

Άσκηση 2

Ψηφιακές Τηλεπικοινωνίες

Συμεωνίδης Θεόδωρος 1064870

Χειμερινό Εξάμηνο, Ακαδημαϊκό έτος 2020-21

Περιεχόμενα

1	Ερώτημα 1	3
2	Ερώτημα 2	4
3	Ερώτημα 3	5
4	Ερώτημα 4	7
5	Κώδικας	8

Ο κώδικας που γράφτηκε έχει επικολληθεί τέλος της εργασίας.

1 Ερώτημα 1

1. Με βάση τις παραπάνω υποδείξεις, υλοποιήστε το σύστημα M-PSK και αναφερθείτε στα βασικά του σημεία.

Απάντηση στο Ερώτημα 1

Το M-PSK όπως και η υλοποίηση του δεν διαφέρει ιδιαίτερα από αυτή του M-PAM που υλοποιήσαμε στην προηγούμενη εργασία. Η κύρια διαφορά είναι στο χώρο σημάτων που χρησιμοποιεί για να γίνει η αναπαράσταση του κάθε συμβόλου. Στη περίπτωση του M-PAM ήταν μονοδιάστατος, τώρα είναι διδιάστατος με βάση τα σήματα $\cos(2\pi f_c t)$ και $-\sin(2\pi f_c t)$. Ο αστερισμός που έχουμε επιλέξει ξεκινάει από το σημείο $s_1 = \cos(2\pi f_c t)$ και τα υπόλοιπα σημεία προκύπτουν ως $s_i = \cos(2\pi f_c t + 2\pi \frac{i-1}{M})$ δηλαδή είναι M ισαπέχοντα σημεία που βρίσκονται πάνω στο μοναδιαίο κύκλο και άρα έχουν μοναδιαίες ενέργειες s .

Αναφερόμαστε συνοπτικά στα επιμέρους συστατικά της υλοποίησης:

- Mapper
Παίρνει ως είσοδο ένα διάνυσμα από bits μεγέθους $K \times 1$, το μετατρέπει σε ένα πίνακα $\frac{K}{M} \times M$ και στη συνέχεια σε κάθε γραμμή έχουμε ένα δυαδικό αριθμό τον οποίο τον μετατρέπουμε σε δεκαδικό. Στη περίπτωση της κωδικοποίησης Gray σε αυτό το σημείο μετατρέπουμε τα ακέραια στοιχεία σε κωδικοποιημένα κατά Gray στοιχεία.
- Διαμορφωτή
Κάθε σύμβολο αντιστοιχίζεται στη κατάλληλη κυματομορφή που αντιστοιχεί σε αυτό το σύμβολο s_m του αστερισμού. Σε ένα διάνυσμα όλες αυτές οι τιμές είναι το διάνυσμα εξόδου από τον πομπό.
- Κανάλι με AWGN
Στο σημείο αυτό υπολογίζεται ο θόρυβος και προστίθεται στο σήμα εξόδου σύμφωνα με την εκφώνηση.
- Αποδιαμορφωτής
Λαμβάνει ως είσοδο το διάνυσμα του σήματος από το κανάλι. Αυτό το σήμα συνήθως είναι μεγαλύτερης διάστασης από τον χώρο σημάτων μας. Οπότε το προβάλει στη βάση σημάτων μας. Αυτό γίνεται υπολογίζοντας το εσωτερικό του γινόμενο με κάθε διάνυσμα βάσης $r_j = \langle r, \phi_j \rangle$, το οποίο είναι το μέτρο της προβολής του σε κάθε συνιστώσα του χώρου μας ή αλλιώς οι συντεταγμένες του διανύσματος \bar{r} όταν το προβάλουμε στη βάση $\phi_1, \phi_2, \dots, \phi_j$. Έτσι ως εξόδο για κάθε σύμβολο έχουμε ένα διδιάστατο διάνυσμα.

- Φωρατής
Το σήμα που αποδιαμορφώθηκε ανήκει στο χώρο σημάτων που έχουμε επιλέξει για τη διαμόρφωση. Ο φωρατής υπολογίζει την Ευκλείδεια απόσταση του από κάθε σημείο που ανήκει στον αστερισμό. Αντιστοιχίζει το σημείο μας στο σημείο που έχει την ελάχιστη απόσταση από αυτό, γιατί αυτό μεγιστοποιεί την υπό συνθήκη πιθανότητα $P(S_i|s_j)$, δεδομένου ότι στάλθηκε s_j να αποκωδικοποιήσω σε s_i της Γκαουσιανής στοχαστικής διαδικασίας.
- Demapper
Αντιστρέφει τη λειτουργία του Mapper. Στην περίπτωση της κωδικοποίησης κατά Gray εκτελεί την επιπλέον πράξη της αποκωδικοποίησης του διανύσματος των ακεραίων. Τέλος, αντιστοιχίζει κάθε δεκαδικό σύμβολο στα bits στα bits της δυαδικής του αναπαράστασης και φτιάχνει ένα διάνυσμα με τα bits των συμβόλων.

2 Ερώτημα 2

Για καθένα από τα δύο συστήματα, μετρήστε την πιθανότητα σφάλματος και σχεδιάστε τις καμπύλες BER για τιμές του $\text{SNR}=[0:2:16]\text{dB}$. Οι καμπύλες BER θα πρέπει να σχεδιαστούν στο ίδιο γράφημα. Στο ίδιο γράφημα, σχεδιάστε και το θεωρητικό BER για το κάθε M-PSK από αυτά που υλοποιήσατε. Σχολιάστε τα αποτελέσματα. Ποιο σύστημα είναι καλύτερο ως προς την πιθανότητα σφάλματος για το ίδιο SNR; Πόσο παραπάνω SNR απαιτείται για να έχει το χειρότερο την ίδια πιθανότητα σφάλματος με το καλύτερο;

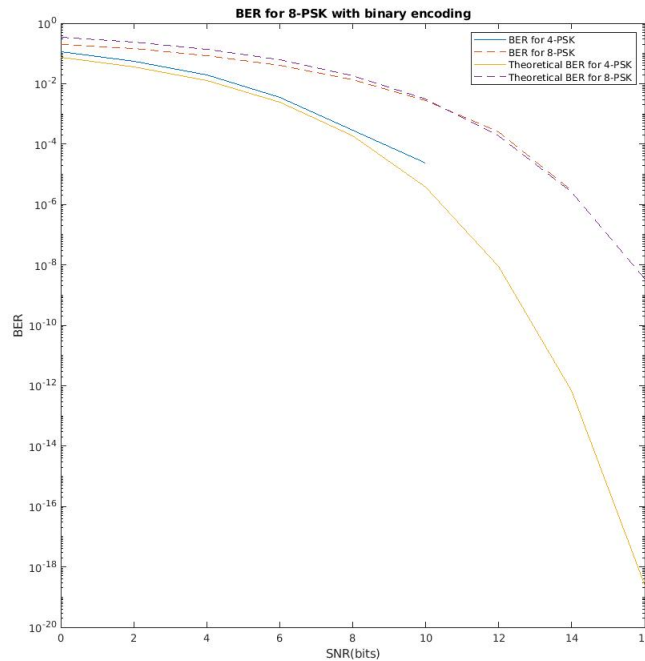
Απάντηση στο Ερώτημα 2

Παρακάτω φαίνονται τις γραφικές παραστάσεις SNR-BER για δυαδική κωδικοποίηση. Έχει υλοποιηθεί και κωδικοποίηση Gray αλλά δεν εισάγουμε τις γραφικές παραστάσεις αυτής καθώς δεν ζητείται. Πρέπει να αναφέρουμε επίσης ότι οι λογαριθμικές γραφικές παραστάσεις έχουν ένα πρόβλημα να γίνουν plot όταν το BER τείνει στο 0 καθώς ο λογάριθμος τείνει στο $-\infty$ και δεν βρήκαμε τρόπο να παραστήσουμε γραφικά την ασύμπτωτη αυτή στο Matlab.

Γενικά, παρατηρούμε ότι το 4-PSK συγκλίνει σε μηδενικό BER πιο γρήγορα από το 8-PSK. Αυτό είναι απολύτως φυσιολογικό αφού μειώνεται η απόσταση μεταξύ των συμβόλων του αστερισμού. Επίσης, παρατηρούμε ότι για $\text{SNR}=0$ έχουν και τα δύο την ίδια (χειρίστη) απόδοση, ενώ όσο αυξάνει το SNR τόσο καλύτερα αποδίδει ως προς το BER το 4-PSK, με την μεγαλύτερη διαφορά τους να βρίσκεται όταν έχουν συγκλίνει σε μηδενικό BER και τα δύο συστήματα. Μια ακόμα παρατήρηση είναι ότι η σύγκλιση γίνεται μη γραμμικά, για την ακρίβεια γίνεται λογαριθμικά. Αυτό μπορούμε να το καταλάβουμε γιατί στην περίπτωση που η σύγκλιση γινόταν γραμμικά τότε η γραφική παράσταση θα είχε την μορφή $\log x$, το οποίο δεν ισχύει.

Είναι προφανές από τα σχήματα, ότι για το ίδιο SNR το 4-PSK σύστημα είναι καλύτερο από το 8-PSK σύστημα. Ωστόσο όπως γνωρίζουμε το αντίτιμο του μεγαλύτερου SNR που απαιτείται για την εκπομπή του 8-PSK μας προσφέρει μεγαλύτερο ρυθμό μετάδοσης bit, αν κρατήσουμε όλες τις υπόλοιπες παραμέτρους ίδιες.

Σχετικά με το πόσο παραπάνω SNR απαιτείται από το 8-PSK (χειρότερο) για να επιτύχει την απόδοση του 4-PSK (καλύτερο) παρατηρούμε ότι όσο μειώνεται το επιθυμητό BER τόσο αυξάνεται και το παραπάνω SNR που απαιτεί το 8-PSK για να επιτύχει την ίδια απόδοση με το 4-PSK. Η μεγαλύτερη διαφορά σε SNR που εμφανίζεται είναι όταν το BER τείνει στο 0 και είναι περίπου 4-4.2dB.



Σχήμα 1: Καμπύλη BER για δυαδική κωδικοποίηση M-PSK, $M = [4, 8]$ bits και SNR = 0:2:16dB

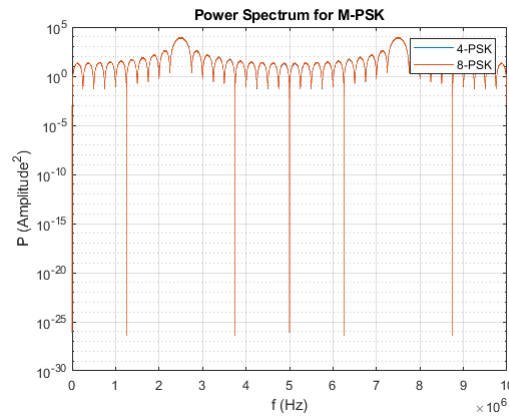
3 Ερώτημα 3

Για καθένα από τα δύο συστήματα, υπολογίστε το φάσμα ισχύος του εκπεμπόμενου σήματος. Σχεδιάστε τα δύο φάσματα στο ίδιο γράφημα σε λογαριθμική κλίμακα. Τι παρατηρείτε ως προς το εύρος ζώνης που απαιτεί το καθένα; Σχολιάστε το εύρος και το πλάτος του κύριου και των δευτερευόντων λοβών κάθε διαμόρφωσης.

Απάντηση στο Ερώτημα 3

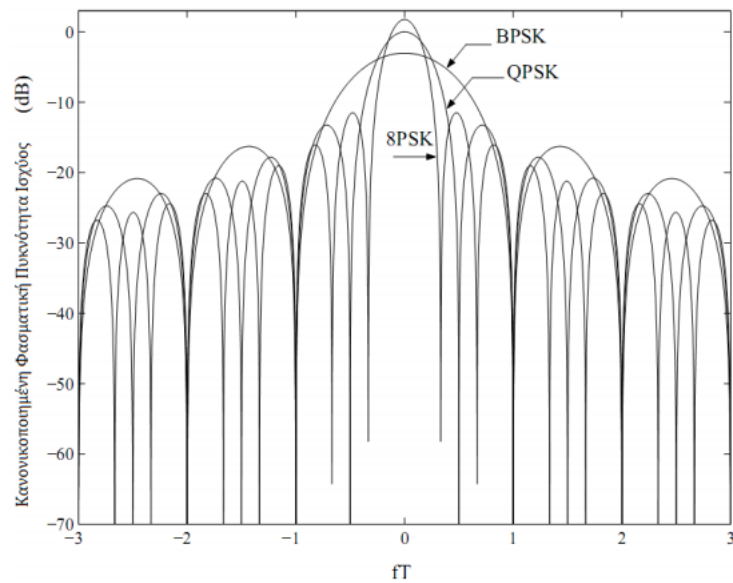
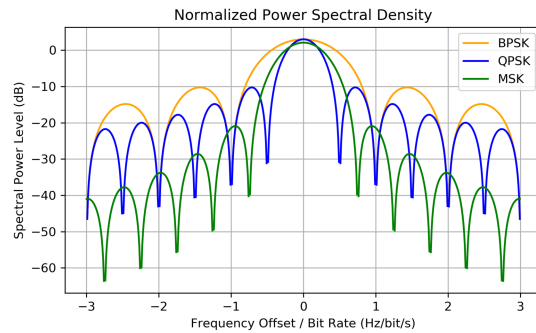
Στο ερώτημα αυτό αντιμετωπίσαμε πρόβλημα στην εκπόνηση παρά τον χρόνο που αφιερώσαμε στην αποσφαλμάτωση που δεν καρποφόρησε. Ουσιαστικά στην παρακάτω εικόνα φαίνεται το σχήμα της PSD που λαμβάνουμε αν υλοποιήσουμε τον υπο-

λογισμό της σύμφωνα με την εκφώνηση. Το πρόβλημα είναι ότι όταν αλλάζουμε το M δεν παρατηρείται διαφοροποίηση στο PSD όπως θα ήταν αναμενόμενο. Έτσι, στο σχήμα παρότι φαίνεται μόνο το 8-PSK αν γίνει αρκετό zoom φαίνεται και το 4-PSK το οποίο ταυτίζεται ακριβώς με το 8-PSK. Αυτό είναι λάθος.



Σχήμα 2:

Για την απάντηση του ερωτήματος ως προς το θεωρητικό κομμάτι αποφασίσαμε να πάρουμε κάποιο σχήμα από το διαδίκτυο. Αυτά τα βλέπουμε παρακάτω. Προφανώς το σύστημα μας δεν είναι βασικής ζώνης όπως του σχήματος. Μπορούμε να σκεφτούμε το 2.5×10^6 ως το κέντρο του κεντρικού λοβού. Ουσιαστικά η κυριότερη διαφορά που παρατηρούμε στο 8-PSK σε σχέση με το 4-PSK είναι ότι απαιτούν το ίδιο εύρος ζώνης αλλά έχουν πολύ διαφορετική κατανομή της ισχύος στο εύρος αυτό. Έτσι όσο αυξάνεται το M διασπορά της ισχύος στο φάσμα των συχνοτήτων μειώνεται. Έτσι στην οριακή περίπτωση, θεωρητικά όσο αυξάνεται το M μειώνεται και το εύρος ζώνης που απαιτεί το σύστημα. Επίσης, παρατηρούμε ότι καθώς αυξάνεται το M μειώνεται η ισχύς στον κεντρικό λοβό αλλά και εμφανίζονται περισσότεροι δευτερεύοντες λοβοί. Ουσιαστικά αυτό που συμβαίνει είναι ότι κάθε φορά που το M υψώνεται στο τετράγωνο (π.χ. 2, 4, 16, ...) τότε στη θέση κάθε λοβού προκύπτουν δύο λοβοί με στο ίδιο ακριβώς πλάτος φάσματος με τον αρχικό αλλά με μικρότερη ισχύ αν τις προσθέσουμε και τις δύο σε σχέση με τον αρχικό. Πιο συγκεκριμένα, στην περίπτωση του 8-PSK ο κεντρικός λοβός του 4-PSK που αποτελείται από 3 υπολοβούς σπάει σε 5 υπολοβούς.



4 Ερώτημα 4

[Θεωρητική] Παρατηρήστε το Σχήμα 7.57 του βιβλίου (Σχήμα 7.33 στο βιβλίο του Καραγιαννίδη) όπου φαίνονται οι καμπύλες SER για διάφορες τιμές του M σε συστήματα M-PSK. Την ίδια μορφή περίπου έχουν και οι αντίστοιχες καμπύλες BER. Τι παρατηρείτε από τα δύο αυτά σχήματα; Σχολιάστε και συγκρίνετε τις διαμορφώσεις M-PSK ως προς το ρυθμό μετάδοσης bits, την πιθανότητα σφάλματος, και το απαιτούμενο εύρος ζώνης όταν αυξάνει το M .

Απάντηση στο Ερώτημα 4

Στο βιβλίο του Καραγιαννίδη το σχήμα που λήφθηκε υπόψιν ήταν το Σχήμα 7.33 της σελίδας 526 γιατί το 7.57 δεν ταίριαζε με τα δεδομένα της εκφώνησης. Επίσης, αφού σύμφωνα με την εκφώνηση θεωρούνται τα BER και SER διαγράμματα ίδια, αν και για παράδειγμα στην περίπτωση που έχουμε Gray κωδικοποίηση με AWGN κανάλι το BER είναι μικρότερο του SER θα αναφερόμαστε μόνο στο SER που είναι και αυτό που απεικονίζεται στο σχήμα του βιβλίου. Αντί για SER μερικές φορές το αναφέρουμε ως πιθανότητα σφάλματος.

Αρχικά, παρατηρούμε ότι για το ίδιο SNR έχουμε μεγαλύτερο SER όσο αυξάνει το M . Αυτό είναι εύκολο να ερμηνευθεί αν λάβει υπόψιν κάποιος ότι κάθε σύμβολο s_i (διάνυσμα στο χώρο σημάτων βάσης) στη πλευρά του δέκτη στο δισδιάστατο χώρο σημάτων μας είναι ένα δείγμα από μια Γκαουσιανή τυχαία μεταβλητή με μέση τιμή τη s_i . Δηλαδή το 2δ διάνυσμα είναι δείγμα από τη 2δ Γκαουσιανή τ.μ. με μέση τιμή s_i . Υπενθυμίζουμε ότι η φώραση γίνεται λαμβάνοντας υπόψιν την Ευκλείδεια απόσταση του δείγματος από κάθε σύμβολο του αστερισμού. Έτσι το SER είναι ανάλογο της απόστασης των συμβόλων στον αστερισμό. Οπότε για σταθερό λόγο SNR όσο πιο πυκνά τα σύμβολα δηλαδή όσο μεγαλύτερο M τόσο μεγαλύτερο SER. Παρατηρούμε ακόμα ότι για μικρότερα M το SER ο μηδενισμός της πιθανότητας σφάλματος συμβαίνει με λιγότερο SNR. Αυτό είναι άμεση απόρροια του προηγούμενου. Δηλαδή όσο αυξάνεται το SNR τόσο μειώνεται η διασπορά της 2δ τ.μ. και αφού όσο αυξάνεται το M μικραίνει η απόσταση μεταξύ των συμβόλων του αστερισμού είναι λογικό για μικρότερες αποστάσεις (μεγάλο M) να απαιτούμε μεγαλύτερα SNR.

Αναφερόμενοι καθαρά σε ρυθμό μετάδοσης με σταθερές όλες τις άλλες παραμέτρους όπως το SNR. Τότε προφανώς όσο αυξάνει το M υπάρχει μεγαλύτερος ρυθμός μετάδοσης. Αφού ο αστερισμός περιλαμβάνει περισσότερα σύμβολα και άρα μπορούμε να κωδικοποιήσουμε περισσότερα bits σε ένα σύμβολο. Προφανώς ο ρυθμός αποστολής δεν έχει καμία σχέση με το ρυθμό λήψης συμβόλων αφού εκεί πρέπει να λάβουμε υπόψιν και τον παράγοντα SNR.

Για σταθερό SNR καθώς και όλες τις υπόλοιπες παραμέτρους, όσο αυξάνει το M η πιθανότητα σφάλματος αυξάνεται. Αυτό μπορούμε να το διαπιστώσουμε πέρα από το σχήμα, και στο τύπο υπολογισμού του P που δίνεται στην εκφώνηση της άσκησης.

Παρατηρούμε επίσης ότι όλες οι PSK διαμορφώσεις απαιτούν το ίδιο εύρος ζώνης. Ωστόσο, όσο αυξάνει το M μειώνεται η διασπορά της ισχύος του σήματος στο εύρος ζώνης. Αυτό μπορούμε να το αντιληφθούμε και από τα σχήματα του Ερωτήματος 3. Αυτή η μείωση της διασποράς της ισχύος του σήματος, έχει ως συνέπεια για αρκετά μεγάλα M να μπορούμε να επαναχρησιμοποιήσουμε ένα μέρος του εύρους ζώνης για διαμορφώσεις σε γειτονικές φέρουσες συχνότητες.

5 Κώδικας

```
1 L_b = randsrc(24^4, 1, [0 1]);  
2 %L_b = randsrc(24^2, 1, [0 1]);  
3  
4 % BINARY (UNIPOLAR) ENCODING
```



```

5
6 M = [4 8];
7 SNR = 0:2:16;
8 [X, Y] = meshgrid(M, SNR);
9 Z1 = zeros(size(X));
10
11 for i=1:size(Z1,1)
12     for j=1:size(Z1,2)
13         [~, BER, ~] = M_PSK(L_b, X(i,j), Y(i,j), 'bin',
14                             0);
15         Z1(i,j)=BER;
16     end
17 end
18 Z1(isinf(Z1)) = 10^-15;
19
20 % figure;
21 % a = axes;
22 % s = mesh(X,Y,Z);
23 % title('BER for M-PSK with binary encoding');
24 % set(a,'ZScale','log');
25 % s.FaceColor = 'flat';
26 % xlabel('M(bits)');
27 % ylabel('SNR(dB)');
28 % zlabel('BER');
29
30 % figure;
31 % semilogy(SNR, Z(:,1));
32 % title('BER for 4-PSK with binary encoding');
33 % xlabel('SNR(bits)');
34 % ylabel('BER');
35 %
36 % figure
37 % semilogy(SNR, Z(:,2));
38 % title('BER for 8-PSK with binary encoding');
39 % xlabel('SNR(bits)');
40 % ylabel('BER');
41
42 % GRAY ENCODING
43
44 % M = [4 8];
45 % SNR = 0:2:16;
46 %
47 % [X, Y] = meshgrid(M, SNR);
48 % Z = zeros(size(X));
49 %

```

```

50 % for i=1:size(Z,1)
51 %     for j=1:size(Z,2)
52 %         m = X(i,j);
53 %         snr = Y(i,j);
54 %         [~, BER, ~] = M_PSK(L_b, m, snr, 'gray', 0);
55 %         Z(i,j)=BER;
56 %
57 %     end
58 % end
59 %
60 % Z(isinf(Z)) = 10^-3;
61 %
62 % figure;
63 % a = axes;
64 % s = mesh(X,Y,Z);
65 % set(a,'ZScale','log')
66 % title('BER for M-PSK with gray encoding');
67 % s.FaceColor = 'flat';
68 % xlabel('M( bits) ');
69 % ylabel('SNR(dB) ');
70 % zlabel('BER');
71
72 % semilogy(SNR, Z(:,1));
73 % title('BER for M-PSK with binary encoding');
74 % xlabel('SNR( bits) ');
75 % ylabel('BER');
76
77 % THEORETICAL BER
78
79 M = [4 8];
80 SNR = 0:2:16;
81
82 [X, Y] = meshgrid(M, SNR);
83 Z2 = zeros(size(X));
84
85 for i=1:size(Z2,1)
86     for j=1:size(Z2,2)
87         m = X(i,j);
88         snr = Y(i,j);
89         if m == 4
90             N0_4 = 10^(-snr/10)/log2(4);
91             Eb_4 = 1/log2(4);
92             Z2(i,j) = qfunc(sqrt(2*Eb_4/N0_4))*(1-1/2*
                    qfunc(sqrt(2*Eb_4/N0_4)));
93         else
94             N0_8 = 10^(-snr/10)/log2(8);

```

```

95         Eb_8 = 1 / log2(8);
96         Z2(i,j) = 2 * qfunc(sqrt(2 * log2(8) * Eb_8 /
          N0_8)*sin(pi/m));
97     end
98 end
99 end
100
101 Z2(isinf(Z1)) = 10^-15;
102
103
104 % figure;
105 % a = axes;
106 % s = mesh(X,Y,Z2);
107 % title('Theoretical BER for M-PSK with binary encoding')
108 ;
109 % set(a,'ZScale','log')
110 % s.FaceColor = 'flat';
111 % xlabel('M(bits)');
112 % ylabel('SNR(dB)');
113 % zlabel('BER');
114
115 % figure;
116 % semilogy(SNR, Z(:,1));
117 % title('Theoretical BER for 4-PSK with binary encoding')
118 ;
119 % xlabel('SNR(bits)');
120 % ylabel('BER');
121 %
122
123 figure;
124 semilogy(SNR, Z1(:,1), '-');hold on;
125 semilogy(SNR, Z1(:,2), '--');hold on;
126 semilogy(SNR, Z2(:,1), '-');hold on;
127 semilogy(SNR, Z2(:,2), '--');
128 title('BER for 8-PSK with binary encoding');
129 legend('BER for 4-PSK', 'BER for 8-PSK', 'Theoretical BER
    for 4-PSK', 'Theoretical BER for 8-PSK')
130 xlabel('SNR(bits)');
131 ylabel('BER');

```

Listing 1: Πηγαίος κώδικας για Erotima_2.m

```

1 L_b = randsrc(24^4, 1, [0 1]);
2
3 smt_4 = M_PSK_Transmitter(L_b, 4, 'gray');
4 smt_8 = M_PSK_Transmitter(L_b, 8, 'gray');
5

```

```

6 T_sample = 0.1 * 10^-6;
7 f_sample = 1/T_sample;
8 n = 2048;
9 f = (0:n-1)*(f_sample/n);
10
11 figure;
12
13 % x_t = smt_4;
14 % [Pxx,F] = periodogram(x_t,[],length(x_t),f_sample);
15 % plot(F,10*log10(Pxx)); hold on;
16 %
17 % x_t = smt_8;
18 % [Pxx,F] = periodogram(x_t,[],length(x_t),f_sample);
19 % plot(F,10*log10(Pxx));
20
21 x_t = smt_4;
22 x_t = reshape(x_t, n, []);
23 P1 = mean((abs(fft(x_t, 2048)).^2), 2);
24
25 x_t = smt_8;
26 x_t = reshape(x_t, n, []);
27 P2 = mean((abs(fft(x_t,2048)).^2), 2);
28
29 semilogy(f,P1); hold on;
30 semilogy(f,P2);
31
32 % plot(f,P1); hold on;
33 % plot(f,P2);
34
35 grid on;
36 title(sprintf('Power Spectrum for M-PSK'));
37 xlabel('f (Hz)');
38 ylabel('P (Amplitude^2)');
39 legend('4-PSK', '8-PSK');

```

Listing 2: Πηγάιος κώδικας για Erotima_3.m

```

1 function [smt] = M_PSK_Transmitter(inpt, M, encoding)
2
3 R = log2(M);
4
5 % We have 4 samples (T_sample) per carrier period (Tc)
   and
6 % 10 carrier periods (Tc) per symbol period (T_sym).
7
8 % T_scale = 4 * 10^-6;
9

```

```

10 T_sample = 1;
11 T_c = 8;
12 f_c = 1/T_c;
13 T_sym = 40;
14
15 g_t = sqrt(2 / T_sym);
16
17 % Constellation
18 constellation = zeros(M, 2);
19 for m = 0:M-1
20     constellation(m+1,:) = [cos(2 * pi * m/M) sin(2 * pi
        * m/M)];
21 end
22
23 % -----Mapper
24
25 blocks_inpt = reshape(inpt.', R, [])';
26 if strcmp(encoding, 'bin')
27     sm = bi2de(blocks_inpt, 'left-msb');
28 else
29     sm = bi2de(blocks_inpt, 'left-msb');
30     sm = bin2gray(sm, 'psk', M);
31 end
32
33 tmp = repelem(sm, T_sym);
34 figure('Position', [10 10 900 600])
35 subplot(4,1,1);
36 plot(1:T_sample:T_sample*T_sym*10, tmp(1:T_sample*T_sym
    *10));
37 title("Subplot 1: Original signal")
38
39
40 % -----Modulation
41
42 smt = zeros(length(sm) * T_sym, 1);
43 t = (0:T_sample:(T_sym - T_sample))';
44
45 for i = 1:length(sm)
46     for j = 1:length(t)
47         smt((i - 1) * T_sym + j) = ...
48             g_t * cos(2 * pi * (sm(i)/M)) * cos(2 * pi *
                f_c * t(j)) + ...
49             g_t * sin(2 * pi * (sm(i)/M)) * sin(2 * pi *
                f_c * t(j));

```

```

50     end
51 end
52
53 subplot(4,1,2);
54 plot(1:T_sample:T_sample*T_sym*10, smt(1:T_sample*T_sym
    *10));
55 title("Subplot 2: Transmitter 's output");
56
57
58 end

```

Listing 3: Πηγαίος κώδικας για M_PSK_Transmitter.m

```

1  function [out, BER, SER] = M_PSK(inpt, M, SNR, encoding,
    show_plot)
2
3  E_s=1;
4
5  R = log2(M);
6
7  % We have 4 samples (T_sample) per carrier period (Tc)
    and
8  % 2 carrier periods (Tc) per symbol period (T_sym).
9
10 % T_scale = 4 * 10^-6;
11
12 T_sample = 1;
13 T_c = 4;
14 f_c = 1/T_c;
15 T_sym = 40;
16
17 g_t = sqrt(2 / T_sym);
18
19 % Constellation
20 constellation = zeros(M, 2);
21 for m = 0:M-1
22     constellation(m+1,:) = [cos(2 * pi * m/M) sin(2 * pi
        * m/M)];
23 end
24
25
26 % -----Mapper
    -----
27
28 blocks_inpt = reshape(inpt.', R, []).';
29 if strcmp(encoding, 'bin')
30     sm = bi2de(blocks_inpt, 'left-msb');

```

```

31 else
32     sm = bi2de(blocks_inpt, 'left-msb');
33     sm = bin2gray(sm, 'psk', M);
34 end
35
36 if show_plot
37     tmp = repelem(sm, T_sym);
38     figure('Position', [10 10 900 600])
39     subplot(4,1,1);
40     plot(1:T_sample:T_sample*T_sym*10, tmp(1:T_sample*
        T_sym*10));
41     title("Subplot 1: Original signal")
42 end
43
44 % -----Modulation
45
46 smt = zeros(length(sm) * T_sym, 1);
47 t = (0:T_sample:(T_sym - T_sample))';
48
49 for i = 1:length(sm)
50     for j = 1:length(t)
51         smt((i - 1) * T_sym + j) = ...
52             g_t * cos(2 * pi * (sm(i)/M)) * cos(2 * pi *
53                 f_c * t(j)) - ...
54             g_t * sin(2 * pi * (sm(i)/M)) * sin(2 * pi *
55                 f_c * t(j));
56     end
57 end
58
59 if show_plot
60     subplot(4,1,2);
61     plot(1:T_sample:T_sample*T_sym*10, smt(1:T_sample*
62         T_sym*10));
63     title("Subplot 2: Transmitter's output");
64 end
65
66 % -----AWGN
67
68 var = (E_s / (2 * log2(M))) * 10 ^ (- SNR / 10);
69 noise = sqrt(var) * randn(length(smt), 1);
70
71 r = smt + noise; % receiving signal
72
73 if show_plot

```

```

71     subplot(4,1,3);
72     plot(1:T_sample:T_sample*T_sym*10, r(1:T_sample*T_sym
73         *10));
74     title("Subplot 3: Signal with AWGN");
75 end
76 %-----Demodulation
77 % Calculates the norm of the projection of vector r to
78 % every basis signal,
79 % which is the inner product <sm(t), ??_j(t)>
80 r = repmat(r, 1, 2);
81
82 % Calculate product sm(t) * ??_j(t)
83 for i = 1:length(r)
84     r(i,1) = r(i,1) * g_t * cos(2*pi*f_c*(i-1));
85     r(i,2) = r(i,2) * g_t * -sin(2*pi*f_c*(i-1));
86 end
87
88 % if show_plot
89     subplot(4,1,4);
90     plot(1:T_sample:T_sample*T_sym*10, r(1:T_sample*
91         T_sym*10));
92     title("Subplot 4: Receiver's signal after filtering
93         ")
94 % end
95
96 % Calculate inner product <sm(t), ??_j(t)>
97 % basically it integrates the previously calculated
98 % product
99 demodulated = zeros(size(r,1)/T_sym, size(r,2));
100 for i = 1:size(demodulated,1)
101     from = ((i-1)*T_sym)+1;
102     to = from + T_sym -1;
103     demodulated(i, :) = sum(r(from:to, :));
104 end
105
106 %-----Decision
107
108 decision=zeros(size(sm));
109 for i=1:length(demodulated)
110     dist=abs(sum(sqrt((constellation-demodulated(i,:))
111         .^2),2));
112     [~,ind]=min(dist);

```



```

109     decision(i)=ind-1;
110 end
111
112 % Calculate SER
113 errorsym=0;
114 totalsymb=0;
115 for i=1:length(sm)
116     if decision(i)~=sm(i)
117         errorsym=errorsym+1;
118     end
119     totalsymb=totalsymb+1;
120 end
121
122
123 % -----Demapper
124
125 if strcmp(encoding,'bin')
126     blocks_out=de2bi(decision,R,'left-msb');
127 else
128     decision=gray2bin(decision,'psk',M);
129     blocks_out=de2bi(decision,R,'left-msb');
130 end
131
132 output_sequence=reshape(blocks_out.',[],1);
133
134 k=inpt(inpt~=output_sequence);
135 BER=length(k)/length(inpt);
136 out=output_sequence;
137 SER=errorsym/totalsymb;
138
139 end

```

Listing 4: Πηγαίος κώδικας για M_PSK.m