

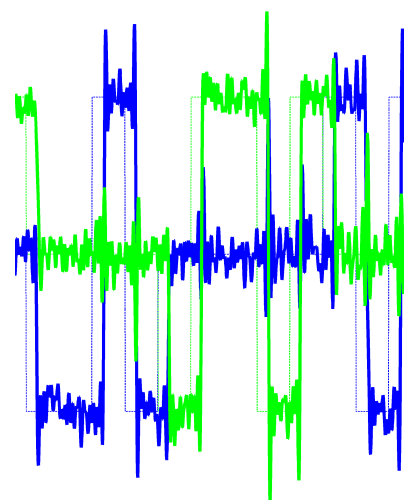
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická

PROJEKT Č. 2

Komplexní reprezentace pásmových signálů

Vypracoval:	Jan HLÍDEK
V rámci předmětu:	Signály a soustavy (X37SGS)
Zpracováno do:	6. 12. 2007
Cvičení:	čtvrtek sudý od 14:30 do 16:00



1. ZADÁNÍ

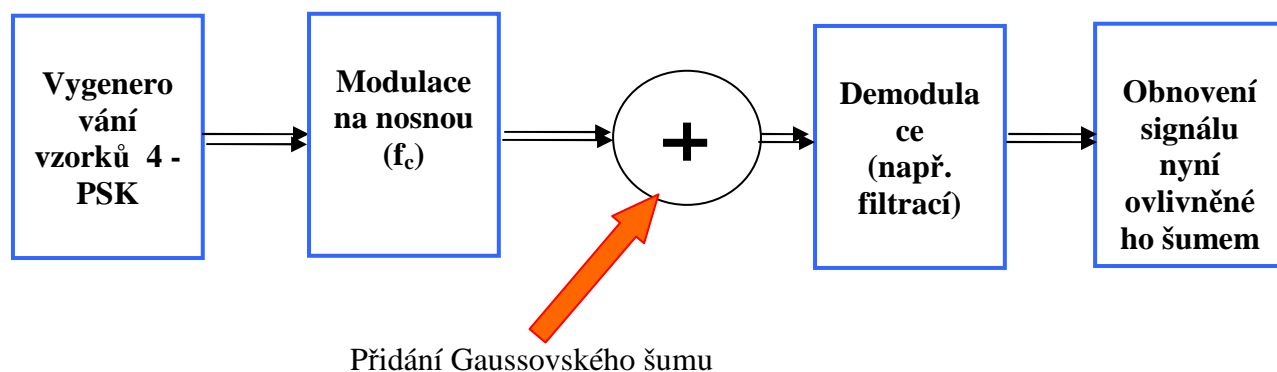
- vygenerujte vzorky komplexní obálky M -stavové PSK digitální modulace $\tilde{s}[k]$ se vzorkovacím kmitočtem f_{sa} a symbolovou rychlostí f_d ; modulační impuls uvažujte obdélníkový; datové symboly jsou nezávislé stejně pravděpodobné
- signál $\tilde{s}[k]$ modulujte na nosný harmonický signál s kmitočtem f_c ; užitečný signál na nosném kmitočtu označíme $s[k]$
- k $s[k]$ přidejte vzorky bílého gaussovského šumu $n[k]$ tak, aby takto získaný signál $x[k] = s[k] + n[k]$ měl požadovanou úroveň odstupů S/N_0 v dB-Hz (S – výkon signálu s , N_0 jednostranná spektrální výkonová hustota aditivního šumu)
- k signálu $x[k]$ určete komplexní obálku $\tilde{x}[k]$ a zobrazte reálnou a imaginární část komplexní obálky v komplexní rovině
- komplexní obálku $\tilde{x}[k]$ určete:
 - filtrační metodou
 - ve spektrální oblasti pomocí operací se spektrem signálu $x[k]$ – dobrovolný úkol
- metody porovnejte – dobrovolný úkol

Pro mé zadání č. 9 platí:

$M = 4$	- pro konkrétní M-PSK
$f_d = 300$ [symbol / s]	- symbolová rychlost
$f_c = 1400$ [Hz]	- nosná frekv.
$S / N_0 = 57$ [dB-Hz]	- odstup signál / šum

2. POSTUP ŘEŠENÍ

Tato úloha dává možnost prozkoumat vlastnosti komplexní obálky a také jejího využití pro práci se signály. Obr. 1 ve zkratce seznamuje s hlavním smyslem této úlohy.



Obr. 1 Rozpracování úlohy s naznačením jednotlivých fází

Pro snadné zorientování v tomto zpracování dané úlohy několik poznámek. Níže jsou uvedeny grafy důležitých průběhů a to přímo ve formě, kde jsou sledované průběhy dosti přiblíženy. Celkový náhled na jednotlivé grafy je k dispozici na konci tohoto zpracování – u zdrojového kódu. To umožňuje snadnou orientaci v kódu a přitom se obrázky dvakrát neopakují, což vede k přehlednosti.

Vysvětleme si nyní užívané značky. V závorce je případně uvedeno, jaká hodnota byla zvolena pro tuto úlohu:

- A - amplituda dané složky signálu ($A = 2$)
- f - frekvence
- f_{sa} - vzorkovací frekvence
- t - čas [s]
- s_{vf} - pásmový signál v časové doméně
- B - šířka pásma
- T_d - doba trvání symbolu (pro mé zadání ... 0,003 s)

Nyní popíšeme samotnou realizaci.

Musíme počítat s tím, že budeme pracovat s poměrně značně strmými skoky, a proto je třeba zvolit dostatečně velký vzorkovací kmitoč. Počet vzorků po navzorkování odpovídá samozřejmě modulační rychlosti (ta vlastně i určuje, jak moc rychle musím vzorkovat) – když dojde za jednotku času k více překmitům ± 1 , je nutno vzorkovat vyšším kmitočtem, abychom tyto vzorky dokázali správně zaznamenat a později opět reprodukovat.

4-PSK je generována v komplexní rovině, užijeme tedy „sin“ a „cos“ ukryté v exponenciále. Vygenerovanou funkci si ovzorkujeme. Na každý „obdélník“ tak připadá několik vzorků.

Po vykreslení v rovině IQ je vidět, že máme zvolenu jednotkovou amplitudu. Toto je tedy vygenerovaný signál a o průběh, který by se mu podobal, se budeme snažit na výstupu. Pro názornost vykreslujeme také časové charakteristiky na vstupu. Ty pak budou sloužit jako srovnání pro výstupní charakteristiky.

Máme tedy již definovanou komplexní obálku $\tilde{s}[k]$. Nyní vytvoříme pásmový signál na základě vztahu:

$$s_{vf}[k] = \text{Re}\{\tilde{s}[k] \cdot e^{j \cdot 2 \cdot \pi \cdot F_c \cdot k}\}.$$

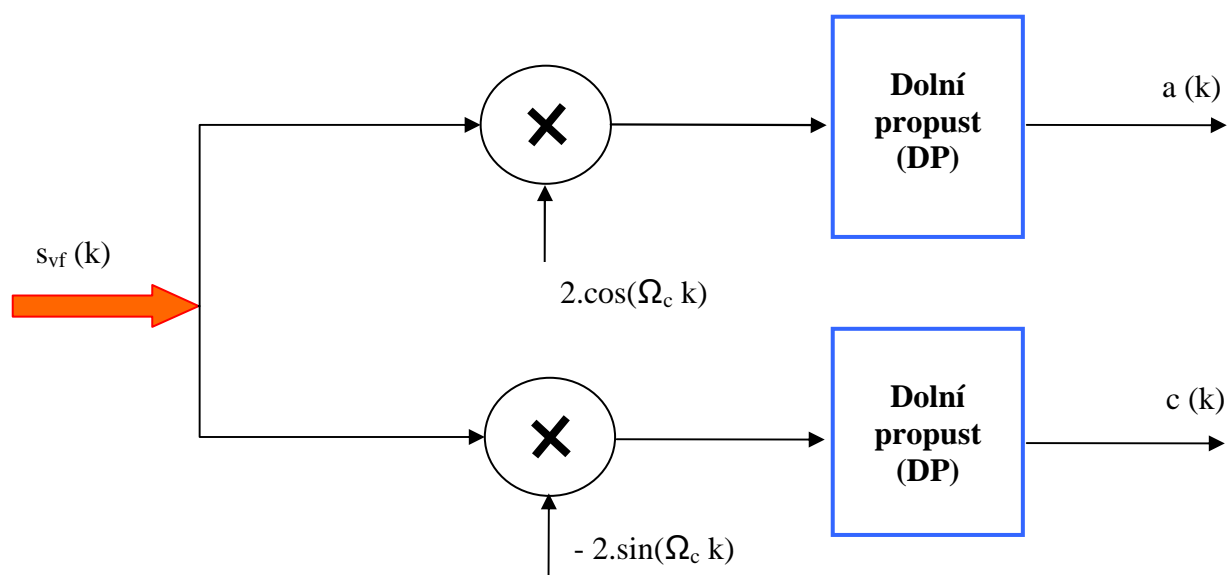
Z výrazu je vidět, že se v podstatě jedná o posun ve spektru o určitou frekvenci F_c . Pokud by frekvence nosné, na kterou signál posuneme, byla větší, mohli bychom signál nyní vyzářit např. anténou. Protože při přenosu se k signálu užitečnému přidává také šum, přidáme ho také, abychom simulovali přenos reálným prostředím.

Již z minulé úlohy máme dobře ověřeno počítání s výkony. To nyní využijeme, protože je nutné spočítat výkon pásmového signálu a přidat k němu šum o požadovaném odstupu od něj. Zadaný odstup S/N_0 je třeba „oddeciblovat“, protože spočtený výkon není v dB. Výkon signálu určíme z časového průběhu jako součet vzorků signálu v absolutní hodnotě na druhou, vhodně zanormováno. Získaný šum s požadovaným odstupem od signálu nakonec přičteme k signálu.

Spektrum komplexní obálky na „přijímací“ straně získáme vyříznutím z posunutého spektra pásmového signálu $2 \cdot S_{vf}(F + F_c)$. Tomuto způsobu se říká filtrační metoda. Její konkrétní provedení je na základě ortogonální reprezentace komplexní obálky:

$$\tilde{s}[k] = a(k) + jc(k)$$

Realizace použitá v této úloze je znázorněna na Obr. 2



Obr. 2 Pohled na celkový časový průběh; detail viz obr. Obr. 3

Filtrace je u této metody stěžejní záležitostí. Filtr jsem navrhoval pomocí nástroje v Matlabu - „fdatool“. Šířka pásma byla volena tak, aby uvnitř zůstalo alespoň 90% výkonu. Je to tedy šířka pásma definovaná prvními minimy „sample“ funkce. Zjednodušeně uvažujeme:

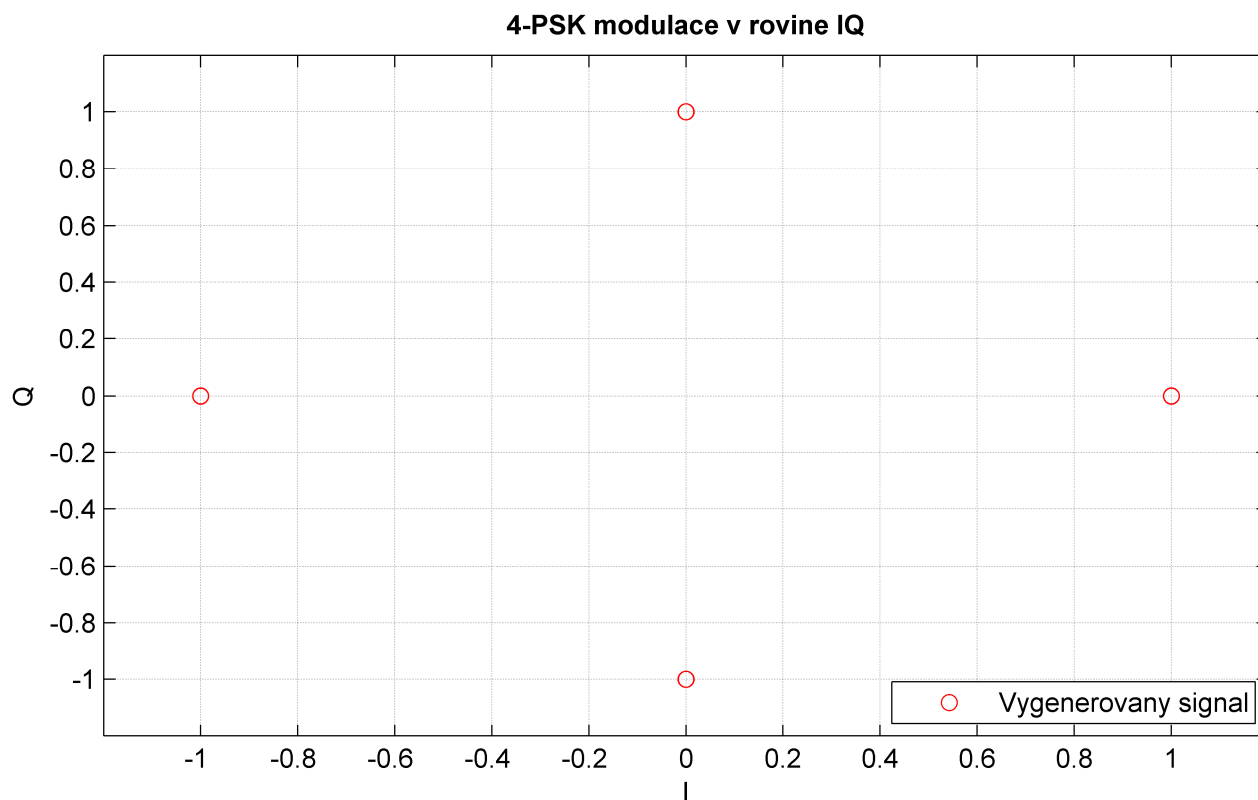
$$B = \frac{2}{T_d} = \frac{2}{0,003} = 600 \text{ Hz}$$

Propustné pásmo pro DP uvažujeme tedy 300Hz.

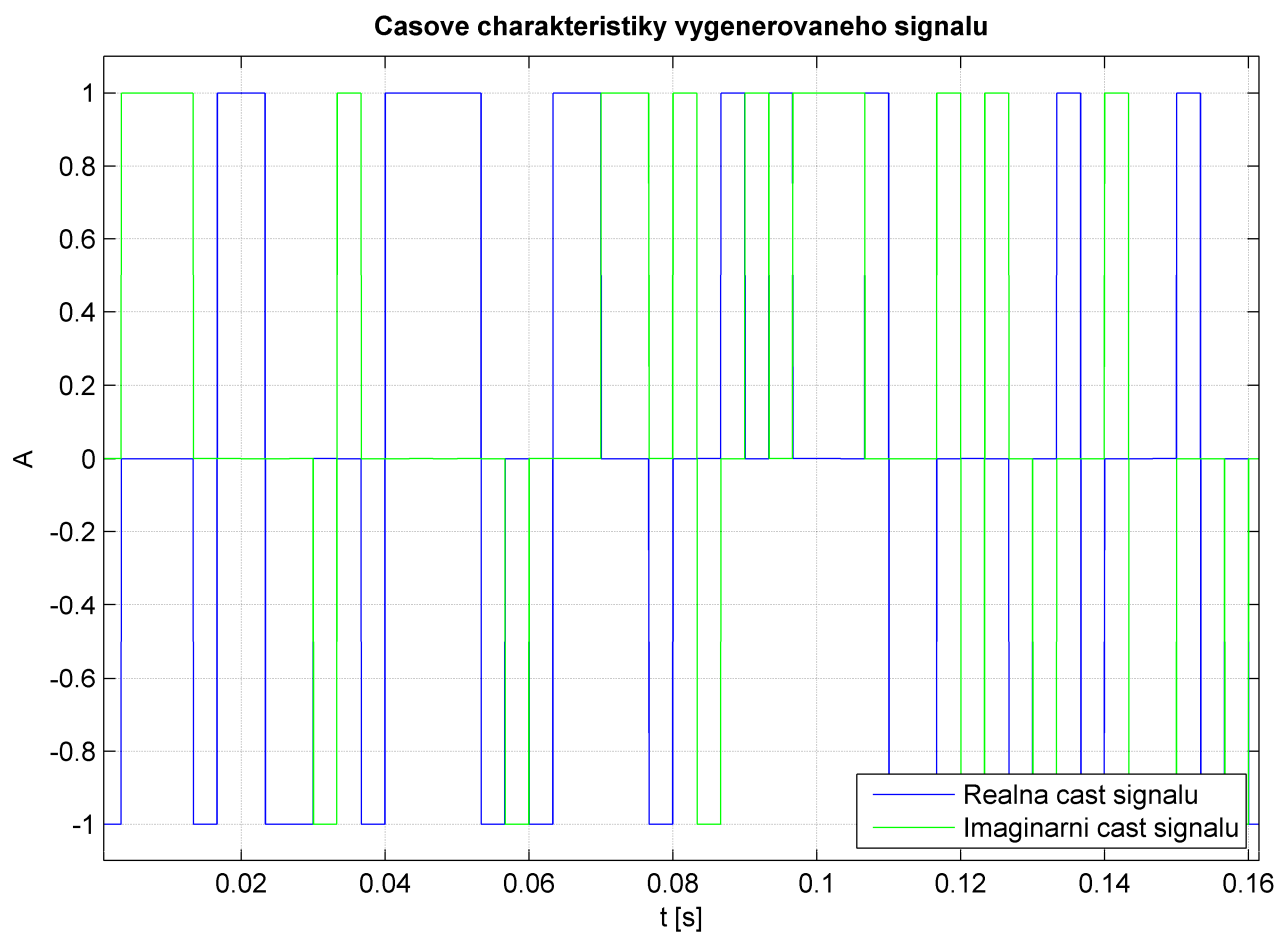
Po všech těchto operacích a závěrečném součtu obou větví dle výše uvedeného vztahu máme opět zrekonstruovanou komplexní obálku, tentokrát ale ovlivněnou šumem a nedokonalostmi filtru.

Informace, které zde nezmiňuji, jsou ve většině případů vyjasněny v dodatku, kde je umístěn zdrojový kód programu i s komentáři, ilustračními obrázky a výpisy některých hodnot proměnných. Informace lze nalézt také v popiscích ke grafům a v samotných grafech.

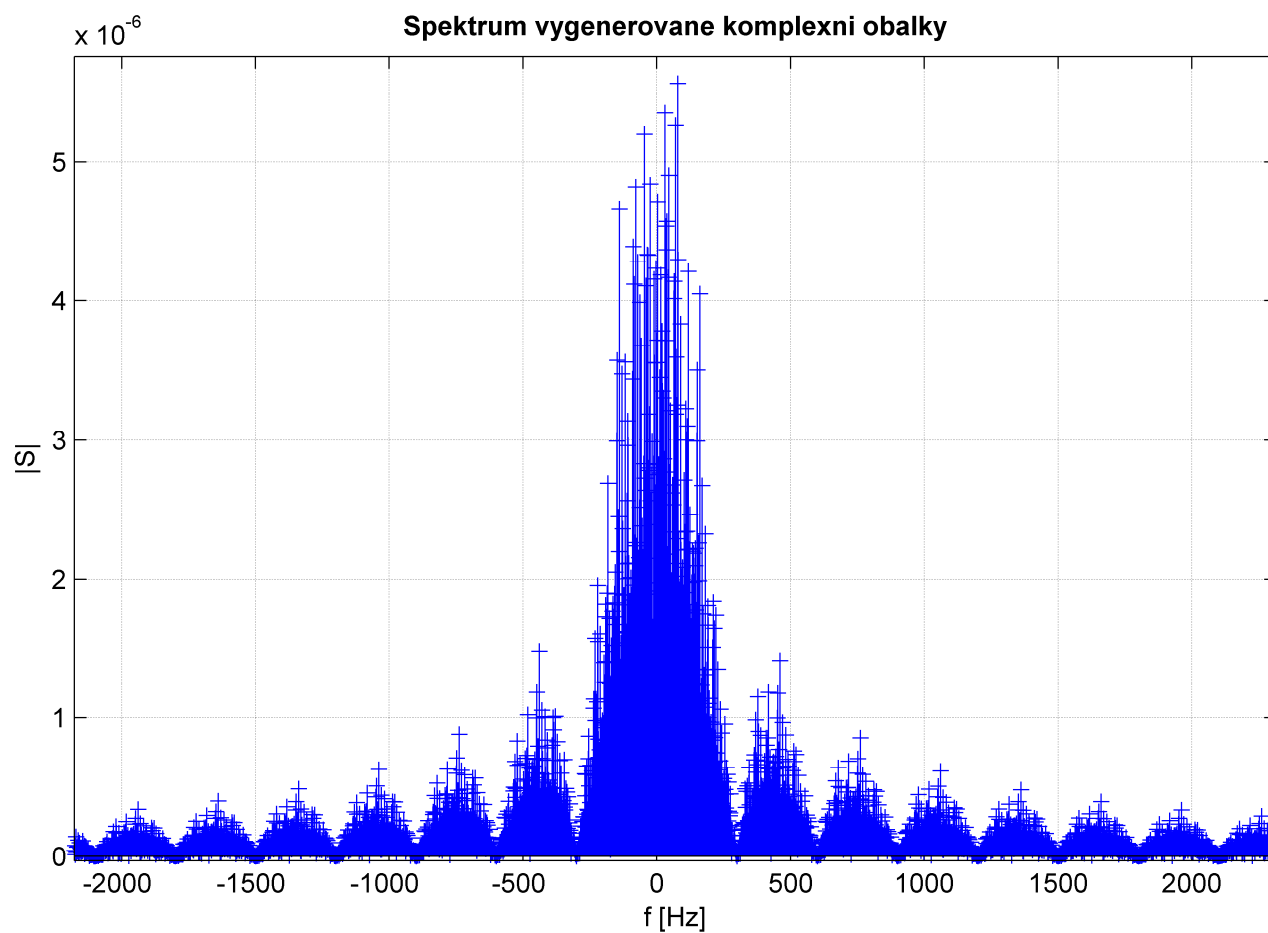
3. GRAFY



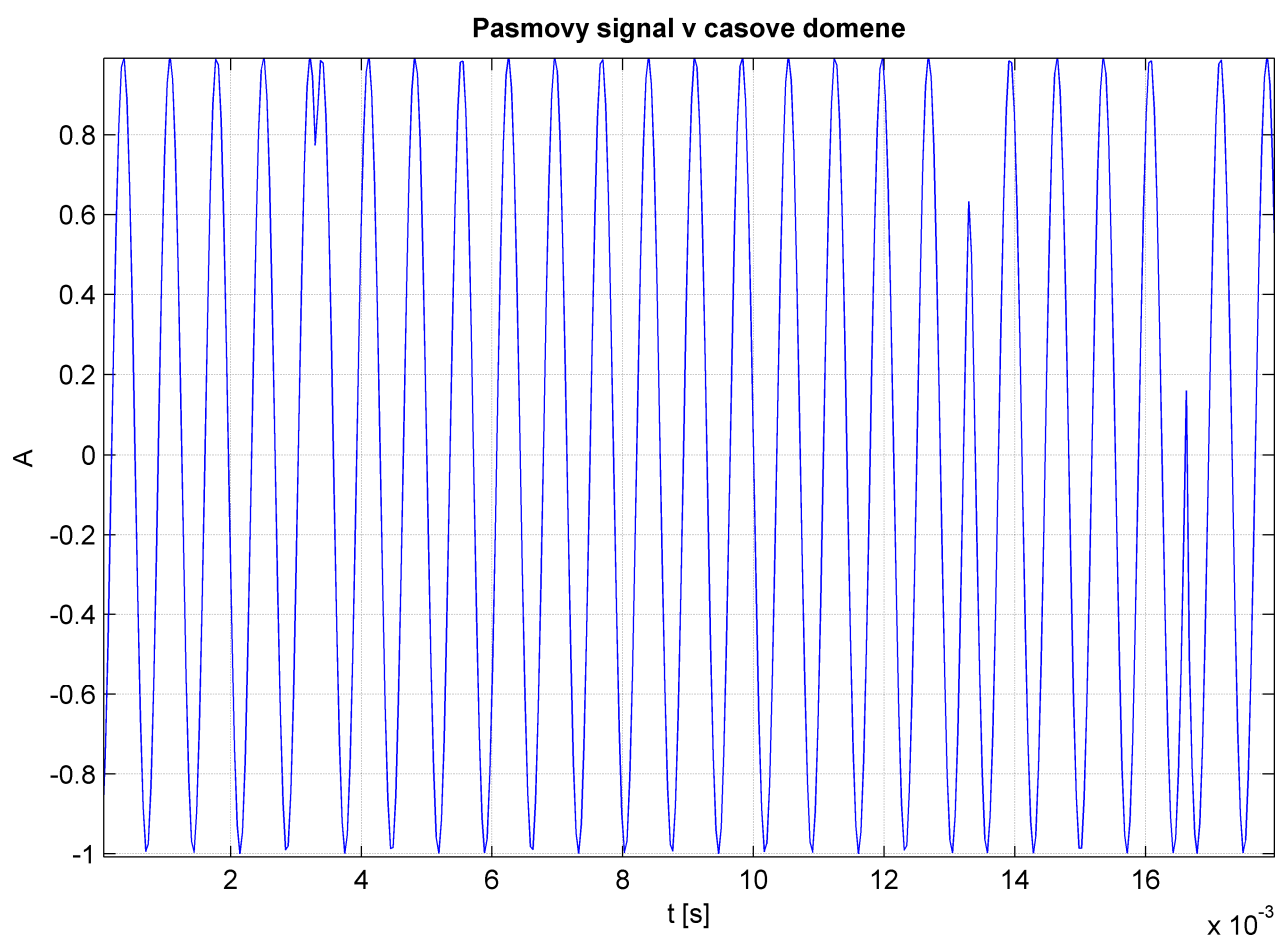
Obr. 3 Zobrazení generovaného signálu v komplexní rovině



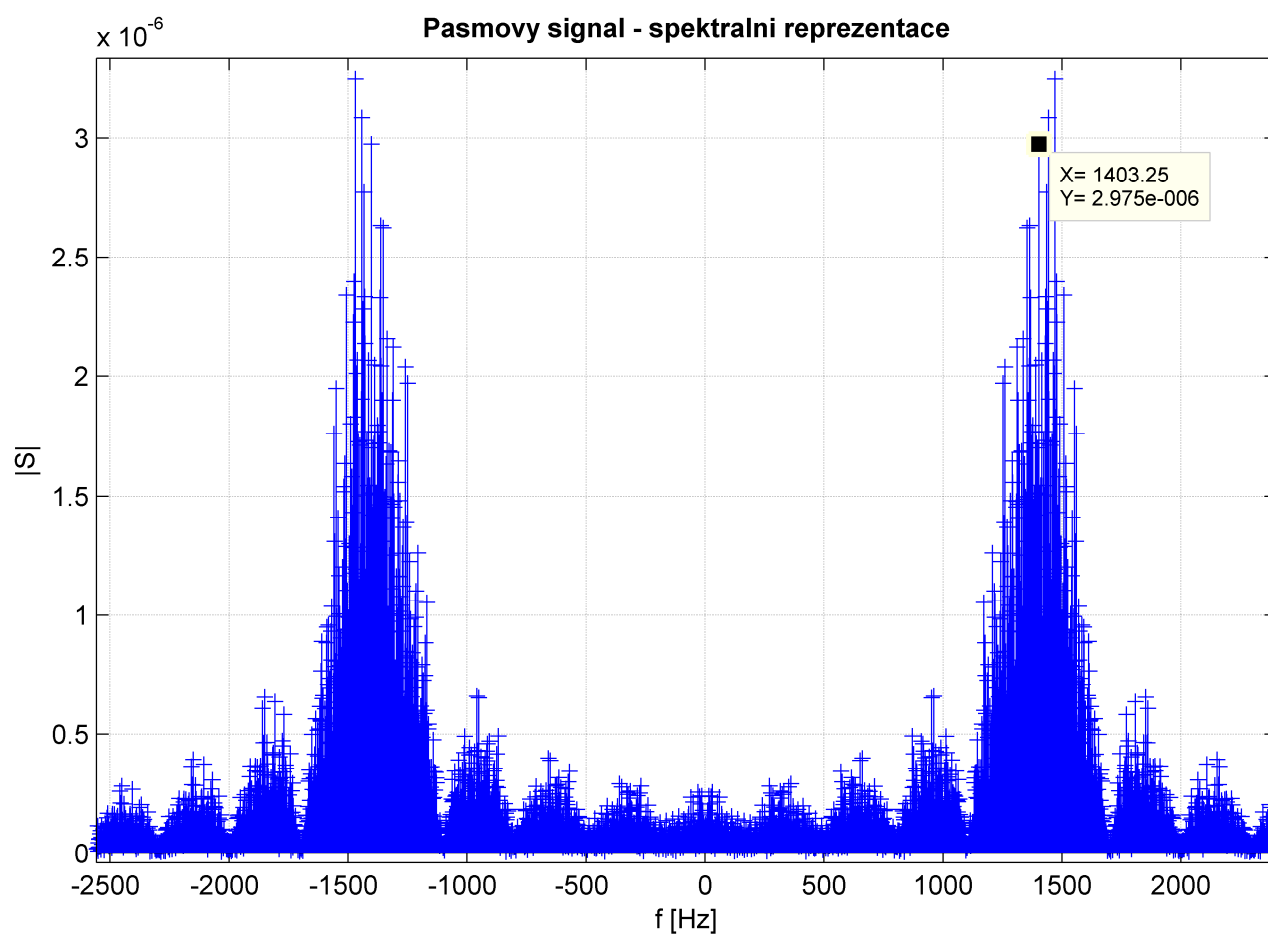
Obr. 4 Komplexní obálka – časové průběhy. Jedná se o úsek signálu – prvních 160 ms



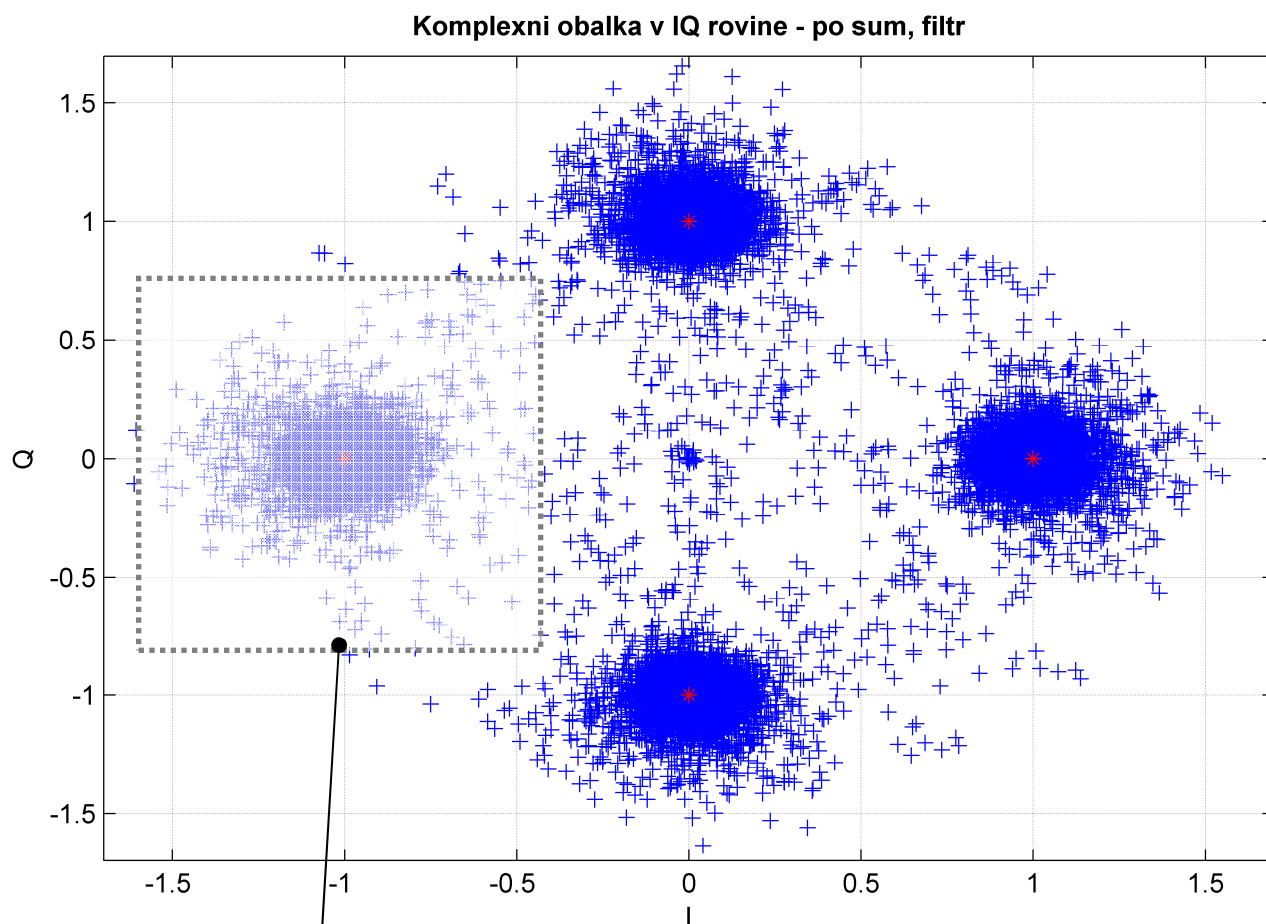
Obr. 5 Komplexní obálka – frekvenční spektrum. Je pěkně vidět sample funkce pocházející z obdélníka vytvořeného v časové doméně.



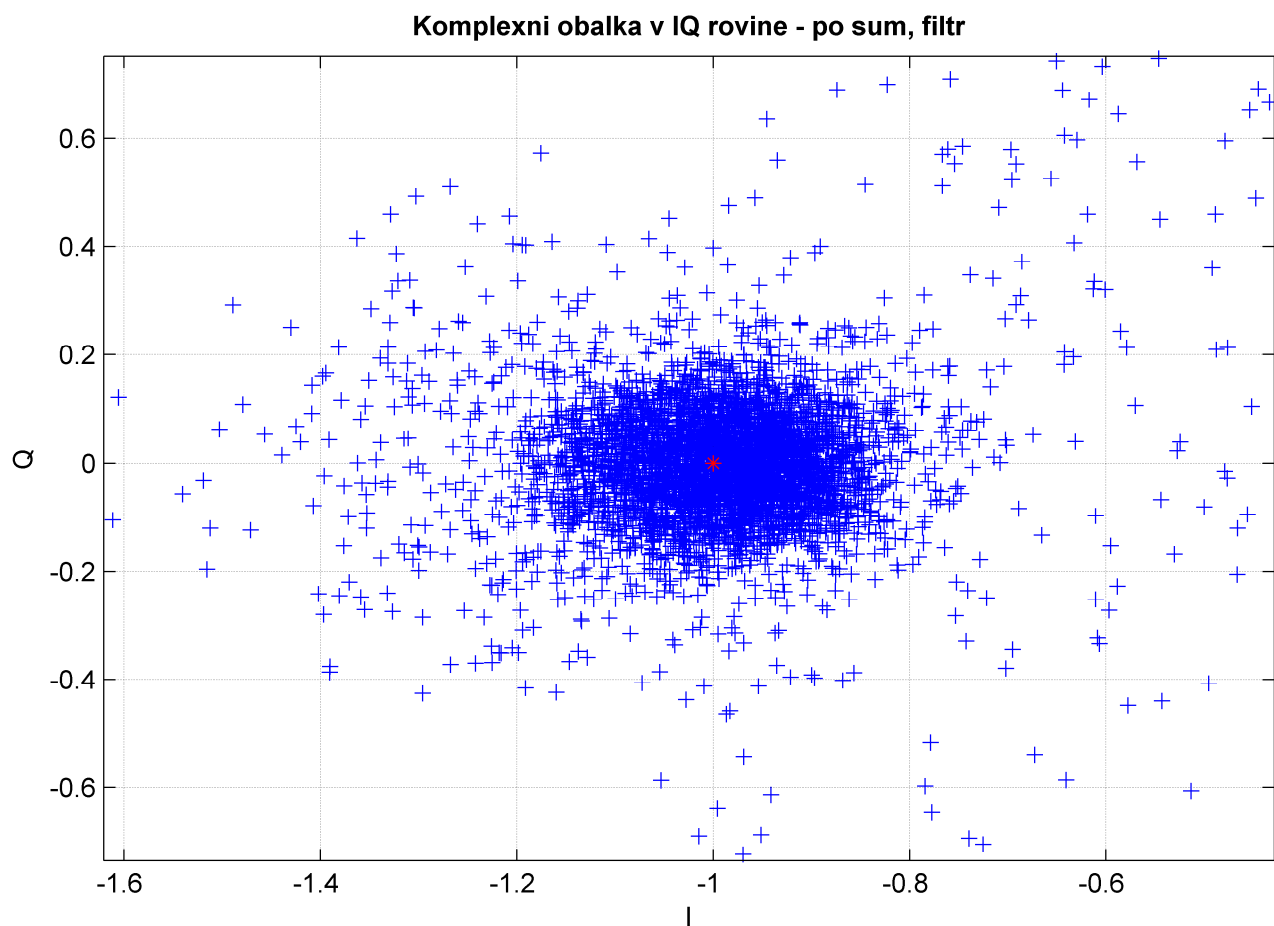
Obr. 6 Pásmový signál – časový průběh. Modulace je vidět např. v určitých skocích harmonické křivky (zlomy).



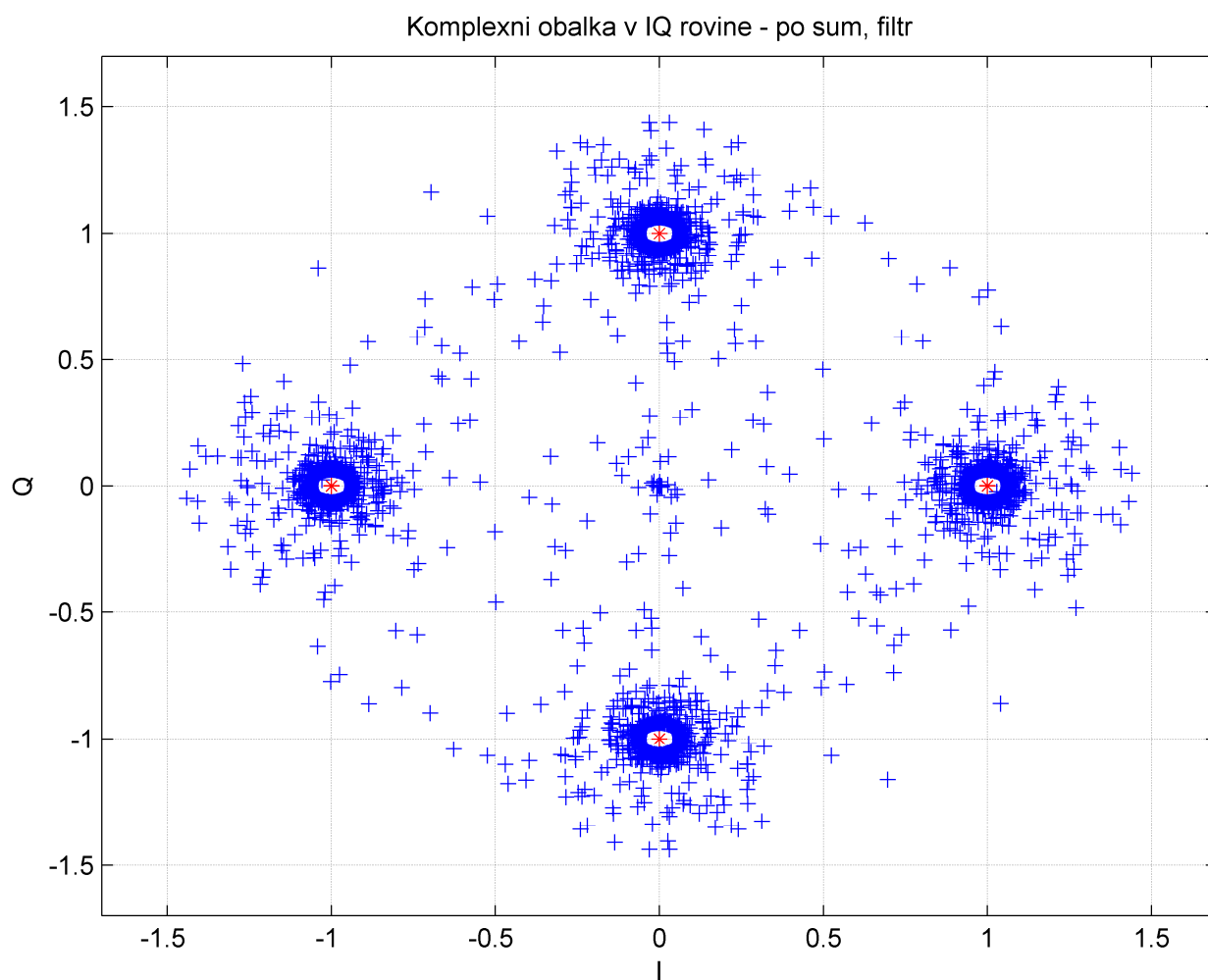
Obr. 7 Pásmový signál zobrazený spektrální charakteristikou. Je vidět, že se podařilo namodulovat na 1400 Hz.



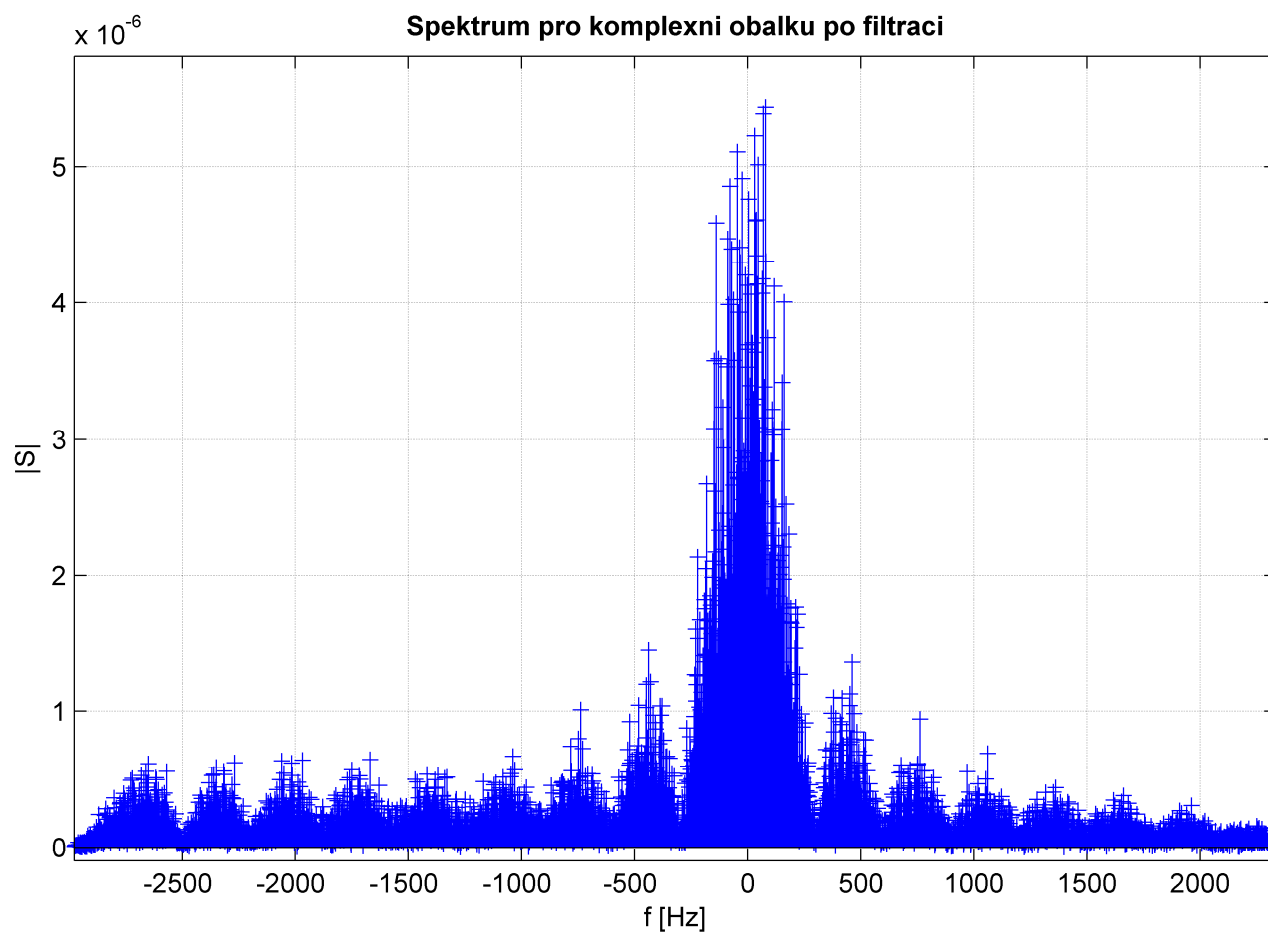
Obr. 8 Výstupní signál po „demodulaci“ a znovuvytvoření komplexní obálky. Je vidět vliv nedokonalosti filtru a také šum. Červené tečky znázorňují teoreticky správné hodnoty – tedy tam by se měly teoreticky nacházet i všechny modré vzorky signálu, pokud by byly všechny podmínky ideální atd.



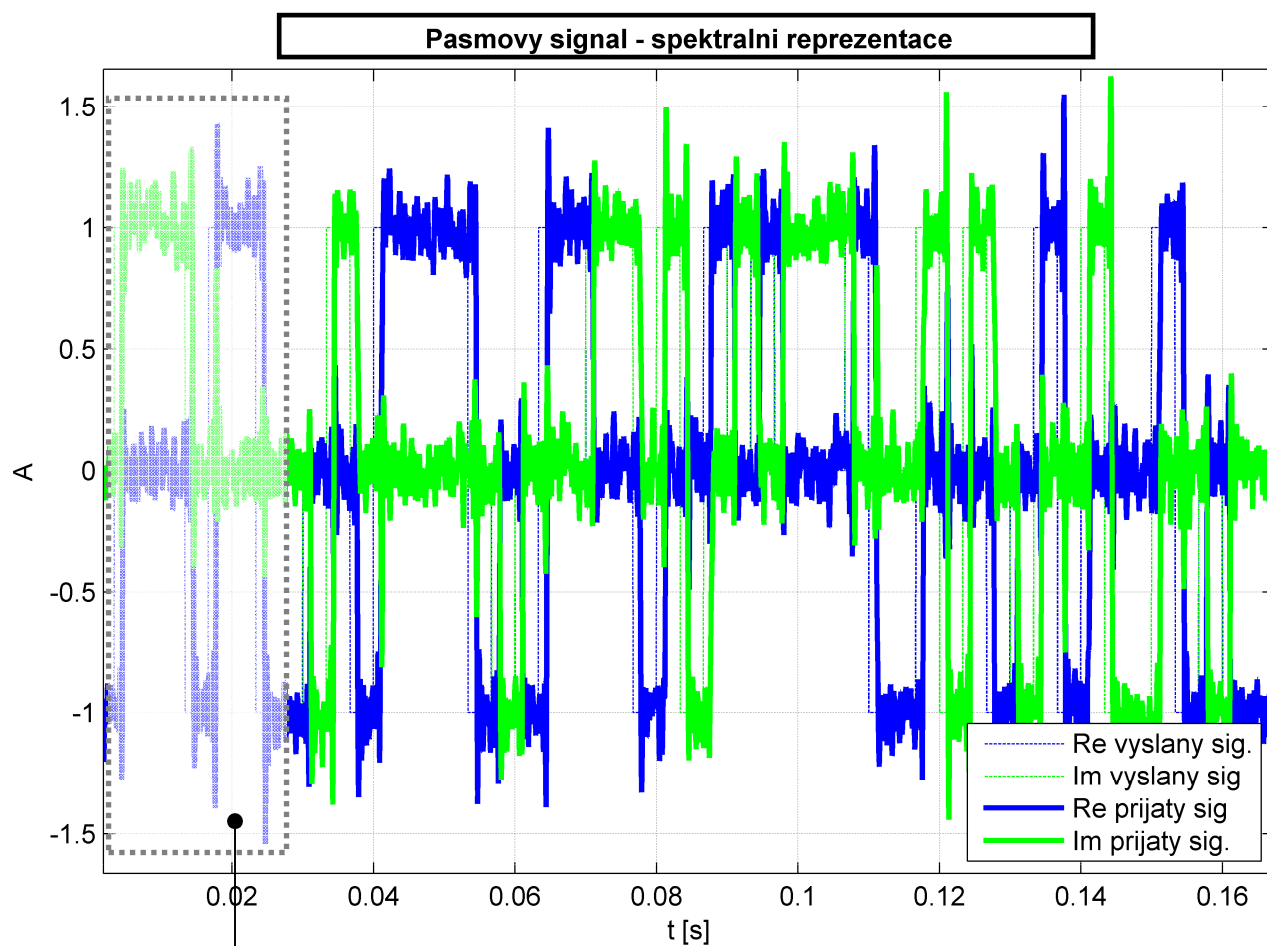
Obr. 9 Detail IQ diagramu pro vystup. Jasně ukazuje, že kolem teoreticky správného výsledku je soustředěno nejvíce vzorků.



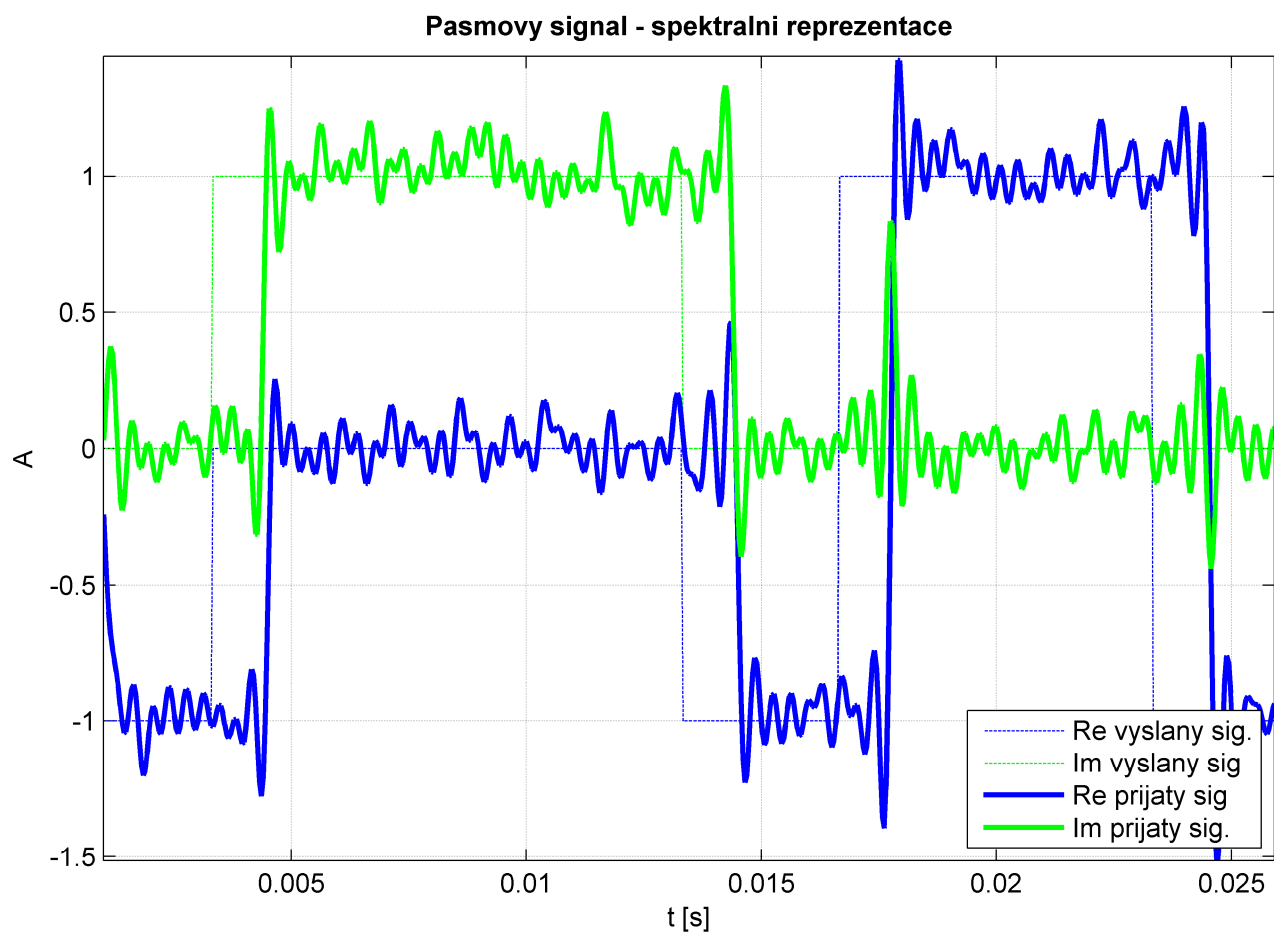
Obr. 10 Ukázka pro odstup $S/N_0 = 257$ dB-Hz – tedy prakticky pro zanedbatelný šum. Je vidět nedokonalost zvoleného postupu filtrace – viz závěr.



Obr. 11 Spektrum komplexní obálky po filtraci – zůstává opravdu jen významná část v okolí 0 Hz.



Obr. 12 Komplexní obálka v časové oblasti. Je pěkně vidět, že dochází k překmitům při snaze o rekonstrukci původního obdélníka.



Obr. 13 Z tohoto detailu je pěkně vidět časový posun – zpoždění. Pravděpodobně způsobuje průchod filtrem.

4. ZÁVĚR

Tato úloha byla velmi zajímavá a názorná a podařilo se mi myslím lépe se zorientovat v otázkách modulace, komplexní obálky, pásmových signálů a návrhu filtrů.

Dle zadaného požadavku na práci s „obdélníkovým“ signálem však práce ukazuje na nějaké obtíže. Obdélníkový modulační impuls je totiž velmi nepříjemný pro modulaci. Ve spektru je to vidět jako průběh „sample“ funkce. V této úloze jsem se rozhodl pro přímou práci s filtrem apod. a to bez různých dalších možností, jak výsledek vylepšit. Výsledek je tedy nutno chápat jako ilustrační a rozhodně ne vhodný pro konkrétní realizaci. Vyladění filtru, abychom dostali na výstupu pěkný signál, je dosti pracné.

Již bez šumu se v IQ diagramu objevuje několik bodů, které jsou naprosto chybně umístěny. Na druhou stranu však naprostá většina bodů je soustředěna okolo skutečně vyslaných vzorků. Pro představu je celkový počet 16 200 vzorků a přijato a vyhodnoceno špatně by bylo pravděpodobně jen několik desítek. Z tohoto poměru vyplývá, že zadání úlohy bylo splněno. Z časových průběhů je vidět ještě jasněji, že základní průběh signálu byl zachován, jen je třeba počítat se zvlněním na hranách obdélníků.

Je nesporné, že je možno dále vylepšovat. Diskutoval jsem s některými kolegy uvažovaná řešení, a co jsem měl možnost porovnat, tak nejlepší výsledky dávala metoda, při které se pracovalo s převzorkováním, filtrací a následným podvzorkováním. Je pak možno zvolit jednodušší filtr a nedojde pak k takovému zpoždění signálu oproti vyslanému (u řešení v této úloze je zpoždění cca 1,3 ms). Výhodné je rovněž použití Hanningova okna či užití jiného modulačního impulsu.

Pokusil jsem se také o určení komplexní obálky pomocí operací se spektrem signálu. Tato metoda se mi však příliš nepodařila realizovat. Pravděpodobně díky obtížím ve spektru způsobených prací s obdélníkem se mi nepodařilo demodulovat nic smysluplného.

I když je tedy možné dosáhnout daleko líbivějších výsledků, rád jsem vyzkoušel tuto v podstatě jednoduchou metodu, protože jsem si tak ověřil vlastnosti komplexní obálky.

5. DODATEK

Contents

- [Zadani](#)
- [Uvodni definovani](#)
- [Vygenerovani M-PSK](#)
- [Charakteristiky vygenerovaneho signalu](#)
- [Prevod na pasmovy signal](#)
- [Pridani sumu](#)
- [Ziskani opet komplexni obalky I](#)
- [Filtrace](#)
- [Konecne ziskani komplexni obalky - II](#)

```
clear all
close all
```

Zadani

```
M = 4;           % Pocet stavu MPSK
fc = 1400;       %Hz - nosna
fd = 300;        % symbol/sec - symbolova rychlost
S_No = 57;       % dB-Hz
```

Uvodni definovani

```
N_dat=200; %% pocet datovych symbolu
fd=300; %% symbolova/bitova rychlost
Q = 1      %% Kolikrat zvetsujeme osy pro prepocet do vf
f_sa = 81*fd
N_sa = floor(N_dat*f_sa)/fd % pocet nasamplovanych vzorku pak odpovida
                             %pomeru mezi vzorkovaci a symbolovou frekvenci
T = N_dat / fd

t = ( 0:N_sa-1)/N_sa*T;      % Casova osa v BB
freqo=-((N_sa-1)/N_sa*f_sa)/2:f_sa/N_sa:((N_sa-1)/N_sa*f_sa)/2;
```

Q =

1

f_sa =

24300

N_sa =

16200

T =

0.6667

Vygenerovani M-PSK

```
pocet_ovzorkovanych_datovych_vzorku = 0;
data = 1*exp(j*2*pi/M * floor(M*randn(N_dat ,1) ) );
for i=0:N_sa-1,
    s(i+1) = data(floor(i*fd/f_sa)+1);
```

```

    pocet_ovzorkovanych_datovych_vzorku = pocet_ovzorkovanych_datovych_vzorku + 1;
end;
pocet_ovzorkovanych_datovych_vzorku = pocet_ovzorkovanych_datovych_vzorku

%S = s';

```

```

pocet_ovzorkovanych_datovych_vzorku =

    16200

```

Charakteristiky vygenerovaného signálu

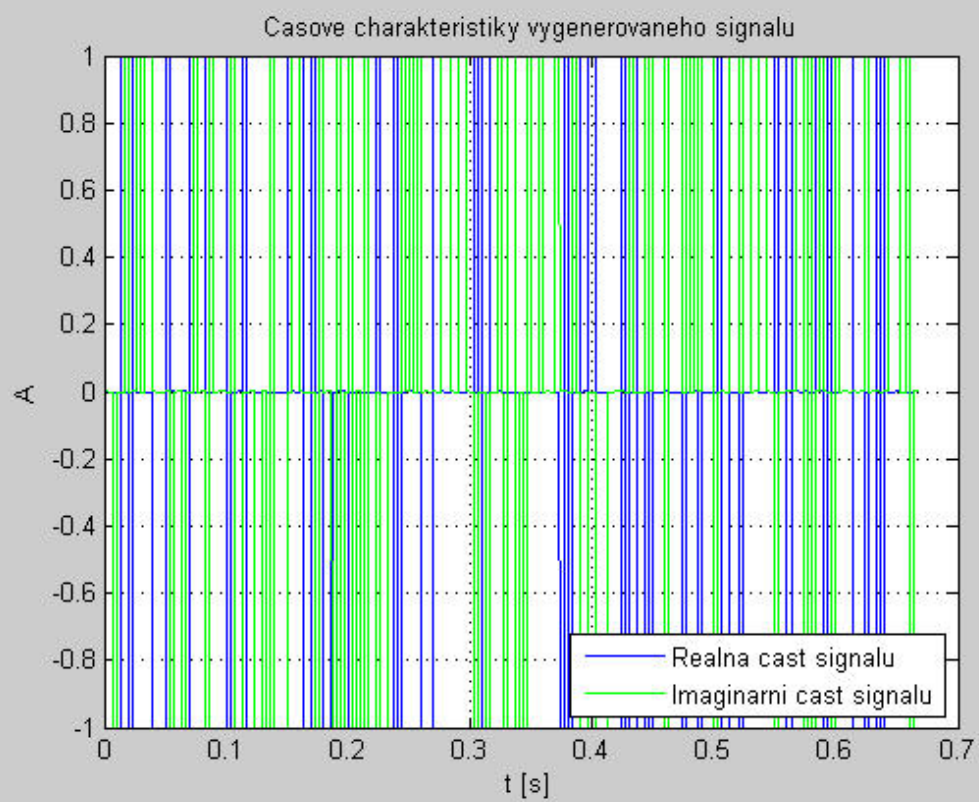
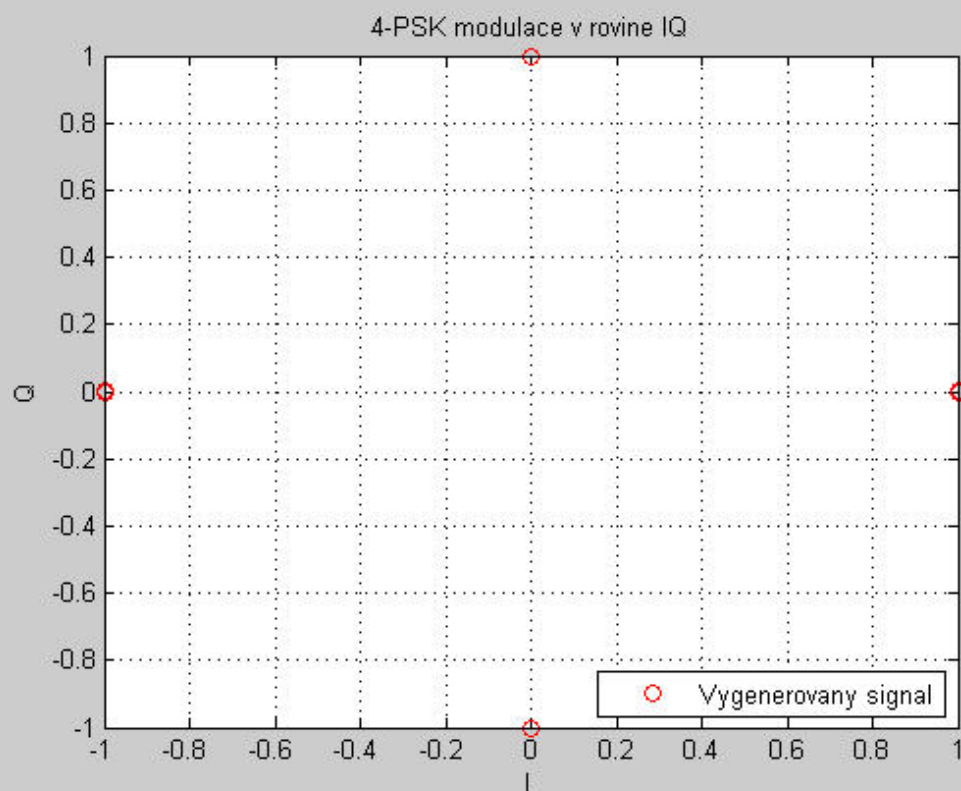
Zakreslení signálu v rovině IQ

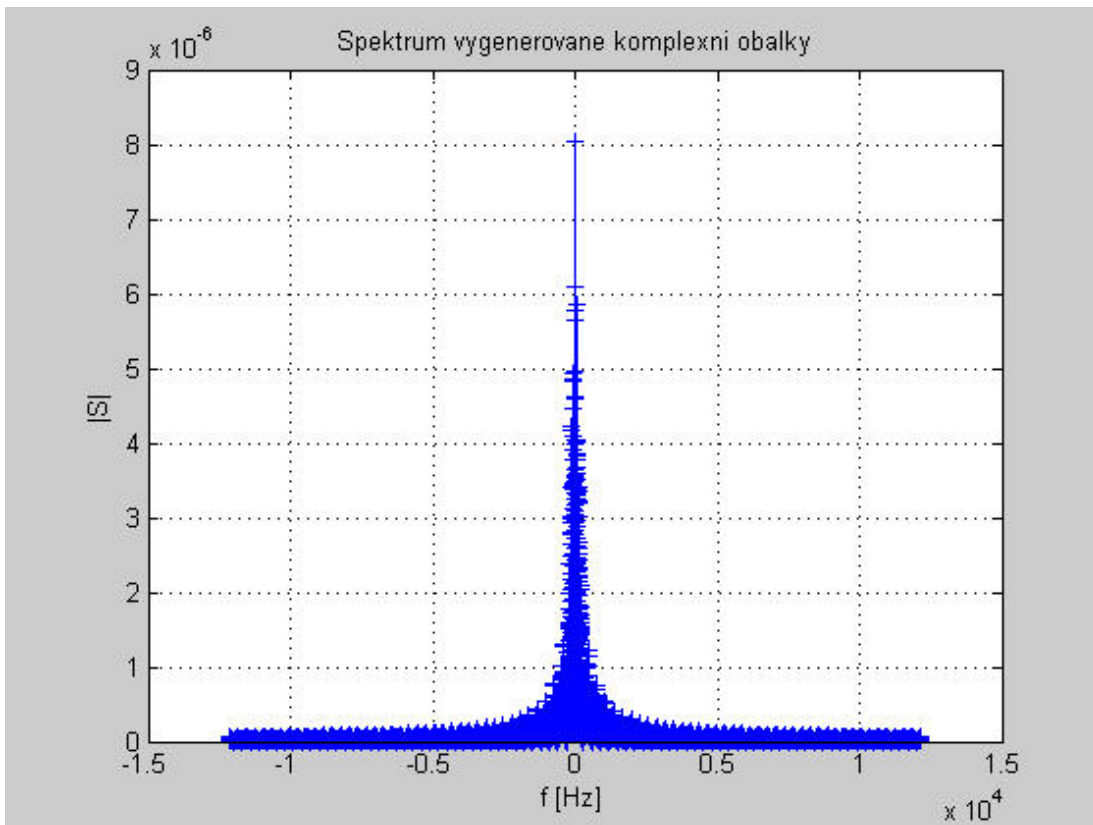
```

figure(1)
plot( s , 'ro')
grid on
title('4-PSK modulace v rovině IQ');
legend('Vygenerovaný signál',4);
xlabel('I');
ylabel('Q');
% Casové charakteristiky signálu pro jeho reálnou a imag. část
figure(2);
plot(t,real(s))
%stem(t,real(s),'b*')
hold on
plot(t,imag(s),'g-')
%stem(t,imag(s),'g-')
grid on
title('Casové charakteristiky vygenerovaného signálu');
legend('Reálná část signálu','Imaginární část signálu',4);
xlabel('t [s]');
ylabel('A');

% Komplexní obálka po vygenerování, její spektrální reprezentace
S = (1/N_sa/f_sa)*fftshift(abs(fft(s)));
figure(3)
stem(freco, S, 'b+');
grid on
title('Spektrum vygenerované komplexní obálky');
xlabel('f [Hz]');
ylabel('|S|');

```





Prevod na pasmovy signal

```
s_vf = real(s.*exp(1j*2*pi*fc.*t));

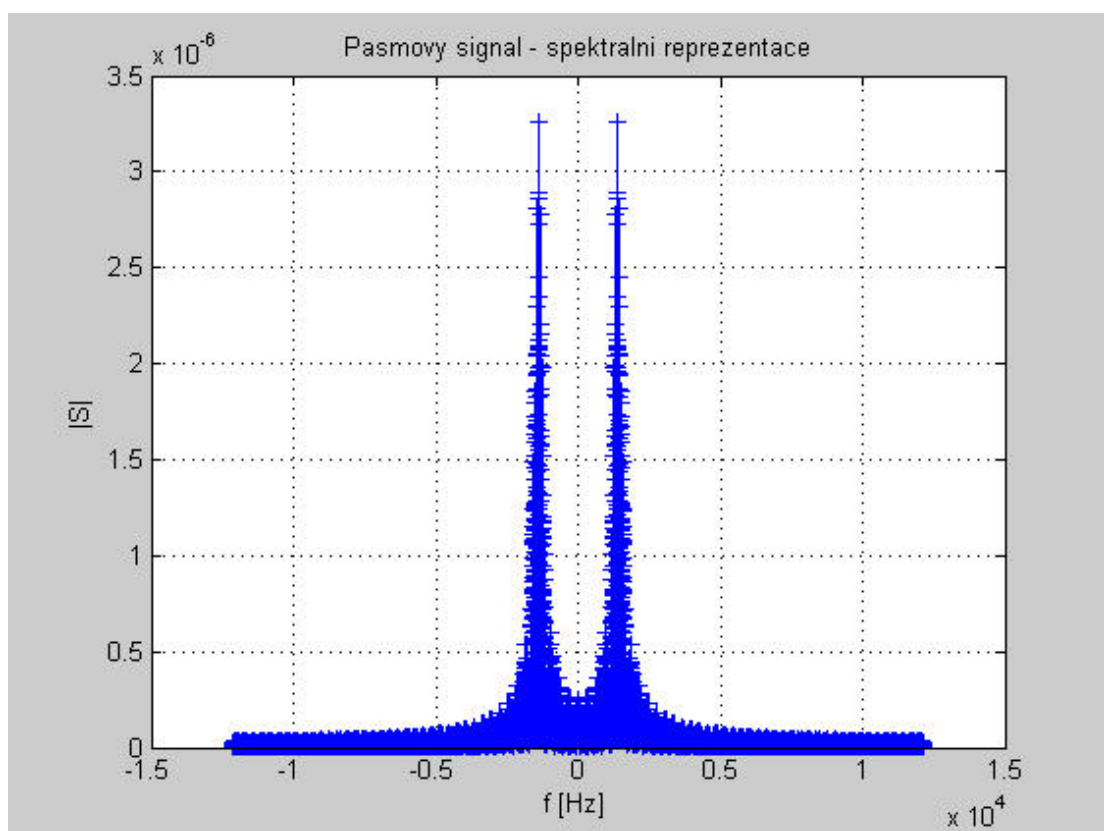
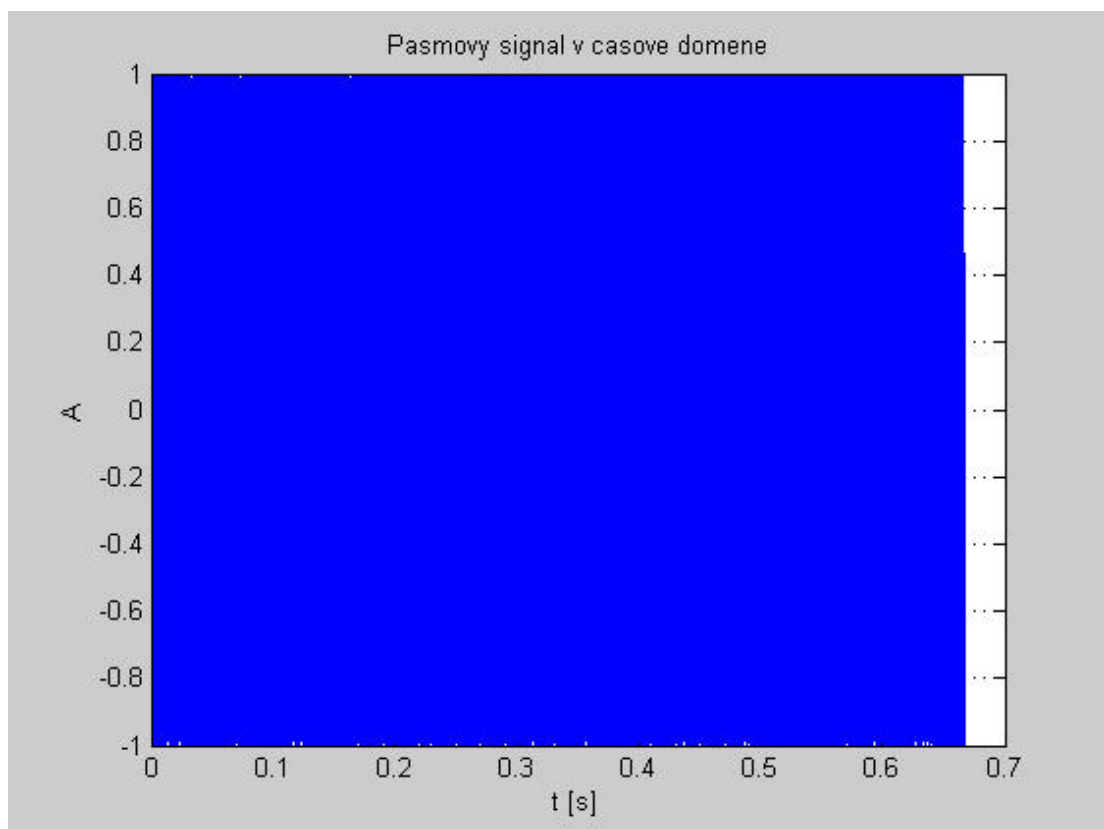
% Zobrazeni pasmoveho signalu v casove domene
figure(4)
plot(t,s_vf)
grid on
title('Pasmovy signal v casove domene');
xlabel('t [s]');
ylabel('A');

% Pro pasmovy signal udelame "sirsi" kmitoctovou osu
% parametr Q značí, o kolik jsme to museli protáhnout
% Protože musí "sedet" dimenze matic, zvýsíme Q-krát počet vzorku v s_vf
% Uzito jen pro vykresleni grafu ... stary fragment, ponechano pro
% zajimavost...
s_vf_up = resample(s_vf,Q,1);
%freco_vf = (-((Q*N_sa-1)/N_sa*f_sa)/2:(f_sa/N_sa):((Q*N_sa-1)/N_sa*f_sa)/2);

% Vykresleni pasmoveho signalu - jiz je namodulovano na nosnou
figure(5)
pasmovy_signal = (1/N_sa/f_sa)*fftshift((abs(fft(s_vf))));
stem(freco,pasmovy_signal,'b+')
grid on
title('Pasmovy signal - spektralni reprezentace');
xlabel('f [Hz]');
ylabel('|S|');

% freco_vf = (0:(f_sa/N_sa):((N_sa-1)/N_sa*f_sa));
% figure(5)
% plot(freco_vf,(1/((N_sa)/f_sa))*((abs(fft(s_vf)))))

% Delame-li spektralni hustotu vstupniho vykonu
% je podelena delkou periody
% C_s = 1/(N_sa.^2)/f_sa*(abs(fft(s))).^2; % umocneni N_sa.^2 provedeno po konzultaci s
% figure(6);
% plot(freco, fftshift(C_s))
```



Pridani sumu

(simulujeme tim pruchod napr. realnym prostredim) --- tim ziskame souctovy signal x

```
x = s_vf;      %%% !!!
noise = randn( N_sa,1 );
P_s_vf = 1/N_sa * s_vf*s_vf'

k = sqrt( (P_s_vf/(10.^(S_No/10)) * (f_sa)/2 ) );
x = x + k.*(noise');
```

```
% t = (( 0:Q*N_sa-1)'/Q*N_sa*T); % pokus
```

```
P_s_vf =  
0.5008
```

Ziskani opet komplexni obalky I

```
s_a = 2*x .* cos(2*pi*fc.*t);  
s_c = -2*x .* sin(2*pi*fc.*t);
```

Filtrace

Zvolena byla tato moznost: pod ni jsou dalsi zkousene moznosti filtru *****
Equiripple Lowpass filter designed using the FIRPM function.

```
% All frequency values are in Hz.  
Fs = f_sa; % Sampling Frequency  
  
Fpass = 300; % Passband Frequency  
Fstop = 3000; % Stopband Frequency  
Dpass = 0.0000000000015501127785; % Passband Ripple  
Dstop = 0.8; % Stopband Attenuation  
dens = 20; % Density Factor  
  
% Calculate the order from the parameters using FIRPMORD.  
[N, Fo, Ao, W] = firpmord([Fpass, Fstop]/(Fs/2), [1 0], [Dpass, Dstop])  
  
% Calculate the coefficients using the FIRPM function.  
B = firpm(N, Fo, Ao, W, {dens});  
% Filtrace:  
s_a_f = filter(B, 1, s_a); % filtrovana soufazova slozka  
s_c_f = filter(B, 1, s_c); % filtrovana kvadrturni slozka  
  
% *****  
  
% % % % BUTTERWORTH:  
% % Fs = f_sa; % Sampling Frequency  
% % Fpass = 300; % Passband Frequency  
% % Fstop = 15000; % Stopband Frequency  
% % Apass = 1; % Passband Ripple (dB)  
% % Astop = 20; % Stopband Attenuation (dB)  
% % match = 'passband'; % Band to match exactly  
% % Construct an FDESIGN object and call its BUTTER method.  
% % h = fdesign.lowpass(Fpass, Fstop, Apass, Astop, Fs);  
% % b = butter(h, 'MatchExactly', match);  
% % s_a_f = filter(b,s_a); % filtrovana soufazova slozka  
% % s_c_f = filter(b,s_c);  
  
% *****  
  
% % CHEBYSHEV Type I Lowpass filter designed using FDESIGN.LOWPASS.  
% % All frequency values are in Hz.  
% % Fs = f_sa; % Sampling Frequency  
% % Fpass = 300; % Passband Frequency  
% % Fstop = 15000; % Stopband Frequency  
% % Apass = 1; % Passband Ripple (dB)  
% % Astop = 2; % Stopband Attenuation (dB)  
% % match = 'passband'; % Band to match exactly  
% % Construct an FDESIGN object and call its CHEBY1 method.  
% % h = fdesign.lowpass(Fpass, Fstop, Apass, Astop, Fs);  
% % b = cheby1(h, 'MatchExactly', match);  
% % s_a_f = filter(b,s_a); % filtrovana soufazova slozka  
% % s_c_f = filter(b,s_c);
```

N =

56

Fo =

0
0.0247
0.2469
1.0000

Ao =

1
1
0
0

W =

1.0e+011 *

5.1609
0.0000

Konecne ziskani komplexni obalky - II

```
% Zobrazeni signalu v rovine IQ po "zasumeni" a "demodulaci"
signal = s_a_f + 1j.*s_c_f;
figure(7)
plot(signal, '+')
hold on
plot(s,'r*')
grid on
title('Komplexni obalka v IQ rovine - po sum, filtr');
xlabel('I');
ylabel('Q');

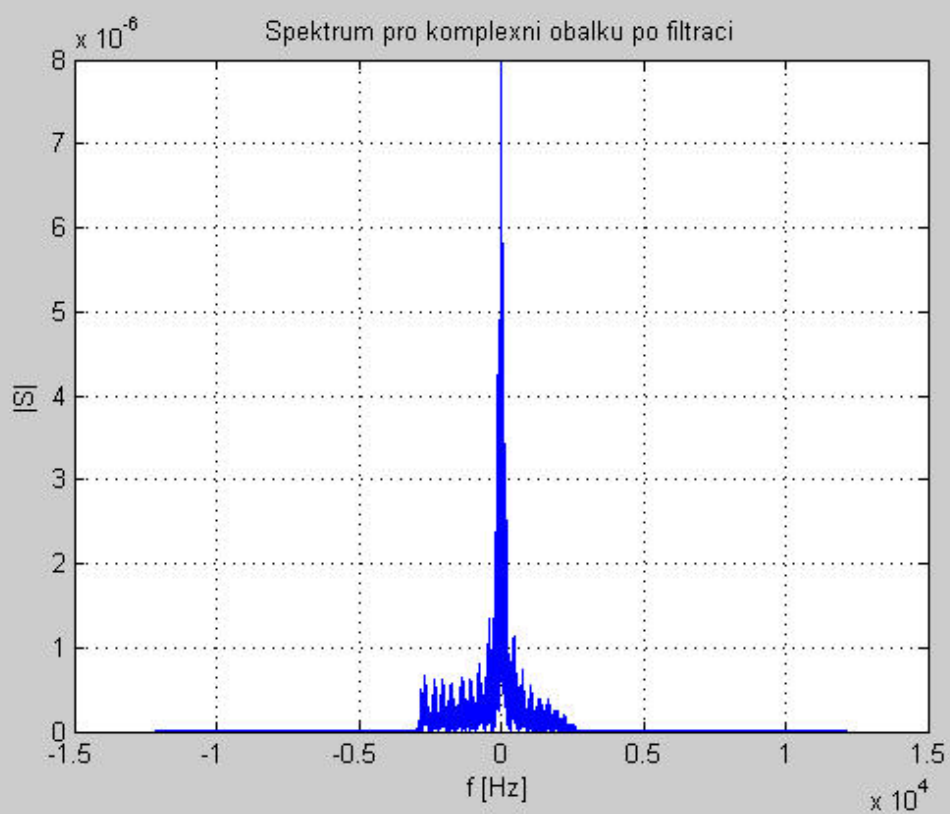
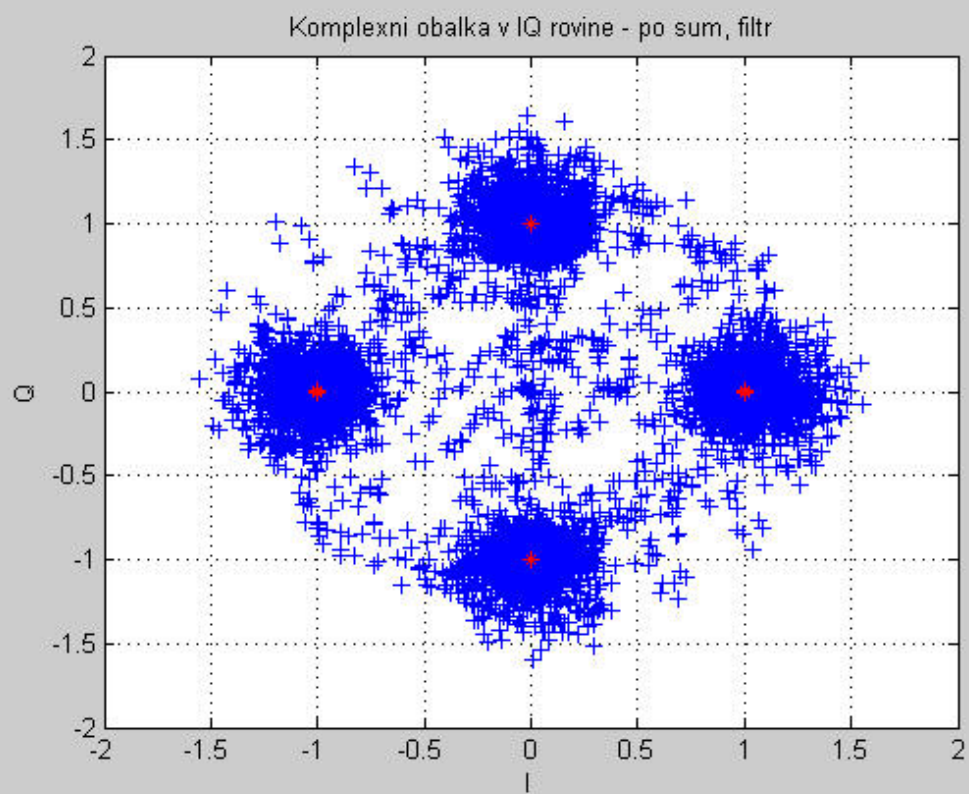
% Vykresleni spektra po filtraci
figure(8)
vyfiltr_obalka = (1/N_sa/f_sa)*fftshift(abs(fft(signal)));
plot(freco, vyfiltr_obalka);
grid on
title('Spektrum pro komplexni obalku po filtraci');
xlabel('f [Hz]');
ylabel('|S|');

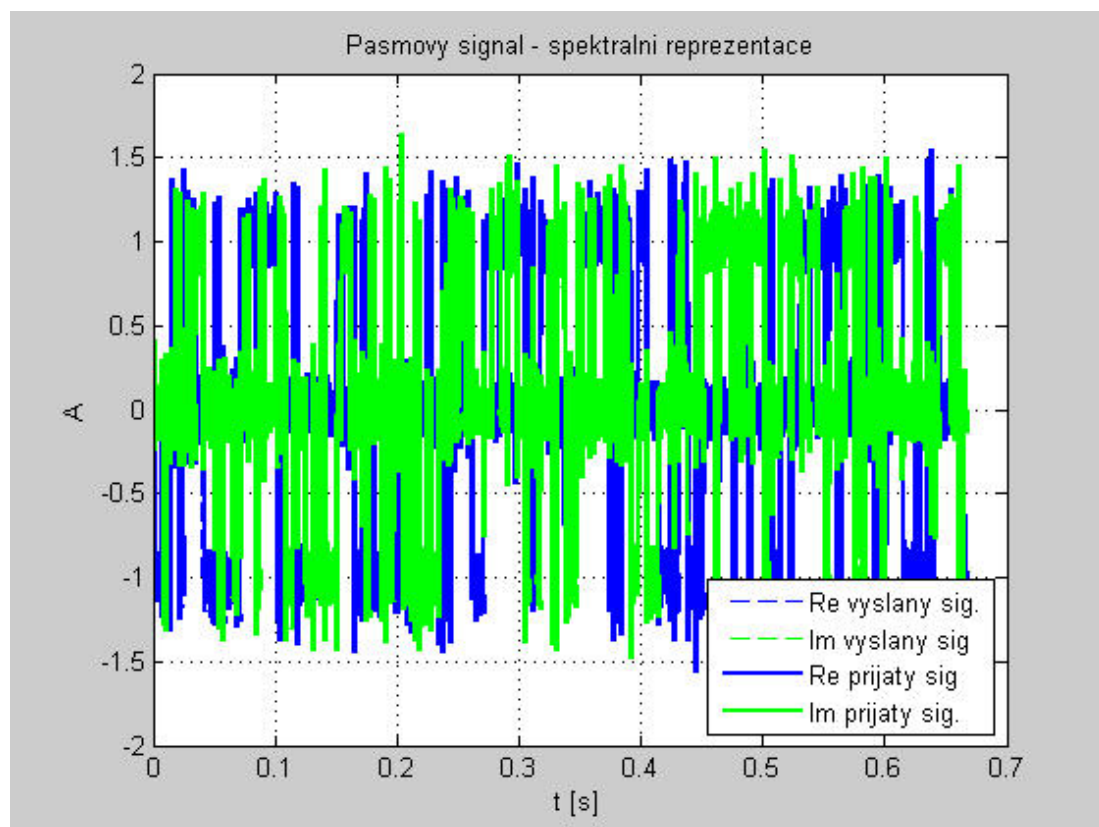
% Vykresleni komplexni obalky v casove domene v porovnani s tim, co bylo
% vyslano
figure(9)
plot(t,real(s),'b--')
hold on
plot(t,imag(s),'g--')
hold on
plot(t,real(signal),'LineWidth',2)
%stem(t,real(s),'b*')
hold on
plot(t,imag(signal),'g-', 'LineWidth',2)
%stem(t,imag(s),'g*')
grid on
title('Pasmovy signal - spektralni reprezentace');
legend('Re{} vyslany sig.', 'Im{} vyslany sig', 'Re{} prijaty sig', 'Im{} prijaty sig.',4);
xlabel('t [s]');
ylabel('A');

% figure(111)
% plot(freco, fftshift(abs(fft(s_a_f))));
% figure(112)
```



```
% plot(frezo, fftshift(abs(fft(lj.*s_c_f))));
```





Published with MATLAB® 7.4