

# Ψηφιακές Επικοινωνίες

## 5<sup>η</sup> Εργαστηριακή Άσκηση

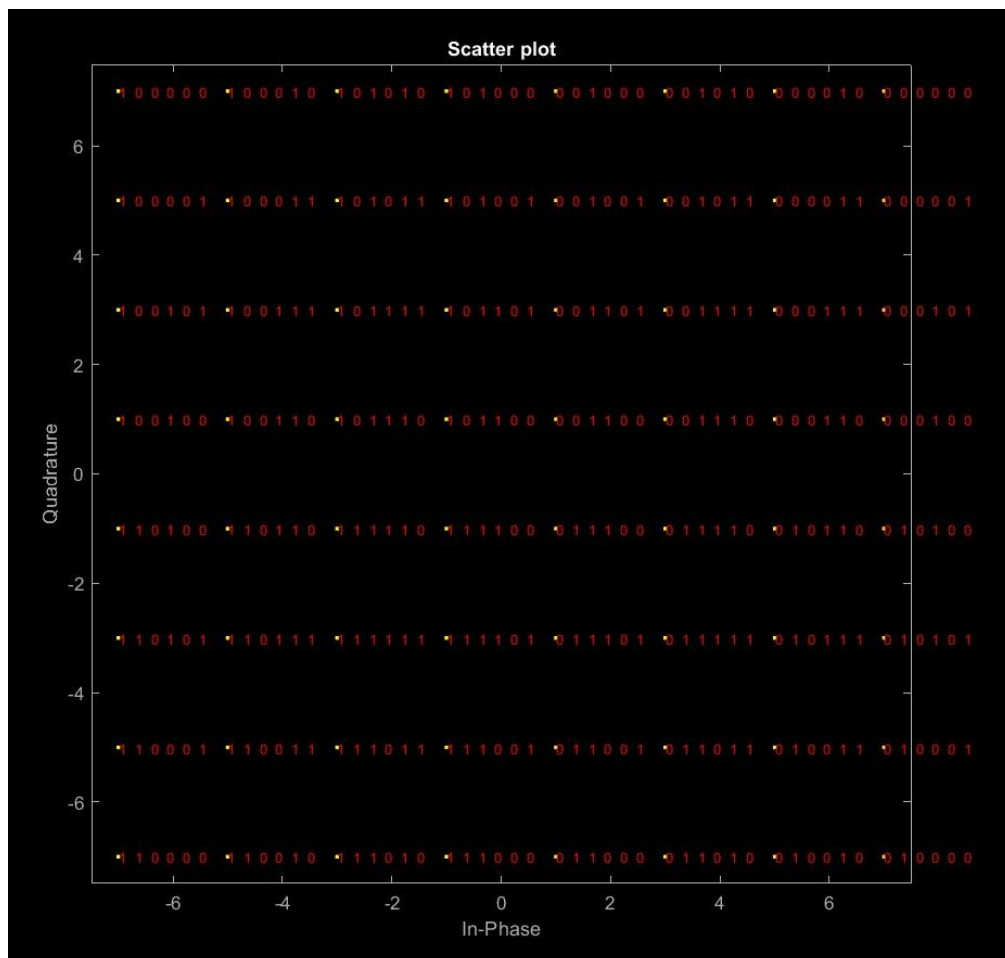
*Άννα Κουτσώνη-03120019*

### Μέρος 1<sup>ο</sup>

Χρησιμοποιούμε τον εξής κώδικα:

```
1 close all;
2 clear all;
3 L=8;
4 l=log2(L);
5 k=2^l;
6 core=[1+1i;1-1i;-1+1i;-1-1i];
7 mapping=core;
8 if(l>1)
9     for j=1:l-1
10         mapping=mapping+j*2*core(1);
11         mapping=[mapping;conj(mapping)];
12         mapping=[mapping;-conj(mapping)];
13     end
14 end
15 scatterplot(mapping);
16 for i=1:length(mapping)
17     text(real(mapping(i)),imag(mapping(i)),num2str(de2bi(i-1,k,'left-msb')),'Color','red','FontSize',6);
18 end
19
```

Το διάγραμμα που προκύπτει είναι το εξής:



## Μέρος 2<sup>ο</sup>

W=2.5 MHz , R=12Mbps

$$\log_2 M \geq \frac{R}{W} (1 + a) \quad , 0 < a \leq 1$$

$$\log_2 M \geq 4.8(1 + a)$$

Επιλέγουμε το μικρότερο M για το οποίο ισχύει η παραπάνω συνθήκη: M=64 και άρα το μεγαλύτερο a για αυτήν την τιμή α=0.25 ώστε να εκμεταλλευτούμε όλο το διαθέσιμο εύρος ζώνης. Για το baud rate:  $\frac{1}{T} = \frac{W}{1+a} = 2$ . Ορίζουμε fc= 10 MHz και nsamp=32 δείγματα ανά T (Fs=32/T=64MHz) και με τις τιμές αυτές η υπερδειγματοληψία είναι επαρκής για να μην υπάρχει aliasing στο ζωνοπερατό σήμα.

Για την εξομοίωση με bertool χρησιμοποιούμε για τον υπολογισμό των λανθασμένων συμβόλων την qam\_errors την οποία καλούμε μέσω της qam\_nyq\_band\_ber\_func.

qam\_nyq\_band\_ber\_func:

```
1 function [ber,numBits] = qam_nyq_band_ber_func(EbNo, maxNumErrs, maxNumBits)
2 import com.mathworks.toolbox.comm.BERTool.*;
3 totErr = 0; % Number of errors observed
4 numBits = 0; % Number of bits processed
5 k=6; % number of bits per symbol
6 Nsymb=10000; % number of symbols in each run
7 nsamp=32; % oversampling,i.e. number of samples per T
8 while((totErr < maxNumErrs) & (numBits < maxNumBits))
9     errors=qam_errors(k,Nsymb,nsamp,EbNo);
10    totErr=totErr+errors;
11    numBits=numBits + k*Nsymb;
12 end % End of loop
13 ber = totErr/numBits;
14 end
```

qam\_errors:

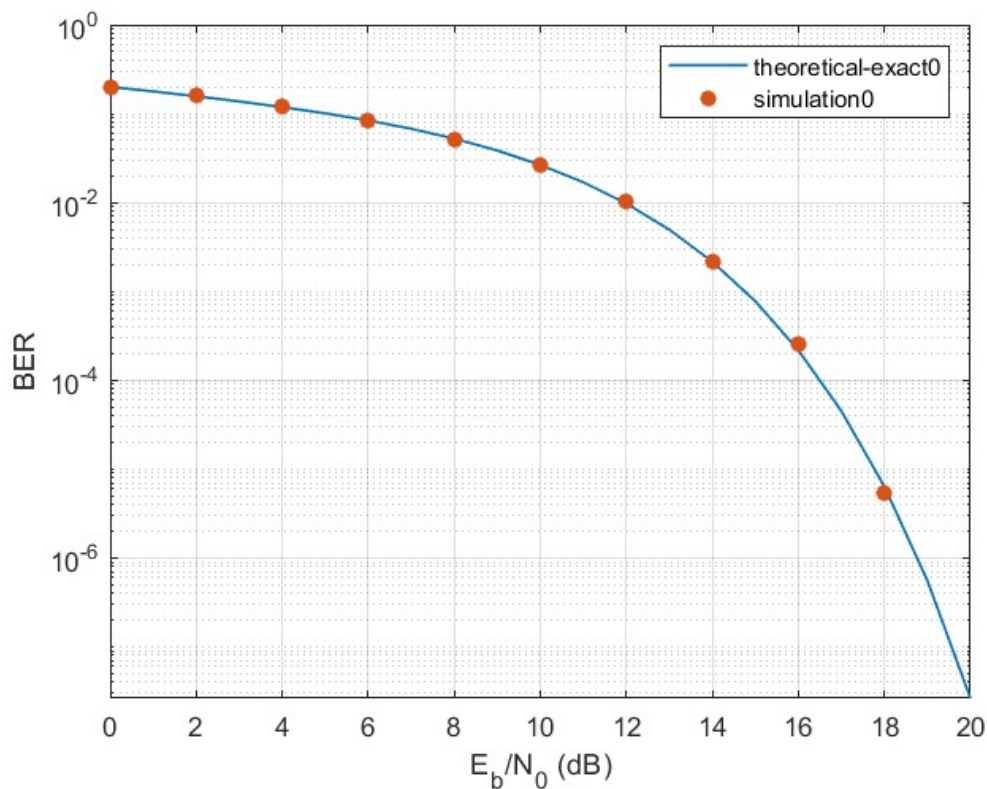
```
1 function errors=qam_errors(k,Nsymb,nsamp,EbNo)
2 M=2^k; % LxL σηματοδότης, L=2^1, 1>0
3 L=sqrt(M);
4 l=log2(L);
5 fc=10; % συχνότητα φέροντος, πολλαπλάσιο του Baud Rate (1/T)
6 SNR=EbNo*10*log10(nsamp/k/2); % SNR ανά δείγμα σήματος
7 core=[1+1i;1-1i;-1+1i;-1-1i];
8 mapping=core;
9 if(l>1)
10 for j=1:l-1
11 mapping=mapping+j*2*core(1);
12 mapping=[mapping;conj(mapping)];
13 mapping=[mapping;-conj(mapping)];
14 end
15 end
16 x=floor(2*rand(k*Nsymb,1)); % τυχαία δυαδική ακολουθία
17 xsym=bi2de(reshape(x,k,length(x)/k).','left-msb');
18 y=[];
19 for n=1:length(xsym)
20 y=[y mapping(xsym(n)+1)];
21 end
22 delay = 8; % Group delay (# περιόδων T)
23 filterorder = delay*nsamp*2;
24 rolloff = 0.25; % συντελεστής εξαπλώσης φίλτρου
25 shaping_filter = rcosdesign(rolloff,2*delay,nsamp,'sqrt');
26 ytx=upsample(y,nsamp);
27 ytx = conv(ytx,shaping_filter);
28 %figure(1); pwelch(real(ytx),[],[],[],nsamp); % σε κλίμακα 1/T
29 m=(1:length(ytx));
```

```

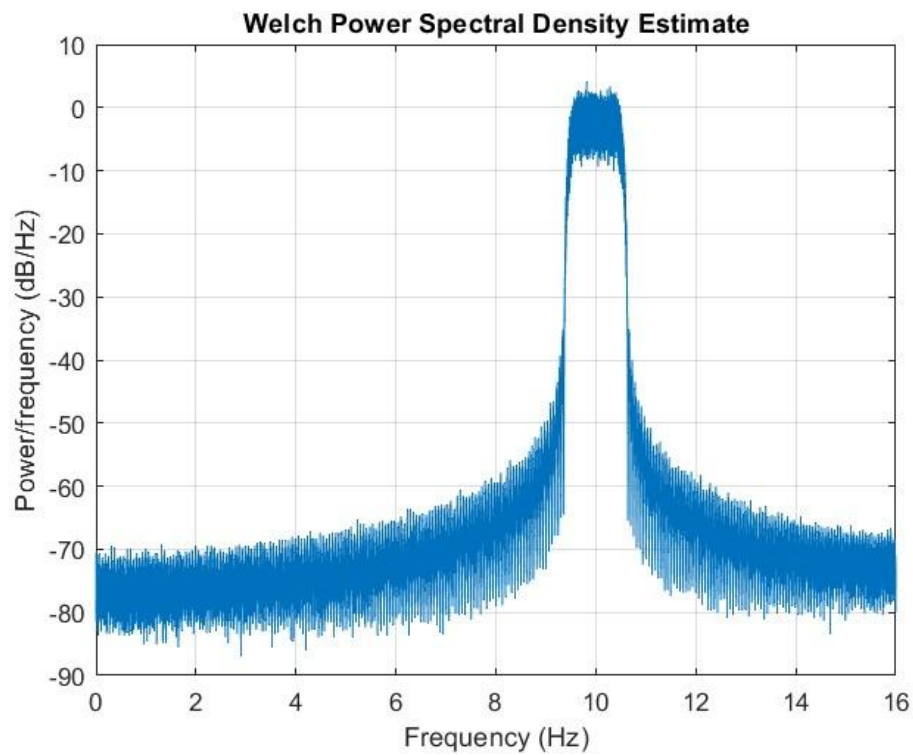
30 s=real(ytx.*exp(1j*2*pi*fc*m/nsamp));
31 %figure(2); pwelch(s,[],[],[],nsamp); % σε κλίμακα 1/T
32 Ps=10*log10(s*s'/length(s)); % ισχύς σήματος, σε db
33 Pn=Ps-SNR; % αντίστοιχη ισχύς θορύβου, σε db
34 n=sqrt(10^(Pn/10))*randn(1,length(ytx));
35 snoisy=s+n; % θορυβώδες ζωνοπερατό σήμα
36 clear ytx xsym s n; % για εξοικονόμηση μνήμης
37 yrx=2*snoisy.*exp(-1j*2*pi*fc*m/nsamp); clear s;
38 yrx = conv(yrx,shaping_filter);
39 %figure(3); pwelch(real(yrx),[],[],[],nsamp); % κλίμακα 1/T
40 yrx = downsample(yrx,nsamp); % Υποδειγμάτιση στο πλέγμα nT.
41 yrx = yrx(2*delay+(1:length(y))); % περικοπή άκρων συνέλιξης.
42 yi=real(yrx); yq=imag(yrx); % συμφασική και εγκάρσια συνιστώσα
43 xrx=[]; % διάνυσμα δυαδικής ακολουθίας εξόδου -- αρχικά κενό
44 q=-L+1:2:L-1;
45 for n=1:length(yrx) % επιλογή πλησιέστερου σημείου
46 [m,j]=min(abs(q-yi(n)));
47 yi(n)=q(j);
48 [m,j]=min(abs(q-yq(n)));
49 yq(n)=q(j);
50 m=m+1;
51 while (mapping(m)~=yi(n)+1i*yq(n)) m=m+1; end
52 xrx=[xrx; de2bi(m-1,k,'left-msb')'];
53 end
54 %scatterplot(yrx); % διάγραμμα διασκορπισμού
55 ber1=sum(not(y==yi+1i*yq));
56 ber2=sum(not(xrx==x));
57 errors=ber2;
58 end
59

```

Προκύπτει το εξής γράφημα από την προσομοίωση:

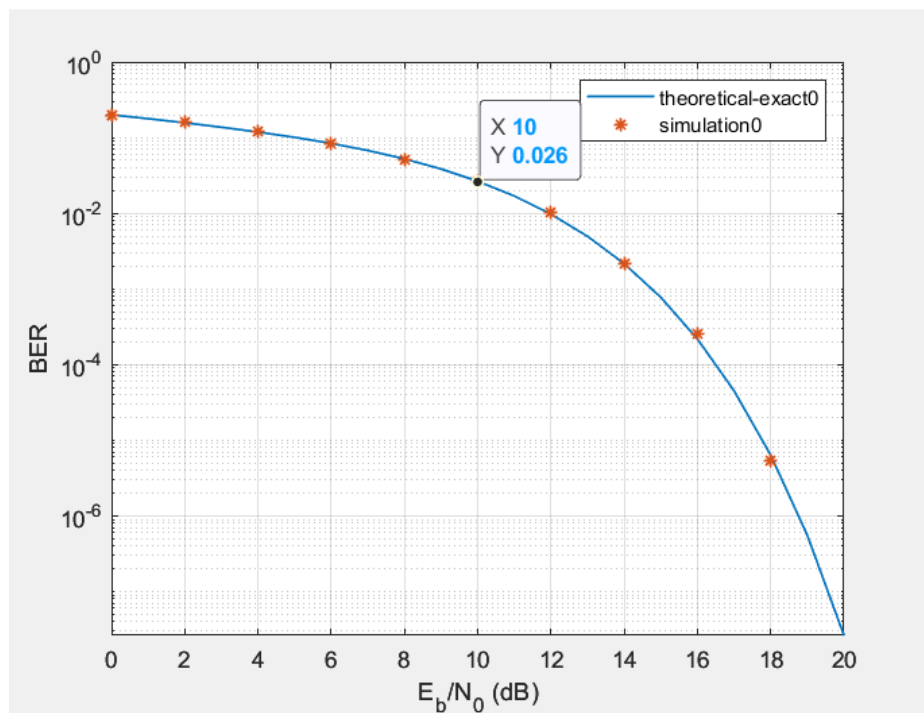


Και μέσω της pwelch το φάσμα του ζωνοπερατού σήματος:

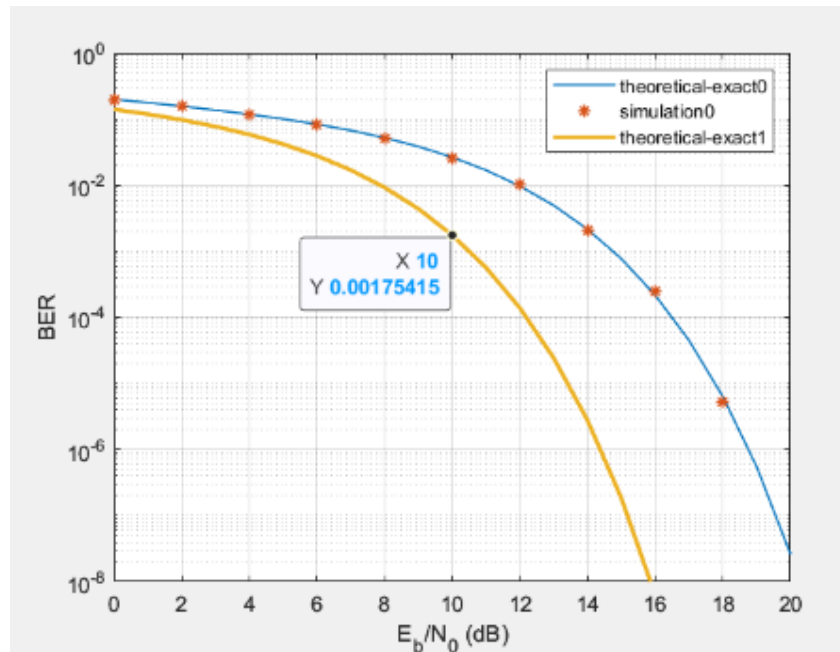


### Μέρος 3°

Για την 64-QAM παρατηρούμε στο διάγραμμα ότι για  $E_b/N_0=10$  db το BER υπερβαίνει την τιμή 0.002:



Σχεδιάζουμε την 16-QAM και παρατηρούμε ότι για  $E_b/N_0=10\text{db}$  ικανοποιείται η συνθήκη  $\text{BER}<0.002$ :

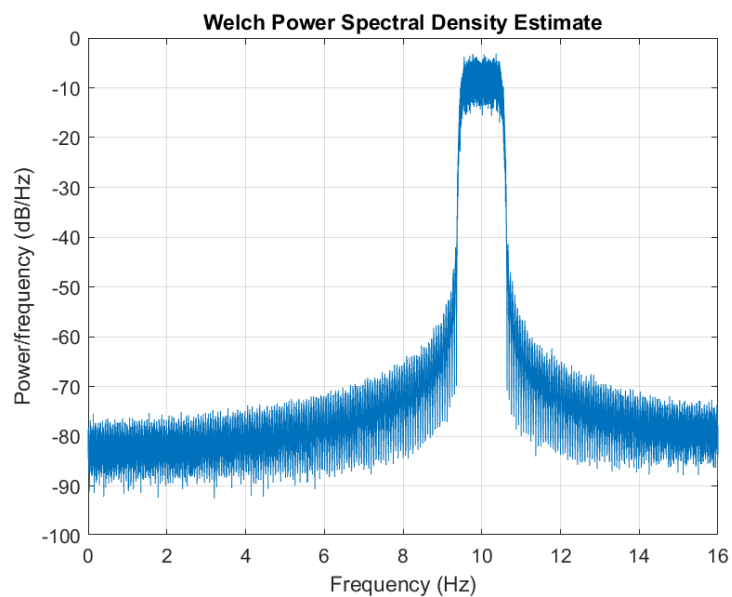


Για τον νέο μέγιστο ρυθμό μετάδοσης  $R'$ :

$$R' = \frac{1}{T} \log_2 16 \text{ και } R = \frac{1}{T} \log_2 64 \Rightarrow$$

$$\frac{R}{R'} = \frac{\log_2 64}{\log_2 16} = \frac{6}{4} \Rightarrow R' = \frac{2}{3} * R = \frac{2}{3} * 12 = 8 \text{ Mbps}$$

Σχεδιάζουμε μέσω της pwelch το φάσμα του ζωνοπερατού σήματος:



Παρατηρούμε ότι δεν υπάρχει διαφοροποίηση διότι τα χαρακτηριστικά που επηρεάζουν το φάσμα δεν αλλάζουν ( $\alpha, T, W$ ).

#### **Μέρος 4<sup>ο</sup>**

Μειώνοντας το rolloff στο μισό για  $\alpha=0.125$  θα έχουμε:

$$R = \frac{1}{T} \log_2 16 = \frac{W}{1 + \alpha} \log_2 16 = \frac{2.5}{1.125} * 4 = 8.889 Mbps \text{ ο μέγιστος ρυθμός μετάδοσης}$$