

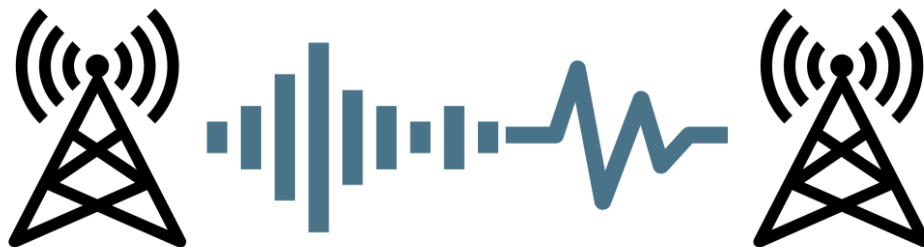


# ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ Ι

---

**Αναφορά 3<sup>ης</sup> Εργαστηριακής Άσκησης**  
«ΜΕΛΕΤΗ ΤΗΛ. ΣΥΣΤΗΜΑΤΟΣ ΜΕ ΔΙΑΜΟΡΦΩΣΗ 8-PSK»

---



















---

 **Μιχάλης Γαλάνης** 201603003

---

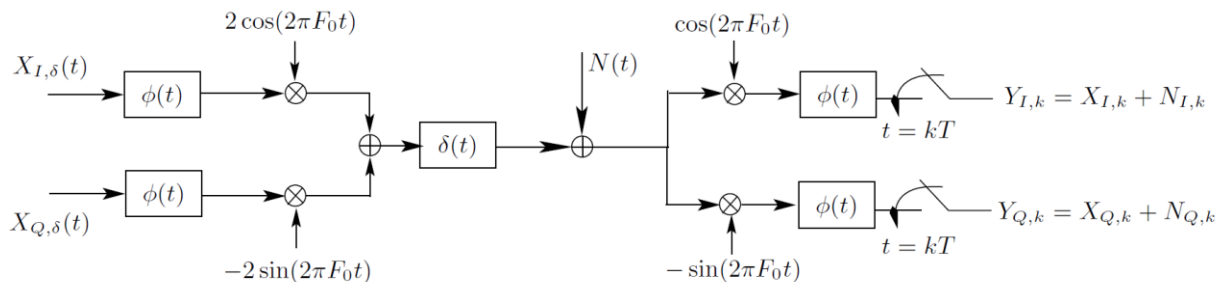
# ΠΕΡΙΕΧΟΜΕΝΑ

	ΠΕΡΙΛΗΨΗ .....	1
	Τμήμα Κώδικα.....	1
	A.1.....	2
	Τμήμα Κώδικα.....	2
	A.2.....	2
	Τμήμα Κώδικα (Κύριο Μέρος).....	2
	Τμήμα Κώδικα (Συνάρτηση) .....	3
	A.3.....	4
	Τμήμα Κώδικα.....	4
	Διαγράμματα .....	5
	A.4.....	6
	Τμήμα Κώδικα.....	6
	Διαγράμματα .....	7
	A.5.....	9
	Τμήμα Κώδικα.....	9
	Διαγράμματα .....	9
	A.6.....	10

	A.7.....	10
	Τμήμα Κώδικα.....	10
	A.8.....	11
	Τμήμα Κώδικα.....	11
	Διαγράμματα .....	12
	A.9.....	13
	Τμήμα Κώδικα.....	13
	Διαγράμματα .....	14
	A.10.....	15
	Τμήμα Κώδικα.....	15
	Διαγράμματα .....	15
	A.11 .....	16
	Τμήμα Κώδικα (Κύριο Μέρος).....	16
	Τμήμα Κώδικα (Συνάρτηση) .....	16

## ΠΕΡΙΛΗΨΗ

Στη 3<sup>η</sup> εργαστηριακή άσκηση προσωμοιώνουμε ένα τηλεπικοινωνιακό σύστημα που χρησιμοποιεί διαμόρφωση 8-PSK και το μελετάμε ως προς την απόδοσή του και τα σφάλματα που παράγονται ανάλογα με το signal to noise ratio (SNR).



Για ακόμη μια φορά, χρησιμοποιούνται στον κώδικα οι παρακάτω ανώνυμες συναρτήσεις για την απλοποίηση του, οι οποίες έχουν εξηγηθεί σε προηγούμενες ασκήσεις:

## Τμήμα Κώδικα

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 % BASIC INFORMATION %
3 % Course: Telecommunication Systems I - Excercise 3 %
4 % Deadline: 20-Dec-18 %
5 % FullName: Mixalis Galanis %
6 % Academic ID: 2016030036 %
7 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
8
9 %Clearing things up
10 close all
11 clearvars
12
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14 % Useful Functions %
15 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
16 %Plotting Preperation Functions
17 set_diag_labels = @(tit, xlab, ylab){title(tit); xlabel(xlab); ylabel(ylab)};
18 set_diag_layout = @(fig, rows, cols, selected){figure(fig); subplot(rows,cols,selected)};
19 %Some Useful Plotting Functions (continuous & discrete)
20 c_plot = @(t, f){ plot(t,f) };
21 c_semilogy = @(t, f){ semilogy(t,f) };
22 d_plot = @(t, f){ stem(t,f) };
23 %Some Useful Signal Processing Functions
24 t_conv = @(t1, t2, dt) (t1(1) + t2(1)):dt:(t1(end) + t2(end)); %Convolution time
25 gen_N_bits = @(N) ((sign(randn(N,1)) + 1)/2); %Generates N Random Bits
26 F_T = @(X,Nf,Ts) (abs(fftshift(fft(X,Nf)*Ts))); %Fourier Transform of X
27 P_X = @(t_X, X,Nf,Ts) ((F_T(X,Nf,Ts).^2)/(length(t_X)*Ts)); %Periodogram of X
```

## A.1

Δημιουργούμε μια ακολουθία ισοπίθανων bits ( $3N$ ) με  $N = 50$ ;

### Τμήμα Κώδικα

```
30 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
31 % A1 - Creating N Bits Sequence %
32 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
33 - N = 50;
34 - bit_seq = gen_N_bits(3*N);
```

## A.2

Κατασκευάσα μια συνάρτηση bits\_to\_PSK\_8 όπου κατασκευάζει μια ακολουθία συμβόλων 8-PSK από μια δυαδική ακολουθία. Αυτό επιτυγχάνεται ομαδοποιώντας τα bit ανά 3άδες και υπολογίζοντας την τιμή:

$$X_n = \begin{bmatrix} X_{I,n} \\ X_{Q,n} \end{bmatrix}$$

Με κάθε διάνυσμα  $X_n$  να παίρνει τιμές από το αλφάβιτο 8-PSK  $\{x_0, \dots, x_m\}$  με:

$$x_m = \begin{bmatrix} A \cos \frac{\pi m}{4} \\ X_{Q,n} \sin \frac{\pi m}{4} \end{bmatrix}$$

Επιλέχθηκε η ακόλουθη διαδρομή για την κωδικοποίηση Gray:

$$000 \rightarrow 001 \rightarrow 011 \rightarrow 010 \rightarrow 110 \rightarrow 111 \rightarrow 101 \rightarrow 100$$

Επιλέχθηκε και το πλάτος ίσο με 1 καθώς πρέπει να ισχύει η συνθήκη:

$$\mathcal{E}[\|\mathbf{X}_n\|_2^2] = \mathcal{E}[X_{I,n}^2] + \mathcal{E}[X_{Q,n}^2] = 1.$$

### Τμήμα Κώδικα (Κύριο Μέρος)

```
36 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
37 % A2 - Generating 8-PSK bits %
38 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
39 - A = 1; %TODO Figure out A's value
40 - bit_8_psk = bits_to_PSK_8(bit_seq, A); %N length
```

## Τμήμα Κώδικα (Συνάρτηση)

```
1 function [psk_bits] = bits_to_PSK_8(b, A)
2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3 % psk_bits = bits_to_PSK_8(b) %
4 % OUTPUT %
5 %     psk_bits: a vector of 8-PSK symbols %
6 % INPUT %
7 %     b: sequence (vector) of binary bits %
8 %     A: Amplitude %
9 % %
10 %     M. Galanis, Dec. 2018 %
11 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
12 L = length(b)/3;
13 psk_bits = zeros(L,2);
14 Xn = [0 0];
15 n = 1;
16
17 for i = 1:3:length(b)
18     if ((b(i) == 0) && (b(i+1) == 0) && (b(i+2) == 0))
19         m = 0;
20     elseif ((b(i) == 0) && (b(i+1) == 0) && (b(i+2) == 1))
21         m = 1;
22     elseif ((b(i) == 0) && (b(i+1) == 1) && (b(i+2) == 1))
23         m = 2;
24     elseif ((b(i) == 0) && (b(i+1) == 1) && (b(i+2) == 0))
25         m = 3;
26     elseif ((b(i) == 1) && (b(i+1) == 1) && (b(i+2) == 0))
27         m = 4;
28     elseif ((b(i) == 1) && (b(i+1) == 1) && (b(i+2) == 1))
29         m = 5;
30     elseif ((b(i) == 1) && (b(i+1) == 0) && (b(i+2) == 1))
31         m = 6;
32     elseif ((b(i) == 1) && (b(i+1) == 0) && (b(i+2) == 0))
33         m = 7;
34     end
35     Xn(1) = A*cos((pi*m)/4);
36     Xn(2) = A*sin((pi*m)/4);
37     psk_bits(n, 1) = Xn(1);
38     psk_bits(n, 2) = Xn(2);
39     n = n + 1;
40 end
41 return
```



## A.3

Σε αυτό το ερώτημα περνάμε τις ακολουθίες  $X_{I,n}$  και  $X_{Q,n}$  από SRRC φίλτρα μορφωποίησης, δηλαδή τις συνελίσσουμε με τη  $\varphi(t)$  και ύστερα απεικονίζουμε τις κυματομορφές εξόδου και τα περιοδογράμματα τους.



### Τμήμα Κώδικα

```

44 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
45 % A3 - Filtering Xi & Xq with SRRC Pulses %
46 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
47 %Settings for plotting the diagrams
48 r = 2; c = 2; fig = 1; sub_fig = 1;
49
50 Nf = 2^11;
51 DT = 1;
52
53 Xi = squeeze(bit_8_psk(:,1));
54 t_Xi = 0:length(Xi);
55 Xq = squeeze(bit_8_psk(:,2));
56 t_Xq = 0:length(Xq);
57
58 %Building up parameters for srcc_pulse function
59 T = 0.01; %Nyquist parameter (>0)
60 over = 10; %Oversampling factor (>0)
61 Ts = T/over; %Sampling period (>0)
62 Fs = 1/Ts; %Sampling frequency (>0)
63 Hd = 4; %Half duration of the pulse (>0)
64 a = 0.5; %Roll-off factor(0<a<1)
65
66 %Calling srcc_pulse function to store phi and t variables
67 [phi, t_phi] = srcc_pulse(T, Ts, Hd, a);
68
69 %Convoluting Xi, Xq with phi
70 Xi_delta = Fs * upsample(Xi,over); %Inserts zeros in between bits
71 Xq_delta = Fs * upsample(Xq,over); %Inserts zeros in between bits
72
73 Xi_conv = conv(Xi_delta, phi)*Ts;
74 Xq_conv = conv(Xq_delta, phi)*Ts;
75
76 t_Xi_conv = t_conv(t_phi, t_Xi, Ts);
77 t_Xq_conv = t_conv(t_phi, t_Xq, Ts);
78
79 %Calculating Periodgrams
80 F_ES = (-Fs/2) : (Fs/Nf) : (Fs/2 - Fs/Nf);
81 PXi = P_X(t_Xi_conv, Xi_conv,Nf,Ts);
82 PXq = P_X(t_Xq_conv, Xq_conv,Nf,Ts);
83
84 %Plotting output diagrams
85 %Xi_conv
86 set_diag_layout(fig, r, c, sub_fig);
87 c_plot(t_Xi_conv,Xi_conv);
88 set_diag_labels("A.3 - Convolution of Xi,n with SRRC pulse","t(s)","");
89 grid on;

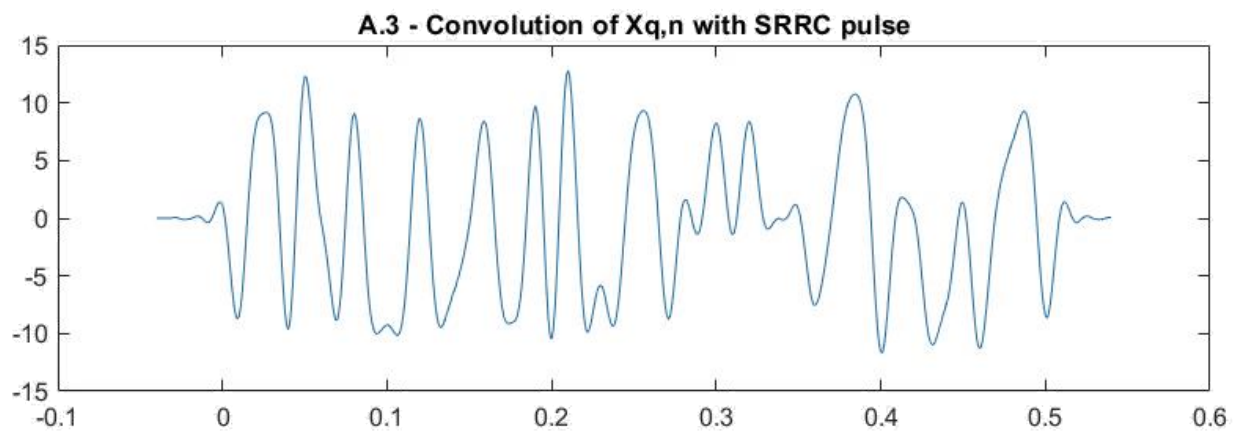
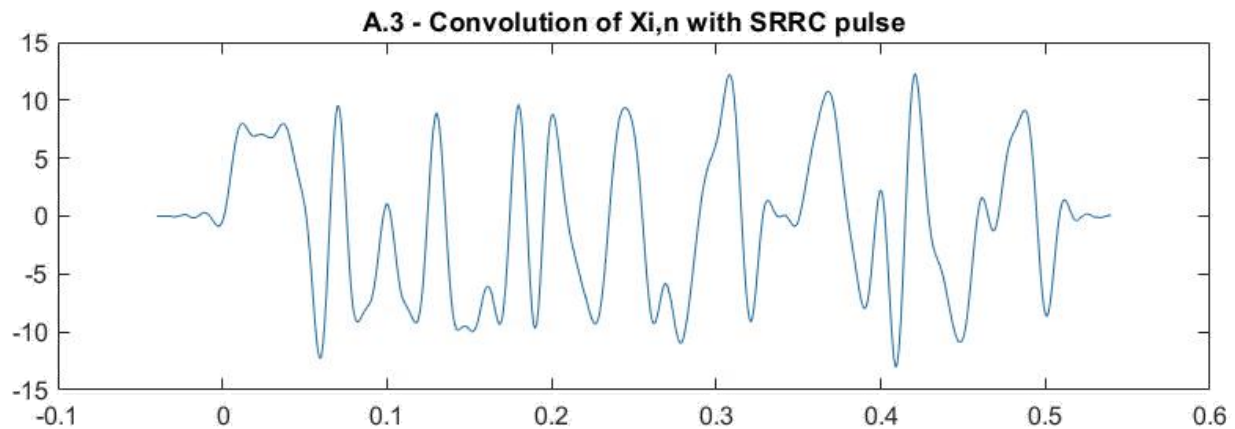
```

```

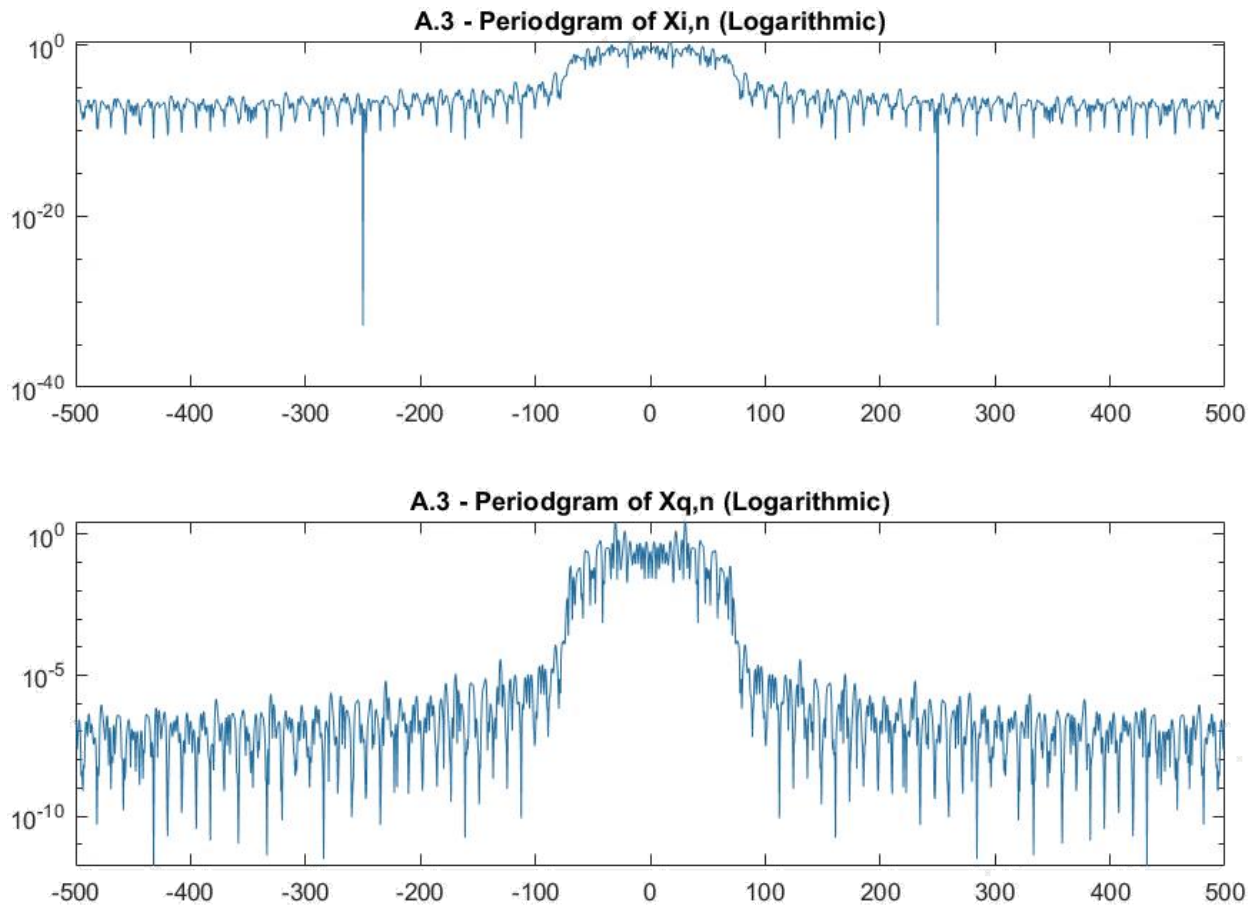
90 %Xq_conv
91 sub_fig = sub_fig + 1;
92 set_diag_layout(fig, r, c, sub_fig);
93 c_plot(t_Xq_conv,Xq_conv);
94 set_diag_labels("A.3 - Convolution of Xq,n with SRRC pulse","t(s)","");
95 grid on;
96 %PXi
97 sub_fig = sub_fig + 1;
98 set_diag_layout(fig, r, c, sub_fig);
99 c_semilogy(F_ES,PXi);
100 set_diag_labels("A.3 - Periodgram of Xi,n (Logarithmic)","F [Hz]","");
101 grid on;
102 %PXq
103 sub_fig = sub_fig + 1;
104 set_diag_layout(fig, r, c, sub_fig);
105 c_semilogy(F_ES,PXq);
106 set_diag_labels("A.3 - Periodgram of Xq,n (Logarithmic)","F [Hz]","");
107 grid on;

```

## Διαγράμματα







## A.4

Εδώ μας ζητήθηκε να πολλαπλασιάσουμε τις κυματομορφές από τους αντίστοιχους φορείς, που όπως βλέπουμε από το αρχικό μας σχήμα, είναι  $2 \cos 2\pi F_0 t$  για το  $X_{I,n}$  και  $-2 \sin 2\pi F_0 t$  για το  $X_{Q,n}$ . Με αυτή τη διαδικασία μετατοπίζουμε τα σήματά σε μια ζώνη μεγαλύτερων συχνοτήτων προκειμένου να μπορούν να μεταδωθούν.

### Τμήμα Κώδικα

```

109 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
110 % A4 - Multiplication of Xi, Xq by carrier %
111 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
112 fig = fig + 1; sub_fig = 1; %New Figure
113 F0 = 200;
114
115 %Calculating Multiplication by Carrier
116 Xi_t = Xi_conv .* (2*cos(2*pi*F0*t_Xi_conv));
117 Xq_t = Xq_conv .* ((-2)*sin(2*pi*F0*t_Xq_conv));

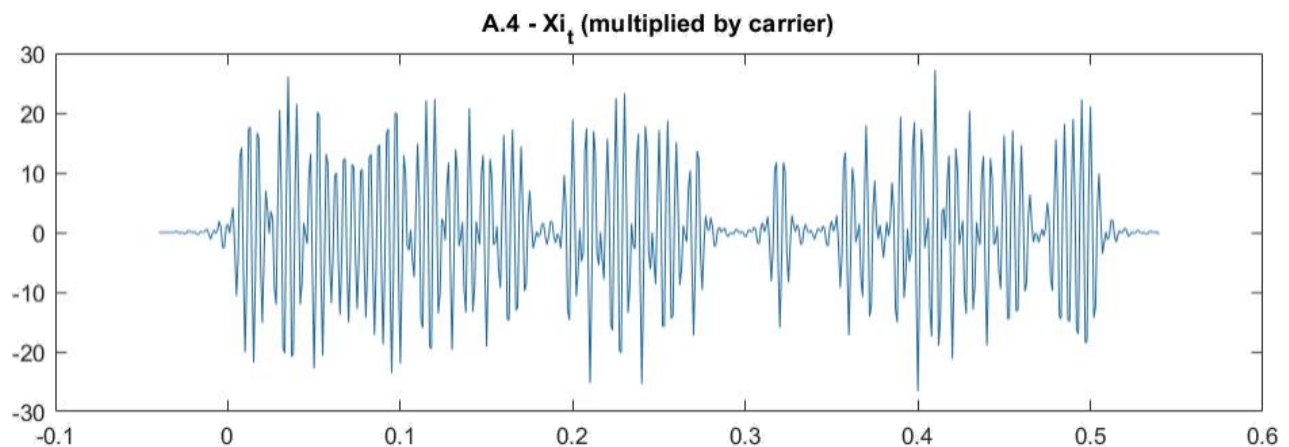
```

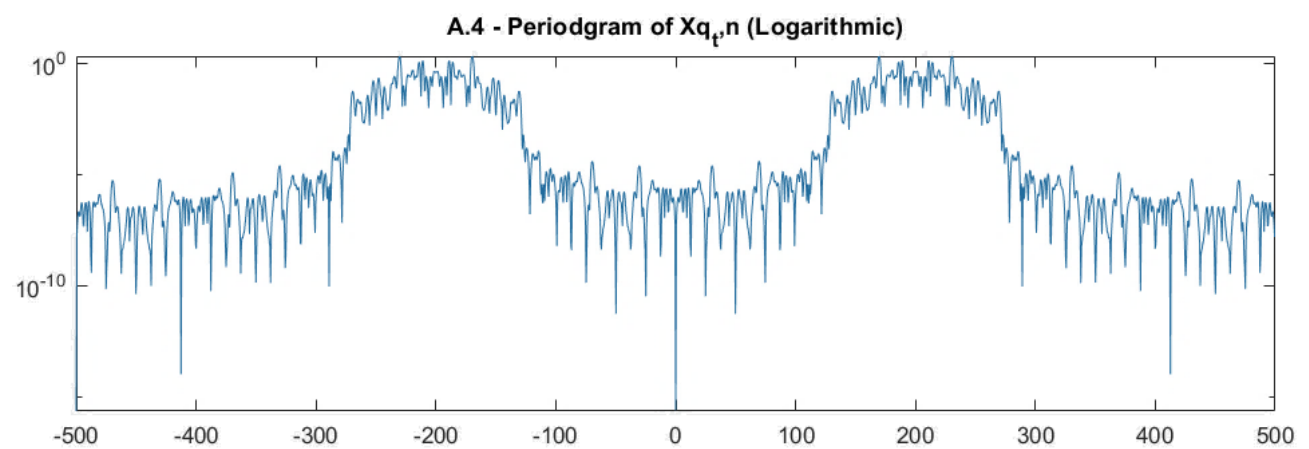
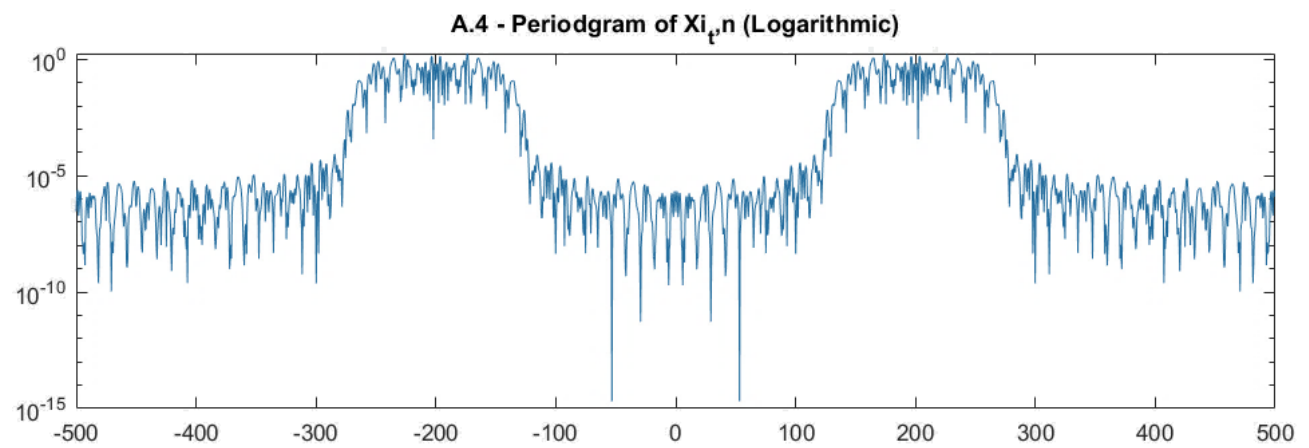
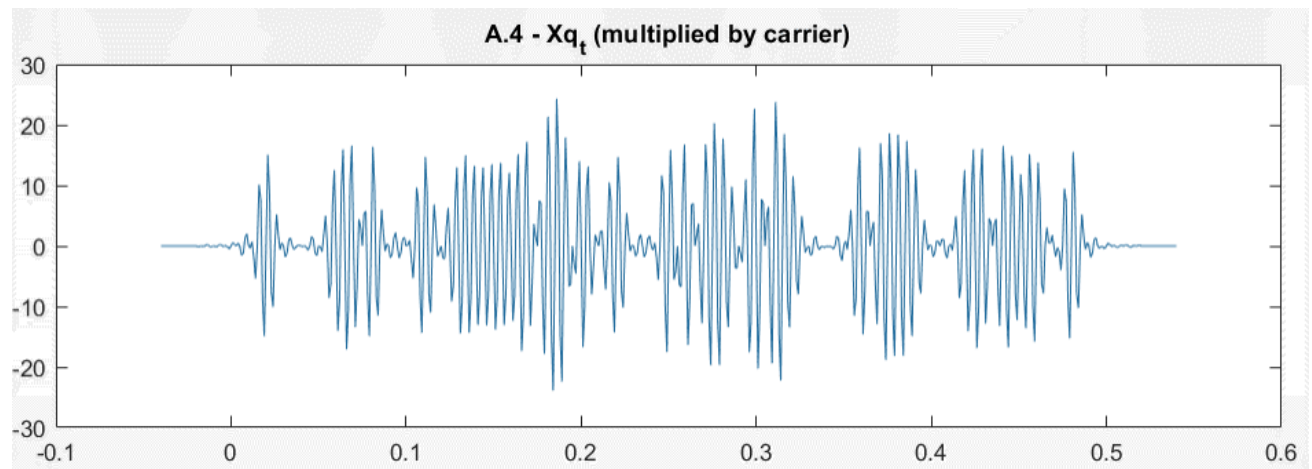
```

119 %Calculating Periodgrams
120 PXi_t = P_X(t_Xi_conv, Xi_t,Nf,Ts);
121 PXq_t = P_X(t_Xq_conv, Xq_t,Nf,Ts);
122
123 %Plotting output diagrams
124 %Xi_t
125 set_diag_layout(fig, r, c, sub_fig);
126 c_plot(t_Xi_conv,Xi_t);
127 set_diag_labels("A.4 - Xi_t (multiplied by carrier)","t(s)","");
128 grid on;
129 %Xq_t
130 sub_fig = sub_fig + 1;
131 set_diag_layout(fig, r, c, sub_fig);
132 c_plot(t_Xq_conv,Xq_t);
133 set_diag_labels("A.4 - Xq_t (multiplied by carrier)","t(s)","");
134 grid on;
135 %PXi
136 sub_fig = sub_fig + 1;
137 set_diag_layout(fig, r, c, sub_fig);
138 c_semilogy(F_ES,PXi_t);
139 set_diag_labels("A.4 - Periodgram of Xi_t,n (Logarithmic)","F [Hz]","");
140 grid on;
141 %PXq
142 sub_fig = sub_fig + 1;
143 set_diag_layout(fig, r, c, sub_fig);
144 c_semilogy(F_ES,PXq_t);
145 set_diag_labels("A.4 - Periodgram of Xq_t,n (Logarithmic)","F [Hz]","");
146 grid on;

```

## Διαγράμματα







## A.5

Συνεχίζουμε υπολογίζοντας την είσοδο του καναλιού  $X(t)$  η οποία ισοδυναμεί με το άθροισμα των σημάτων  $\mathbf{X_i\_t}$  και  $\mathbf{Xq\_t}$ . Συγκεκριμένα έχουμε:



### Τμήμα Κώδικα

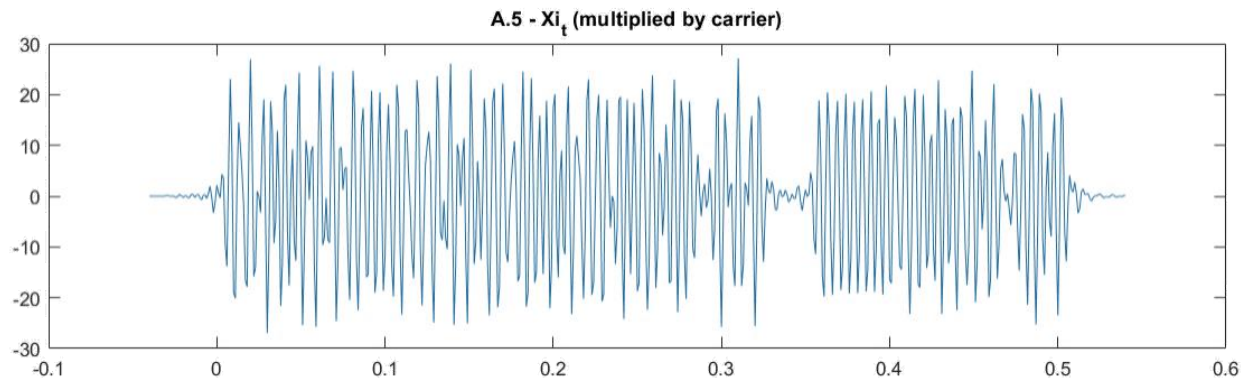
```

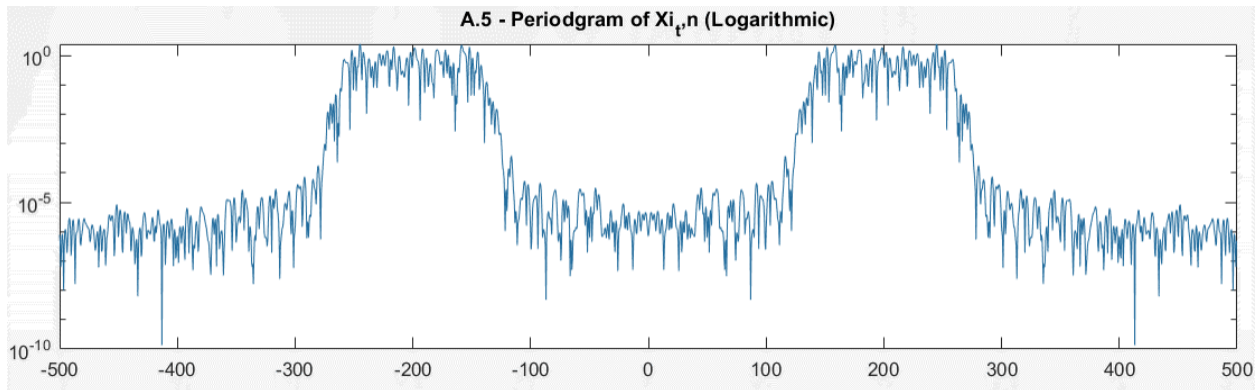
148  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
149  % A5 - Multiplication of Xi, Xq by carrier
150  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
151  fig = fig + 1; sub_fig = 1; r = 2; c = 2; %New Figure
152
153  X = Xi_t + Xq_t;
154
155  PX = P_X(t_Xi_conv, X,Nf,Ts);
156
157  %Plotting output diagrams
158  %X
159  set_diag_layout(fig, r, c, sub_fig);
160  c_plot(t_Xi_conv,X);
161  set_diag_labels("A.5 - Xi_t (multiplied by carrier)","t(s)","");
162  grid on;
163  %PX
164  sub_fig = sub_fig + 1;
165  set_diag_layout(fig, r, c, sub_fig);
166  c_semilogy(F_ES,PX);
167  set_diag_labels("A.5 - Periodogram of Xi_t,n (Logarithmic)","F [Hz]","");
168  grid on;

```



### Διαγράμματα





## A.6

Υποθέτουμε ότι το κανάλι είναι ιδανικό, συνεπώς το σήμα παραμένει αμετάβλητο (χωρίς αλλοίωση). Το σύστημα έχει δηλαδή κρουστική απόκριση  $\delta(t)$ .

## A.7

Προσθέτουμε τώρα λευκό θόρυβο Gaussian, έτσι ώστε να προσωμοιώσουμε «ρεαλιστικά» το σύστημα, το οποίο έχει διασπορά:

$$\sigma_w^2 = F_s \cdot 10^{\left(-\frac{SNR}{10}\right)}$$

Αναθέτουμε την ενδεικτική τιμή 10 για SNR:

## Τμήμα Κώδικα

```

170  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
171  % A7 - White Gassuan Noise %
172  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
173  SNR = 10; %Setting up signal to noise ratio
174  sw = Fs * (10^(-(SNR/10)));
175  wn= randn(1,length(X)) .* sqrt(sw);
176  X_wn = X + wn;

```



## A.8

Παρόμοια με το ερώτημα A.4, αποδιαμορφώνουμε το σήμα πολλαπλασιάζοντας την ενθόρυβη κυματομορφή με τους αντίστοιχους φορείς και απεικονίζοντας τα αποτελέσματα:



### Τμήμα Κώδικα

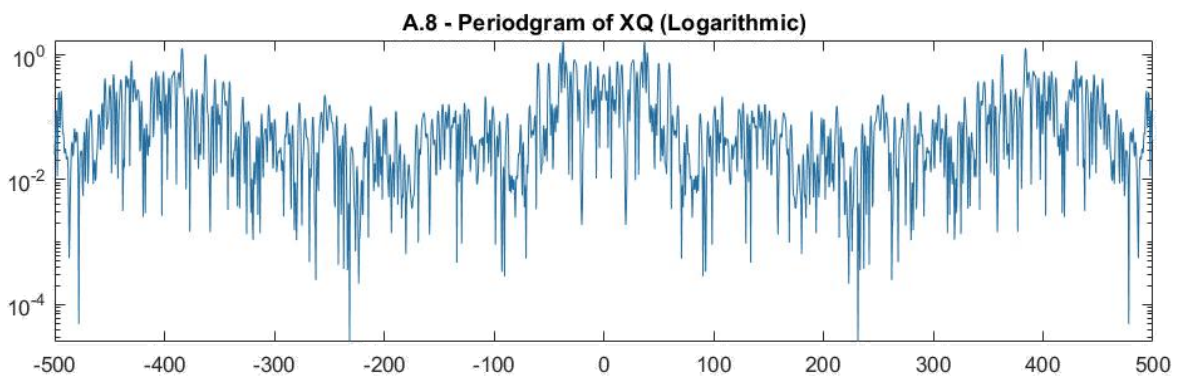
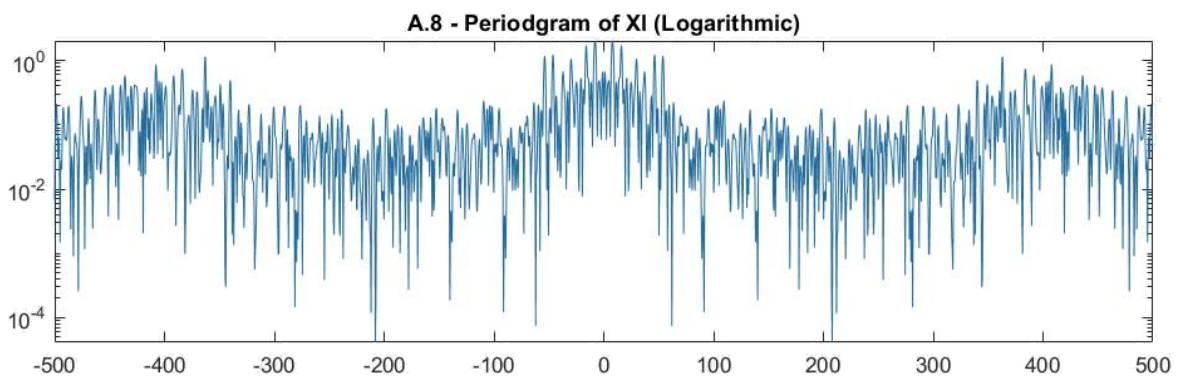
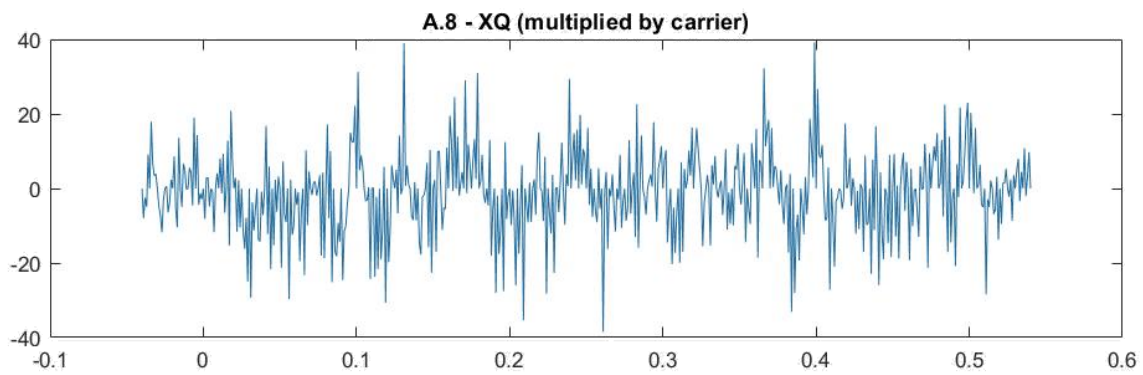
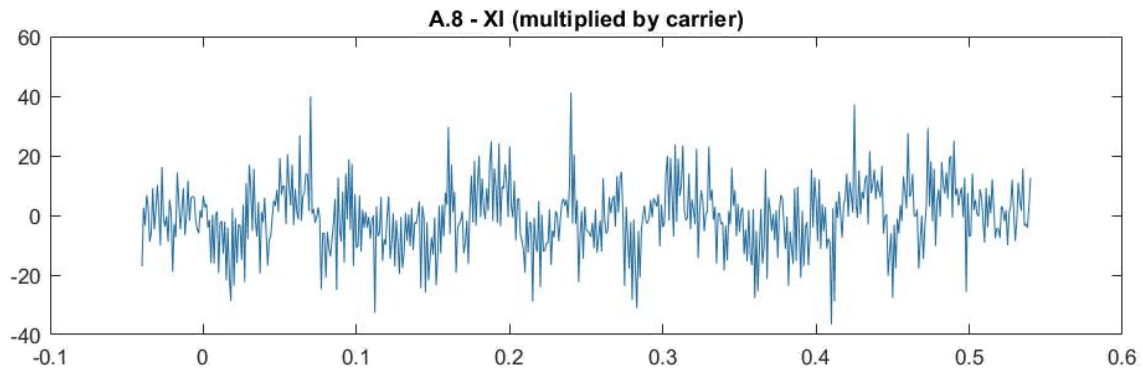
```

179 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
180 % A8 - Multiplication of X by carrier %
181 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
182 - fig = fig + 1; sub_fig = 1; r = 2; c = 2; %New Figure
183
184 - XI = X .* cos(2*pi*F0*t_Xi_conv);
185 - XQ = X .* cos(2*pi*F0*t_Xq_conv);
186
187 %Calculating Periodgrams
188 - PXI = P_X(t_Xi_conv, XI,Nf,Ts);
189 - PXQ = P_X(t_Xq_conv, XQ,Nf,Ts);
190
191 %Plotting output diagrams
192 %XI
193 - set_diag_layout(fig, r, c, sub_fig);
194 - c_plot(t_Xi_conv,XI);
195 - set_diag_labels("A.8 - XI (multiplied by carrier)","t(s)","");
196 - grid on;
197 %XQ
198 - sub_fig = sub_fig + 1;
199 - set_diag_layout(fig, r, c, sub_fig);
200 - c_plot(t_Xq_conv,XQ);
201 - set_diag_labels("A.8 - XQ (multiplied by carrier)","t(s)","");
202 - grid on;
203 %PXI
204 - sub_fig = sub_fig + 1;
205 - set_diag_layout(fig, r, c, sub_fig);
206 - c_semilogy(F_ES,PXI);
207 - set_diag_labels("A.8 - Periodgram of XI (Logarithmic)","F [Hz]","");
208 - grid on;
209 %PXQ
210 - sub_fig = sub_fig + 1;
211 - set_diag_layout(fig, r, c, sub_fig);
212 - c_semilogy(F_ES,PXQ);
213 - set_diag_labels("A.8 - Periodgram of XQ (Logarithmic)","F [Hz]","");
214 - grid on;

```



## Διαγράμματα





## A.9

Φιλτράρουμε και πάλι τα σήματα με παλμούς SRRC και οι κυματομορφές και τα περιοδογράμματα φαίνονται στα επόμενα σχήματα:



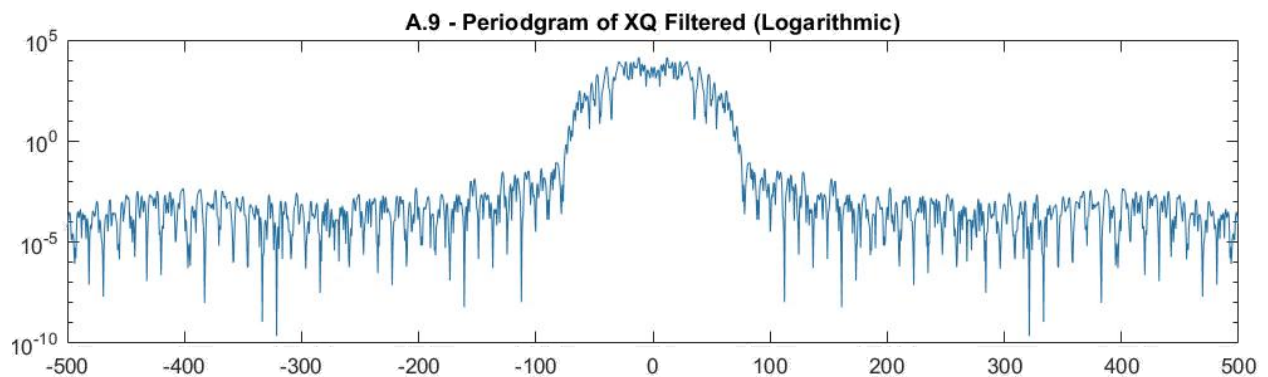
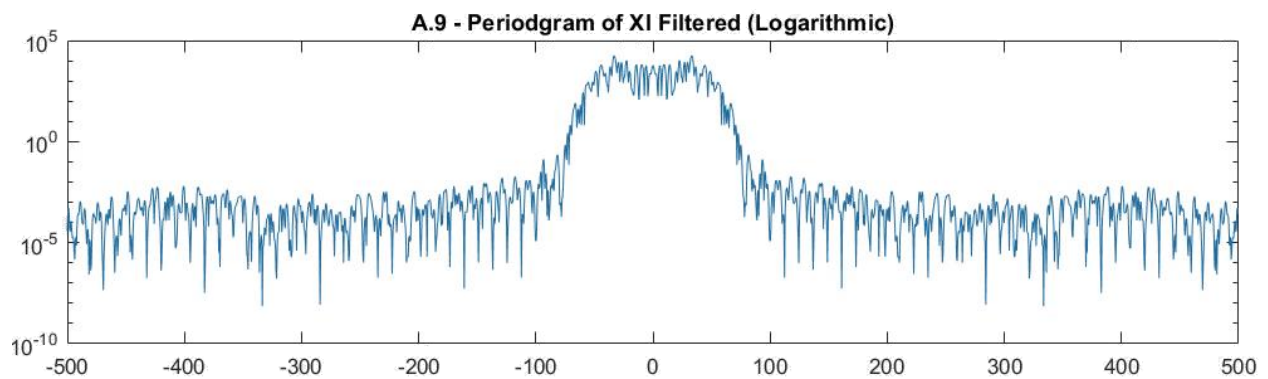
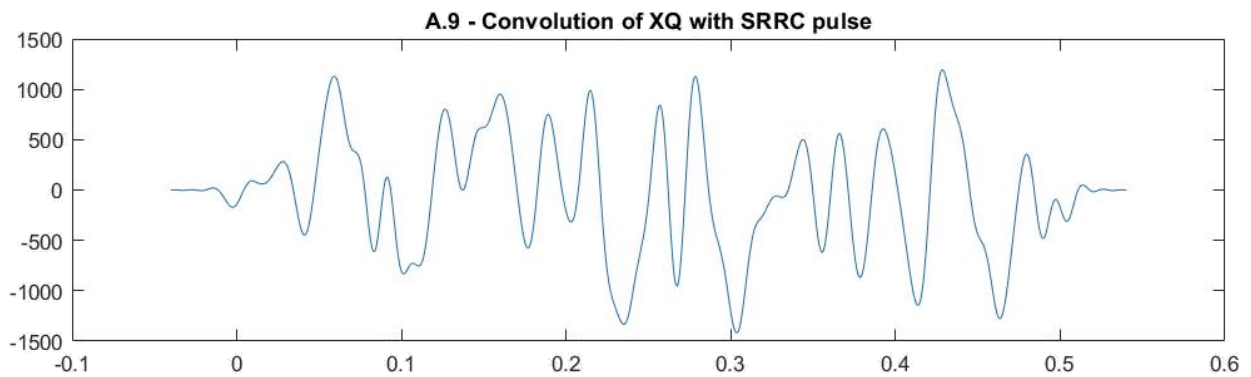
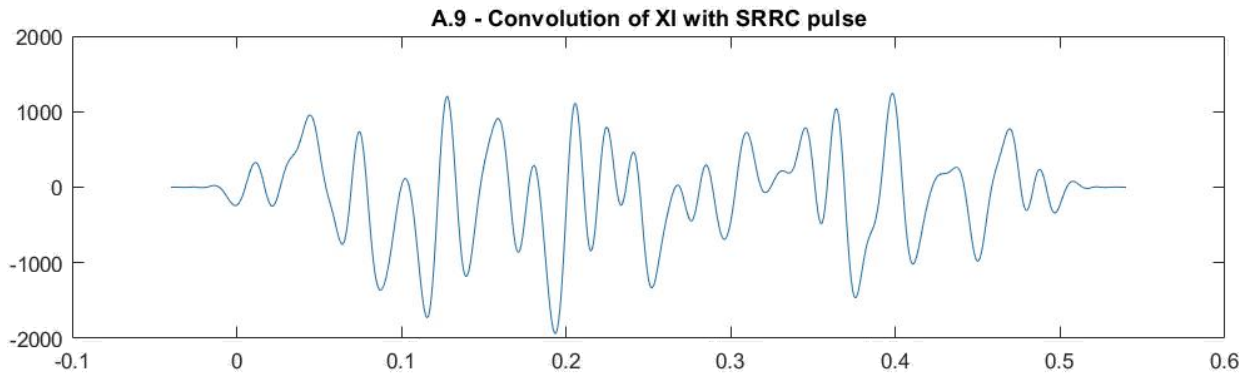
### Τμήμα Κώδικα

```

216 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
217 % A9 - Filtering XI & XQ with SRRC Pulses %
218 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
219 fig = fig + 1; sub_fig = 1; r = 2; c = 2; %New Figure
220
221 XI_conv = conv(XI, phi)*Ts;
222 XQ_conv = conv(XQ, phi)*Ts;
223
224 t_XI_conv = t_conv(t_phi, t_Xi, Ts);
225 t_XQ_conv = t_conv(t_phi, t_Xq, Ts);
226
227 %Calculating Periodgrams
228 F_ES = (-Fs/2) : (Fs/Nf) : (Fs/2 - Fs/Nf);
229 PXI_2 = P_X(t_XI_conv, XI_conv,Nf,Ts);
230 PXQ_2 = P_X(t_XQ_conv, XQ_conv,Nf,Ts);
231
232 %Plotting output diagrams
233 %XI_conv
234 set_diag_layout(fig, r, c, sub_fig);
235 c_plot(t_XI_conv,XI_conv);
236 set_diag_labels("A.9 - Convolution of XI with SRRC pulse","t(s)","");
237 grid on;
238 %XQ_conv
239 sub_fig = sub_fig + 1;
240 set_diag_layout(fig, r, c, sub_fig);
241 c_plot(t_XQ_conv,XQ_conv);
242 set_diag_labels("A.9 - Convolution of XQ with SRRC pulse","t(s)","");
243 grid on;
244 %PXI_2
245 sub_fig = sub_fig + 1;
246 set_diag_layout(fig, r, c, sub_fig);
247 c_semilogy(F_ES,PXI_2);
248 set_diag_labels("A.9 - Periodgram of XI Filtered (Logarithmic)","F [Hz]","");
249 grid on;
250 %PXQ_2
251 sub_fig = sub_fig + 1;
252 set_diag_layout(fig, r, c, sub_fig);
253 c_semilogy(F_ES,PXQ_2);
254 set_diag_labels("A.9 - Periodgram of XQ Filtered (Logarithmic)","F [Hz]","");
255 grid on;

```





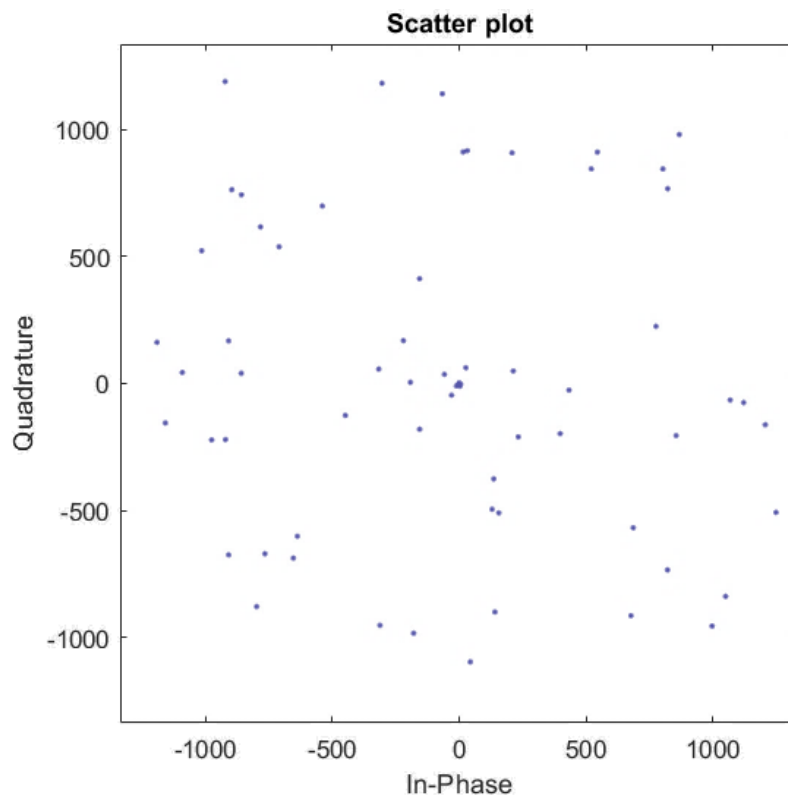
## A.10

Χρησιμοποιώντας τη συνάρτηση **scatterplot** του matlab, απεικονίζουμε την ακολουθία εξόδου Υ. Παρατηρούμε οτι έχουμε σχετικά μεγάλες αποκλίσεις αλλά αυτό κυρίως οφείλεται στη μικρή σχετικά τιμή του SNR και στις λίγα δείγματα των σημάτων (Μικρό Ν).

### Τμήμα Κώδικα

```
257 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
258 % A10 - Using Scatterplot to sample plotted signals %
259 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
260 cout = 1;
261
262 for i = 1:over:length(t_XI_conv)
263     X_wn(count,1) = XI_conv;
264     X_wn(count,2) = XQ_conv;
265     count = count + 1;
266 end
267 Y = [XI_conv XQ_conv];
268 scatterplot(Y');
```

### Διαγράμματα





## A.11

Κατασκευάστηκε τη συνάρτηση `[est_X,est_bit_seq] = detect_PSK_8(Y,A)` η οποία κάνει χρήση του nearest neighbor algorithm έτσι ώστε να παράξει μια εκτιμώμενη ακολουθία PSK.



### Τμήμα Κώδικα (Κύριο Μέρος)

```
270 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
271 % A11 - Detecting PSK Sequence %
272 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
273 [est_X, est_bit_seq] = detect_PSK_8(Y,A);
```



### Τμήμα Κώδικα (Συνάρτηση)

```
1 function [est_X,est_bit_seq] = detect_PSK_8(Y,A)
2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3 % [est_X,est_bit_seq] = detect_PSK_8(Y,A) %
4 % OUTPUT %
5 %     est_X: estimated symbols %
6 %     est_bit_seq: estimated bit sequence %
7 % INPUT %
8 %     Y: PSK Sequence %
9 %     A: Amplitude %
10 % %
11 % M. Galanis, Dec. 2018 %
12 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
13 est_X = zeros(1, length(Y));
14 m_length = 2^3;
15
16 possible_values_i = zeros(1,m_length);
17 for m = 0: m_length - 1
18     possible_values_i(m + 1) = A*cos((m*pi)/4);
19 end
20
21 possible_values_q = zeros(1,m_length);
22 for m = 0 : m_length - 1
23     possible_values_q(m + 1) = A*sin((m*pi)/4);
24 end
25
26 for i = 1 : length(Y)
27     distances = zeros(1,m_length);
28     for m = 1 : m_length
29         if (i < length(Y) / 2)
30             distances(m) = sqrt((Y(i,1) - possible_values_i(m))^2 + (Y(i,1) - possible_values_i(m))^2);
31         else
32             distances(m) = sqrt((Y(i,1) - possible_values_q(m))^2 + (Y(i,1) - possible_values_q(m))^2);
33         end
34     end
35
36     index = 1;
37     for m = 1 : m_length
38         if (distances(m) == min(distances))
39             est_X(i) = distances(m);
40             index = m;
41         end
42     end
```

```
44 -     b = de2bi(est_X(index));
45 -     est_bit_seq(3 * i) = b(1);
46 -     est_bit_seq(3 * i + 1) = b(2);
47 -     est_bit_seq(3 * i + 2) = b(3);
48 - end
49 - return
```