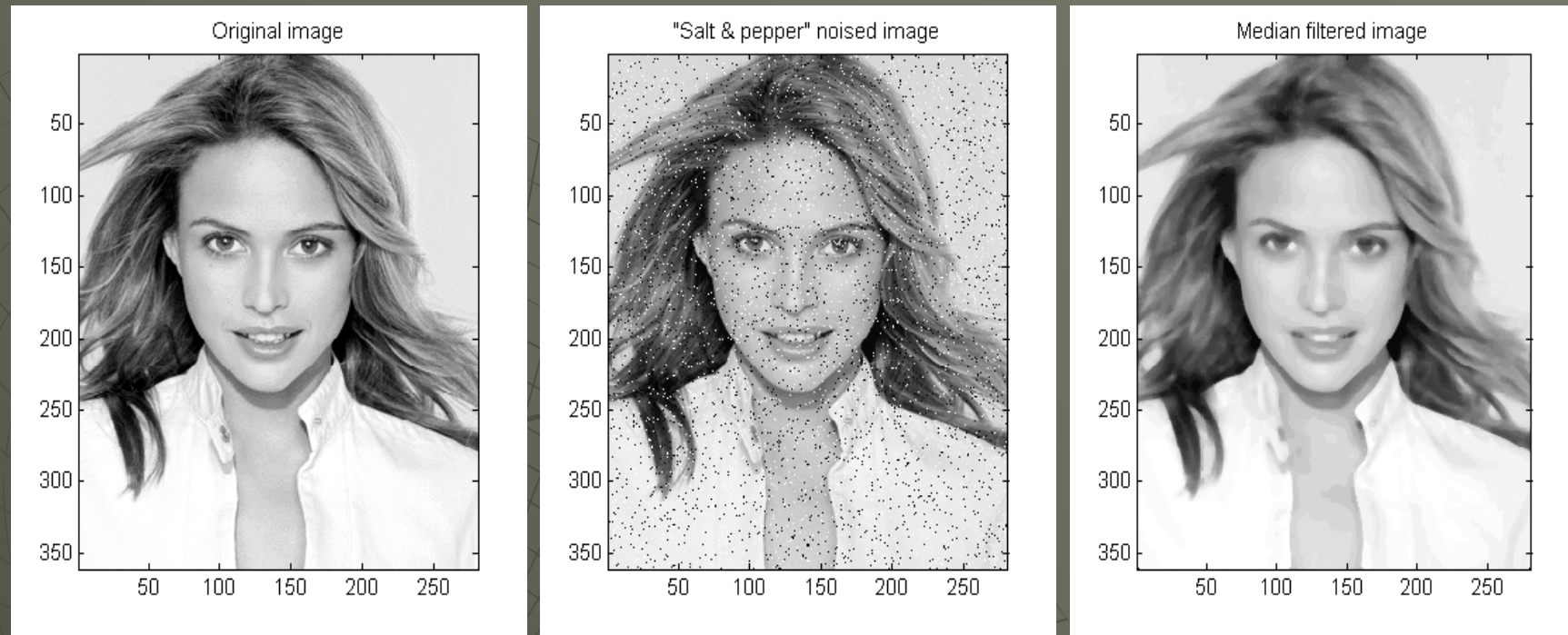
Filtry medianowe są filtrami nieliniowymi i działają podobnie do liniowych filtrów dolnoprzepustowych LP.

- Filtry medianowe mają dobre właściwości zachowywania krawędzi na obrazie (stąd określenie *edge-preserving filter*).
- Zaletą filtrów medianowych jest całkowite usuwanie silnych zakłóceń impulsowych (liniowe filtry LP tylko takie zakłócenia uśredniają).
- **Jedynym parametrem filtra medianowego jest rozmiar maski.**

Przykład: Jeżeli 2-wymiarowy filtr medianowy ma maskę $h(i,j)$ o wymiarach 3×3 , to dla punktu $x(3,2)$ obrazu z przykładu w obręb maski wchodzi piksele, których uszeregowanie daje wartości: **1, 5, 7, 8, 8, 10, 12, 13, 14**. W wyniku filtracji medianowej otrzymujemy wartość środkową ciągu, tzn. $y(3,2)=8$.



Rys. Porównanie efektów działania liniowego filtra uśredniającego (LP) z filtrem medianowym dla sygnału jednowymiarowego



Rys. Efekt działania filtra medianowego o masce 5×5 na obrazie z zakłóceniami impulsowymi (tzw. szum *salt & pepper*)

Korelacja obrazów

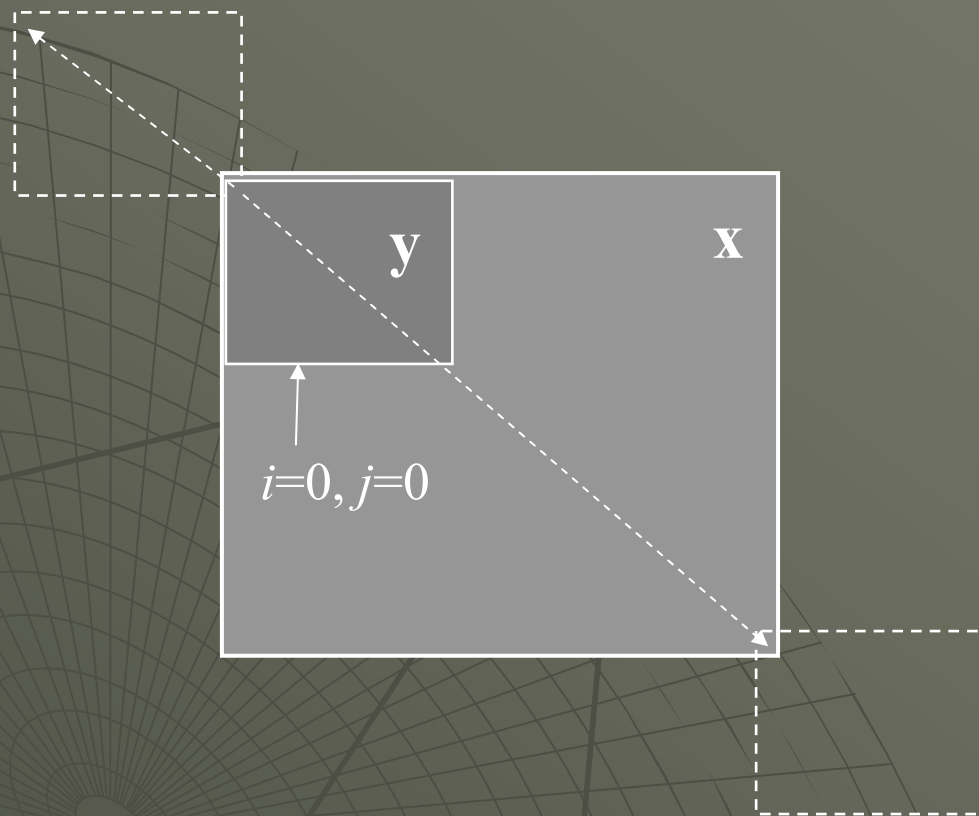
Dwuwymiarową korelację wzajemną obrazów (sygnałów dwuwymiarowych) $x(m,n)$ i $y(m,n)$ określonych w postaci macierzy \mathbf{x} o wymiarach $M_x \times N_x$ i wzorca \mathbf{y} o mniejszych wymiarach $M_y \times N_y$ (indeksowanie od zera) określa się jako:

$$R_{xy}(i, j) = \frac{1}{M_y N_y} \sum_{m=0}^{M_x-1} \sum_{n=0}^{N_x-1} x(m, n) \cdot y(m+i, n+j)$$

$$-(M_y - 1) \leq i \leq M_x - 1$$

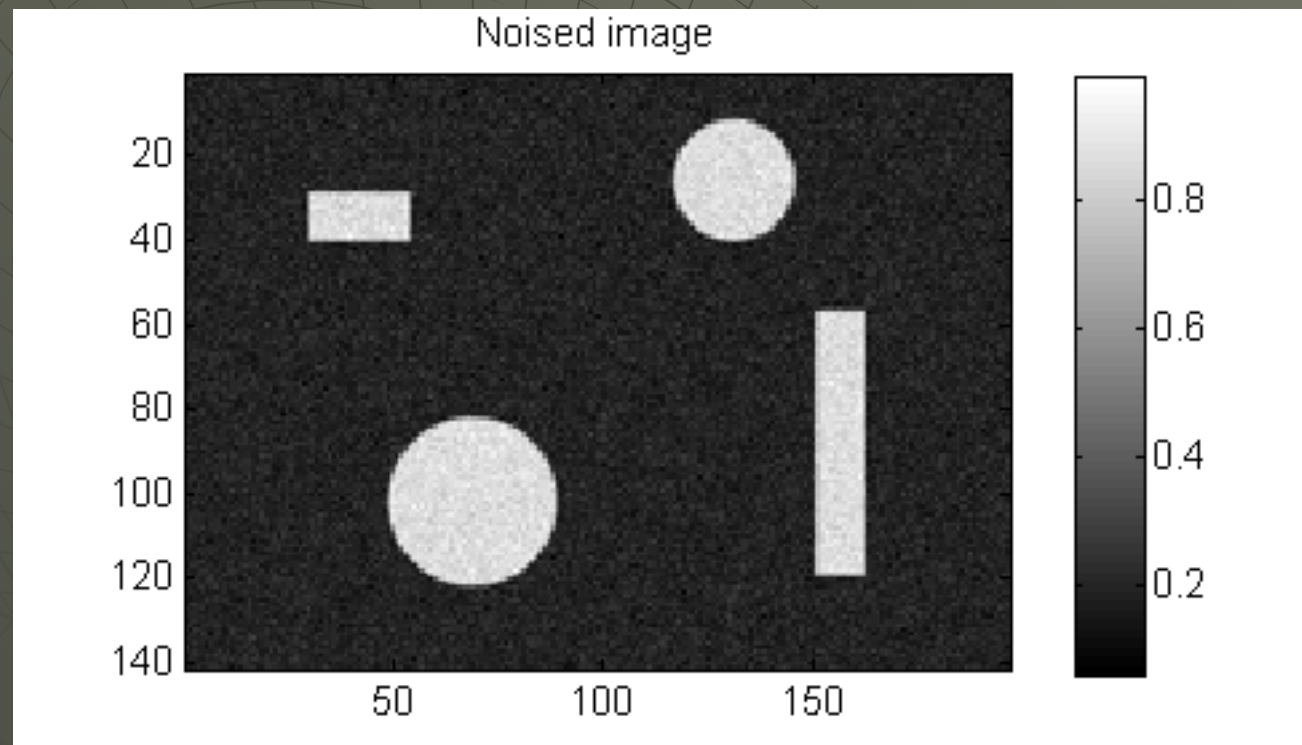
$$-(N_y - 1) \leq j \leq N_x - 1$$

Macierz korelacji \mathbf{R}_{xy} ma wymiary $(M_x + M_y - 1) \times (N_x + N_y - 1)$. Współczynniki korelacji osiągają największe wartości dla przesunięć (i, j) , dla których wzorzec jest podobny do odpowiedniego fragmentu obrazu.

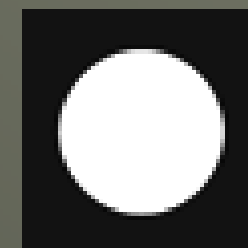


Rys. Sposób obliczania korelacji wzajemnej dwóch macierzy. Wyznaczanie korelacji polega na przesuwaniu jednego obrazu („wzorca”) na tle drugiego i obliczaniu dla każdego przesunięcia $R_{xy}(i,j)$ jako sumy iloczynów nakładających się wyrazów (wartości pikseli) macierzy **x** i **y**

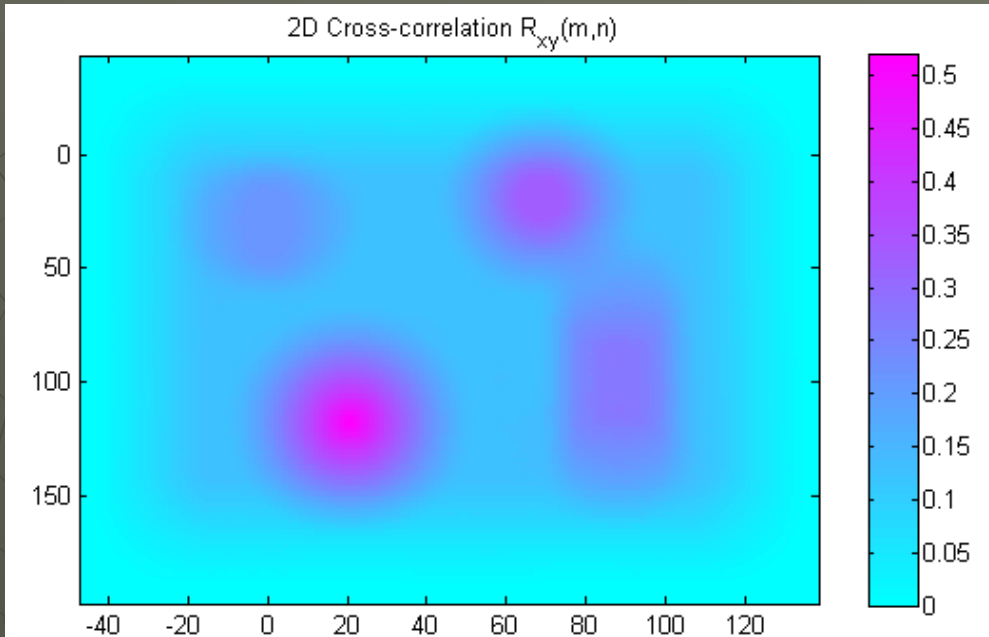
Przykład: Obliczanie korelacji obrazu zakłóconego szumem gaussowskim i wzorca (poszukiwanie elementów podobnych do wzorca na obrazie)



obraz

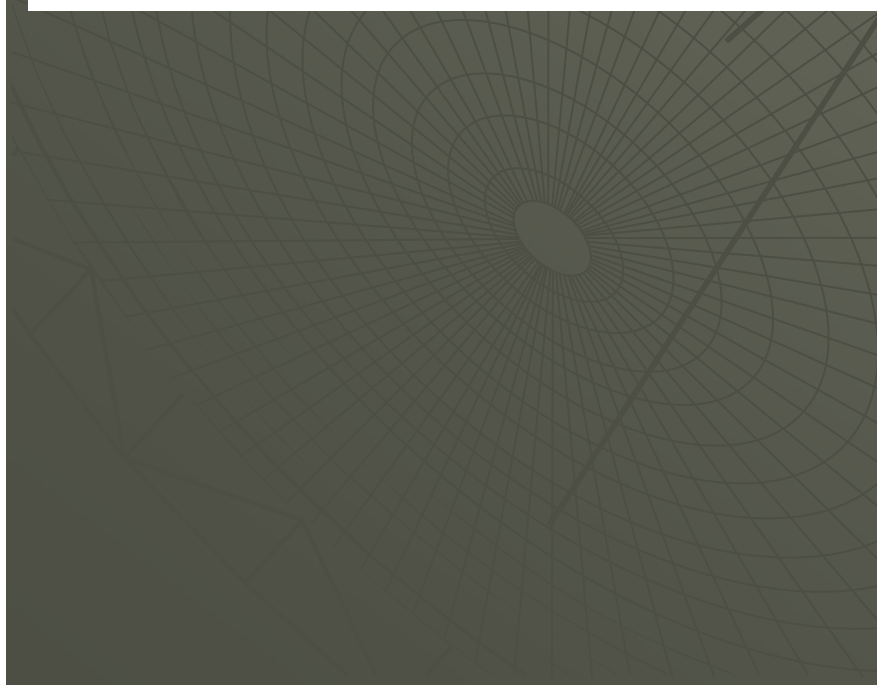
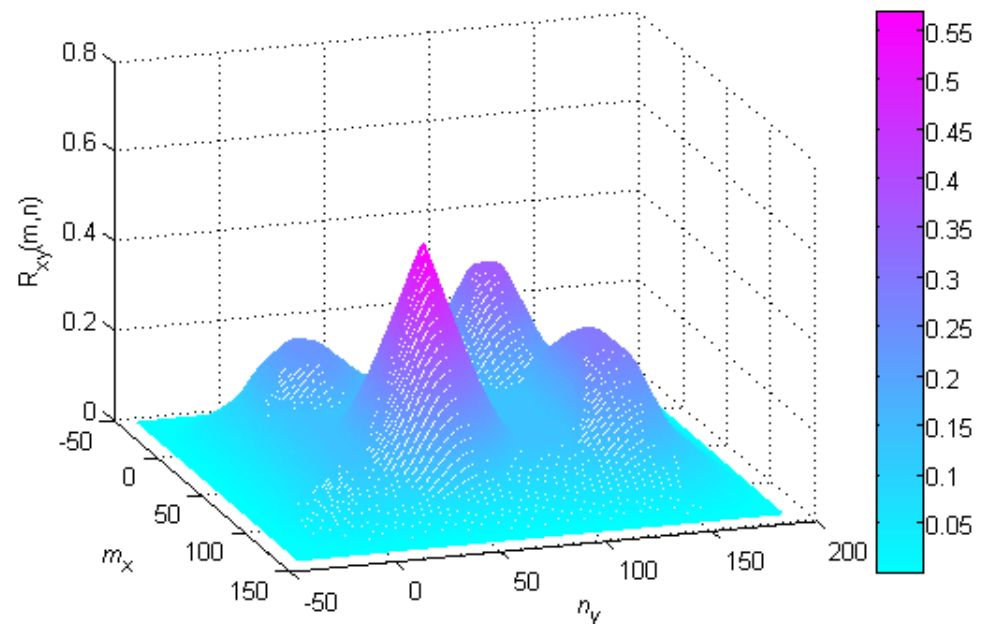


worzec



Rys. Korelacja wzajemna obrazów w formie mapy barw i wykresu 3-wymiarowego.

Korelacja $R_{xx}(0,0)$ odpowiada sytuacji, kiedy lewe górne rogi ($x(0,0)$ i $y(0,0)$) obu obrazów pokrywają się.



Dyskretna transformata kosinusowa DCT (Discrete Cosine Transform)

Transformatę kosinusową DCT dyskretnego sygnału $x(n)$ definiuje się jako:

$$DCT[x(n)] = X_c(k) = p(k) \sum_{n=0}^{N-1} x(n) \cdot \cos \frac{(2n+1)k\pi}{2N}, \quad k = 0, 1, \dots, N-1$$

$$p(k) = \begin{cases} \sqrt{1/N} & \text{dla } k = 0 \\ \sqrt{2/N} & \text{dla } k \neq 0 \end{cases}$$

- Dla sygnałów rzeczywistych współczynniki DCT mają wartości rzeczywiste.
- W DCT nie ma symetrii współczynników jak w transformacie Fouriera DFT, *wszystkie są od siebie niezależne.*
- W porównaniu z DFT transformata DCT ma lepsze właściwości skupiania energii sygnału (suma kwadratów niewielu początkowych współczynników DCT reprezentuje większość energii sygnału).

Odwrotna transformata kosinusowa:

$$x(n) = IDCT[X_C(k)] = p(n) \sum_{k=0}^{N-1} X_C(k) \cdot \cos \frac{(2k+1)n\pi}{2N}, \quad n = 0, 1, \dots, N-1$$

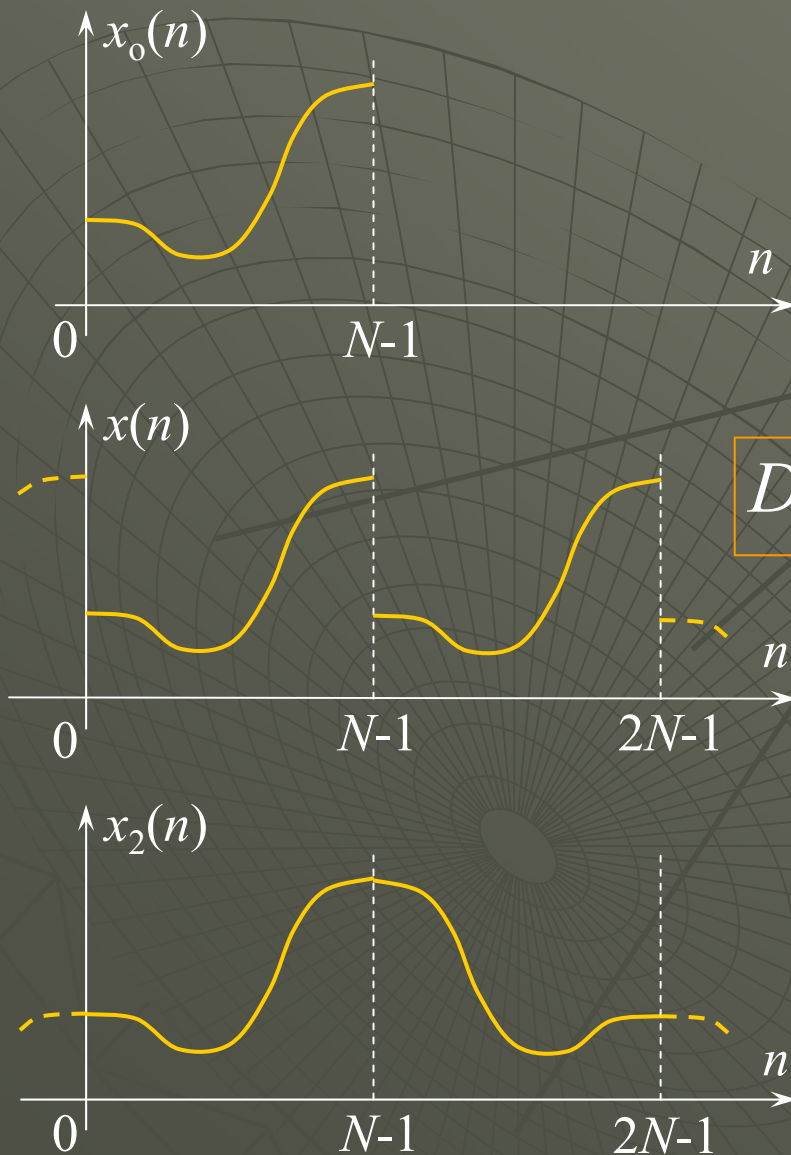
Związek pomiędzy DCT i DFT

Współczynniki N -punktowej transformaty kosinusowej $DCT_N[x_0(n)]$ są równe pierwszym N współczynnikom $2N$ -punktowej transformaty Fouriera $DFT_{2N}[x_2(n)]$ wektora

$$x_2(n) = \begin{cases} x(n) & \text{dla } 0 \leq n \leq N-1 \\ x(2N-n-1) & \text{dla } N \leq n \leq 2N-1 \end{cases}$$

stanowiącego *lustrzanie symetryczną* wersję 2 wycinków powielonego okresowo $x_0(n)$. (Transformata DFT symetrycznego wycinka $x_2(n)$ ma współczynniki rzeczywiste.)

Transformata DCT odcinka sygnału z nieciągłościami na końcach daje taki sam wynik jak transformata DFT sygnału bez nieciągłości.



Rys. Interpretacja odcinka sygnału jako fragmentu sygnału okresowego przez transformaty DFT i DCT – powielenie DFT z nieciągłościami i powielenie DCT bez nieciągłości

Równość transformat:

$$DCT_N[x(n)] = \{DFT_{2N}[x_2(n)]\}_{k=0..(N-1)}$$

$$DCT_N[x(n)] = X_C(k)$$

$$DFT_{2N}[x_2(n)] = X_F(k)$$

$$k = 0, 1, \dots, N-1$$

DFT odcinka z nieciągłościami zawiera rozmycie widma wprowadzające fałszywe składowe o wyższych częstotliwościach.

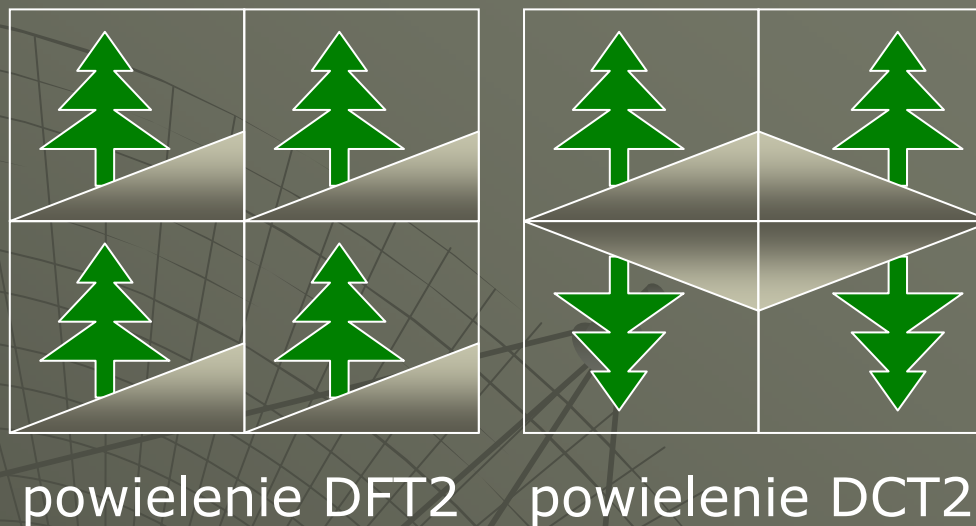
Dwuwymiarowa transformata kosinusowa (DCT2)

Dwuwymiarową transformatę kosinusową definiuje się jako:

$$\begin{aligned}
 DCT[x(m, n)] &= X_C(k, l) = \\
 &= p(k)q(l) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m, n) \cos \frac{(2m+1)k\pi}{2M} \cos \frac{(2n+1)l\pi}{2N}, \\
 &0 \leq k \leq M-1, \quad 0 \leq l \leq N-1
 \end{aligned}$$

gdzie:

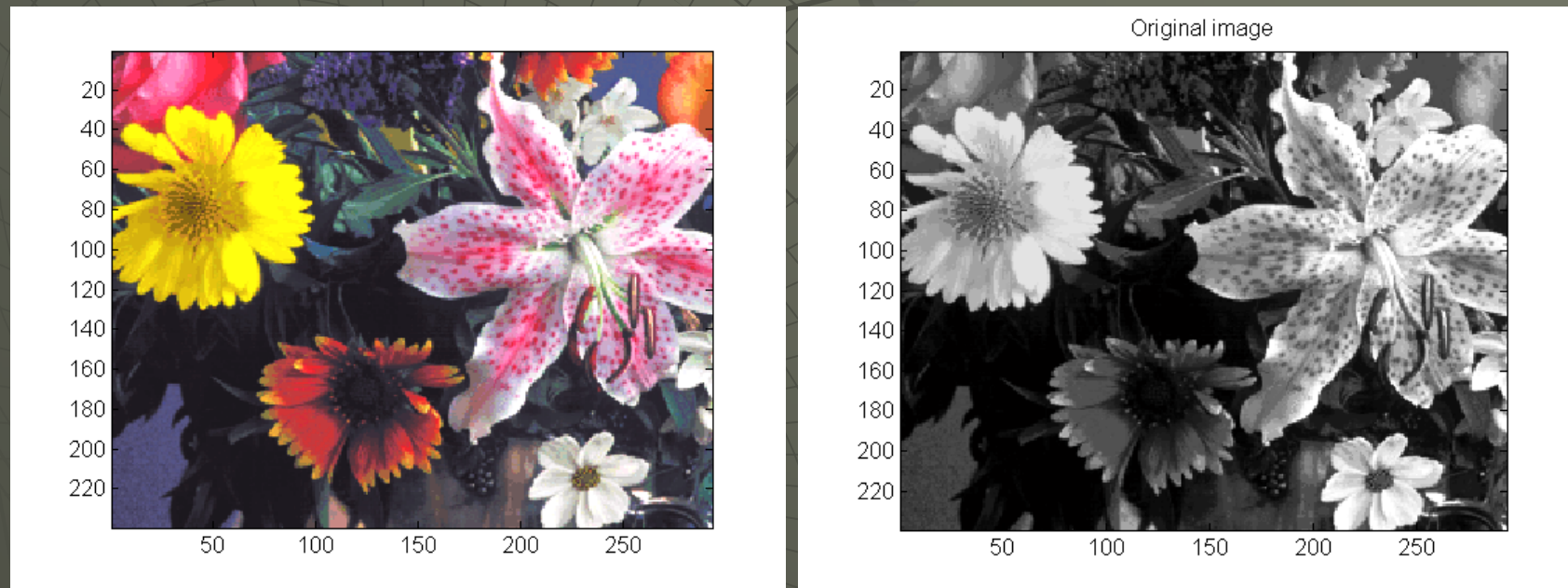
$$p(k) = \begin{cases} \sqrt{1/M} & \text{dla } k = 0 \\ \sqrt{2/M} & \text{dla } k \neq 0 \end{cases}, \quad q(l) = \begin{cases} \sqrt{1/N} & \text{dla } l = 0 \\ \sqrt{2/N} & \text{dla } l \neq 0 \end{cases}$$



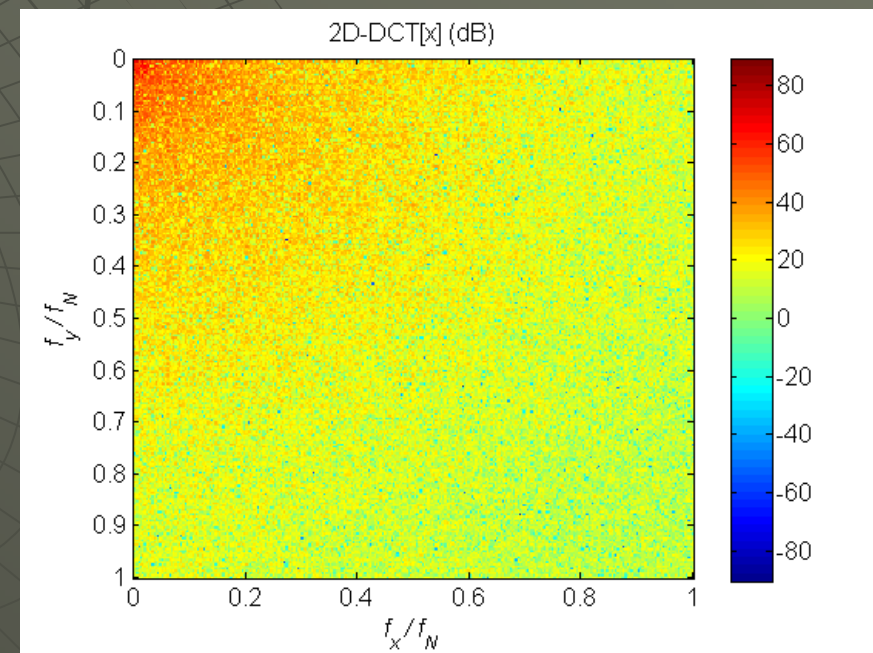
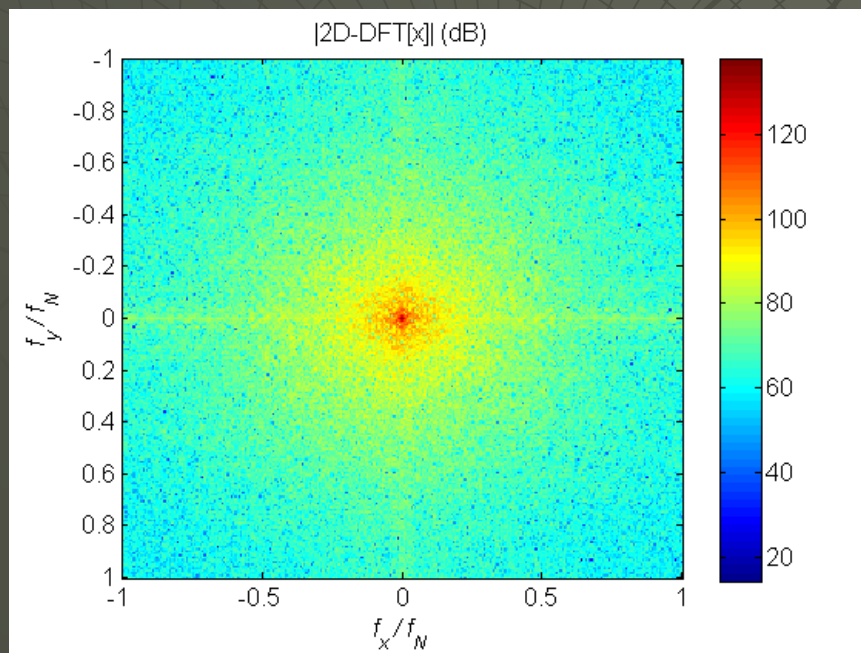
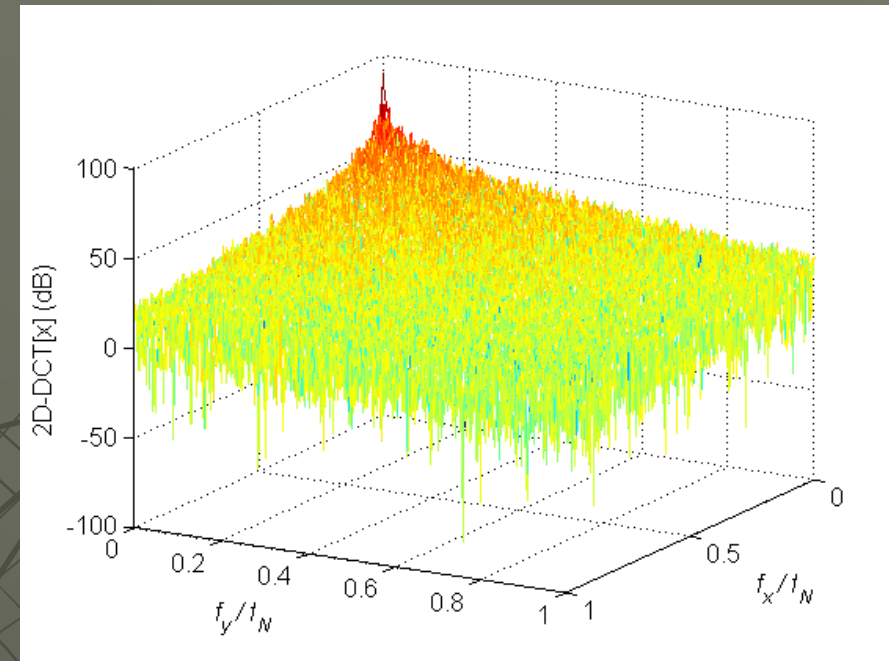
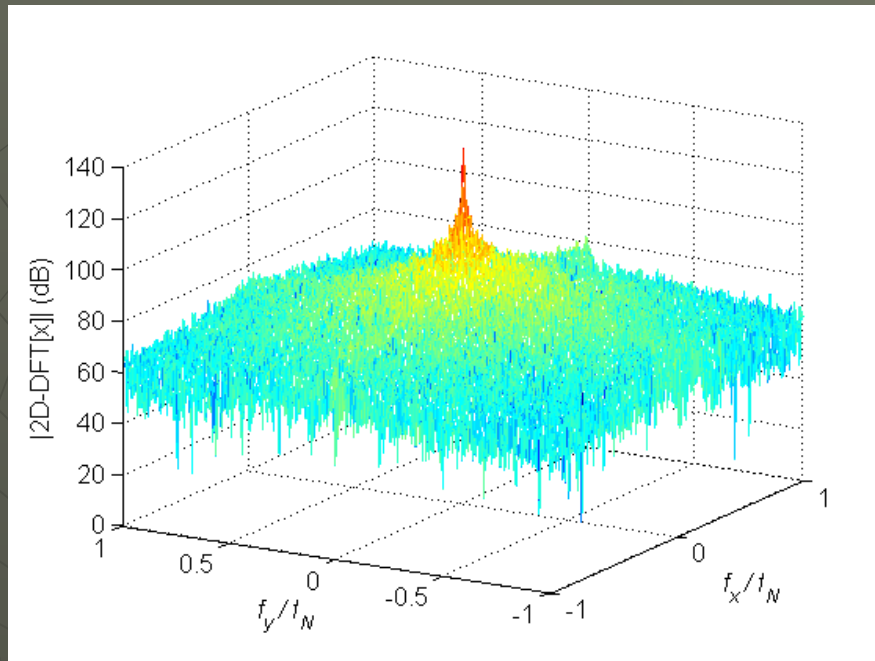
Rys. Okresowe powielenie dwuwymiarowego bloku obrazu przez DFT2 (z nieciągłościami) i DCT2 (zwierciadlana symetria bez nieciągłości na granicach bloków)

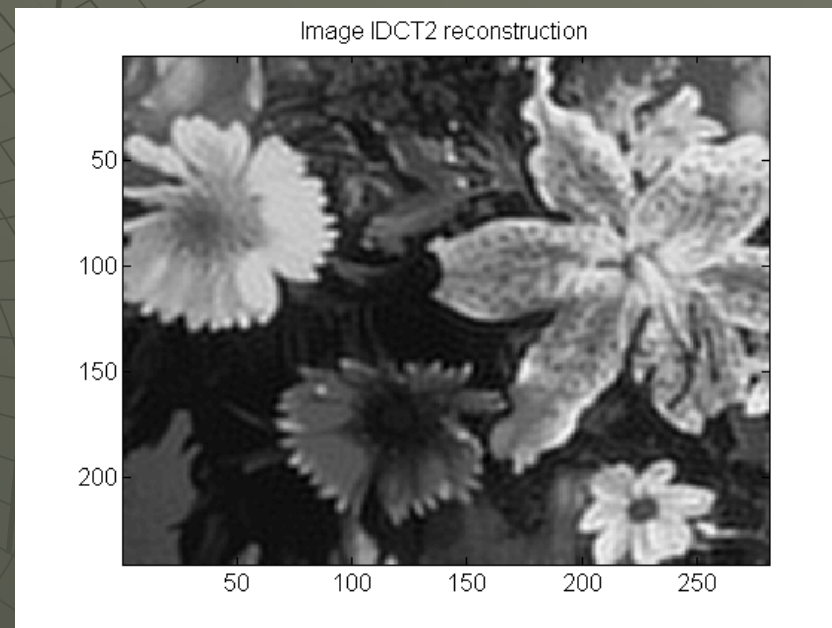
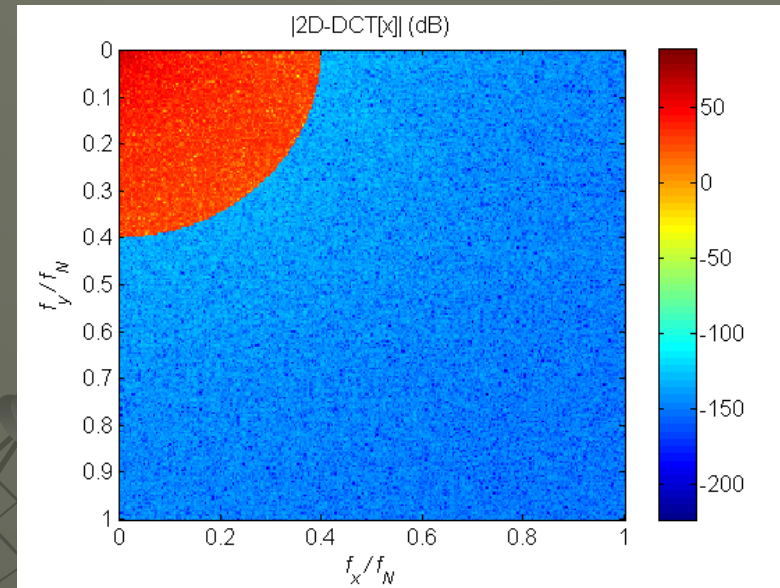
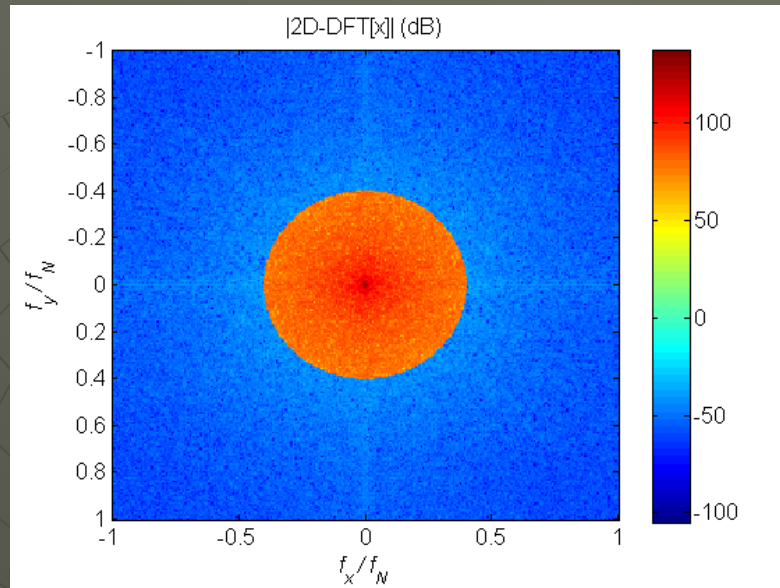
DCT2 jest szeroko stosowana np. przy kompresji obrazów. Ze względu na to, że traktuje blok obrazu (obraz jest zwykle dzielony na bloki 8x8 pikseli) jako wycinek bez nieciągłości na brzegach, DCT2 wprowadza mniej tzw. „artefaktów blokowych”. Zniekształcenia przy krawędziach bloków dla DFT2 są bardziej widoczne, szczególnie po kompresji (np. obcięciu części współczynników).

Przykład: Transformacja DFT2 i DCT2 obrazu i rekonstrukcja po obcięciu części współczynników transformacji poza okręgiem $r=0.4$ częstotliwości Nyquista (idealna filtracja dolnoprzepustowa). W przypadku DFT2 częstotliwość $f=0$ znajduje się w centrum wykresów widma, w przypadku DCT2 - w lewym górnym rogu



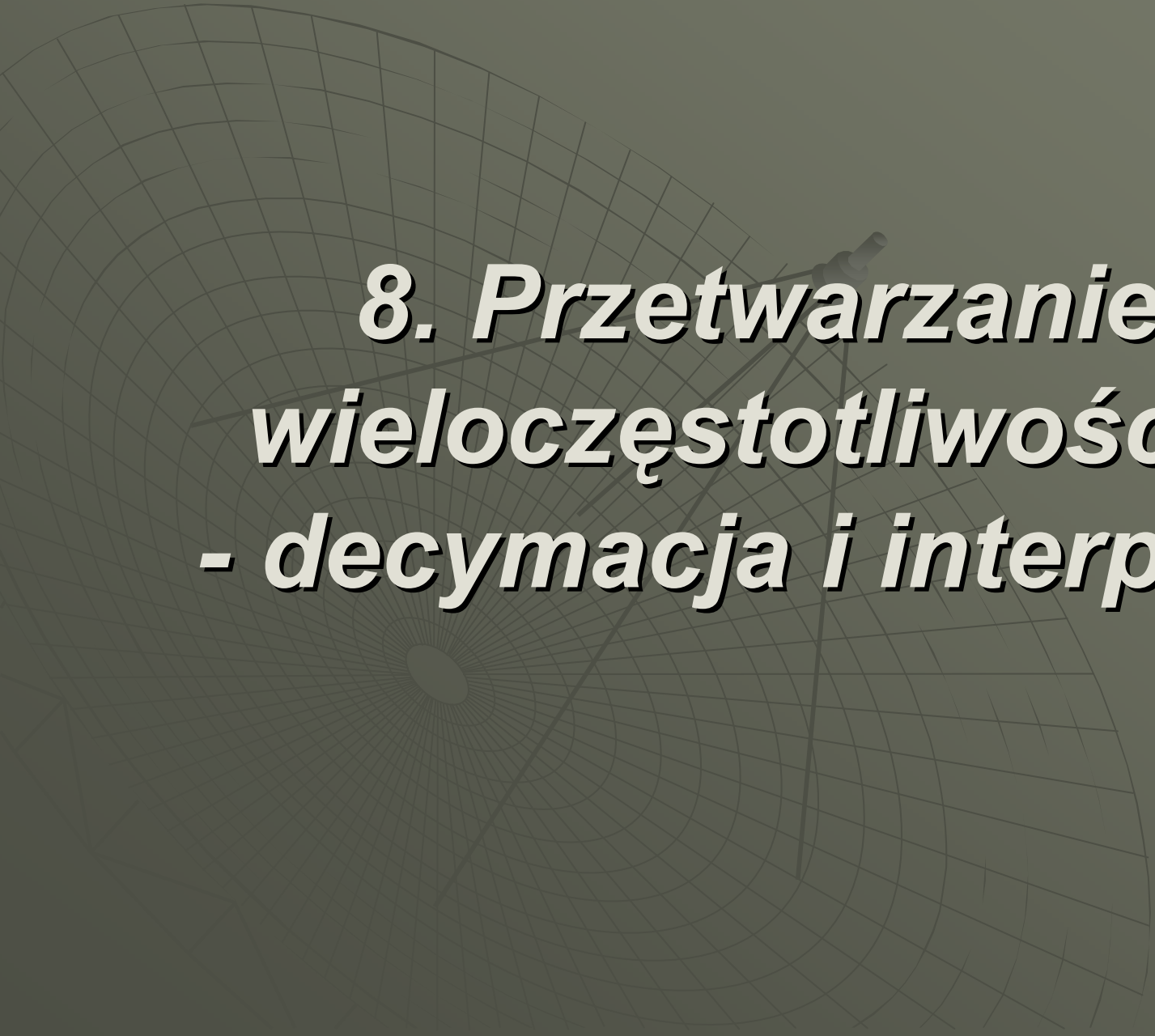
Rys. Obraz oryginalny w kolorze i w odcieniach szarości





obcięcie i odtworzenie DFT2

obcięcie i odtworzenie DCT2



8. Przetwarzanie wieloczęstotliwościowe - decymacja i interpolacja

Typowe cele przetwarzania wieloczęstotliwościowego (*multirate processing*):

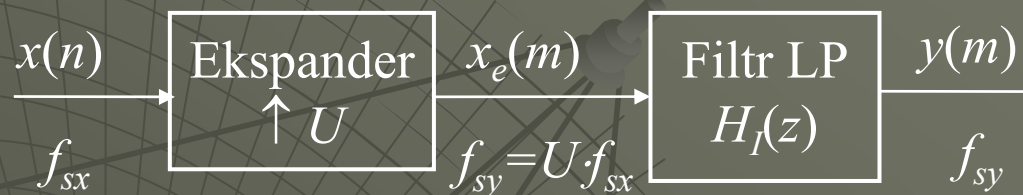
- konwersja sygnału, aby mógł on być przetwarzany z możliwie najmniejszą częstotliwością,
- dopasowanie częstotliwości do określonego standardu, pojemności kanału transmisyjnego lub mocy obliczeniowej procesora DSP.

Przetwarzanie wieloczęstotliwościowe nie ma odpowiednika w dziedzinie analogowej.

Obejmuje też bardziej zaawansowane tematy, takie jak synteza banków filtrów pasmowoprzepustowych do rozkładu i rekonstrukcji sygnałów czy rozkład sygnału za pomocą transformacji falkowych (*wavelets*).

Interpolacja sygnału

Interpolacja o czynnik U polega na rekonstrukcji (estymacji) $U-1$ wartości sygnału $x(n)$, próbkowanego z częstotliwością f_{sx} , pomiędzy istniejącymi próbkami.



Rys. Schemat interpolacji o czynnik U

Ekspander - wstawia $U-1$ próbek zerowych pomiędzy każde sąsiednie próbki sygnału wejściowego (*upsampling*). Sygnał wyjściowy z ekspandera

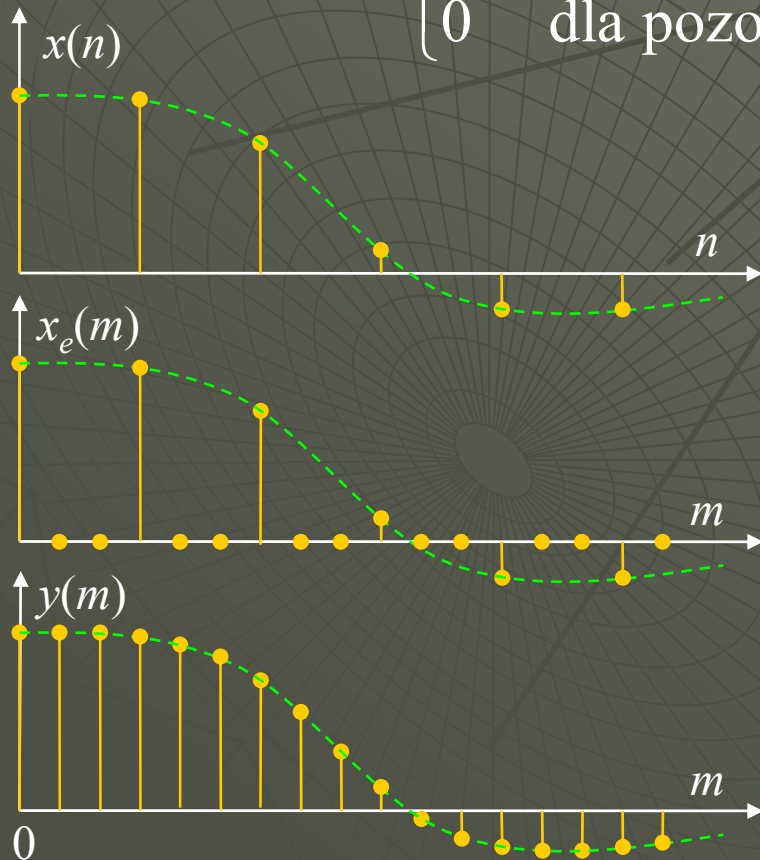
$$x_e(m) = \begin{cases} x(n)|_{n=m/U} = x(m/U) & \text{dla } m = 0, \pm U, \pm 2U, \dots \\ 0 & \text{dla pozostałych } m \end{cases}$$

ma większą częstotliwość próbkowania $f_{sy} = U \cdot f_{sx}$. W dziedzinie częstotliwości zespolonej $X_e(z) = X(z^U)$.

Filtr dolnoprzepustowy (LP) $H_I(z)$ - nadaje dodanym próbkom wartości niezerowe przez interpolację pomiędzy próbkami oryginalnymi generując na wyjściu sygnał interpolowany $y(m)$. Idealny filtr LP interpolatora ma charakterystykę widmową

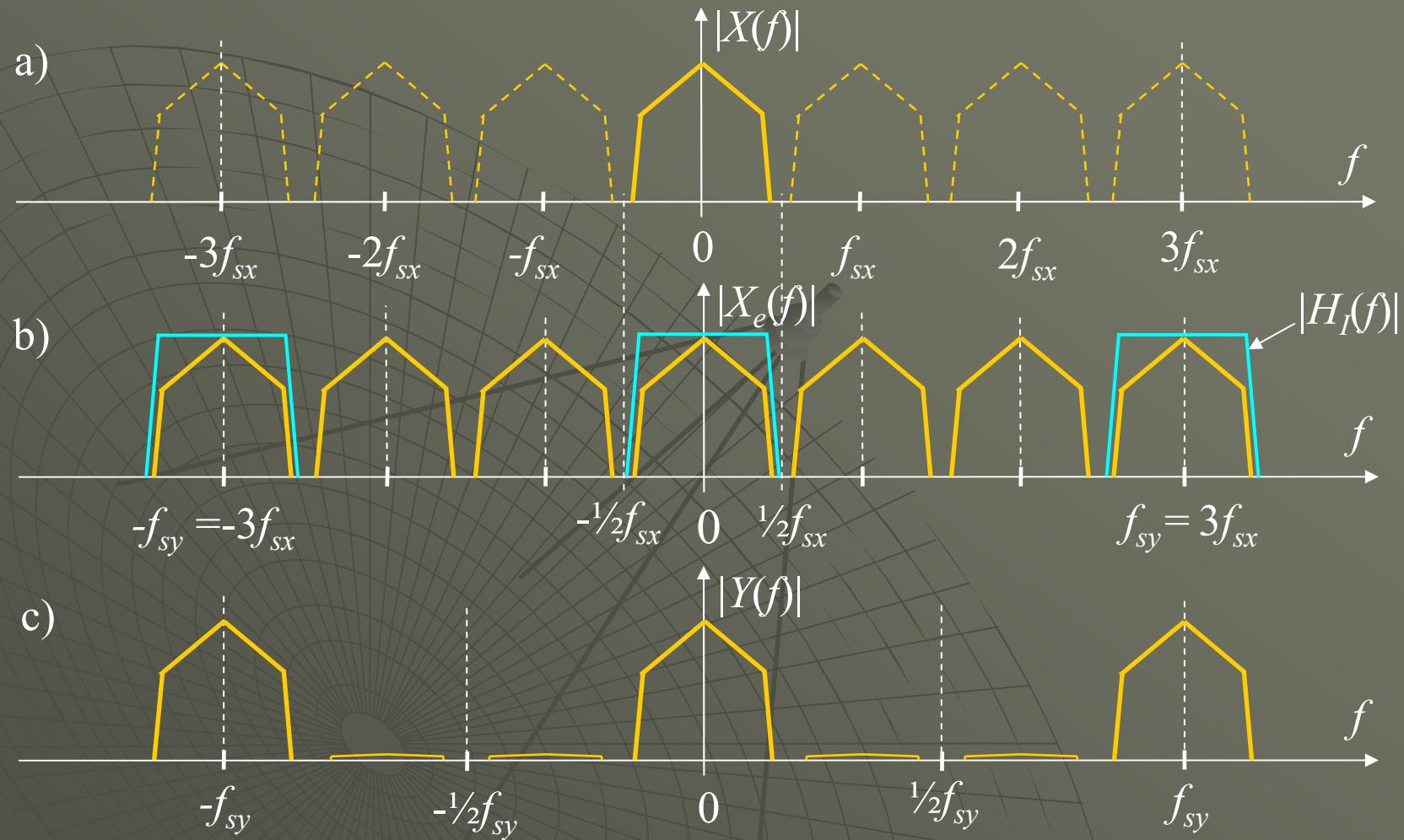
$$H_I(e^{j\Omega}) = \begin{cases} U & \text{dla } 0 \leq |\Omega| \leq \frac{\pi}{U} \\ 0 & \text{dla pozostałych } \Omega \end{cases}$$

$$\Omega_x = \frac{2\pi f_x}{f_{sx}}, \quad \Omega_y = \frac{2\pi f_y}{f_{sy}}$$

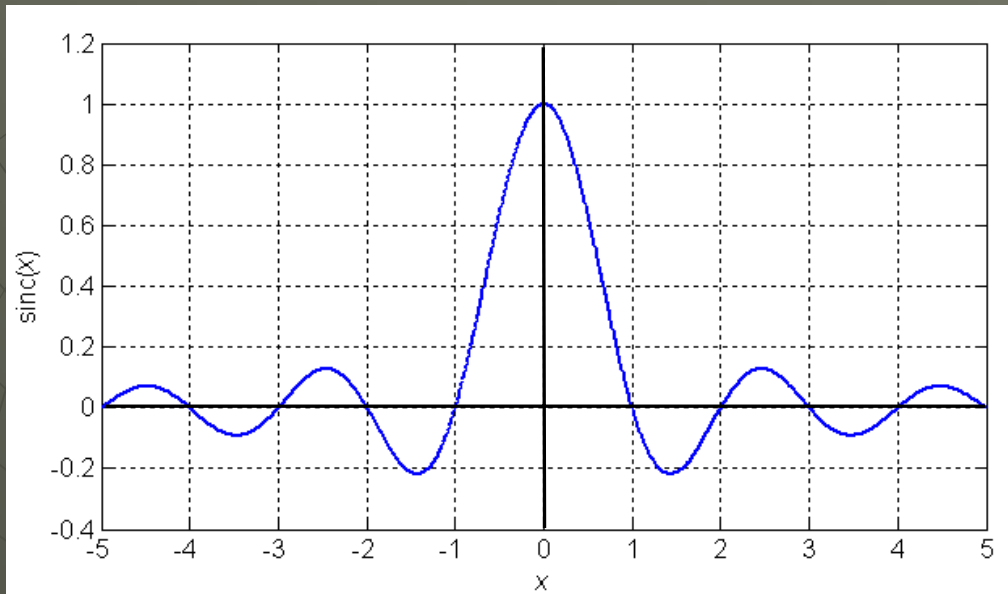


i zerowe opóźnienie grupowe $\tau(\Omega)$. Skala pulsacji $\Omega = \Omega_y = \Omega_x / U$, wzmocnienie U zapewnia takie same amplitudy obwiedni sygnałów $x(n)$ i $y(m)$.

Rys. Przebiegi sygnałów w procesie interpolacji o czynnik $U=3$:
 a) sygnał wejściowy,
 b) sygnał wyjściowy z ekspandera (*upsampling*), c) sygnał wyjściowy po filtracji wygładzającej LP



Rys. Widma sygnałów w procesie interpolacji o czynnik $U=3$: a) sygnał wejściowy, b) sygnał z ekspandera i charakterystyka $|H_I(f)|$ filtra LP, c) sygnał wyjściowy po filtracji

Rys. Wykres funkcji $\text{sinc}(x)$

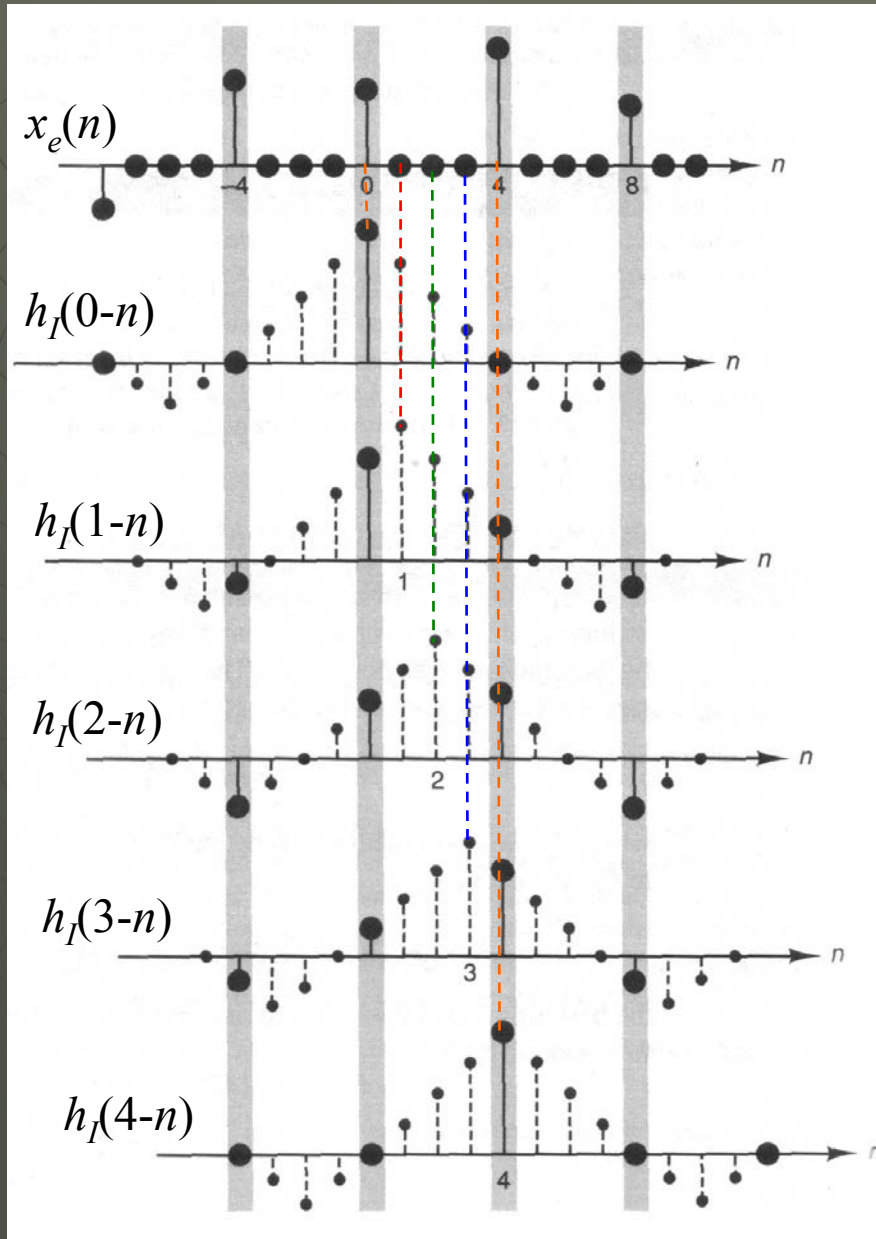
$H_I(e^{j\Omega})$ odpowiada nieskończonej i *nieprzyczynowej* charakterystyce impulsowej:

$$h_I(m) = \begin{cases} \frac{\sin(m\pi/U)}{m\pi/U} & \text{dla } m \neq 0 \\ 1 & \text{dla } m = 0 \end{cases}$$

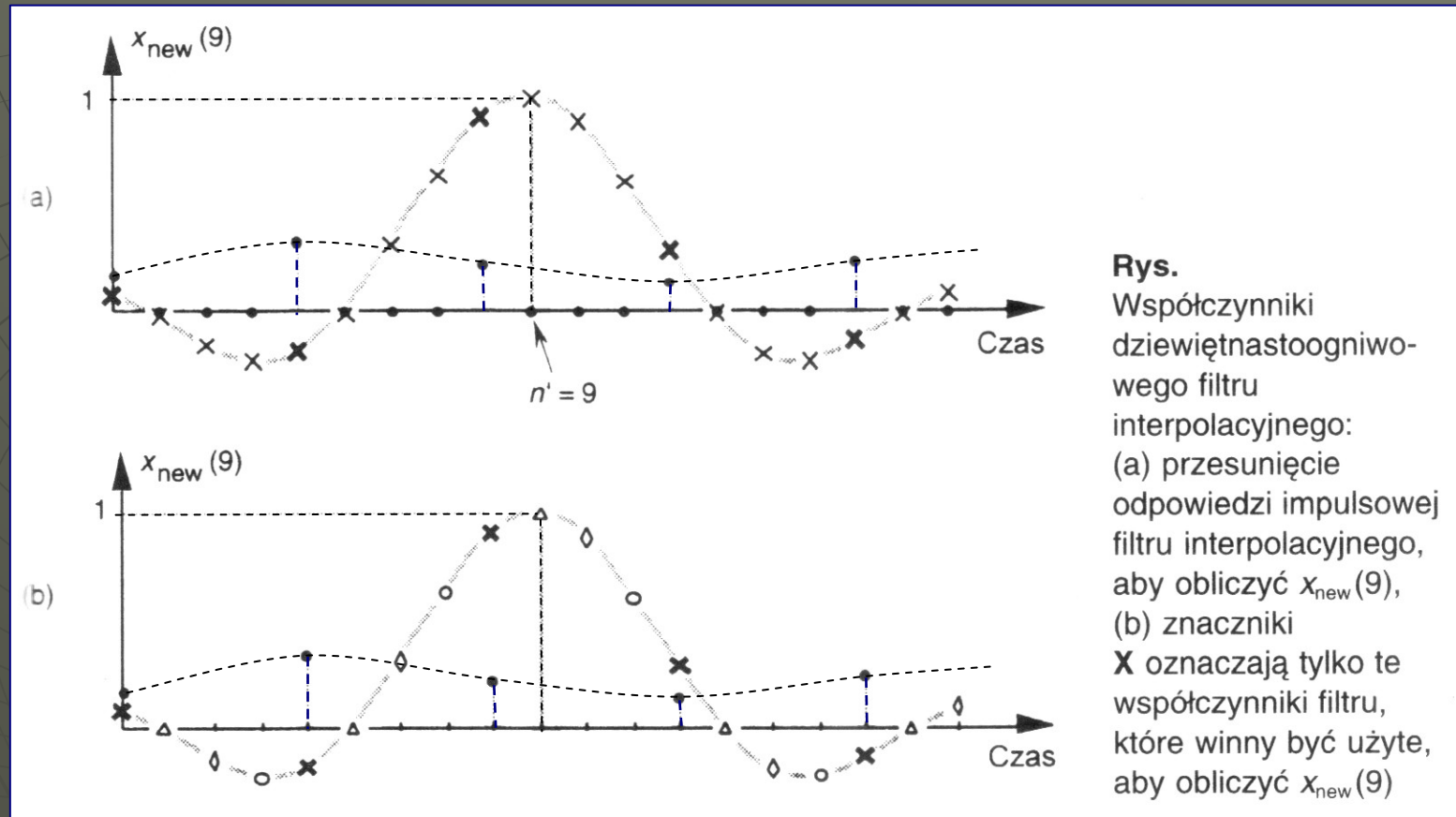
gdzie $\sin(\pi x)/(\pi x) = \text{sinc}(x)$.

W dziedzinie czasu wyjście interpolatora jest splotem:

$$y(m) = h_I(m) * x_e(m) = \sum_{k=-\infty}^{\infty} h_I(m-k)x_e(k) = \sum_{k=-\infty}^{\infty} h_I(m-k)x(k/U)$$

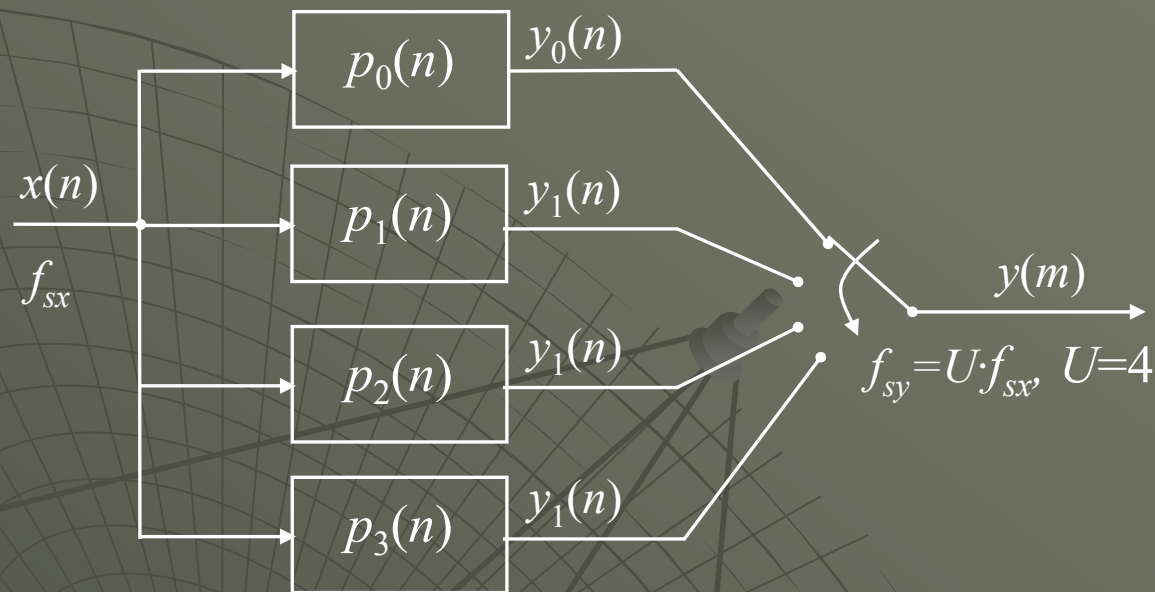


Rys. Interpolacja jako proces filtracji liniowej – splotu $x_e(n)$ z charakterystyką impulsową filtra $h_I(n)$. Odwrócona charakterystyka $h_I(-n)$ jest przesuwana na tle sygnału $x_e(n)$. Dla każdego przesunięcia $y(n)$ to suma iloczynów niezerowych próbek i współczynników filtra. Zwrócić uwagę, kiedy zera $x_e(n)$ nakładają się na $h_I(k-n)$ i kiedy zera $h_I(k-n)$ nakładają się na $x_e(n)$.



Rys. Dekompozycja filtra interpolacyjnego na U filtrów, których wyjścia są obliczane z częstotliwością f_s/U

- współczynniki Δ gdy $n' = 0, 4, 8, \dots, 4n$ $p_0(n) = h_I(4n)$
- współczynniki x gdy $n' = 1, 5, 9, \dots, 4n+1$ $p_1(n) = h_I(4n+1)$
- współczynniki o gdy $n' = 2, 6, 10, \dots, 4n+2$ $p_2(n) = h_I(4n+2)$
- współczynniki \diamond gdy $n' = 3, 7, 11, \dots, 4n+3$ $p_3(n) = h_I(4n+3)$



Rys. Przedstawienie interpolacji jako wybieranych cyklicznie sygnałów wyjściowych z U prostszych filtrów polifazowych, działających z niższą częstotliwością f_{sx} , otrzymanych z filtra $h_I(m)$ interpolatora.

Proste interpolacje

- ekstrapolacja zerowego rzędu ZOH (*zero order hold*, powtarzanie próbek)

$$y(m) = \begin{cases} x_e(0) & m = 0, 1, \dots, U-1 \\ x_e(U) & m = U, U+1, \dots, 2U-1 \\ x_e(2U) & m = 2U, 2U+1, \dots, 3U-1 \\ \vdots & \end{cases}$$

Przyczynowa ch-ka impulsowa:

$$h_{\text{ZOH}}(m) = \delta(m) + \delta(m-1) + \dots + \delta(m-(U-1))$$

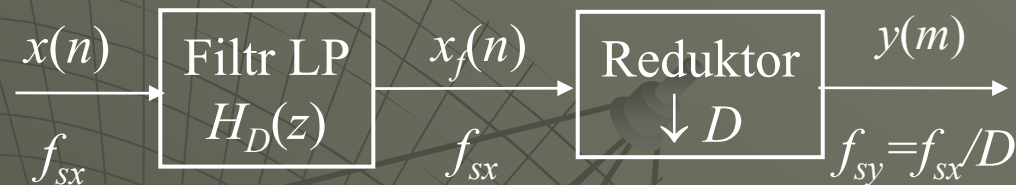
-interpolacja liniowa pomiędzy dwiema próbkami sygnału oryginalnego.

Nieprzyczynowa ch-ka impulsowa:

$$h_{\text{lin}}(m) = \begin{cases} 1 - |m|/U & \text{dla } |m| \leq U-1 \\ 0 & \text{dla pozostałych } m \end{cases}$$

Decymacja sygnału

Decymacja o czynnik D polega na D -krotnym zmniejszeniu częstotliwości próbkowania sygnału oryginalnego. Jest operacją odwrotną do interpolacji



Rys. Schemat decymacji o czynnik D

Filtr dolnoprzepustowy $H_D(z)$ - ogranicza pasmo sygnału wejściowego $x(n)$ do zakresu D razy mniejszego. Charakterystyka widmowa idealnego filtra LP decymatora (skala $\Omega = \Omega_x = \Omega_y/D$):

$$H_D(e^{j\Omega}) = \begin{cases} 1 & \text{dla } 0 \leq |\Omega| \leq \frac{\pi}{D} \\ 0 & \text{dla pozostałych } \Omega \end{cases}$$

Filtracja LP zapobiega aliasingowi po następującej dalej operacji redukcji częstotliwości próbkowania - pozostawieniu co D -tej próbki sygnału przefiltrowanego $x_f(n)$: $f_{sy} = f_{sx}/D$. Zmniejszenie szybkości próbkowania ogranicza na wyjściu reduktora D razy pasmo Nyquista.

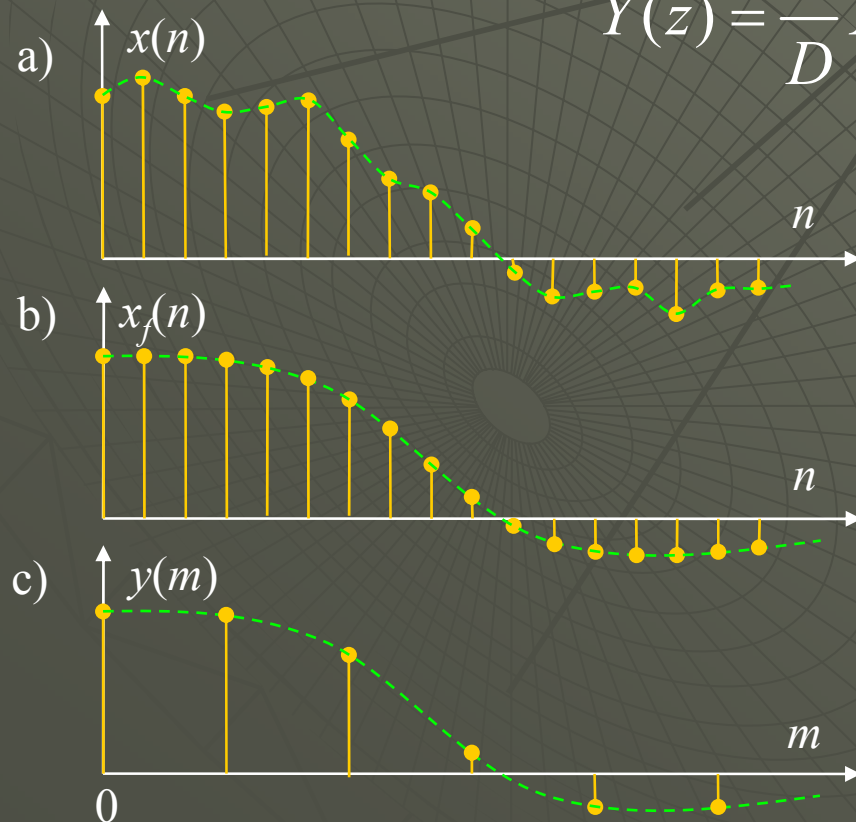
Sygnał wyjściowy decymatora:

$$y(m) = x_f(n) \big|_{n=mD} = h_D(n) * x(n) \big|_{n=mD} = \sum_{k=-\infty}^{\infty} h_D(k) \cdot x(mD - k)$$

gdzie $h_D(n)$ jest charakterystyką impulsową filtra $H_D(z)$.

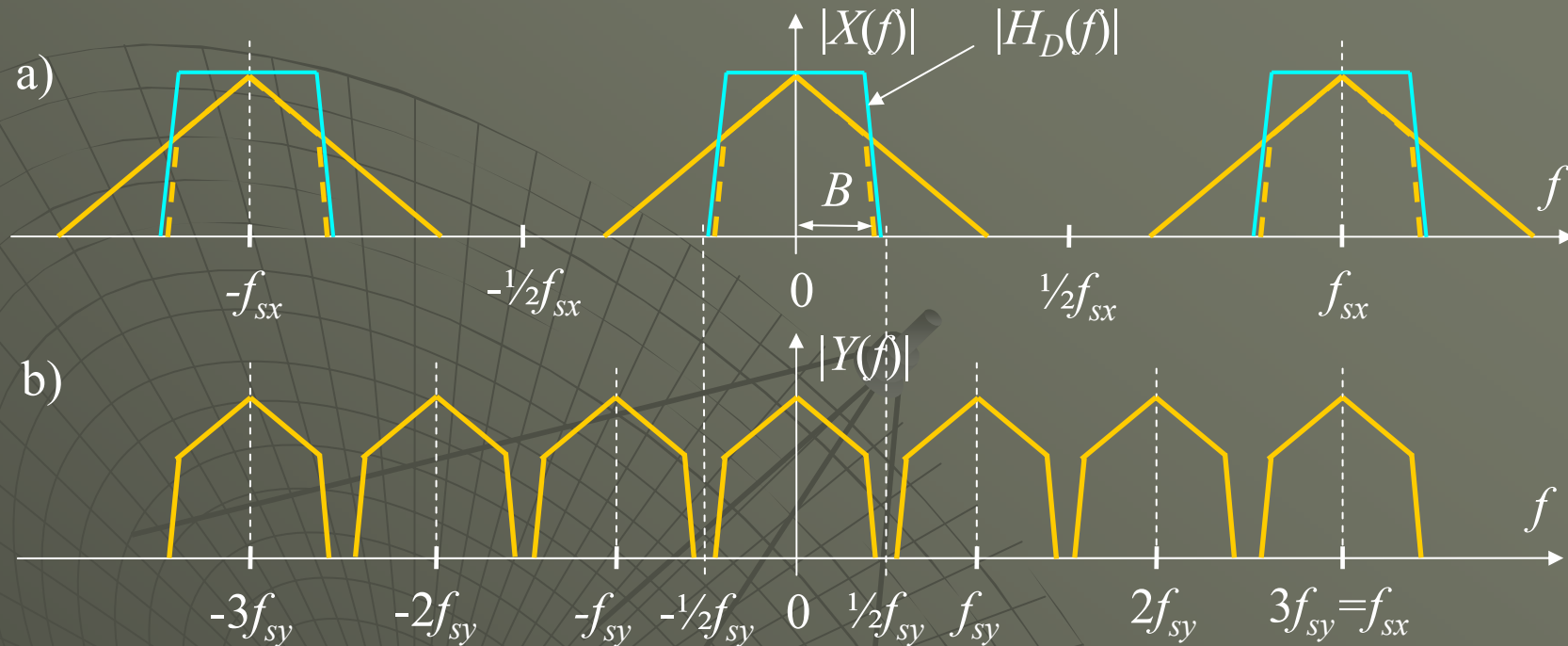
W przypadku idealnej filtracji LP w dziedzinie transformaty Z :

$$Y(z) = \frac{1}{D} X(z^{1/D})$$

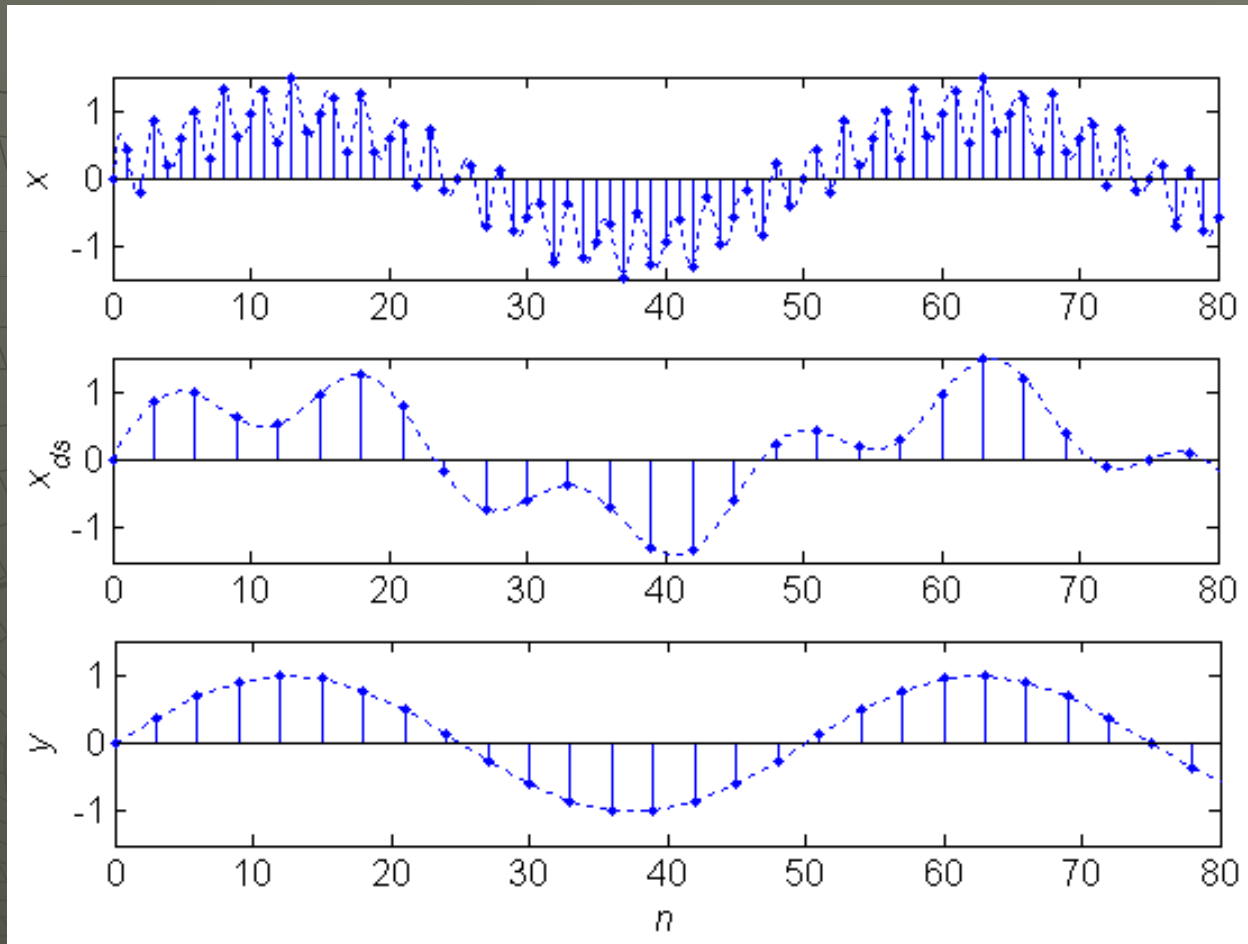


Decymacja może być prowadzona jako operacja *przyczynowa*.

Rys. Przebiegi sygnałów w procesie decymacji o czynnik $D=3$:
 a) sygnał wejściowy, b) sygnał po filtracji LP, c) sygnał wyjściowy po redukcji częstotliwości próbkowania (*downsampling*)



Rys. Widma sygnałów w procesie decymacji o czynnik $D=3$:
 a) sygnał wejściowy i charakterystyka $|H_D(f)|$ filtra LP ograniczającego pasmo, b) sygnał wyjściowy po filtracji i redukcji częstotliwości próbkowania

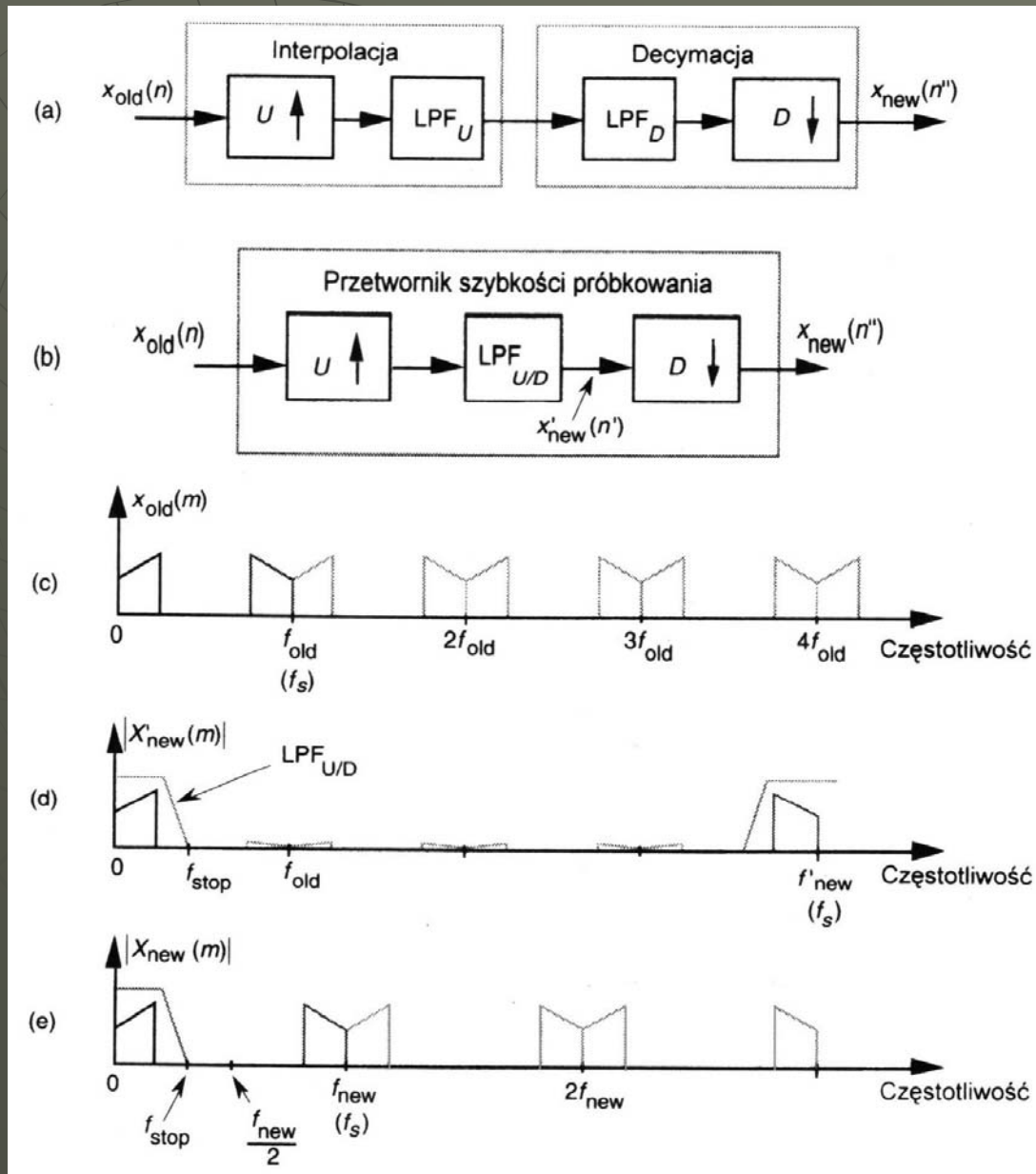


Rys. Decymacja $D=3$ razy bez filtracji LP i z filtracją: a) sygnał wejściowy, b) sygnał decymowany (*downsampled*, aliasing składowej o $f/f_s=0.4$), c) sygnał filtrowany i decymowany

$$x(n) = A_1 \sin \Omega_1 n + A_2 \sin \Omega_2 n$$

$$A_1 = 1.0, \Omega_1 = 2\pi \cdot 0.02, \quad A_2 = 0.4, \Omega_2 = 2\pi \cdot 0.4$$

Połączenie interpolacji i decymacji



Rys. Przetworzenie częstotliwości próbkowania o czynnik ułamkowy $U/D=4/3$:

a) kombinacja interpolacji i decymacji,

b) zastąpienie dwóch filtrów LP jednym o transmitancji $H_I(z)H_D(z)$ o węższym z dwóch pasm przepuszczenia,

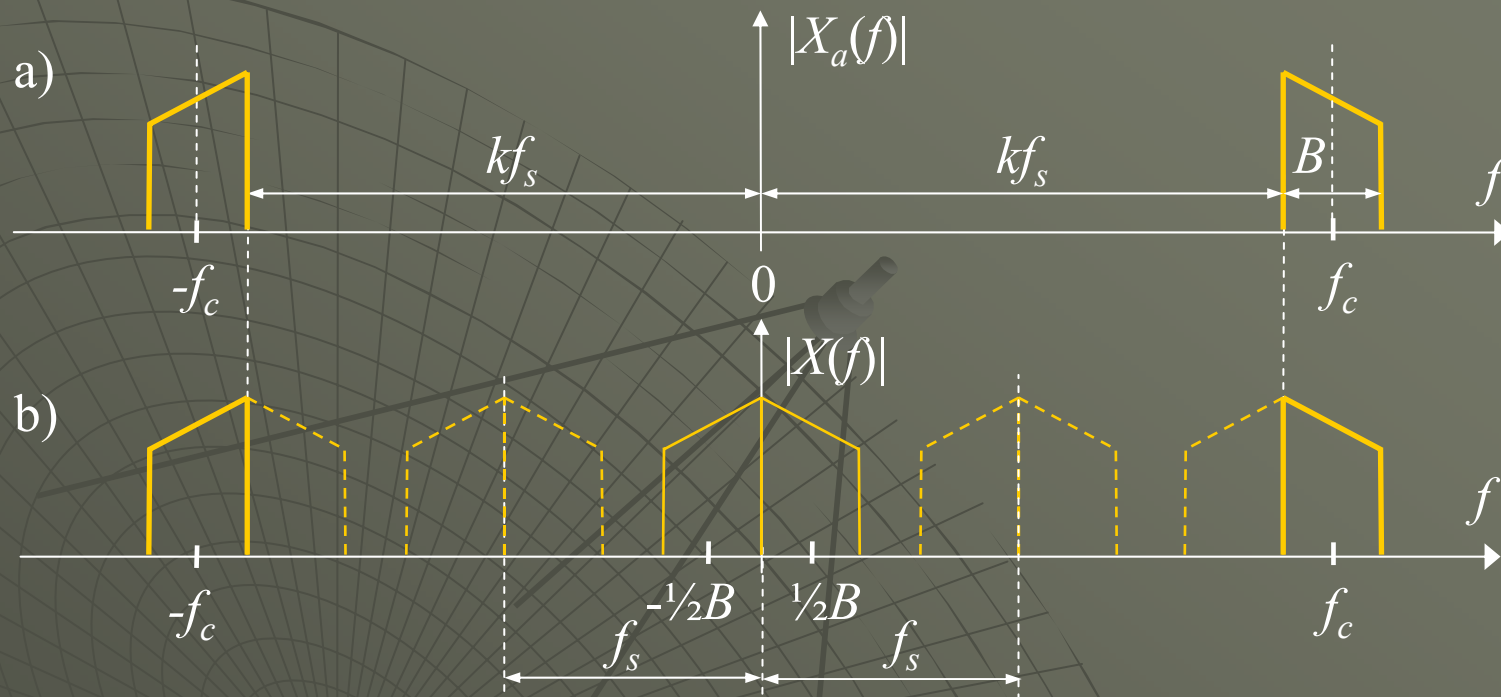
c) widmo sygnału wejściowego,

d) widmo sygnału wyjściowego filtra po interpolacji,

e) ostateczne widmo sygnału wyjściowego po decymacji

$$f_{sx} = f_{s_old} \quad f_{sy} = f_{s_new}$$

Próbkowanie sygnału pasmowego



Rys. 7. Próbkowanie sygnału pasmowego: a) widmo oryginalnego sygnału ciągłego, b) powielenia widma sygnału próbkowanego, B – szerokość pasma sygnału, f_c – częstotliwość nośna (centralna), f_s – częstotliwość próbkowania pasmowego

Próbkowanie dolnopasmowe sygnału o pasmowym widmie wymaga dużej częstotliwości próbkowania: $\frac{1}{2}f_{SLP} \geq f_c + B/2$.

Zastosowanie redukcji szybkości próbkowania (*downsampling*, decymacja bez filtracji) umożliwia uzyskanie powielenia oryginalnego widma wokół częstotliwości 0 wskutek *aliasingu*, bez straty informacji. *Jest to oszczędna metoda reprezentacji sygnałów pasmowych.*

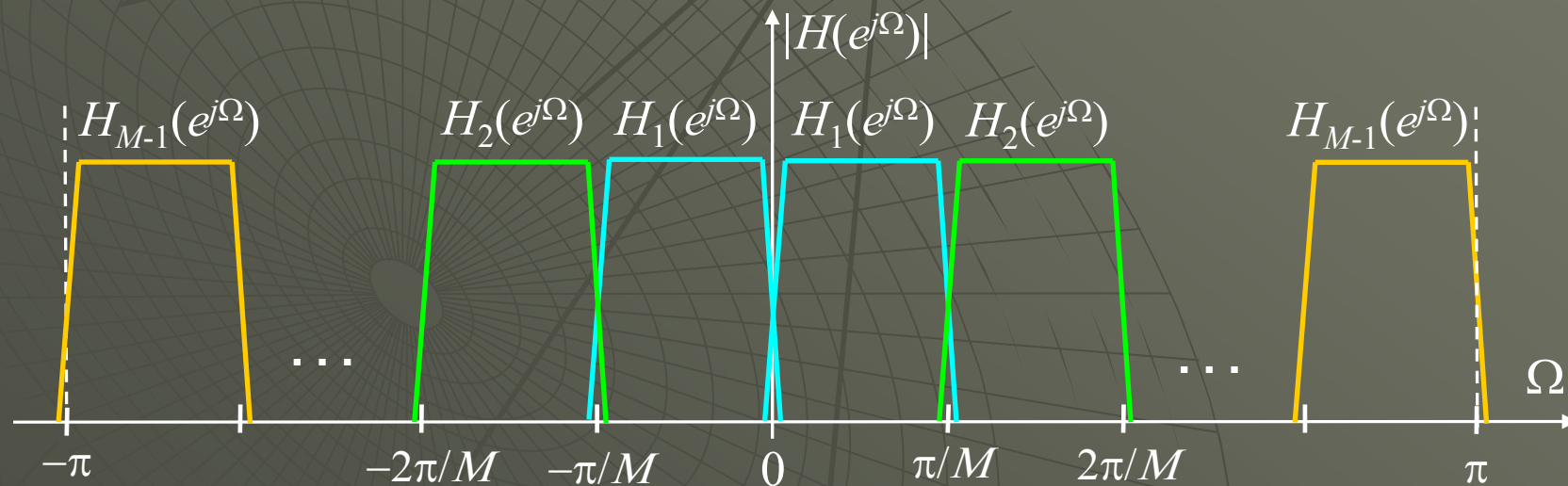
Warunki uniknięcia aliasingu ($k \in \mathbb{N}$):

$$f_s \geq 2B \quad \text{i} \quad \frac{f_c + B/2}{k + 1/2} \leq f_s \leq \frac{f_c - B/2}{k}$$

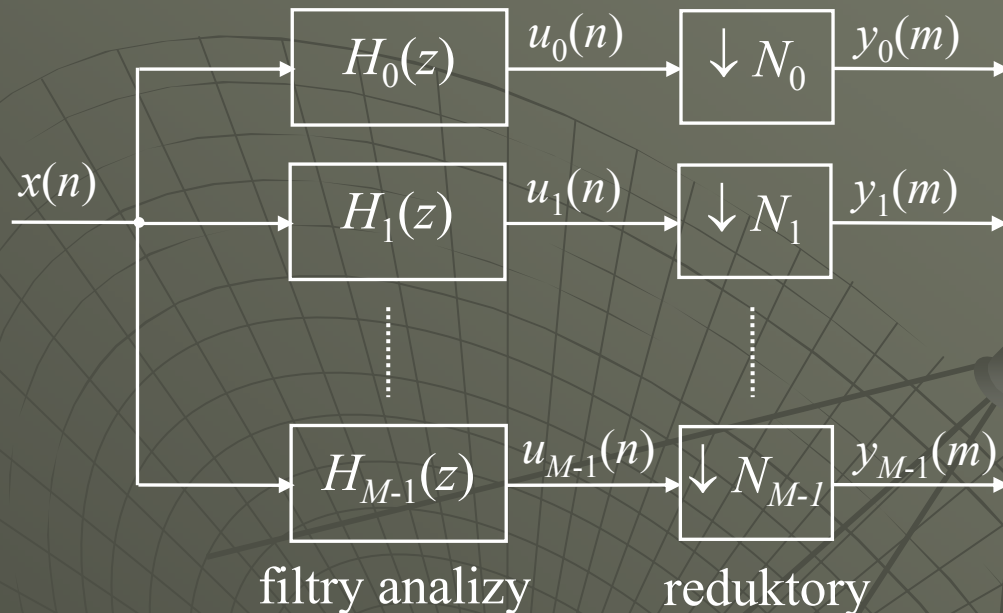
Odtworzenie sygnału oryginalnego jest możliwe po interpolacji pasmoprzepustowej: powrocie do większej częstotliwości próbkowania i filtracji BP wydzielającej tylko oryginalne pasmo sygnału.

Podpasmowa dekompozycja sygnału za pomocą banku filtrów

Przetwarzanie sygnału w podpasmach częstotliwości jest stosowane w wielu praktycznych aplikacjach, m.in. algorytmach kompresji audio (MPEG-1,2 z poziomem mp3 oraz MPEG-4) i obrazów (JPEG2000). Jedną z metod dekompozycji sygnału na składowe pasmowe jest zastosowanie banku (zespołu) filtrów BP pokrywających zakres widma sygnału, a następnie decymacja składowych sygnału w podpasmach, co umożliwia np. kompresję sygnału (w zależności od wariacji składowych sygnału w podpasmach reprezentuje się je w formie zawierającej mniej lub więcej bitów).



Rys. Charakterystyki amplitudowo-częstotliwościowe M -pasmowego banku filtrów otrzymane w wyniku rzeczywistej modulacji kosinusowej charakterystyki impulsowej $p(n)$ prototypu LP



Rys. M -kanałowy bank (zespół) filtrów podpasmowej analizy sygnału (*subband decomposition*)

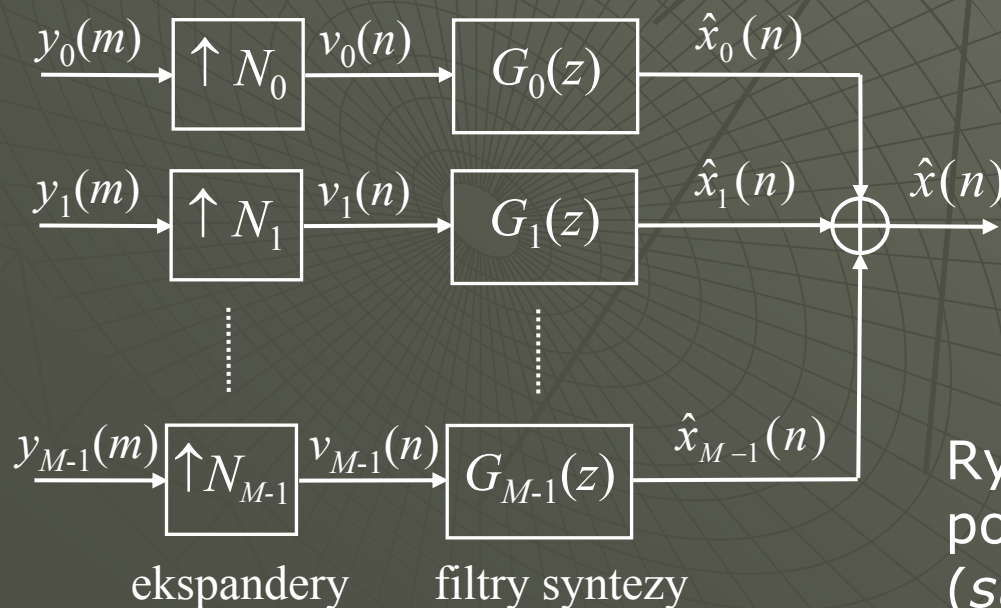
Odpowiednie umiejscowienie charakterystyk $H_k(e^{j\Omega})$ filtrów na osi częstotliwości można uzyskać w efekcie modulacji kosinusowej (w dziedzinie czasu) charakterystyki impulsowej $p(n)$, $0 \leq n \leq L-1$, filtra prototypowego LP o pulsacji granicznej $\Omega_p = \pi/(2M)$. Charakterystyki impulsowe poszczególnych filtrów BP banku:

$$h_k(n) = 2p(n) \cdot \cos \left[\frac{\pi}{M} \left(k + \frac{1}{2} \right) \left(n - \frac{L-1}{2} \right) + (-1)^k \frac{\pi}{4} \right], \quad k = 0, 1, \dots, M-1$$

Modulacja kosinusowa charakterystyki impulsowej powoduje przesunięcie dwóch powieleń charakterystyki widmowej na osi pulsacji Ω :

$$h(n) \cdot \cos \Omega_0 n = h(n) \frac{e^{-j\Omega_0 n} + e^{j\Omega_0 n}}{2} \leftrightarrow \frac{1}{2} \left[H(e^{j(\Omega - \Omega_0)}) + H(e^{j(\Omega + \Omega_0)}) \right]$$

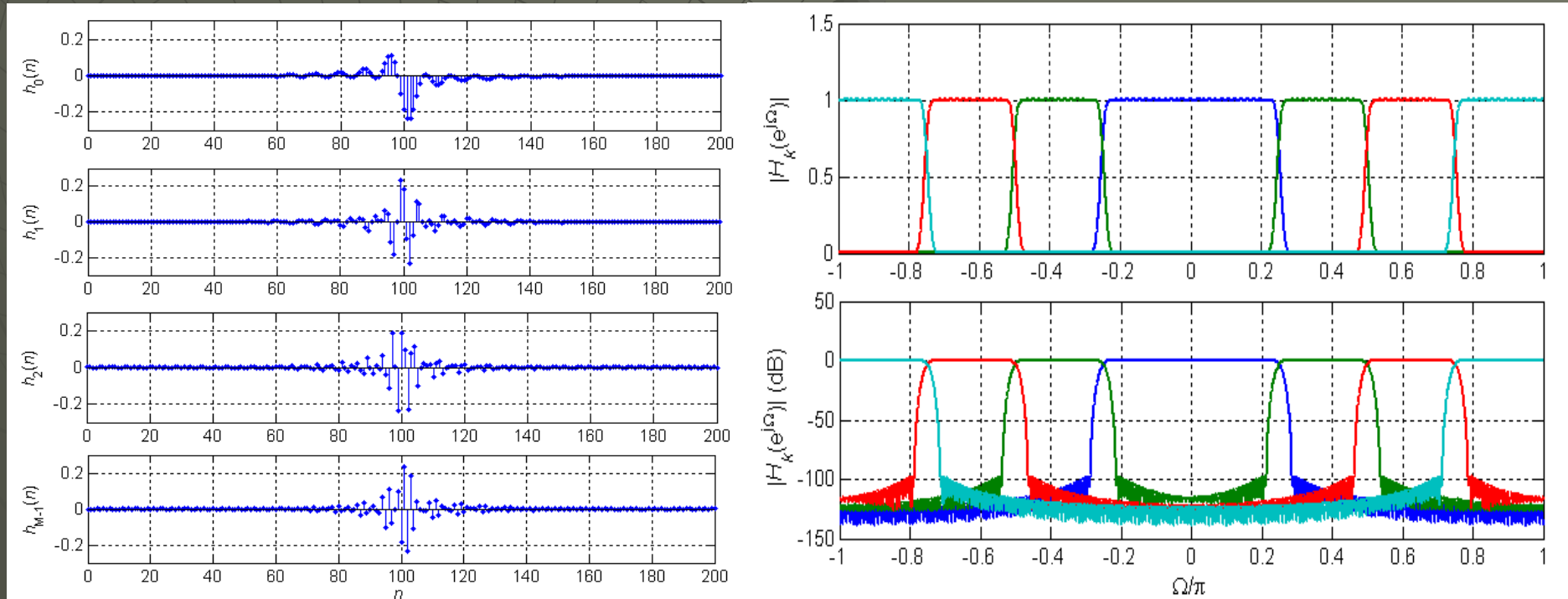
Odtworzenie $\hat{x}(n)$ sygnału oryginalnego uzyskuje się po interpolacji, czyli przepuszczeniu zdecydowanych składowych $y_k(m)$ przez ekspandery i filtry BP rekonstrukcji



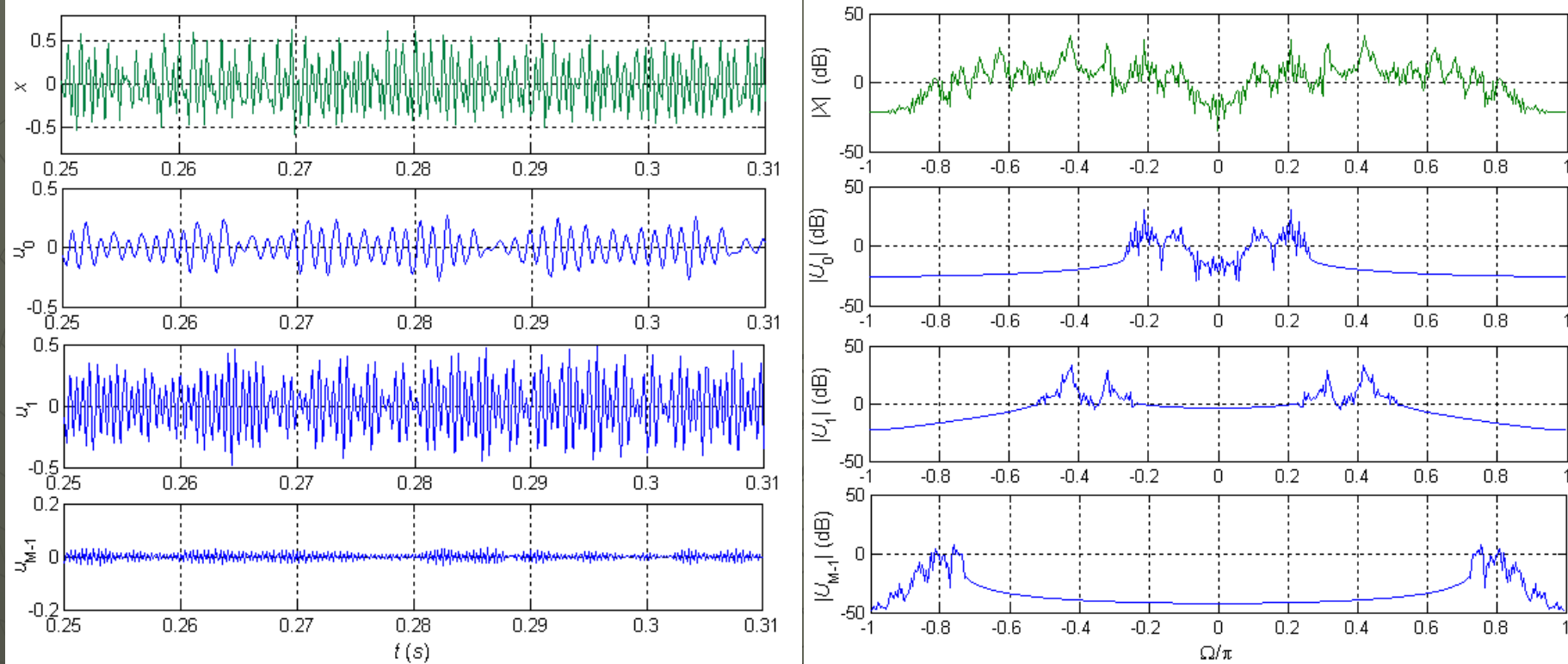
Rys. M -kanałowy bank filtrów podpasmowej syntezy sygnału (*subband reconstruction*)

Przykład

Projekt i wyniki działania banku filtrów dekompozycji i rekonstrukcji sygnału z modulacją kosinusową charakterystyki prototypu; liczba pasm $M=4$, długość odpowiedzi impulsowych filtrów SOI $L=201$, tłumienie w paśmie zaporowym $r_s=96\text{dB}$, liczba bitów do alokacji na próbkę $R=8$, częstotliwość próbkowania $f_s=8\text{kHz}$

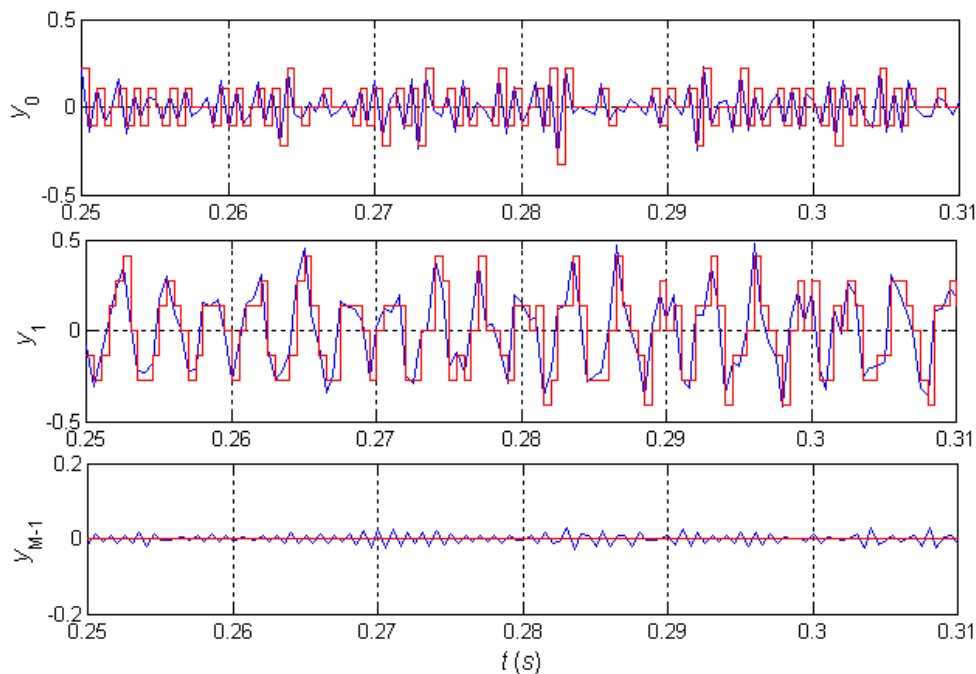


Rys. Charakterystyki impulsowe $h_k(n)$ i częstotliwościowe amplitudowe $|H_k(e^{j\Omega})|$, $k=0,1,\dots,M-1$ filtrów analizy. Ch-ki impulsowe filtrów syntezy $g_k(n)$ są odwrócone w czasie



Rys. Działanie filtrów analizy: dekompozycja sygnału wejściowego $x(n)$ na składowe podpasmowe $u_k(n)$ i ich widma amplitudowe

-  $x(n)$, fire8.wav
 $f_s=8\text{kHz}$, 16bit
-  $u_0(n)$
-  $u_1(n)$
-  $u_2(n)$
-  $u_3(n)$

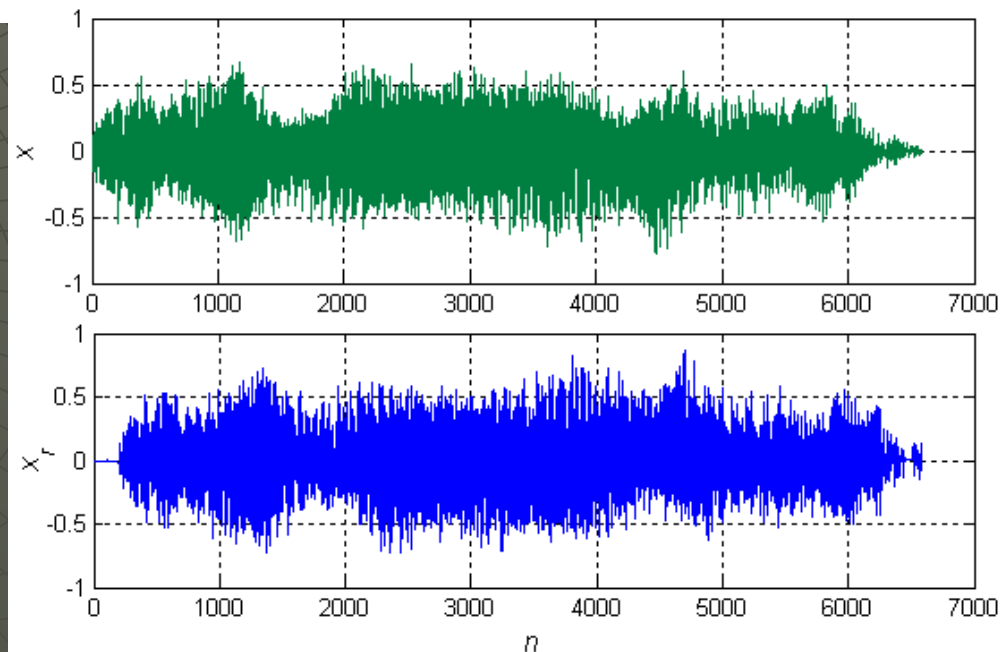


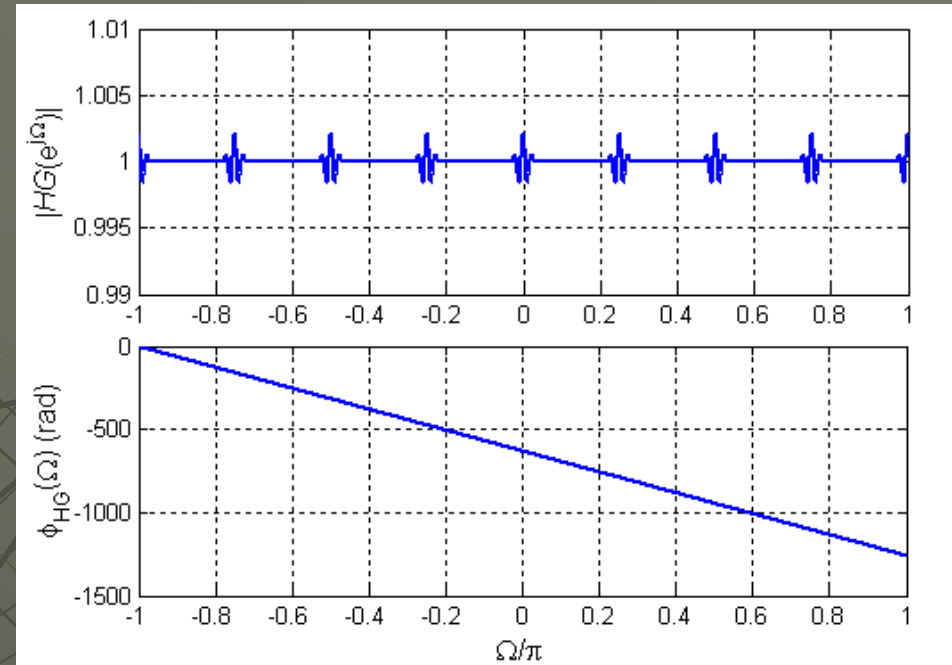
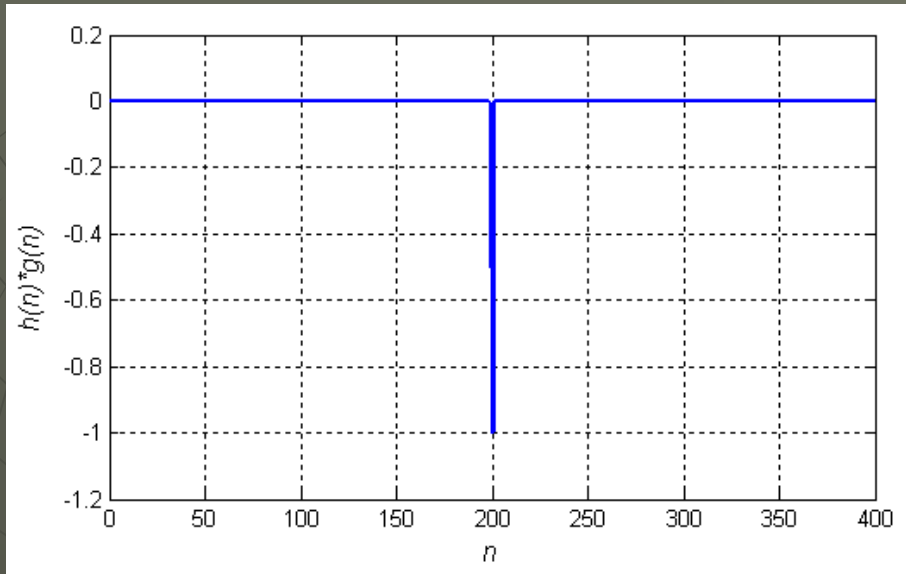
Rys. *Downsampling* o czynnik $M=4$ (do $f_{sy}=2\text{kHz}$) i kwantowanie sygnałów podpasmowych. Alokacja $R=8$ bitów na podpasma dla rozpatrywanego kodowanego sygnału: $R_0=3, R_1=3, R_2=2, R_3=0$

Rys. Sygnał oryginalny $x(n)$ i zrekonstruowany ze skwantowanych podpasm przez filtry syntezy $x_r(n)$. Zwrócić uwagę na przesunięcie fazowe



$x_r(n)$





Rys. *Sumaryczna* charakterystyka impulsowa (idealna powinna być funkcją impulsową $\delta(n)$) i częstotliwościowa filtrów analizy $h(n)$ i filtrów syntezy $g(n)$ banku

Opis matematyczny banku filtrów

Zależności pomiędzy transformatami Z sygnałów w banku:

- sygnały wyjściowe z filtrów analizy:

$$U_k(z) = H_k(z) \cdot X(z), \quad k = 0, 1, \dots, M-1$$

- sygnały wyjściowe z reduktorów:

$$Y_k(z) = \frac{1}{N} \sum_{l=0}^{N-1} U_k(w_N^l z^{1/N}), \quad w_N = e^{-j2\pi/N}$$

$Y_k(z)$ jest sumą powielonych poprzesuwanych (zmodulowanych przez w_N^l) widm $U_k(z^{1/N})$.

- sygnały wyjściowe z ekspanderów:

$$V_k(z) = Y_k(z^N)$$

- sygnały wyjściowe z filtrów syntezy (odtworzone składowe podpasmowe):

$$\hat{X}_k(z) = G_k(z) \cdot V_k(z)$$

- odtworzony sygnały wyjściowy:

$$\begin{aligned}\hat{X}(z) &= \sum_{k=0}^{M-1} \hat{X}_k(z) = \sum_{k=0}^{M-1} G_k(z) Y_k(z^N) = \sum_{k=0}^{M-1} \left(G_k(z) \frac{1}{N} \sum_{l=0}^{N-1} H_k(w_N^l z) X(w_N^l z) \right) = \\ &= \sum_{l=0}^{N-1} \left(X(w_N^l z) \frac{1}{N} \sum_{k=0}^{M-1} H_k(w_N^l z) G_k(z) \right) = \sum_{l=0}^{N-1} X(w_N^l z) A_l(z)\end{aligned}$$

gdzie:
$$A_l(z) = \frac{1}{N} \sum_{k=0}^{M-1} H_k(w_N^l z) G_k(z)$$

Jeżeli spełnione są warunki:

$$A_0(z) = \frac{1}{N} \sum_{k=0}^{M-1} H_k(z) G_k(z) = \alpha \cdot z^{-n_0} \quad (*)$$

$$A_l(z) = 0, \quad l = 1, 2, \dots, N-1$$

to odtworzenie:
$$\hat{x}(n) = \alpha \cdot x(n - n_0)$$

jest przeskalowanym i opóźnionym, ale *niezniekształconym* sygnałem oryginalnym, a składowe zmodulowane są wytłumione.

Wzory (*) są warunkami dokładnej rekonstrukcji (*perfect reconstruction* – PR), które powinny być spełnione, przynajmniej w przybliżeniu, przez zaprojektowany bank filtrów.

Niezerowe wartości $A_l(z)$ dla $l \neq 0$, spowodowane nieidealnymi charakterystykami filtrów, powodują tzw. *przecieki widmowe* składowych pomiędzy podpasmami.

Tabela. Przykład kodowania podpasmowego sygnału 8kHz dla różnych szybkości transmisji

Pasma	Współcz. decymacji D z 8kHz	Zakresy podpasm (Hz)	Częstotl. próbkowania w podpasmach (Hz)	Liczba bitów/próbkę R_k dla <i>bitrate'u</i> $I =$		
				16 kb/s	24 kb/s	32 kb/s
0	8	0 - 500	1000	4	5	5
1	8	500 - 1000	1000	4	5	5
2	4	1000 - 2000	2000	2	4	4
3	4	2000 - 3000	2000	2	3	4
4	4	3000 - 4000	2000	0	0	3



9. Filtracja optymalna i adaptacyjna

Zagadnienie estymacji (odtworzenia) użytecznego sygnału $s(n)$ na podstawie innego dostępnego sygnału $x(n)$, który jest zaszumioną wersją $s(n)$:

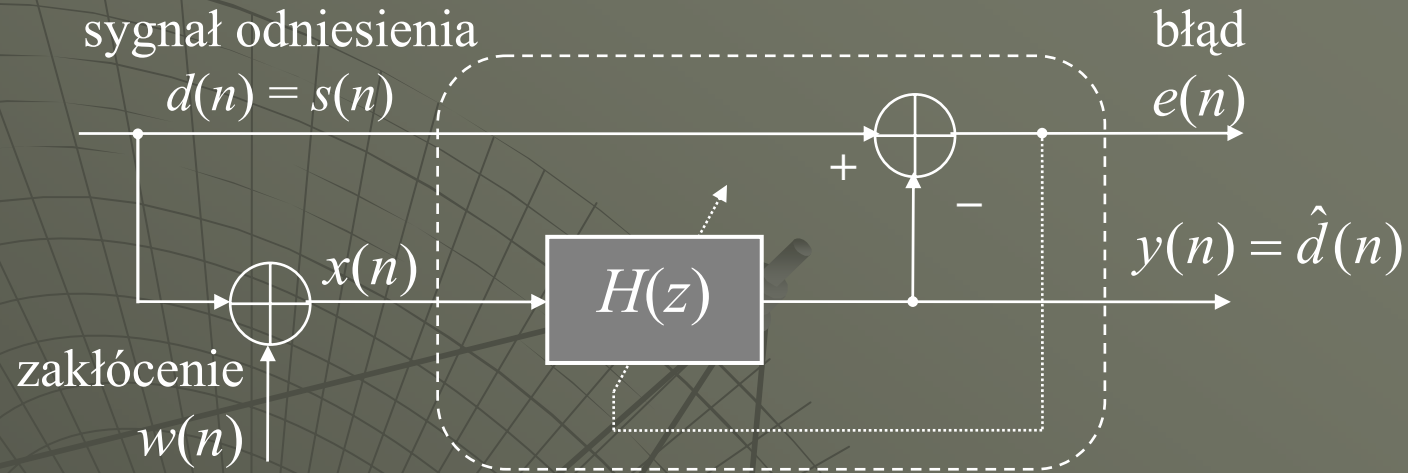
$$x(n) = s(n) + w(n)$$

($w(n)$ – niepożądane zakłócenie)

Widmo zakłócenia pokrywa się zwykle z widmem sygnału użytecznego i filtracja zakłócenia jest związana z usunięciem części sygnału użytecznego. Dlatego filtrów optymalnych nie projektuje się w dziedzinie częstotliwości, ale z wykorzystaniem statystycznych właściwości sygnału podlegającego filtracji.

Zwykle rozpatruje się estymacją liniową przy addytywnych zakłóceniach.

Filtracja optymalna. Filtr Wienera



Rys. Struktura filtra optymalnego. Wyjście filtra jest estymatą sygnału odniesienia

Sygnały wejściowe: $x(n)$ – sygnał filtrowany, $d(n)$ – sygnał odniesienia.

Sygnały wyjściowe: $y(n)$ – wynik filtracji, $e(n) = d(n) - y(n)$ – błąd estymacji.

Zakładamy, że sygnały wejściowe są *stacjonarne* (w szerszym sensie). W takim przypadku parametry $H(z)$ są wyznaczone jeden raz przez minimalizację kryterium błędu związanego z $e(n)$.

Projektuje się zwykle filtry nierekursywne (o skończonej odpowiedzi impulsowej SOI-FIR). Filtry NOI-IIR są potencjalnie niestabilne.

$$H(z) = h_0 + h_1 z^{-1} + h_2 z^{-2} + \dots + h_M z^{-M}$$

$$y(n) = \sum_{k=0}^M h_k x(n-k) = \mathbf{h}^T \mathbf{x}(n)$$

gdzie: $\mathbf{h} = [h_0, h_1, h_2, \dots, h_M]^T$ $\mathbf{x}(n) = [x(n), x(n-1), x(n-2), \dots, x(n-M)]^T$

Przez filtrację optymalną rozumie się filtrację minimalizującą kryterium wartości oczekiwanej błędu kwadratowego:

$$J(\mathbf{h}) = E[e^2(n)]$$

$$e(n) = d(n) - y(n) = d(n) - \sum_{k=0}^M h_k x(n-k)$$

Błąd dopasowania jest funkcją współczynników filtra $\{h_k\}$:

$$J(\mathbf{h}) = E \left[\left(d(n) - \mathbf{h}^T \mathbf{x}(n) \right)^2 \right] = E[d^2(n)] - 2\mathbf{h}^T \mathbf{r}_{dx}^{(n)} + \mathbf{h}^T \mathbf{R}_{xx}^{(n)} \mathbf{h}$$

gdzie:

$\mathbf{R}_{xx}^{(n)} = E[\mathbf{x}(n)\mathbf{x}^T(n)]$ - macierz autokorelacji sygnału wejściowego

$$\left[\mathbf{R}_{xx}^{(n)} \right]_{i,j} = R_{xx}^{(n)}(i-j) = E[x(n-i)x(n-j)], \quad i, j = 0, 1, \dots, M$$

$\mathbf{r}_{dx}^{(n)} = E[d(n)\mathbf{x}(n)]$ - wektor korelacji wzajemnej $d(n)$ i $\mathbf{x}(n)$:

$$\left[\mathbf{r}_{dx}^{(n)} \right]_i = r_{dx}^{(n)}(i) = E[d(n)x(n-i)], \quad i = 0, 1, \dots, M$$

Warunek ekstremum (minimum) $E[e^2(n)]$:

$$\frac{\partial J(\mathbf{h})}{\partial \mathbf{h}} = -2\mathbf{r}_{dx}^{(n)} + 2\mathbf{R}_{xx}^{(n)} \mathbf{h} = \mathbf{0}$$

Warunek minimum prowadzi do układu liniowego tzw. *równań normalnych*:

$$\mathbf{R}_{xx}^{(n)} \cdot \mathbf{h} = \mathbf{r}_{dx}^{(n)}$$

Rozwiązanie tych równań daje optymalne współczynniki filtra (filtr Wienera):

$$\mathbf{h}^{opt} = [\mathbf{R}_{xx}^{(n)}]^{-1} \mathbf{r}_{dx}^{(n)}$$

$\det \mathbf{R}_{xx}^{(n)} \neq 0$ - warunek wystarczającego pobudzenia

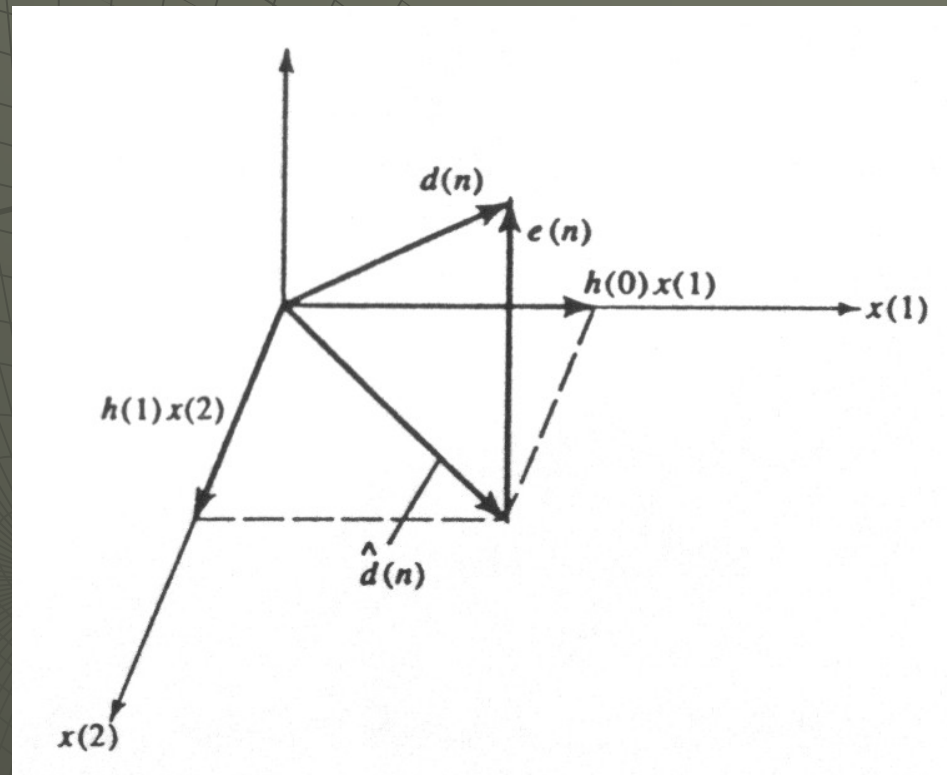
$$\begin{bmatrix} h_0^{opt} \\ h_1^{opt} \\ \vdots \\ h_M^{opt} \end{bmatrix} = \begin{bmatrix} R_{xx}^{(n)}(0) & R_{xx}^{(n)}(1) & \cdots & R_{xx}^{(n)}(M) \\ R_{xx}^{(n)}(1) & R_{xx}^{(n)}(0) & \cdots & R_{xx}^{(n)}(M-1) \\ \vdots & \vdots & \ddots & \vdots \\ R_{xx}^{(n)}(M) & R_{xx}^{(n)}(M-1) & \cdots & R_{xx}^{(n)}(0) \end{bmatrix}^{-1} \begin{bmatrix} r_{dx}^{(n)}(0) \\ r_{dx}^{(n)}(1) \\ \vdots \\ r_{dx}^{(n)}(M) \end{bmatrix}$$

Dla optymalnego filtra Wienera zachodzi zależność:

$$\frac{\partial J(\mathbf{h})}{\partial h_k} = 2E \left[e(n) \frac{\partial e(n)}{\partial h_k} \right] = -2E[e(n)x(n-k)] = 0, \quad k = 0, 1, \dots, M$$

Oznacza to, że filtr optymalny usuwa korelację błędu estymacji ze wszystkimi poprzednimi (i bieżącą) próbkami sygnału wejściowego (właściwość ortogonalności):

$$r_{ex}^{(n)}(k) = E[e(n)x(n-k)] = 0, \quad k = 0, 1, \dots, M$$



Rys. Geometryczna interpretacja właściwości ortogonalności estymaty otrzymanej metodą błędu średniokwadratowego MSE dla dwóch wymiarów ($M=2$). Wyjście filtra $y(n) = \hat{d}(n)$

Estymacja współczynników filtra optymalnego z próbek

W zagadnieniach praktycznych teoretyczne kryterium oraz korelacje wyznaczone z wartości oczekiwanych zastępuje się ich estymatami

$\hat{\mathbf{R}}_{xx}$ i $\hat{\mathbf{r}}_{dx}$ obliczonymi z próbek sygnałów $x(n)$ i $d(n)$:

$$\mathbf{x} = [x(0), x(1), x(2), \dots, x(N-1)]^T \quad \mathbf{d} = [d(0), d(1), d(2), \dots, d(N-1)]^T$$

Zwykle liczba próbek jest dużo większa niż długość filtra $N \gg M$ i otrzymuje się *układ nadokreślony*, tzn. *liczba równań jest większa niż liczba niewiadomych* (poszukiwanych współczynników filtra), rozwiązywany przez minimalizację sumy błędów kwadratowych dopasowania.

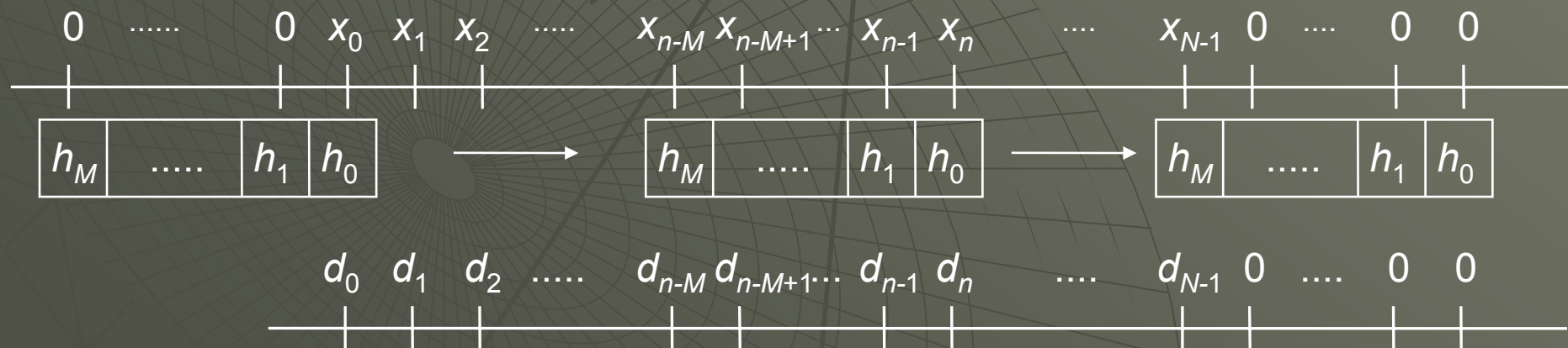
Istnieją dwie metody obliczeń (w zależności od zakresu sumowania we wzorze na estymatę błędu dopasowania) dające różne wyniki.

1. Metoda autokorelacji

$$J_n(\mathbf{h}) = \sum_{n=0}^{N+M-1} e^2(n), \quad e(n) = d(n) - y(n)$$

Jest $N+M$ równań na wartości wyjściowe $y(n)$. Kolejne próbki $y(n)$ są obliczane jako suma iloczynów współczynników filtra i odpowiadających im próbek $x(n)$ (jak na rysunku).

Ciąg próbek sygnału wejściowego jest na krańcach uzupełniany zerami. Poza pierwszymi N równaniami w pozostałych po prawej stronie występują zera.

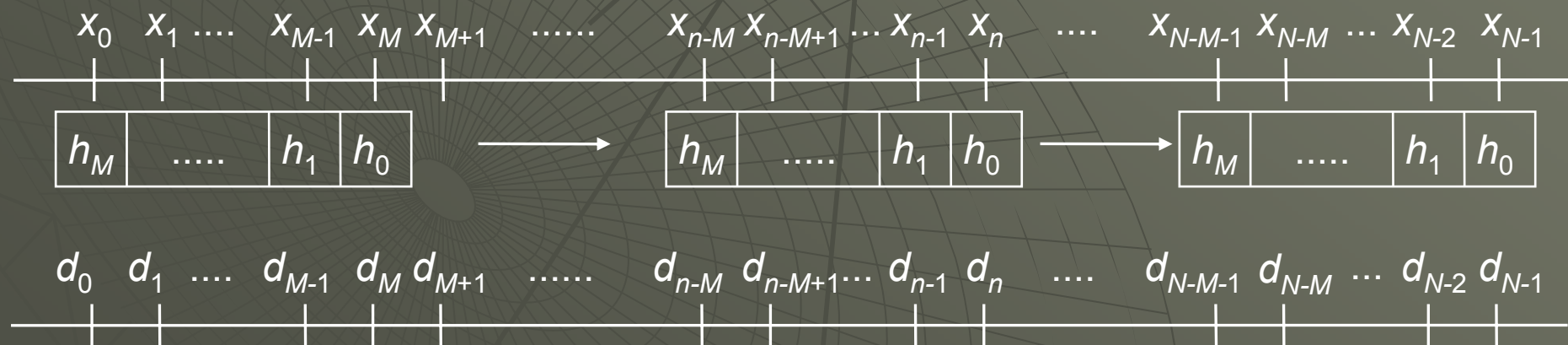


Metoda autokorelacji prowadzi do symetrycznej macierzy *równań normalnych* i gwarantuje, że filtr $H(z)$ jest minimalnofazowy.

2. Metoda kowariancji

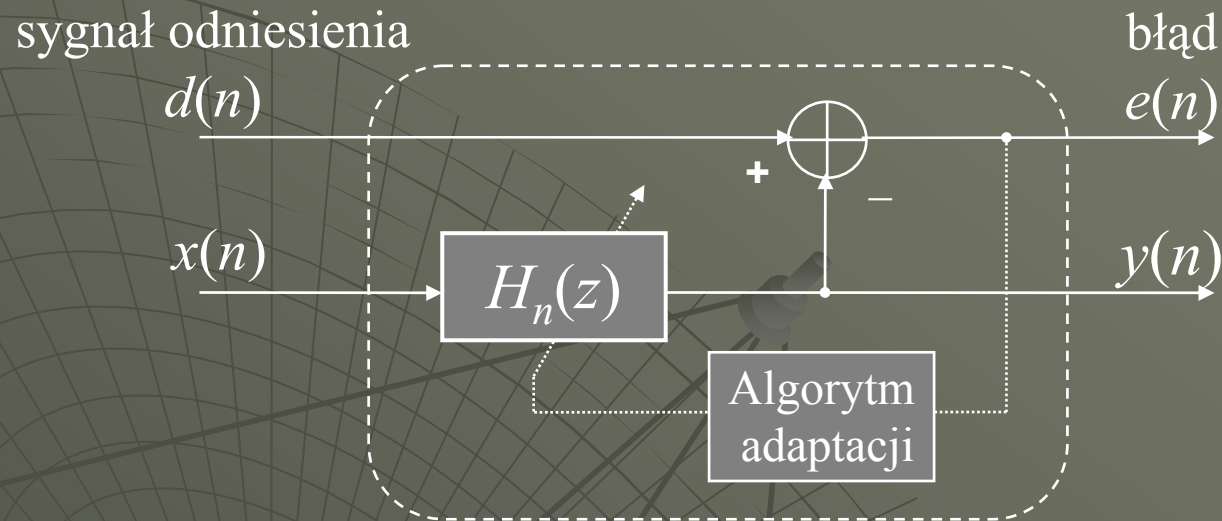
$$J_n(\mathbf{h}) = \sum_{n=M}^{N-1} e^2(n), \quad e(n) = d(n) - y(n)$$

Jest $N-M$ równań na wartości wyjściowe $y(n)$. Tworzenie równań jest ograniczone tylko do takiego zakresu, żeby wektor współczynników filtra "nie wychodził" poza rekord próbek wejściowych. Nie wykorzystuje się informacji z pierwszych M próbek sygnału odniesienia.



Metoda kowariancji nie gwarantuje minimalnofazowości $H(z)$.

Filtracja adaptacyjna (parametry filtra zmienne w czasie)



Rys. Struktura filtra adaptacyjnego. Parametry $\mathbf{h}(n)$ filtra $H_n(z)$ są *zmiennie* i dopasowywane na *bieżąco* według określonego algorytmu adaptacyjnego minimalizującego kryterium błędu związane z $e(n)$

Jeżeli wszystkie sygnały w układzie są stacjonarne, to po okresie adaptacji współczynniki filtra powinny się ustalić i przyjąć wartości optymalne (warunek zbieżności procesu adaptacji):

$$\lim_{n \rightarrow \infty} \mathbf{h}(n) = \mathbf{h}^{opt}$$

W adaptacyjnym przetwarzaniu sygnałów stosuje się metody optymalizacji wieloparametrycznej ($M+1$ parametrów $H_n(z)$), które polegają na poszukiwaniu minimów deterministycznych wieloargumentowych funkcji kosztu $J(\cdot)$.

Najczęściej stosowane są metody gradientowe, w których modyfikacja $\Delta \mathbf{h}(n)$ jest w każdej chwili proporcjonalna do wektora ujemnego gradientu funkcji kosztu:

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \Delta \mathbf{h}(n) = \mathbf{h}(n-1) - \frac{1}{2} \mu \mathbf{W}^{(n)} \cdot \nabla_n$$

gdzie gradient:
$$\nabla_n = \frac{\partial J(\mathbf{h}(n))}{\partial \mathbf{h}(n)} = \left[\frac{\partial J(n)}{\partial h_0(n)}, \frac{\partial J(n)}{\partial h_1(n)}, \dots, \frac{\partial J(n)}{\partial h_M(n)} \right]^T$$

$\mathbf{W}^{(n)}$ jest dodatkową macierzą wagową, która ma poprawiać zbieżność (np. odwrotność hesjanu $\mathbf{W}^{(n)} = [\nabla_n^2]^{-1}$ w procedurze Newtona-Raphsona).

Współczynnik adaptacji μ decyduje o szybkości przestrajania. Ze względu na zbieżność algorytmu zachodzi warunek:

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad \text{gdzie: } \lambda_{\max} - \text{największa wartość własna macierzy } \mathbf{R}_{xx}^{(n)}$$

Filtry adaptacyjne LMS (*Least Mean Squares*)

Kryterium błędu

$$J_n = e^2(n)$$

Filtracja LMS minimalizacja *chwilową* (a nie oczekiwaną) wartość błędu kwadratowego, dlatego nazywa się ją filtracją *bez pamięci* albo *optymalizacją stochastyczną*. Asymptotycznie dla $n \rightarrow \infty$ algorytm LMS minimalizuje średni błąd kwadratowy.

Metoda ta obejmuje szeroką rodzinę algorytmów opisanych ogólną zależnością:

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \mu(n)\mathbf{W}(n)e(n)\mathbf{x}(n)$$

gdzie $\mathbf{W}(n)$ jest jak poprzednio macierzą wagową, która ma poprawiać zbieżność adaptacji.

1. Filtr LMS

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \mu e(n) \mathbf{x}(n)$$

$$e(n) = d(n) - y(n) \quad y(n) = \mathbf{h}^T(n-1) \mathbf{x}(n)$$

Zaleta: prostota i mała złożoność obliczeniowa.
Wada: wolna zbieżność algorytmu.

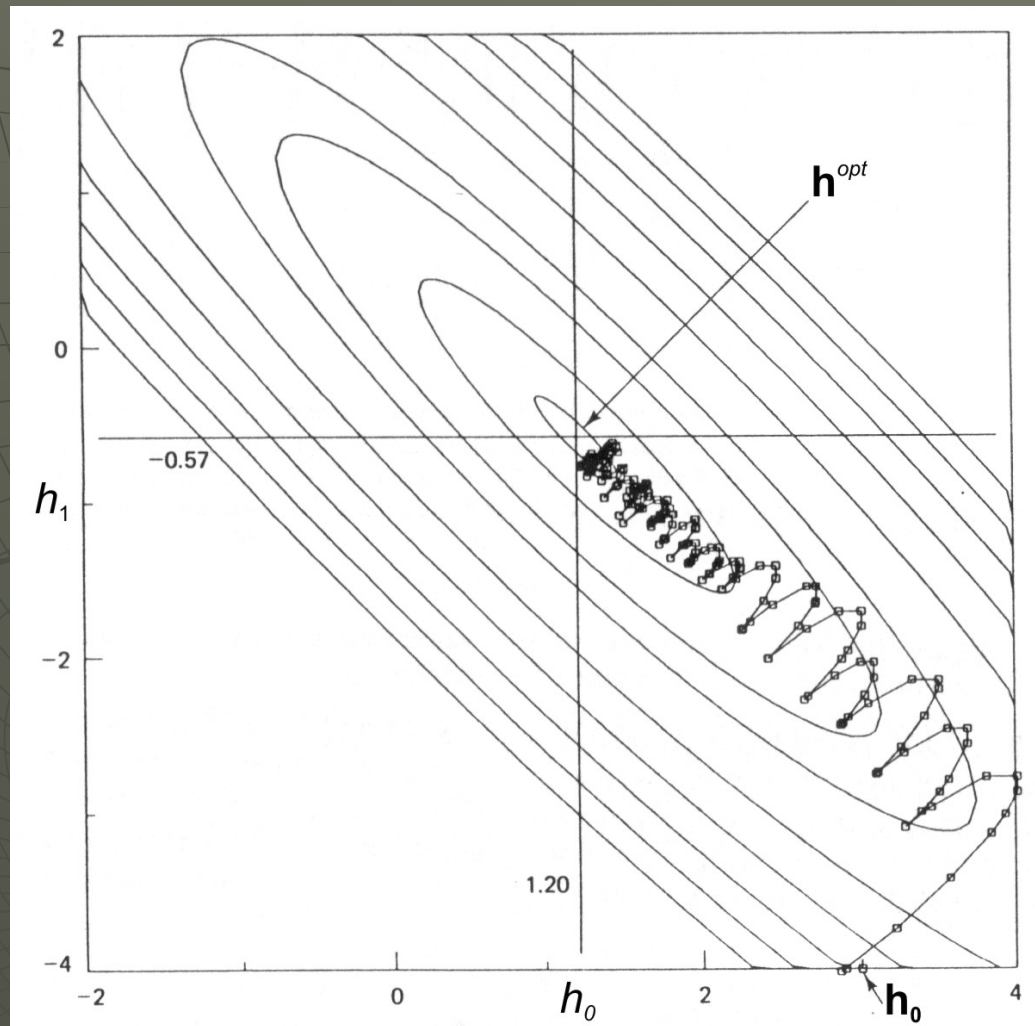
2. Normalizowany filtr LMS (NLMS)

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \mu(n) e(n) \mathbf{x}(n)$$

$$\mu(n) = \frac{\mu}{\gamma + \mathbf{x}^T(n) \mathbf{x}(n)} = \frac{\mu}{\gamma + \sum_{k=0}^M x^2(n-k)},$$

Parametr $\mu(n)$ jest unormowany względem mocy sygnału wejściowego. Mały parametr γ zapobiega zerowaniu się mianownika.

Filtry NLMS charakteryzują się szybszą zbieżnością i lepszą stabilnością w porównaniu z LMS.



Rys. Trajektoria zbieżności współczynników filtra NLMS do wartości optymalnej w przestrzeni 2-wymiarowej dla układu kasowania korelacji (wejście filtra z szumem)

3. Zdekorelowany filtr LMS

Zamiast $\mathbf{x}(n)$ na wejście podawany jest zdekorelowany sygnał:

$$\mathbf{v}(n) = \mathbf{x}(n) - \alpha(n)\mathbf{x}(n-1), \quad \alpha(n) = \frac{\mathbf{x}^T(n)\mathbf{x}(n-1)}{\mathbf{x}^T(n-1)\mathbf{x}(n-1)}$$

który ma stosunek $\lambda_{min}/\lambda_{max}$ bliski jedności, co zwiększa szybkość adaptacji.

$\alpha(n)$ - współczynnik korelacji wektorów $\mathbf{x}(n)$ i $\mathbf{x}(n-1)$.

Filtry adaptacyjne RLS (*Recursive Least Squares*)

Popularnym algorytmem adaptacji jest algorytm LS (*Least Squares*) z kryterium błędu w postaci sumy kwadratów:

$$J_n = \sum_{k=0}^n e^2(k)$$

lub sumy ważonej WLS (*Weighted LS*):

$$J_n = \sum_{k=0}^n \lambda^{n-k} e^2(k)$$

λ - współczynnik zapominania starych błędów z zakresu $0.95 \leq \lambda \leq 1$

Algorytm WLS umożliwia adaptację do zmieniających się charakterystyk sygnałów, kiedy dane z przeszłości tracą aktualność. Filtry LS nazywa się filtrami z *pamięcią*, ponieważ kryterium zależy od sumy błędów. W trakcie adaptacji trajektoria w przestrzeni parametrów dąży w kierunku minimum J mniej chaotycznie niż w przypadku metody LMS.

Równanie adaptacji współczynników:

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \mathbf{K}(n)e(n)$$

gdzie: $\mathbf{K}(n) = [\hat{\mathbf{R}}_{xx}^{(n)}]^{-1} \mathbf{x}(n)$ - wzmacnienie Kalmana

$$e(n) = d(n) - y(n) \quad y(n) = \mathbf{h}^T(n-1)\mathbf{x}(n)$$

Ze względu na konieczność odwracania w każdym kroku macierzy autokorelacji algorytm LS implementuje się w formie *rekurencyjnej* RLS (*Recursive LS*):

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \mathbf{K}(n)e(n)$$

$$\mathbf{K}(n) = \mathbf{P}(n)\mathbf{x}(n) = \frac{\lambda^{-1}\mathbf{P}(n-1)\mathbf{x}(n)}{1 + \lambda^{-1}\mathbf{x}^T(n)\mathbf{P}(n-1)\mathbf{x}(n)} \quad \text{gdzie: } \mathbf{P}(n) = [\hat{\mathbf{R}}_{xx}^{(n)}]^{-1}$$

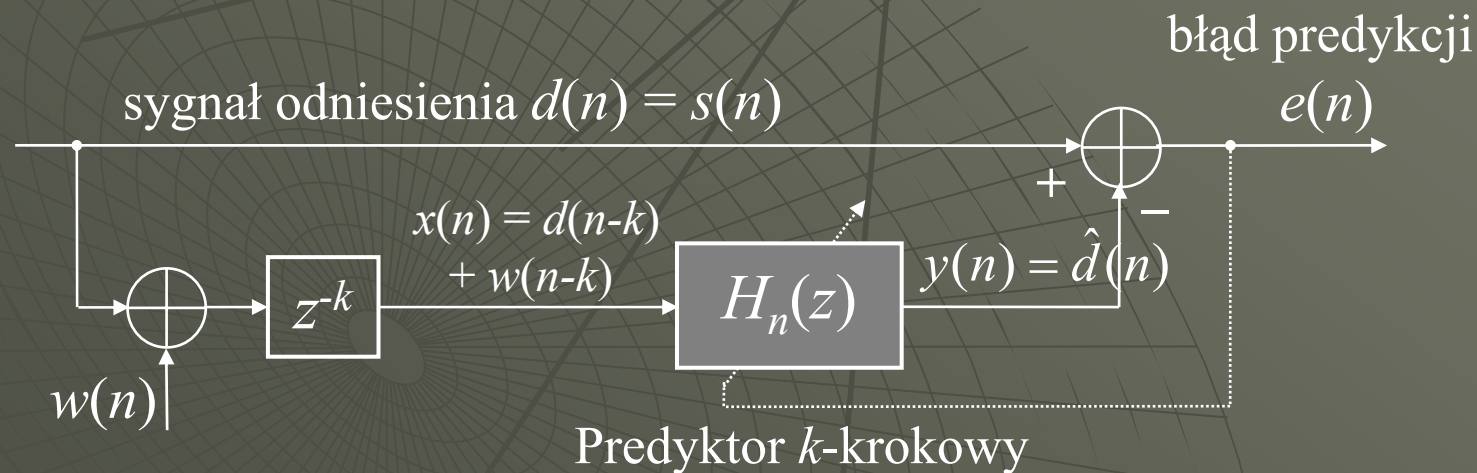
$$\mathbf{P}(n) = \lambda^{-1}[\mathbf{I} - \lambda^{-1}\mathbf{K}(n)\mathbf{x}^T(n)]\mathbf{P}(n-1)$$

z warunkami początkowymi: $\mathbf{P}(0) = \mathbf{I} \frac{1}{\hat{\sigma}_x^2}$, $\mathbf{h}(0) = \mathbf{h}_0$

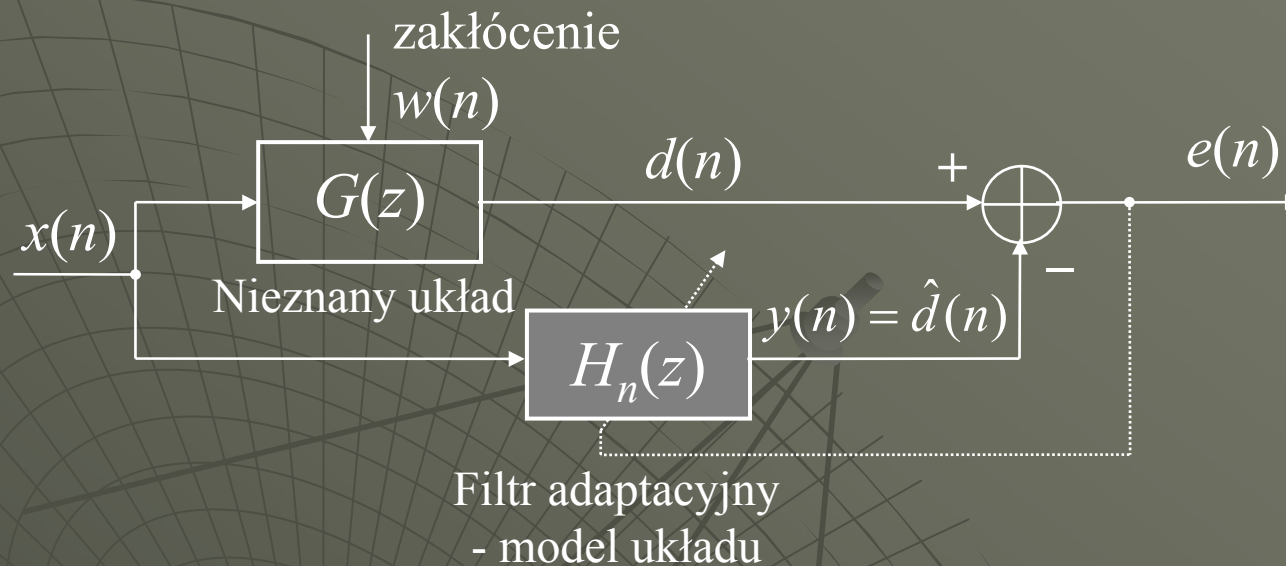
Algorytm RLS jest bardziej złożony obliczeniowo niż LMS, ale daje szybszą i płynniejszą zbieżność.

Typowe zastosowania filtracji adaptacyjnej

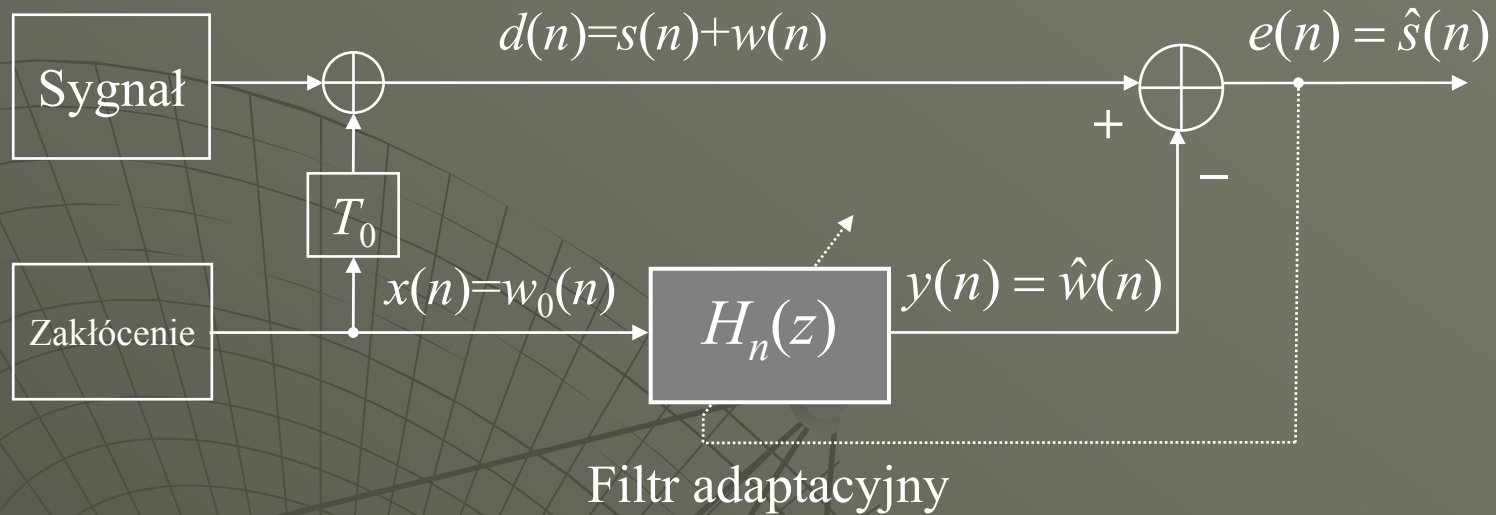
Wyjście filtra $y(n)$ jest zawsze estymatą tylko tej składowej sygnału odniesienia $d(n)$, która jest skorelowana z sygnałem wejściowym $x(n)$. Jeżeli $d(n) = s(n) + x_1(n)$, gdzie tylko $x_1(n)$ jest skorelowane z $x(n)$, to wyjście $y(n) = \hat{x}_1(n)$.



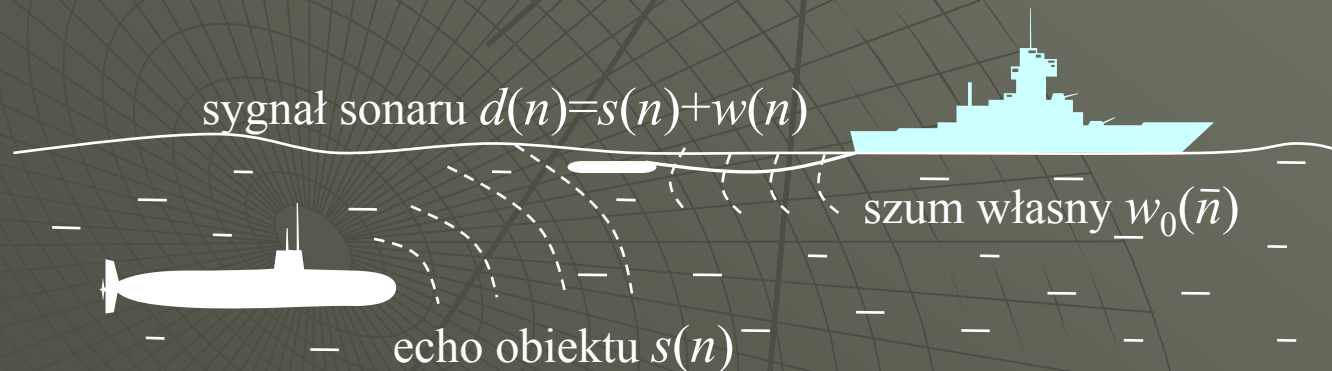
Rys. Adaptacyjna predykcja sygnału



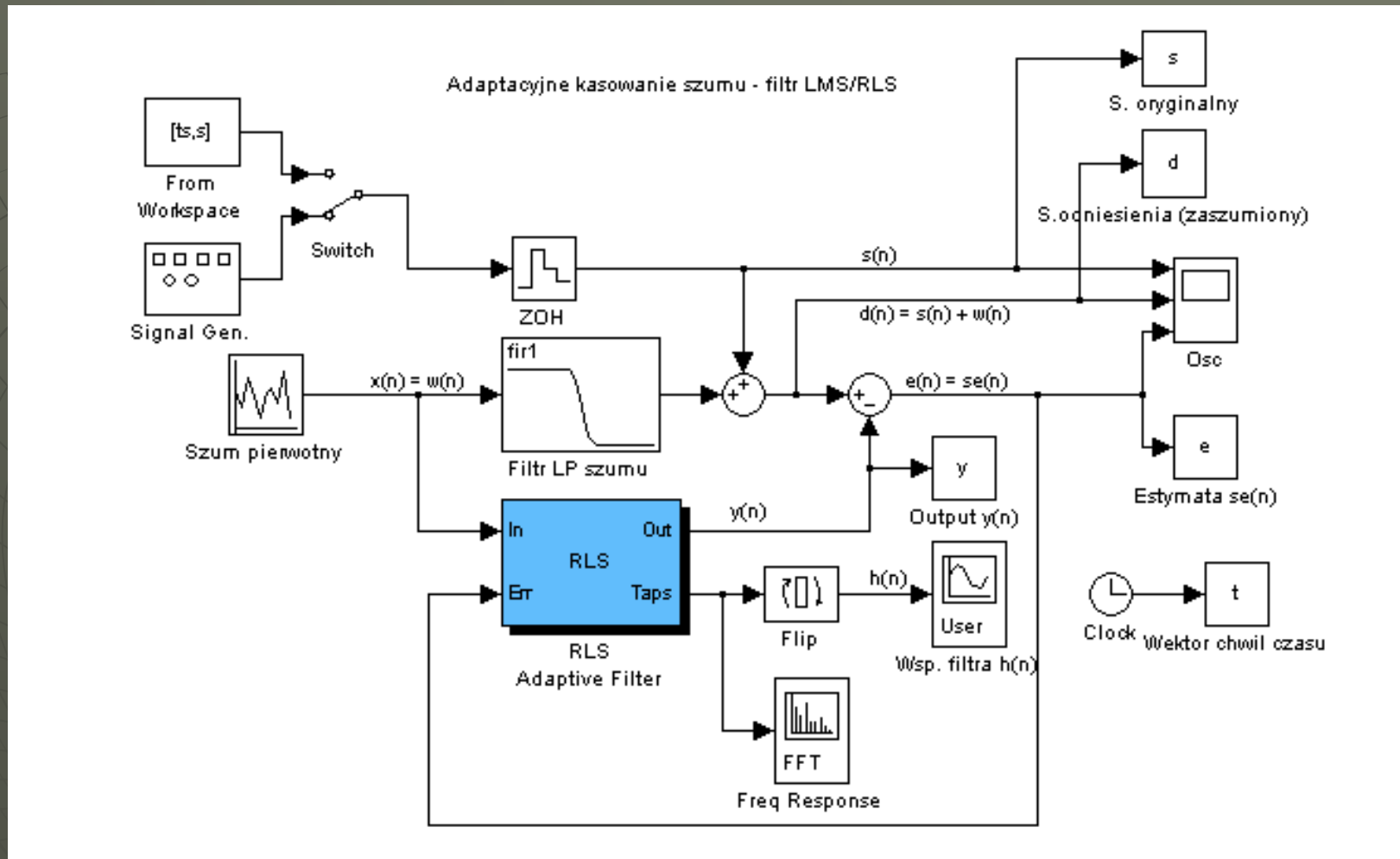
Rys. Identyfikacja adaptacyjna nieznanego układu dynamicznego. Transmitancja filtra adaptacyjnego jest estymatą transmitancji identyfikowanego układu: $H(z) = \hat{G}(z)$



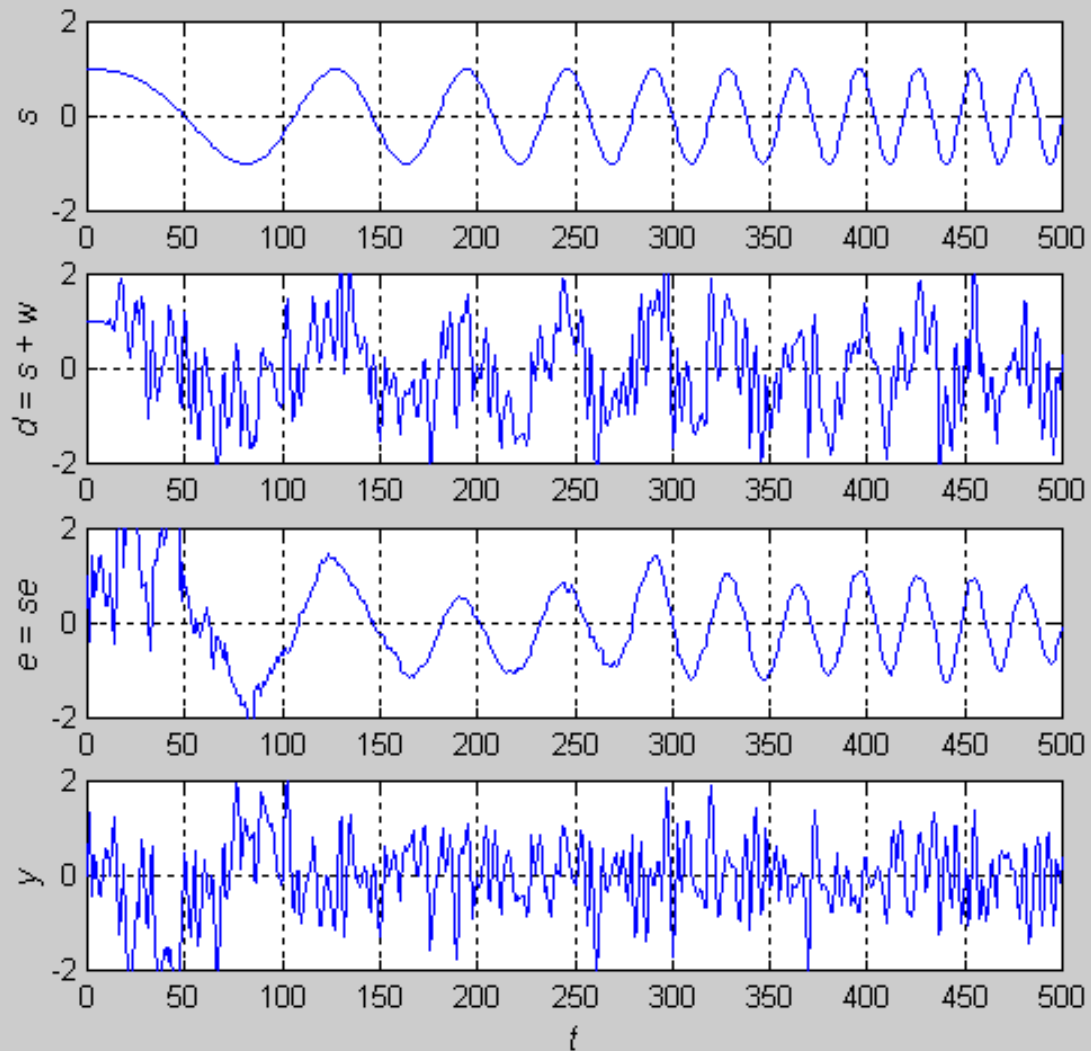
Rys. Adaptacyjne kasowanie zakłóceń (usuwanie interferencji/korelacji)



Rys. Przykład zastosowania adaptacyjnego kasowania szumu. Zakłócenie skorelowane z szumem własnym okrętu dociera do sonaru jako jego stłumiona i opóźniona wersja $w(n) = \alpha \cdot w_0(n - n_0)$. Zadaniem filtracji adaptacyjnej jest wytłumienie tej składowej w sygnale sonaru.




Rys. Przykład: Model układu kasowania szumu. Sygnał $s(n)$ – LFM od 0.02 do 0.25 rad/s, sygnał wejściowy $x(n)$ - szum biały o wariancji $\sigma_w^2=1$, zakłócenie $w(n) - x(n)$ filtrowany filtrem LP o $\Omega_p/\pi=0.5$, filtr adaptacyjny RLS długości $M= 31$, warunki początkowe $\mathbf{h}(0)=0$, $R_{xx}(0)=0.5$, $T_s=1$.




Rys. Wyniki symulacji układu adaptacyjnego kasowania szumu

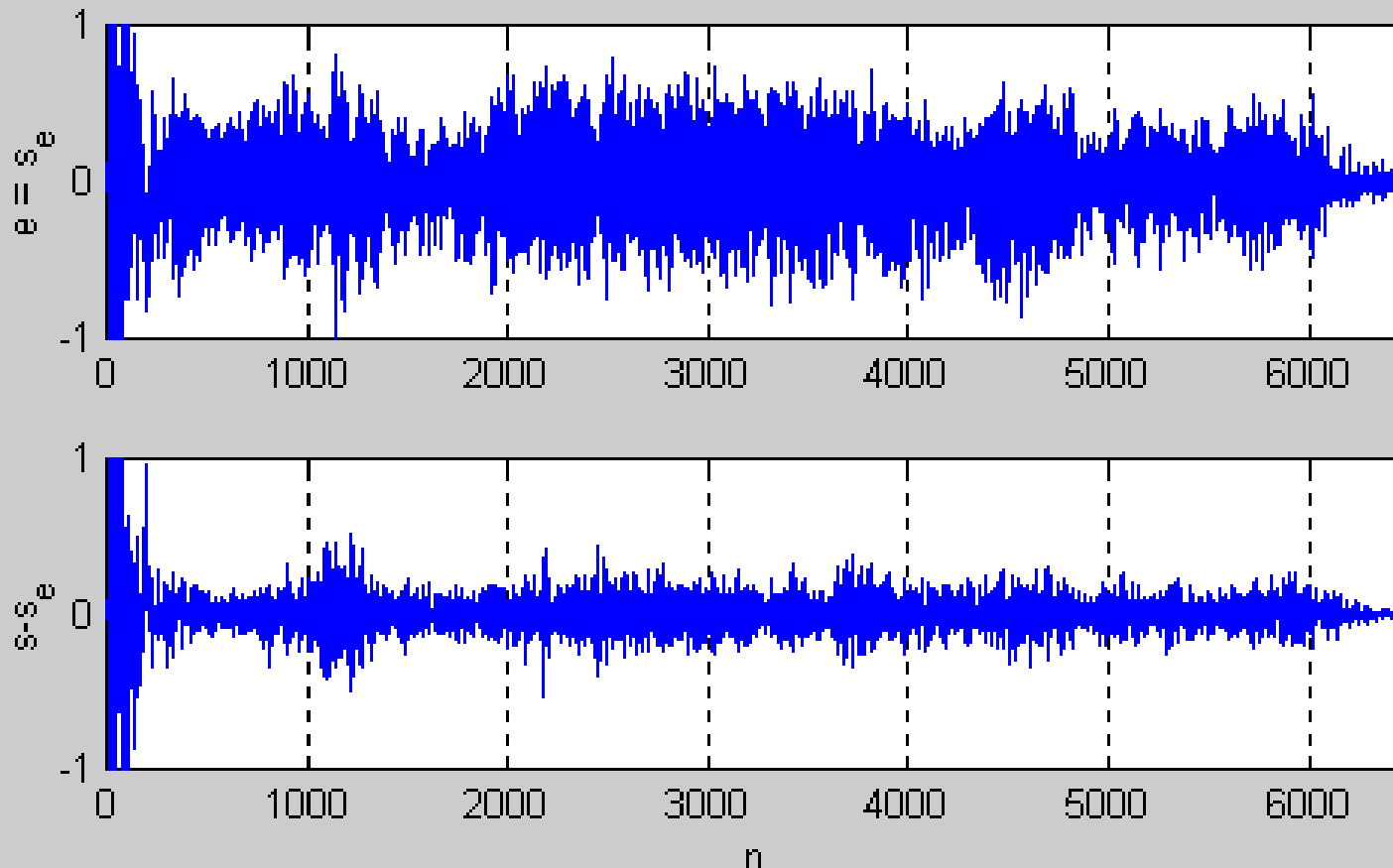
Przykład: Adaptacyjne usuwanie szumu z sygnału dźwiękowego

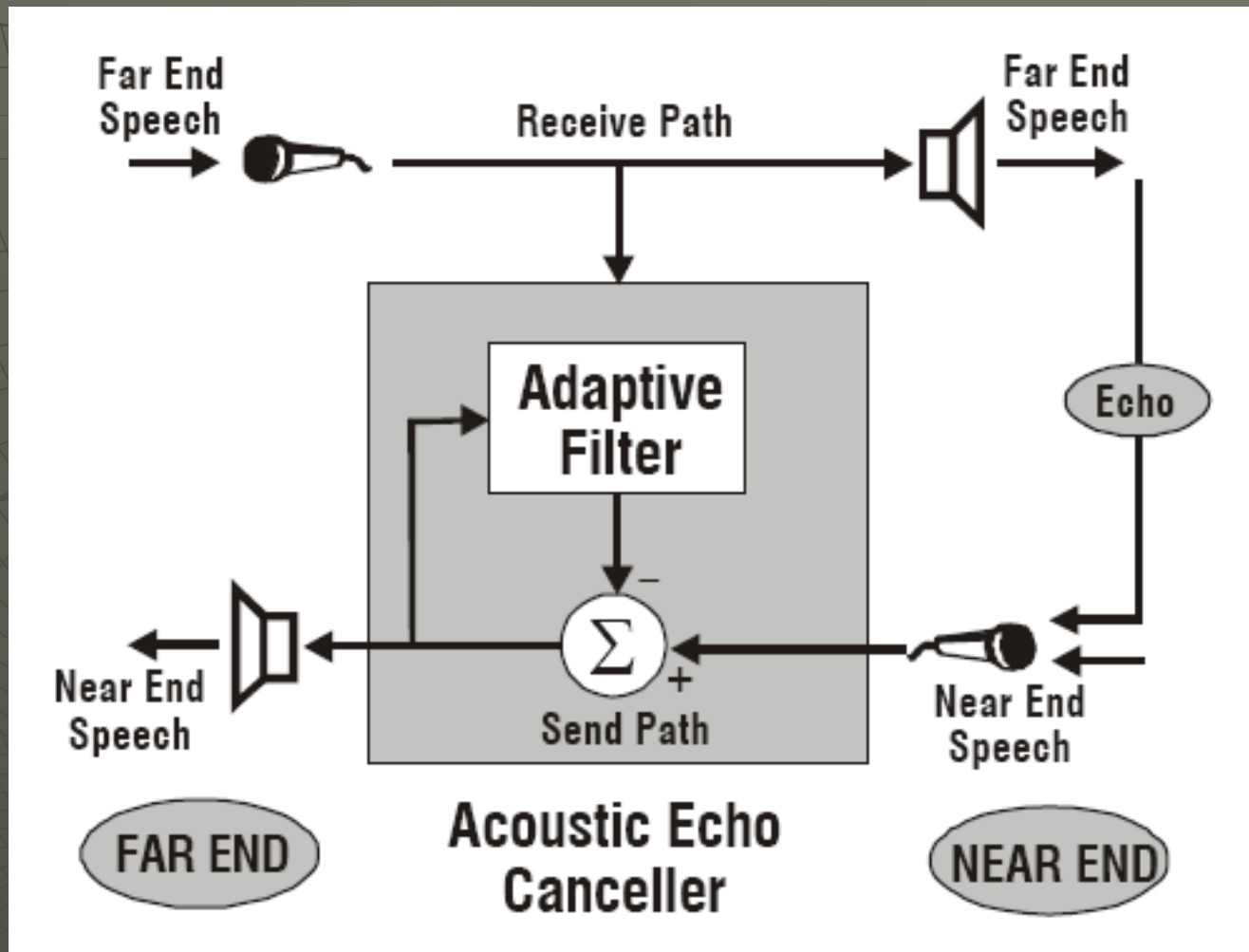
($\sigma_w^2=3$):

 $s(n)$, fire8.wav
 $f_s=8\text{kHz}$, 16bit

 $d(n)=s(n)+w(n)$

 $e(n)=\hat{s}(n)$





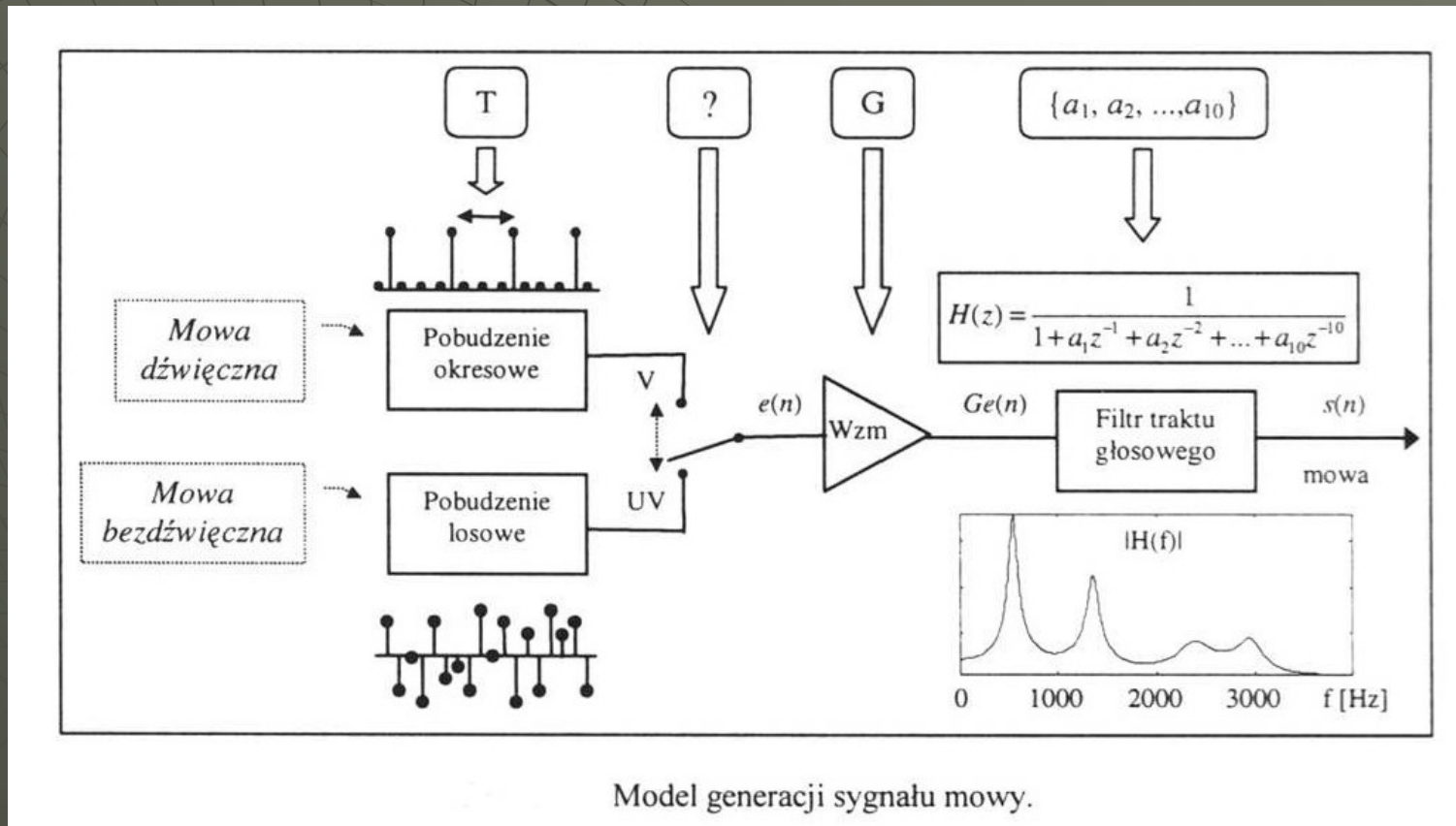
Rys. Filtr adaptacyjny w układzie usuwania echa. Sygnałem odniesienia jest sygnał abonenta bliskiego zakłócony echem od abonenta odległego. Sprężenie zwrotne to sygnał wychodzący z wytłumionym echem.



10. Liniowe kodowanie predykcyjne LPC

Model traktu głosowego (dekoder LPC)

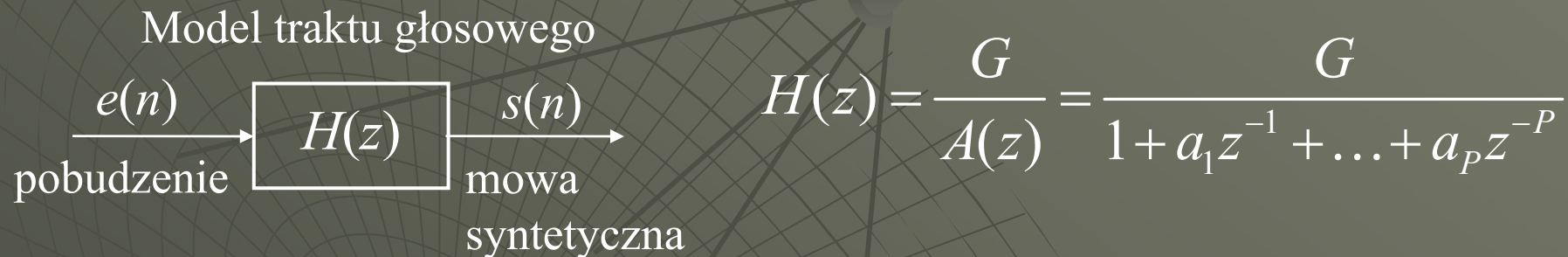
W zastosowaniach telekomunikacyjnych przyjmuje się, że w sygnale mowy podstawowa informacja jest zawarta w paśmie 300-3500 Hz (stąd $f_s=8\text{kHz}$).



Rys. Generacja (syntezy) sygnału mowy w metodzie LPC (*Linear Predictive Coding*)

Charakterystyka amplitudowa **traktu głosowego** ma cztery maksima (*formanty* – związane z częstotliwościami rezonansowymi). Do modelowania tego typu widma (ze stromymi pikami) stosuje się filtry IIR *all-pole*.

W kodowaniu LPC przyjmuje się filtr rzędu $P=10$ (modelowanie 4 pików wymaga co najmniej 4 par zespolonych biegunów):



Odpowiednie równanie różnicowe filtra traktu głosowego:

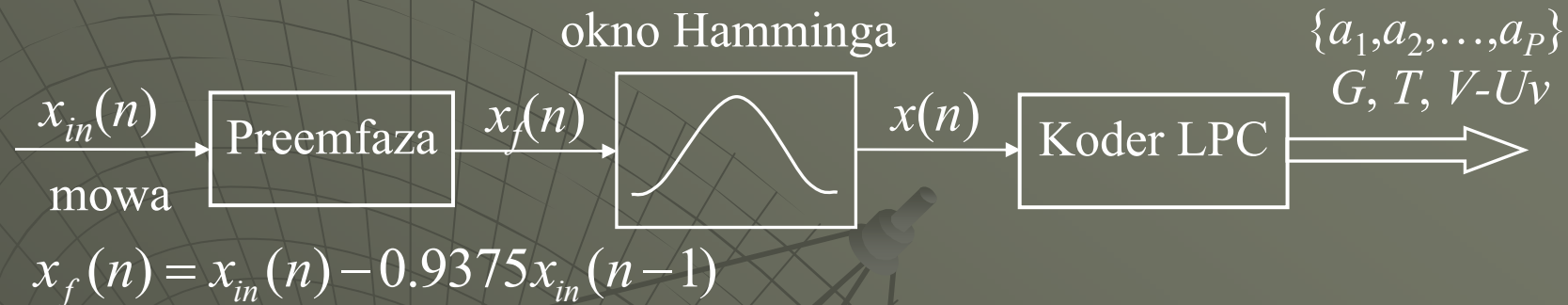
$$s(n) = -\sum_{k=1}^P a_k s(n-k) + Ge(n)$$

Pobudzenie filtra syntezy:

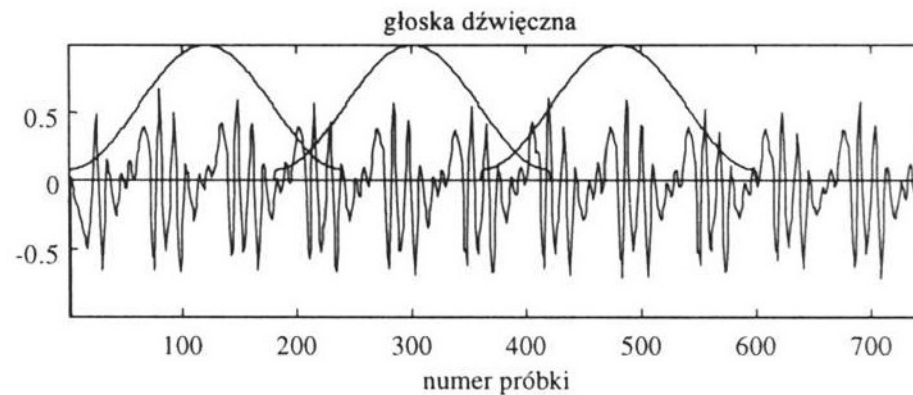
- dla głosek dźwięcznych (*voiced*) – zbliżone do impulsowego o częstotliwościach ok. 80-480 Hz dla mężczyzn (bas – baryton – tenor) i ok. 160-960 Hz dla kobiet (alt – mezzosopran – sopran),
- dla głosek bezdźwięcznych (*unvoiced*) – zbliżone do szumu białego

Kodowanie LPC

Wstępne przetwarzanie sygnału



Preemfaza podbija piki widma w zakresie wyższych częstotliwości.



Wycinanie fragmentu mowy do analizy: okno Hamminga o długości 240 elementów jest przesuwane z krokiem 180 próbek i mnożone z sygnałem.

Rys. Wycinanie ramki sygnału mowy, $f_s=8\text{kHz}$, okno analizy $N=240$ próbek (30ms), ramka syntezy $M=180$ próbek (22.5ms)

Wyznaczanie optymalnego filtra traktu głosowego

Do optymalizacji filtra traktu stosuje się modelowanie predykcyjne (1-krokowe):

$$\hat{x}(n | n-1) = -\sum_{k=1}^P a_k x(n-k)$$

Współczynniki $\mathbf{a}=[a_1, a_2, \dots, a_P]^T$ będą potem wykorzystywane w dekodерze do syntezy mowy. Błąd predykcji:

$$\varepsilon(n) = x(n) - \hat{x}(n) = x(n) + \sum_{k=1}^P a_k x(n-k), \quad \varepsilon(n) = Ge(n)$$

Dla każdej ramki N próbek poszukuje się optymalnych wartości współczynników \mathbf{a} tak, aby średniokwadratowy błąd predykcji był jak najmniejszy. Minimalizowana jest funkcja kosztu:

$$V(\mathbf{a}) = \sum_{n=P}^{N-1} \varepsilon(n)^2$$

Optymalne współczynniki predyktora:

$$\mathbf{a} = -\mathbf{R}_{xx}^{-1} \mathbf{r}_{xx}$$

$$\mathbf{R}_{xx} = \begin{bmatrix} R_{xx}(0) & R_{xx}(1) & \cdots & R_{xx}(P-1) \\ R_{xx}(1) & R_{xx}(0) & \cdots & R_{xx}(P-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_{xx}(P-1) & R_{xx}(P-2) & \cdots & R_{xx}(0) \end{bmatrix}, \quad \mathbf{r}_{xx} = \begin{bmatrix} R_{xx}(1) \\ R_{xx}(2) \\ \vdots \\ R_{xx}(P) \end{bmatrix}$$

gdzie estymaty autokorelacji (metoda kowariancji):

$$R_{xx}(m) = \frac{1}{N-P} \sum_{n=P}^{N-1} x(n)x(n+m)$$

Rekurencyjny algorytm Levinsona-Durbina wyznaczania współczynników odbicia r_i filtra kratowniczowego

$$\begin{aligned}
 E_0 &= R_{xx}(0), \quad a_{0i} = 0 \\
 \text{for } i &= 1, 2, \dots, P: \left\{ \begin{aligned}
 r_i &= \frac{1}{E_{i-1}} \left[R_{xx}(i) + \sum_{j=1}^{i-1} a_{(i-1)j} R_{xx}(i-j) \right] \\
 E_i &= (1 - r_i^2) E_{i-1} \\
 a_{ii} &= -r_i \\
 \text{for } j &= 1, 2, \dots, i-1: \quad a_{ij} = a_{(i-1)j} - r_i a_{(i-1)(i-j)}
 \end{aligned} \right. \quad (1-4)
 \end{aligned}$$

gdzie:

r_i – współczynniki odbicia realizacji kratowniczowej (*lattice filter*) filtra *all-pole* $H(z)$ do syntezy mowy w dekodерze,

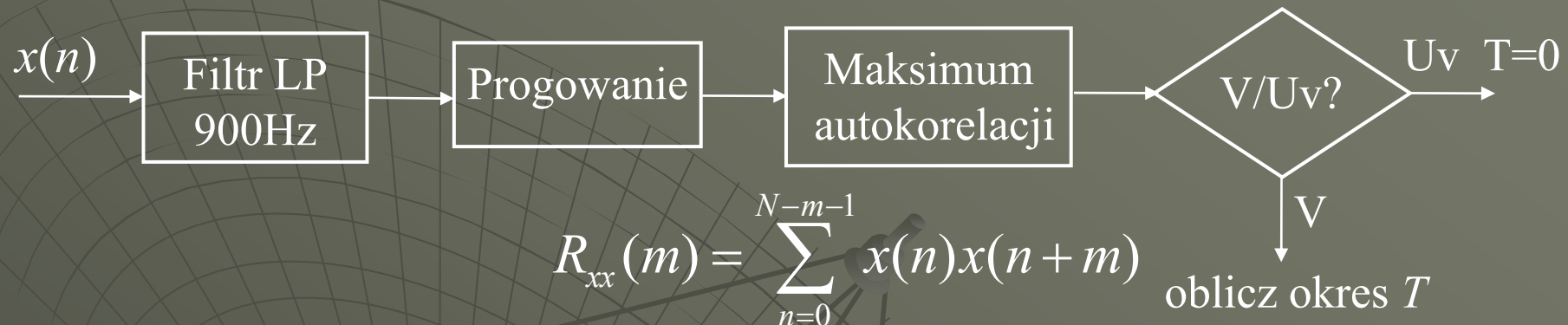
$a_{Pi} = a_i$ – współczynniki filtra FIR predykcji,

E_P – błąd kwadratowy pomiędzy rzeczywistym sygnałem wejściowym $x(n)$ a jego predykcją $\hat{x}(n)$

Równania (1-4) są rozwiązywane rekurencyjnie dla $i = 1, 2, \dots, P = 10$

Filtr kratownicowy ma większą odporność na błędy kwantowania - do zapisu współczynników wystarczy mniejsza liczba bitów.

Układ decyzyjny ramka dźwięczna-bezdźwięczna (V/U_v)



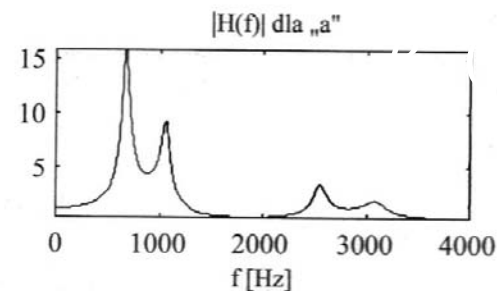
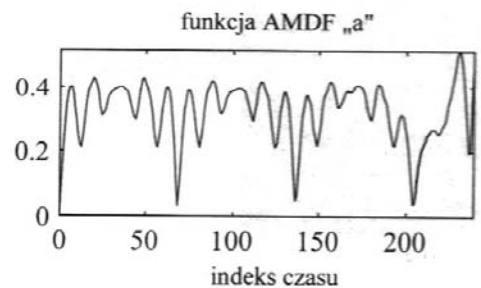
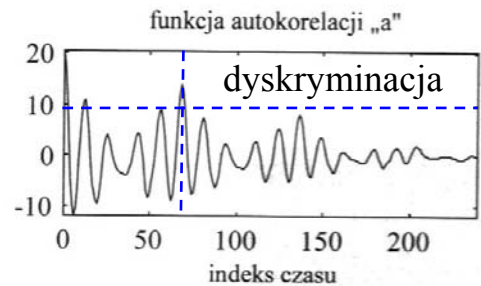
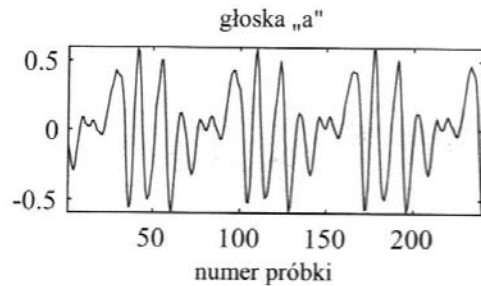
Stosuje się też zmodyfikowany algorytm polegający na analizie autokorelacji *wartości estymowanych* $\hat{x}(n)$ zamiast rzeczywistych $x(n)$:

$$R_{\hat{x}\hat{x}}(m) = \sum_{n=1}^{N-m-1} \hat{x}(n)\hat{x}(n+m) = \sum_{n=1}^P R_{aa}(n)R_{xx}(m-n)$$

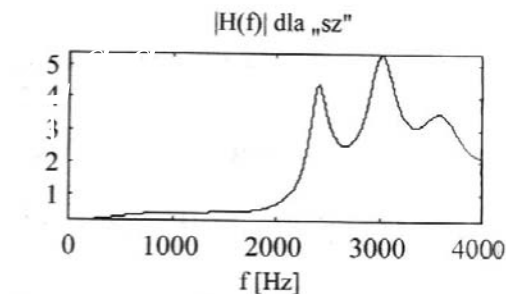
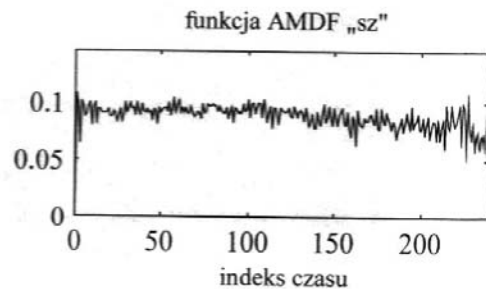
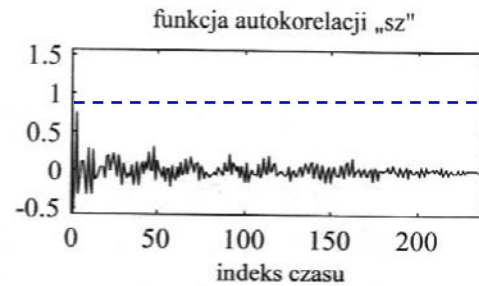
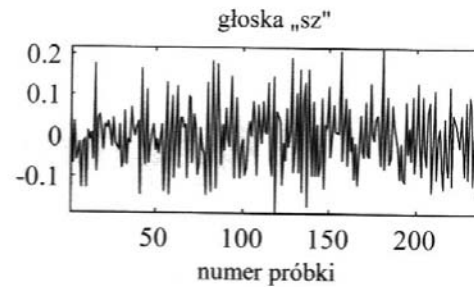
gdzie:

$$\hat{x}(n) = -\sum_{k=1}^P a_k x(n-k) \quad R_{aa}(m) = \sum_{i=1}^P a_i a_{i+m}$$

a) głoska „a”



b) głoska „sz”



Rys. Klasyfikowanie V/Uv ramki na podstawie autokorelacji R_{xx} . Poziom dyskryminacji

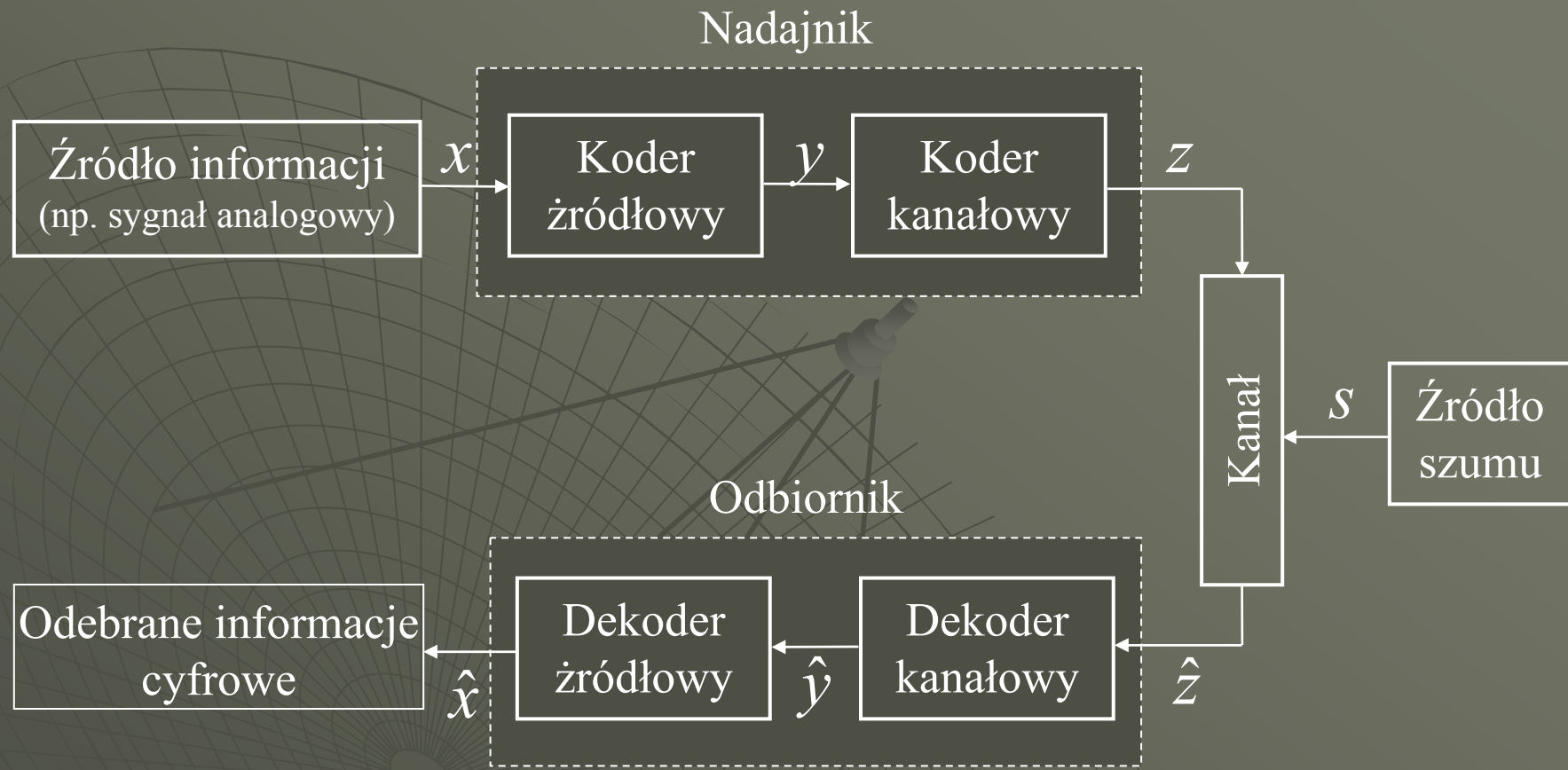
$$0.25 \leq \frac{R_{xx}^{peak}(m)}{R_{xx}(0)} \leq 0.35$$

dla $m > m_0$, gdzie $m_0 = f_s \cdot T_{min}$ (minimalny okres tonu podstawowego, czyli maksymalna częstotliwość, np. 900Hz

Na kodowaniu LPC oparty jest standard FS1015 (US Army, 1984):
 $I=2.4\text{kb/s}$, $R_{in}=12$ bitów, $f_s=8\text{kHz}$, ramka 22.5ms ($N=180$ próbek),
zakodowana ramka – 54 bity ($M=10$ skwantowanych
współczynników realizacji kratownicowej filtra, zakodowana
amplituda G i okres tonu podstawowego T , flaga V-Uv), kompresja
ok. 40 razy ($=96/2.4$).
(Do zapisu niektórych współczynników odbicia wystarczają tylko 2
bity.)

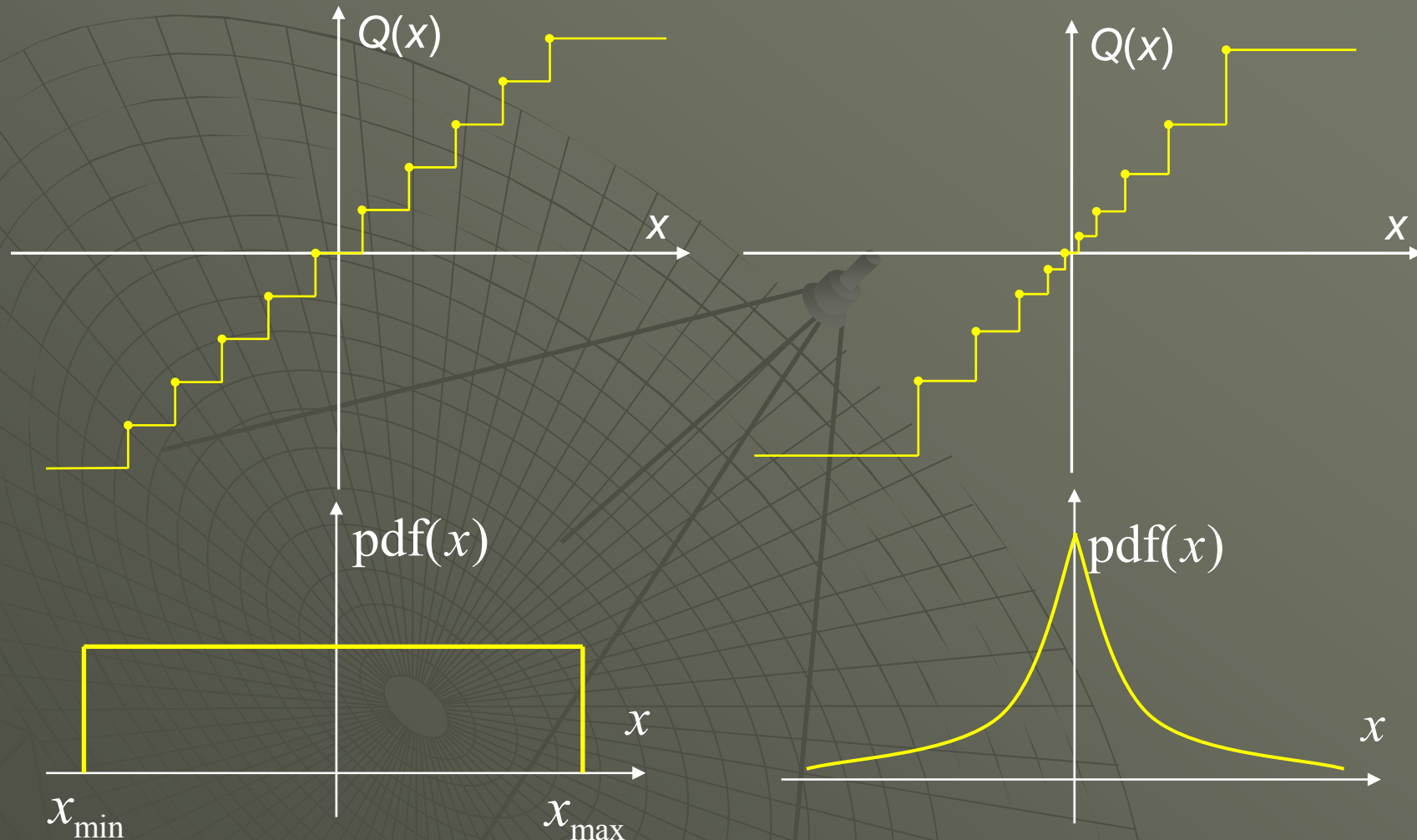


11. Elementy kodowania sygnałów

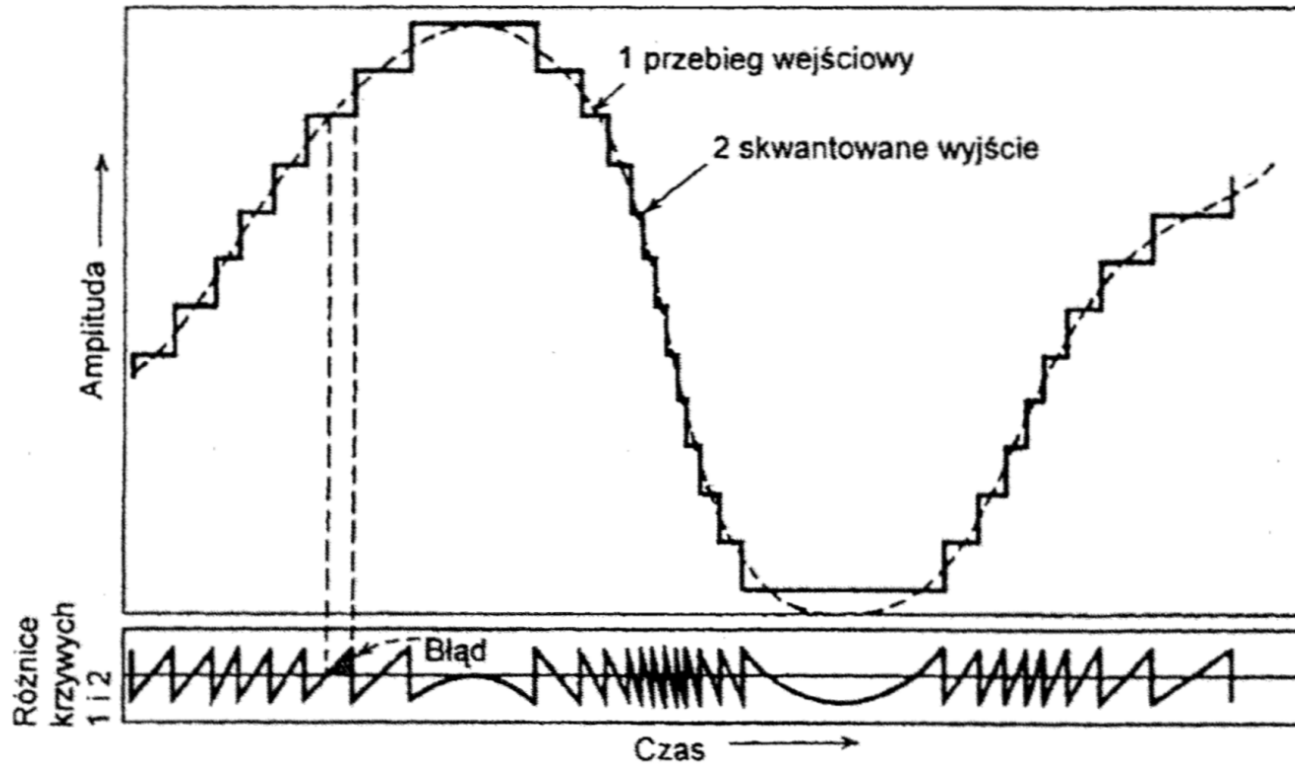


Rys. Podstawowe elementy modelu cyfrowego kanału komunikacyjnego. Koder źródłowy realizuje przetwarzanie A/C i/lub kompresję, koder kanałowy – modulację i dodawanie informacji nadmiarowej

Kodowanie PCM (Pulse-Code Modulation)



Rys. Kwantowanie jednostajne i niejednostajne związane z gęstością prawdopodobieństwa sygnału kwantowanego. Za pomocą rozkładu jednostajnego modeluje się np. sygnał wizyjny, do modelowania sygnału mowy stosuje się często rozkład Laplace'a.



Rys. Przebieg błędu kwantowania w czasie dla kwantowania jednostajnego bez przekroczenia zakresu

Kwantowanie niejednostajne dostosowane do (niejednostajnego) rozkładu prawdopodobieństwa sygnału kwantowanego zmniejsza wariancję błędu kwantowania.

Uzyskuje się je przetwarzając próbki z jednostajnego przetwornika A/C (liniowa PCM) przez charakterystykę komandera $C(x)$, często z jednoczesnym zmniejszeniem długości R słowa bitowego.

Zakres dynamiki (*dynamic range*) – stosunek największej wartości nie zniekształconego sygnału (sinusoidalnego w przypadku sygnałów audio) do wartości najmniejszej nie maskowanej szumami (amplitudy rms szumu - układów elektronicznych, połączeń, medium - w przypadku analogowych sygnałów audio).

W przypadku sygnałów cyfrowych zakres dynamiki jest ograniczony przez błąd kwantowania, $SNR \approx 6 \cdot R$ (dB), R – liczba bitów ADC lin-PCM.

- Dla kwantowania 16-bitowego teoretyczny zakres dynamiki to 96 dB (SNR), w praktyce obserwuje się ok. 90 dB.
- Zakres dynamiki słuchu ludzkiego to ok. 130-140 dB.

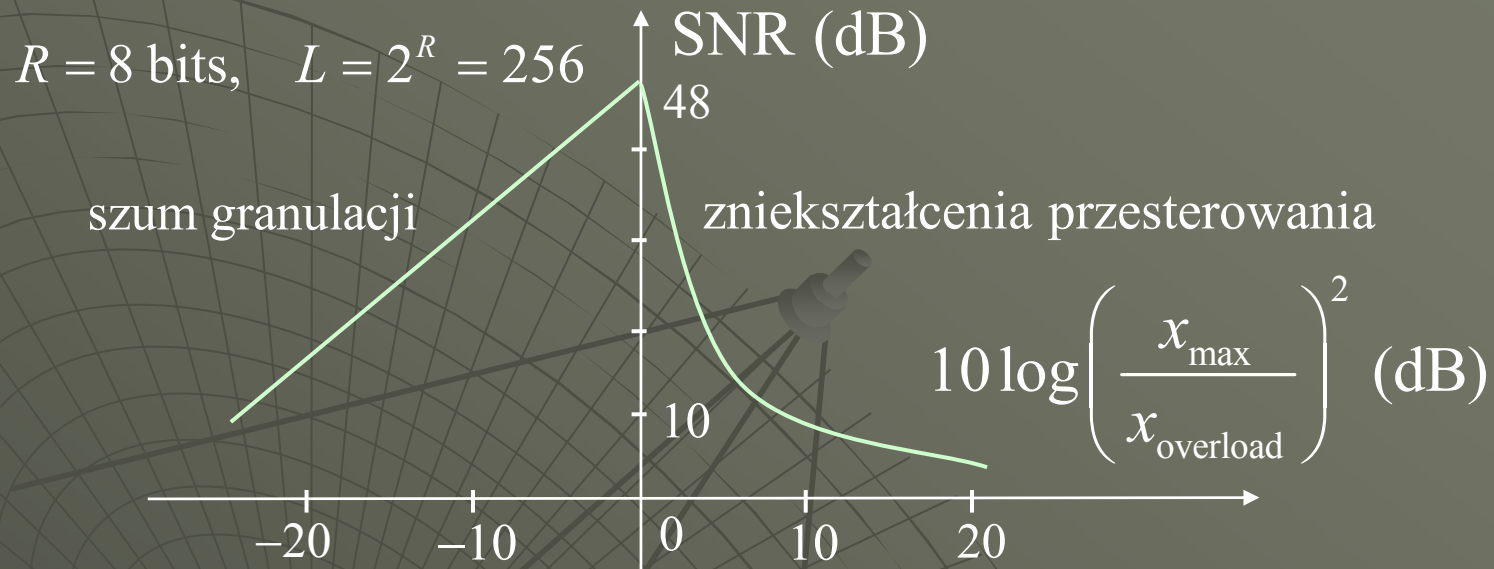
Zastosowania hi-fi: wymagany zakres dynamiki >90 dB (110dB poziomu max. minus ok. 20dB szumu tła, pasmo min. 20kHz, stąd standardy $f_s=44.1$ kHz i 48kHz, reprezentacja $R \geq 16$ bitów.

HD PCM: $f_s=96$ lub 192kHz, $R=24$ bity

Telefonia cyfrowa: wymagany zakres dynamiki ok. 35-40dB, pasmo min. 0.3÷3kHz, stąd $f_s=8$ kHz, długość słowa $R=8$ bitów.

Szybkość transmisji danych PCM przez kanał (*bitrate*):

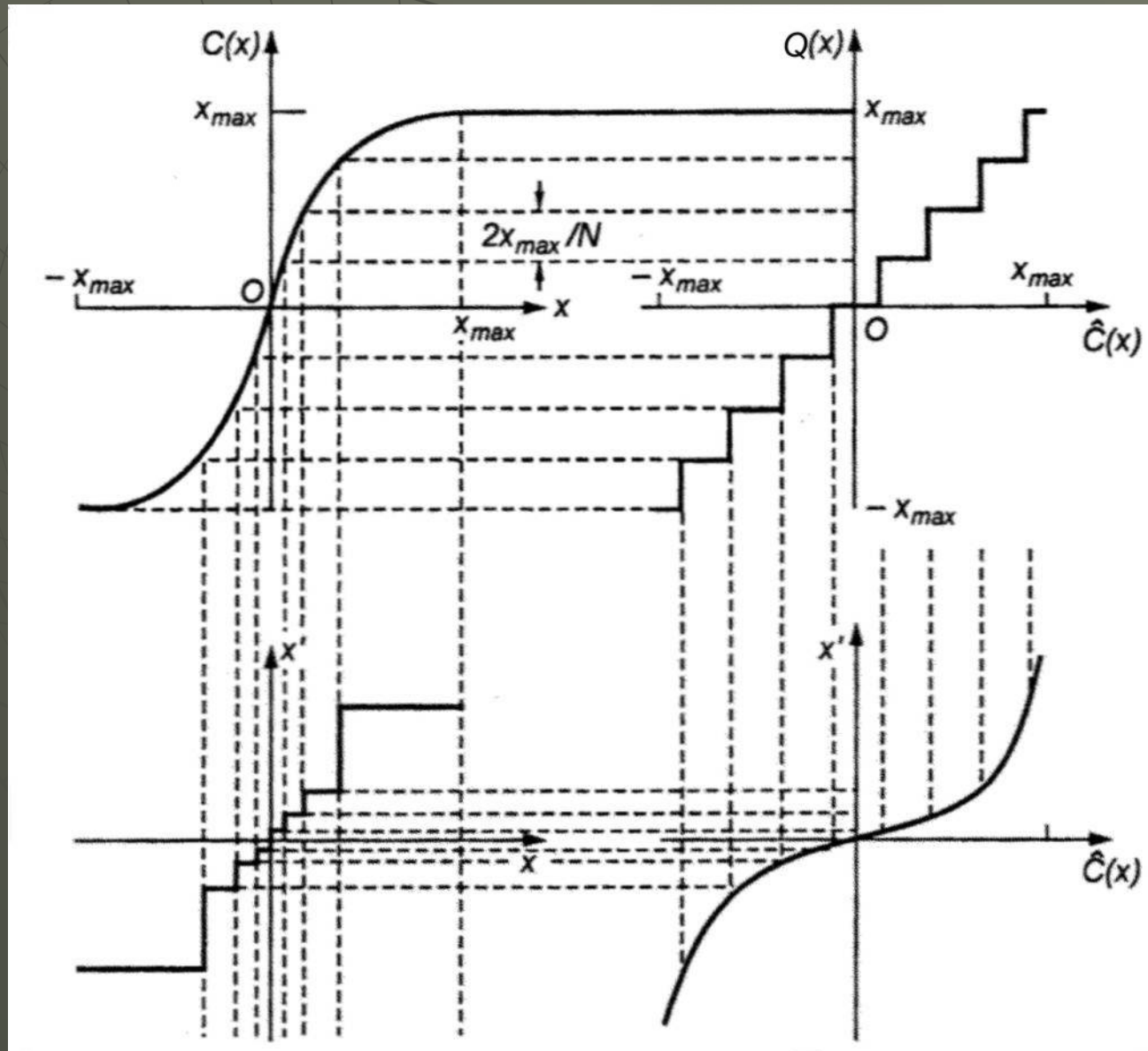
$$I = f_s \cdot R \quad [\text{bits/s}]$$



Rys. Stosunek maksymalnej amplitudy sygnału do błędu kwantowania liniowej PCM w zakresie liniowości i w przypadku przesterowania

Ważną zaletą systemów PCM jest to, że SNR kwantowania rośnie wykładniczo ze wzrostem bitrate'u I (liniowo w skali dB).

Logarytmiczne kodowanie PCM



$$G_C = \left. \frac{dC(x)}{dx} \right|_{x \rightarrow 0}$$

(wzmocnienie określające wzrost rozdzielczości dla małych sygnałów w porównaniu z lin-PCM)

$$\begin{aligned} SNR_{\text{small_signals}} &= \\ &= 6.02R + 20 \cdot \log G_C \end{aligned}$$

Rys. Charakterystyka kompandera $C(x)$ i ekspandera dla kodowania logarytmicznego PCM

log-PCM reguła A (A-law)

Charakterystyka kompresora (liniowa dla $|x| \leq x_{\max}/A$):

$$C(x) = \begin{cases} \frac{A|x|}{1 + \ln A} \operatorname{sgn} x & \text{dla } 0 \leq \frac{|x|}{x_{\max}} \leq \frac{1}{A} \\ x_{\max} \frac{1 + \ln \left(\frac{A|x|}{x_{\max}} \right)}{1 + \ln A} \operatorname{sgn} x & \text{dla } \frac{1}{A} \leq \frac{|x|}{x_{\max}} \leq 1 \end{cases}$$

Standard europejski: $A = 87.56$, $\frac{\max \Delta}{\min \Delta} = A$

$$SNR_{\text{large_signals}} = 6.02R - 9.99 \quad (\text{dB})$$

$$G_C = \frac{A}{1 + \ln A} = 24.08 \quad (\text{dB})$$

Przyrost SNR dla małych sygnałów daje możliwość skrócenia słowa o 4 bity bez istotnej utraty jakości (8 bitów zamiast 12).

log-PCM reguła μ (μ -law)

Charakterystyka kompresora:

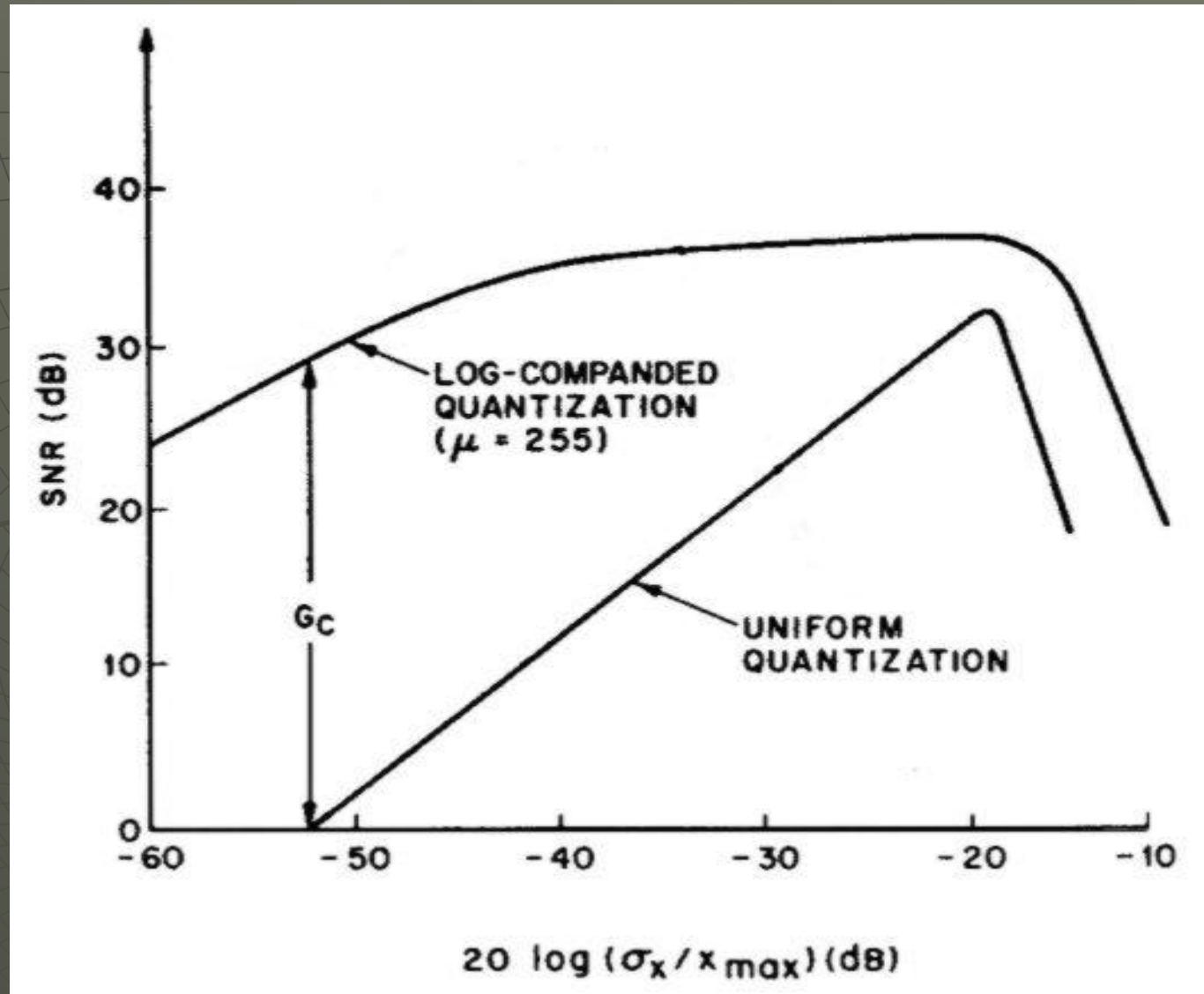
$$C(x) = x_{\max} \frac{\ln \left(1 + \frac{\mu |x|}{x_{\max}} \right)}{\ln(1 + \mu)} \operatorname{sgn} x$$

Standard w USA i Japonii: $\mu = 255$, $\frac{\max \Delta}{\min \Delta} = 1 + \mu$

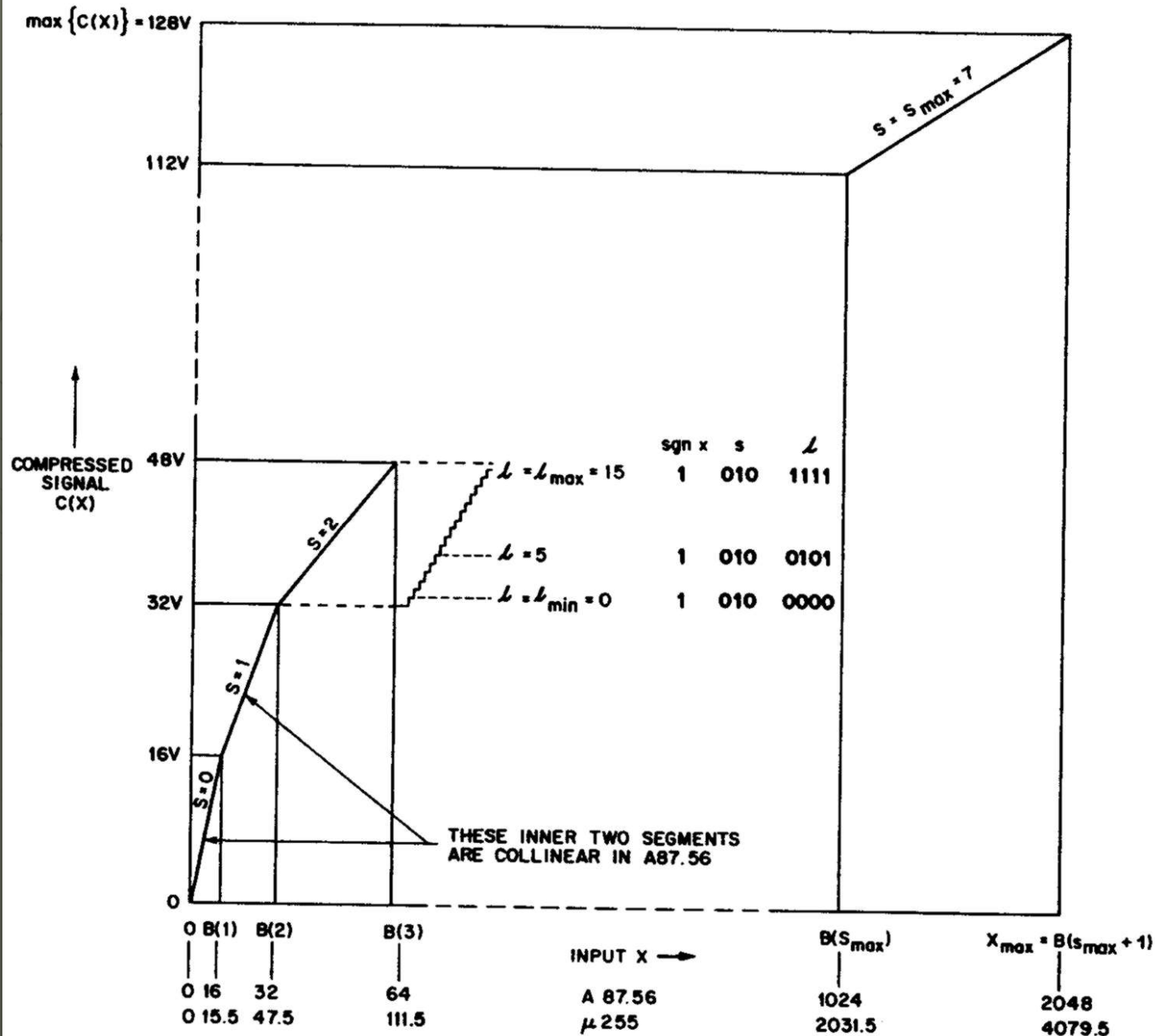
$$SNR_{\text{large_signals}} = 6.02R - 10.1 \quad (\text{dB})$$

$$G_C = \frac{\mu}{\ln(1 + \mu)} = 33.25 \quad (\text{dB})$$

Przyrost SNR dla małych sygnałów daje możliwość skrócenia słowa o 5 bitów bez istotnej utraty jakości (8 bitów zamiast 13).



Rys. SNR dla 8-bitowego logarytmicznego kwantowania PCM μ -law w porównaniu z kwantowaniem liniowym



Rys.
Kawałkami
liniowa
(segmentowa)
implementacja
charakterystyk
i $C(x)$
kompandera
log-PCM (skala
OX dla reguł A
i μ , $R=8$
bitów)

Tabela konwersji dla enkodera wg reguły A (11→7, bez bitu znaku)

Kod wejściowy liniowy	Kod po kompresji
0000000wxyz	000wxyz
0000001wxyz	001wxyz
000001wxyza	010wxyz
00001wxyzab	011wxyz
0001wxyzabc	100wxyz
001wxyzabcd	101wxyz
01wxyzabcde	110wxyz
1wxyzabcdef	111wxyz

Tabela konwersji dla dekodera wg reguły A (7→11)

Kod skompresowany	Kod wyjściowy po dekompresji
000wxyz	0000000wxyz
001wxyz	0000001wxyz
010wxyz	000001wxyz1
011wxyz	00001wxyz10
100wxyz	0001wxyz100
101wxyz	001wxyz1000
110wxyz	01wxyz10000
111wxyz	1wxyz100000

Modulacja delta DM (Delta Modulation)

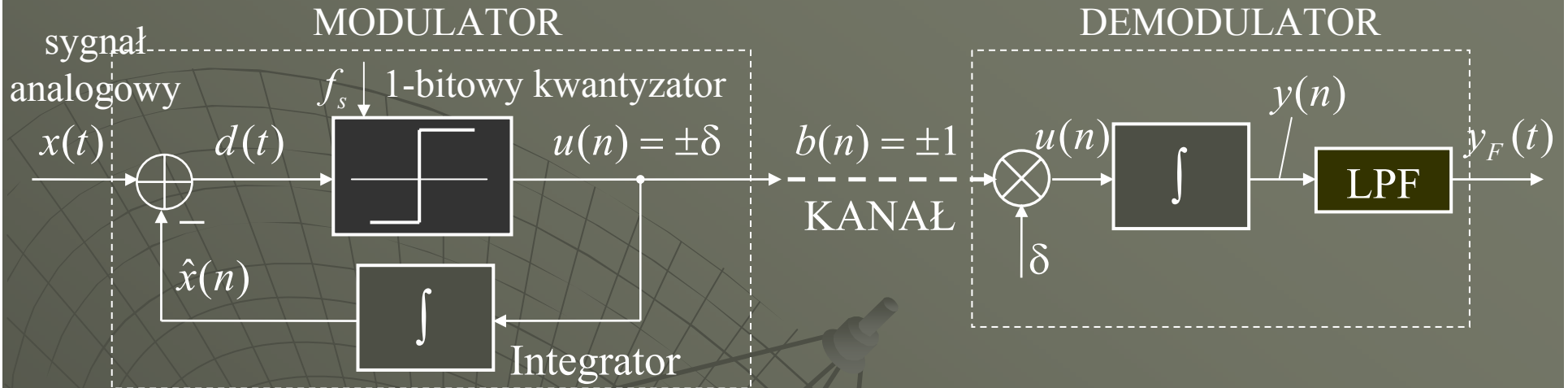
DM polega na 1-bitowym kodowaniu różnicy między kolejnymi próbkami sygnału. Jest przedstawicielem modulacji PDM (*Pulse Density Modulation*).

DM daje dobre wyniki w kodowaniu sygnałów o dużej autokorelacji sąsiednich próbek próbek ($R_{xx}(1) \rightarrow 1$), jak mowa. Stosowana do digitalizacji mowy z jakością telekomunikacyjną.

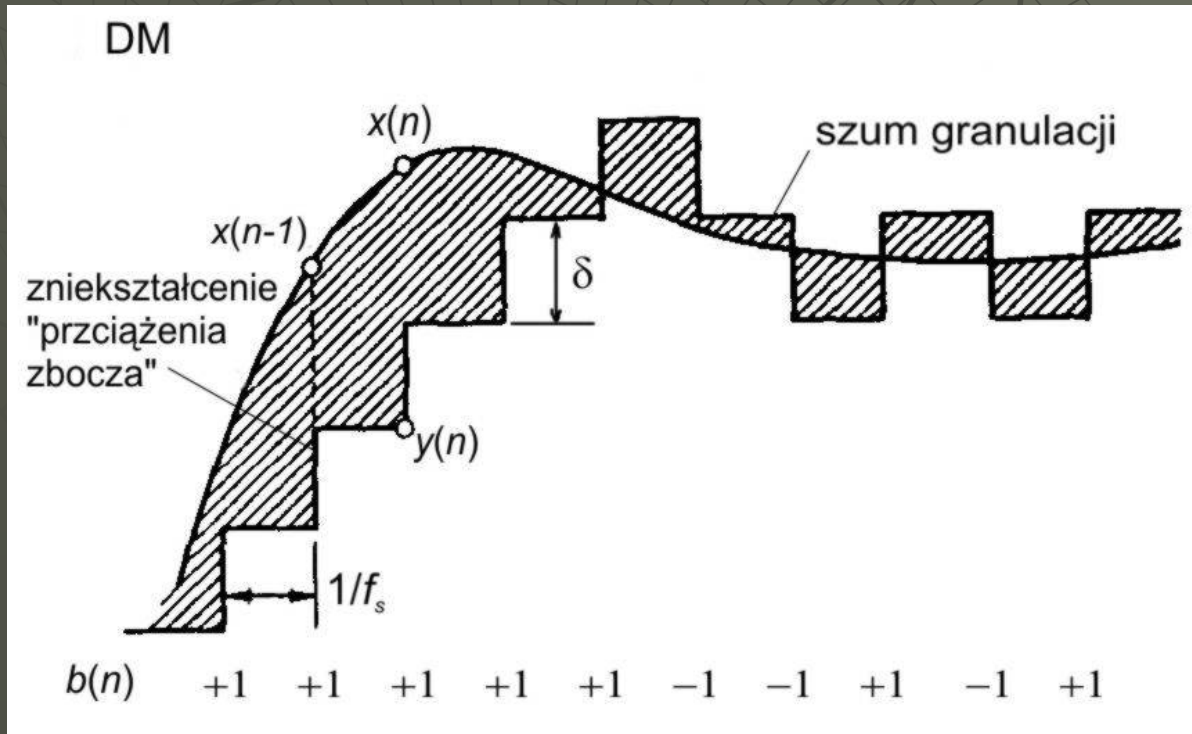
Zalety: prosta realizacja układowa: strumień bitów bez *word-framing*, 1-bitowy ADC, niepotrzebny układ S/H (*Sample & Hold*) ani filtr AA.

Wady: SNR kwantowania rośnie tylko liniowo ze wzrostem I (2-krotny wzrost I daje wzrost SNR o 9 dB: 6dB z 2-krotnego zmniejszenia przyrostu δ + 3dB z podwojenia szerokości pasma szumu). Konieczność wysokiego bitrate $I=1 \cdot f_s$, czyli dużego *oversamplingu* (64x i więcej),

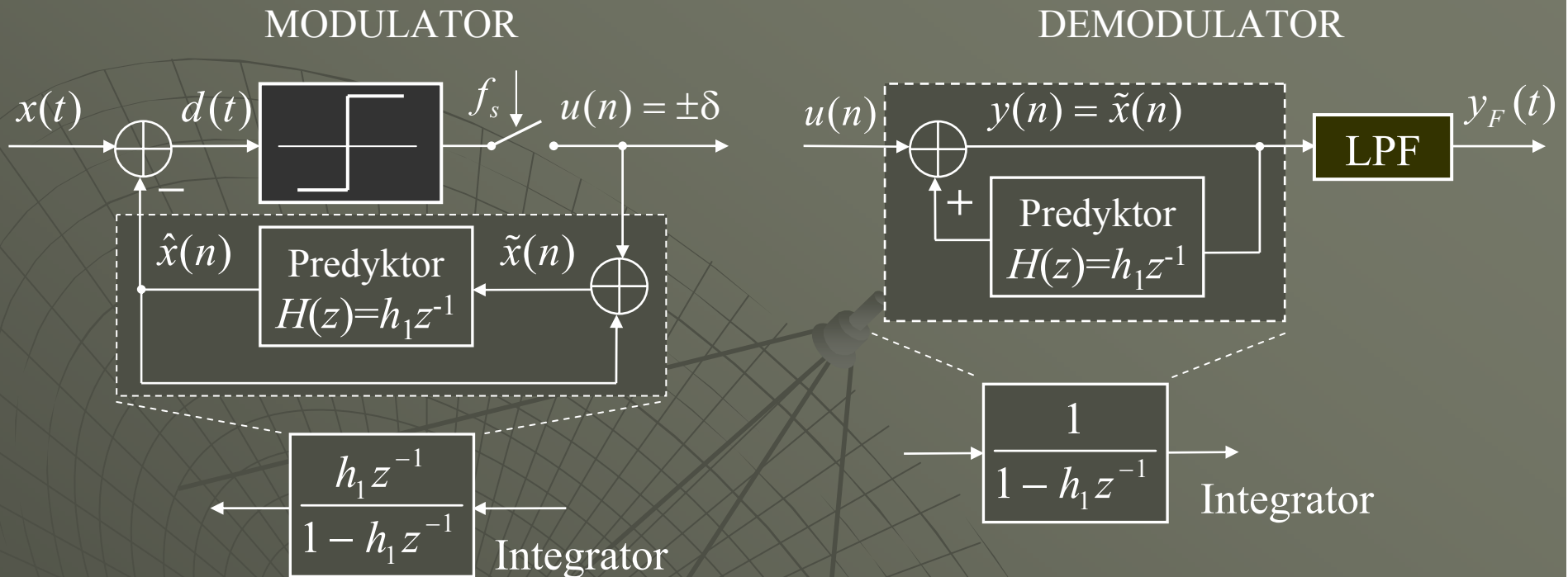
Do przesyłania strumień bitowy DM przekodowuje się często do postaci PCM (*word-framing*).



Rys. Schemat blokowy modulatora i demodulatora DM



Rys. Przebiegi sygnału wejściowego $x(t)$ i sygnału $y(n)$ w demodulatorze DM. W praktyce bity $b(n)=0$ lub 1 . Dokładność kodowania zależy od: 1) wielkości przyrostu δ , 2) częstotliwości próbkowania f_s



Równania modulatora i demodulatora w chwilach próbkowania $t=nT_s$ i założeniu braku przekłamań bitów w kanale ($u(n)$ w modulatorze i demodulatorze są takie same, wtedy $y(n) = \tilde{x}(n)$):

$$d(n) = x(n) - \hat{x}(n), \quad u(n) = \delta \cdot \text{sgn } d(n), \quad \tilde{x}(n) = \hat{x}(n) + u(n)$$

$$\hat{x}(n) = h_1 \tilde{x}(n-1) = h_1 \hat{x}(n-1) + h_1 u(n-1)$$

$$y(n) = h_1 y(n-1) + u(n), \quad u(n) = \delta \cdot b(n)$$

$h_1=1$ – idealny integrator, efekty potencjalnych przekłamań bitów w kanale kumulują się, predykcja = opóźnione wyjście:

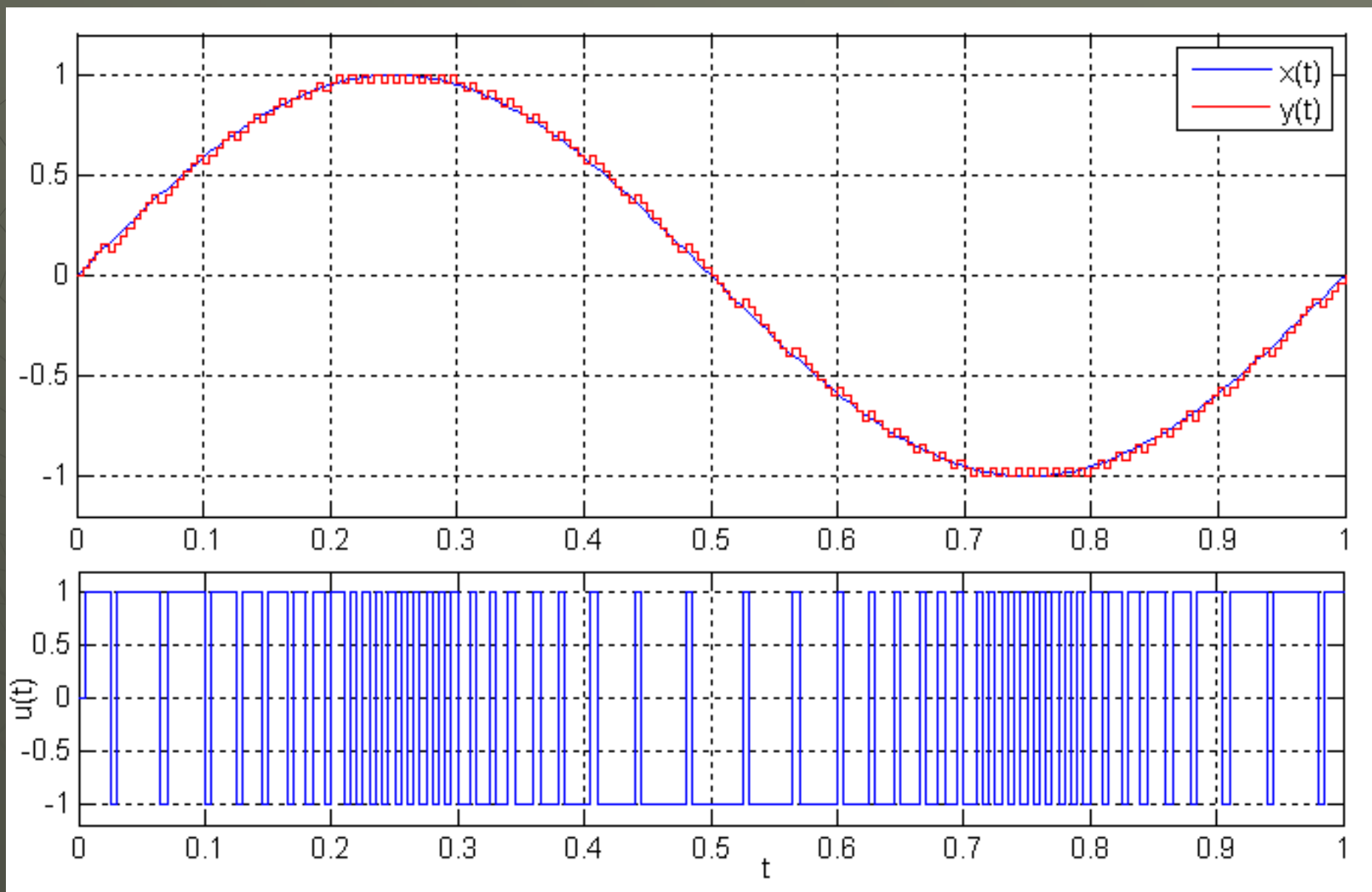
$$\hat{x}(n) = \tilde{x}(n-1) = y(n-1)$$

$h_1 < 1$ – integrator z przeciekiem (*leaky integrator*), odporniejszy na błędy transmisji, często przyjmuje się $h_1 = R_{xx}(1)/\sigma_x^2$ (bliskie 1)

W DM z podwójnym całkowaniem transmitancja predyktora

$$H(z) = h_1 z^{-1} + h_2 z^{-2}$$

($h_1=0 \rightarrow$ predyktor 2-krokowy). DM-DI daje lepszy przyrost SNR $\sim f_s^{2.5}$, ale mogą wystąpić problemy ze stabilnością.

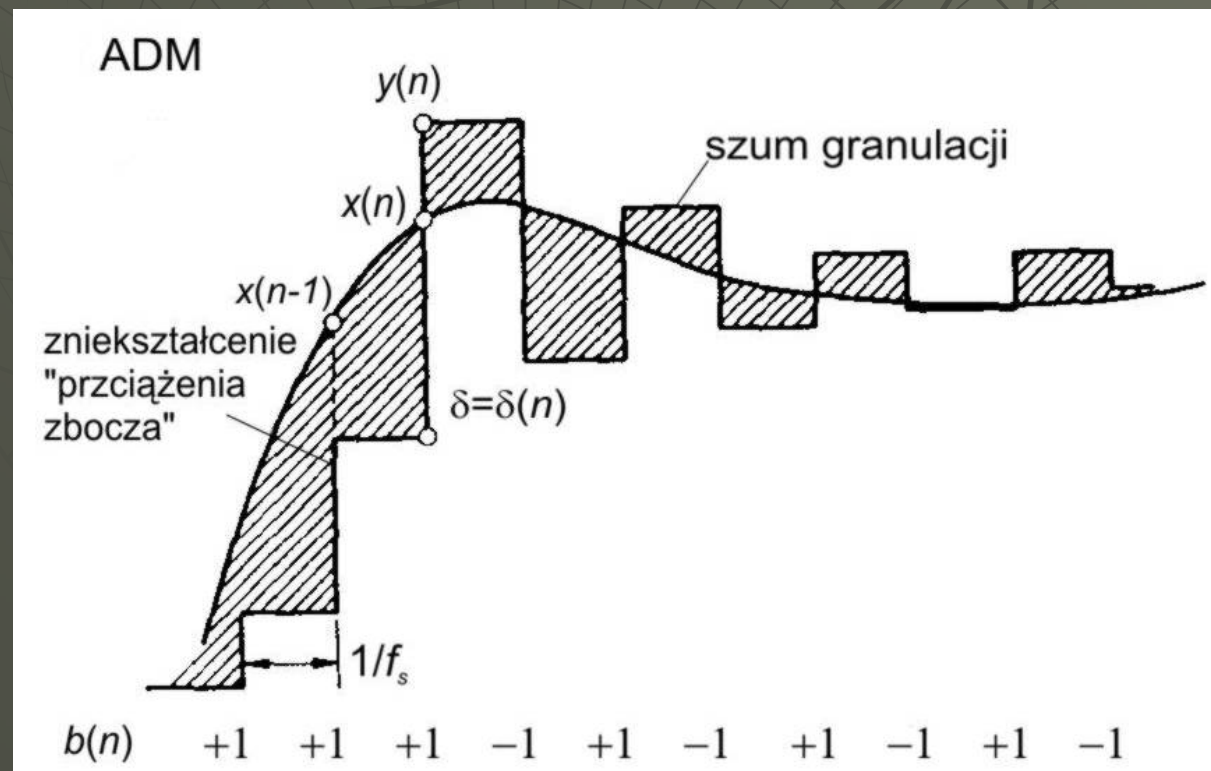


Rys. Modulacja DM sygnału sinusoidalnego ($f_s=200\text{Hz}$, $\delta=0.04$)

Adaptacyjna modulacja delta ADM (Adaptive DM)

Możliwość adaptacyjnej zmiany przyrostu $\delta = \delta(n)$ umożliwia wzrost SNR bez zwiększania I . Zależy to od dopuszczalnego zakresu wartości $\delta(n)$ i algorytmu zmieniania przyrostu.

Aby demodulator mógł odtworzyć sygnał, algorytm adaptacji musi opierać się *na informacji z ciągu bitów $b(n)$, a nie na sygnale wejściowym $x(n)$* (AQB – Adaptive Quantization Backward).



Rys. Przebiegi sygnału wejściowego $x(t)$ i sygnału $y(n)$ w demodulatorze ADM. W praktyce bity $b(n) = 0/1$

Prosty sposób adaptacji:

- 2 ostatnie bity jednakowe $b(n)=b(n-1) \rightarrow \delta$ jest zwiększane

p razy: $\delta(n)=p\delta(n-1)$,

- 2 ostatnie bity różne $b(n)\neq b(n-1) \rightarrow \delta$ jest zmniejszane

q razy: $\delta(n)=q\delta(n-1)$.

Warunek stabilności: $p \cdot q \leq 1$.

Wartości optymalne dla mowy: $p=1.5$, $q=1/1.5$ (wartości $p=2$, $q=1/2$ są wygodne w realizacji bez istotnego pogorszenia jakości).

Standard *High Information DM (2-bit memory)*: $\delta(n)=M(n)\delta(n-1)$

$$M(n) = \begin{cases} 2 & \text{jeśli } b(n) = b(n-1) = b(n-2) \\ 0.5 & \text{jeśli } b(n) \neq b(n-1) \\ 1 & \text{w pozostałych przypadkach} \end{cases}$$

Ogólny algorytm ADM-AQ z pamięcią m -bitową:

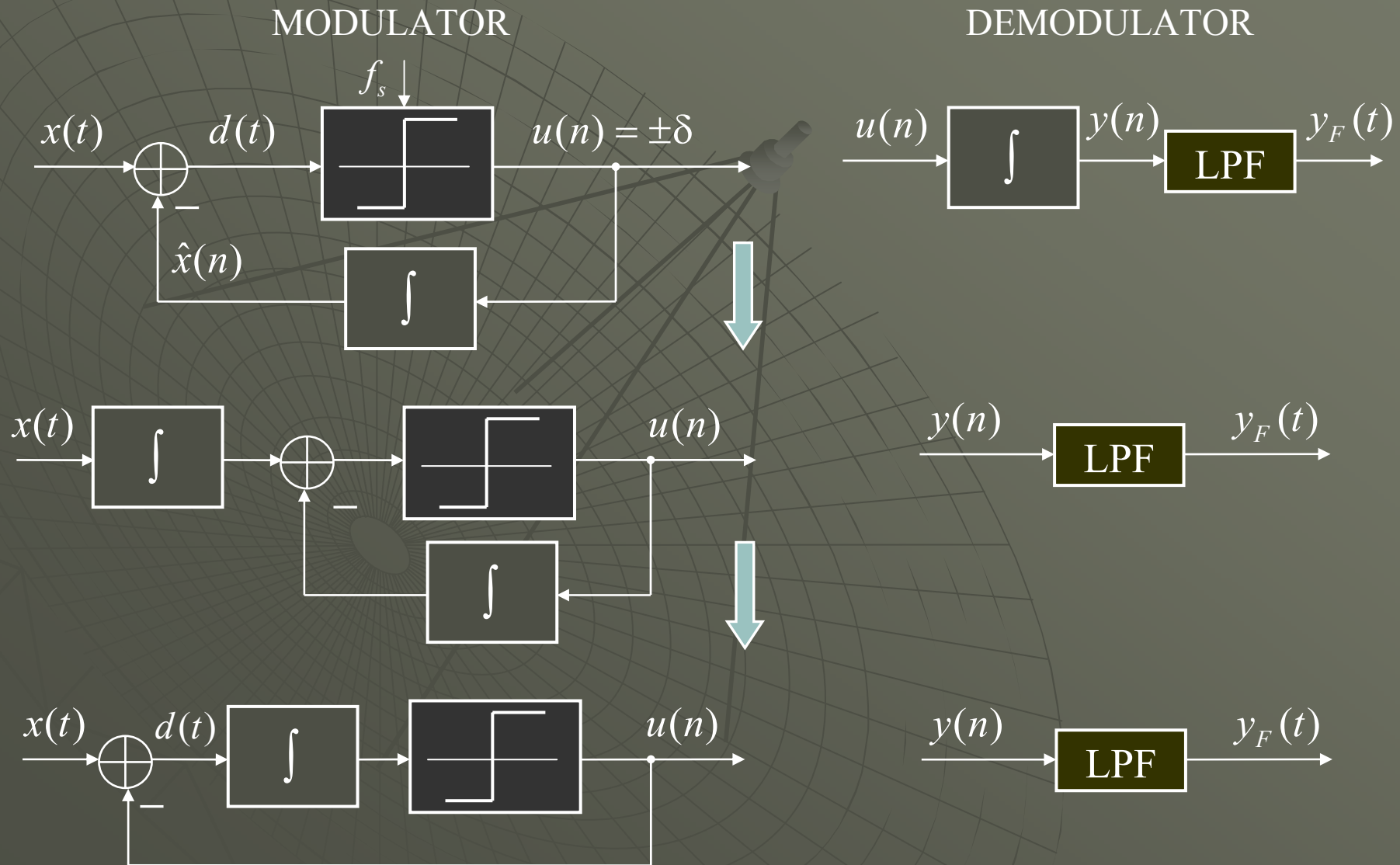
$$\delta(n) = M(n) \cdot \delta(n)$$

$$M(n) = f[b(n), b(n-1), \dots, b(n-m)]$$

Największą wagę ma ostatni bit $b(n)$. Typowy zakres $\delta_{max}/\delta_{min}=128$.

Modulacja Sigma-Delta $\Sigma-\Delta$

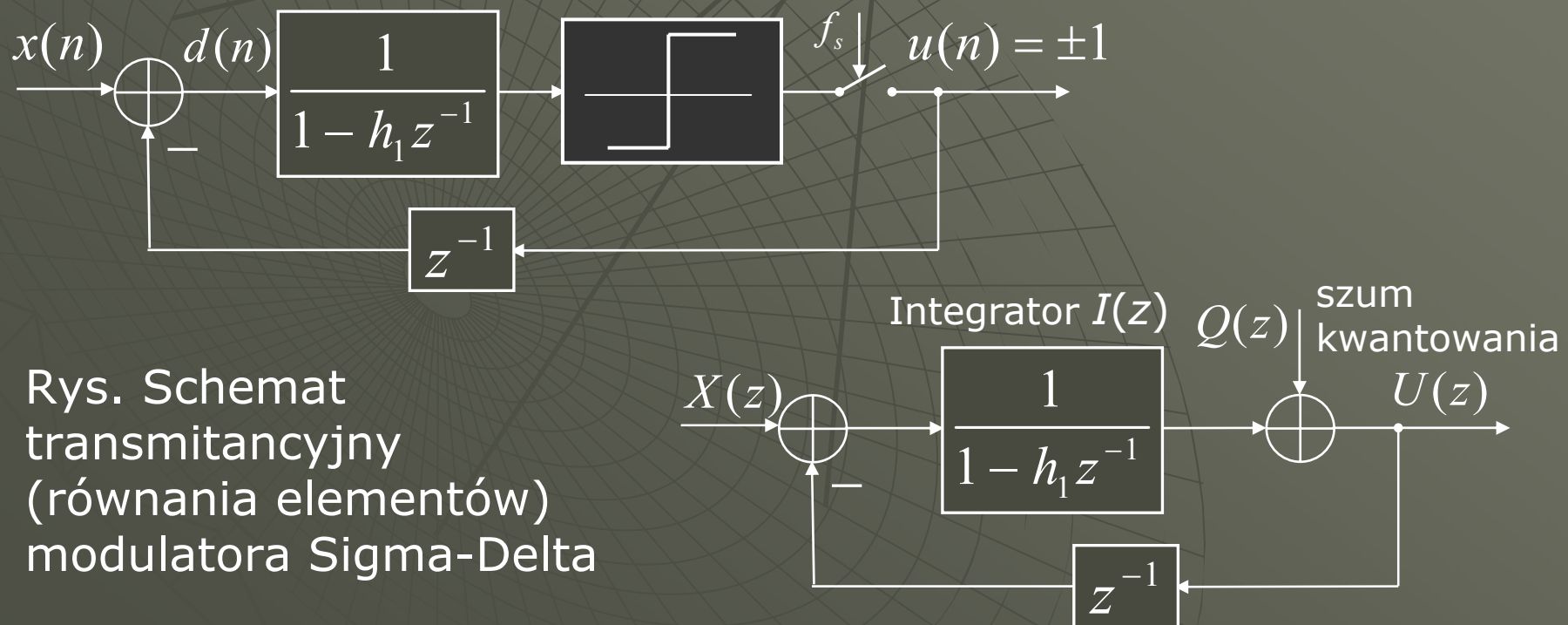
Jest to adaptacja DM do kodowania sygnałów o słabszej autokorelacji.



Całkowanie sygnału wejściowego *przed modulacją DM* (preemfaza niskich częstotliwości) zwiększa autokorelację $R_{xx}(1)$.

Modulacja Σ - Δ redukuje demodulator do uśredniającego filtra LP.

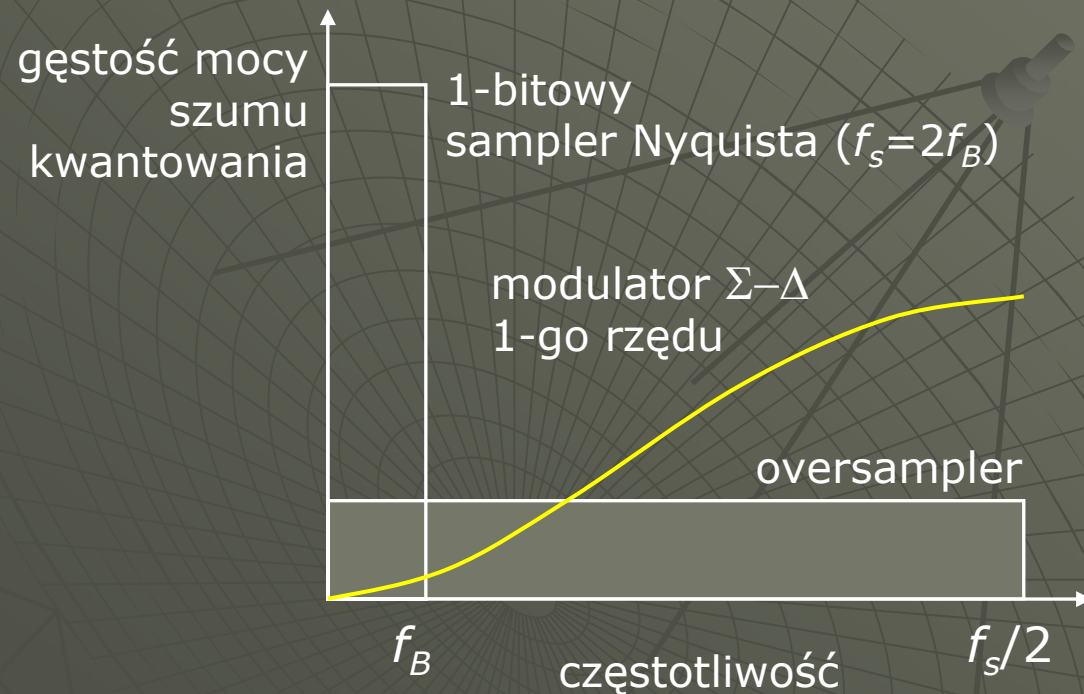
Ważne: Przy modulacji Σ - Δ charakterystyka widmowa błędu przeciążenia zbrocza (*overload distortion*) jest *niezależna od częstotliwości*. Właściwość ta sprawia, że modulator bardzo dobrze nadaje się do przetwarzania A/C.



Rys. Schemat transmitancyjny (równania elementów) modulatora Sigma-Delta

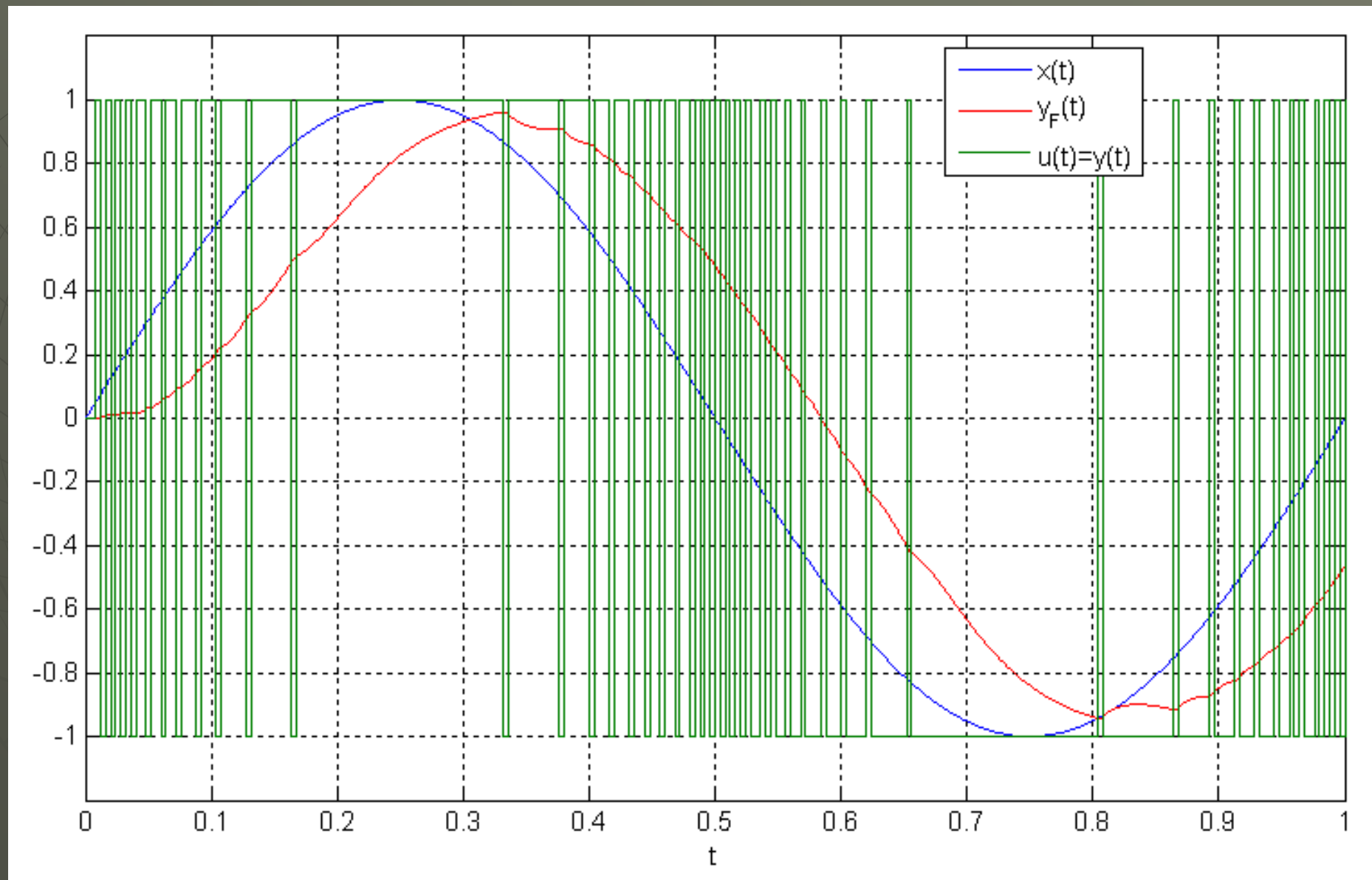
Równanie przetwarzania sygnału wejściowego o transformacie $X(z)$ i szumu kwantowania (granulacji) $Q(z)$ przez modulator $\Sigma-\Delta$ pierwszego rzędu dla idealnego integratora ($h_1=1$):

$$Y(z) = X(z) + (1 - z^{-1})Q(z)$$



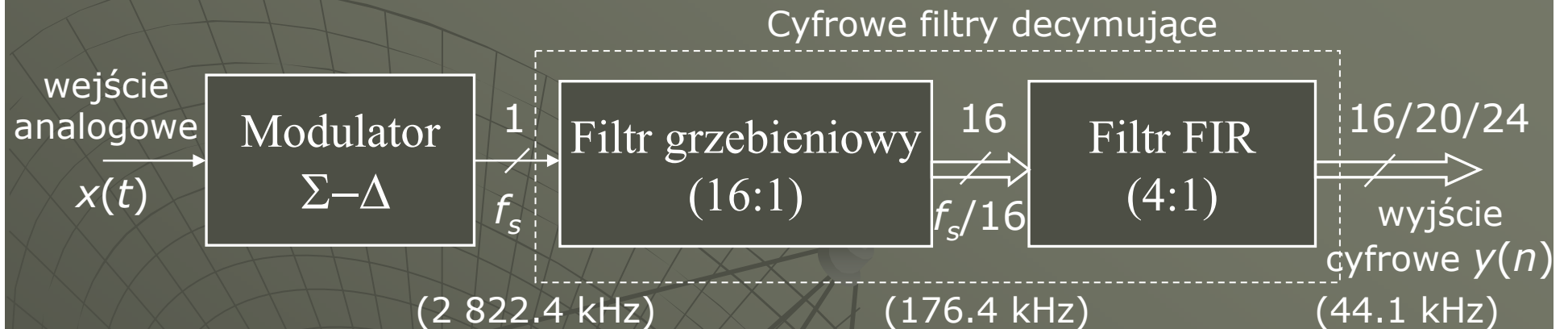
Rys. Kształtowanie szumu kwantowania (tylko szumu granulacji) przez modulator $\Sigma-\Delta$ pierwszego rzędu. f_B – pasmo częstotliwości sygnału (zakres widma sygnału)

Uzyskanie SNR kwantowania powyżej 90 dB jest możliwe przez zastosowanie modulatora $\Sigma-\Delta$ wyższego rzędu który ma jeszcze mniejszą gęstość mocy szumu w paśmie do f_B (łączenie modulatorów 1-go rzędu, powoduje mnożenie ch-ki gęstości mocy szumu).



Rys. Modulacja i demodulacja Σ - Δ sygnału sinusoidalnego, filtr LP 2-go rzędu

Przetwarzanie A/C z modulacją $\Sigma-\Delta$



Rys. Przetwornik A/C $\Sigma-\Delta$ z dwustopniową decymacją (częstotliwości próbkowania przykładowe, oversampling 64x)

Wartość sygnału wyjściowego po każdym okresie decymacji (o czynnik D) i bez kwantowania jest równa średniej D bitów wychodzących z modulatora $\Sigma-\Delta$:

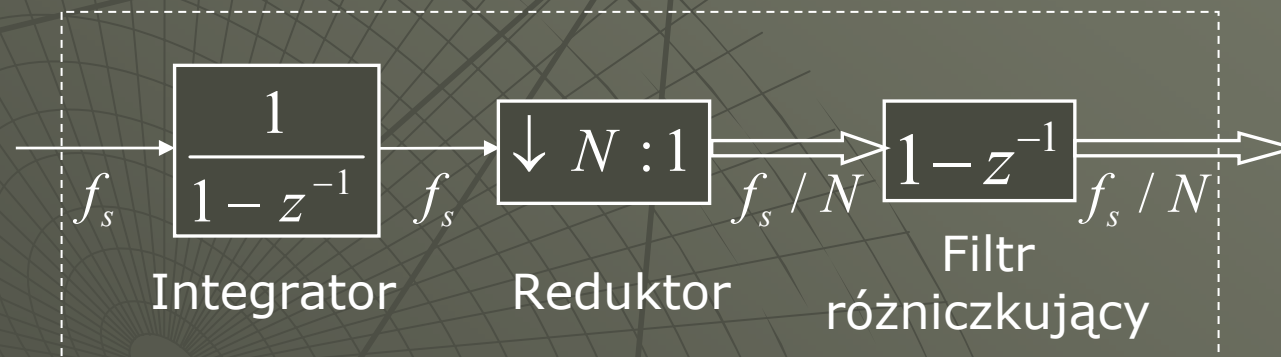
$$y(n = mD) = \frac{1}{N} \sum_{k=0}^{D-1} u(n - k)$$

Przykład: Ciąg 16 bitów: 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0.
Po decymacji 16:1 wyjście (średnia bitów) = $7/16 = 0111$ binarnie (przy rozdzielczości $R=4$ -bitowej).

Transmitancja i równanie różnicowe filtra grzebieniowego o długości N (rzędu $N-1$):

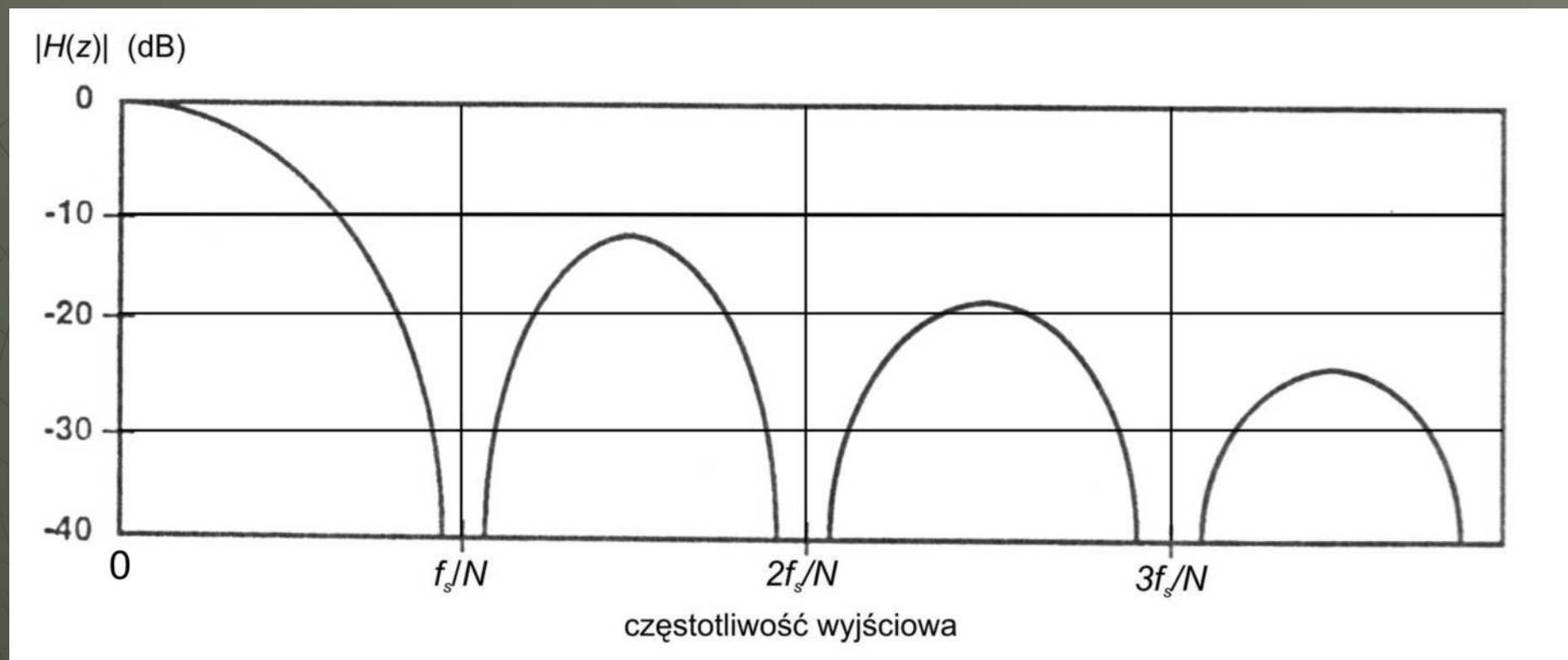
$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{N-1} z^{-k} = 1 + z^{-1} + \dots + z^{-(N-1)} = \frac{1}{1 - z^{-1}} (1 - z^{-N})$$

$$y(n) = \sum_{k=0}^{N-1} x(n-k) = x(n) + x(n-1) + \dots + x(n-N+1)$$



Rys. Struktura jednostopniowego filtra grzebieniowego (*comb filter*) z decymacją

Filtr grzebieniowy *nie wymaga układu mnożącego* (współczynniki równe 1). *Może pracować z dużą częstotliwością*. Całkowanie na wysokiej f_s redukuje obszar pamięci do przechowywania wyników pośrednich.

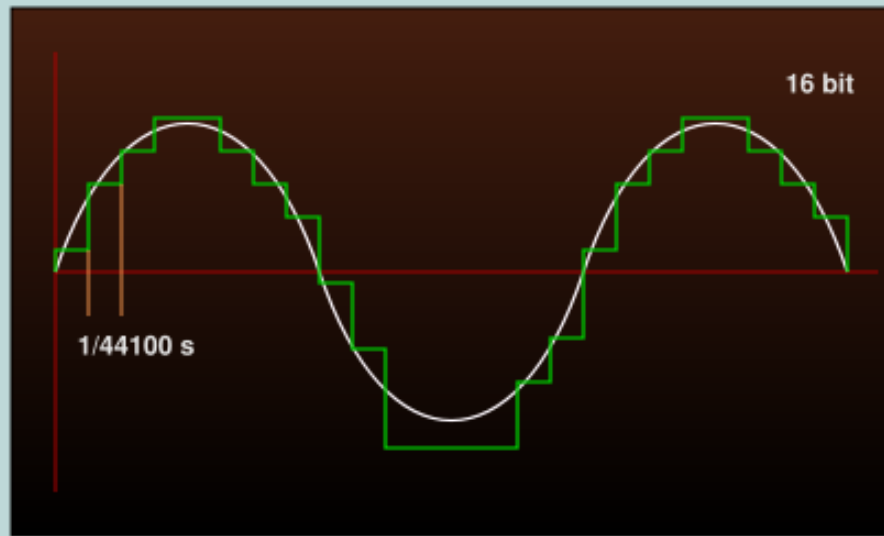


Rys. Charakterystyka amplitudowa jednostopniowego filtra grzebieniowego.

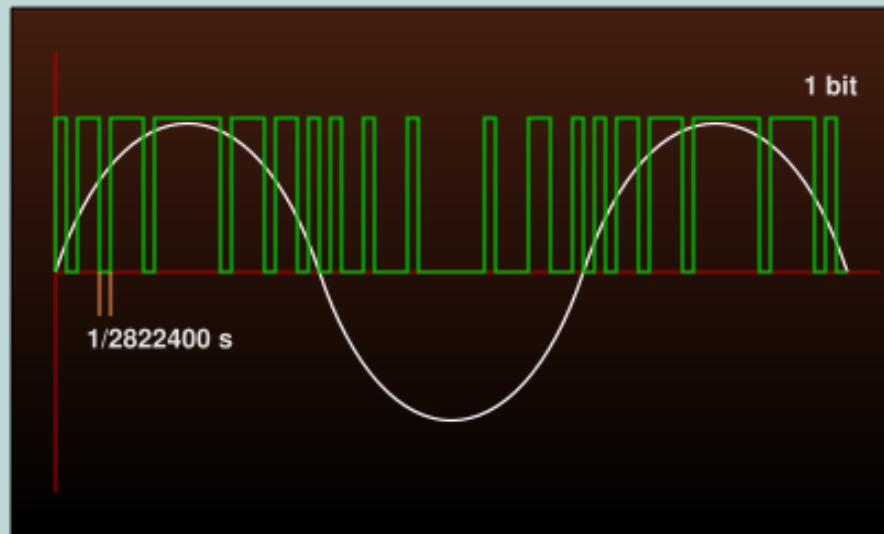
Kaskadowe łączenie filtrów grzebieniowych obniża boczne listki wypadkowej charakterystyki amplitudowej (zwiększa tłumienie powyżej f_s/N).

Filtr FIR typu HP jest niezbędny do skompensowania nierówności charakterystyki filtra grzebieniowego.

PCM



DSD

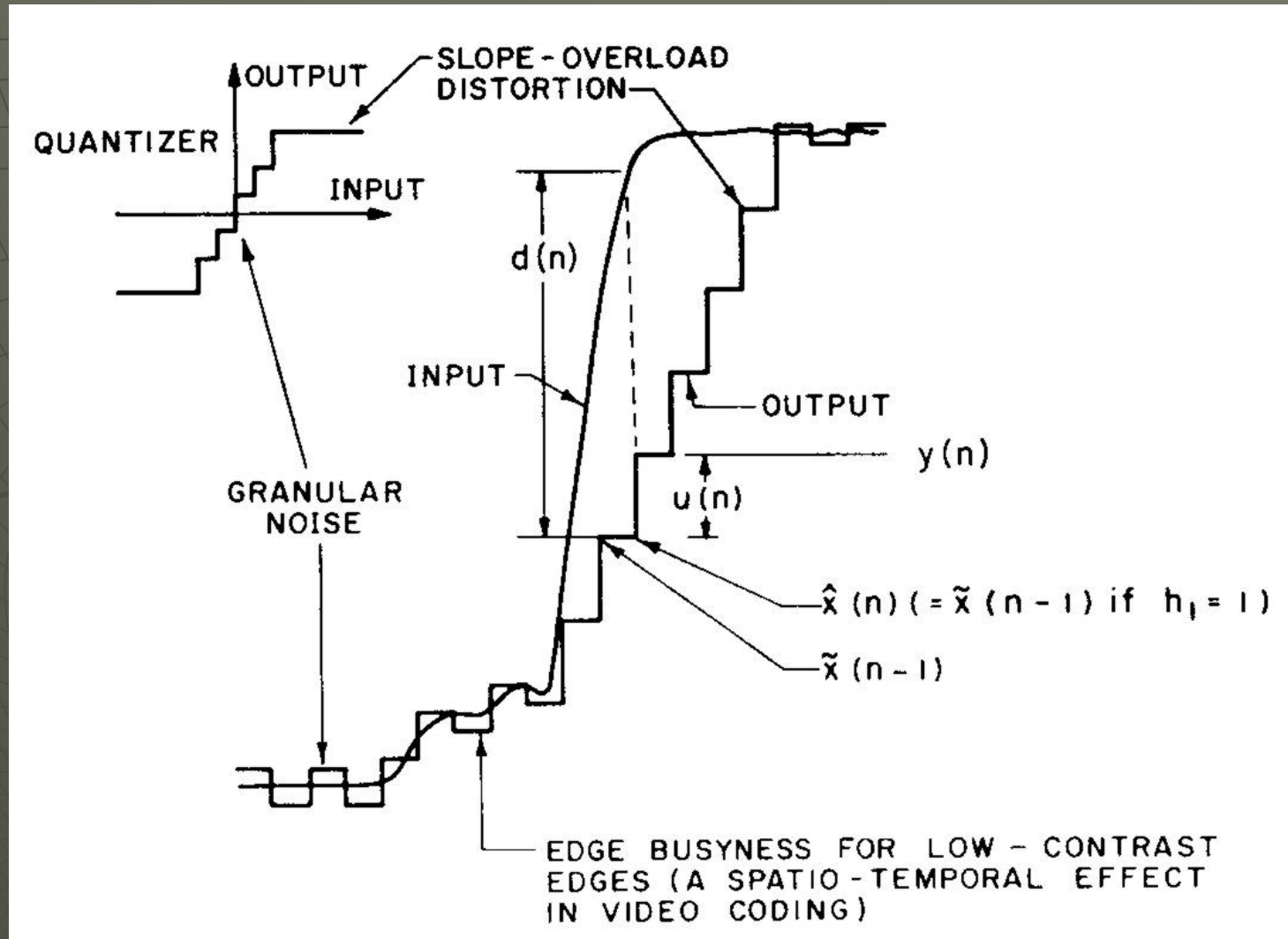


DSD (*Direct Stream Digital*) - system modulacji PDM stworzony przez firmy Sony i Philips, używany do kodowania dźwięku cyfrowego, np. na dyskach SACD (*Super Audio Compact Disc*).

Różnicowa modulacja PCM - DPCM (Differential PCM)

Kodowanie DPCM łączy cechy różnicowe DM z wielobitową reprezentacją PCM. Reprezentacja różnicy między kolejnymi próbkami sygnału jest kilkubitowa ($R=3$ lub 4), co daje większą rozróżnialność i umożliwia zmniejszenie f_s przy tej samej szybkości przesyłania I .

DPCM przypomina ADM o zmiennym przyroście, ale większa ilość informacji jest dostępna w czasie jednego okresu próbkowania, co jest wykorzystywane w algorytmie (różnica $d(n) = x(n) - \hat{x}(n)$ ma kilka bitów). Koder DPCM przesuwą środek zakresu kwantowania do predykcji $\hat{x}(n)$, co umożliwia precyzyjniejsze kwantowanie wokół tej wartości. Predyktor – filtr FIR o N współczynnikach – jest bardziej złożony niż w DM.



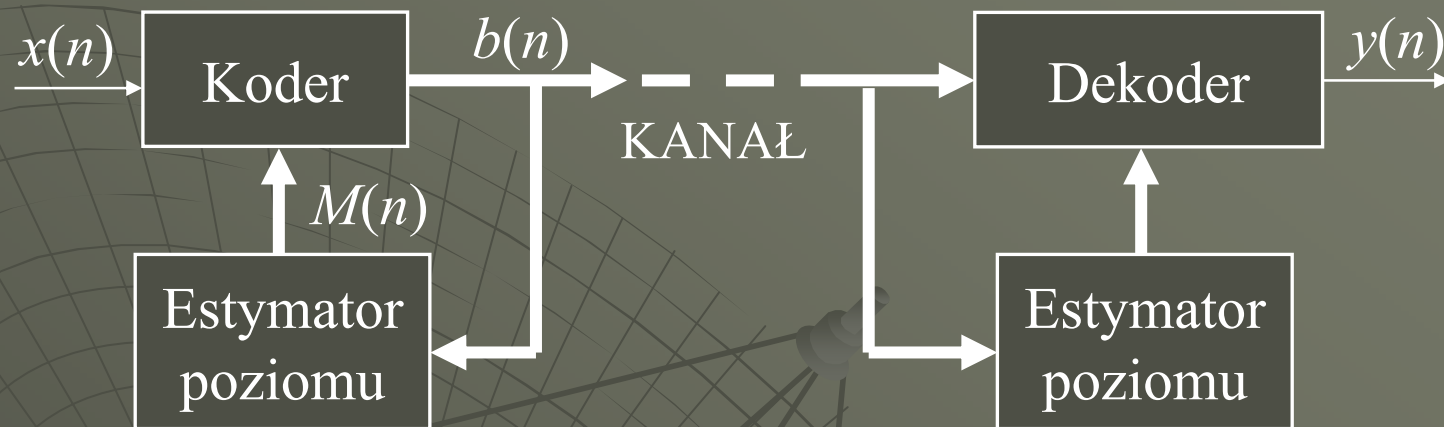
Rys. Ilustracja kodowania DPCM sygnału

Adaptacyjna różnicowa modulacja PCM - ADPCM

Adaptacja w układzie DPCM może polegać na:

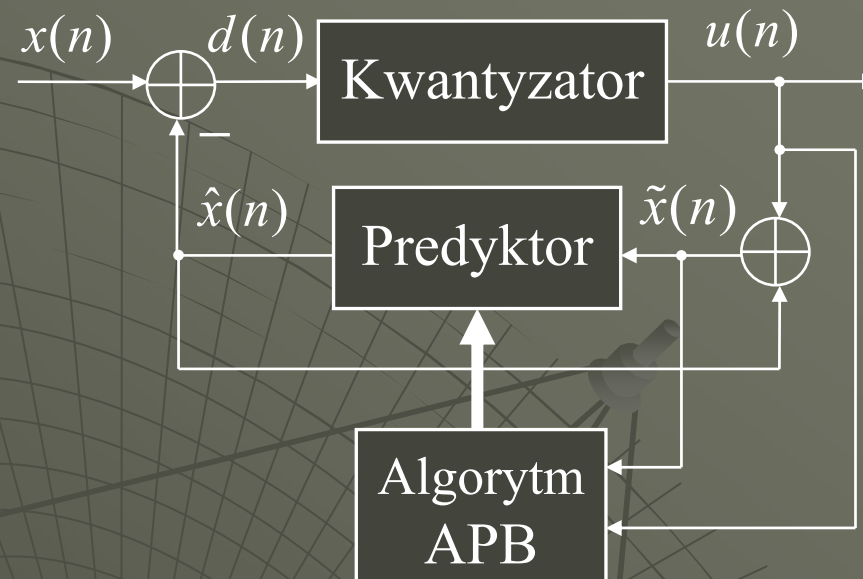
- 1) adaptacji przyrostu kwantowania $\delta(n)$ (AQ),
- 2) adaptacji filtra predykcji $H(z)$ (AP)

Jakość kodowania zależy nie tylko od SNR, ale także od widma błędu rekonstrukcji $r(n)=y(n)-x(n)$. W układach DPCM-AQ widmo to jest typu LP, co jest zaletą w stosunku do białego widma PCM (przy takim samym SNR). Subiektywnie ADPCM daje poprawę 2.5 bita w stosunku do log-PCM, z punktu widzenia energii błędu kwantowania 1.5 bita.



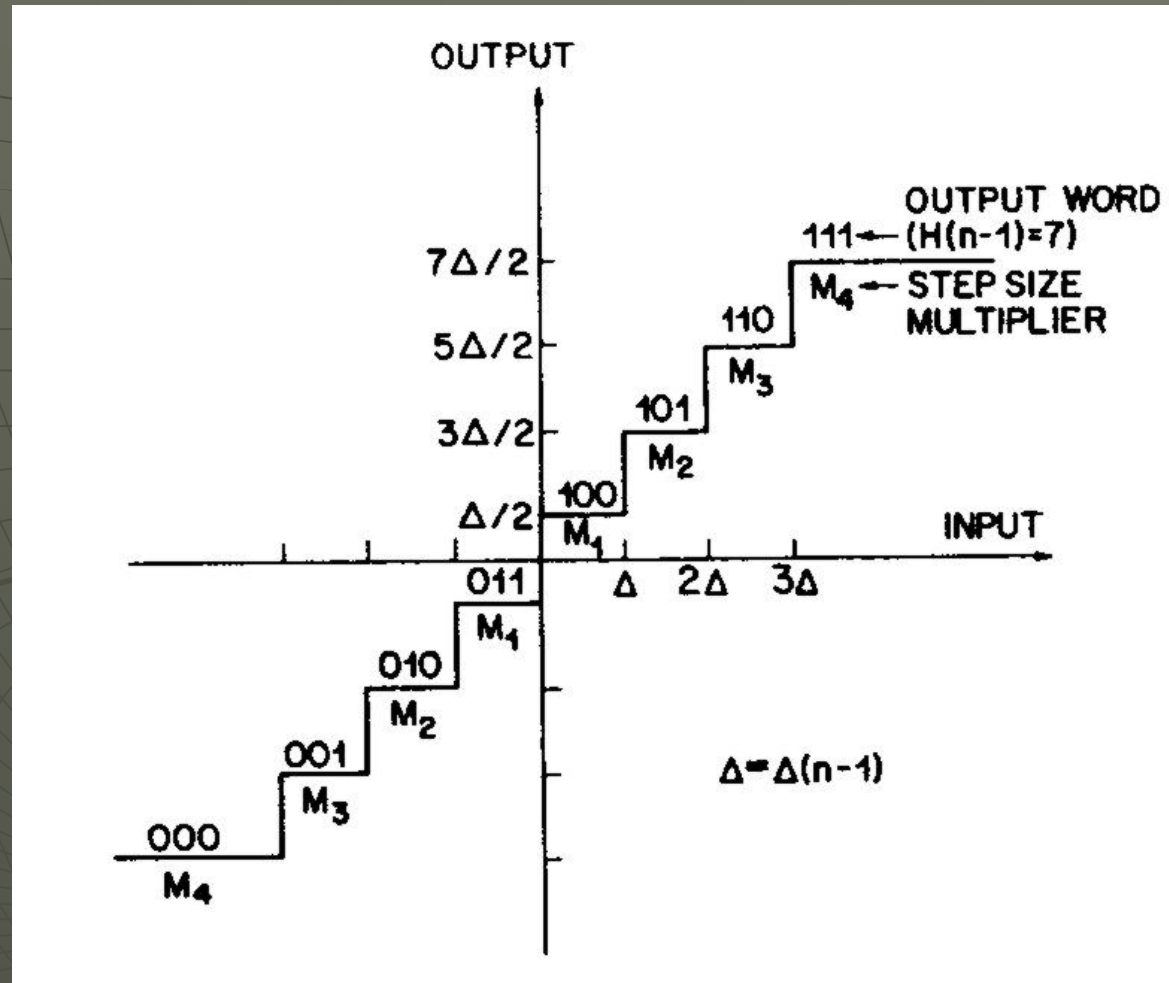
Rys. Schemat blokowy AQB (*Adaptive Quantization Backward*).

Adaptacja kroku kwantowania $\delta(n)$ jest przeprowadzana na podstawie wartości bitów $b(n)$, a nie wartości rzeczywistych $x(n)$, żeby dekodery mógł zrekonstruować sygnał (najczęściej stosowana metoda). Estymator poziomu oblicza wariancję skwantowanego sygnału ($M(n)$ – mnożnik kroku kwantowania)



Rys. Schemat predykcji APB (*Adaptive Prediction Backward*).
 Predyktor $H(z)=1/A(z)$ jest filtrem adaptacyjnym *all-pole*.
 Minimalizuje się wariancję błędu predykcji $d(n)$ w funkcji
 parametrów \mathbf{h} predyktora (wektor o długości 4 do 6).

Metody adaptacji kroku kwantowania i filtra predykcji opierające się na rzeczywistych wartościach sygnału $x(n)$ są nazywane metodami *forward*: AQF i APF. Dokładność tych metod jest większa, ale wymagają buforowania próbek i przesyłania dodatkowych informacji do dekodera, dlatego są stosowane rzadziej.



Rys. Adaptacyjne 3-bitowe kwantowanie AQB z jednym słowem pamięci. Rysunek pokazuje charakterystykę kwantyzatora $Q(x)$ w chwili $(n-1)$ [Jayant, 1973]

$$\delta(n) = M[H(n-1)] \cdot \delta(n-1) \quad (\text{oznaczenie kroku na rys. } \Delta = \delta)$$

Koder	PCM			DPCM		
	R	2	3	4	2	3
M_1	0.60	0.85	0.80	0.80	0.90	0.90
M_2	2.20	1.00	0.80	1.60	0.90	0.90
M_3		1.00	0.80		1.25	0.90
M_4		1.50	0.80		1.75	0.90
M_5			1.20			1.20
M_6			1.60			1.60
M_7			2.00			2.00
M_8			2.40			2.40

Tabela. Mnożniki kroku kwantowania adaptacji AQB z jednostajnym kwantyzatorem dla sygnału mowy.

Komercyjny system DPCM-APB-AQB dla mowy (specyfikacja CCITT'83 G.721, stosowany m.in. w modemach), $R=4$ bity, $I=32$ kb/s, $SNR \geq 30$ dB w szerokim zakresie częstotliwości, stabilna praca przy bitowym współczynniku błędu rzędu 10^{-3} .