

ΤΕΧΝΟΛΟΓΙΚΟ ΕΠΙΔΕΚΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΟΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΙΑΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΕΦΑΡΜΟΓΗ ΤΗΣ ΤΕΧΝΙΚΗΣ OFDM ΣΕ
ΣΥΣΤΗΜΑ ΕΠΙΚΟΙΝΩΝΙΩΝ

ΜΠΟΥΓΙΟΥΚΟΣ ΝΙΚΟΛΑΟΣ

ΕΠΟΠΤΕΥΟΝ ΚΑΘΗΓΗΤΗΣ : ΑΝΑΣΤΑΣΙΟΣ ΔΡΟΣΟΠΟΥΛΟΣ

ΠΑΤΡΑ 2014

ΠΡΟΛΟΓΟΣ

Η παρούσα πτυχιακή εργασία επινοήθηκε στα πλαίσια των προπτυχιακών μου σπουδών στο ΤΕΙ δυτικής Ελλάδας στο τμήμα ηλεκτρολογίας. Θα ήθελα να ευχαριστήσω τους καθηγητές μου για τις γνώσεις που μου παρείχαν καθ' όλη τη διάρκεια των σπουδών μου. Ιδιαίτερα, θα ήθελα να ευχαριστήσω τον καθηγητή μου και επιβλέποντα της πτυχιακής εργασίας κ. Αναστάσιο Δροσόπουλο για την υπόδειξη του θέματος, την καθοδήγηση του , τις συμβουλές του και για την εποικοδομητική συνεργασία που είχαμε. Επίσης, θα ήθελα να ευχαριστήσω τη μαθηματικό και φίλη μου κ. Ελένη Κακαγιάννη για τις συμβουλές της και τη πολύτιμη βοήθειά της κατά την διεκπεραίωση της εργασίας. Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου, που στήριξε με κάθε τρόπο τις προσπάθειες μου σε όλη τη διάρκεια της φοίτησής μου.

Περιεχόμενα

<u>Περίληψη</u>	<u>3</u>
<u>Έννοια και εξέλιξη των τηλεπικοινωνιών</u>	<u>4</u>
<u>Εισαγωγή στο OFDM</u>	<u>5</u>
<u>Fading διάλειψης</u>	<u>7</u>
<u>Rayleigh fading</u>	<u>10</u>
<u>Rician fading</u>	<u>14</u>
<u>Weibull fading</u>	<u>15</u>
<u>Διασυμβολική παρεμβολή (ISI)</u>	<u>15</u>
<u>Κριτήριο Nyquist ISI</u>	<u>17</u>
<u>Σχεδιασμός και προσομοίωση OFDM</u>	<u>19</u>
<u>Προετοιμασία ανάλυσης OFDM</u>	<u>24</u>
<u>Βασικό demo OFDM</u>	<u>27</u>
<u>Δεύτερη επίδειξη OFDM (large)</u>	<u>30</u>
<u>Δεύτερη επίδειξη OFDM (small)</u>	<u>33</u>
<u>Πρόγραμμα εξομοίωσης σε MATLAB</u>	<u>35</u>
<u>Βιβλιογραφία</u>	<u>76</u>

ΠΕΡΙΛΗΨΗ

Ο χορός των τηλεπικοινωνιών και κυρίως των ασύρματων συστημάτων έχει μια ραγδαία εξέλιξη τα τελευταία χρόνια. Διαρκώς, όλο και περισσότερα τηλεπικοινωνιακά προϊόντα εισβάλουν στην καθημερινότητα μας. Η τεχνολογία των ασύρματων τηλεπικοινωνιών αναζητά τρόπους ώστε να ικανοποιηθούν οι απαιτούμενες ενέργειες για τη μετάδοση της πληροφορίας. Καθώς τα συστήματα επικοινωνίας έχουν αυξήσει την ταχύτητα μετάδοσης της πληροφορίας, ο χρόνος μετάδοσης αναγκαστικά έχει γίνει μικρότερος. Δεδομένου ότι ο χρόνος καθυστέρησης παραμένει σταθερός, οι εξωτερικές παρεμβολές, θόρυβος, δημιουργούν έναν περιορισμό στην επικοινωνία υψηλού ρυθμού μεταφοράς δεδομένων. Έτσι, δημιουργήθηκε η ανάγκη για την ανακάλυψη ενός λογισμικού, που θα περιόριζε τις επιπτώσεις του θορύβου και θα μετέφερε μεγάλο αριθμό δεδομένων στο διαθέσιμο φάσμα με υψηλή ταχύτητα.

Στην εργασία, αρχικά, δίνετε μια εισαγωγή στον χώρο των τηλεπικοινωνιών και στις διάφορες παρεμβολές, όπως η ISI, φαινόμενο των διαλείψεων fading και δίνεται έμφαση στα φαινόμενα Rayleigh fading, Rician fading, Weibull fading, καθώς και το κριτήριο Ευσταθίας Nyquist. Ένα πρόγραμμα έχει δημιουργηθεί σε matlab για να διερευνήσει την ορθογώνια πολυπλεξία, με διαίρεση συχνότητας συστημάτων επικοινωνίας. Το πρόγραμμα αυτό, μας δίνει τη δυνατότητα να αναλύσουμε συστήματα που είναι αρκετά πολύπλοκα σε θεωρητική βάση. Εφαρμόζει τετραγωνική διαμόρφωση πλάτους σήματος (QAM) και συγκρίνει με OFDM για να αποδείξει την ισχύ στα κανάλια πολλαπλών διαδρομών. Εξηγεί γιατί το OFDM είναι ανθεκτικό στην παρεμβολή ISI, καθώς μπορεί να στείλει ένα μεγάλο όγκο δεδομένων με πολύ χαμηλή ταχύτητα μετάδοσης, αποφεύγοντας έτσι την παρεμβολή ISI. Ο κύριος στόχος της πτυχιακής εργασίας ήταν η εφαρμογή του OFDM σε ένα ηχητικό σήμα και η προσομοίωση του.

Εννοια και εξέλιξη τηλεπικοινωνιών

Η έμφυτη ανάγκη του ανθρώπου για επικοινωνία τον οδήγησε στην αξιοποίηση της τεχνολογίας, με ιδιαίτερα ευρηματικό τρόπο, προκειμένου να εξαλείψει την απόσταση και να πετύχει τρόπους απομακρυσμένης επικοινωνίας. Μάλιστα, σε κάθε ιστορική περίοδο το πολιτιστικό επίπεδο προσδιορίζονταν και από μια ικανότητα διεξαγωγής γρήγορης και αξιόπιστης επικοινωνίας. Η λέξη τηλεπικοινωνία προέρχεται από το αρχαίο ελληνικό επίρρημα τηλε- που σημαίνει μακριά/από απόσταση και το ρήμα κοινώνω που σημαίνει μετέχω, έρχομαι σε επαφή.

Με τον ορό τηλεπικοινωνίες telecommunication αντιπροσωπεύεται η κάθε μορφή ηλεκτρομαγνητικής ενσύρματης ή ασύρματης, οπτικής ή ακουστικής επικοινωνίας που πραγματοποιείται ανεξαρτήτως απόστασης. Αυτή η διαδικασία πλέον γίνεται με την αποστολή ηλεκτρομαγνητικών κυμάτων, Αυτήν τη διαδικασία πραγματοποιούν κατάλληλες συσκευές όπως το τηλέφωνο ή ο ασύρματος. Οι πρώτες μορφές τηλεπικοινωνιών πραγματοποιούνται με την χρήση φωτιάς. Ο Όμηρος αναφέρει πώς οι αχαιοί, χρησιμοποιούσαν τις φρυκτωρίες, δηλαδή μεγάλες φωτιές στις κορυφές των βουνών για να αναγγείλουν την πτώση της Τρώας στις Μυκήνες. Η φωτιά ως μέσο επικοινωνίας χρησιμοποιείτο μέχρι τα μέσα του 19 αιώνα.

Κομβικό σημείο στις τηλεπικοινωνίες αποτελεί τον 19^ο αιώνα η θεμελίωση της ηλεκτρομαγνητικής θεωρίας από τον James Clerk Maxwell, το 1864 κατάφερε να αποδείξει την ηλεκτρομαγνητική φύση του Φώτος. Η διάδοση του Φώτος και των ηλεκτρομαγνητικών κυμάτων γίνεται με κυματικές διαταραχές ίδιας συχνότητας. Βασισμένος σε αυτήν τη θεωρία ο Marconi υλοποίησε για πρώτη φορά το 1897 τον τηλεγράφο ένα σύστημα ασύρματης μετάδοσης. Ο Marconi βασίστηκε στην θεωρία του Maxwell και κατασκεύασε μια διάταξη στην οποία γραπτά σύμβολα μεταδίδονται από τον ένα σταθμό στον άλλο με τη βοήθεια του ηλεκτρικού ρεύματος. Στις αρχές του 20^ο αιώνα μεταδοθήκαν σήματα μέσω τηλεγράφου σε εξαιρετικά μεγάλες αποστάσεις ενώ στην πορεία μεταδόθηκε και φωνή με τη βοήθεια ενισχυτικών λυχνιών. Η πρώτη ασύρματη ζεύξη βρήκε αρχικά εφαρμογή στις ΗΠΑ σε αστυνομικά τμήματα και μετέδιδε στους 2 μεγακύκλους, ενώ αρχικά χρησιμοποιήθηκε διαμόρφωση πλάτους (amplitude modulation-AM).

Η παρουσίαση το 1935 της διαμόρφωσης συχνότητας (frequency modulation-FM) από τον Edwin Armstrong έδωσε νέα ώθηση στα ασύρματα συστήματα επικοινωνίας. Στα τέλη της δεκαετίας του '30 η FM διαμόρφωση γίνεται η κύρια τεχνική διαμόρφωσης στα κινητά συστήματα επικοινωνίας στον κόσμο. Η ζήτηση για υπηρεσίες ασύρματων επικοινωνιών άρχισε να αυξάνεται. Έτσι, η AT&T BELL LABORATORIES υλοποίησε το improved mobile telephone service (IMTS). Το IMTS ήταν το πρώτο σύστημα κινητών επικοινωνιών το οποίο συνδεόταν με δίκτυο δημόσιας επικοινωνίας και χρησιμοποιούσε πλήρως αμφίδρομες ζεύξεις (full duplex).

Τα συστήματα τηλεπικοινωνιών σχεδιάζονται εν γένει από τους μηχανικούς τηλεπικοινωνιών. Οι πρώτοι εφευρέτες στον τομέα ήταν ο Ελίσια Γκρέι, ο Νικόλα Τέσλα, ο Γκουλιέλμο Μαρκόνι και ο Τζον Λόγκι Μπερντ. Τα τηλεπικοινωνιακά τους συστήματα ήταν αναλογικά αλλά κατά τις τελευταίες δεκαετίες του εικοστού αιώνα επικράτησαν καθολικά τα ψηφιακά συστήματα, καταγόμενα από τα δίκτυα υπολογιστών. Έτσι, οι περισσότερες τηλεπικοινωνιακές

συσκευές σήμερα είναι ηλεκτρονικοί υπολογιστές, ακόμα και αν δεν έχουν την εμφάνιση ενός τυπικού μικροϋπολογιστή γενικής χρήσης.

Εισαγωγή στο OFDM

Το OFDM (orthogonal frequency division multiplexing) σημαίνει “ πολυπλεξία με ορθογώνια διαίρεση συχνότητας. Είναι ένας τύπος διαμόρφωσης, σύμφωνα με τον οποίο η πληροφορία που προέρχεται από μια μόνο πηγή χωρίζεται σε πόλους φορείς (sub carriers). Αυτό εξυπηρετεί στην καταπολέμηση των επιπτώσεων των πολλαπλών διαδρομών και στο να προστεθεί μια διάσταση διαφορετικότητας στο σήμα εκπομπής.

Η βασική αρχή του OFDM είναι να διαιρεί υψηλούς ρυθμούς μετάδοσης σε χαμηλότερους ρυθμούς που μεταδίδονται ταυτόχρονα πέρα από έναν υπό-φορέα. Η ενδοσυμβολική παρεμβολή (intersymbol interference-isi) αποβάλλεται σχεδόν τελείως με την εισαγωγή ενός χρόνου φρουράς σε κάθε σύμβολο OFDM. Στο διάστημα φρουράς, το σύμβολο OFDM αποφεύγει την παρεμβολή (intercarrier interference ici). Για τη σχεδίαση συστημάτων OFDM λαμβάνονται υπόψη διάφοροι παράμετροι όπως ο αριθμός των υπο-φορέων, ο χρόνος φρουράς, η διάρκεια του συμβόλου, το διάστημα των υπο-φορέων, ο τύπος διαμόρφωσης ανά υπο-φορέα. Η επιλογή των παραμέτρων επηρεάζεται από την απόκλιση συστημάτων, όπως είναι το διαθέσιμο εύρος ζώνης και ο απαραίτητος ρυθμός δυαδικών ψηφίων. Ορθογώνια είναι η τεχνική που διανέμει δεδομένα σε ένα μεγάλο αριθμό φορέων που απέχουν μεταξύ τους σε μεγάλες συχνότητες. Η τεχνική αυτή βοηθά τους αποδιαμορφωτές να μην βλέπουν τις άλλες συχνότητες. Διαίρεση συχνότητας υπό την έννοια ότι οι υπο-φορείς συχνότητας μεταφέρουν ασύνδετα με σταθερή απόσταση και τύπο διαμόρφωσης. Πολυπλεξία με την έννοια που συντίθεται σε ένα ενιαίο κανάλι. Στη διόρθωση σφάλματος η κωδικοποίηση είναι απαραίτητη επειδή μερικοί υποφορείς μπορεί να είναι σε μεγάλη εξασθένιση. Έτσι, η κωδικοποίηση όλων των bit με διαπλοκή είναι απαραίτητη.

Η πτυχιακή θα επικεντρωθεί στην OFDM της έρευνας και της προσομοίωσης. Το OFDM είναι ιδιαίτερα κατάλληλο για επικοινωνία υψηλής ταχύτητας λόγω της αντίστασης στην ISI, καθώς τα συστήματα επικοινωνίας έχουν αυξήσει την ταχύτητα μεταφοράς πληροφοριών ο χρόνος για κάθε μετάδοση αναγκαστικά γίνεται μικρότερος. Ο χρόνος καθυστέρησης δεδομένου ότι προκαλείται από πολλαπλές διαδρομές παραμένει σταθερός, το ISI είναι μεγάλο πρόβλημα για υψηλό ρυθμό μεταφοράς δεδομένων. Το OFDM αποφεύγει αυτό το πρόβλημα με την αποστολή πολλών μεταδόσεων χαμηλής ταχύτητας ταυτόχρονα.

Το OFDM ξεκίνησε στα τέλη 1950 αφού είχε γίνει υποχρεωτική η ανάγκη για μετάδοση δεδομένων όπου η χρήση του φάσματος μετάδοσης θα ήταν ποιο αποδοτική και θα άφηνε εκτός το φαινόμενο των παρεμβολών του σήματος (θόρυβος). Καθοριστικό ρόλο για την ανάπτυξη του OFDM αποτέλεσε το (frequency-division multiplexing -FDM) ο Chang στα τέλη της δεκαετίας του '60 δημοσίευσε

την δομή του OFDM, σύμφωνα με την αρχή χρησιμοποίησης ορθογώνιων επικαλυπτόμενων πολυτονικών σημάτων για μεταφορά δεδομένων. Το 1960 η OFDM χρησιμοποιήθηκε σε διάφορα συστήματα του στρατού των ΗΠΑ συγκεκριμένα το KATHRYN χρησιμοποιήθηκαν πανό από 34 παράλληλα κανάλια με 84 hertz το καθένα. Για την εύκολη παραγωγή και λήψη OFDM σημάτων ο Weinstein πρότεινε το 1971 τη χρήση του διακριτού μετασχηματισμού Φουριέ(DFT),ο cimini το 1985 πρότεινε το OFDM για ασύρματες τηλεπικοινωνίες. Όμως, στις αρχές του '90 οι τεχνολογικές εξελίξεις κατέστησαν δυνατή την εφαρμογή του. Σήμερα η OFDM βρίσκει εφαρμογή σε :

- ADSL,SDSL και VDSL
- Wi-Fi(IEEE 802.11a/g)
- DAB systems
- DVB
- IEEE802.16 H WiMAX
- FLASH-OFDM
- Certain Ultra wideband (UWB)
- Power line communication
- MOCA home networking

Το OFDM έχει διαφορές έναντι των άλλων προτύπων όπως:

- ❖ Υψηλή φασματική απόδοση σε σύγκριση με άλλους δίπολους πλευρικής ζώνης σημάτων της διαμόρφωσης φάσματος.
- ❖ Μπορεί εύκολα να προσαρμοστεί στις δύσκολες συνθήκες του καναλιού χωρίς πολύπλοκες εξισώσεις στο πεδίο του χρόνου.
- ❖ Ανθεκτικό σε στενή ζώνη.
- ❖ Ανθεκτικό σε διασυμβολική παρεμβολή (ISI)και στις διαλείψεις που προκαλούνται από τη διάδοση πολλαπλών διαδρομών.
- ❖ Για αποτελεσματικότερη εφαρμογή χρησιμοποιεί Fast Fourier Transform(FFT)
- ❖ Χαμηλή ευαισθησία σε σφάλματα συγχρονισμού του χρόνου.
- ❖ Φίλτρα με δέκτη ανά κανάλι δεν απαιτούνται (σε αντίθεση με τα συμβατικά FDM).
- ❖ Διευκόλυνση δικτύων ενιαίας συχνότητας (SFN).

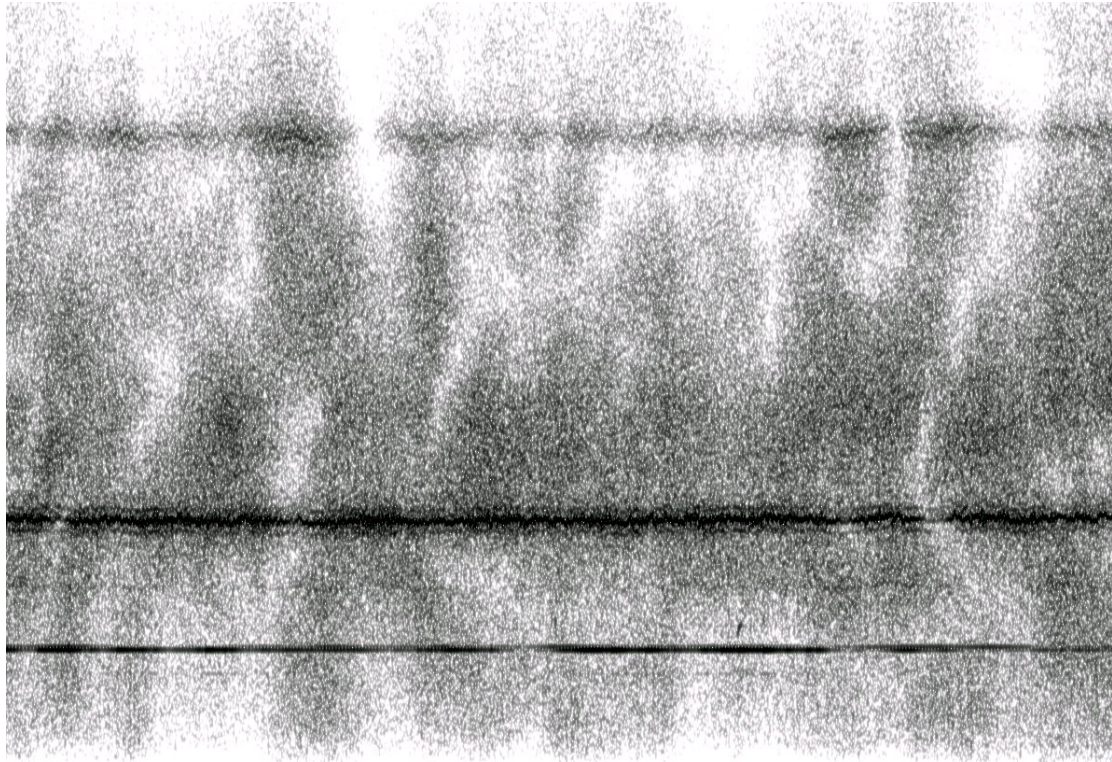
Αλλά και μειονεκτήματα που δημιουργούν προβλήματα:

- ❖ Ευαίσθητο στη Doppler μετατόπιση
- ❖ Ευερέθιστα σε προβλήματα συγχρονισμού συχνότητας.
- ❖ Υψηλή αναλογία peak-to-peak κατά το μέσο όρο-power (PAPR) πλήττεται από την κακή απόδοση ισχύος.
- ❖ Η ατέλεια της αποτελεσματικότητας που προκαλείται από κυκλικό πρόθεμα/διάστημα προστασίας.

Το OFDM σήμα αποτελείται από άθροισμα υπο-φορέων τα οποία είναι διαμορφωμένα από σύμβολα ψηφιακής διαμόρφωσης βασικής ζώνης όπως το QAM. Το QAM είναι ένας ψηφιακός διαμορφωτής σήματος. Μεταφέρει δυο ψηφιακά ρεύματα bit αυξομειώνοντας τα πλάτη των δύο κυμάτων με τη χρήση του κλαδώματος πλάτους (ASK) ψηφιακό σχήμα διαμόρφωσης. Τα δύο φέροντα κύματα είναι στις 90° μοίρες το ένα από το άλλο και έτσι ονομάζονται συστήματα τετραγωνισμού. Όπως όλοι οι διαμορφωτές σημάτων ο QAM μεταφέρει δεδομένα αλλάζοντας κάποια πτυχή ενός φέροντος σήματος το οποίο συνήθως είναι ημιτονοειδές. Όπως σε πολλά συστήματα ψηφιακής διαμόρφωσης, το διάγραμμα διατάσσεται σε τετραγωνικό πλέγμα με ίσιες, κάθετες και οριζόντιες τετραγωνικές αποστάσεις. Δεδομένου ότι σε ψηφιακές τηλεπικοινωνίες τα στοιχεία είναι σε δυαδικό, ο αριθμός των κουκίδων στο πλέγμα θα είναι δύναμη του 2 (2-4-8,...). Οι πιο κοντινές μορφές είναι 16-QAM, 64-QAM και 256-QAM.

Φαινόμενο fading (διαλείψεις)

Στις ασύρματες επικοινωνίες οι διαλείψεις είναι η εξασθένιση που επηρεάζει ένα σήμα για ορισμένη διάδοση. Η εξασθένιση μπορεί να μεταβάλλεται με το χρόνο και τη γεωγραφική θέση ραδιοσυχνοτήτων. Στα ασύρματα συστήματα, οι διαλείψεις μπορεί να οφείλονται είτε στη διάδοση μέσω πολλαπλών διαδρομών, που αναφέρεται ως multipath, είτε λόγω σκίασης από εμπόδια που επηρεάζουν τη διάδοση κυμάτων και μερικές φορές αναφέρεται ως σκιά. Η διάλειψη είναι η παραμόρφωση ενός διαμορφωμένου σήματος λόγω μετάδοσης του στο ασύρματο περιβάλλον. Η προσομοίωση μιας τέτοιας μετάδοσης γίνεται με τη μοντελοποίηση ενός καναλιού διάλειψης (fading channel). Μαθηματικά, συνήθως οι διαλείψεις μοντελοποιούνται με τυχαία διακύμανση του πλάτους και της φάσης του μεταδιδόμενου κύματος που διαφέρει στο χρόνο από το αρχικό.



Συχνοεπιλεκτικό χρονικά μεταβαλλόμενο fading προκαλεί ένα βάθος μοτίβο για να εμφανιστεί σε ένα φασματογράφημα

Η παρουσία των ανακλαστήρων στο περιβάλλον γύρω από ένα πομπό και δέκτη δημιουργούν πολλαπλές διαδρομές όπου ένα μεταδιδόμενο σήμα μπορεί να διαβεί. Αυτό έχει ως αποτέλεσμα ο δεκτής να βλέπει την υπέρθεση πολλών αντιγραφών του μεταδιδόμενου σήματος που διέρχονται με διαφορετική πορεία. Κάθε αντίγραφο σήματος έχει διαφορά στην εξασθένηση, στην καθυστέρηση και στη μετατόπιση της φάσης, ενώ ταξιδεύει στον χώρο από την πηγή στο δέκτη. Αυτή η διαδικασία μπορεί να οδηγήσει, είτε σε εποικοδομητική, είτε σε καταστρεπτική παρέμβαση, ενίσχυση ή εξασθένηση της ισχύος του σήματος που παρατηρήθηκε στον δέκτη. Ισχυρή καταστροφική παρέμβαση είναι συχνά ένα βαθύ fading και μπορεί να οδηγήσει σε προσωρινή βλάβη της επικοινωνίας, λόγω σοβαρής μύωσης στο κανάλι του λόγου σήματος ως προς τον θόρυβο.

Ένα κοινό παράδειγμα για βαθύ fading είναι η εμπειρία του να σταματήσει σε ένα φανάρι να ακούγεται μια εκπομπή στα FM, και ενώ το αμάξι κινηθεί ένα κλάσμα του μέτρου η εκπομπή να ξεκινά πάλι να μεταδίδεται. Η ατέλεια της εκπομπής οφείλεται από τη διακοπή σε ένα σήμα, όπου το σήμα αντιμετωπίζει ένα καταστροφικό fading. Κινητά τηλέφωνα μπορούν επίσης να παρουσιάσουν στιγμιαία τέτοια εξασθένηση. Τα fading μοντέλα τα χρησιμοποιούμε συχνά για να υποδείξουμε τις επιδράσεις της ηλεκτρομαγνητικής μετάδοσης πληροφοριών μέσω του αέρα. Fading μοντέλα καναλιού χρησιμοποιούνται επίσης και σε υποβρύχιες ακουστικές επικοινωνίες, με το μοντέλο στρέβλωσης να προκαλείται από το νερό. Από μια μαθηματική άποψη, η διάλεια μοντελοποιείται ως μια χρονικά μεταβαλλόμενη τυχαία άλγη στο πλάτος και στη φάση του μεταδιδόμενου σήματος.

Με τον όρο αργό η γρήγορο fading αναφερόμαστε στην ταχύτητα με την οποία το μέγεθος και η αλλαγή φάσης επιβάλλονται στο κανάλι από τις αλλαγές του σήματος. Ο χρόνος συνοχής είναι ένα μέτρο ελάχιστου χρόνου που απαιτείται για την αλλαγή μεγέθους ή της φάσης του καναλιού ώστε να μην σχετίζεται με την προηγούμενη τιμή της. Αργή διάλειαση προκύπτει όταν ο χρόνος συνοχής του καναλιού είναι μεγάλος σε σχέση με τον χρόνο καθυστέρησης του καναλιού. Σε αυτό το καθεστώς το πλάτος και η αλλαγή της φάσης που επιβάλλονται στο κανάλι μπορεί να θεωρηθεί κατά προσέγγιση σταθερή κατά την περίοδο χρήσης. Η αργή διάλειαση μπορεί να προκληθεί από συμβάντα όπως σκίαση όπου ένα μεγάλο εμπόδιο όπως λόφος ή μεγάλο κτίριο συσκοτίζει την κύρια διαδρομή του σήματος μεταξύ πομπού και δέκτη. Στη λαμβανόμενη ισχύς η αλλαγή που προκαλείται από την σκίαση μοντελοποιείται χρησιμοποιώντας μια λογαριθμική κατανομή με μια τυπική απόκλιση. Η γρήγορη διάλειαση συμβαίνει όταν ο χρόνος συνοχής του καναλιού είναι μικρός σε σχέση με τον χρόνο καθυστέρησης του καναλιού. Σε αυτό το καθεστώς το πλάτος και η αλλαγή της φάσης που επιβάλλονται από το κανάλι ποικίλλουν σημαντικά κατά την περίοδο χρήσης. Ένα γρήγορο κανάλι που εξασθενεί ο πομπός μπορεί να επωφεληθεί από τις διακυμάνσεις των συνθηκών του καναλιού χρησιμοποιώντας διαχωρισμό χρόνου για να βοηθήσει στην αύξηση της ανθεκτικότητας της επικοινωνίας σε ένα προσωρινό βαθύ fading. Παρά το γεγονός ότι μια βαθιά fading μπορεί να διαγράψει προσωρινά κάποια από τις πληροφορίες που μεταδίδονται, η χρήση ενός κώδικα διόρθωσης σφαλμάτων σε συνδυασμό με την επιτυχημένη μετάδοση bits κατά την διάρκεια του χρόνου σε άλλες περιπτώσεις (interleaving) μπορεί να επιτρέψει τα σβησμένα κομμάτια. Σε μια αργή fading δεν είναι δυνατή η χρήση ποικιλομορφίας χρόνου επειδή ο πομπός βλέπει μόνο ένα κανάλι μέσω του περιορισμού καθυστέρησης. Συνεπώς σε μια βαθιά διάλειαση που διαρκεί σε όλο το χρόνο μεταφοράς δεν μπορεί να μετριάσει με την χρήση κωδικοποίησης.

Ο χρόνος συνοχής του καναλιού σχετίζεται με μια ποσότητα που είναι γνωστή ως εξάπλωση Doppler. Όταν ο χρήστης κινείται η ταχύτητα του χρήστη προκαλεί μια μετατόπιση στην συχνότητα του σήματος που μεταδίδεται σε κάθε διαδρομή του σήματος. Αυτή η μετατόπιση είναι γνωστή ως μετατόπιση Doppler. Σήματα που ταξιδεύουν κατά μήκος με διαφορετικές διαδρομές μπορούν να έχουν διαφορετικές μετατοπίσεις, που αντιστοιχούν σε διαφορετικούς ρυθμούς μεταβολής της φάσης. Κανάλια με μεγάλη εξάπλωση Doppler έχουν συνιστώσες σήματος με διαφορετική αλλαγή στην φάση και στην πάροδο του χρόνου. Από τις διαλείψεις εξαρτάται κατά πόσο τα συστατικά του σήματος λειτουργούν εποικοδομητικά ή καταστροφικά. Σε γενικές γραμμές ο χρόνος συνοχής σχετίζεται αντιστρόφως με την εξάπλωση Doppler,

και εκφράζεται ως:

$$T_c \approx \frac{1}{D_s}$$

όπου T_c είναι ο χρόνος συνοχής, D_s είναι η εξάπλωση Doppler.

Επιλεκτική διάλειψη και η επιλεκτική εξασθένηση συχνότητας είναι ένα ραδιόφωνο πολλαπλασιασμού ανωμαλίας που προκαλείται από μερική κατάργηση ενός ραδιοφωνικού σήματος ή από μόνη της. Το σήμα φθίνει στον δεκτή από δυο διαφορετικά μονοπάτια και τουλάχιστον μια από τις διαδρομές Αλλά επιμηκύνεται ή επιβραδύνεται. Αυτό συμβαίνει συνήθως νωρίς το πρωί ή νωρίς το απόγευμα καθώς τα η διάδοση των διάφορων στρωμάτων που κινούνται στην ιονόσφαιρα γίνεται ξεχωριστά και δεν μπορούν να συνδυαστούν. Οι δυο διαδρομές μπορεί να είναι εναερόκυμα (skywave) ή επιγηόκυμα (groundwave). Καθώς η συχνότητα φορέας του σήματος μεταβάλλεται, το μέγεθος της μεταβολής του πλάτους ποικίλει. Το εύρος ζώνης συνοχής μετρά το διαχωρισμό της συχνότητας από την οποία τα δυο σήματα θα επηρεαστούν με άσχετες διαλείψεις.

Σε επίπεδη εξασθένηση το εύρος ζώνης συνοχής του καναλιού είναι μεγαλύτερο από το εύρος ζώνης του σήματος. Ως εκ τούτου όλες οι συνιστώσες συχνότητας του σήματος θα επηρεαστούν από το ίδιο μέγεθος αποσβέσεως. Στην επιλεκτική εξασθένηση συχνότητας το εύρος ζώνης συνοχής του καναλιού είναι μικρότερο από το εύρος ζώνης του σήματος. Ως εκ τούτου διαφορετικές συνιστώσες συχνότητας σήματος επηρεάζονται από άσχετες διαλείψεις. Δεδομένου ότι οι διαφορετικές συνιστώσες συχνότητας του σήματος επηρεάζονται ανεξάρτητα, είναι εξαιρετικά απίθανο όλα τα μέρη του σήματος να επηρεαστούν ταυτόχρονα από μια βαθιά εξασθένηση. Ορισμένα συστήματα διαμόρφωσης όπως OFDM και CDMA είναι κατάλληλα για τη χρήση αδιαφορισμού συχνότητας για να παρέχει ευρωστία στις διαλείψεις. Ο OFDM διαιρεί το σήμα ευρείας ζώνης σε πόλους, ευρέως διαδεδομένους υπό-φορείς στενής ζώνης, που ο καθένας εκτίθεται σε επίπεδη εξασθένηση και όχι σε επιλεκτική εξασθένηση συχνότητας.

Το αποτέλεσμα μπορεί να αντιμετωπιστεί με την εφαρμογή κάποιου συστήματος ποικιλομορφίας για παράδειγμα ένα OFDM με υποφέρουσα διαπλοκή προς τα εμπρός, ή με την χρήση δυο δεκτών με χωριστές κεραιές που απέχουν ένα τέταρτο-μήκους κύματος ή με ένα ειδικά σχεδιασμένο δίκτυο ποικιλομορφίας με δυο κεραιές. Ένας τέτοιος δέκτης συγκρίνει συνεχώς τα σήματα που φτάνουν στις δυο κεραιές και παρουσιάζουν καλύτερο σήμα.

Το fading έχει διαφορές μορφές και χωρίζεται σε μοντέλα ανάλογα με τον τρόπο που επηρεάζουν το σήμα. Τέτοια μοντέλα είναι:

- *Dispersive fading*
- Rayleigh fading
- Rician fading
- Weibull fading

Rayleigh fading

Rayleigh fading είναι ένα στατικό μοντέλο για την επίδραση του περιβάλλοντος στη διάδοση ενός ραδιοφωνικού σήματος που χρησιμοποιείται από ασύρματες συσκευές. Rayleigh fading μοντέλα υιοθετούν ένα σήμα που έχει πέραση μέσα από ένα μέσο μετάδοσης που ονομάζεται κανάλι επικοινωνίας. Σύμφωνα με την κατανομή Rayleigh η ακτινική συνιστώσα του αθροίσματος των δυο μεταβλητών είναι ασυσχέτιστες Gaussian τυχαίες μεταβλητές. Το Rayleigh fading θεωρείται ένα λογικό μοντέλο για την τροποσφαιρική και ιονοσφαιρική διάδοση του σήματος, καθώς και για των πυκνοδομημένων αστικών περιοχών στη διάδοση των ραδιοκυμάτων. Η απόσβεση της Rayleigh εφαρμόζεται όταν δεν υπάρχει κυρίαρχη μεταβλητή και μετακινείται κατά μήκος της γραμμής επαφής μεταξύ πομπού και δεκτή.

Η απόσβεση Rayleigh είναι ένα λογικό μοντέλο όταν υπάρχουν πολλά αντικείμενα στο περιβάλλον που σκεδάζουν το ραδιοφωνικό σήμα πριν φτάσει στον δεκτή. Το κεντρικό οριακό θεώρημα υποστηρίζει ότι αν υπάρχει αρκετά μεγάλη διασπορά στο κανάλι η κρουστική απόκριση μοντελοποιείται ως μια Gaussian διαδικασία ανεξάρτητα από την κατανομή. Αν δεν υπάρχει κυρίαρχο συστατικό στην διασπορά μια τέτοια διαδικασία θα είχε μηδενική μέση τιμή και η φάση θα ήταν ομοιόμορφα κατανομημένη μεταξύ 0 και 2π . Καλώντας μια τυχαία μεταβλητή R θα έχει μια συνάρτηση πυκνότητας:

$$p_R(r) = \frac{2r}{\Omega} e^{-r^2/\Omega}, \quad r \geq 0$$

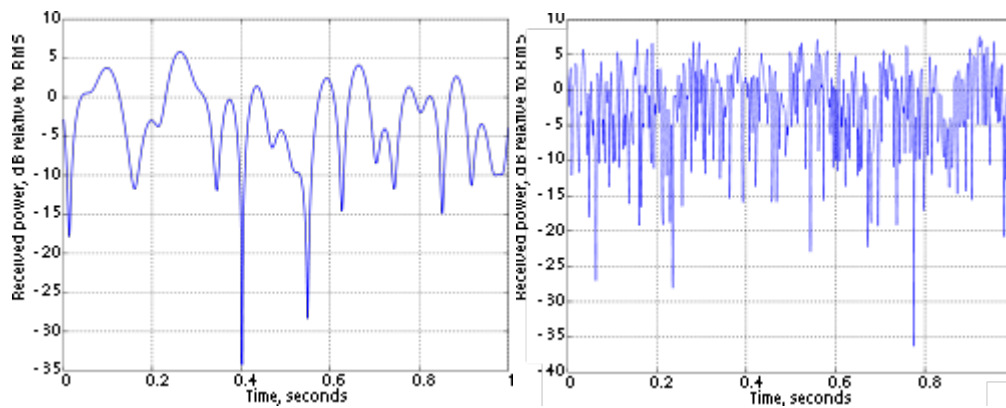
$$\text{Όπου } \Omega = E(R^2).$$

Συχνά, το κέρδος της φάσης στα στοιχεία στρέβλωσης ενός καναλιού αντιπροσωπεύεται από ένα σύνθετο αριθμό. Στην περίπτωση αποσβέσεως Rayleigh επιδεικνύεται ότι τα πραγματικά και τα φανταστικά τμήματα της απόκρισης μοντελοποιούνται από ανεξάρτητες και ταυτόσημα κατανομημένες μηδενικής μέσης τιμής gaussian διεργασίες ώστε το πλάτος της απόκρισης να είναι το άθροισμα των δυο αυτών διαδικασιών.

Η απαίτηση να υπάρχουν πολλοί σκεδαστές σημαίνει ότι η διάλειαση Rayleigh μπορεί να είναι χρήσιμο μοντέλο για πυκνοδομημένες περιοχές και κέντρα πόλεων όπου δεν υπάρχει καμία οπτική επαφή μεταξύ πομπού και δεκτή, καθώς πολλά κτίρια και διάφορα αντικείμενα εξασθενούν, αντανακλούν, και διαθλούν το σήμα. Σε τροποσφαιρική και ιονοσφαιρική διάδοση σήματος τα πολλά σωματίδια στα ανώτερα ατμοσφαιρικά στρώματα ενεργούν ως σκεδαστές και αυτό το είδος περιβάλλοντος μπορεί να προσέγγισει διάλειαση Rayleigh. Αν το περιβάλλον είναι τέτοιο ώστε εκτός από τη σκέδαση υπάρχει και έντονο κυρίαρχο σήμα, φαίνεται στον

δεκτή, που συνήθως λαμβάνει από μια γραμμή όρασης, τότε η μέση τιμή τις τυχαίας διαδικασίας δεν θα είναι πλέον μηδέν μεταβάλλοντας γύρω από τα επίπεδα τη δύναμη της διαρπάζουσας διαδρομής. Μια τέτοια κατάσταση μπορεί να μοντελοποιηθεί ως Rician διάλεια.

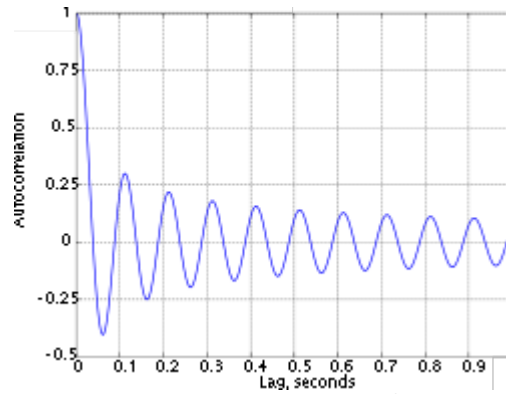
Το πόσο γρήγορα σβήνει το κανάλι επηρεάζεται από το πόσο γρήγορα συνδέεται ο δεκτής με τον πομπό. Κίνηση προκαλεί η μετατόπιση Doppler στις λαμβανόμενες συνιστώσες του σήματος. Τα στοιχεία διώχνουν τη διακύμανση του ρεύματος πάνω από ένα δευτερόλεπτο από την σταθερή διέλευση του σήματος μέσω μιας ενιαίας διαδρομής απόσβεσης Rayleigh καναλιών με μέγιστη μετατόπιση Doppler των 10Hz και 100Hz. Αυτές οι μετατοπίσεις αντιστοιχούν σε 6 χιλ/ώρα (4mph) και 60 χιλ/ώρα (40mph) αντίστοιχα στα 1800MHz. Αυτό είναι το κλασικό σχήμα της διάλειψης Rayleigh. Σημειώνουμε κυρίως τη βαθιά εξασθένιση όπου η ισχύς του σήματος μπορεί να μειωθεί κατά 30-40dB.



ένα δευτερόλεπτο απόσβεσης Rayleigh με μέγιστη μετατόπιση Doppler των 10Hz και 100Hz αντίστοιχα.

Δεδομένου ότι βασίζεται σε μια καλά μελετημένη κατανομή με ειδικές ιδιότητες, η κατανομή Rayleigh προσφέρεται για ανάλυση στα βασικά χαρακτηριστικά που επηρεάζουν την απόδοση του ασύρματου δικτύου. Οι παράμετροι που συζητούνται εδώ είναι για μη στατικό κανάλι. Εάν ένα κανάλι δεν αλλάζει με τον χρόνο, δεν έχει διαλείψεις και παραμένει σε κάποιο συγκεκριμένο επίπεδο. Οι ξεχωριστές ιδιότητες του καναλιού στην περίπτωση αυτή θα είναι ασυσχέτιστες ή μια με την άλλη. Μόλις εισαχθεί σχετική κίνηση μεταξύ πομπού και δεκτή τότε η διάλειψη μεταβάλλεται ως προς το χρόνο.

Η κανονική συνάρτηση αυτο-συσχέτισης ενός καναλιού Rayleigh με κίνηση με σταθερή ταχύτητα είναι μηδενικής τάξεως, πρώτου βαθμού μιας συνάρτησης Bessel: $R(\tau) = J_0(2\pi f_d \tau)$ σε καθυστέρηση T όταν η μέγιστη μετατόπιση είναι $f_d T$. Η συνάρτηση αυτο-συσχέτισης του καναλιού Rayleigh διάλειψης δείχνεται παραπάνω από 10Hz μέγιστη μετατόπιση Doppler όπως δείχνεται στο σχήμα:



Η συνάρτηση αυτό-συσχέτισης του 10Hz Doppler απόσβεσης Rayleigh κανάλι.

Το Level crossing rate είναι ένα είδος μέτρησης της ταχύτητας διάλειψης. Το πόσο συχνά η διάλειαση επηρεάζει κάποιο όρο για Rayleigh διάλειαση είναι: $LCR = \sqrt{2\pi} f_d \rho e^{-\rho^2}$ όπου f_d είναι η μέγιστη μετατόπιση Doppler και ρ είναι το

κατώτερο όριο στη μέση τετραγωνική ρίζα (RMS):
$$\rho = \frac{R_{\text{thresh}}}{R_{\text{rms}}}$$

Η μέση διάρκεια fade ποσοτικοποιεί το πόσο καιρό το σήμα περνά κάτω από

το όριο P . Για απόσβεση Rayleigh η διάρκεια fade είναι:
$$AFD = \frac{e^{\rho^2} - 1}{\rho f_d \sqrt{2\pi}}$$

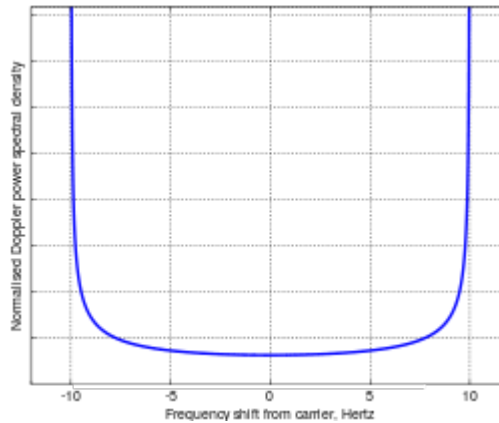
Για μια συγκεκριμένη τιμή το κατώτερο όριο P είναι μέσης διαρκείας εξασθένισης και το ποσοστό ισόπεδης διάβασης δίνετε από: $AFD \times LCR = 1 - e^{-\rho^2}$.

Η Doppler φασματική πυκνότητα ισχύος ενός καναλιού με διαλείψεις, περιγράφει πόσο φασματική διερεύνηση προκαλεί. Αυτό δείχνει πως μια καθαρή συχνότητα π.χ. μια καθαρή ημιτονοειδής η οποία είναι στο πεδίο της συχνότητας απλωμένη σε όλο το φάσμα της πληροφορίας όταν περνά μέσα από το κανάλι. Είναι ο μετασχηματισμός Fourier της συνάρτησης αυτό-συσχέτισης χρόνου. Για Rayleigh διάλειαση με μια κάθετη κεραία λήψης με ευαισθησία σε όλες τις κατευθύνσεις, αυτό

$$S(\nu) = \frac{1}{\pi f_d \sqrt{1 - \left(\frac{\nu}{f_d}\right)^2}},$$

έχει αποδειχτεί ότι είναι:

όπου ν η μετατόπιση συχνότητας σε σχέση με τη φέρουσα συχνότητα. Αυτό το σχήμα παρουσιάζεται για μέγιστη μετατόπιση Doppler των 10Hz. Το σχήμα κύπελο είναι μια κλασική μορφή του φάσματος Doppler.



Το κανονικοποιημένο Doppler φάσματος ισχύος τις απόσβεσης Rayleigh με μέγιστη μετατόπιση Doppler 10Hz.

Όπως περιγράφεται παραπάνω, ένα κανάλι μπορεί να μοντελοποιηθεί με την παραγωγή στα πραγματικά και φανταστικά μέρη ενός μιγαδικού αριθμού σύμφωνα με την ανεξάρτητη μεταβλητή Gaussian. Ωστόσο αυτό συμβαίνει μερικές φορές όταν οι διακυμάνσεις πλάτους παρουσιάζουν ενδιαφέρον. Υπάρχουν δυο προσεγγίσεις για αυτό. Σε αμφότερες περιπτώσεις ο στόχος είναι να παράγει ένα σήμα που έχει φάσμα ισχύος Doppler.

Ένας άλλος τρόπος να παράγει ένα σήμα με το απαιτούμενο φάσμα ισχύος Doppler είναι να περάσει ένα λευκό Gaussian θόρυβο σήματος μέσω ενός Gaussian φίλτρου με απόκριση συχνότητας ίση με την τετραγωνική ρίζα του φάσματος Doppler. Αν και απλούστερο από τα παραπάνω μοντέλα και το μη ντετερμινιστικό, παρουσιάζει ορισμένα ζητήματα εφαρμογής που σχετίζονται με την ανάγκη φίλτρων υψηλής τάξης για να προσεγγίσουμε την τετραγωνική ρίζα στην απόκριση και την δειγματοληψία της Gaussian κυματομορφής με κατάλληλο ρυθμό.

Rician διάλειαση

Η Rician διάλειαση είναι ένα στοχαστικό μοντέλο για το ραδιοφωνικό πολλαπλασιασμό. Ανωμαλία προκαλείται από κατάργηση ενός ραδιοφωνικού σήματος. Από μόνο του το σήμα καταλήγει στον δεκτή από πολλές διαδρομές και επιμηκύνεται ο χρόνος ή συντομεύει. Η Rician διάλειαση συμβαίνει όταν μια από τις διαδρομές, τυπικά μια γραμμή συστήματος όψεως είναι πιο ισχυρή από τις άλλες. Σε Rician διάλειαση το κέρδος πλάτους χαρακτηρίζεται από μια κατανομή.

Rician διάλειαση είναι ένα εξειδικευμένο μοντέλο για στοχαστικές διαλείψεις όταν δεν υπάρχει κανένα πρόβλημα σε καμία γραμμή του σήματος. Μερικές φορές θεωρείται ως μια ειδική περίπτωση της γενικότερης έννοιας Rician διάλειαση.

Μια Rician διάλειαση σε ένα κανάλι μπορεί να περιγραφεί από δυο παραμέτρους K και Ω . K είναι η αναλογία μεταξύ της ισχύος στην άμεση πορεία και τη δύναμη στις άλλες διάσπαρτες διαδρομές. Ω είναι η συνολική ισχύς από τις δυο διαδρομές ($\Omega = \nu^2 + 2\sigma^2$) και δρα ως παράγοντας κλιμάκωσης με την διανομή. Το πλάτος του ληφθέντος σήματος είναι R

τότε:

$$\nu^2 = \frac{K}{1+K} \Omega \sigma^2 = \frac{\Omega}{2(1+K)}$$

$$f(x) = \frac{2(K+1)x}{\Omega} \exp\left(-K - \frac{(K+1)x^2}{\Omega}\right) I_0\left(2\sqrt{\frac{K(K+1)}{\Omega}}x\right),$$

και

Όπου το $I_0(\cdot)$ είναι σειρά τροποποιημένης συναρτήσεων Bessel πρώτου είδους.

Weidull διάλεια

Weidull διάλεια είναι ένα απλό στατικό μοντέλο διάλεια που χρησιμοποιείται στις τηλεπικοινωνίες με βάση την κατανομή Weidull . Εμπορικές μελέτες έχουν δείξει ότι είναι ένα αποτελεσματικό μοντέλο τόσο σε εσωτερικούς όσο και σε εξωτερικούς χώρους.

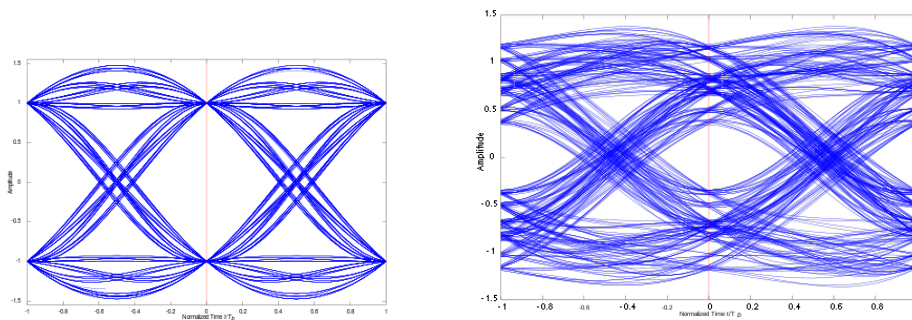
Διασυμβολική παρεμβολή (ISI)

Στις τηλεπικοινωνίες η διασυμβολική παρεμβολή (ISI) είναι μια μορφή στρέβλωσης του σήματος στο οποίο ένα σύμβολο παρεμβαίνει με μεταγενέστερα σύμβολα. Αυτό είναι ένα ανεπιθύμητο φαινόμενο. Είναι σύμβολα που προέρχονται από αντανάκλαση του πρότυπου σήματος και καταλήγουν στον δέκτη με ίδια συχνότητα και πλάτος αλλά πιο αργή ταχύτητα και επιδρούν όπως ο θόρυβος, καθιστώντας αναξιόπιστο το σήμα. Η ISI συνήθως προκαλείται από τη διάδοση πολλαπλών διαδρομών ή την μη γραμμική απόκριση της συχνότητας του καναλιού που προκαλεί πολλά διαδοχικά σύμβολα μαζί “ θολούρα ”. Η παρουσία του ISI στο σύστημα εισάγει σφάλματα διατάξεως στην έξοδο του δεκτή. Ως εκ τούτου στο σχεδιασμό φίλτρων εκπομπής και λήψεως ο στόχος είναι να ελαχιστοποιηθούν οι επιδράσεις του ISI και να παραδίδονται τα ψηφιακά δεδομένα στον προορισμό τους με το μικρότερο δυνατό ποσοστό σφάλματος.

Μια από τις κύριες αιτίες τις διασυμβολικής παρεμβολής είναι γνωστό ως διάδοση πολλαπλών διαδρομών στο οποίο ένα σήμα φτάνει στον δέκτη μέσω διαφορετικών διαδρομών. Οι αιτίες αυτού περιλαμβάνουν ανάκλαση,(να πηδήσει σε κτίρια) διάθλαση και αιμοβόρικές επιδράσεις όπως η ιονοσφαιρική ανάκλαση και η ατμοσφαιρική σωλήνωση. Δεδομένου ότι οι διαφορετικές διαδρομές είναι διαφορετικής απόστασης αυτό έχει σαν αποτέλεσμα τις διαφορετικές εκδόσεις του σήματος που φτάνει στον δέκτη με διαφορετικούς χρόνους. Οι καθυστερήσεις αυτές σημαίνουν ότι μέρος ενός συγκεκριμένου σύμβολου θα εξαπλωθεί στα επόμενα σύμβολα παρεμβάλλοντας έτσι στη σωστή αντίληψη των σύμβολων. Επιπλέον οι διαφορετικές διαδρομές συχνά στρεβλώνουν το πλάτος και τη φάση του σήματος προκαλώντας περεταίρω παρεμβολή στο λαμβανόμενο σήμα.

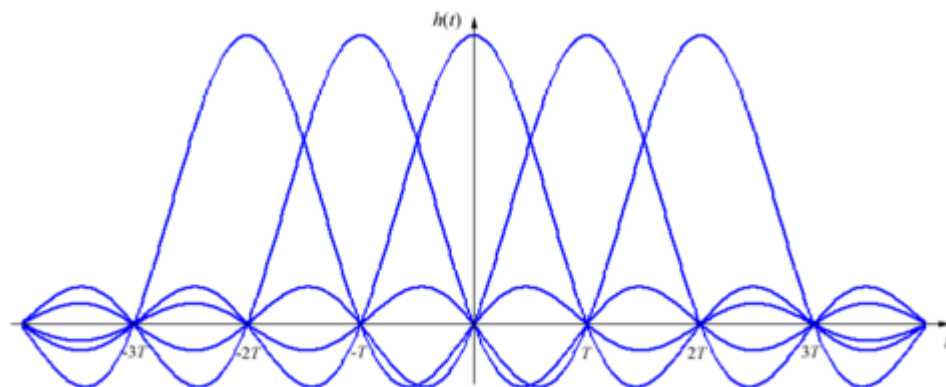
Μια άλλη αιτία της διασυμβολικής παρεμβολής είναι η μετάδοση ενός σήματος μέσω φραγμένων καναλιών σε κάποια ζώνη. Δηλαδή, όπου η απόκριση συχνότητας είναι μηδέν ή πάνω από ένα ορισμένο όριο ξεπερνώντας την συχνότητα αποκοπής. Λαμβάνοντας ένα τέτοιο σήμα μέσω ενός διαύλου έχει σαν αποτέλεσμα την εκπομπή των συχνοτήτων πάνω από την συχνότητα αποκοπής. Επιπλέον το πλάτος των συνιστωσών συχνότητας κάτω από την συχνότητα αποκοπής μπορεί επίσης να εξασθενίσει το κανάλι.

Το ποσοστό θορύβου που απαιτείται για να πάρει ο δεκτής λάθος σήμα δίνεται από την απόσταση μεταξύ του σήματος και του πλάτους στο σημείο μηδέν κατά το χρόνο της δειγματοληψίας. Για να ερμηνευθεί σωστά το σήμα πρέπει να λαμβάνονται δείγματα κάπου ανάμεσα στα δυο σημεία. Οι επιδράσεις της ISI φαίνονται στην δεύτερη εικόνα στην οποία είναι αισθητή η επίδραση από κανάλι πολλών τροχιών. Μειώνετε τόσο το περιθώριο θορύβου και το φάσμα, που μπορεί να πραγματοποιηθεί δειγματοληψία, πράγμα που σημαίνει ότι η αναλογία σφαλμάτων με bit θα είναι μεγαλύτερη.



Υπάρχουν διάφορες τεχνικές για να καταπολεμηθεί το πρόβλημα της διασυμβολικής παρεμβολής ISI. Μερικές από αυτές τις τεχνικές είναι:

- Σύστημα σχεδιασμού έτσι ώστε η κρουστική απόκριση να έχει πολύ λίγη ενέργεια από το επόμενο σύμβολο.



Διαδοχική αποστολή με μηδενική ISI.

- Διαχωρισμός των συμβόλων στο χρόνο με περιόδους αναμονής.
- Εφαρμογή ενός ισοσταθμιστή στο δεκτή.
- Εφαρμογή ανιχνευτή ακολουθίας στον δεκτή.

Υπάρχουν επίσης κωδικοποιημένα συστήματα διαμόρφωσης που δημιουργούν σκόπιμα μια ελεγχόμενη ποσότητα ISI στο σύστημα από την πλευρά του πομπού. Αυτό είναι ευρέως γνωστό ως ταχύτερη Nyquist σηματοδοσία.

Κριτήριο Nyquist ISI

Στις επικοινωνίες, το κριτήριο Nyquist ISI περιγράφει συνθήκες που όταν πληρούνται από ένα κανάλι επικοινωνίας σε οδηγεί σε κάποια διασυμβολική παρεμβολή η ISI. Παρέχει μια μέθοδο για την κατασκευή λειτουργίας που περιορίζει τις επιπτώσεις της διασυμβολικής παρεμβολής. Όταν διαδοχικά σύμβολα μεταδίδονται μέσω ενός διαύλου διαμορφώσεως από ένα γραμμικό λογισμικό διαμορφώσεως όπως ASK, QAM, κ.λπ. ή από παλμική απόκριση, προκαλεί ένα μεταδιδόμενο σύμβολο για να εξαπλωθεί στο πεδίο του χρόνου. Αυτό προκαλεί διασυμβολική παρεμβολή, επειδή τα προηγούμενα σύμβολα που μεταδίδονται επηρεάζουν τη λήψη, εκείνη τη στιγμή στο σύμβολο μειώνοντας έτσι αισθητά την ανοχή στον θόρυβο. Το θεώρημα Nyquist αφορά το πεδίο του χρόνου με προϋπόθεση την ισοδύναμη κατάσταση "συχνότητα-domain". Το κριτήριο Nyquist είναι άμεσα συνδεδεμένο με το θεώρημα δειγματοληψίας του Nyquist Shannon με μια διαφοροποίηση.

Το κριτήριο Nyquist υποδηλώνει την παλμική απόκριση του διαύλου $h(t)$ τότε η προϋπόθεση για ένα ISI χωρίς απόκριση εκφράζεται ως:

$$h(nT_s) = \begin{cases} 1; & n = 0 \\ 0; & n \neq 0 \end{cases}$$

Για όλους τους ακεραίους αριθμούς n , το T_s είναι σύμβολο περιόδου. Το θεώρημα Nyquist λέει ότι αυτό είναι ισοδύναμο με:

$$\frac{1}{T_s} \sum_{k=-\infty}^{+\infty} H\left(f - \frac{k}{T_s}\right) = 1 \quad \forall f$$

Όπου $H(f)$ είναι ο μετασχηματισμός Fourier της $h(t)$. Αυτό είναι το κριτήριο Nyquist ISI.

Το κριτήριο αυτό μπορεί να κατανεμηθεί διαισθητικά με τον ακόλουθο τρόπο: η μετατόπιση της συχνότητας δημιουργεί αντίγραφα του $H(f)$ και μπορεί να προσθέσει μέχρι μια σταθερή τιμή.

Στην πράξη το κριτήριο αυτό εφαρμόζεται με βάση το φιλτράρισμα όσον αφορά την ακολουθία συμβόλων, ως σταθμισμένο ερέθισμα, συνάρτηση δέλτα του Dirac. Όταν το φιλτράρισμα βασικής ζώνης στο σύστημα επικοινωνίας ικανοποιεί το κριτήριο Nyquist τότε τα σύμβολα μπορούν να μεταδίδονται μέσω ενός καναλιού με επίπεδη απόκριση μέσα σε ένα περιορισμένο φάσμα συχνοτήτων χωρίς ISI. Παράδειγμα τέτοιων φίλτρων baseband είναι το φίλτρο sinc ως ιδανική περίπτωση.

Για να επαληθευτεί το κριτήριο πρέπει πρώτα να εκφραστεί το λαμβανόμενο σήμα από την άποψη του μεταδιδόμενου σύμβολου και την απόκριση του διαύλου. Ας υποθέσουμε ότι η λειτουργία $h(t)$ είναι η κρουστική απόκριση στο κανάλι $X[n]$. Τα σύμβολα που πρέπει να αποστέλλονται με περίοδο ενός σύμβολου της T_s . Το λαμβανόμενο σήμα $Y(t)$ θα είναι υπό μορφή (όπου ο θόρυβος έχει αγνοηθεί για λόγους απλότητας) :

$$y(t) = \sum_{n=-\infty}^{\infty} x[n] \cdot h(t - nT_s)$$

Τη δειγματοληψία σε αυτό το σήμα σε διάστημα T_s μπορούμε να την εκφράσουμε $Y(t)$ ως εξίσωση διακριτού χρόνου:

$$y[k] = y(kT_s) = \sum_{n=-\infty}^{\infty} x[n] \cdot h[k - n]$$

Αν γράφουμε το $h[0]$ ως όρο του αθροίσματος ξεχωριστά μπορούμε να το εκφράσουμε ως εξής :

$$y[k] = x[k] \cdot h[0] + \sum_{n \neq k} x[n] \cdot h[k - n]$$

Και αυτό το συμπεραίνουμε όταν το αποτέλεσμα του $h[n]$ ικανοποιεί:

$$h[n] = \begin{cases} 1; & n = 0 \\ 0; & n \neq 0 \end{cases}$$

Μόνο ένα μεταδιδόμενο σύμβολο έχει επίδραση στη ληφθέντα $Y[k]$ σε δειγματοληπτική διαδικασία αφαιρώντας έτσι κάθε ISI. Αυτό σημαίνει στο πεδίο του χρόνου ότι υπάρχει κανάλι χωρίς ISI. Και προκύπτει μια συχνότητα-domain και εκφράζουμε αυτή την κατάσταση σε συνεχή χρόνο για όλους τους ακεραίους n :

$$h(nT_s) = \begin{cases} 1; & n = 0 \\ 0; & n \neq 0 \end{cases}$$

Πολλαπλασιάζουμε μια $h(t)$ με ένα ποσό της συνάρτησης δέλτα $\delta(t)$ που διαχωρίζεται από το διάστημα T_s . Αυτό είναι ισοδύναμο της δειγματοληπτικής

απόκρισης όπως παραπάνω αλλά με χρήση συνεχούς χρόνου. Η μια πλευρά της κατάστασης μπορεί στη συνέχεια να εκφράζεται ως ώθηση στην αρχική συνάρτηση:

$$h(t) \cdot \sum_{k=-\infty}^{+\infty} \delta(t - kT_s) = \delta(t)$$

Μετασχηματισμό Fourier σε τόσα μέλη όσα έχουμε στη σχέση:

$$H(f) * \frac{1}{T_s} \sum_{k=-\infty}^{+\infty} \delta\left(f - \frac{k}{T_s}\right) = 1$$

και

$$\frac{1}{T_s} \sum_{k=-\infty}^{+\infty} H\left(f - \frac{k}{T_s}\right) = 1$$

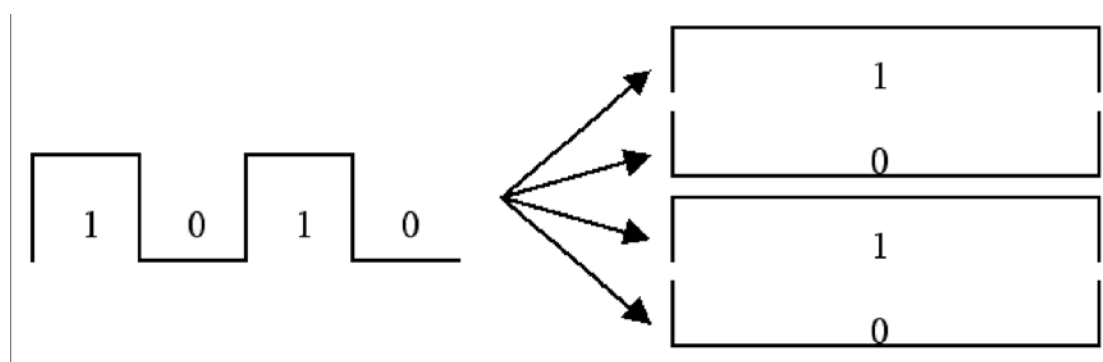
Αυτό είναι το κριτήριο Nyquist ISI και έστω ένας δίαυλος επικοινωνίας να το ικανοποιεί, τότε δεν υπάρχει ISI μεταξύ των διαφορετικών δειγμάτων.

Σχεδιασμός και προσομοίωση (OFDM)

Ένα πρόγραμμα που έχει γραφτεί σε MATLAB για να διερευνήσει ορθογωνιακή πολυπλεξία διαίρεσης συχνότητας (OFDM) συστημάτων επικοινωνίας μπορεί να αναλύσει συστήματα που είναι πολύ πολύπλοκα θεωρητικά, να κάνει τετραγωνική διαμόρφωση πλάτους (QAM) σήματος πολλαπλού φορέα (OFDM) και να συγκρίνει για να αποδείξει την ισχύ στα κανάλια πολλαπλών διαδρομών .

Ένα κοινό πρόβλημα που βρίσκεται σε επικοινωνίες υψηλής ταχύτητας είναι οι παρεμβολές μεταξύ των συμβόλων (ISI). ISI συμβαίνει όταν μια μετάδοση παρεμβάλεται με τον εαυτό της και ο δεκτής δεν μπορεί να αποκωδικοποιήσει την μετάδοση σωστά. Για παράδειγμα, όταν στέλνουμε ένα σήμα από έναν πομπό σε ένα δέκτη και στο περιβάλλον που αποστέλλουμε το σήμα, παρεμβάλλονται κτίρια, λόφοι και άλλες ανακλαστικές επιφάνειες. Τότε στον δεκτή φτάνουν περισσότερα από ένα αντίγραφα του σήματος από πολλές κατευθύνσεις με ίδια πληροφορία, αλλά διαφορετική ταχύτητα, λόγω της ανάκλασης. Τότε ο δεκτής δεν μπορεί να αποκωδικοποιήσει την πληροφορία. Στην ορολογία της επικοινωνίας αυτό ονομάζεται multipath δηλαδή πολλαπλών διαδρομών. Από άμεσες διαδρομές χρειάζεται περισσότερο χρόνο να φτάσει στον δεκτή και τα αντίγραφα καθυστερούν του σήματος προκαλώντας παρεμβολές ISI.

Το OFDM είναι κατάλληλο για επικοινωνία υψηλής ταχύτητας λόγω της μεγάλης αντίστασης που έχει στην ISI. Καθώς τα συστήματα επικοινωνίας έχουν αυξήσει την ταχύτητα μετάδοσης της πληροφορίας, ο χρόνος μετάδοσης αναγκαστικά έχει γίνει μικρότερος και δεδομένου ότι ο χρόνος καθυστέρησης παραμένει σταθερός, το ISI δημιουργεί έναν περιορισμό στην επικοινωνία υψηλού ρυθμού μεταφοράς δεδομένων. Το OFDM αποφεύγει αυτό το πρόβλημα με την αποστολή πολλών μεταδόσεων χαμηλής ταχύτητας ταυτόχρονα. Όταν αποστέλλουμε μια πληροφορία με τέσσερα ψηφία (1010) σε "σειρά" και π.χ. διαρκεί τέσσερα δευτερόλεπτα, στο κάθε ψηφίο αντιστοιχεί ένα δευτερόλεπτο. Με το OFDM θα σταλούν και τα τέσσερα κομμάτια ταυτόχρονα "παράλληλα". Στην περίπτωση αυτή κάθε κομμάτι των δεδομένων έχει διάρκεια τέσσερα δευτερόλεπτα. Αυτή η επιπλέον διάρκεια οδηγεί σε λιγότερα προβλήματα με την ISI. Ένας άλλος λόγος για να χρησιμοποιήσουμε το OFDM είναι η χαμηλή πολυπλοκότητα εφαρμογής τους στα συστήματα υψηλής ταχύτητας σε σύγκριση με τις παραδοσιακές τεχνικές ενιαίου φορέα.

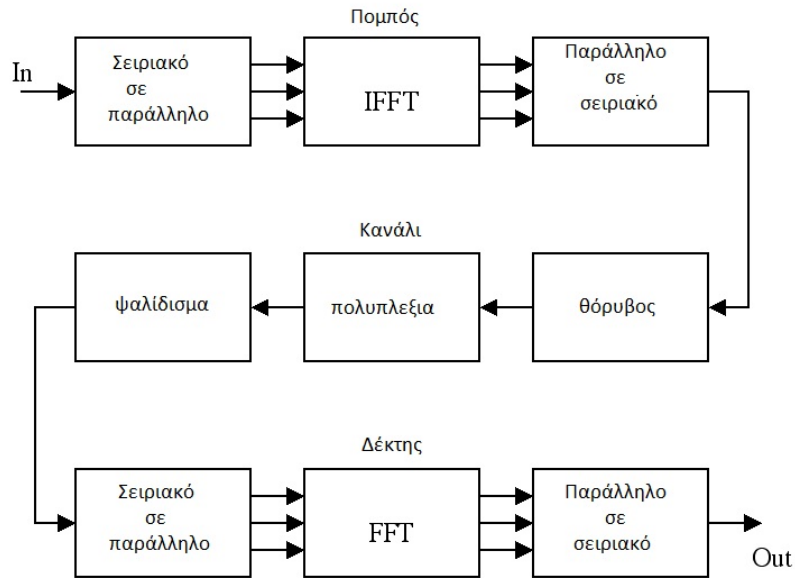


Αριστερά η μετάδοση τις πληροφορίας γίνεται σε σειρά και δεξιά με OFDM παράλληλα.

Με την ταχεία ανάπτυξη ψηφιακής επικοινωνίας κατά τα τελευταία χρόνια η ανάγκη για υψηλή ταχύτητα μετάδοσης δεδομένων έχει αυξηθεί. Νέες τεχνικές διαμόρφωσης πολλαπλού φορέα όπως είναι και το OFDM δημιουργούνται με μεγάλη ζήτηση για ταχύτερη και μεγαλύτερη χωρητικότητα στην επικοινωνία. Σύστημα επικοινωνίας πολλαπλού φορέα για πρώτη φορά σχεδιάστηκε και υλοποιήθηκε το 1960 αλλά δεν ήταν ακόμα ώριμα για εφαρμογή με FFT και κατέρρευσε. Όμως, τα ελκυστικά τους χαρακτηριστικά προκάλεσαν έντονο ενδιαφέρον και πολλοί ερευνητές έσπευσαν να δημιουργήσουν νέα πρότυπα επικοινωνίας. Η επεξεργαστική ισχύς των σύγχρονων επεξεργαστών έχει αυξηθεί σε σημείο όπου το OFDM έχει γίνει εφικτό και οικονομικό.

Το πρόγραμμα αποτελείται από την έρευνα και την προσομοίωση ενός συστήματος επικοινωνίας (OFDM). Για την ανάγκη κατανόησης του προγράμματος MATLAB δημιουργήσαμε ένα διάγραμμα ροής. Ο πομπός μετατρέπει πρώτα τα δεδομένα από ένα συριακό ρεύμα σε παράλληλες σειρές που κάθε σύνολο δεδομένων περιέχει ένα σύμβολο S_i για κάθε υπο-φορέα. Για παράδειγμα ένα σύνολο τεσσάρων σημείων θα είναι (S_0, S_1, S_2, S_3) . Πριν την πραγματοποίηση του Inverse Fast Fourier Transform (IFFT), διατάσσεται στον οριζόντιο άξονα το πεδίο της συχνότητας. Αυτή η συμμετρική διάταξη περί του κατακόρυφου άξονα είναι αναγκαία για την χρήση του IFFT. Ένας αντίστροφος μετασχηματισμός Fourier μετατρέπει τα δεδομένα στο

πεδίο των συχνοτήτων σε δείγματα στο πεδίο του αντιστοίχου χρόνου. Ειδικότερα ο IFFT είναι χρήσιμος για το OFDM διότι παράγει δείγματα μιας κυματομορφής με συνιστώσες συχνότητας που ικανοποιούν τις συνθήκες της ορθογωνίας. Στη συνέχεια η μετατροπή παράλληλο σε σειριακό μπλοκ δημιουργεί το σήμα OFDM διαδοχικά, εξάγοντας δείγματα στο πεδίο του χρόνου.



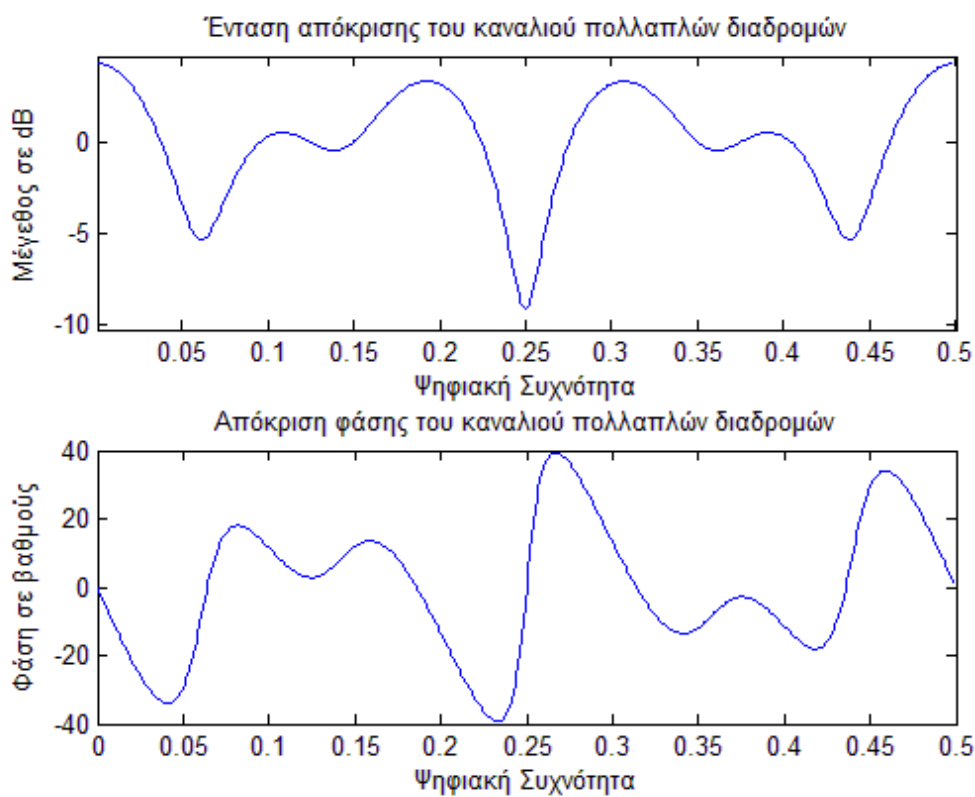
Η προσομοίωση του καναλιού επιτρέπει την εξέταση των κοινών χαρακτηριστικών των ασύρματων καναλιών όπως είναι ο θόρυβος, οι πολλαπλές διαδρομές και το ψαλίδισμα. Με την προσθήκη τυχαίων δεδομένων στην μετάδοση σήματος προσομοιώνεται ο θόρυβος. Η προσομοίωση των πολλαπλών διαδρομών περιλαμβάνει την προσθήκη εξασθενημένων και καθυστερημένων αντιγράφων που μεταφέρουν το ίδιο σήμα με το πρωτότυπο. Αυτό προσομοιώνει το πρόβλημα στην ασύρματη επικοινωνία όταν το σήμα προσομοιώνεται σε πολλά μονοπάτια. Για παράδειγμα ένας δεκτής, μπορεί να δει ένα σήμα μέσω μιας άμεσης διαδρομής, καθώς και μιας διαδρομής που αναπήδα από ένα κτίριο. Και τέλος, το ψαλίδισμα προσομοιώνει το πρόβλημα του κορεσμού ενισχυτή. Αυτό είναι ένα πρόβλημα που επηρεάζει την OFDM όπου η κορυφή σε μέση αναλογία ισχύος είναι υψηλή. Ο δεκτής εκτελεί την αντίστροφη διαδικασία από τον πομπό. Πρώτον, τα δεδομένα από το OFDM χρίζονται από σειριακό ρεύμα σε παράλληλες σειρές. Το fast Fourier transform (FFT) μετατρέπει από το πεδίο του χρόνου πίσω στο πεδίο των συχνοτήτων. Τα μεγέθη της συχνότητας αντιστοιχούν πίσω στα αρχικά δεδομένα. Τέλος, το παράλληλο σε σειριακό μπλοκ μετατρέπει αυτά τα παράλληλα δεδομένα σε ένα σειριακό ρεύμα για να ανακτήσει τα αρχικά δεδομένα εισόδου.

Η προσομοίωση MATLAB δέχεται αρχεία κειμένου, ήχου, δυαδικού συστήματος, ημιτονοειδή και τυχαία δεδομένα. Στη συνέχεια, δημιουργεί τη μετάδοση OFDM και προσομοιώνει ένα κανάλι. Προσπαθεί να ανακτήσει τα

δεδομένα εισόδου και εκτελεί μια ανάλυση για τον προσδιορισμό του ποσοστού σφάλματος της μετάδοσης. Για να συγκρίνετε OFDM σε ένα κλασικό ενιαίο φορέα επικοινωνίας πρέπει να εκτελεστεί μια προσομοίωση της τάξης των 16 QAM. Οι προσομοιώσεις αυτές, είναι δυναμικές επιτρέποντας στον χρήστη να ρυθμίσει τις παραμέτρους για τον καθορισμό των χαρακτηριστικών του συστήματος επικοινωνίας. Δυο απλές γραφικές παραστάσεις αναπτύχθηκαν σε OFDM με διεπαφή (GUI) .

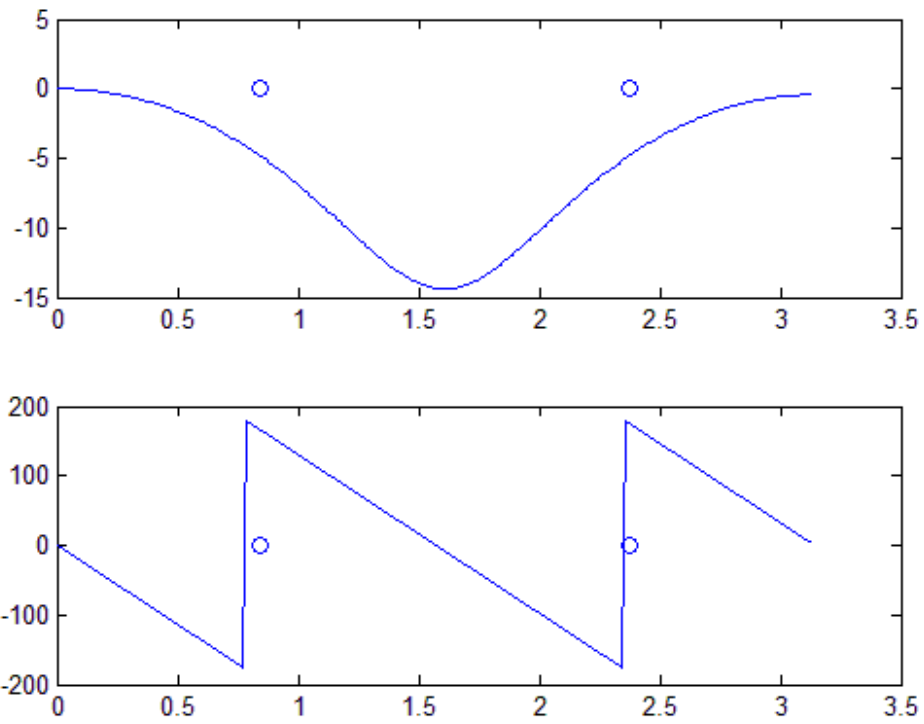
Σε αυτή την γραφική παράσταση απεικονίζεται η προσομοίωση ενός καναλιού πολλαπλών διαδρομών.

Γραφική 1.1

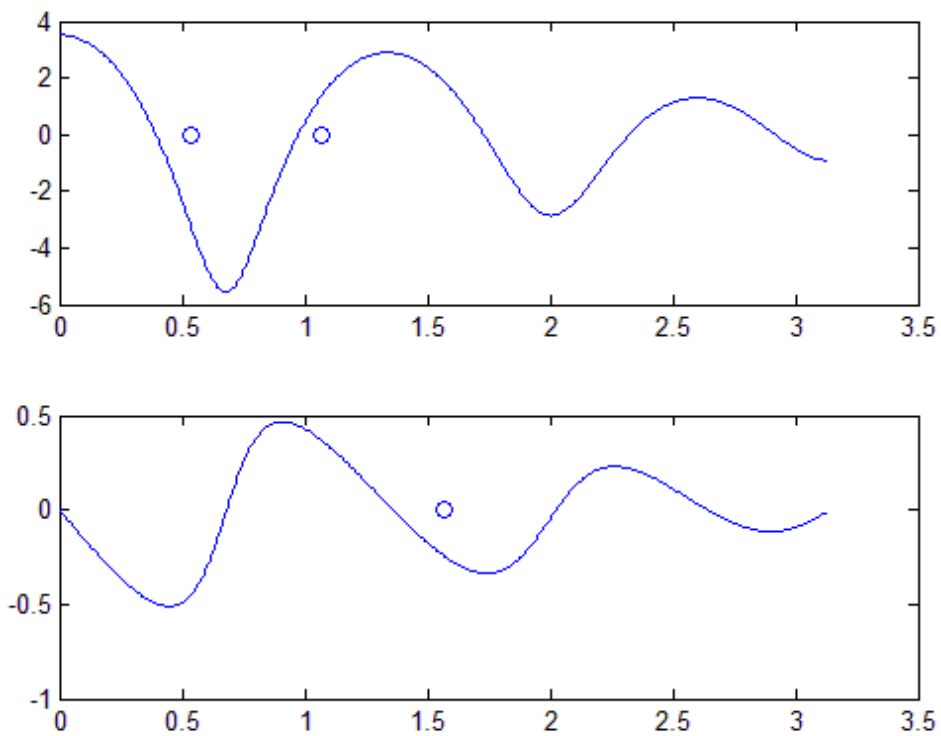


Σε αυτή την γραφική παράσταση βλέπουμε την δημιουργία ενός φίλτρου QAM και τον υπολογισμό απόκρισης συχνότητας. Έχουμε στην επάνω γραφική το mag σε dB και στην δεύτερη τη φάση σε μόρες.

Γραφική 1.2

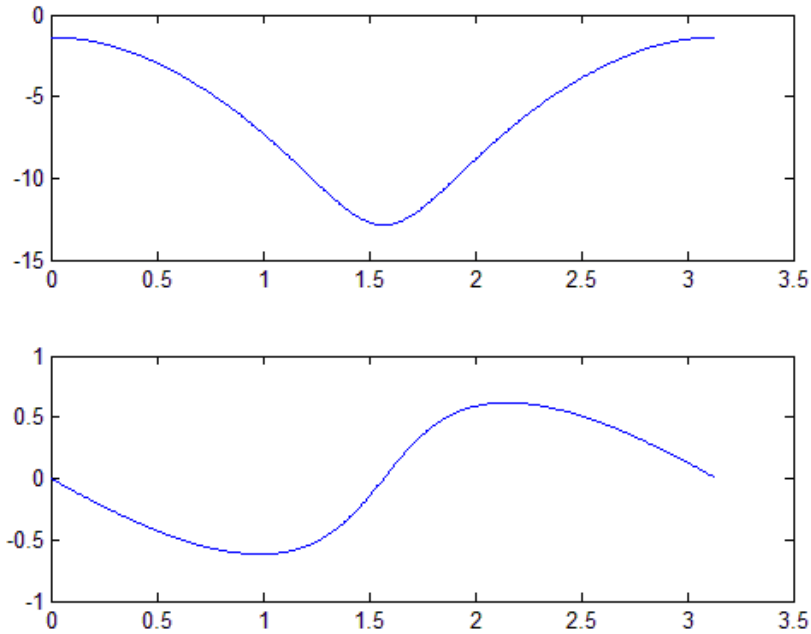


Σχεδιασμός φίλτρου καθορίζοντας την καθυστέρηση κοιτάζοντας mag και τη φάση απόκρισης.



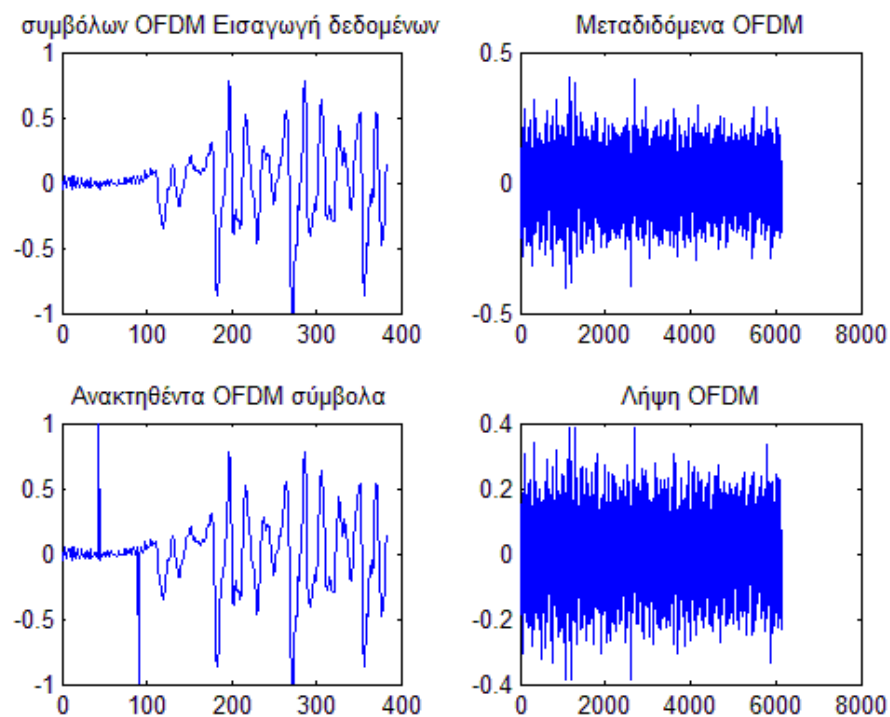
Σχεδιασμός φίλτρου που καθορίζει την απόκριση φάσης σε συγκεκριμένες συχνότητες.

Γραφική 1.3



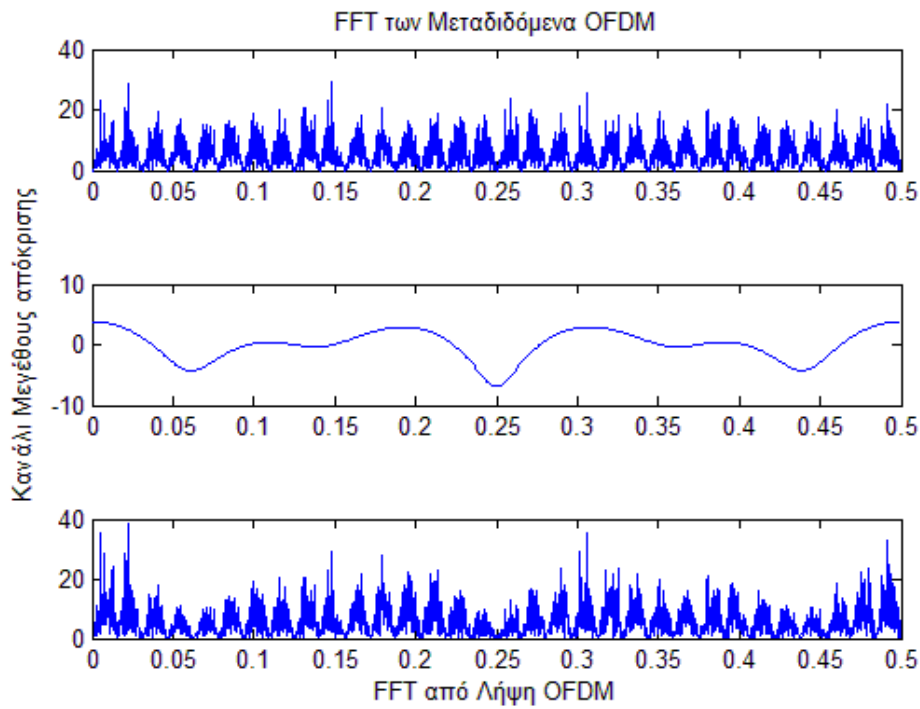
Προετοιμασία ανάλυσης OFDM:

Γραφική 1.4

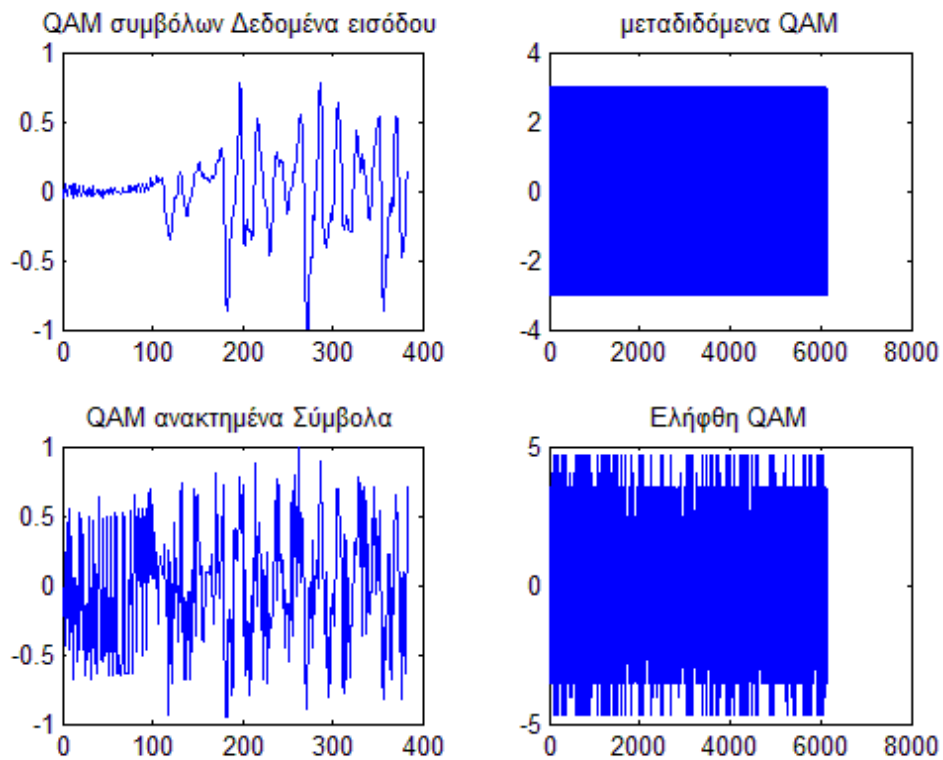


Στην 1.4 γραφική παράσταση απεικονίζεται ένα σήμα εκπομπής πριν την εφαρμογή OFDM, τη διάρκεια της μετάδοσης και την λήψη του σήματος.

Γραφική 1.5

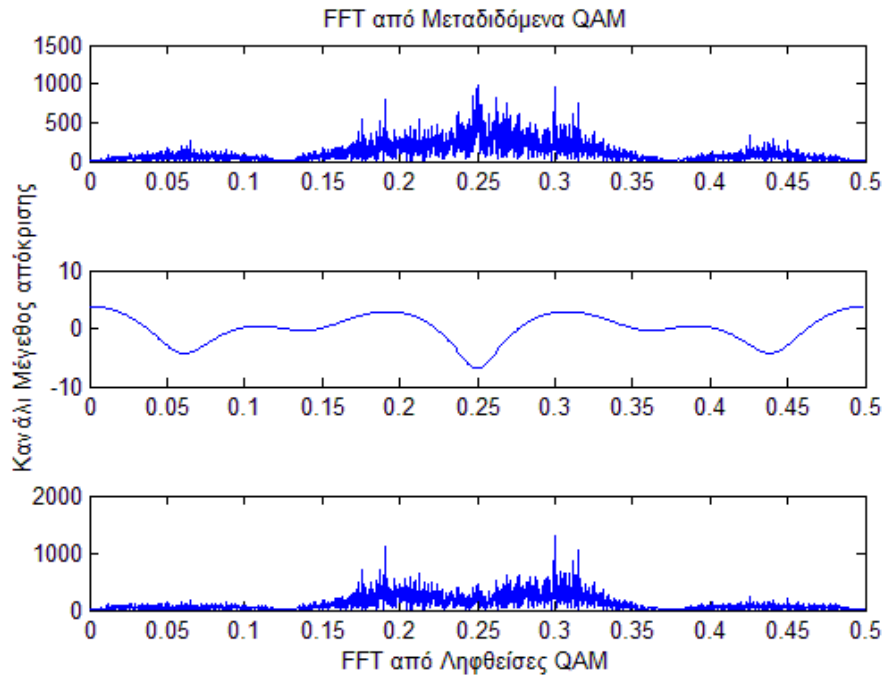


Γραφική 1.6



Στην απεικόνιση 1.5 φαίνεται ο γρήγορος μετασχηματισμός Fourier κατά την εκπομπή του σήματος και κατά τη λήψη, καθώς και το αρχικό σήμα. Στην γραφική 1.6 φαίνεται το σήμα μετά από εφαρμογή QAM κατά την διάρκεια της εκπομπής της μετάδοσης και της λήξης.

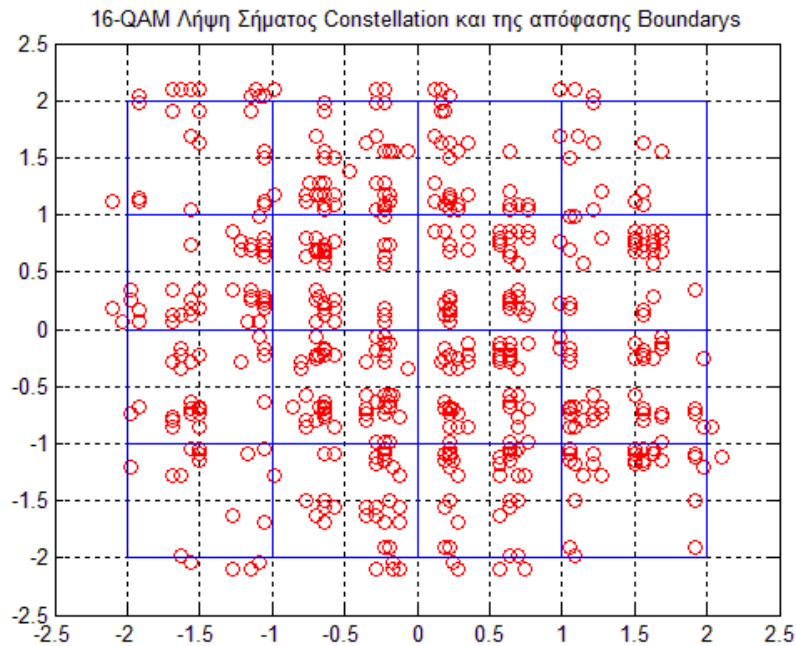
Γραφική 1.7



Στην απεικόνιση 1.7 φαίνεται ο γρήγορος μετασχηματισμός Fourier κατά την εκπομπή του σήματος και κατά τη λήψη αλλά και το αρχικό σήμα.

Βασικό demo OFDM

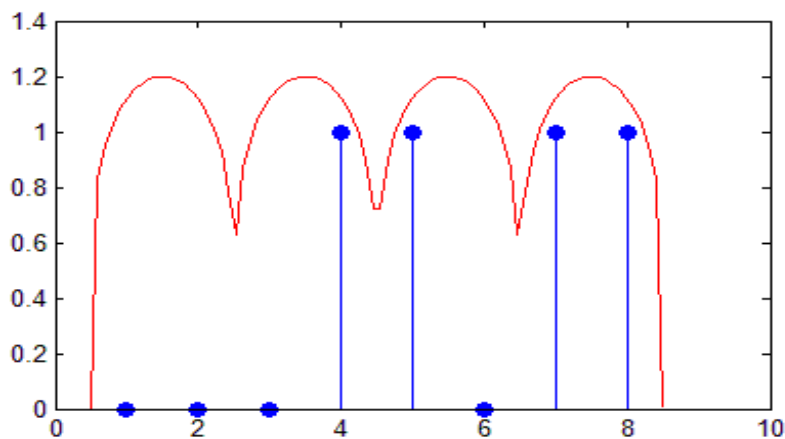
Γραφική 1.8



Εδώ φαίνεται η λήψη του σήματος με δίμορφη 16 QAM , η κατανομή στη λήψη και το φάσμα του σήματος.

Στις παρακάτω γραφικές απεικονίζεται το βασικό demo του OFDM (ορθογώνια πολυπλεξία διαίρεση συχνότητας)

Γραφική 2

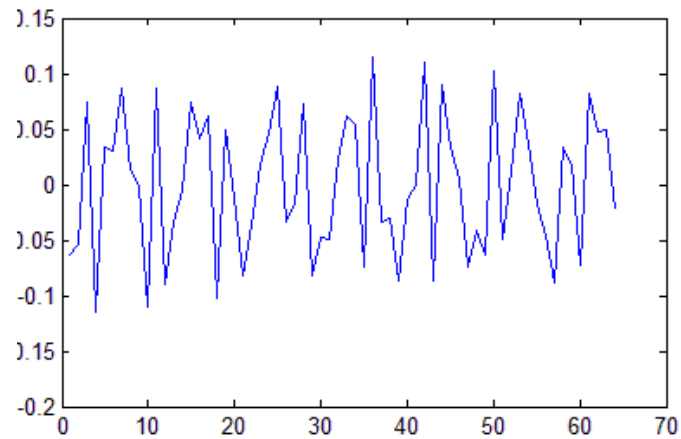


OFDM σε ένα IFFT χρησιμοποιείται για να θέσει τους δυαδικούς αριθμούς σε πολλές συχνότητες. λόγω της μαθηματικά που εμπλέκονται σε ένα IFFT, αυτές οι συχνότητες δεν παρεμβάλλονται εαθη άλλο (σε όρους επικοινωνίας αυτό ονομάζεται «ορθογωνιότητα»). το διάγραμμα δείχνει ότι κάθε ομάδα 2 μπλε σημεία δεδομένων υπό ένα κόκκινο εξόγκωμα θα είναι σε μια συχνότητα.

Next

Close

Γραφική2.1

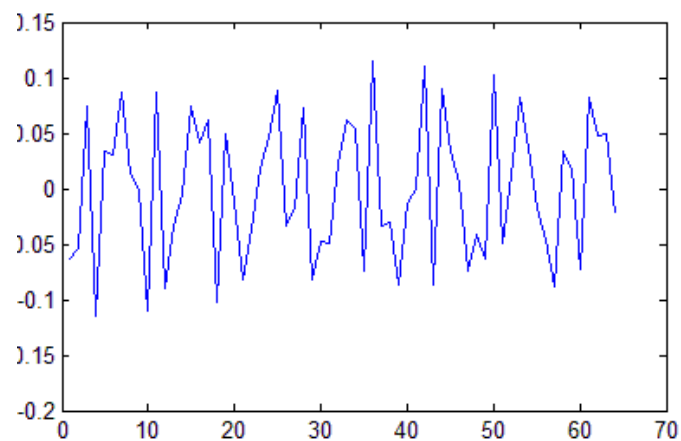


η IFFT μαθηματικά έχει πλέον ολοκληρωθεί. έχει δημιουργήσει ένα σήμα OFDM που αντιστοιχεί στα δυαδικά δεδομένα. το διάγραμμα δείχνει το σήμα που παράγεται από το IFFT

Next

Close

Γραφική2.2

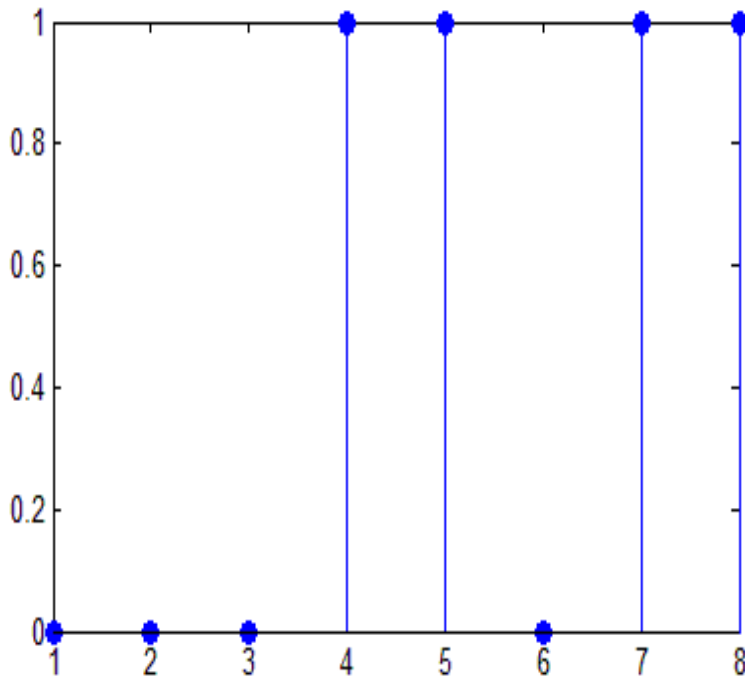


τώρα αυτό το OFDM μπορεί να μεταδοθεί μέσω των μέσων ενημέρωσης και, στη συνέχεια, αυτό το μέσο (ή κανάλι επικοινωνίας) θα μπορούσε να είναι ενσύρματη ή ασύρματη. Είναι μία φορά για να καλύψουν τα αρχικά τα δυαδικά δεδομένα.

Next

Close

Γραφική 2.3



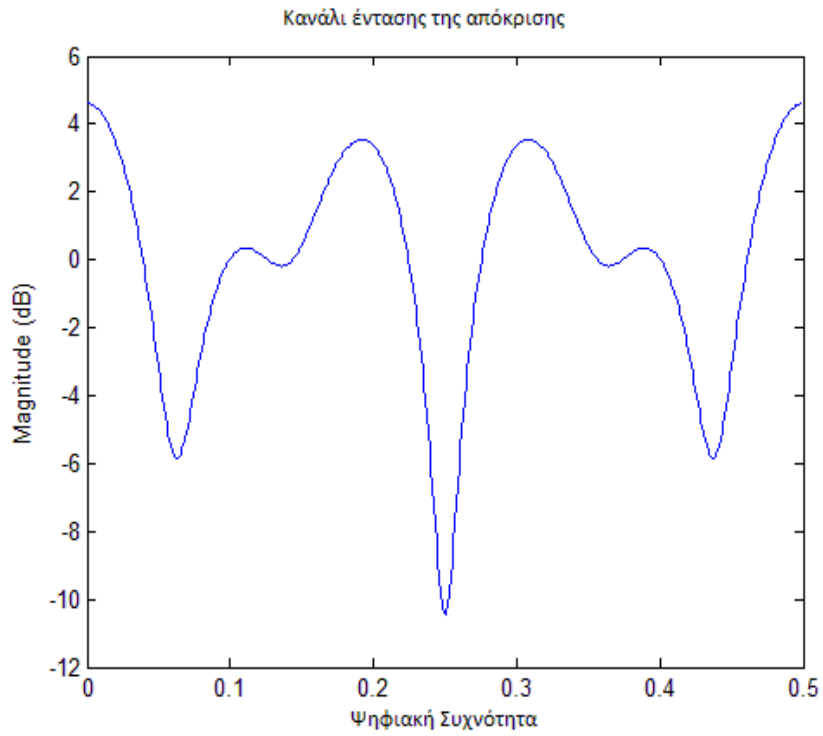
τελικά ένα FFT χρησιμοποιείται για να ανακτήσει τα δυαδικά δεδομένα, όπως φαίνεται στο διάγραμμα. Σημειώστε ότι ο FFT είναι το αντίθετο των IFFT που χρησιμοποιούνται για την παραγωγή του σήματος OFDM. εφ' όσον το κανάλι δεν στρεβλώνει το σήμα OFDM πάρα πολύ, τα αρχικά δυαδικά δεδομένα μπορούν να ανακτηθούν.

Start Over

Close

Η δεύτερη επίδειξη "soundgui_win" δείχνει ένα πιο τεχνικό παράδειγμα. Συγκρίνει OFDM έως 16-QAM σε ένα κανάλι πολλών τροχιών. Ο χρήστης μπορεί να επιλέξει μικρό ή μεγάλο ύψος πόλη-διαδρομών. Οι παρακάτω εικόνες δείχνουν την αλληλουχία του demo με επεξήγηση στο πλαίσιο. Σε αυτή την αλληλουχία εικόνων απεικονίζεται το μεγάλο ύψος πόλη-διαδρομών.

Δεύτερη επίδειξη OFDM (large)



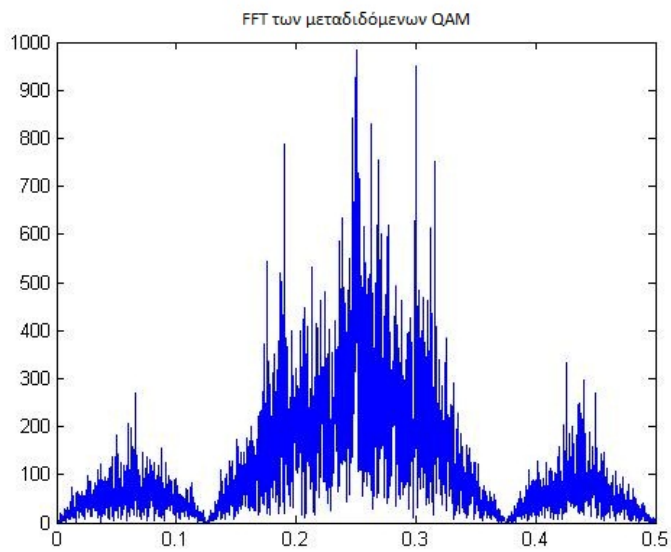
Και OFDM χρησιμοποιώντας ένα αρχείο ήχου ως είσοδο για να επιδειχθούν τα πλεονεκτήματα της χρήσης OFDM με ένα κανάλι πολλών τροχιών.
Επιλέξτε τη δύναμη των πολλαπλών διαδρομών που υπάρχουν στο κανάλι και το διάγραμμα στο θα δείξει την τρέχουσα απόκριση συχνότητας καναλιών

Next

Multipath Channel

Large

Close



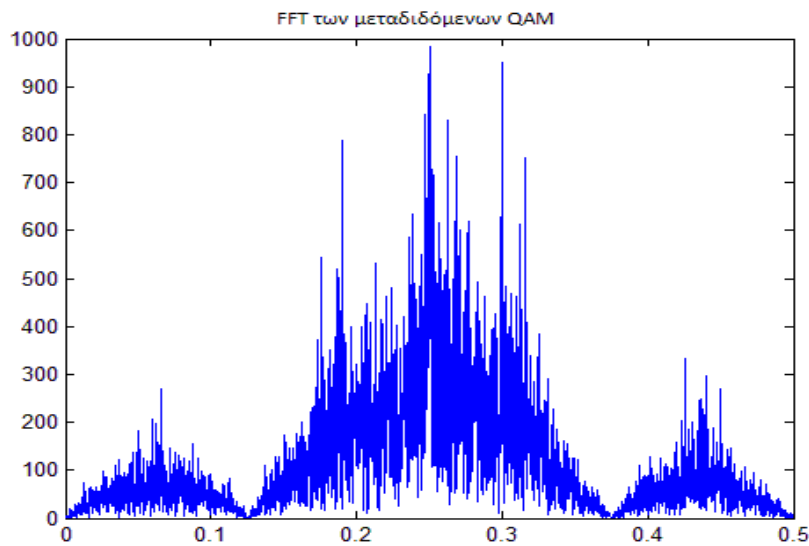
Εδώ είναι ένα πεδίο των συχνοτήτων (FD) αναπαράσταση των δεδομένων QAM να μεταδοθεί. Πιέστε το next πλήκτρο για να συνεχίσετε

Next

Multipath Channel

Large

Close



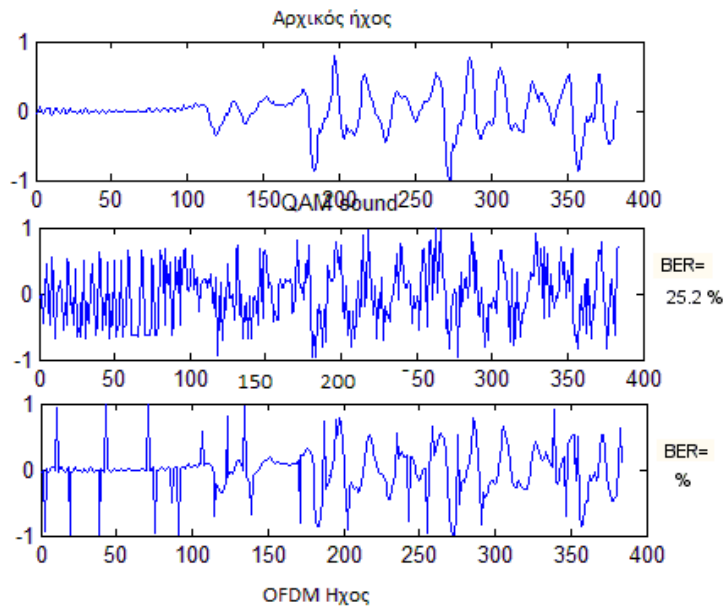
Εδώ είναι ένα πεδίο των συχνοτήτων (FD) αναπαράσταση των δεδομένων OFDM να μεταδοθεί. Πιέστε το next πλήκτρο για να συνεχίσετε

Next

Multipath Channel

Large

Close



Εδώ είναι τα τελικά οικόπεδα των ανακτηθέντων αρχείων ήχου, μαζί με το ποσοστό σφάλματος δυαδικού ψηφίου (BER) για OFDM και QAM. κλικ σε οποιοδήποτε από τα 3 κουμπιά για να ακούσετε αυτά ήχος. δεδομένου ότι OFDM χειρίζεται multipath καλύτερη, ο ήχος είναι λιγότερο παραμορφωμένο. Ο μακρύς ήχος αποδεικνύει περισσότερα παραδείγματα που έχουν ήδη υποστεί επεξεργασία offline

Start O..

Generated Sounds

Original

QAM

OFDM

Longer Sounds

Original

QAM

OFDM

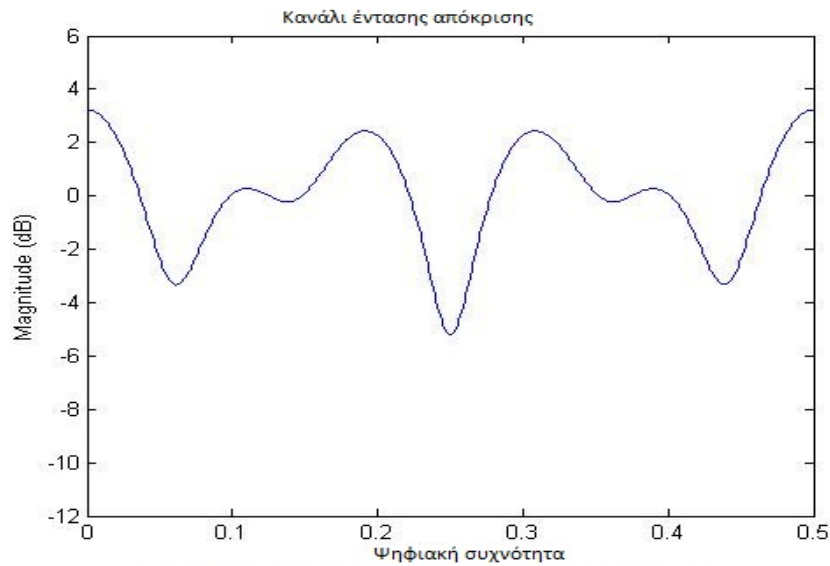
Close

Οι δύο GUI χρησιμοποιούν τον ίδιο κώδικα προσομοίωσης αλλά δεν περιέχει όλες τις δυνατότητες. Επιτρέπεται επίσης η λεπτομερή ανάλυση του συστήματος επικοινωνίας. Τα οικόπεδα δείχνουν εισόδους αλλά και εξόδους OFDM, όπως και εισόδους, εξόδους 16-QAM.

Δεύτερη επίδειξη OFDM (large)

Δεύτερη επίδειξη OFDM (small)

Σε αυτή την αλληλουχία εικόνων απεικονίζεται το μικρό ύψος πόλη-διαδρομών.



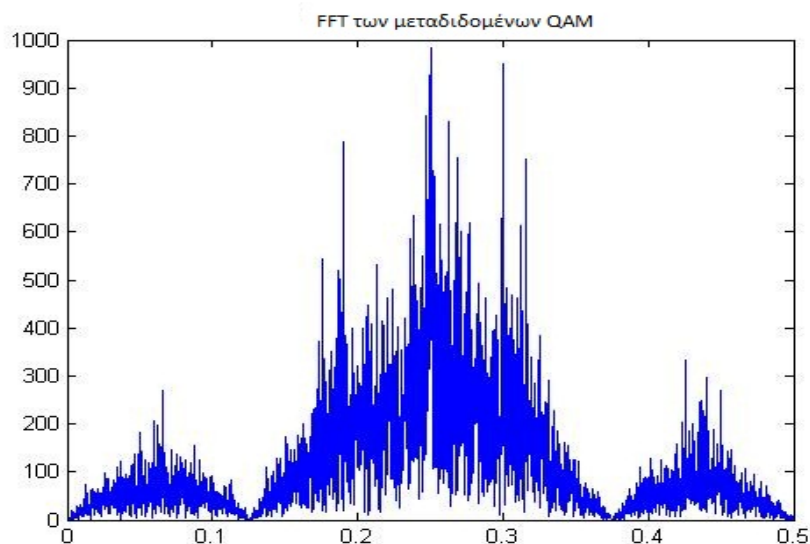
Και OFDM χρησιμοποιώντας ένα αρχείο ήχου ως είσοδο για να επιδειχθούν τα πλεονεκτήματα της χρήσης OFDM με ένα κανάλι πολλαπλών τροχιών. Επιλέξτε τη δύναμη των πολλαπλών διαδρομών που υπάρχουν στο κανάλι και το διάγραμμα θα δείξει την τρέχουσα απόκριση συχνότητας καναλιών.

Next

Multipath Channel

Small

Close



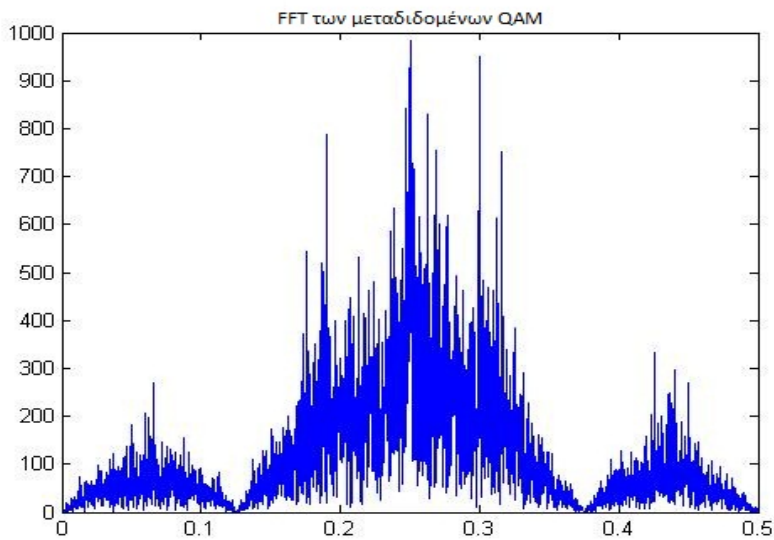
Εδώ είναι ένα πεδίο συχνοτήτων (FD) αναπαράσταση των δεδομένων QAM να μεταδοθεί. Πιέστε το πλήκτρο next για να συνεχίσετε.

Next

Multipath Channel

Small

Close



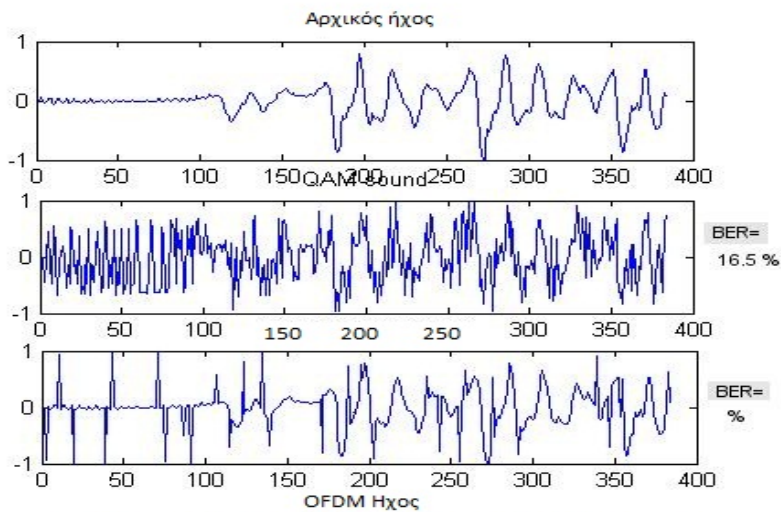
Εδώ είναι ένα πεδίο συχνοτήτων (FD) αναπαράσταση του δεδομένου OFDM να μεταδοθεί. Πιεστέ το πλήκτρο next για να συνεχίσετε.

Next

Multipath Channel

Small

Close



Εδώ είναι τα οικόπεδα των ανακτηθέντων αρχείων ήχου μαζί με το ποσοστό σφάλματος δυαδικού ψόφιου (BER) για OFDM και QAM. Κλικ σε οποιοδήποτε από τα τρία κουμπιά για να ακουστή ο ήχος δεδομένου ότι OFDM χειρίζεται multipath καλύτερη, ο ήχος είναι λιγότερο παραμορφωμένος. Ο μακρύς ήχος αποδεικνύει περισσότερα παραδείγματα που έχουν υπέστη offline

Start O...

Generated Sounds

Original

QAM

OFDM

BER=
16.5 %

Longer Sounds

Original

QAM

OFDM

BER=
%

Close

Πρόγραμμα εξομοίωσης σε MATLAB

% Φίλτρο Σχεδιασμός καθορίζοντας καθυστέρηση σε μονάδες και

% κοιτάζοντας mag και τη φάση απόκριση

% Good default values for fft_size = 128 and num_carriers = 32

```
delay_1 = 6;           %    6
attenuation_1 = 0.35;  %    0.35
delay_2 = 10;         %    10
attenuation_2 = 0.30; %    0.30

num = [1, zeros(1, delay_1-1), attenuation_1, zeros(1, delay_2-delay_1-1),
attenuation_2];
[H, W] = freqz(num, 1, 512); % υπολογιστεί η απόκριση συχνότητας
mag = 20*log10(abs(H)); % μέγεθος σε dB
phase = angle(H) * 180/pi; % γωνία φάσης σε μοίρες
figure(9), clf
subplot(211), plot(W/(2*pi),mag)
title('Ένταση απόκρισης του καναλιού πολλαπλών διαδρομών')
xlabel('ψηφιακή συχνότητα'), ylabel('Μεγέθος σε dB')
subplot(212), plot(W/(2*pi),phase)
title('Απόκριση φάσης του καναλιού πολλαπλών διαδρομών')
xlabel('ψηφιακή συχνότητα'), ylabel('Φάση σε βαθμούς')
```

break

% Design filter using MATLAB command 'fir2'

```
nn = 40; % order of filter
f = [0, 0.212, 0.253, 0.293, 0.5];
m = [1, 1, 0.5, 1, 1];
num = fir2(nn, 2*f, m);
den = 1;
```

```
[H, W] = freqz(num, den, 256); % Υπολογίστε απόκριση συχνοτήτων
mag = 20*log10(abs(H)); % Get mag in dB
phase = angle(H)*180/pi; % Get phase in degrees
```

```
clf
subplot(211), plot(W/(2*pi),mag)
subplot(212), plot(W/(2*pi),phase)
```

break

% Φίλτρο σχεδιασμό χρησιμοποιώντας MATLAB εντολή 'fir1'

```
% Αυτά coeffs λειτουργούν καλά για OFDM εναντίον QAM!!!
% nn = 4; % Για φίλτρου
% w1 = 0.134; % χαμηλής αποκοπής της ζώνης αποκοπής
```

```

% wh = 0.378; % high cutoff of stopband
% nn = 4; % order of filter
% wl = 0.195; % low cutoff of stopband
% wh = 0.309; % high cutoff of stopband

nn = 8; % order of filter
wl = 0.134; % low cutoff of stopband
wh = 0.378; % high cutoff of stopband
num = fir1(nn, 2*[wl, wh], 'stop');
den = 1;

[H, W] = freqz(num, den, 256); % Compute freq response
mag = 20*log10(abs(H)); % Get mag in dB
phase = angle(H)*180/pi; % Get phase in degrees

clf
subplot(211), plot(W,mag), hold on, plot(wl*2*pi,0,'o'), plot(wh*2*pi,0,'o')
subplot(212), plot(W,phase), hold on, plot(wl*2*pi,0,'o'), plot(wh*2*pi,0,'o')
hold off

break
% Σχεδιάστε το φίλτρο καθορίζοντας καθυστέρηση σε μονάδες και κοιτάζοντας mag
και τη φάση απόκριση
n = 512;

d1 =4;
a1 = 0.2;
d2 = 5;
a2 = 0.3;

num = [1, zeros(1, d1-1), a1, zeros(1, d2-d1-1), a2]
den = [1];

[H, W] = freqz(num, den, n);
% F = 0:.1:pi;
% H = freqz(num, den, F*180/pi, 11025);

mag = 20*log10(abs(H));
% phase = angle(H * 180/pi);
phase = angle(H);

clf
subplot(211), plot(W,mag), hold on, plot(0.17*pi,0,'o'), plot(0.34*pi,0,'o')
subplot(212), plot(W,phase), hold on, plot(pi/2,0,'o')
hold off

break
% Φίλτρο Σχεδιασμός καθορίζοντας απάντηση mag σε συγκεκριμένες συχνότητες

n = 2;

```

```

f = [0, 0.25, 0.5];
mag = [1, .05, 1];

[num, den] = yulewalk(n,2*f,mag);

[H, W] = freqz(num, den);

mag = 20*log10(abs(H));
phase = angle(H * 180/pi);

clf
subplot(211), plot(W,mag)
subplot(212), plot(W,phase)

setup

QAM

OFDM

analysis

% Analysis

disp(' '), disp('-----')
disp('προετοιμασία Ανάλυσης')

figure(1), clf
if (input_type == 1) & (test_input_type == 1)
    subplot(221), stem(data_in), title('OFDM Δυαδικά Δεδομένα εισόδου');
    subplot(223), stem(output), title('OFDM ανακτημένα Δυαδικά Δεδομένα');
else
    subplot(221), plot(data_samples), title('OFDM συμβόλου δεδομένων εισόδου');
    subplot(223), plot(output_samples), title('Ανακτήθηκαν OFDM σύμβολα');
end
subplot(222), plot(xmit), title('μεταδιδόμενα OFDM');
subplot(224), plot(recv), title('Λήψη OFDM');

% dig_x_axis = (1:length(QAM_tx_data))/length(QAM_tx_data);
% figure(4), clf, subplot(212)
% freq_data = abs(fft(QAM_rx_data));
% L = length(freq_data)/2;

dig_x_axis = (1:length(xmit))/length(xmit);
figure(2), clf

if channel_on ==1
    num = [1, zeros(1, d1-1), a1, zeros(1, d2-d1-1), a2];
    den = [1];

```

```

[H, W] = freqz(num, den, 512);
mag = 20*log10(abs(H));
phase = angle(H) * 180/pi;

subplot(313)
freq_data = abs(fft(recv));
L = length(freq_data)/2;
plot(dig_x_axis(1:L), freq_data(1:L))
xlabel('FFT των Λήψη OFDM')
axis_temp = axis;

subplot(311),
freq_data = abs(fft(xmit));
plot(dig_x_axis(1:L), freq_data(1:L)), axis(axis_temp)
title('FFT των Μεταδιδόμενα OFDM')

subplot(312)
plot(W/(2*pi),mag),
ylabel('Κανάλι Μεγέθους απόκρισης')
else
subplot(212)
freq_data = abs(fft(recv));
L = length(freq_data)/2;
plot(dig_x_axis(1:L), freq_data(1:L))
xlabel('FFT των Λήψη OFDM')
axis_temp = axis;

subplot(211),
freq_data = abs(fft(xmit));
plot(dig_x_axis(1:L), freq_data(1:L)), axis(axis_temp)
title('FFT των Μεταδιδόμενα OFDM')
end

% if file_input_type == 4
% figure(5)
% subplot(211)
% image(data_in);
% colormap(map);
% subplot(212)
% image(output);
% colormap(map);
% end

if do_QAM == 1 % analyze if QAM was done

figure(3), clf
if (input_type == 1) & (test_input_type == 1)
subplot(221), stem(data_in), title('QAM Δυαδικά Δεδομένα εισόδου');
subplot(223), stem(QAM_data_out), title('QAM ανακτημένα Δυαδικά
Δεδομένα')

```

```

else
    subplot(221), plot(data_samples), title('QAM συμβόλων Δεδομένα εισόδου');
    subplot(223), plot(QAM_output_samples), title('QAM ανακτημένα Σύμβολα');
end
subplot(222), plot(QAM_tx_data), title('μεταδιδόμενα QAM');
subplot(224), plot(QAM_rx_data), title('Ελήφθη QAM');

dig_x_axis = (1:length(QAM_tx_data))/length(QAM_tx_data);
figure(4), clf

if channel_on ==1
    subplot(313)
    freq_data = abs(fft(QAM_rx_data));
    L = length(freq_data)/2;
    plot(dig_x_axis(1:L), freq_data(1:L))
    xlabel('FFT από Ληφθείσες QAM')
    axis_temp = axis;

    subplot(311),
    freq_data = abs(fft(QAM_tx_data));
    plot(dig_x_axis(1:L),freq_data(1:L)), axis(axis_temp)
    title('FFT από Μεταδιδόμενα QAM')

    subplot(312)
    plot(W/(2*pi),mag)
    ylabel('Κανάλι Μέγεθος απόκρισης')
else
    subplot(212)
    freq_data = abs(fft(QAM_rx_data));
    L = length(freq_data)/2;
    plot(dig_x_axis(1:L), freq_data(1:L))
    title('FFT από Ληφθείσες QAM')
    axis_temp = axis;

    subplot(211),
    freq_data = abs(fft(QAM_tx_data));
    plot(dig_x_axis(1:L),freq_data(1:L)), axis(axis_temp)
    title('FFT από Μεταδιδόμενα QAM')
end

% Plots the QAM Received Signal Constellation
figure(5), clf, plot(xxx,yyy,'ro'), grid on, axis([-2.5 2.5 -2.5 2.5]), hold on

% % Overlay plot of transmitted constellation
% x_const = [-1.5 -0.5 0.5 1.5 -1.5 -0.5 0.5 1.5 -1.5 -0.5 0.5 1.5 -1.5 -0.5 0.5 1.5];
% y_const = [-1.5 -1.5 -1.5 -1.5 -0.5 -0.5 -0.5 -0.5 0.5 0.5 0.5 0.5 1.5 1.5 1.5 1.5];
% plot(x_const, y_const, 'b*')

% Overlay of constellation boundarys
x1 = [-2 -2]; x2 = [-1 -1]; x3 = [0 0]; x4 = [1 1]; x5 = [2 2]; x6 = [-2 2];

```



```

y1 = [-2 -2]; y2 = [-1 -1]; y3 = [0 0]; y4 = [1 1]; y5 = [2 2]; y6 = [-2 2];
plot(x1,y6), plot(x2,y6), plot(x3,y6), plot(x4,y6), plot(x5,y6)
plot(x6,y1), plot(x6,y2), plot(x6,y3), plot(x6,y4), plot(x6,y5)

hold off
title('16-QAM Λήψη Σήματος Constellation και της απόφασης Boundaries')

binary_err_bits_QAM = 0;
for i = 1:length(data_in)
    err = abs(data_in(i)-QAM_data_out(i));
    if err > 0
        binary_err_bits_QAM = binary_err_bits_QAM + 1;
    end
end
BER_QAM = 100 * binary_err_bits_QAM/data_length;
end

figure(6), clf
if channel_on == 1
    subplot(211), plot(W/(2*pi),mag),title('Κανάλι Μεγέθους απόκρισης')
    xlabel('ψηφιακή συχνότητα'),ylabel('μέγεθος σε dB')
    subplot(212), plot(W/(2*pi),phase),title('Κανάλι Φάση απόκρισης')
    xlabel('ψηφιακή συχνότητα'),ylabel('Φάση σε βαθμούς')
else
    title('Channel is turned off - No frequency response to plot')
end

% Compare output to input and count errors
binary_err_bits_OFDM = 0;
for i = 1:length(data_in)
    err = abs(data_in(i)-output(i));
    if err > 0
        binary_err_bits_OFDM = binary_err_bits_OFDM + 1;
    end
end
BER_OFDM = 100 * binary_err_bits_OFDM/data_length;
disp(strcat('OFDM: BER=', num2str(BER_OFDM,3), ' %'))
disp(strcat('    Number of error bits=', num2str(binary_err_bits_OFDM)))

if (do_QAM == 1)
    disp(strcat('QAM: BER=', num2str(BER_QAM,3), ' %'))
    disp(strcat('    ', num2str(binary_err_bits_QAM)))
end

% Display text file before and after modulation
if (input_type == 2) & (file_input_type == 2)
    original_text_file = char(data_samples')
    if do_QAM == 1
        edit QAM_text_out.txt
    end
end

```

```

edit OFDM_text_out.txt
end

% Listen to sounds
if (input_type == 2) & (file_input_type == 3)
    do_again = '1';
    while ( ~(isempty(do_again)) )
        disp(' ')
        disp('Πατήστε οποιοδήποτε πλήκτρο για να ακούσετε τον αρχικό ήχο'), pause
        sound(data_samples,11025)
        disp('Πατήστε οποιοδήποτε πλήκτρο για να ακούσετε τον ήχο μετά την
μετάδοση OFDM'), pause
        sound(output_samples,11025)
        if do_QAM == 1
            disp('Πατήστε οποιοδήποτε πλήκτρο για να ακούσετε τον ήχο μετά τη
διαβίβαση QAM'), pause
            sound(QAM_output_samples,11025)
        end
        do_again = ";
        do_again = input('Πληκτρολογήστε "1" για να ακούσετε τους ήχους ξανά ή
πατήστε το κουμπί "Επιστροφή" για να τερματίσετε ', 's');
    end
end
end

```

```

function fig = basicgui_win()
% Αυτό είναι το μηχάνημα που δημιουργείται αναπαράσταση ενός αντικειμένου
γραφικών Λαβή
% και τα παιδιά της. Σημειώστε ότι οι τιμές λαβή μπορεί να αλλάξει όταν αυτά τα
αντικείμενα
% δημιουργούνται ξανά. Αυτό μπορεί να προκαλέσει προβλήματα με τυχόν
επιστροφές κλήσεων σε γραπτή
% εξαρτώνται από την αξία της λαβής κατά τη στιγμή το αντικείμενο έχει
αποθηκευτεί.
% Αυτό το πρόβλημα επιλύεται με την αποθήκευση της εξόδου ως ένα σχήμα-αρχείο.
%
% Για να ανοίξετε ξανά αυτό το αντικείμενο, απλώς πληκτρολογήστε το όνομα του
M-file στο MATLAB
% προτροπή. Το M-αρχείο και τη MAT αρχείο του πρέπει να είναι στο δρόμο σας.
%
% ΣΗΜΕΙΩΣΗ: ορισμένα νεότερα στοιχεία σε MATLAB μπορεί να μην έχουν
αποθηκευτεί σε αυτό
% M-file, λόγω των περιορισμών αυτής της μορφής, η οποία έχει αντικατασταθεί από
% FIG-αρχεία. Τα σχήματα που έχουν σχολιαστεί χρησιμοποιώντας τα εργαλεία
επεξεργασίας οικόπεδο
% είναι ασυμβίβαστες με τη μορφή M-file/MAT-file, και θα πρέπει να αποθηκευτούν
ως
% FIG-αρχεία.

```

```

load basicgui_win

h0 = figure('Color',[0.8 0.8 0.8], ...
'Colormap',mat0, ...
'CreateFcn','OFDMguiFn figure', ...
'FileName','C:\TEMP\alanb\ofdmproj_win\basicgui_win.m', ...
'PaperPosition',[18 180 576.0000000000001 432.0000000000002], ...
'PaperUnits','points', ...
'Position',[273 175 640 429], ...
'Tag','Fig1', ...
'ToolBar','none');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[1 1 1], ...
'FontName','Courier', ...
'HorizontalAlignment','left', ...
'ListboxTop',0, ...
'Position',[6.827586206896553      3.724137931034484      297.3103448275863
68.89655172413795], ...
'String','Basic OFDM Demo. Press Next.', ...
'Style','text', ...
'Tag','StaticTextFeedback');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[0.3 0.3 0.3], ...
'ListboxTop',0, ...
'Position',[309.1034482758622      -0.6206896551724139      88.13793103448278
267.5172413793104], ...
'Style','frame', ...
'Tag','Frame1');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[0.733333 0.733333 0.733333], ...
'Callback','OFDMguiFn next', ...
'FontSize',14, ...
'ListboxTop',0, ...
'Position',[321.5172413793104      234.6206896551725      65.79310344827587
22.96551724137932], ...
'String','Next', ...
'Tag','PushbuttonNext');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[0.733333 0.733333 0.733333], ...
'Callback','OFDMguiFn close', ...
'FontSize',14, ...
'ListboxTop',0, ...
'Position',mat1, ...
'String','Close', ...
'Tag','PushbuttonClose');

```

```

h1 = axes('Parent',h0, ...
    'Units','points', ...
    'Box','on', ...
    'CameraUpVector',[0 1 0], ...
    'Color',[1 1 1], ...
    'ColorOrder',mat2, ...
    'Position',[19.24137931034483      89.3793103448276      261.9310344827587
167.5862068965518], ...
    'Tag','Axes1', ...
    'Visible','off', ...
    'XColor',[0 0 0], ...
    'YColor',[0 0 0], ...
    'ZColor',[0 0 0]);
h2 = line('Parent',h1, ...
    'Color',[0 0 1], ...
    'Tag','Axes1Line1', ...
    'XData',1, ...
    'YData',0);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','center', ...
    'Position',[0.997624703087886 -1.20817843866171 17.32050807568877], ...
    'Tag','Axes1Text4', ...
    'VerticalAlignment','cap', ...
    'Visible','off');
set(get(h2,'Parent'),'XLabel',h2);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','center', ...
    'Position',[-0.1900237529691211 -0.01115241635687725 17.32050807568877], ...
    'Rotation',90, ...
    'Tag','Axes1Text3', ...
    'VerticalAlignment','baseline', ...
    'Visible','off');
set(get(h2,'Parent'),'YLabel',h2);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','right', ...
    'Position',[-0.1520190023752969 1.104089219330855 17.32050807568877], ...
    'Tag','Axes1Text2', ...
    'Visible','off');
set(get(h2,'Parent'),'ZLabel',h2);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','center', ...
    'Position',[0.997624703087886 1.059479553903346 17.32050807568877], ...

```

```

    'Tag','Axes1Text1', ...
    'VerticalAlignment','bottom', ...
    'Visible','off');
set(get(h2,'Parent'),'Title',h2);
if nargin > 0, fig = h0; end

```

% bin2eight

```

%
% Μετατρέπει το δυαδικό δεδομένων σε μια μορφή οκτώ bit
% Δέχεται συστοιχία 1x8 και επιστρέφει το αντίστοιχο δεκαδικό

```

```

y = 0;
k = 0;
for i = 1:8
    y = y + x(8-k)*2^k;
    k = k+1;
end

```

% bin2pol

```

%
% Μετατρέπει το δυαδικό αριθμό (0,1) σε πολικές αριθμούς (-1,1)
% Δέχεται ένα 1-D σειρά από δυαδικούς αριθμούς

```

```

y = ones(1,length(x));
for i = 1:length(x)
    if x(i) == 0
        y(i) = -1;
    end
end

```

% ch

```

recv = xmit; % κανάλι εφαρμόζεται σε recv, δεν τροποποιούν δεδομένα που
διαβιβάζονται

```

```

if channel_on == 1
    disp('Προσομοίωση Καναλιού')

```

```

    norm_factor = max(abs(recv)); % Κανονικοποίηση όλα τα δεδομένα πριν από την
εφαρμογή

```

```

    recv = (1/norm_factor) * recv; % κανάλι για μια δίκαιη σύγκριση

```

```

    ch_clipping

```

```

    ch_multipath

```

```

    ch_noise

```

```
    recv = norm_factor * recv;    % Επαναφορά του μεγέθους των δεδομένων για τη  
    σωστή αποκωδικοποίηση
```

```
end
```

```
% ch_clipping
```

```
for i = 1:length(recv)  
    if recv(i) > clip_level  
        recv(i) = clip_level;  
    end  
    if recv(i) < -clip_level  
        recv(i) = -clip_level;  
    end  
end
```

```
% ch_multipath
```

```
copy1=zeros(size(recv));  
for i=1+d1:length(recv)  
    copy1(i)=a1*recv(i-d1);  
end
```

```
copy2=zeros(size(recv));  
for i=1+d2:length(recv)  
    copy2(i)=a2*recv(i-d2);  
end
```

```
recv=recv+copy1+copy2;
```

```
% ch_noise (operate on recv)
```

```
% τυχαίου θορύβου που ορίζεται από το επίπεδο του θορύβου πλάτους
```

```
if already_made_noise == 0 % παράγουν μόνο μία φορά και να το χρησιμοποιήσετε  
και για τις δύο QAM και OFDM
```

```
    noise = (rand(1,length(recv))-0.5)*2*noise_level;  
    already_made_noise = 1;
```

```
end
```

```
recv = recv + noise;
```

```
% ComputeChannelGUI.m οικόπεδα το τρέχον κανάλι
```

```
popupHnd1=findobj('Tag','PopupMenuMultipath');
```

```
noChannel = 0;
```

```
if get(popupHnd1,'Value') == 3    % Large
```

```
    d1 = 6;
```

```
    a1 = 0.4;
```

```
    d2 = 10;
```

```

    a2 = 0.3;
elseif get(popupHnd1,'Value') == 2 % Small
    d1 = 6;
    a1 = 0.25;
    d2 = 10;
    a2 = 0.20;
else % None
    noChannel = 1;
    channel_on = 0;
    break
end
num = [1, zeros(1, d1-1), a1, zeros(1, d2-d1-1), a2];
den = [1];
[H, W] = freqz(num, den);
mag = 20*log10(abs(H));
phase = angle(H) * 180/pi;
% plot(W/(2*pi),mag) % σχόλιο μου έξωδο κανονικά

% eight2bin
%
% Μετατρέπει οκτώ bit δεδομένα (0-255 δεκαδικά ψηφία) σε δυαδική μορφή για
επεξεργασία.

y = zeros(1,8);
k = 0;
while x > 0
    y(8-k) = rem(x,2);
    k = k+1;
    x = floor(x/2);
end

% Run OFDM simulation

tic % Ξεκινήστε το χρονόμετρο να υπολογίσει πόσο καιρό παίρνει προσομοίωση
QAM
disp(' '),disp('-----')
disp('OFDM Simulation')

tx
ch
rx

% Σταματήστε το χρονόμετρο να υπολογίσει πόσο καιρό παίρνει προσομοίωση QAM
OFDM_simulation_time = toc;
if OFDM_simulation_time > 60
    disp(strcat('Χρόνος για την προσομοίωση OFDM =',
num2str(OFDM_simulation_time/60), ' minutes.'));
else

```

```

disp(strcat('Χρόνος για την προσομοίωση OFDM =',
num2str(OFDM_simulation_time), ' seconds.));
end

```

OFDMguiFn(action)

% Ενοποιεί όλες τις GUI επανακλήσεις σε μία κύρια λειτουργία

```

stringArray = [...
% Slide 1
'Καλώς ήρθατε στο βασικό OFDM (Orthogonal Frequency ...
'Division Multiplexing) demo. Παρακαλούμε κάντε κλικ στο Next ...
'κουμπί για να ξεκινήσετε. ...
'
'
'
';...
% Slide 2
'Ας υποθέσουμε ότι θέλουμε να μεταδώσει την ακόλουθη δυαδική ...
'δεδομένα χρησιμοποιώντας OFDM: [0 0 0 1 0 1 1 1]. ...
'To διάγραμμα δείχνει αυτό το δυαδικά δεδομένα. ...
'
'
'
';...
% Slide 3
'Σε ένα IFFT OFDM (Transform αντίστροφος ταχύς μετασχηματισμός Fourier)
είναι ...
'χρησιμοποιείται για να θέσει τις δυαδικούς αριθμούς σε πολλές συχνότητες. ...
'Λόγω των μαθηματικών που εμπλέκονται σε ένα IFFT, αυτές ...
'συχνότητες δεν παρεμβάλλονται άλλες εαθ (σε ...
'επικοινωνιακούς όρους, αυτό ονομάζεται "Ορθογωνικότητα"). ...
'To διάγραμμα δείχνει ότι κάθε ομάδα 2 μπλε σημεία δεδομένων ...
'κάτω από ένα κόκκινο εξόγκωμα θα τεθεί σε μια συχνότητα. ...
% Slide 4
'Tα μαθηματικά IFFT έχει πλέον ολοκληρωθεί. Αυτό έχει δημιουργήσει μια ...
'OFDM σήματος που αντιστοιχεί στα δυαδικά δεδομένα. ...
'To διάγραμμα δείχνει το σήμα που παράγεται από το IFFT. ...
'
'
'
';...
% Slide 5 - same plot
'Tώρα, αυτό το σήμα OFDM μπορεί να μεταδοθεί μέσω ενός ...
'μέσων και στη συνέχεια έλαβε. Αυτό το μέσο (ή "Channel" σε ...
'επικοινωνίας) θα μπορούσε να είναι ενσύρματη ή ασύρματη. Μόλις η ...
'σήμα που λαμβάνει, η αντίστροφη διαδικασία γίνεται για να ...
'ανακτήσει το αρχικό δυαδικά δεδομένα. ...
'
'
'
';...

```



```

% Slide 6
'Τέλος, ένα FFT (Fast Fourier Transform) χρησιμοποιείται για να '...
'ανακτήσει τα δυαδικά δεδομένα, όπως φαίνεται στο διάγραμμα. σημείωση '...
'ότι ο FFT είναι το αντίθετο του IFFT που χρησιμοποιείται για να '...
'παράγει το σήμα OFDM. Εφ'όσον δεν το Καναλιού '...
'δεν στρεβλώνουν το σήμα OFDM πάρα πολύ, το αρχικό '...
'δυαδικά δεδομένα μπορούν να ανακτηθούν. '...
'
'];

```

```

switch(action)
case 'next' %-----
textHnd1=findobj('Tag','StaticTextFeedback');
nextHnd1=findobj('Tag','PushbuttonNext'); % χειρισμού για το κουμπί Επόμενο
axisHnd1=findobj('Tag','Axes1');
global COUNTER
if isempty(COUNTER)
COUNTER = 0; % προετοιμαστεί COUNTER αν δεν υπάρχει
end
COUNTER = COUNTER + 1;
[r c]=size(stringArray);
if COUNTER > r
COUNTER = 0;
close(gcf)
% basicGUI
basicGUI_win % για την έκδοση των Windows
else
set(textHnd1,'String',stringArray(COUNTER,:))
switch(COUNTER)
case 1
% disp('Slide 1')
case 2
% disp('Slide 2')
setupGUI % δημιουργεί τις μεταβλητές GUI
set(axisHnd1,'Visible','on')
% Stem Οικόπεδο δυαδικών δεδομένων
stem(data_in,'filled')
case 3
% disp('Slide 3')
setupGUI % δημιουργεί τις μεταβλητές GUI
% προσθέστε ομάδες γύρω από το οικόπεδο στελέχους
y=1.2*abs(sin(linspace(0,4*pi,80))).^(1/5);
x=linspace(0.5,8.5,80);
plot(x,y,'r'),hold on
stem(data_in,'filled'),hold off
case 4
% disp('Slide 4')
setupGUI
% Εκτελέστε το IFFT και να εμφανίσετε τα αποτελέσματα
tx

```

```

        plot(xmit)
    case 5
        % disp('Slide 5')
        % same plot
    case 6
        % disp('Slide 6')
        setupGUI
        tx, ch, rx
        stem(output,'filled')

        set(nextHnd1,'String','Start Over') % Επαναλαμβάνω, εάν είναι
επιθυμητό
    otherwise
        disp('error')
    end
end

case 'close' %-----
    clear global COUNTER
    close(gcf)

case 'figure' %-----
    % Αυτό καλείται όταν το ποσοστό αυτό δημιουργείται για πρώτη φορά ή OXI;
    textHnd1=findobj('Tag','StaticTextFeedback');
    axisHnd1=findobj('Tag','Axes1');
    set(textHnd1,'String','Basic OFDM Demo') % προεπιλεγμένα Μηνύματα
κειμένου
    set(axisHnd1,'Visible','off') % απόκρυψη άξονα για να ξεκινήσει
end

```

OFDMguiFnSound(action)

% Ενοποιεί όλες τις GUI επανακλήσεις σε μία κύρια λειτουργία
 % Alan Brooks the man wrote this

```

stringArray = [...
    % Slide 1
    'Καλώς ήρθατε στο demo ήχου OFDM. Αυτό προσομοιώνει QAM '...
    'και OFDM χρησιμοποιώντας ένα αρχείο ήχου ως είσοδο για να αποδείξει '...
    'τα πλεονεκτήματα της χρήσης OFDM με πολυδρόμων '...
    'channel. '...
    'Επιλέξουν τη δύναμη πολλαπλής διαδρομής παρόντες στην '...
    'κανάλι και το οικόπεδο θα δείξει τα τρέχοντα κανάλια '...
    'απόκριση συχνότητας. ';...
    % Slide 2
    'Εδώ είναι ένα πεδίο των συχνοτήτων (FD) εκπροσώπηση της '...
    'QAM δεδομένα που πρέπει να διαβιβάζονται. '...
    'Πατήστε οποιοδήποτε πλήκτρο για να συνεχίσετε. '...
    ' '...
    ' '...
    ' '...

```

```

';...
% Slide 2b
'Για QAM (single-φορέα) μετάδοση, αυτό το οικόπεδο '...
'δείχνει το κανάλι απόκριση συχνότητας (μαύρο) και το '...
'ληφθέντα δεδομένα (γαλάζιο) που επικαλύπτει το αρχικό '...
'δεδομένα (μπλε). Σημειώστε ότι τα λαμβανόμενα δεδομένα είναι ελαφρώς '...
'στρεβλωθεί λόγω της εξασθένησης του καναλιού που προκαλείται από '...
'multipath. '...
'Πατήστε οποιοδήποτε πλήκτρο για να συνεχίσετε. ';...
% Slide 2c
'Εδώ είναι ένα πεδίο των συχνοτήτων (FD) εκπροσώπηση της '...
'OFDM δεδομένα που πρέπει να διαβιβάζονται. '...
'Πατήστε οποιοδήποτε πλήκτρο για να συνεχίσετε. '...
'
'
'
';...
% Slide 2d
'Για OFDM (multi-carrier) μετάδοση, αυτό το οικόπεδο '...
'δείχνει το κανάλι (μαύρο) και λαμβανόμενων δεδομένων (φως '...
'μπλε) επικαλύφθηκαν επί των αρχικών δεδομένων (μπλε). σημείωση '...
'ότι η OFDM ληφθέντων δεδομένων παρουσιάζει επίσης πολλαπλών διαδρομών
'...
'παραμόρφωση. Επίσης, παρατηρούμε ότι το σήμα OFDM είναι '...
'απλώνονται πάνω περισσότερο εύρος ζώνης από QAM από το OFDM '...
'Χρησιμοποιεί πολλές συχνότητες φέροντος. ';...
% Slide 3
'Εδώ είναι οι τελικές οικόπεδα των ανακτηθέντων αρχείων ήχου '...
'μαζί με το Bit ο ρυθμός σφαλμάτων (BER) για OFDM και QAM. '...
'Κάντε κλικ σε οποιοδήποτε από τα 3 κουμπιά για να ακούσετε αυτούς τους ήχους.
'...
'Από OFDM χειρίζεται multipath καλύτερη, ο ήχος είναι '...
'λιγότερο παραμορφωμένη. '...
'Οι Long ήχοι αποδεικνύουν πλέον παραδείγματα που έχουν '...
'ήδη υποβληθεί σε επεξεργασία εκτός σύνδεσης. '];

```

```

switch(action)
case 'next' %-----
textHnd1=findobj('Tag','StaticTextFeedback');
nextHnd1=findobj('Tag','PushbuttonNext'); % χειρισμού για το κουμπί Επόμενο
% axis handlers
axisHnd1=findobj('Tag','Axes1'); % main
axisHnd2=findobj('Tag','AxesOriginal'); % original
axisHnd3=findobj('Tag','AxesQAM'); % QAM
axisHnd4=findobj('Tag','AxesOFDM'); % OFDM
% multipath handlers
textHnd2=findobj('Tag','StaticTextMultipath');
popupHnd1=findobj('Tag','PopupMenuMultipath');

```

```

% Generated Sounds handlers
textHnd3=findobj('Tag','StaticTextGenSounds');
OriginalHnd1=findobj('Tag','PushbuttonOriginal');
QAMHnd1=findobj('Tag','PushbuttonQAM');
OFDMHnd1=findobj('Tag','PushbuttonOFDM');
% Long Sounds handlers
textHnd4=findobj('Tag','StaticTextLongSounds');
OriginalLongHnd1=findobj('Tag','PushbuttonOriginalLong');
QAMLongHnd1=findobj('Tag','PushbuttonQAMLong');
OFDMLongHnd1=findobj('Tag','PushbuttonOFDMLong');
% BER handlers
textHnd5=findobj('Tag','StaticTextBER1'); % label
textHnd6=findobj('Tag','StaticTextBER2'); % label
textHnd7=findobj('Tag','StaticTextBERQAM'); % OFDM BER field
textHnd8=findobj('Tag','StaticTextBEROFDM'); % QAM BER field
global COUNTER
if isempty(COUNTER)
    COUNTER = 0; % προετοιμαστεί COUNTER αν δεν υπάρχει
end
COUNTER = COUNTER + 1;
[r c]=size(stringArray);
if COUNTER > r
    COUNTER = 0;
close(gcf)
% SoundGUI
SoundGUI_win % windows only
else
set(textHnd1,'String',stringArray(COUNTER,:))
switch(COUNTER)
case 1
    % disp('Slide 1')
    % Show/Hide the GUI
    set(nextHnd1,'String','Next')
    % show multipath controls
    set(textHnd2,'Visible','on')
    set(popupHnd1,'Visible','on')
    % enable multipath controls
    set(textHnd2,'Enable','on')
    set(popupHnd1,'Enable','on')
    % show main axis
    set(axisHnd1,'Visible','on'),axes(axisHnd1)
    % hide other axis's
    set(axisHnd2,'Visible','off')
    set(axisHnd3,'Visible','off')
    set(axisHnd4,'Visible','off')
    % hide generated sounds stuff
    set(textHnd3,'Visible','off')
    set(OriginalHnd1,'Visible','off')
    set(QAMHnd1,'Visible','off')
    set(OFDMHnd1,'Visible','off')

```

```

% hide long sounds stuff
set(textHnd4,'Visible','off')
set(OriginalLongHnd1,'Visible','off')
set(QAMLongHnd1,'Visible','off')
set(OFDMLongHnd1,'Visible','off')
% hide the BER displays
set(textHnd5,'Visible','off')
set(textHnd6,'Visible','off')
set(textHnd7,'Visible','off')
set(textHnd8,'Visible','off')
set(popupHnd1,'Value',1) % no channel by default
% default plot
plot(0:.05:.5,zeros(1,11)),axis([0 0.5 -12 6]),title('Κανάλι Μέγεθος
απόκρισης')
xlabel('ψηφιακή συχνότητα'),ylabel('Μέγεθος (dB)')

case {2, 3, 4, 5}
% disp('Slide 2')
% disable ελέγχου πολλαπλών διαδρομών
set(textHnd2,'Enable','off')
set(popupHnd1,'Enable','off')
setupSoundGUI % δημιουργεί τις μεταβλητές ήχου GUI
set(textHnd1,'String','QAM Simulation... Please Wait')
QAM
set(textHnd1,'String',stringArray(COUNTER,:))
fft_temp = abs(fft(QAM_tx_data));
fft_temp = fft_temp(1:floor(0.5*length(fft_temp))); % truncate (+
spectrum)
dig_x_axis = (1:length(fft_temp)) / (2*length(fft_temp));
plot(dig_x_axis, fft_temp)
title('FFT των Μεταδιδόμενα QAM')
% τον υπολογισμό της ΚΑΚ και κατάσταση για διαφάνεια 6
global BER_QAM_TEMP;
binary_err_bits_QAM = 0;
for i = 1:length(data_in)
    err = abs(data_in(i)-QAM_data_out(i));
    if err > 0
        binary_err_bits_QAM = binary_err_bits_QAM + 1;
    end
end
BER_QAM_TEMP = 100 * binary_err_bits_QAM/data_length;
COUNTER = COUNTER + 1;
pause

% disp('Slide 2b')
set(textHnd1,'String',stringArray(COUNTER,:))
hold on
% QAM Plotting
fft_temp = abs(fft(QAM_rx_data));
fft_temp = fft_temp(1:floor(0.5*length(fft_temp))); % truncate

```

```

    plot(dig_x_axis, fft_temp,'c'),title(' ')
    % channel display
    if channel_on == 1
        ComputeChannelGUI
        size_mag=max(mag)-min(mag); % για την κλίμακα οικόπεδο
        κανάλι
        plot(W/(2*pi),(0.5*max(fft_temp)/size_mag)*(mag
        abs(min(mag))) + 0.5*max(fft_temp),'k')
        end
    hold off
    COUNTER = COUNTER + 1;
    pause

    % disp('Slide 2c')
    set(textHnd1,'String','OFDM Simulation... Please Wait')
    OFDM
    set(textHnd1,'String',stringArray(COUNTER,:))
    fft_temp = abs(fft(xmit));
    fft_temp = fft_temp(1:floor(0.5*length(fft_temp))); % truncate
    dig_x_axis = (1:length(fft_temp)) / (2*length(fft_temp));
    plot(dig_x_axis, fft_temp)
    title('FFT των Μεταδιδόμενα OFDM')
    % τον υπολογισμό της BER και κατάσταση για διαφάνεια 6
    global BER_OFDM_TEMP;
    binary_err_bits_OFDM = 0;
    for i = 1:length(data_in)
        err = abs(data_in(i)-output(i));
        if err > 0
            binary_err_bits_OFDM = binary_err_bits_OFDM + 1;
        end
    end
    BER_OFDM_TEMP = 100 * binary_err_bits_OFDM/data_length;
    COUNTER = COUNTER + 1;
    pause

    % disp('Slide 2d')
    set(textHnd1,'String',stringArray(COUNTER,:))
    hold on
    % OFDM Plotting
    fft_temp = abs(fft(recv));
    fft_temp = fft_temp(1:floor(0.5*length(fft_temp))); % truncate
    plot(dig_x_axis, fft_temp,'c'),title(' ')
    % channel display
    if channel_on == 1
        plot(W/(2*pi),(0.5*max(fft_temp)/size_mag)*(mag
        abs(min(mag))) + 0.5*max(fft_temp),'k')
        end
    hold off

case 6

```

```

% disp('Slide 3')
setupSoundGUI
% hide main axis
    plot(0) % clear the plot
    axis off
    % set(axisHnd1,'Visible','off')
% show other axis's
    set(axisHnd2,'Visible','on')
    set(axisHnd3,'Visible','on')
    set(axisHnd4,'Visible','on')
% hide multipath controls
    set(textHnd2,'Visible','off')
    set(popupHnd1,'Visible','off')
% show generated sound buttons
    set(textHnd3,'Visible','on')
    set(OriginalHnd1,'Visible','on')
    set(QAMHnd1,'Visible','on')
    set(OFDMHnd1,'Visible','on')
% show long sounds stuff
    set(textHnd4,'Visible','on')
    set(OriginalLongHnd1,'Visible','on')
    set(QAMLongHnd1,'Visible','on')
    set(OFDMLongHnd1,'Visible','on')
% show the BER displays
    set(textHnd5,'Visible','on')
    set(textHnd6,'Visible','on')
    set(textHnd7,'Visible','on') % QAM
    set(textHnd8,'Visible','on') % OFDM
% Display the BERs
global BER_QAM_TEMP;
global BER_OFDM_TEMP;
set(textHnd7,'String',strcat(num2str(BER_QAM_TEMP,3),'%'))
set(textHnd8,'String',strcat(num2str(BER_OFDM_TEMP,3),'%'))
clear global BER_QAM_TEMP; % τον καθαρισμό των
διεθνοποιημένων
clear global BER_OFDM_TEMP;
% Plot the Sounds
% Note: axes(handle) sets to plot on the handle axis
axes(axisHnd2)
plot(wavread(file_name)),title('Original sound')
axes(axisHnd3)
plot(wavread('QAM_out.wav')),title('QAM sound')
axes(axisHnd4)
plot(wavread('OFDM_out.wav')),title('OFDM sound')
set(nextHnd1,'String','Start Over') % repeat if desired

otherwise
    disp('error')
    COUNTER = 0;
end

```

```

end

case 'mp_channel' %-----
    ComputeChannelGUI
    if noChannel ~= 1
        % large or small case
        plot(W/(2*pi),mag),axis([0 0.5 -12 6]),title('Κανάλι Μέγεθος απόκρισης')
        xlabel('ψηφιακή συχνότητα'),ylabel('Μέγεθος (dB)')
    else
        % none case
        plot(0:.05:.5,zeros(1,11)),axis([0 0.5 -12 6]),title('Κανάλι Μέγεθος
απόκρισης')
        xlabel('ψηφιακή συχνότητα'),ylabel('Μέγεθος (dB)')
    end

case 'close' %-----
    clear global COUNTER
    close(gcf)

case 'PlayOriginal' %-----
    sound(wavread('shortest.wav'),11025)

case 'PlayQAM' %-----
    sound(wavread('QAM_out.wav'),11025)

case 'PlayOFDM' %-----
    sound(wavread('OFDM_out.wav'),11025)

case 'PlayOriginalLong' %-----
    sound(wavread('Long.wav'),11025)

case 'PlayQAMLong' %-----
    sound(wavread('QAM_Long.wav'),11025)

case 'PlayOFDMLong' %-----
    sound(wavread('OFDM_Long.wav'),11025)

case 'figure' %-----
    % Αυτό καλείται όταν το ποσοστό αυτό δημιουργείται για πρώτη φορά ή OXI?
    % textHnd1=findobj('Tag','StaticTextFeedback');
    % axisHnd1=findobj('Tag','Axes1');
    % set(textHnd1,'String','Sound OFDM Demo') % default text message
    % set(axisHnd1,'Visible','off') % hide Axis to begin
end

% pol2bin
%
```



```

% Μετατρέπει πολική αριθμούς (-1,1) σε δυαδικούς αριθμούς (0,1)
% Δέχεται ένα 1-D διάταξη των πολικών αριθμούς
% Αφαιρεί μηδενικά, δεδομένου ότι δεν είναι έγκυρα δεδομένα

% % Αφαιρέστε μηδενικά - δεν χρειάζονται αποκωδικοποίηση με ευφυή
% last_data=length(x);
% while x(last_data) == 0
%   last_data = last_data - 1;
% end

y = ones(1,length(x));
for i = 1:length(x)
    if x(i) == -1
        y(i) = 0;
    end
end

QAM
% QAM.m συγκρίνει OFDM (πολλαπλού) σε πολλαπλά επίπεδα QAM (ενιαίο
φορέα)
% όταν μεταδίδουν την ίδια # bits σε μια δεδομένη χρονική περίοδο

read    % ανάγνωση δεδομένων για QAM - δεν επηρεάζει OFDM
data_in_pol = bin2pol(data_in);    % Μετατρέπει το δυαδικό δεδομένων σε πολικές
δεδομένα

% ελέγξτε για να δείτε αν num_carriers είναι μια δύναμη του 2
is_pow_2 = num_carriers;
temp_do_QAM = 0;
if is_pow_2 ~= 2
    while temp_do_QAM == 0
        temp_do_QAM = rem(is_pow_2,2);
        is_pow_2 = is_pow_2/2;
        if is_pow_2 == 2
            temp_do_QAM = -99;    % είναι μια δύναμη του 2 -> μπορεί να κάνει QAM
        end
    end
else
    temp_do_QAM = -99; % 2 είναι μια δύναμη του 2
end
if temp_do_QAM ~= -99
    do_QAM = 0; % δεν το κάνουμε, αν δεν είναι δυνατόν
    disp(' '),disp('ΣΦΑΛΜΑ: Δεν είναι δυνατή η εκτέλεση QAM επειδή num_carriers
δεν είναι έγκυρη.')
```

```

    disp(' Παρακαλώ δείτε "setup.m" για λεπτομέρειες.')
```

```

end

if do_QAM == 1
```

tic % Ξεκινήστε το χρονόμετρο να υπολογίσει πόσο καιρό παίρνει προσομοίωση QAM

```
disp(' '), disp('-----')
disp('QAM simulation'), disp('Transmitting')

% Pad with zeros so data can be divided evenly
data_length = length(data_in_pol);
r = rem(data_length,num_carriers);
if r ~= 0
    for i = 1:num_carriers-r
        data_in_pol(data_length+i) = 0; %εισόδου pad με μηδενικά για να
ολοκληρωθεί το τελευταίο σύνολο δεδομένων
    end %Ταχύτητα βελτίωση δυνατό
end
data_length = length(data_in_pol); %ενημέρωση μετά γεμίσει

num_OFDM_symbols = ceil(data_length / (2*num_carriers));
% σύμβολα QAM αριθμός που αντιπροσωπεύει ίση ποσότητα δεδομένων σε ένα
σύμβολο OFDM
num_QAM_symbols = num_carriers / 2;
% αριθμός δειγμάτων ανά σύμβολο QAM
num_symbol_samples = fft_size / num_QAM_symbols;

% μετατρέπουν πολικές δεδομένων [-1, 1], 4 επίπεδο, τα δεδομένα [-3, -1, 1, 3]
data_in_4 = zeros(1,data_length/2);
for i = 1:2:data_length
    data_in_4(i - (i-1)/2) = data_in_pol(i)*2 + data_in_pol(i+1);
end

% ορίζουν τα σημεία δείγματος μεταξύ 0 και 2 * pi
ts = linspace(0, 2*pi*QAM_periods, num_symbol_samples+1);

% Δημιουργία δεδομένων 16-QAM
% συνολικό μήκος της μετάδοσης 16-QAM
tx_length = num_OFDM_symbols * num_QAM_symbols * num_symbol_samples;
QAM_tx_data = zeros(1,tx_length);
for i = 1:2:data_length/2
    for k = 1:num_symbol_samples
        QAM_tx_data(k+((i-1)/2)*num_symbol_samples) = data_in_4(i)*cos(ts(k)) +
data_in_4(i+1)*sin(ts(k));
    end
end

%Κάνετε προσομοίωση κανάλι δεδομένων QAM

xmit = QAM_tx_data; % δεδομένα ch χρήσεις »XMIT» και επιστρέφει »recv»
ch
QAM_rx_data = recv; % αποθηκεύσετε τα δεδομένα QAM μετά κανάλι
clear recv % αφαίρεση 'recv »και έτσι δεν θα παρεμβαίνει με OFDM
```

```

clear xmit          % αφαίρεση 'recn »και έτσι δεν θα παρεμβαίνει με OFDM

disp('Receiving')   % Ανάκτηση δεδομένα Δυαδικά (Decode QAM)
cos_temp = zeros(1,num_symbol_samples); %
sin_temp = cos_temp; %
xxx = zeros(1,data_length/4); % Προετοιμαστεί για μηδενικά για ταχύτητα
yyy = xxx; %
QAM_data_out_4 = zeros(1,data_length/2); %

for i = 1:2:data_length/2 % "cheating"
    for k = 1:num_symbol_samples
        % πολλαπλασιάζουμε με φορείς για την παραγωγή υψηλής συχνότητας και
        % διάρκειας των αρχικών δεδομένων
        cos_temp(k) = QAM_rx_data(k+((i-1)/2)*num_symbol_samples) * cos(ts(k));
        sin_temp(k) = QAM_rx_data(k+((i-1)/2)*num_symbol_samples) * sin(ts(k));
    end
    % LPF και να αποφασίσει - θα κάνουμε πολύ απλό LPF από το μέσο όρο
    xxx(1+(i-1)/2) = mean(cos_temp);
    yyy(1+(i-1)/2) = mean(sin_temp);
    % Αναδόμηση δεδομένων σε σειριακή μορφή
    QAM_data_out_4(i) = xxx(1+(i-1)/2);
    QAM_data_out_4(i+1) = yyy(1+(i-1)/2);
end

% Κάντε την απόφασή μεταξύ [-3, -1, 1, 3]
for i = 1:data_length/2
    if QAM_data_out_4(i) >= 1, QAM_data_out_4(i) = 3;
    elseif QAM_data_out_4(i) >= 0, QAM_data_out_4(i) = 1;
    elseif QAM_data_out_4(i) >= -1, QAM_data_out_4(i) = -1;
    else QAM_data_out_4(i) = -3;
    end
end

% Μετατρέψτε 4 δεδομένα επίπεδο [-3, -1, 1, 3] επιστροφή σε πολικές δεδομένα [-
1, 1]
QAM_data_out_pol = zeros(1,data_length); % "cheating"
for i = 1:2:data_length
    switch QAM_data_out_4(1 + (i-1)/2)
        case -3
            QAM_data_out_pol(i) = -1;
            QAM_data_out_pol(i+1) = -1;
        case -1
            QAM_data_out_pol(i) = -1;
            QAM_data_out_pol(i+1) = 1;
        case 1
            QAM_data_out_pol(i) = 1;
            QAM_data_out_pol(i+1) = -1;
        case 3
            QAM_data_out_pol(i) = 1;
            QAM_data_out_pol(i+1) = 1;
    end
end

```

```

        otherwise
            disp('Σφάλμα ανιχνεύεται στο διακόπτη statment - Αυτό δεν πρέπει να
συμβαίνει.');
```

end

end

QAM_data_out = pol2bin(QAM_data_out_pol); % μετατρέψετε πάλι σε δυαδικό

% Σταματήστε το χρονόμετρο να υπολογίσει πόσο καιρό παίρνει προσομοίωση QAM

QAM_simulation_time = toc;

if QAM_simulation_time > 60

disp(strcat('Time for QAM simulation=', num2str(QAM_simulation_time/60), '
minutes.');

else

disp(strcat('Time for QAM simulation=', num2str(QAM_simulation_time), '
seconds.');

end

end

% read

% *****FILE INPUT

SETUP*****

if input_type == 2

if file_input_type == 1

%δυαδικό αρχείο εισόδου

end

if file_input_type == 2

%αρχείο εισόδου κειμένου

file = fopen(file_name,'rt');

data_samples = fread(file,'char');

fclose(file);

data_in = zeros(1,8*length(data_samples));

for i = 1:length(data_samples)

data_in(1 + (i-1)*8:(i-1)*8 + 8) = eight2bin(data_samples(i));

end

end

if file_input_type == 3

%είσοδο ήχου του αρχείου

data_samples=wavread(file_name);

%πρέπει να ομαλοποιηθούν από -1:1 έως 0:255 για μετατροπή 8 bit

data_samples_resized = round(128*data_samples +127);

data_in = zeros(1,8*length(data_samples_resized));

for i = 1:length(data_samples_resized)

data_in(1 + (i-1)*8:(i-1)*8 + 8) = eight2bin(data_samples_resized(i));

end

end

end

```

if file_input_type == 4
    %αρχείο εισόδου εικόνας
    [data_in,map]=imread(file_name);    % διαβάστε την εικόνα και το αντίστοιχο
    χρώμα του χάρτη για οθόνη

end

end

% rx

disp('Receiving')

rx_chunk

% εκτελέσει fft να ανακτήσει την αρχική δεδομένα από σύνολα πεδίο του χρόνου
recv_spaced_chunks = zeros(num_chunks,fft_size);
for i = 1:num_chunks
    recv_spaced_chunks(i,1:fft_size) = fft(recv_td_sets(i,1:fft_size));
    % Σημείωση: 'γύρο () «ξεφορτώνεται μικρή αριθμητική λάθος στην Matlab, αλλά
    θα χρειαστεί ένα κατώφλι για ένα πρακτικό σύστημα
    % 17.04.2001 - ξεφορτώθηκε »round () να κάνει αποκωδικοποίηση πιο έξυπνα
end

rx_dechunk

output = pol2bin(output); % Μετατρέπει πολική σε δυαδικό

write

% rx_chunk

% σπάσει λαμβανόμενο σήμα σε pallel set για αποδιαμόρφωση
recv_td_sets = zeros(num_chunks,fft_size);
for i = 1:num_chunks
    for k = 1:fft_size
        recv_td_sets(i,k) = recv(k + (i-1)*fft_size);
    end
end

% rx_dechunk

% βγάλτε μηδενικά ανάμεσα από recv απόσταση κομμάτια -> recv παραγεμισμένο
κομμάτια
recv_padded_chunks = zeros(num_chunks, num_carriers+num_zeros);
i = 1;
for k = zeros_between +1:zeros_between +1:fft_size/2
    recv_padded_chunks(1:num_chunks,i) = recv_spaced_chunks(1:num_chunks,k);
end

```

```

    i = i+1;
end

% βγάλτε num_zeros από παραγεμισμένο κομμάτι -> recv_chunks
recv_chunks = zeros(num_chunks, num_carriers);
recv_chunks = recv_padded_chunks(1:num_chunks,
num_zeros+1:num_carriers+num_zeros);

% Ανάκτηση ρεύμα bit με την τοποθέτηση των δεδομένων πεδίου
ανακατασκευάστηκε συχνότητα σε σειρά
recv_dechunked = zeros(1, num_chunks*num_carriers);
for i = 1:num_chunks
    for k = 1:num_carriers
        recv_dechunked(k + (i-1)*num_carriers*2) = real(recv_chunks(i,k));
        recv_dechunked(k + (i-1)*num_carriers*2 + num_carriers) =
imag(recv_chunks(i,k));
    end
end

% βγάλτε μηδενικά από την έξοδο -> έξοδος
output_analog = recv_dechunked(1:data_length);
output = sign(output_analog);

% setup
disp(' '), disp('-----')
disp('Simulation Setup')

% OFDM Setup -----
fft_size = 128 % θα πρέπει να είναι μια δύναμη του 2 για γρήγορους υπολογισμούς
% περισσότερα σημεία = δείγματα τομέα περισσότερο χρόνο (ομαλότερη &
περισσότερους κύκλους)
num_carriers = 32 % πρέπει να είναι <= fft_size / 4
% τον αριθμό των φορέων που χρησιμοποιούνται για κάθε κομμάτι
δεδομένων
% νέα var - δηλώνει ακόμα απόστασης ή παραλλαγές των μεταφορέων μεταξύ fft
σημεία
input_type = 2;
% 1 = test input
test_input_type = 1;
% 1 = bit καθορίζεται (δυναδικό)
binary_data = [0 1 0 1 0 1 0 1];
% 2 = τυχαία ροή δεδομένων (δείγματα στο εύρος 0-255)
num_symbols = 9;
% 3 = ημιτονοειδές
frequency = 2;
num_samples = 50;
% 2 = εξωτερική είσοδο του αρχείου
file_name = 'shortest.wav'; % Το όνομα του αρχείου εισόδου

```

```

file_input_type = 3;
    % 1 = δυαδικό (δεν έχει εφαρμοστεί)
    % 2 = text          % Demo file: 'text.txt'
    % 3 = sound        % Demo files: 'shortest.wav' & 'shorter.wav'
    % 4 = image (not implemented)

% QAM Setup -----
do_QAM = 1;    % (1=on, 0=off)
QAM_periods = 10; % καθορίζει τον αριθμό των περιόδων ανά QAM SymbOS (1 =
2 * π)

% Channel Simulation Parameters -----
channel_on = 1;    % 1=on, 0=off
clip_level = 1.0; % 0.0 - 1.0 (0-100%)
    % Μέγιστο μέγεθος του σήματος είναι φορές «clip_level» το πλήρες μέγεθος του
σήματος
noise_level = 0.0; % 0.0 - 1.0 (0-100%)
    % Μέγεθος του θορύβου είναι «noise_level» φορές το μέγεθος του σήματος
% Multipath Κανάλι Προσομοίωση
    % Καλή προεπιλογή όταν το μέγεθος fft = 128 και αριθμός φορέων = 32:
    % d1=6; a1=0.30; d2=10; a2=0.25
d1 = 6;    % καθυστέρηση σε μονάδες
a1 = 0.30; % παράγοντα εξασθένησης - σήμα πολλαπλών διαδρομών είναι x% του
μεγέθους ή αρχικού σήματος
d2 = 10;   % καθυστέρηση για τη δεύτερη σήματος πολλαπλών διαδρομών
a2 = 0.25; % συντελεστής εξασθένησης για δεύτερο σήμα πολλαπλών διαδρομών

% ***** TEST INPUT SETUP - DO NOT MODIFY *****
*****

if input_type == 1
    if test_input_type == 1
        %καθορίσετε δυαδική είσοδο bit-by-bit
        data_in = binary_data;
    end
    if test_input_type == 2
        %τυχαίας εισόδου καθορίζονται από παραμέτρους
        num_levels = 255; %αριθμός των πιθανών επιπέδων ενός συμβόλου
            %πρέπει να είναι ακέραιος αριθμός μεταξύ 1-255
        data_samples = round(rand(1,num_symbols)*(num_levels-1));
        data_in = zeros(1,8*length(data_samples));
        for i = 1:length(data_samples)
            data_in(1 + (i-1)*8:(i-1)*8 + 8) = eight2bin(data_samples(i));
        end
    end
    if test_input_type == 3
        %ρεύμα δεδομένων αναπαριστά δείγματα ημιτονοειδές κύμα

```

```

t = linspace(0,1,num_symbols); %ομοιόμορφα αριθμό χώρος των δειγμάτων
%λαμβάνουν δείγματα 8-bit των ημιτονοειδές κύμα
data_samples = round(127.5*sin(frequency*2*pi*t) +127.5);
data_in = zeros(1,8*length(data_samples));
for i = 1:length(data_samples)
    data_in(1 + (i-1)*8:(i-1)*8 + 8) = eight2bin(data_samples(i));
end
end
end

already_made_noise = 0; % προετοιμασίας (δεν αλλάζουν)

```

% SetupGUI.m θέτει τις μεταβλητές basicGUI

```

% Διαμορφώστε τις κατάλληλες μεταβλητές setup.m
fft_size = 64;
num_carriers = 4;
input_type = 1; test_input_type = 1;
channel_on = 0;
do_QAM = 0;
data_samples = [0 0 0 1 1 0 1 1]; % δεδομένα πρέπει να διαβιβάζονται
data_in = data_samples;

```

% SetupSoundGUI.m θέτει τις μεταβλητές SoundGUI

```

% Διαμορφώστε τις κατάλληλες μεταβλητές setup.m
fft_size = 128;
num_carriers = 32;
input_type = 2; file_input_type = 3; file_name = 'shortest.wav';
channel_on = 1;
do_QAM = 1;
QAM_periods = 10;
clip_level = 1.0; % 0.0 - 1.0 (0-100%)
noise_level = 0.0;
already_made_noise = 0;
ComputeChannelGUI

```

SoundGUI_win

```

% Αυτό είναι το μηχάνημα που δημιουργείται αναπαράσταση ενός αντικειμένου
γραφικών Λαβή
% και τα παιδιά της. Σημειώστε ότι οι τιμές λαβή μπορεί να αλλάξει όταν αυτά τα
αντικείμενα
% δημιουργούνται ξανά. Αυτό μπορεί να προκαλέσει προβλήματα με τυχόν
επιστροφές κλήσεων σε γραπτή
% εξαρτώνται από την αξία της λαβής κατά τη στιγμή το αντικείμενο έχει
αποθηκευτεί.
% Αυτό το πρόβλημα επιλύεται με την αποθήκευση της εξόδου ως ένα σχήμα-αρχείο.
%

```



```

% Για να ανοίξετε ξανά αυτό το αντικείμενο, απλώς πληκτρολογήστε το όνομα του
M-file στο MATLAB
% prompt. The M-file and its associated MAT-file must be on your path.
%
% ΣΗΜΕΙΩΣΗ: ορισμένα νεότερα στοιχεία σε MATLAB μπορεί να μην έχουν
αποθηκευτεί σε αυτό
% M-file, λόγω των περιορισμών αυτής της μορφής, η οποία έχει αντικατασταθεί από
% FIG-αρχεία. Τα σχήματα που έχουν σχολιαστεί χρησιμοποιώντας τα εργαλεία
επεξεργασίας οικόπεδο
% είναι ασυμβίβαστες με τη μορφή M-file/MAT-file, και θα πρέπει να αποθηκευτούν
ως
% FIG-files.

```

```
load SoundGUI_win
```

```

h0 = figure('Color',[0.9 0.9 0.9], ...
'Colormap',mat0, ...
'CreateFcn','OFDMguiFn figure', ...
'FileName','C:\TEMP\alanb_win_fix_GUI\ofdmproj_win\SoundGUI_win.m', ...
'PaperPosition',[18 180 576.00000000000001 432.00000000000002], ...
'PaperUnits','points', ...
'Position',[231 85 765 601], ...
'Tag','Fig1', ...
'ToolBar','none');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[1 1 1], ...
'FontName','Courier', ...
'HorizontalAlignment','left', ...
'ListboxTop',0, ...
'Position',[40.3448275862069      3.103448275862069      296.6896551724138
70.13793103448278], ...
'String','Sound OFDM Demo', ...
'Style','text', ...
'Tag','StaticTextFeedback');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[0.3 0.3 0.3], ...
'ListboxTop',0, ...
'Position',mat1, ...
'Style','frame', ...
'Tag','Frame1');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[0.733333 0.733333 0.733333], ...
'Callback','OFDMguiFnSound next', ...
'FontSize',14, ...
'ListboxTop',0, ...
'Position',[372.4137931034484      337.0344827586208      60.82758620689656
22.3448275862069], ...

```

```

    'String','Next', ...
    'Tag','PushbuttonNext');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[0.733333 0.733333 0.733333], ...
    'Callback','OFDMguiFnSound close', ...
    'FontSize',14, ...
    'ListboxTop',0, ...
    'Position',[372.4137931034484      6.827586206896553      61.44827586206898
24.82758620689656], ...
    'String','Close', ...
    'Tag','PushbuttonClose');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[0.733333 0.733333 0.733333], ...
    'Callback','OFDMguiFnSound mp_channel', ...
    'Enable','off', ...
    'ListboxTop',0, ...
    'Position',[372.4137931034484 180 68.89655172413795 20.48275862068966], ...
    'String',mat2, ...
    'Style','popupmenu', ...
    'Tag','PopupMenuMultipath', ...
    'Value',1, ...
    'Visible','off');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[0.733333 0.733333 0.733333], ...
    'Enable','off', ...
    'FontWeight','bold', ...
    'ListboxTop',0, ...
    'Position',mat3, ...
    'String','Multipath Channel', ...
    'Style','text', ...
    'Tag','StaticTextMultipath', ...
    'Visible','off');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[0.733333 0.733333 0.733333], ...
    'FontWeight','bold', ...
    'ListboxTop',0, ...
    'Position',[371.7931034482759      314.0689655172415      88.75862068965519
12.41379310344828], ...
    'String','Generated Sounds', ...
    'Style','text', ...
    'Tag','StaticTextGenSounds', ...
    'Visible','off');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[0.733333 0.733333 0.733333], ...
    'Callback','OFDMguiFnSound PlayOriginal', ...

```

```

'FontSize',14, ...
'ListboxTop',0, ...
'Position',[371.7931034482759      290.4827586206897      72.00000000000001
21.72413793103449], ...
'String','Original', ...
'Tag','PushbuttonOriginal', ...
'Visible','off');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[0.733333 0.733333 0.733333], ...
'Callback','OFDMguiFnSound PlayQAM', ...
'FontSize',14, ...
'ListboxTop',0, ...
'Position',[371.7931034482759      265.6551724137931      73.24137931034484
21.10344827586208], ...
'String','QAM', ...
'Tag','PushbuttonQAM', ...
'Visible','off');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[0.733333 0.733333 0.733333], ...
'Callback','OFDMguiFnSound PlayOFDM', ...
'FontSize',14, ...
'ListboxTop',0, ...
'Position',[371.7931034482759      242.0689655172414      73.24137931034484
20.48275862068966], ...
'String','OFDM', ...
'Tag','PushbuttonOFDM', ...
'Visible','off');
h1 = axes('Parent',h0, ...
'Units','points', ...
'Box','on', ...
'CameraUpVector',[0 1 0], ...
'CameraUpVectorMode','manual', ...
'Color',[1 1 1], ...
'ColorOrder',mat4, ...
'Position',[42.20689655172415      285.5172413793104      269.3793103448276
61.44827586206898], ...
'Tag','AxesOriginal', ...
'XColor',[0 0 0], ...
'YColor',[0 0 0], ...
'ZColor',[0 0 0]);
h2 = line('Parent',h1, ...
'Color',[0 0 1], ...
'Tag','Line1', ...
'XData',1, ...
'YData',0);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...

```

```

    'HorizontalAlignment','center', ...
    'Position',[0.9976905311778291 -1.571428571428571 17.32050807568877], ...
    'Tag','Text13', ...
    'VerticalAlignment','cap');
set(get(h2,'Parent'),'XLabel',h2);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','center', ...
    'Position',mat5, ...
    'Rotation',90, ...
    'Tag','Text14', ...
    'VerticalAlignment','baseline');
set(get(h2,'Parent'),'YLabel',h2);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','right', ...
    'Position',[-0.3187066974595843 1.836734693877551 17.32050807568877], ...
    'Tag','Text15', ...
    'Visible','off');
set(get(h2,'Parent'),'ZLabel',h2);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','center', ...
    'Position',[0.9976905311778291 1.163265306122449 17.32050807568877], ...
    'Tag','Text16', ...
    'VerticalAlignment','bottom');
set(get(h2,'Parent'),'Title',h2);
h1 = axes('Parent',h0, ...
    'Units','points', ...
    'Box','on', ...
    'CameraUpVector',[0 1 0], ...
    'CameraUpVectorMode','manual', ...
    'Color',[1 1 1], ...
    'ColorOrder',mat6, ...
    'Position',[43.44827586206898      206.0689655172414      268.1379310344828
58.96551724137932], ...
    'Tag','AxesQAM', ...
    'XColor',[0 0 0], ...
    'YColor',[0 0 0], ...
    'ZColor',[0 0 0]);
h2 = line('Parent',h1, ...
    'Color',[0 0 1], ...
    'Tag','Line2', ...
    'XData',1, ...
    'YData',0);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...

```

```

    'HandleVisibility','off', ...
    'HorizontalAlignment','center', ...
    'Position',[0.9976798143851507 -1.595744680851064 17.32050807568877], ...
    'Tag','Text9', ...
    'VerticalAlignment','cap');
set(get(h2,'Parent'),'XLabel',h2);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','center', ...
    'Position',[-0.1252900232018562 -0.02127659574468144 17.32050807568877], ...
    'Rotation',90, ...
    'Tag','Text10', ...
    'VerticalAlignment','baseline');
set(get(h2,'Parent'),'YLabel',h2);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','right', ...
    'Position',[-0.3294663573085847 4.680851063829786 17.32050807568877], ...
    'Tag','Text11', ...
    'Visible','off');
set(get(h2,'Parent'),'ZLabel',h2);
h2 = text('Parent',h1, ...
    'Color',[0 0 0], ...
    'HandleVisibility','off', ...
    'HorizontalAlignment','center', ...
    'Position',mat7, ...
    'Tag','Text12', ...
    'VerticalAlignment','bottom');
set(get(h2,'Parent'),'Title',h2);
h1 = axes('Parent',h0, ...
    'Units','points', ...
    'Box','on', ...
    'CameraUpVector',[0 1 0], ...
    'CameraUpVectorMode','manual', ...
    'Color',[1 1 1], ...
    'ColorOrder',mat8, ...
    'Position',[44.06896551724139      128.4827586206897      268.7586206896552
58.34482758620691], ...
    'Tag','AxesOFDM', ...
    'XColor',[0 0 0], ...
    'YColor',[0 0 0], ...
    'ZColor',[0 0 0]);
h2 = line('Parent',h1, ...
    'Color',[0 0 1], ...
    'Tag','Line3', ...
    'XData',1, ...
    'YData',0);
h2 = text('Parent',h1, ...

```

```

'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','center', ...
'Position',[0.9953703703703704 -1.602150537634408 17.32050807568877], ...
'Tag','Text5', ...
'VerticalAlignment','cap');
set(get(h2,'Parent'),'XLabel',h2);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','center', ...
'Position',mat9, ...
'Rotation',90, ...
'Tag','Text6', ...
'VerticalAlignment','baseline');
set(get(h2,'Parent'),'YLabel',h2);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','right', ...
'Position',[-0.3333333333333333 7.43010752688172 17.32050807568877], ...
'Tag','Text7', ...
'Visible','off');
set(get(h2,'Parent'),'ZLabel',h2);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','center', ...
'Position',[0.9953703703703704 1.172043010752688 17.32050807568877], ...
'Tag','Text8', ...
'VerticalAlignment','bottom');
set(get(h2,'Parent'),'Title',h2);
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[0.733333 0.733333 0.733333], ...
'FontWeight','bold', ...
'ListboxTop',0, ...
'Position',[371.7931034482759 142.7586206896552 85.65517241379313
12.41379310344828], ...
'String','Longer Sounds', ...
'Style','text', ...
'Tag','StaticTextLongSounds', ...
'Visible','off');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[0.733333 0.733333 0.733333], ...
'Callback','OFDMguiFnSound PlayQAMLong', ...
'FontSize',14, ...
'ListboxTop',0, ...

```

```

    'Position',[371.7931034482759      90.62068965517243      76.96551724137933
20.48275862068966], ...
    'String','QAM', ...
    'Tag','PushbuttonQAMLong', ...
    'Visible','off');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[0.733333 0.733333 0.733333], ...
    'Callback','OFDMguiFnSound PlayOFDMLong', ...
    'FontSize',14, ...
    'ListboxTop',0, ...
    'Position',[371.7931034482759      65.17241379310346      76.96551724137933
21.10344827586208], ...
    'String','OFDM', ...
    'Tag','PushbuttonOFDMLong', ...
    'Visible','off');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[0.733333 0.733333 0.733333], ...
    'Callback','OFDMguiFnSound PlayOriginalLong', ...
    'FontSize',14, ...
    'ListboxTop',0, ...
    'Position',mat10, ...
    'String','Original', ...
    'Tag','PushbuttonOriginalLong', ...
    'Visible','off');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[0.9 0.9 0.9], ...
    'ListboxTop',0, ...
    'Position',[315.9310344827587      242.0689655172414      26.06896551724138
11.17241379310345], ...
    'String','BER=', ...
    'Style','text', ...
    'Tag','StaticTextBER2', ...
    'Visible','off');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[0.9 0.9 0.9], ...
    'ListboxTop',0, ...
    'Position',[317.1724137931035      161.3793103448276      25.44827586206897
9.931034482758623], ...
    'String','BER=', ...
    'Style','text', ...
    'Tag','StaticTextBER1', ...
    'Visible','off');
h1 = uicontrol('Parent',h0, ...
    'Units','points', ...
    'BackgroundColor',[1 1 1], ...
    'HorizontalAlignment','left', ...

```

```

'ListboxTop',0, ...
'Position',mat11, ...
'Style','text', ...
'Tag','StaticTextBERQAM', ...
'Visible','off');
h1 = uicontrol('Parent',h0, ...
'Units','points', ...
'BackgroundColor',[1 1 1], ...
'HorizontalAlignment','left', ...
'ListboxTop',0, ...
'Position',[322.1379310344829    145.8620689655173    37.86206896551725
12.41379310344828], ...
'Style','text', ...
'Tag','StaticTextBEROFDM', ...
'Visible','off');
h1 = axes('Parent',h0, ...
'Units','points', ...
'Box','on', ...
'CameraUpVector',[0 1 0], ...
'Color',[1 1 1], ...
'ColorOrder',mat12, ...
'Position',mat13, ...
'Tag','Axes1', ...
'XColor',[0 0 0], ...
'YColor',[0 0 0], ...
'ZColor',[0 0 0]);
h2 = line('Parent',h1, ...
'Color',[0 0 1], ...
'Tag','Line4', ...
'XData',1, ...
'YData',0);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','center', ...
'Position',[0.9979959919839679 -1.14 17.32050807568877], ...
'Tag','Text4', ...
'VerticalAlignment','cap');
set(get(h2,'Parent'),'XLabel',h2);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','center', ...
'Position',[-0.1603206412825651 -0.004999999999999893 17.32050807568877],
...
'Rotation',90, ...
'Tag','Text3', ...
'VerticalAlignment','baseline');
set(get(h2,'Parent'),'YLabel',h2);
h2 = text('Parent',h1, ...

```



```

'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','right', ...
'Position',[-0.2164328657314629 1.155 17.32050807568877], ...
'Tag','Text2', ...
'Visible','off');
set(get(h2,'Parent'),'ZLabel',h2);
h2 = text('Parent',h1, ...
'Color',[0 0 0], ...
'HandleVisibility','off', ...
'HorizontalAlignment','center', ...
'Position',[0.9979959919839679 1.04 17.32050807568877], ...
'Tag','Text1', ...
'VerticalAlignment','bottom');
set(get(h2,'Parent'),'Title',h2);
if nargin > 0, fig = h0; end

% tx

disp('Transmitting')

read

data_in_pol = bin2pol(data_in); % Μετατρέπει το δυαδικό δεδομένων σε πολικές
δεδομένα

tx_chunk

% εκτελεί IFFT για τη δημιουργία του χρόνου κυματομορφή τομέα που
αντιπροσωπεύει τα δεδομένα
td_sets = zeros(num_chunks,fft_size);
for i = 1:num_chunks
    td_sets(i,1:fft_size) = real(iff(spaced_chunks(i,1:fft_size)));
end

tx_dechunk

% tx_chunk

data_length = length(data_in_pol); %αριθμό των συμβόλων στην αρχική
είσοδο
num_chunks = ceil(data_length/(2*num_carriers)); %Δεδομένων 2 για κάθε
μεταφορέα (πραγματικό και φανταστικό)
r = rem(data_length,2*num_carriers);

if r ~= 0
    for i = 1:num_carriers*2-r
        data_in_pol(data_length+i) = 0; %είσοδου pad με μηδενικά για να ολοκληρωθεί
το τελευταίο σύνολο δεδομένων
    end %Ταχύτητα βελτίωση δυνατό

```

```

end

% να σπάσει τα δεδομένα σε κομμάτια
chunks = zeros(num_chunks,num_carriers); % για ταχύτητα
for i = 1:num_chunks
    % *****chunk done
    for k = 1:num_carriers
        chunks(i,k) = data_in_pol(2*num_carriers*(i-1)+k) +
data_in_pol(2*num_carriers*(i-1)+k+num_carriers)*j;
    end
end

% Κομμάτια συμπλήρωση με μηδενικά ώστε num_carriers και fft_size είναι συμβατά
% Μόλις συμβατά, περαιτέρω απόσταση απλοποιείται
num_desired_carriers = num_carriers;
num_zeros = 0;
thinking = 1;
while thinking == 1 % Συνεχίστε αν οι μεταφορείς αριθμός και το μέγεθος fft δεν
είναι συμβατά
    if rem(fft_size/2,num_desired_carriers) == 0
        thinking = 0;
    else
        num_desired_carriers = num_desired_carriers + 1;
        num_zeros = num_zeros + 1;
    end
end

padded_chunks = zeros(num_chunks,num_carriers + num_zeros); % για ταχύτητα
padded_chunks(1:num_chunks,num_zeros + 1:num_carriers + num_zeros) = chunks;

% υπολογίσουμε μηδενικά μεταξύ
zeros_between = ((fft_size/2) - (num_carriers + num_zeros))/(num_carriers +
num_zeros);

spaced_chunks = zeros(num_chunks,fft_size); % για την ταχύτητα - επιπλέον χώρο
για την αναδίπλωση αργότερα
% προσθέστε μηδενικά μεταξύ
i = 1;
for k = zeros_between + 1:zeros_between + 1:fft_size/2
    spaced_chunks(1:num_chunks,k) = padded_chunks(1:num_chunks,i);
    i = i+1;
end

% αναδίπλωση δεδομένων για την παραγωγή μια περίεργη λειτουργία για την είσοδο
IFFT
for i = 1:num_chunks
    % Σημείωση: index = 1 είναι στην πραγματικότητα DC freq για IFFT -> δεν
παίρνει αντιγραφεί y-άξονα
    spaced_chunks(i,fft_size:-1:fft_size/2+2) = conj(spaced_chunks(i,2:fft_size/2));
end

```

```

end

% tx_dechunk

% Κατασκευάστε το σήμα για τη μετάδοση με την τοποθέτηση σετ πεδίο του χρόνου
σε σειρά
xmit = zeros(1,num_chunks*fft_size);
for i = 1:num_chunks
    for k = 1:fft_size
        xmit(k + (i-1)*fft_size) = td_sets(i,k);
    end
end

% write

% *****TEST OUTPUT*****
if input_type == 1

    if test_input_type == 1
        %ήδη δυαδική - δεν κάνουν τίποτα
    end

    if (test_input_type == 2) | (test_input_type == 3)
        %τυχαία δείγματα εισόδου ή ημιτονοειδές κύμα
        output_samples = zeros(1,floor(length(output)/8)); %επιπλέον μηδενικά δεν
είναι αρχικά δεδομένα
        for i = 1:length(output_samples)
            output_samples(i) = bin2eight(output(1 + (i-1)*8:(i-1)*8 + 8));
        end
        if do_QAM == 1
            QAM_output_samples = zeros(1,floor(length(QAM_data_out)/8));
            for i = 1:length(QAM_output_samples)
                QAM_output_samples(i) = bin2eight(QAM_data_out(1 + (i-1)*8:(i-1)*8 +
8));
            end
        end
    end
end

% *****FILE OUTPUT*****
if input_type == 2

    if file_input_type == 1
        %δυαδικό αρχείο εξόδου - δεν υλοποιούνται
    end

    if file_input_type == 2
        %έξοδο από το αρχείο κειμένου

```

```

    output_samples = zeros(1,floor(length(output)/8)); %επιπλέον μηδενικά δεν
    είναι αρχικά δεδομένα
    for i = 1:length(output_samples)
        output_samples(i) = bin2eight(output(1 + (i-1)*8:(i-1)*8 + 8));
    end
    file = fopen('OFDM_text_out.txt','wt+');
    fwrite(file,output_samples,'char');
    fclose(file);
    if do_QAM == 1
        QAM_output_samples = zeros(1,floor(length(QAM_data_out)/8));
        %επιπλέον μηδενικά δεν είναι αρχικά δεδομένα
        for i = 1:length(QAM_output_samples)
            QAM_output_samples(i) = bin2eight(QAM_data_out(1 + (i-1)*8:(i-1)*8 +
8));
        end
        file = fopen('QAM_text_out.txt','wt+');
        fwrite(file,QAM_output_samples,'char');
        fclose(file);
    end
end

if file_input_type == 3
    output_samples_big = zeros(1,floor(length(output)/8)); %επιπλέον μηδενικά δεν
    είναι αρχικά δεδομένα
    for i = 1:length(output_samples_big)
        output_samples_big(i) = bin2eight(output(1 + (i-1)*8:(i-1)*8 + 8));
    end
    %μετατρέψτε δυναμικό εύρος από 0:255 έως -1:1
    output_samples = (output_samples_big-127)/128;
    %έξοδο ήχου του αρχείου
    wavwrite(output_samples, 11025, 8, 'OFDM_out.wav')
    if do_QAM == 1
        QAM_data_out_big = zeros(1,floor(length(QAM_data_out)/8));
        for i = 1:length(QAM_data_out_big)
            QAM_data_out_big(i) = bin2eight(QAM_data_out(1 + (i-1)*8:(i-1)*8 +
8));
        end
        %μετατρέψτε δυναμικό εύρος από 0:255 έως -1:1
        QAM_output_samples = (QAM_data_out_big-127)/128;
        %sound file output
        wavwrite(QAM_output_samples, 11025, 8, 'QAM_out.wav')
    end
end

if file_input_type == 4
    %αρχείο εξόδου εικόνας - δεν υλοποιούνται
end

end

```

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. “Design and Simulation of Orthogonal Frequency Division Multiplexing (OFDM) Signaling”: Alan C. Brooks Stephen J. Hoelzer
2. ‘OFDM SIMULATION in MATLAB’ by : Paul Guanming Lin
June
3. “OFDM’’: john W. Thomas
4. “INTERNATIONAL JOURNAL OF ELECTRONICS AND COMMUNICATION ENGINEERING & TECHNOLOGY (IJCET)’’: Volume 4, Issue 3, May – June, 2013, pp. 162-176
5. Άνθρωπος και
επιστήμη: http://anthroposkaiepistimi.blogspot.gr/2011/12/blog-post_17.html
6. History of OFDMA and How it Works: http://www.webopedia.com/DidYouKnow/Computer_Science/OFDMA.asp
7. History of OFDM: <http://ratnuu.wordpress.com/2012/03/16/history-of-ofdm/>
8. Wikipedia: <http://en.wikipedia.org>
9. Math works: <http://www.mathworks.com/products/simulink>