



Αναφορά 1<sup>ης</sup> Εργαστηριακής Άσκησης στις Ψηφιακές Επικοινωνίες II

«ADSL DMT»

Μπουφίδης Ιωάννης 03120162

Ο παρακάτω συμπληρωμένος κώδικας αποτελεί προσομοίωση ενός ζεύγους πομπού-δέκτη για αποστολή και λήψη DMT συμβόλων με διαμόρφωση υποδιαύλου 16-QAM για κάθε υποδιαύλο του εύρους ζώνης.

```
%%%%%%%% Simulation of a DMT Tx-RX System over an AWGN line %%%%%%%%%%
%%----- winter semester 2023 -----%%

clear all; close all;

%% System parameters
K=255; % number of subcarriers
Tu=1/4000; % DMT symbol duration - 4000 frames per second
Fi=4312.5; % subcarrier spacing
d=0; % guard interval portion -- zero in this simulation
delta=d*Tu; % guard interval duration
Ts=Tu+delta; % actual symbol duration
Ks=2*(K+1); % FFT size
Fs=Ks*Fi; % Nyquist frequency
T=1/Fs; % time-sampling period

% Gain vector
% It contains the gain for each subcarrier (zero means "don't use")
G=[0 ones(1,K-1) 0];
for i=1:31 G(i)=0; end % voice and uplink band -- not used
G(64)=0; % sub-carrier 64 will be used as pilot
for i=180:200 G(i)=0; end % very noisy band -- not used
for i=201:230 G(i)=1.5; end % higher gain band
for i=231:256 G(i)=0; end % very noisy band -- not used
used_carriers=sum(G~=zeros(1,256));

% other parameters
M = 16; % Size of signal constellation per subcarrier - same for all
mb = log2(M); % Number of bits per subcarrier
Ns = 500; % Number of DMT symbols to process.
% Use a small value of Ns (e.g. 10) for demonstration
% and a large value (e.g. 500) for BER estimation
n_bits=Ns*mb*used_carriers; % number of bits to generate
Rate=mb*used_carriers/Tu; % actual transmission rate, in bps
nsamp = 16; % Oversampling rate, for analog signal simulation
Tn=T/nsamp; % oversampling period
SNR=5; % signal-to-noise-ratio, in db
%% Input bit stream
% Create a binary data stream as a vector.
x = randi([0 1], 1, n_bits); % Random binary data stream
```

```

%% QAM constellation encoding
% A. Define a vector for mapping bits to symbols using
% orthogonal full-grid constellation and Gray coding.
% The vector corresponds to 16-QAM constellation.
mapping = [0 1 3 2 4 5 7 6 12 13 15 14 8 9 11 10].';

% B. Do ordinary binary-to-decimal mapping.
xsym=bi2de(reshape(x,mb,length(x)/mb).','left-msb');

% C. Mapping to QAM constellation
xsym = mapping(xsym+1);
y=qammod(xsym,M)';
% show a scatterplot of the QAM-encoded data
scatterplot(y);
%% Exercise # 1
%-----
% Display the 16-QAM constellation with the associated
% 4-bit codes shown next to each point of the constellation
% ANSWER
qam_mapping = qammod(mapping, M);
for k = 1 : length(qam_mapping)
    text(real(qam_mapping(k)), imag(qam_mapping(k)), num2str( ...
        de2bi(mapping(k), log2(M), 'left-msb')), 'FontSize', 6, 'Color', 'white');
end
title('16-QAM Constellation');
xlabel('real');
ylabel('imaginary');
pause
%-----

% D. Compute weights over the entire spectrum
power=y*y'/length(y); amp=sqrt(power);
k=1;
y_mat=[];
for n=1:Ns
    z=zeros(2*(K+1),1);
    for i=1:length(G)
        if G(i)~=0
            z(i)=G(i)*y(k);
            z(2*(K+1)-i+1)=z(i)';
            k=k+1;
        end
    end
    z(64)=1.5*amp; z(2*(K+1)-63)=z(64); % pilot
    y_mat=[y_mat z];
end
% Stem Plot of a DMT symbol (amplitudes of complex values)
figure;
stem(abs(y_mat(:,1)));
title('carrier modulation within a DMT symbol');
xlabel('carrier index'); ylabel('amplitude');
pause

% Compute IFFT to get DMT time symbols
s=2*(K+1)*ifft(y_mat, 'symmetric');
figure; pwelch(reshape(s,Ns*2*(K+1),1),[],[],[],Fs);
pause

%%
% Upsampling by using the MATLAB interp function.
% This is needed for the simulation of the analog signal
for n=1:length(s(1,:))
    s_up(:,n)=interp(s(:,n),nsamp);
end

```

```

% plot spectrum of upsampled signal and explain the high freq spectral images
figure; pwelch(reshape(s_up,Ns*2*(K+1)*nsamp,1),[],[],[],Fs);
Fs1=Fs*nsamp; Ks1=Ks*nsamp; % the sampling frequency is changed accordingly
pause

%% add noise
sr=awgn(s_up,SNR,'measured');
figure; pwelch(reshape(sr,Ns*2*(K+1)*nsamp,1),[],[],[],Fs);
pause
clear s_up;

%% Filter out the out-of-band noise and downsample the signal
% filter-out the bandpass components
% LP (Parks-McClellan) filtering
order=16*nsamp;
f1=1.1*Fi*(K+1)/Fs1; f2=1.3*f1;
fpts=[0 2*[f1 f2] 1];
mag=[1 1 0 0]; wt=[1 1];
b = firpm(order,fpts,mag,wt);
a=1;
for i=1:length(sr(1,:))
    dummy=[sr(:,i);zeros(order,1)];
    dummy1=filter(b,a,dummy);
    delay=order/2;
    sr_up(:,i)=dummy1(delay+(1:length(sr(:,1)))));
end
figure; pwelch(reshape(sr_up,Ns*2*(K+1)*nsamp,1),[],[],[],Fs);
pause
sr_down=downsample(sr_up,nsamp);
figure; pwelch(reshape(sr_down,Ns*2*(K+1),1),[],[],[],Fs);
% sr_down=downsample(sr,nsamp); %without filtering
clear sr sr_up;

%% Delineate DMT symbols and perform FFT to get the QAM symbols
yr_mat=1/Ks*fft(sr_down, Ks);
% keep only the payload carriers
yr=[];
for n=1:length(yr_mat(1,:))
    for i=1:length(G)
        if G(i)~=0
            yr=[yr yr_mat(i,n)/G(i)];
        end
    end
end
scatterplot(yr);

%% QAM decoding and error counting
%-----
% Exercise # 2
% Complete the code to implement QAM decoding and error counting.
% For each element of the received vector yr (complex value)
% find the nearest point on the 16-QAM constellation
% Compare the results with the transmitted vector y,
% count the errors and compute the BER
%-----
% ANSWER
%% Error counting
L=sqrt(M);
q=[-L+1:2:L-1];
errors=0;
for n=1:length(yr)
    [m,j]=min(abs(q-real(yr(n)))));
    y_real=q(j);
    [m,j]=min(abs(q-imag(yr(n)))));
    y_imag=q(j);

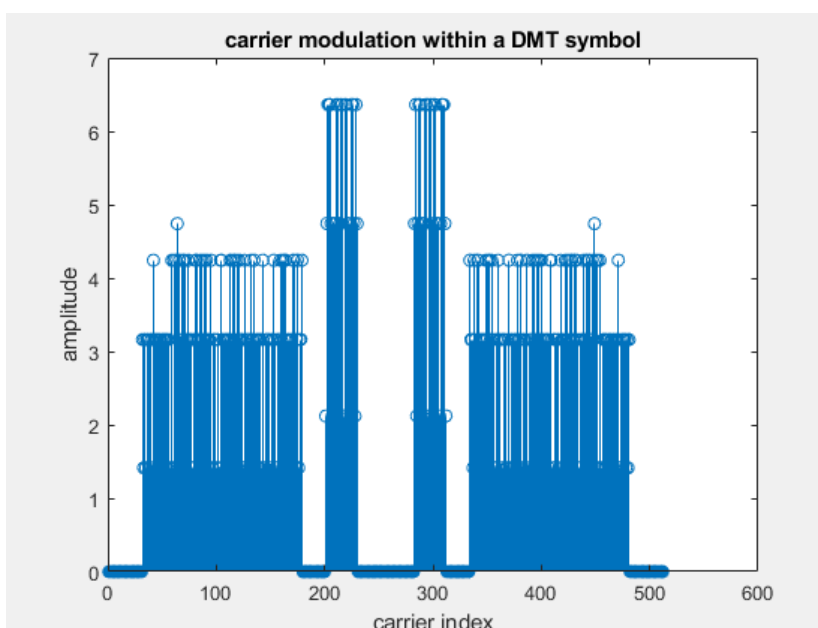
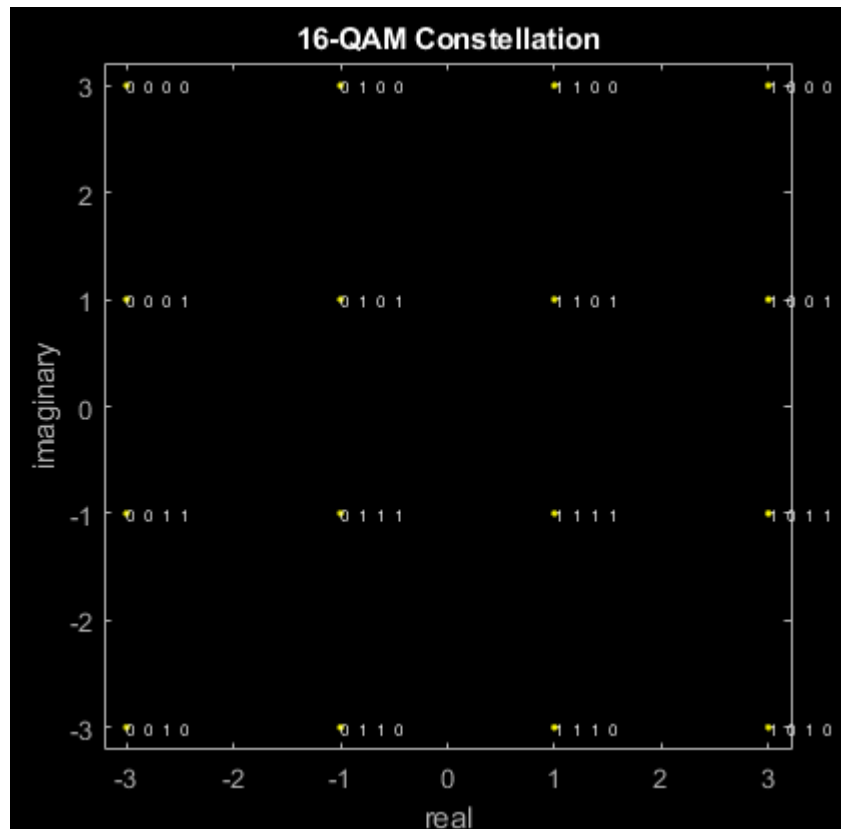
```

```

if not(y(n)==(y_real+1j*y_imag))
    errors=errors+1;
end
end
qam_ber = errors/(n_bits/mb);
dmt_ser = 1-(1-qam_ber)^used_carriers;
fprintf('The BER of 16-QAM is: %f\n', qam_ber);
fprintf('The Error Rate of DMT Symbols is: %f\n', dmt_ser);
%-----

```

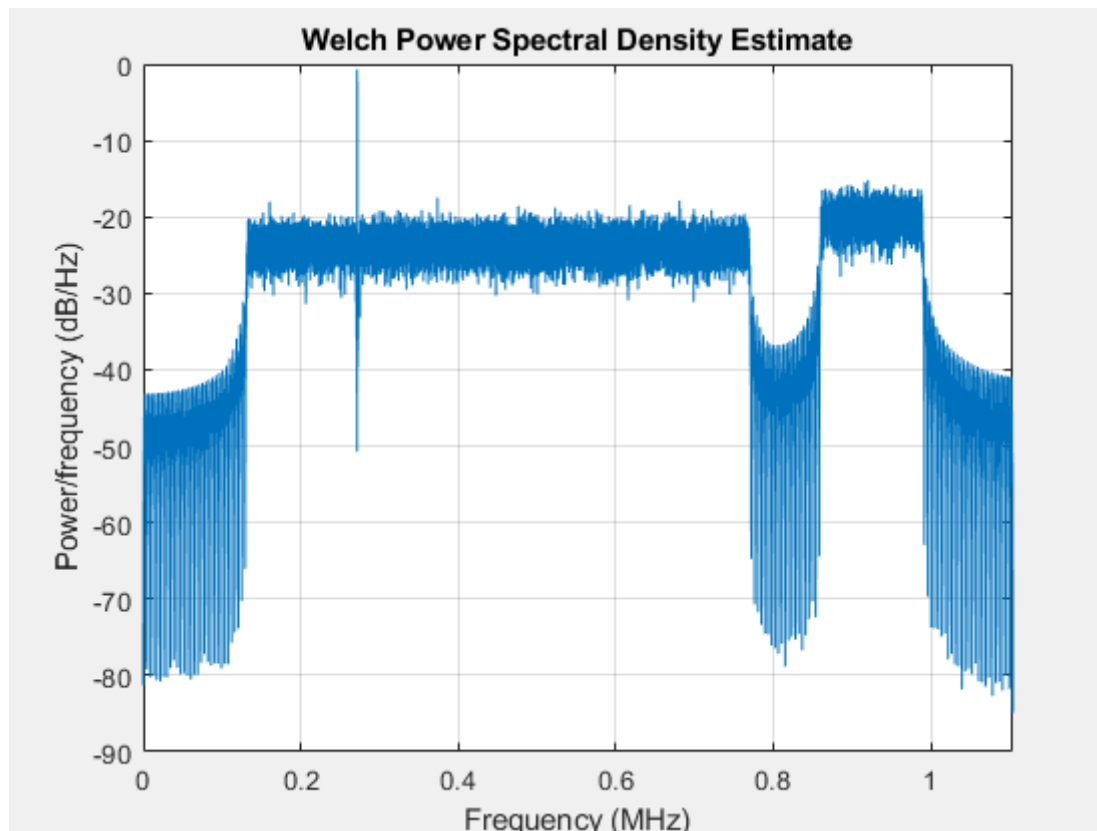
Μέσω του παραπάνω κώδικα σχεδιάζεται ο σηµατικός αστερισµός 16-QAM (όπως φαίνεται στο διπλανό σχήµα) πλήρους ορθογωνικού πλέγµατος, µε σηµειωµένες τις δυαδικές λέξεις δίπλα σε κάθε σηµείο του µε κωδικοποίηση Gray.



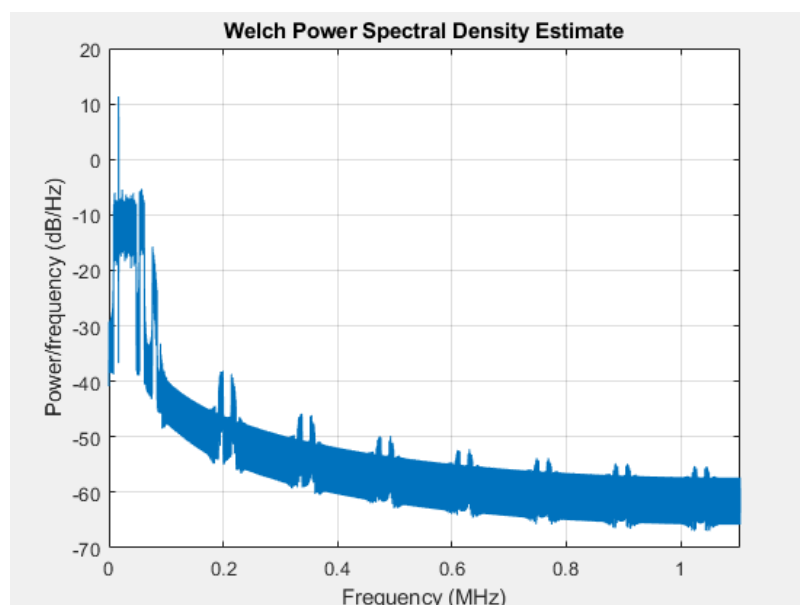
Αφού έχουµε επιλέξει ποιους υποδιαύλους θέλουµε να χρησιµοποιήσουµε και ποιους όχι (θέτοντας '1' και '0' αντίστοιχα στον πίνακα G) και έχουµε υπολογίσει το amplitude του εκπεµπόµενου συµβόλου µετά την διαµόρφωση, µπορούµε να υπολογίσουµε το amplitude κάθε επιµέρους υποδιαύλου ανάλογα µε το gain του, όπως φαίνεται στο διπλανό διάγραµµα (εδώ, two sided spectrum).

Στην συνέχεια ο κώδικας παράγει το φάσμα ισχύος του σήματος (όπως φαίνεται στο παρακάτω σχήμα), το οποίο αντιστοιχεί στο προηγούμενο διάγραμμα. Συγκεκριμένα, παρατηρούμε ξανά ότι η δεύτερη ζώνη συχνοτήτων έχει amplitude μεγαλύτερο απ' την πρώτη, ενώ η δεύτερη ζώνη (η οποία ξεκινάει στην 200<sup>η</sup> περίπου carrier) ξεκινάει σωστά λίγο μετά τα 0.85 MHz, καθώς το εύρος ζώνης κάθε υποδιαύλου ισούται με 4312.5 Hz ( $200 * 4312.5 = 862500 \text{ Hz}$ ).

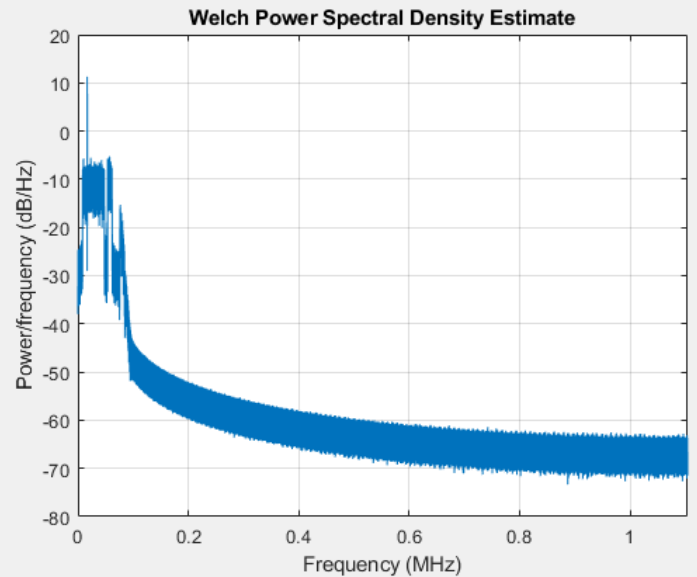
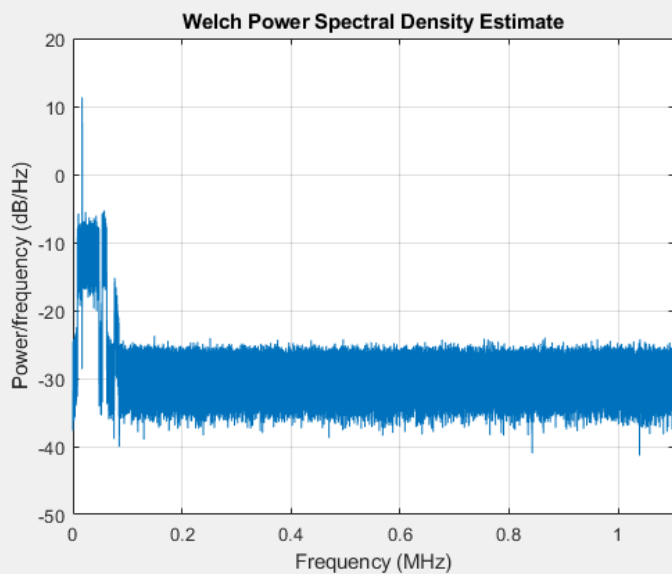
Στον 64<sup>ο</sup> υποδιαύλο ( $\sim 0.27 \text{ MHz}$ ) παρατηρούμε ένα spike. Αυτό συμβαίνει, καθώς προκειμένου να χρησιμοποιήσουμε αυτόν τον υποδιαύλο σαν pilot, έχουμε κάνει την gained τιμή του ίση με 1.5 φορά το amplitude του εκπεμπόμενου συμβόλου.



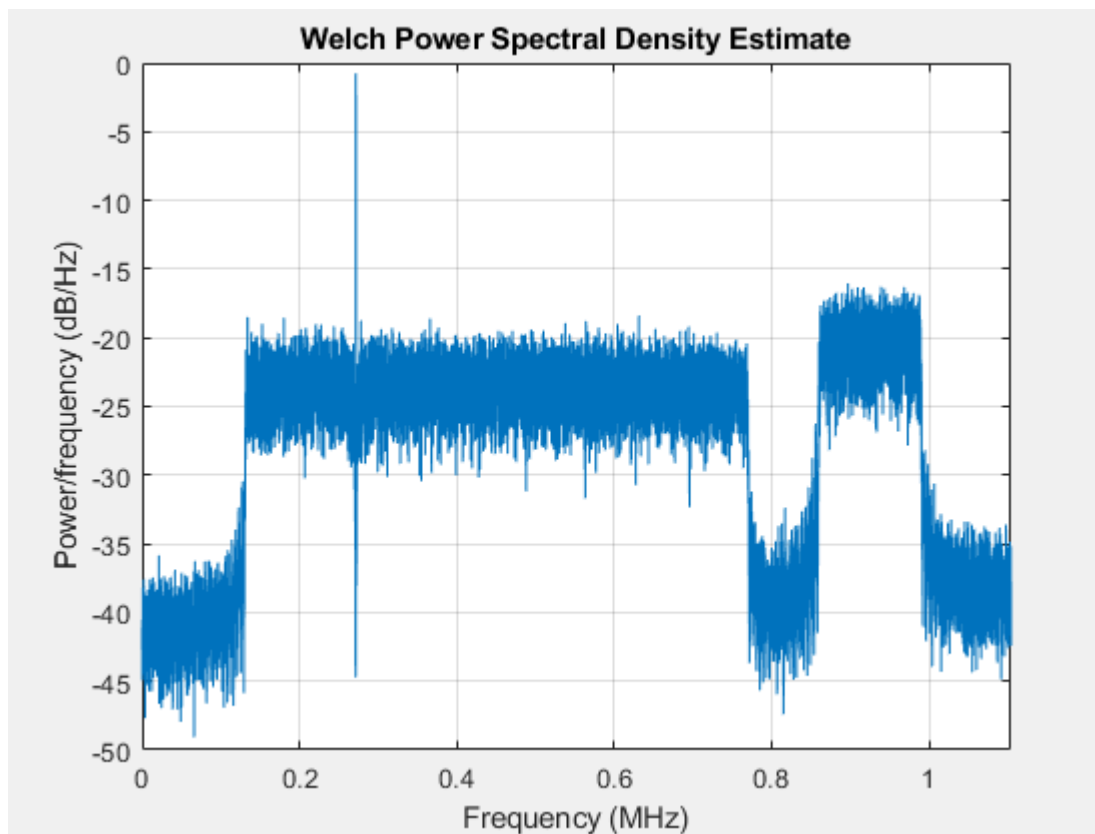
Στην συνέχεια, κατά την υπερδειγμάτιση του σήματος, προστίθενται δείγματα μεταξύ των υπάρχοντων δειγμάτων του σήματος, με αποτέλεσμα να αυξάνεται η συχνότητα δειγματοληψίας κατά **nsamp** φορές. Αυτή η αύξηση της δειγματοληψίας, ωστόσο, εκτείνει το φάσμα του σήματος σε υψηλότερες συχνότητες, γεγονός που προκαλεί και τις «εικόνες» του φάσματος του σήματος σε μεγαλύτερες συχνότητες που παρατηρούνται στο διπλανό σχήμα.



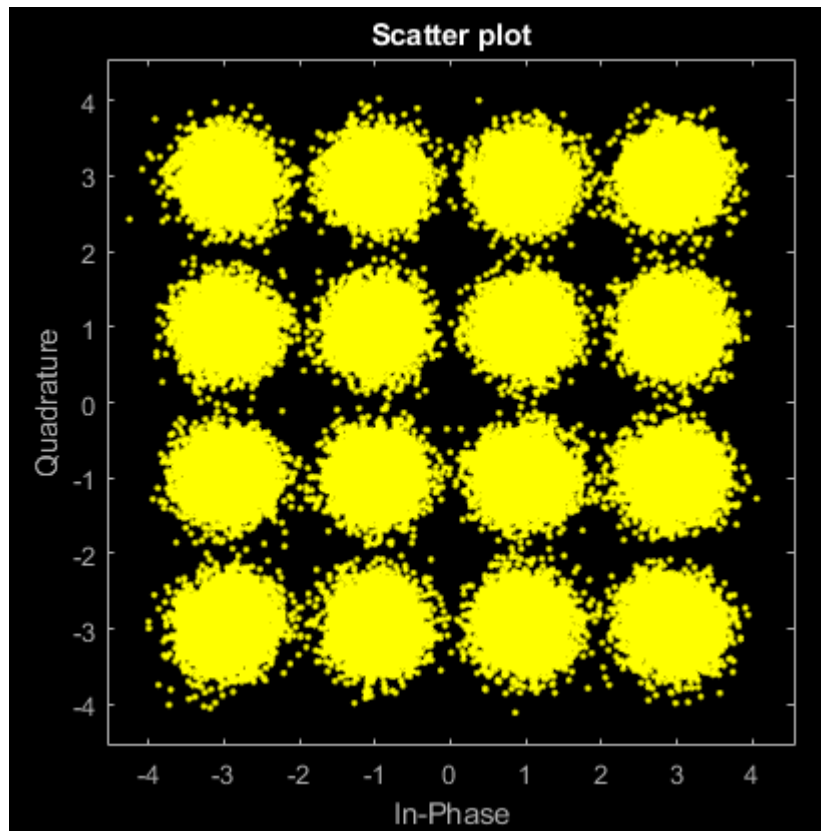
Αφού περάσουμε το σήμα από ένα θορυβώδες (SNR=5) κανάλι (αριστερό σχήμα), το φιλτράρουμε με ένα βαθυπερατό Parks-McClellan φίλτρο (δεξί σχήμα), με σκοπό να αποκόψουμε τις «εικόνες» στις υψηλές συχνότητες που παρουσιάστηκαν λόγω της υπερδειγμάτισης.



Μετά το φιλτράρισμα, υποδειγματίζουμε το σήμα, με αποτέλεσμα να λάβουμε ένα σήμα με παρόμοιο φάσμα με αυτό που έστειλε ο πομπός (κάτω σχήμα), πράγμα που είναι αναμενόμενο.



Αφού εφαρμόσουμε FFT στο ληφθέν σήμα και λάβουμε τα QAM σύμβολα που το αποτελούν, είμαστε σε θέση να δημιουργήσουμε το διάγραμμα διασκορπισμού του ληφθέντος σήματος, όπως φαίνεται στο παρακάτω σχήμα.



Τελικά, αφού αποκωδικοποιήσουμε τα σύμβολα που λάβαμε, μπορούμε να υπολογίσουμε το Bit Error Rate (BER) της 16-QAM διαμόρφωσης, αλλά και την πιθανότητα λήψης εσφαλμένου DMT συμβόλου μέσω της σχέσης

$$1 - (1 - P_b)^{used\_carriers}, \text{ όπου } P_b \text{ το BER της 16-QAM}$$

Η σχέση αυτή προκύπτει από το γεγονός, ότι η  $(1 - P_b)^{used\_carriers}$  ισούται με την πιθανότητα κανένα σύμβολο 16-QAM που μεταδόθηκε (μέσω οποιουδήποτε από τους χρησιμοποιούμενους υποδιαύλους) να αποκωδικοποιήθηκε εσφαλμένα.

The BER of 16-QAM is: 0.000667

The Error Rate of DMT Symbols is: 0.111339

Στην προκειμένη προσομοίωση, αυτά τα δύο μεγέθη λαμβάνουν τις τιμές που φαίνονται στα διπλανά μηνύματα.