ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Διαδραστική προσομοίωση Ψηφιακών Επικοινωνιών με ελεύθερο λογισμικό**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΑΡΑΒΑΝΤΙΝΟΥ-ΚΑΡΛΑΤΟΥ ΣΩΤΗΡΗ

Επιβλέπων

Μήτρου Νικόλαος

Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2024

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Διαδραστική προσομοίωση Ψηφιακών Επικοινωνιών με ελεύθερο λογισμικό**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΑΡΑΒΑΝΤΙΝΟΥ-ΚΑΡΛΑΤΟΥ ΣΩΤΗΡΗ

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την\_\_\_\_\_\_\_\_\_  
  
.............................. .............................. ..............................

Ρουσσάκη Ιωάννα  
Καθηγητής Ε.Μ.Π.

Σύκας Ευστάθιος  
Καθηγητής Ε.Μ.Π.

Μήτρου Νικόλαος  
Καθηγητής Ε.Μ.Π.

Μήτρου Νικόλαος  
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2024

.............................   
Αραβαντινός-Καρλάτος Γ. Σωτήρης   
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © Αραβαντινός-Καρλάτος Σωτήρης, 2024   
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

# Περίληψη

Η παρούσα διπλωματική εργασία εστιάζει στην ανάπτυξη μιας ιστοσελίδας με διαδραστικές δυνατότητες, η οποία παρέχει προσομοιώσεις συστημάτων ψηφιακής επικοινωνίας μέσω χρήσης ελεύθερου λογισμικού. Το έργο βασίζεται στη θεωρητική βάση των συστημάτων ψηφιακής επικοινωνίας, από τις σημειώσεις του μαθήματος Ψηφιακές Επικοινωνίες Ι, και προχωρά σε αναπαραστάσεις και προσομοιώσεις των βασικών κεφαλαίων που αφορούν την ψηφιακή επεξεργασία σήματος, τις τεχνικές ανίχνευσης και διαμόρφωσης, όπως και τα χαρακτηριστικά του φάσματος των ψηφιακών κυματομορφών.

Η εργασία περιλαμβάνει ανασκόπηση των εργαλείων λογισμικού που χρησιμοποιήθηκαν, αναλύει τα κριτήρια επιλογής τους και τεκμηριώνει τα πλεονεκτήματά τους σε προσομοιώσεις. Επιπλέον, καταγράφονται οι μεθοδολογίες και οι τεχνικές υλοποίησης που εφαρμόστηκαν για τη διαδραστική απεικόνιση της θεωρίας. Στα αποτελέσματα, παρουσιάζονται οι προσομοιώσεις με συγκρίσεις των αποτελεσμάτων με τις θεωρητικές αναλύσεις, καθώς και οι προκλήσεις που προέκυψαν.

Τέλος, η εργασία προτείνει μελλοντικές επεκτάσεις, όπως η εισαγωγή νέων εργαλείων και εκπαιδευτικών δραστηριοτήτων, ενώ παρέχει αναλυτική τεκμηρίωση του κώδικα και οδηγούς χρήσης στο παράρτημα.

**Λέξεις-Κλειδιά:** Ψηφιακές Επικοινωνίες, Προσομοίωση, Διαδραστική Ιστοσελίδα, Ελεύθερο Λογισμικό, Ψηφιακή Επεξεργασία Σήματος, Ψηφιακή Διαμόρφωση, QAM, PSK, FSK, MSK, Matched Filters, Χαρακτηριστικά Φάσματος, Jupyter Book, Python

.

# Abstract

This thesis focuses on the development of a website with interactive capabilities that provides simulations of digital communication systems using open-source software. The project covers the theoretical foundation of digital communication systems, based on the notes from the course Digital Communications I, and progresses to representations and simulations of the key chapters related to digital signal processing, detection, modulation techniques, and the spectral characteristics of digital waveforms.

The thesis includes a review of the software tools used, analyzes the criteria for their selection, and documents their advantages and performance in simulations. Furthermore, the methodologies and implementation techniques used for the interactive representation of the theory are outlined, with emphasis on the simulations developed for each chapter. The results section presents the simulations with comparisons of the outcomes with theoretical analyses, as well as the challenges encountered.

Finally, the thesis suggests future extensions, such as the introduction of new tools and educational activities, while also providing detailed code documentation and user guides in the appendix..

**Keywords:** Digital Communications, Simulation, Interactive Website, Open-Source Software, Digital Signal Processing, Digital Modulation, QAM, PSK, FSK, MSK, Matched Filters, Spectral Characteristics, Jupyter Book, Python.

# Ευχαριστίες

Αρχικά, θα ήθελα να εκφράσω τις ειλικρινείς μου ευχαριστίες προς τον επιβλέποντα καθηγητή μου, κ. Νικόλαο Μήτρου, για την εμπιστοσύνη που μου έδειξε αναθέτοντάς μου αυτή τη διπλωματική εργασία. Μέσα από αυτό το έργο, μου δόθηκε η ευκαιρία να εμβαθύνω σε ένα άκρως ενδιαφέρον και πολύτιμο αντικείμενο, που αφορά την ανάπτυξη μιας ιστοσελίδας με διαδραστικές δυνατότητες συστημάτων ψηφιακής επικοινωνίας χρησιμοποιώντας ελεύθερο λογισμικό. Θερμές ευχαριστίες αξίζουν και στην κ. Κωνσταντία Σακκά για την αδιάλειπτη υποστήριξη και καθοδήγησή της καθ' όλη τη διάρκεια της εκπόνησης αυτής της εργασίας.

Αυτή η διπλωματική εργασία σηματοδοτεί την ολοκλήρωση των προπτυχιακών μου σπουδών στο τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου. Καθώς φτάνω στο τέλος αυτής της πορείας, νιώθω την ανάγκη να εκφράσω τις θερμές μου ευχαριστίες στους γονείς μου, Γεράσιμο και Μαρία, για την αμέριστη στήριξή τους καθ’ όλη τη διάρκεια των σπουδών μου. Η υποστήριξή τους ήταν καθοριστική για την πρόοδό μου. Επίσης, ευχαριστώ τα αδέλφια μου που ήταν πάντα δίπλα μου σε αυτό το ταξίδι.

Τέλος, θα ήθελα να εκφράσω την ευγνωμοσύνη μου προς όλους τους φίλους που στάθηκαν στο πλευρό μου κατά τη διάρκεια των φοιτητικών μου χρόνων. Μαζί μοιραστήκαμε τις δυσκολίες των μαθημάτων, τις προκλήσεις των εργασιών και τις απαιτητικές εξεταστικές περιόδους, δημιουργώντας πολύτιμες αναμνήσεις που θα με συνοδεύουν για πάντα. Σας ευχαριστώ από καρδιάς όλους.

# Οργάνωση του τόμου

Ο παρών τόμος είναι οργανωμένος σε επτά κύρια κεφάλαια, τα οποία καλύπτουν την ανάπτυξη μιας διαδραστικής ιστοσελίδας με προσομοιώσεις συστημάτων ψηφιακής επικοινωνίας, καθώς και τη θεωρητική και πρακτική τους προσέγγιση μέσω ελεύθερου λογισμικού.

Στο πρώτο κεφάλαιο παρουσιάζεται η εισαγωγή στην εργασία, οι στόχοι που τέθηκαν, και η συνολική δομή της. Στο δεύτερο κεφάλαιο γίνεται μια ανασκόπηση της βιβλιογραφίας, με αναφορές στις σημειώσεις του μαθήματος Ψηφιακές Επικοινωνίες Ι και στις ενότητες που θα γίνουν διαδραστικές, καλύπτοντας θέματα όπως η ψηφιακή επεξεργασία σήματος, η ανίχνευση μέσω προσαρμοσμένων φίλτρων και οι διαμορφώσεις L-ASK, QAM, PSK, FSK και MSK.

Το τρίτο κεφάλαιο επικεντρώνεται στη μεθοδολογία που ακολουθήθηκε για την υλοποίηση των διαδραστικών προσομοιώσεων, με έμφαση στη μετάβαση από τη MATLAB στην Python και τη χρήση Jupyter Notebooks μέσα σε ένα Jupyter Book. Το τέταρτο κεφάλαιο παρουσιάζει τα αποτελέσματα των προσομοιώσεων και αναλύει τις προκλήσεις που προέκυψαν κατά τη διάρκεια της υλοποίησης.

Στο πέμπτο κεφάλαιο εξετάζονται οι μελλοντικές επεκτάσεις της εργασίας, με την εισαγωγή νέων σημειώσεων και εργαλείων που μπορούν να ενσωματωθούν. Το έκτο κεφάλαιο συνοψίζει το σύνολο της εργασίας παρουσιάζοντας περιληπτικά κάθε κομμάτι της, ενώ το έβδομο κεφάλαιο καταγράφει τις πηγές που χρησιμοποιήθηκαν για τη θεωρητική και πρακτική υποστήριξη της εργασίας.

Στα παραρτήματα παρέχονται αναλυτικοί οδηγοί εγκατάστασης των εργαλείων που χρησιμοποιήθηκαν, καθώς και τεκμηρίωση του κώδικα και οδηγίες χρήσης της τελικής διαδραστικής εφαρμογής που αναπτύχθηκε.

# Πίνακας περιεχομένων

[Περίληψη 2](#_Toc179297057)

[Abstract 3](#_Toc179297058)

[Ευχαριστίες 2](#_Toc179297059)

[Οργάνωση του τόμου 2](#_Toc179297060)

[Πίνακας περιεχομένων 3](#_Toc179297061)

[Κατάλογος Σχημάτων 11](#_Toc179297062)

[Εισαγωγή 12](#_Toc179297063)

[1. Στόχοι της Εργασίας 13](#_Toc179297064)

[2. Δομή της Εργασίας 14](#_Toc179297065)

[Ανασκόπηση της Βιβλιογραφίας 15](#_Toc179297066)

[1. Οι σημειώσεις του μαθήματος 16](#_Toc179297067)

[Chapter 1: Digital Signal Processing in Telecommunications 16](#_Toc179297068)

[Chapter 3: Optimal digital detection – Matched filters 18](#_Toc179297069)

[Chapter 4: Spectral Characteristics of Digital Waveforms & Nyquist signaling 19](#_Toc179297070)

[Chapter 5: Digital Modulation QAM and PSK 20](#_Toc179297071)

[Chapter 6: Digital Modulation FSK and MSK 21](#_Toc179297072)

[2. Εργαλεία που χρησιμοποιήθηκαν 22](#_Toc179297073)

[Εισαγωγή 22](#_Toc179297074)

[Διερεύνηση Εργαλείων 22](#_Toc179297075)

[Κριτήρια Επιλογής 28](#_Toc179297076)

[Επιλογή Εργαλείων 30](#_Toc179297077)

[Μεθοδολογία, Σχεδίαση και Υλοποίηση 41](#_Toc179297078)

[1. Εισαγωγή 42](#_Toc179297079)

[2. Μέθοδος υλοποίησης του συνολικού project 42](#_Toc179297080)

[3. Υλοποίηση διαδραστικών στοιχείων 44](#_Toc179297081)

[Chapter 1: Digital Signal Processing in Telecommunications 44](#_Toc179297082)

[Chapter 3: Optimal digital detection – Matched filters 49](#_Toc179297083)

[Chapter 4: Spectral Characteristics of Digital Waveforms & Nyquist signaling 50](#_Toc179297084)

[Chapter 5: Digital Modulation QAM and PSK 51](#_Toc179297085)

[Chapter 6: Digital Modulation FSK and MSK 52](#_Toc179297086)

[Αποτελέσματα 53](#_Toc179297087)

[1. Εισαγωγή 54](#_Toc179297088)

[2. Αποτελέσματα Προσομοιώσεων 55](#_Toc179297089)

[Chapter 1: Digital Signal Processing in Telecommunications 55](#_Toc179297090)

[Chapter 3: Optimal digital detection – Matched filters 62](#_Toc179297091)

[Chapter 4: Spectral Characteristics of Digital Waveforms & Nyquist signaling 64](#_Toc179297092)

[Chapter 5: Digital Modulation QAM and PSK 66](#_Toc179297093)

[Chapter 6: Digital Modulation FSK and MSK 69](#_Toc179297094)

[3. Συμπεράσματα 71](#_Toc179297095)

[Μελλοντικές επεκτάσεις 72](#_Toc179297096)

[1. Επέκταση σε Εργαστηριακές Ασκήσεις 73](#_Toc179297097)

[2. Εισαγωγή νέων εργαλείων 73](#_Toc179297098)

[Σύνοψη 74](#_Toc179297099)

[1. Συνοπτική παρουσίαση του σκοπού 75](#_Toc179297100)

[2. Περίληψη κύριων βημάτων και μεθοδολογίας 76](#_Toc179297101)

[3. Βασικά αποτελέσματα 77](#_Toc179297102)

[4. Συμπεράσματα 78](#_Toc179297103)

[Πηγές 79](#_Toc179297104)

[Παράρτημα Ι 80](#_Toc179297105)

[1. Πλήρης Οδηγός Εγκατάστασης 80](#_Toc179297106)

[Παράρτημα ΙΙ 84](#_Toc179297107)

[1. User Guide for Jupyter Book with Sphinx-Thebe 84](#_Toc179297108)

[Εκκίνηση Διαδραστικού Κώδικα 84](#_Toc179297109)

[Αλλαγή Εργαστηριακής Άσκησης ή Εργαλείου BER 85](#_Toc179297110)

[Διαδραστικότητα Στα Διαγράμματα 86](#_Toc179297111)

[Αν Δεν Φαίνεται το Αρχικό Διάγραμμα 86](#_Toc179297112)

[Αναμονή Χωρίς Αποτελέσματα 86](#_Toc179297113)

[Επανεκκίνηση και Εκτέλεση Όλων 86](#_Toc179297114)

[2. Documentation: Basic Elements of a Jupyter Book 87](#_Toc179297115)

[Δομή του Jupyter Book 88](#_Toc179297116)

[Αρχεία Περιεχομένου 89](#_Toc179297117)

[Ρυθμίσεις του Jupyter Book 89](#_Toc179297118)

[Προσθήκη Κώδικα 91](#_Toc179297119)

[Διαδραστικότητα με το Thebe 92](#_Toc179297120)

[Δημιουργία και Ανάπτυξη του Βιβλίου 92](#_Toc179297121)

[Συμπέρασμα 92](#_Toc179297122)

# Κατάλογος Σχημάτων

**No table of figures entries found.**

Κεφάλαιο 1

# Εισαγωγή

## **Στόχοι της Εργασίας**

Οι στόχοι της παρούσας εργασίας επικεντρώνονται στην αναζήτηση, αξιολόγηση, επιλογή και χρήση κατάλληλων εργαλείων που θα υποστηρίξουν τη δημιουργία μιας διαδραστικής ιστοσελίδας για την προσομοίωση συστημάτων ψηφιακών επικοινωνιών. Η πλατφόρμα αυτή στοχεύει να προσφέρει στους φοιτητές τη δυνατότητα να εξερευνούν και να κατανοούν καλύτερα τις αρχές των ψηφιακών επικοινωνιών, μέσα από πρακτικές εφαρμογές και πειραματισμό.

Ένας βασικός στόχος της εργασίας είναι να εντοπιστούν εργαλεία που προσφέρουν επαρκή διαδραστικότητα και ευκολία χρήσης, διατηρώντας παράλληλα την ευελιξία που απαιτείται για την προσομοίωση και ανάλυση συστημάτων σε πραγματικό χρόνο. Η έρευνα επικεντρώνεται σε εργαλεία ανοιχτού κώδικα, τα οποία επιτρέπουν την ενσωμάτωση διαδραστικών στοιχείων σε εκπαιδευτικά περιβάλλοντα.

Η γλώσσα προγραμματισμού Python, λόγω της ευρείας χρήσης της στις επιστήμες των δεδομένων και της ανάλυσης, αποτελεί κεντρικό εργαλείο στην υλοποίηση των στόχων αυτών. Τέλος, απαραίτητο στοιχείο θα αποτελέσει η ενσωμάτωση kernel στην ιστοσελίδα για τη ζωντανή εκτέλεση του κώδικα Python απευθείας στον φυλλομετρητή, εξασφαλίζοντας, έτσι, για τους χρήστες, έναν εύκολο και άμεσο τρόπο αλληλεπίδρασης με τις προσομοιώσεις σε πραγματικό χρόνο.

## Δομή της Εργασίας

Η δομή της παρούσας εργασίας βασίζεται σε μια οργανωμένη και συστηματική προσέγγιση της έρευνας, αξιολόγησης και επιλογής εργαλείων, που θα χρησιμοποιηθούν για την ανάπτυξη μιας διαδραστικής ιστοσελίδας για την προσομοίωση συστημάτων ψηφιακών επικοινωνιών. Η εργασία συνδέει θεωρητικά και πρακτικά εργαλεία που υποστηρίζουν την επίτευξη των στόχων της, όπως είναι η παροχή ενός εκπαιδευτικού περιβάλλοντος για την καλύτερη κατανόηση των βασικών αρχών των ψηφιακών επικοινωνιών μέσω πρακτικών εφαρμογών.

Στα αρχικά κεφάλαια, παρουσιάζεται η έρευνα και η αξιολόγηση των διαθέσιμων εργαλείων ανοιχτού κώδικα, που θα μπορούσαν να χρησιμοποιηθούν για την ανάπτυξη της πλατφόρμας. Ενώ έπειτα, γίνεται ανάλυση των εργαλείων που τελικά επιλέχθηκαν, με λεπτομέρειες για τις δυνατότητες που προσφέρουν στην υλοποίηση διαδραστικών προσομοιώσεων.

Παρακάτω, περιγράφεται η μεθοδολογία σχεδίασης και υλοποίησης της διαδραστικής πλατφόρμας, καταγράφοντας τους τρόπους με τους οποίους τα εργαλεία χρησιμοποιήθηκαν για την ανάπτυξη προσομοιώσεων για κάθε κεφάλαιο. Ύστερα, παρουσιάζονται τα αποτελέσματα από τις προσομοιώσεις και τις προκλήσεις που αντιμετωπίστηκαν.

Επιπλέον, αναφερόμαστε σε μελλοντικές επεκτάσεις της εργασίας, οι οποίες περιλαμβάνουν την προσθήκη των εργαστηριακών ασκήσεων από το μάθημα Ψηφιακές Επικοινωνίες Ι, καθώς και την ενσωμάτωση νέων εργαλείων που θα βελτιώσουν τη λειτουργικότητα της πλατφόρμας και τη μαθησιακή εμπειρία.

Στο τέλος της εργασίας, παρατίθενται οι αναφορές, ένας οδηγός εγκατάστασης εργαλείων, καθώς και η τεκμηρίωση του κώδικα και οι οδηγίες χρήσης για τις προσομοιώσεις που αναπτύχθηκαν.

Κατά αυτόν τον τρόπο, η εργασία οικοδομεί μια συνεκτική σύνδεση μεταξύ θεωρίας και πράξης, προκειμένου να επιτευχθεί η δημιουργία ενός πλήρως λειτουργικού διαδραστικού περιβάλλοντος μάθησης για τις ψηφιακές επικοινωνίες, συνδυάζοντας τις βέλτιστες πρακτικές από τα δύο πεδία.

Κεφάλαιο 2

# Ανασκόπηση της Βιβλιογραφίας

## Οι σημειώσεις του μαθήματος

Η ενότητα αυτή περιγράφει συνοπτικά τα βασικά κεφάλαια της θεωρίας του μαθήματος Ψηφιακές Επικοινωνίες Ι. Σε κάθε κεφάλαιο, προσδιορίζεται ο στόχος της θεωρητικής ανάλυσης και επισημαίνονται τα τμήματα θεωρίας για τα οποία, με βάση τις αντίστοιχες ερωτήσεις στις εργαστηριακές ασκήσεις, θα πρέπει να βρεθούν εργαλεία προσομοίωσής τους με διαδραστικό τρόπο.

Αυτά τα στοιχεία θα παρέχουν στους χρήστες τη δυνατότητα να αλληλεπιδρούν με τα συστήματα ψηφιακών επικοινωνιών σε πραγματικό χρόνο, αναλύοντας και τροποποιώντας τα σήματα, βοηθώντας έτσι στην κατανόηση της συμπεριφοράς τους σε διάφορες συνθήκες.

Καλύπτονται κρίσιμα θέματα, όπως η ψηφιακή επεξεργασία σήματος, οι βέλτιστες τεχνικές ανίχνευσης μέσω φίλτρων προσαρμογής, τα χαρακτηριστικά φάσματος ψηφιακών κυματομορφών, καθώς και οι διάφορες τεχνικές διαμόρφωσης ASK, QAM, PSK, FSK και MSK.

### Chapter 1: Digital Signal Processing in Telecommunications

#### Σύνοψη κεφαλαίου

Το κεφάλαιο αυτό εισάγει τη διαδικασία ψηφιοποίησης και την ψηφιακή επεξεργασία σημάτων στο πεδίο των τηλεπικοινωνιών. Αναλύονται τα βασικά ζητήματα που σχετίζονται με τη μετάβαση από το συνεχές στο διακριτό πεδίο μέσω της δειγματοληψίας, όπως η δημιουργία φασματικών εικόνων και το πρόβλημα του aliasing. Παρουσιάζεται ο Διακριτός Μετασχηματισμός Fourier, ως η φασματική αναπαράσταση σημάτων διακριτού χρόνου στο διακριτό πεδίο συχνοτήτων, και ο αναγνώστης εισάγεται στον σχεδιασμό και την υλοποίηση ψηφιακών φίλτρων. Δίνονται παραδείγματα και εργαστηριακές ασκήσεις για την ανάλυση φάσματος και τη χρήση ψηφιακών φίλτρων. Στα παραρτήματα περιλαμβάνονται βασικά θεωρήματα και ιδιότητες των μετασχηματισμών Fourier στον συνεχή και διακριτό χρόνο, σε συνοπτική μορφή πίνακα.

#### Κομμάτια που θα κάνουμε διαδραστικά

* Δημιουργία & Οπτικοποίηση σημάτων: Παραγωγή σημάτων για την προσομοίωση συστημάτων ψηφιακών επικοινωνιών και δημιουργία γραφημάτων για να απεικονιστούν τα χρονικά και φασματικά διαγράμματα των σημάτων.
* Διαδραστική επεξεργασία μεταβλητών σήματος: Με τη χρήση διαδραστικών εργαλίων, να μπορούν οι χρήστες να τροποποιούν τις παραμέτρους του σήματος και να παρατηρούν σε πραγματικό χρόνο τις αλλαγές.
* Προσθήκη θορύβου: Δυνατότητα προσθήκης θορύβου στο σήμα για παρακολούθηση των επιδράσεών του στην απόδοση της επικοινωνίας.
* Ανάλυση Φάσματος: Δημιουργία γραφημάτων φασματικής ανάλυσης.
* Απεικόνιση Φίλτρου (Filter Visualization): Δυνατότητα επιλογής διαφορετικών φίλτρων και οπτική αναπαράσταση της απόκρισης τους με μορφή stem plot.
* Ανάλυση Συχνοτήτων Φίλτρων (Filter Frequency Response): Διαδραστική απεικόνιση της φασματικής απόκρισης φίλτρων για διαφορετικά μήκη φίλτρων και παραθύρων.
* Ανάλυση Equiripple Φίλτρων: Δυνατότητα επιλογής equiripple φίλτρων διαφορετικών μηκών και σύγκριση της φασματικής τους απόκρισης.
* Εφαρμογή Χαμηλοπερατού Φίλτρου (Low Pass Filter Application): Εφαρμογή φίλτρων χαμηλής διέλευσης και απεικόνιση της απόκρισης συχνοτήτων του φιλτραρισμένου σήματος.
* Φασματική Πυκνότητα Ισχύος (Power Spectral Density): Διαδραστική σύγκριση φίλτρων equiripple και ανάλυση της φασματικής πυκνότητας ισχύος των φιλτραρισμένων σημάτων.
* Ανάλυση Απόκρισης Bandpass Φίλτρων: Εφαρμογή φίλτρων bandpass και οπτική απεικόνιση της απόκρισης τους τόσο στην συχνότητα όσο και στην φασματική πυκνότητα του φιλτραρισμένου σήματος.
* Εφαρμογή Parks-McClellan Φίλτρου: Σχεδιασμός φίλτρου bandpass με τη μέθοδο Parks-McClellan για επίτευξη συγκεκριμένης εξασθένισης στα stop bands.
* Εφαρμογή Φίλτρου με 2 passbands: Σχεδιασμός και υλοποίηση φίλτρου με δύο passbands στο MATLAB, με απεικόνιση της φασματικής του απόκρισης.

### Chapter 3: Optimal digital detection – Matched filters

#### Σύνοψη κεφαλαίου

Αυτό το κεφάλαιο παρουσιάζει τη μέθοδο βέλτιστης αναγνώρισης παλμών, οι οποίοι μεταφέρουν ψηφιακά δεδομένα και έχουν υποστεί παραμόρφωση κατά τη μετάδοση λόγω προσθετικού λευκού θορύβου. Η μέθοδος υλοποιείται μέσω του λεγόμενου προσαρμοσμένου φίλτρου (matched filter), το οποίο λειτουργεί ουσιαστικά ως συσχετιστής σήματος. Στον δέκτη, μια σειρά προσαρμοσμένων φίλτρων (που αντιστοιχούν στους διάφορους, γραμμικά ανεξάρτητους παλμούς του συμφωνημένου ρεπερτορίου μετάδοσης - γνωστό ως αστερισμός σημάτων) υπολογίζει τη συσχέτιση κάθε παλμού με τον εισερχόμενο παλμό (μέσω πολλαπλασιασμού και ολοκλήρωσης σε κάθε βασική περίοδο, Τ). Το φίλτρο με την υψηλότερη έξοδο υποδεικνύει τον παλμό με την μεγαλύτερη πιθανότητα εκπομπής (δηλαδή, την μεγαλύτερη πιθανότητα να έχει σταλεί, με βάση το ληφθέν σήμα).

Μετά την παρουσίαση των βασικών χαρακτηριστικών των προσαρμοσμένων φίλτρων, γίνεται γενίκευση σε Ν-διάστατους γραμμικούς χώρους σημάτων. Στη συνέχεια, εξετάζεται συνοπτικά η ανάλυση σφαλμάτων και δίνονται παραδείγματα αναγνώρισης ημιτονικών παλμών (FSK) και διαμόρφωσης πλάτους παλμού (PAM ή ASK).

#### Κομμάτια που θα κάνουμε διαδραστικά

* Δημιουργία τυχαίων σημάτων: Δημιουργία τυχαίων σημάτων και παρουσίαση του ιστογράμματός τους.
* Εφαρμογή θορύβου: Επίδραση του θορύβου στο σήμα με τη ρύθμιση του λόγου Eb/No μέσω ενός διαδραστικού slider και τον υπολογισμό του ιστόγραμματος για διάφορες τιμές.
* ASK Bit Error Rate (BER) Visualization: Η διαδραστική δυνατότητα σύγκρισης διαμορφώσεων Μ-ASK με επιλογές για πειραματική και θεωρητική τιμή του BER. Να μπορούν οι χρήστες να επιλέξουν μεταξύ διαφορετικών διαμορφώσεων, δειγμάτων ανά σύμβολο, και τύπου matched filter για να δουν τα αποτελέσματα σε πραγματικό χρόνο.
* Ανάλυση με φίλτρα: Οπτικοποιήσεις απόκρισης φίλτρων πάνω σε ορθογώνιο και ημιτονοειδές σήμα.

### Chapter 4: Spectral Characteristics of Digital Waveforms & Nyquist signaling

#### Σύνοψη κεφαλαίου

Στόχος αυτού του κεφαλαίου είναι η ανάλυση των χαρακτηριστικών φάσματος των ψηφιακών κυματομορφών και η παρουσίαση της σηματοδότησης Nyquist, η οποία χρησιμοποιείται για την ελαχιστοποίηση των παραμορφώσεων λόγω της διασποράς και την αποφυγή του φαινομένου της διασυμβολικής παρεμβολής. Το κεφάλαιο εισάγει τις αρχές της σηματοδότησης Nyquist και εξετάζει τα σχετικά θέματα, όπως το εύρος ζώνης και την αποτελεσματικότητα των σημάτων. Περιλαμβάνονται διαδραστικές ασκήσεις που εξετάζουν την αναπαράσταση του φάσματος των ψηφιακών κυματομορφών και την ανάλυση των συστημάτων που βασίζονται στη σηματοδότηση Nyquist, εστιάζοντας στη διασφάλιση της μέγιστης απόδοσης των σημάτων χωρίς παραμόρφωση.

#### Κομμάτια που θα κάνουμε διαδραστικά

* Ρυθμιζόμενες παράμετροι: Εισαγωγή διαδραστικού τρόπου ρύθμισης παραμέτρων μήκους bitstream, roll-off factor, nsamp (δείγματα ανά σύμβολο), group delay, και σειρά του φίλτρου.
* Οπτικοποίηση σημάτων και φίλτρων: Δημιουργία γραφημάτων για την παρουσίαση του φίλτρου, του φιλτραρισμένου σήματος, τη σύγκριση μεταξύ των αρχικών και φιλτραρισμένων σημάτων και την ανάλυση της φασματικής πυκνότητας ισχύος του ληφθέντος σήματος.
* BER καμπύλες: Με δυνατότητα επιλογής επιπέδου M-ASK, εμφάνισης των θεωρητικών και πειραματικών καμπυλών σε σχέση με το Eb/N0 και σύγκριση μεταξύ κωδικοποίησης Gray και Natural.

### Chapter 5: Digital Modulation QAM and PSK

#### Σύνοψη κεφαλαίου

Το κεφάλαιο αυτό εξετάζει δύο από τις πιο συχνά χρησιμοποιούμενες μεθόδους ψηφιακής διαμόρφωσης, τη διαμόρφωση QAM (Τετραγωνική Διαμόρφωση Πλάτους) και τη διαμόρφωση PSK (Διαμόρφωση Μετατόπισης Φάσης). Παρουσιάζει τη δομή του πομπού και του δέκτη για κάθε διαμόρφωση, την ανάλυση σφαλμάτων σε περιβάλλον λευκού θορύβου και τα φασματικά χαρακτηριστικά των σημάτων. Ο στόχος είναι η κατανόηση του τρόπου με τον οποίο τα συστήματα αυτά αξιοποιούνται σε συνθήκες πραγματικού θορύβου και η επίτευξη υψηλότερων ταχυτήτων μετάδοσης. Οι φοιτητές θα συμμετέχουν σε προσομοιώσεις που περιλαμβάνουν την υλοποίηση και ανάλυση σημάτων QAM και PSK, καθώς και την κατανόηση των επιπτώσεων του θορύβου και του φάσματος σε αυτά τα συστήματα διαμόρφωσης.

#### Κομμάτια που θα κάνουμε διαδραστικά

* QAM και PSK Constellations: Διαδραστικά διαγράμματα αστερισμών για L-QAM και M-PSK. Οι χρήστες να μπορούν να επιλέξουν το επίπεδο διαμόρφωσης και να δουν τις αντίστοιχες θέσεις των συμβόλων στον αστερισμό.
* QAM και PSK Καμπύλες BER: Δημιουργία γραφημάτων για τις καμπύλες BER τόσο για QAM όσο και για PSK διαμορφώσεις, όπου οι χρήστες μπορούν να συγκρίνουν την θεωρητική και πειραματική καμπύλη και να ρυθμίσουν παραμέτρους όπως το roll-off, τις συχνότητες f1 και f2, και το bitrate για την ανάλυση της πιθανότητας σφάλματος.
* Διαδραστική Φασματική Πυκνότητα Ισχύος: Δυνατότητα οπτικοποίησης της φασματικής πυκνότητας ισχύος για διάφορες διαμορφώσεις QAM και συχνοτήτων f1 και f2.
* Διαδραστικός Υπολογισμός Παραμέτρων: Δυνατότητα υπολογισμού παραμέτρων, με βάση τις παραμέτρους (R, M και roll-off) που επιλέγει ο χρήστης, όπως ο ρυθμός.

### Chapter 6: Digital Modulation FSK and MSK

#### Σύνοψη κεφαλαίου

Ο σκοπός του Το κεφάλαιο αυτό επικεντρώνεται στις τεχνικές ψηφιακής διαμόρφωσης FSK (Διαμόρφωση Μετατόπισης Συχνότητας) και MSK (Ελάχιστη Μετατόπιση Φάσης). Εξετάζονται οι βασικές αρχές αυτών των διαμορφώσεων, η συμπεριφορά τους σε συνθήκες θορύβου, οι πιθανότητες σφαλμάτων και τα φασματικά τους χαρακτηριστικά. Η MSK παρουσιάζεται ως μια ειδική περίπτωση της FSK που επιτυγχάνει ελάχιστο εύρος ζώνης και μεγαλύτερη αποδοτικότητα στην επικοινωνία. Οι φοιτητές θα εργαστούν σε προσομοιώσεις που αφορούν τη διαμόρφωση FSK και MSK, θα παρατηρήσουν τις φασματικές ιδιότητες των σημάτων και θα αξιολογήσουν την απόδοση αυτών των τεχνικών διαμόρφωσης υπό την επίδραση θορύβου.

#### Κομμάτια που θα κάνουμε διαδραστικά

* Αριθμός σφαλμάτων για Eb/No: Γράφημα στο οποίο ο χρήστης ρυθμίζει ένα εύρος του Eb/No, και στη συνέχεια εμφανίζεται η καμπύλη που απεικονίζει τον αριθμό των σφαλμάτων σε σχέση με το Eb/No για συστήματα FSK.
* FSK Bit Error Rate: Διαδραστική σύγκριση των καμπυλών BER για coherent και non coherent συστήματα FSK. Οι χρήστες μπορούν να δουν τις καμπύλες που αντιπροσωπεύουν την θεωρητική και πειραματική απόδοση.
* Ανάλυση Φάσματος coherent και non coherent FSK: Διαδραστικό γράφημα ανάλυσης της φασματικής πυκνότητας ισχύος τόσο για coherent όσο και για non coherent FSK συστήματα, όπου οι χρήστες μπορούν να προσαρμόσουν τις παραμέτρους όπως ο αριθμός συμβόλων και το Eb/No.
* MSK Bit Error Rate: Διαδραστικό γράφημα για σύγκριση μεταξύ των καμπυλών BER με και χωρίς precoding για τη διαμόρφωση MSK. Οι χρήστες μπορούν να δουν τις θεωρητικές και πειραματικές καμπύλες, συγκρίνοντας τις επιδόσεις για διαφορετικές τιμές Eb/No.

## Εργαλεία

### Εισαγωγή

|  |  |
| --- | --- |
| GitHub |  |

Για την οργάνωση του γενικότερου πλαισίου του έργου, την αποθήκευση των αρχείων και την τελική ανάπτυξη της διαδικτυακής εφαρμογής, χρησιμοποιήθηκε το GitHub (GitHub - Documentation, n.d.). Η πλατφόρμα αυτή επέτρεψε τη διαχείριση του κώδικα και την ασφαλή αποθήκευση των αρχείων του έργου, ενώ μέσω της υπηρεσίας GitHub Pages πραγματοποιήθηκε το τελικό deployment της εφαρμογής, προσφέροντας έναν εύκολο και αξιόπιστο τρόπο για τη δημοσίευση του περιεχομένου στο διαδίκτυο.

### Διερεύνηση Εργαλείων

Για την επίτευξη των παραπάνω διαδραστικών δυνατοτήτων θα χρειαστεί να βρεθούν εργαλεία γύρω από τις εξής κατηγορίες:

* Διαδραστική πλατφόρμα παρουσίασης του τελικού αποτελέσματος.
* Ενσωμάτωση kernel για εκτέλεση κώδικα σε πραγματικό χρόνο.
* Βιβλιοθήκες python για την προσομοίωση συστημάτων ψηφιακής επικοινωνίας.
* Γραφική αναπαράσταση δεδομένων.
* Διαδραστικοί μέθοδοι εισαγωγής παραμέτρων.

#### Διαδραστική πλατφόρμα παρουσίασης του τελικού αποτελέσματος.

Για την παρουσίαση του τελικού αποτελέσματος με διαδραστικό τρόπο, υπάρχουν διάφορες πλατφόρμες και εργαλεία που επιτρέπουν την ενσωμάτωση κώδικα, γραφημάτων και διαδραστικών στοιχείων σε ένα ενιαίο και προσβάσιμο περιβάλλον. Ακολουθεί η λίστα από αυτά που δοκιμάσαμε:

|  |  |
| --- | --- |
| 1. Jupyter Book: Εργαλείο για τη δημιουργία βιβλίων και τεκμηρίωσης από Jupyter Notebooks και αρχεία Markdown, με υποστήριξη διαδραστικών στοιχείων.   (Jupyter Book - Documentation, n.d.) |  |
| 1. Voila: Μετατρέπει Jupyter Notebooks σε αυτόνομες διαδραστικές web εφαρμογές χωρίς την ανάγκη γραφής JavaScript.   (Voila - Documentation, n.d.) |  |
| 1. MkDocs: Στατικός site generator που μετατρέπει αρχεία Markdown σε όμορφα τεκμηριωμένα ιστοσελίδες.   (MkDocs - Documentation, n.d.) |  |
| 1. Dash: Πλαίσιο για τη δημιουργία διαδραστικών web εφαρμογών για ανάλυση δεδομένων με Python.   (Dash - Documentation, n.d.) |  |
| 1. Google Colab: Διαδικτυακό περιβάλλον για την εκτέλεση Jupyter Notebooks, με υποστήριξη για Python και ενσωμάτωση με το Google Drive.   (Google Colab - Documentation, n.d.) |  |

#### Ενσωμάτωση kernel για εκτέλεση κώδικα σε πραγματικό χρόνο.

Για την ενσωμάτωση ενός kernel, που επιτρέπει την εκτέλεση κώδικα σε πραγματικό χρόνο μέσα από ιστοσελίδες, προχωρήσαμε μετά την επιλογή του Jupyter Book ως πλατφόρμα δομής της εφαρμογής μας. Έτσι, δοκιμάσαμε τις παρακάτω επιλογές που μας προσέφερε:

Διερεύνηση Εργαλείων

Μετά την επιλογή του Jupyter Book ως πλατφόρμας για τη δομή της εφαρμογής μας, εξετάσαμε τις επιλογές που προσφέρει το Jupyter Book για την ενσωμάτωση ενός kernel που επιτρέπει την εκτέλεση κώδικα σε πραγματικό χρόνο μέσα από ιστοσελίδες. Οι επιλογές που παρέχει το Jupyter Book είναι οι εξής:

|  |  |
| --- | --- |
| 1. notebook\_interface: Επιτρέπει την ενσωμάτωση των Jupyter Notebooks στο βιβλίο, παρέχοντας τη δυνατότητα προβολής του κώδικα και των αποτελεσμάτων του. Ωστόσο, δεν υποστηρίζει την εκτέλεση κώδικα σε πραγματικό χρόνο από τον χρήστη μέσα από την ιστοσελίδα. (Jupyter Book - Notebook Interface, n.d.) |  |
| 1. binderhub\_url: Παρέχει τη δυνατότητα σύνδεσης με ένα BinderHub server, επιτρέποντας την εκτέλεση των notebooks σε ένα απομακρυσμένο περιβάλλον. Οι χρήστες μπορούν να ανοίξουν τα notebooks σε ένα περιβάλλον Binder και να εκτελέσουν τον κώδικα. (Jupyter Book - Binder, n.d.) |  |
| 1. jupyterhub\_url: Επιτρέπει τη σύνδεση με ένα JupyterHub server, όπου οι χρήστες μπορούν να εκτελούν τα notebooks σε ένα κοινόχρηστο περιβάλλον εργασίας. (Jupyter Book - JupyterHub, n.d.) |  |
| 1. thebe: Μια JavaScript βιβλιοθήκη που μετατρέπει στατικές σελίδες HTML σε διαδραστικές, επιτρέποντας την εκτέλεση κώδικα απευθείας μέσα από την ιστοσελίδα. Το Thebe συνδέεται με έναν Jupyter kernel μέσω ενός backend, όπως το Binder. (Jupyter Book - Thebe, n.d.) |  |
| 1. colab\_url: Παρέχει συνδέσμους που επιτρέπουν στους χρήστες να ανοίξουν τα notebooks στο Google Colab, όπου μπορούν να εκτελέσουν τον κώδικα σε ένα cloud-based περιβάλλον. (Jupyter Book – Google Colab, n.d.) |  |

#### Βιβλιοθήκες python συστημάτων ψηφιακής επικοινωνίας

Για την προσομοίωση ψηφιακών συστημάτων στην Python, υπάρχουν αρκετές βιβλιοθήκες που προσφέρουν πλούσιες λειτουργίες και εργαλεία. Ακολουθεί η λίστα από αυτές που δοκιμάσαμε:

|  |  |
| --- | --- |
| 1. NumPy: Παρέχει υποστήριξη για πολυδιάστατους πίνακες και μαθηματικές συναρτήσεις υψηλής απόδοσης. (NumPy - Documentation, n.d.) |  |
| 1. SciPy: Προσφέρει ένα ευρύ φάσμα επιστημονικών και τεχνικών υπολογισμών, συμπεριλαμβανομένων σημάτων και επεξεργασίας εικόνων. (SciPy - Documentation, n.d.) | C:\Users\Sotiris\AppData\Local\Microsoft\Windows\INetCache\Content.Word\scipy.png |
| 1. scipy.signal: Υπο-πακέτο του SciPy, εξειδικευμένο στην επεξεργασία σημάτων, φίλτρων και συστημάτων. (scipy.signal - Documentation, n.d.) | C:\Users\Sotiris\AppData\Local\Microsoft\Windows\INetCache\Content.Word\scipy.png |
| 1. CommPy: Βιβλιοθήκη για την προσομοίωση τηλεπικοινωνιακών συστημάτων, συμπεριλαμβανομένων διαμορφώσεων, καναλιών και κωδικοποίησης. (CommPy - Documentation, n.d.) |  |
| 1. math: Ενσωματωμένη βιβλιοθήκη της Python με βασικές μαθηματικές συναρτήσεις. (Math Module - Documentation, n.d.) |  |
| 1. Matplotlib: Βιβλιοθήκη για γραφική απεικόνιση δεδομένων σε 2D και 3D. (Matplotlib - Documentation, n.d.) |  |
| 1. SimPy: Βιβλιοθήκη για την προσομοίωση διακριτών γεγονότων, χρήσιμη για συστήματα επικοινωνίας. (SimPy - Documentation, n.d.) |  |
| 1. PyWavelets: Βιβλιοθήκη για ανάλυση wavelet, χρήσιμη στην επεξεργασία σημάτων. (PyWavelets - Documentation, n.d.) |  |
| 1. scikit-dsp-comm: Παρέχει εργαλεία για επεξεργασία ψηφιακών σημάτων και επικοινωνιών. (scikit-dsp-comm - Documentation, n.d.) |  |
| 1. PySDR: Συνοδευτική βιβλιοθήκη για το βιβλίο "Python SDR", με εργαλεία για Software Defined Radio. (PySDR GitHub, n.d.) |  |

#### Γραφική αναπαράσταση δεδομένων

Από την πληθώρα βιβλιοθηκών Python για τη γραφική αναπαράσταση δεδομένων, έγινε έρευνα στις παρακάτω:

|  |  |
| --- | --- |
| 1. Matplotlib: Μια από τις πιο δημοφιλείς βιβλιοθήκες για τη δημιουργία στατικών, κινούμενων και διαδραστικών γραφημάτων στην Python. (Matplotlib - Documentation, n.d.) |  |
| 1. Seaborn: Βιβλιοθήκη που βασίζεται στο Matplotlib και προσφέρει υψηλού επιπέδου διεπαφές για στατιστικά γραφήματα. (Seaborn - Documentation, n.d.) |  |
| 1. Plotly: Βιβλιοθήκη για διαδραστικά γραφήματα που μπορούν να ενσωματωθούν σε ιστοσελίδες. (Plotly - Documentation, n.d.) |  |
| 1. Bokeh: Παρέχει δυνατότητες για διαδραστικά γραφήματα σε ιστοσελίδες μέσω Python. (Bokeh - Documentation, n.d.) |  |
| 1. PyPlot: Μέρος του Matplotlib. Εύκολη στη χρήση διεπαφή για τη δημιουργία γραφημάτων, παρόμοια με το MATLAB. (PyPlot - Documentation, n.d.) | **C:\Users\Sotiris\AppData\Local\Microsoft\Windows\INetCache\Content.Word\matplotlib.png** |
| 1. ggplot: Βιβλιοθήκη εμπνευσμένη από το ggplot2 της R, για δηλωτική δημιουργία γραφημάτων. (ggplot - Documentation, n.d.) |  |
| 1. Altair: Βιβλιοθήκη για δηλωτική στατιστική οπτικοποίηση βασισμένη στο Vega-Lite. (Altair - Documentation, n.d.) |  |
| 1. Pandas Visualization: Παρέχει απλές μεθόδους για τη δημιουργία γραφημάτων απευθείας από δεδομένα Pandas. (Pandas Visualization - Documentation, n.d.) |  |
| 1. Holoviews: Διευκολύνει τη δημιουργία σύνθετων γραφημάτων με λιγότερο κώδικα. (Holoviews - Documentation, n.d.) |  |
| 1. VisPy: Βιβλιοθήκη για γρήγορη οπτικοποίηση δεδομένων χρησιμοποιώντας GPU. (VisPy - Documentation, n.d.) |  |

#### Διαδραστικοί μέθοδοι εισαγωγής παραμέτρων.

Για την υλοποίηση διαδραστικών μεθόδων εισαγωγής παραμέτρων στην Python, σε περιβάλλοντα όπως τα Jupyter Notebooks, εξετάσαμε τα ακόλουθα εργαλεία:

|  |  |
| --- | --- |
| 1. IPython: Προηγμένη διεπαφή για την Python που παρέχει διαδραστικό περιβάλλον εργασίας, συμπεριλαμβανομένων δυνατοτήτων όπως τα Jupyter Notebooks. (IPython - Documentation, n.d.) |  |
| 1. ipywidgets: Βιβλιοθήκη που επιτρέπει τη δημιουργία διαδραστικών widget σε Jupyter Notebooks, διευκολύνοντας την εισαγωγή παραμέτρων και την αλληλεπίδραση με τον κώδικα. (ipywidgets - Documentation, n.d.) |  |
| 1. Bokeh: Βιβλιοθήκη για τη δημιουργία διαδραστικών οπτικοποιήσεων σε ιστοσελίδες, υποστηρίζοντας widgets για εισαγωγή παραμέτρων. (Bokeh - Documentation, n.d.) |  |
| 1. Plotly Dash: Πλατφόρμα για τη δημιουργία διαδραστικών web εφαρμογών με Python, επιτρέποντας την εισαγωγή παραμέτρων μέσω διαφόρων στοιχείων διεπαφής. (Plotly Dash - Documentation, n.d.) |  |
| 1. Streamlit: Πλαίσιο για τη δημιουργία web εφαρμογών για μηχανική μάθηση και επιστήμη δεδομένων με απλή σύνταξη Python. (Streamlit - Documentation, n.d.) |  |
| 1. Voila: Επιτρέπει τη μετατροπή Jupyter Notebooks σε διαδραστικές web εφαρμογές χωρίς την ανάγκη γραφής κώδικα JavaScript. (Voila - Documentation, n.d.) |  |
| 1. Panel: Βιβλιοθήκη για τη δημιουργία διαδραστικών dashboards και εφαρμογών, υποστηρίζοντας πολλαπλές βιβλιοθήκες γραφικών. (Panel - Documentation, n.d.) |  |
| 1. Gradio: Επιτρέπει τη δημιουργία απλών διεπαφών χρήστη για μηχανική μάθηση και επιστήμη δεδομένων. (Gradio - Documentation, n.d.) |  |
| 1. PySimpleGUI: Παρέχει μια απλή διεπαφή για τη δημιουργία GUI εφαρμογών με Python. (PySimpleGUI - Documentation, n.d.) |  |
| 1. Tkinter: Ενσωματωμένη βιβλιοθήκη της Python για τη δημιουργία GUI εφαρμογών. (Tkinter - Documentation, n.d.) |  |

### Κριτήρια Επιλογής

Για την επιλογή των κατάλληλων εργαλείων, λάβαμε υπόψη τα ακόλουθα κριτήρια:

* Συμβατότητα με τις Απαιτήσεις της Προσομοίωσης: Πρέπει να παρέχει τις απαραίτητες λειτουργίες για την προσομοίωση των ψηφιακών συστημάτων που καλύπτονται στο μάθημα.
* Ευκολία Χρήσης: Το εργαλείο πρέπει να είναι εύκολο στην εκμάθηση και χρήση, με απλή σύνταξη και κατανοητές λειτουργίες.
* Δυνατότητες Διαδραστικότητας: Η δυνατότητα δημιουργίας διαδραστικών εφαρμογών που επιτρέπουν στον χρήστη να αλληλεπιδρά με τα δεδομένα σε πραγματικό χρόνο.
* Υποστήριξη Κοινότητας: Μια ενεργή κοινότητα χρηστών και προγραμματιστών εξασφαλίζει την συνεχή βελτίωση του εργαλείου και την επίλυση προβλημάτων.
* Διαθεσιμότητα Τεκμηρίωσης: Η ύπαρξη εκτενούς και κατανοητής τεκμηρίωσης είναι απαραίτητη για την αποτελεσματική χρήση του εργαλείου.
* Απόδοση και Ταχύτητα: Οι βιβλιοθήκες πρέπει να είναι αποδοτικές και να επιτρέπουν την επεξεργασία μεγάλων όγκων δεδομένων σε εύλογο χρόνο.
* Συμβατότητα: Η δυνατότητα ενσωμάτωσης με άλλα εργαλεία για την ανάπτυξη της ολοκληρωμένης εφαρμογής.
* Υποστήριξη Markdown και Κώδικα: Η δυνατότητα συγγραφής σε Markdown και η ενσωμάτωση κώδικα είναι απαραίτητη για την παρουσίαση της θεωρίας και των παραδειγμάτων.
* Δυνατότητες Οργάνωσης Περιεχομένου: Πρέπει να επιτρέπει την οργάνωση του υλικού σε κεφάλαια, ενότητες και υποενότητες, με εύκολη πλοήγηση.
* Δυνατότητες Δημοσίευσης: Η πλατφόρμα πρέπει να επιτρέπει την εύκολη δημοσίευση του υλικού στο διαδίκτυο, ώστε να είναι προσβάσιμο από τους φοιτητές.
* Ωριμότητα και Σταθερότητα: Προτίμηση εργαλείων που έχουν δοκιμαστεί στον χρόνο και θεωρούνται σταθερά.

### Επιλογή Εργαλείων

Με βάση τα παραπάνω κριτήρια, και έπειτα από εκτενής δοκιμές πάνω σε κάθε εργαλείο, επιλέξαμε να χρησιμοποιήσουμε τα ακόλουθα:

#### Διαδραστική πλατφόρμα παρουσίασης του τελικού αποτελέσματος.

|  |  |
| --- | --- |
| Jupyter Book  (Jupyter Book - Documentation, n.d.) |  |

Γιατί το επιλέξαμε:

* Ενσωμάτωση με Jupyter Notebooks: Το Jupyter Book σχεδιάστηκε για να μετατρέπει Jupyter Notebooks, αρχεία Markdown και αρχεία MyST Markdown (μία επέκταση του προηγούμενου) σε ένα συνεκτικό βιβλίο, επιτρέποντας την άμεση αξιοποίηση του ήδη υπάρχοντος υλικού.
* Υποστήριξη Διαδραστικότητας: Επιτρέπει την ενσωμάτωση διαδραστικών στοιχείων, όπως τα ipywidgets, και υποστηρίζει την εκτέλεση κώδικα μέσα στις σελίδες του βιβλίου.
* Εύκολη Οργάνωση Περιεχομένου: Παρέχει εργαλεία για την οργάνωση του υλικού σε κεφάλαια και ενότητες, με αυτόματη δημιουργία πίνακα περιεχομένων και πλοήγησης.
* Δυνατότητες Προσαρμογής: Προσφέρει θέματα και επιλογές διαμόρφωσης για την προσαρμογή της εμφάνισης και της λειτουργικότητας του βιβλίου.
* Εύκολη Δημοσίευση: Μπορεί να δημοσιευτεί σε διάφορες πλατφόρμες, όπως το GitHub Pages, καθιστώντας το προσβάσιμο μέσω web browser.
* Υποστήριξη Κοινότητας: Ως μέρος του οικοσυστήματος Jupyter, έχει μια ενεργή κοινότητα και συνεχή ανάπτυξη.

Πλεονεκτήματα:

* Συνεκτική Παρουσίαση: Ενοποιεί κείμενο, κώδικα και διαδραστικά στοιχεία σε ένα ενιαίο περιβάλλον.
* Υποστήριξη Εκτελέσιμου Κώδικα: Επιτρέπει στους χρήστες να εκτελούν κώδικα μέσα στο βιβλίο, ενισχύοντας την διαδραστικότητα.
* Προσαρμογή Δομής: Επιτρέπει την προσαρμογή των συνδέσμων και της δομής του βιβλίου για καλύτερη πλοήγηση.
* Δυνατότητα Χρήσης Επεκτάσεων: Μπορεί να επεκταθεί με πρόσθετες λειτουργίες μέσω επεκτάσεων (π.χ. sphinx-inline tabs, thebe)

**Πώς συνέβαλε στην επίτευξη του στόχου της εργασίας**

* Ενοποίηση Υλικού: Το Jupyter Book μας επέτρεψε να συγκεντρώσουμε όλα τα κεφάλαια, τις σημειώσεις, τον κώδικα και τα διαδραστικά στοιχεία σε ένα ενιαίο και εύκολα προσβάσιμο βιβλίο.
* Διαδραστική Μάθηση: Μέσω του live code run που θα εξηγηθεί μετά. Οι φοιτητές μπορούν να αλληλεπιδρούν με τα διαδραστικά στοιχεία, να τροποποιούν παραμέτρους και να βλέπουν τα αποτελέσματα σε πραγματικό χρόνο μέσα από το βιβλίο.
* Ευκολία Συντήρησης και Ενημέρωσης: Οι ενημερώσεις στο περιεχόμενο και τον κώδικα μπορούν να γίνονται εύκολα και να δημοσιεύονται άμεσα.
* Υποστήριξη Συνεργασίας: Η χρήση του Git για τη διαχείριση του περιεχομένου επιτρέπει τη συνεργασία μεταξύ των εκπαιδευτών και τη συνεισφορά από πολλαπλούς χρήστες.
* Δυνατότητα Επεκτασιμότητας: Το Jupyter Book μπορεί να επεκταθεί και να προσαρμοστεί για μελλοντικές ανάγκες, όπως η προσθήκη νέων κεφαλαίων ή διαδραστικών στοιχείων.

**Συμπέρασμα**

Η επιλογή του Jupyter Book ως πλατφόρμας παρουσίασης του τελικού αποτελέσματος ανταποκρίνεται πλήρως στις ανάγκες της εργασίας. Μας επέτρεψε να δημιουργήσουμε ένα ολοκληρωμένο, διαδραστικό και εύκολα προσβάσιμο εκπαιδευτικό υλικό, που ενισχύει την κατανόηση των ψηφιακών επικοινωνιών και παρέχει στους φοιτητές ένα πολύτιμο εργαλείο μάθησης.

#### Ενσωμάτωση kernel για εκτέλεση κώδικα σε πραγματικό χρόνο.

Αφού αξιολογήσαμε τις διαθέσιμες επιλογές, επιλέξαμε να χρησιμοποιήσουμε το Thebe, δημιουργώντας έναν kernel στο Binder.

|  |  |
| --- | --- |
| Thebe  (Jupyter Book - Thebe, n.d.) |  |

Γιατί το επιλέξαμε:

* Διαδραστικότητα μέσα από το Jupyter Book: Το Thebe ενσωματώνεται εύκολα στο Jupyter Book, επιτρέποντας τη μετατροπή των στατικών κελιών κώδικα σε διαδραστικά. Οι χρήστες μπορούν να εκτελούν και να τροποποιούν τον κώδικα απευθείας μέσα από την ιστοσελίδα.
* Ευκολία Ενσωμάτωσης: Απαιτεί ελάχιστες αλλαγές στο αρχείο διαμόρφωσης του Jupyter Book. Με την προσθήκη `use\_thebe: true`, ενεργοποιείται η λειτουργικότητα του Thebe.
* Υποστήριξη Python και Βιβλιοθηκών: Λειτουργεί με Jupyter kernels, υποστηρίζοντας πλήρως την Python και τις απαιτούμενες βιβλιοθήκες μας, όπως NumPy, SciPy και Matplotlib.
* Εμπειρία Χρήστη: Οι χρήστες δεν χρειάζεται να εγκαταστήσουν πρόσθετο λογισμικό ή να φύγουν από την ιστοσελίδα. Η αλληλεπίδραση με τον κώδικα γίνεται ομαλά και άμεσα.

|  |  |
| --- | --- |
| Binder  (Jupyter Book - Binder, n.d.) |  |

Γιατί το επιλέξαμε:

* Σύνδεση με Thebe: Το Binder μπορεί να χρησιμοποιηθεί ως backend για το Thebe, παρέχοντας τον απαραίτητο Jupyter kernel για την εκτέλεση του κώδικα.
* Ευκολία Ρύθμισης: Η δημιουργία ενός περιβάλλοντος στο Binder γίνεται μέσω ενός public Git repository, όπου καθορίζουμε τις εξαρτήσεις και τις βιβλιοθήκες που χρειάζονται.
* Δωρεάν Υπηρεσία: Το mybinder.org προσφέρει δωρεάν hosting, καθιστώντας το ιδανικό για εκπαιδευτικούς σκοπούς και επιτρέποντας την ευρεία πρόσβαση χωρίς κόστος.
* Αναπαραγωγιμότητα: Διασφαλίζει ότι όλοι οι χρήστες εκτελούν τον κώδικα σε ένα σταθερό και ελεγχόμενο περιβάλλον, αποφεύγοντας προβλήματα συμβατότητας.

**Πώς συνέβαλαν στην επίτευξη του στόχου της εργασίας**

* Διαδραστική Εκτέλεση Κώδικα: Με το Thebe, οι φοιτητές μπορούν να εκτελούν κώδικα σε πραγματικό χρόνο μέσα από τις σελίδες του Jupyter Book, τροποποιώντας παραμέτρους και παρατηρώντας άμεσα τις επιπτώσεις στα αποτελέσματα και τα γραφήματα.
* Ενσωμάτωση με το Jupyter Book: Η λύση λειτουργεί άψογα με τη δομή και το περιεχόμενο που έχουμε δημιουργήσει, χωρίς την ανάγκη σημαντικών αλλαγών ή πρόσθετων ρυθμίσεων.
* Ευκολία Πρόσβασης: Οι φοιτητές χρειάζονται μόνο έναν web browser για να έχουν πλήρη πρόσβαση στο υλικό και τη διαδραστική λειτουργικότητα, διευκολύνοντας την εξ αποστάσεως μάθηση.
* Αναπαραγωγιμότητα και Συνέπεια: Το Binder εξασφαλίζει ότι όλοι οι χρήστες εκτελούν τον κώδικα στο ίδιο περιβάλλον, με τις ίδιες βιβλιοθήκες και εκδόσεις, παρέχοντας συνεπή αποτελέσματα.
* Ασφάλεια: Το Binder παρέχει ένα ασφαλές περιβάλλον εκτέλεσης, απομονώνοντας τον κώδικα των χρηστών και προστατεύοντας τους διακομιστές από κακόβουλες ενέργειες.
* Υποστήριξη Κοινότητας: Η ενεργή κοινότητα γύρω από το Thebe και το Binder μας επέτρεψε να αξιοποιήσουμε παραδείγματα, να επιλύσουμε προβλήματα και να προσαρμόσουμε τη λύση στις ανάγκες μας.

**Συμπέρασμα**

Με την επιλογή του Thebe και τη δημιουργία ενός kernel στο Binder, καταφέραμε να ενσωματώσουμε έναν διαδραστικό kernel στο Jupyter Book, επιτρέποντας την εκτέλεση κώδικα σε πραγματικό χρόνο μέσα από τις ιστοσελίδες μας. Αυτή η λύση ανταποκρίνεται πλήρως στις ανάγκες μας για ένα ευέλικτο, προσβάσιμο και διαδραστικό εκπαιδευτικό εργαλείο, ενισχύοντας την εμπειρία μάθησης των φοιτητών και διευκολύνοντας την κατανόηση των ψηφιακών συστημάτων επικοινωνιών.

#### Βιβλιοθήκες python συστημάτων ψηφιακής επικοινωνίας

|  |  |
| --- | --- |
| 1. Numpy   (NumPy - Documentation, n.d.) |  |

Γιατί το επιλέξαμε: Αποτελεί τη βάση για επιστημονικούς υπολογισμούς στην Python. Παρέχει αποδοτικούς πολυδιάστατους πίνακες και λειτουργίες που είναι απαραίτητες για την επεξεργασία σημάτων.

Πλεονεκτήματα:

* Υψηλή απόδοση σε αριθμητικούς υπολογισμούς.
* Ευρεία χρήση και μεγάλη κοινότητα.
* Εύκολη ενσωμάτωση με άλλες βιβλιοθήκες.

|  |  |
| --- | --- |
| 1. SciPy   (SciPy - Documentation, n.d.) |  |

Γιατί το επιλέξαμε: Προσφέρει προηγμένες επιστημονικές λειτουργίες που δεν υπάρχουν στο NumPy, ιδιαίτερα στον τομέα της επεξεργασίας σημάτων.

Πλεονεκτήματα:

* Πλούσια συλλογή αλγορίθμων και λειτουργιών.
* Υπο-πακέτα όπως το `scipy.signal` είναι εξειδικευμένα στην επεξεργασία σημάτων.

|  |  |
| --- | --- |
| 1. scipy.signal   (scipy.signal - Documentation, n.d.) |  |

Γιατί το επιλέξαμε: Εξειδικευμένο στην ανάλυση και επεξεργασία σημάτων, παρέχει εργαλεία για φίλτρα, μετασχηματισμούς Fourier και άλλα.

Πλεονεκτήματα:

* Ευρεία γκάμα εργαλείων για σχεδίαση και ανάλυση φίλτρων.
* Λειτουργίες για τη δημιουργία και επεξεργασία σημάτων.

|  |  |
| --- | --- |
| 1. commpy   (CommPy - Documentation, n.d.) |  |

Γιατί το επιλέξαμε: Είναι μια βιβλιοθήκη σχεδιασμένη για προσομοίωση συστημάτων επικοινωνιών, καλύπτοντας διαμορφώσεις, κανάλια και κωδικοποίηση.

Πλεονεκτήματα:

* Εξειδικευμένες λειτουργίες για ψηφιακές διαμορφώσεις όπως ASK, PSK, FSK, QAM.
* Εργαλεία για την προσομοίωση θορύβου και καναλιών επικοινωνίας.
* Εύκολη χρήση και καλή τεκμηρίωση.

|  |  |
| --- | --- |
| 1. Math   (Math Module - Documentation, n.d.) |  |

Γιατί το επιλέξαμε: Παρέχει βασικές μαθηματικές συναρτήσεις που είναι απαραίτητες για την υλοποίηση των αλγορίθμων.

Πλεονεκτήματα:

* Ενσωματωμένη στη Python, δεν απαιτείται επιπλέον εγκατάσταση.
* Απλή και γρήγορη για βασικούς υπολογισμούς.

**Πώς συνέβαλαν στην επίτευξη του στόχου της εργασίας**

1. NumPy και SciPy: Συνέβαλαν στην αποδοτική επεξεργασία και ανάλυση των ψηφιακών σημάτων, επιτρέποντας τη δημιουργία και διαχείριση μεγάλων δεδομένων με υψηλή απόδοση.
2. scipy.signal: Επέτρεψε τον σχεδιασμό και την ανάλυση ψηφιακών φίλτρων, απαραίτητων για τις εργαστηριακές ασκήσεις που αφορούν στην επεξεργασία σημάτων και στην εφαρμογή φίλτρων.
3. CommPy: Παρείχε εξειδικευμένες λειτουργίες για τις ψηφιακές διαμορφώσεις που εξετάζονται στο μάθημα, διευκολύνοντας την προσομοίωση συστημάτων επικοινωνίας και την ανάλυση της απόδοσής τους.
4. math: Χρησιμοποιήθηκε για βασικές μαθηματικές λειτουργίες που δεν απαιτούσαν τις πιο βαριές δομές δεδομένων του NumPy, συμβάλλοντας στην απλότητα και αποδοτικότητα του κώδικα.

**Συμπέρασμα**

Συνολικά, οι βιβλιοθήκες αυτές συνεργάστηκαν αρμονικά για να προσφέρουν ένα πλήρες σύνολο εργαλείων που κάλυψαν όλες τις απαιτήσεις της προσομοίωσης των ψηφιακών συστημάτων στο πλαίσιο του μαθήματος. Η επιλογή τους βασίστηκε στην ευρεία αποδοχή τους από την επιστημονική κοινότητα, την πλούσια τεκμηρίωσή τους και την ικανότητά τους να ικανοποιήσουν τις συγκεκριμένες ανάγκες της εργασίας.

#### Γραφική αναπαράσταση δεδομένων

|  |  |
| --- | --- |
| Matplotlib  (Matplotlib - Documentation, n.d.) |  |

Γιατί το επιλέξαμε:

* Ευρεία Χρήση και Ωριμότητα: Το Matplotlib είναι μία από τις παλαιότερες και πιο καθιερωμένες βιβλιοθήκες γραφικών στην Python, με μεγάλη κοινότητα χρηστών και προγραμματιστών.
* Πλούσιες Δυνατότητες: Υποστηρίζει μια μεγάλη ποικιλία γραφημάτων, συμπεριλαμβανομένων των 2D και 3D γραφημάτων, χρονικών διαγραμμάτων, φασμάτων, διαγραμμάτων αστερισμών, histograms, stem plots κ.λπ.
* Ευκολία Χρήσης: Παρέχει μια απλή και κατανοητή διεπαφή (PyPlot), που επιτρέπει τη γρήγορη δημιουργία γραφημάτων με σύνταξη παρόμοια με το MATLAB.
* Διαδραστικότητα: Αν και είναι κυρίως σχεδιασμένο για στατικά γραφήματα, μπορεί να υποστηρίξει βασική διαδραστικότητα και animation, τα οποία είναι επαρκή για τις ανάγκες του έργου.
* Συμβατότητα: Ενσωματώνεται άριστα με άλλες βιβλιοθήκες όπως NumPy και SciPy, διευκολύνοντας την απευθείας απεικόνιση δεδομένων από πίνακες και αποτελέσματα υπολογισμών.
* Τεκμηρίωση και Πόροι: Διαθέτει εκτενή τεκμηρίωση και μεγάλο αριθμό παραδειγμάτων, που διευκολύνουν την εκμάθηση και την επίλυση προβλημάτων.

Πλεονεκτήματα:

* Προσαρμοστικότητα: Επιτρέπει την πλήρη προσαρμογή των γραφημάτων, από τα βασικά στοιχεία έως τις πιο λεπτομερείς ρυθμίσεις.
* Σταθερότητα: Ως ώριμη βιβλιοθήκη, είναι σταθερή και αξιόπιστη για παραγωγική χρήση.
* Υποστήριξη 2D και 3D Γραφημάτων: Ανάγκη για απεικόνιση δεδομένων σε τρεις διαστάσεις, όπως επιφάνειες και πλέγματα, μπορεί να καλυφθεί.

**Πώς συνέβαλε στην επίτευξη του στόχου της εργασίας**

* Οπτικοποίηση Σημάτων: Χρησιμοποιήθηκε για την απεικόνιση χρονικών διαγραμμάτων των σημάτων, επιτρέποντας στους χρήστες να δουν την εξέλιξη του σήματος στον χρόνο.
* Ανάλυση Φάσματος: Διευκόλυνε την δημιουργία γραφημάτων φασματικής πυκνότητας ισχύος, βοηθώντας στην κατανόηση των φασματικών χαρακτηριστικών των σημάτων.
* Διαγράμματα Αστερισμών: Επιτρέπει την απεικόνιση των συμβόλων σε διαμορφώσεις όπως QAM και PSK, βοηθώντας στην ανάλυση της απόδοσης των διαμορφώσεων.
* Δυνατότητα Διαδραστικότητας: Σε συνδυασμό με άλλες βιβλιοθήκες (όπως το ipywidgets), μπορεί να υποστηρίξει βασική διαδραστικότητα, επιτρέποντας στους χρήστες να τροποποιούν παραμέτρους και να βλέπουν τις αλλαγές στα γραφήματα σε πραγματικό χρόνο.
* Ενσωμάτωση σε Πλατφόρμες: Μπορεί να χρησιμοποιηθεί σε περιβάλλοντα όπως Jupyter Book, με πολλά Jupyter Notebooks εσωτερικά, διευκολύνοντας την παρουσίαση και την αλληλεπίδραση με τον κώδικα και τα γραφήματα.

**Συμπέρασμα**

Συνοψίζοντας, το Matplotlib καλύπτει πλήρως τις ανάγκες του έργου για γραφική αναπαράσταση δεδομένων, παρέχοντας την ευελιξία και τα εργαλεία που απαιτούνται για την οπτικοποίηση των αποτελεσμάτων των προσομοιώσεων ψηφιακών συστημάτων. Η επιλογή του βασίστηκε στην ευρεία αποδοχή του, την πλούσια τεκμηρίωση και την ικανότητά του να ικανοποιήσει τις συγκεκριμένες απαιτήσεις του μαθήματος.

#### Διαδραστικοί μέθοδοι εισαγωγής παραμέτρων.

|  |  |
| --- | --- |
| 1. IPython   (IPython - Documentation, n.d.) |  |

Γιατί το επιλέξαμε:

* Ενσωμάτωση με Jupyter Notebooks: Το IPython αποτελεί τη βάση για τα Jupyter Notebooks, παρέχοντας ένα ισχυρό διαδραστικό περιβάλλον για την εκτέλεση κώδικα Python.
* Δυνατότητες Διαδραστικότητας: Προσφέρει εντολές και μαγικές λειτουργίες που διευκολύνουν την ανάπτυξη διαδραστικών εφαρμογών.
* Υποστήριξη Πλούσιου Περιεχομένου: Επιτρέπει την ενσωμάτωση πλούσιου περιεχομένου όπως HTML, βίντεο και widgets μέσα στα notebooks.

Πλεονεκτήματα:

* Ευκολία Χρήσης: Παρέχει ένα φιλικό προς τον χρήστη περιβάλλον με δυνατότητες αυτόματης συμπλήρωσης και εύκολης εκτέλεσης κώδικα.
* Επεκτασιμότητα: Μπορεί να επεκταθεί με τη χρήση extensions και μαγικών εντολών.
* Κοινότητα και Τεκμηρίωση: Διαθέτει μεγάλη κοινότητα και πλούσια τεκμηρίωση.

|  |  |
| --- | --- |
| 1. ipywidgets   (ipywidgets - Documentation, n.d.) |  |

Γιατί το επιλέξαμε:

* Ενσωμάτωση με Jupyter Notebooks: Σχεδιασμένο ειδικά για χρήση μέσα στα notebooks, επιτρέπει την εύκολη δημιουργία διαδραστικών widgets.
* Πλούσια Συλλογή Widgets: Παρέχει μια μεγάλη ποικιλία από προκαθορισμένα widgets όπως sliders, text boxes, dropdown menus, buttons κ.λπ.
* Εύκολη Σύνδεση με Κώδικα: Επιτρέπει τη σύνδεση των widgets με συναρτήσεις Python, ώστε οι αλλαγές στις παραμέτρους να ενημερώνουν άμεσα τα αποτελέσματα και τα γραφήματα.

Πλεονεκτήματα:

* Ευκολία Χρήσης: Με απλή σύνταξη, οι χρήστες μπορούν να δημιουργήσουν διαδραστικές διεπαφές μέσα σε λίγες γραμμές κώδικα.
* Διαδραστικότητα σε Πραγματικό Χρόνο: Οι αλλαγές στις παραμέτρους αντικατοπτρίζονται άμεσα στα γραφήματα και τα αποτελέσματα, βελτιώνοντας την κατανόηση των εννοιών.
* Συμβατότητα με Matplotlib: Μπορεί να χρησιμοποιηθεί σε συνδυασμό με το Matplotlib για την ενημέρωση γραφημάτων σε πραγματικό χρόνο.

**Πώς συνέβαλαν στην επίτευξη του στόχου της εργασίας**

* Διαδραστική Επεξεργασία Παραμέτρων: Με τη χρήση των ipywidgets, οι φοιτητές μπορούν να τροποποιούν παραμέτρους των σημάτων και των συστημάτων, όπως το Eb/No, το roll-off factor, το επίπεδο διαμόρφωσης κ.λπ., και να βλέπουν άμεσα τις επιπτώσεις στα αποτελέσματα.
* Οπτικοποίηση Αποτελεσμάτων σε Πραγματικό Χρόνο: Συνδυάζοντας τα widgets με το Matplotlib, τα γραφήματα ενημερώνονται δυναμικά, βοηθώντας τους φοιτητές να κατανοήσουν βαθύτερα τη συμπεριφορά των συστημάτων υπό διαφορετικές συνθήκες.
* Ενίσχυση της Μάθησης: Η δυνατότητα αλληλεπίδρασης με τις παραμέτρους και η άμεση οπτικοποίηση των αποτελεσμάτων ενισχύει την κατανόηση και τη διατήρηση των γνώσεων.
* Ευκολία Ανάπτυξης: Η απλή σύνταξη και η άμεση ενσωμάτωση με το υπάρχον περιβάλλον εργασίας διευκολύνει τους εκπαιδευτές στην ανάπτυξη εκπαιδευτικού υλικού και προσομοιώσεων.
* Συνεργατικότητα: Τα Jupyter Notebooks με ενσωματωμένα widgets μπορούν να διαμοιραστούν εύκολα μεταξύ των φοιτητών, επιτρέποντας την κοινή χρήση και την συνεργατική μάθηση.
* Μείωση Απαιτήσεων Εγκατάστασης: Δεδομένου ότι τα εργαλεία λειτουργούν μέσα στα Jupyter Notebooks, οι φοιτητές δεν χρειάζεται να εγκαταστήσουν πρόσθετο λογισμικό ή να ρυθμίσουν περίπλοκα περιβάλλοντα.

**Συμπέρασμα**

Η επιλογή των IPython και ipywidgets βασίστηκε στην ικανότητά τους να παρέχουν ένα ισχυρό, ευέλικτο και εύχρηστο περιβάλλον για τη διαδραστική εισαγωγή παραμέτρων και την άμεση οπτικοποίηση των αποτελεσμάτων. Με την ενσωμάτωσή τους στο ήδη γνωστό περιβάλλον των Jupyter Notebooks, διευκολύνουν την εκπαιδευτική διαδικασία και ενισχύουν την ενεργή συμμετοχή των φοιτητών στην μάθηση των ψηφιακών συστημάτων επικοινωνίας.

Κεφάλαιο 3

# Μεθοδολογία, Σχεδίαση και Υλοποίηση

## Εισαγωγή

Σε αυτό το κεφάλαιο, αρχικά, παρουσιάζεται η συνολική μεθοδολογία που ακολουθήθηκε για την ολοκληρωμένη ανάπτυξη της διαδικτυακής διαδραστικής εφαρμογής. Στη συνέχεια, παρατίθενται τα τμήματα κώδικα που ενσωματώνουν διαδραστικές λειτουργίες, οι οποίες συνιστούν τη βάση για την ολοκλήρωση του έργου.

## Μέθοδος υλοποίησης του συνολικού project

Η παρούσα εργασία περιγράφει αναλυτικά τα πρώτα στάδια υλοποίησης της διαδικτυακής εφαρμογής, η οποία αναπτύχθηκε σε συνεργασία με τον Λάμπρο Φραγκουλόπουλο. Ειδικότερα, στο παρόν τεύχος παρατίθεται η ανάλυση των βημάτων 1-9 και , τα οποία αποτελούν το θεμέλιο για την ανάπτυξη του συνολικού έργου. Για περαιτέρω πληροφορίες σχετικά με τα επόμενα βήματα και την εξέλιξη του έργου, ο αναγνώστης παραπέμπεται στην αντίστοιχη εργασία του Λάμπρου Φραγκουλόπουλου [Εργασία Λάμπρου].

1. **Αναζήτηση, δοκιμή και τελική επιλογή των κατάλληλων εργαλείων για την ολοκλήρωση του project.**
2. **Δημιουργία αποθετηρίου στο GitHub για τη διαχείριση των αρχείων του έργου.**
3. **Κατασκευή βασικών κομματιών κώδικα python γύρω από την θεωρία και τις εργαστηριακές ασκήσεις του μαθήματος "Ψηφιακές Επικοινωνίες Ι", με διαδραστικές δυνατότητες. (Βάση μας ο κώδικας matlab του μαθήματος)**
4. **Ανάπτυξη και διαμόρφωση του Jupyter Book ως εργαλείο παρουσίασης και εκτέλεσης του εκπαιδευτικού υλικού.**
5. **Τροποποίηση του αρχείου config.yml για την ενσωμάτωση της επιλογής binder, επιτρέποντας την εκτέλεση κώδικα σε πραγματικό χρόνο μέσω ενός εξωτερικού kernel.**
6. **Δημιουργία του αρχείου requirements.txt, περιλαμβάνοντας όλες τις απαιτούμενες βιβλιοθήκες Python για την εξασφάλιση της λειτουργικότητας του kernel μέσω του MyBinder.**
7. **Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για τη δημιουργία των αρχείων της ιστοσελίδας.**
8. **Χρήση του GitHub Pages για την ανάπτυξη και φιλοξενία της διαδικτυακής εφαρμογής.**
9. **Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για την δημοσίευση της ιστοσελίδας στο GitHub Page που δημιουργήσαμε.**
10. Αναδιάρθρωση της δομής του Jupyter Book μέσω αλλαγών στο αρχείο toc.yml, ώστε να ευθυγραμμίζεται με τη διάρθρωση των εργαστηριακών ασκήσεων του μαθήματος "Ψηφιακές Επικοινωνίες Ι".
11. Μετάφραση του περιεχομένου των εργαστηριακών ασκήσεων στα Αγγλικά.
12. Ενσωμάτωση του μεταφρασμένου περιεχομένου στα Jupyter Notebooks που δημιουργήθηκαν. Ένα για κάθε άσκηση.
13. Εντοπισμός των σημείων που απαιτούν διαδραστικότητα και ενσωμάτωση από τον υπάρχον κώδικα, ή και εκ νέου σχεδιασμός, αντίστοιχων λειτουργιών.
14. Μετατροπή του υπάρχοντος βοηθητικού κώδικα από MATLAB σε Python, με χρήση των βιβλιοθηκών ψηφιακής επικοινωνίας και γραφικής απεικόνισης δεδομένων. (Με την βοήθεια των κομματιών κώδικα από το βήμα 3)
15. Προσθήκη διαδραστικών λειτουργιών στον κώδικα Python μέσω των κατάλληλων βιβλιοθηκών. (Με την βοήθεια των κομματιών κώδικα από το βήμα 3)
16. Ανανέωση του αρχείου requirements.txt, ώστε να περιλαμβάνει τις τελικές βιβλιοθήκες python που χρησιμοποιήθηκαν.
17. Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για τη δημιουργία των αρχείων της ιστοσελίδας.
18. Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για την δημοσίευση της ιστοσελίδας στο GitHub Page που δημιουργήσαμε.

## Υλοποίηση διαδραστικών στοιχείων

### Chapter 1: Digital Signal Processing in Telecommunications

#### Δημιουργία & Οπτικοποίηση σημάτων:

Παραγωγή σημάτων για την προσομοίωση συστημάτων ψηφιακών επικοινωνιών και δημιουργία γραφημάτων για να απεικονιστούν τα χρονικά και φασματικά διαγράμματα των σημάτων.

**Παράδειγμα:**

**[Κώδικας – Chapter 1 – Signal Creation & Visualization](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/main/theory_components/Chapter%201/signal_creation_and_visualization.ipynb)**

import numpy as np

import matplotlib.pyplot as plt

# Create a Sinusoidal Signal

Fs = 2000                  # Sampling frequency in Hz

Ts = 1 / Fs                # Sampling period in seconds

T = 0.1                    # Signal duration in seconds

t = np.arange(0, T, Ts)    # Time vector for signal

A = 1                      # Signal amplitude

x = A \* np.sin(2 \* np.pi \* 100 \* t)  # Generate sinusoidal signal

L = len(x)                 # Length of the signal

N = 1 \* L                  # Length of Fourier Transform

Fo = Fs / N                # Frequency resolution

Fx = np.fft.fft(x, N)      # DFT of the signal

freq = np.arange(0, N) \* Fo  # Frequency vector

# Plot the sinusoidal signal in time domain

plt.figure(figsize=(12, 10))

# Time-domain plot

plt.subplot(2, 1, 1)

plt.plot(t, x)

plt.title('Sinusoidal Signal in Time Domain')

plt.xlabel('Time (s)')

plt.ylabel('Amplitude')

# Frequency-domain (FFT) plot

plt.subplot(2, 1, 2)

plt.plot(freq, np.abs(Fx))  # Plot only the positive frequencies

plt.title('FFT of Sinusoidal Signal')

plt.xlabel('Frequency (Hz)')

plt.ylabel('Magnitude')

# Display the plots

plt.tight\_layout()

plt.show()

* **Προσθήκη θορύβου:**

Δυνατότητα προσθήκης θορύβου στο σήμα για παρακολούθηση των επιδράσεών του στην απόδοση της επικοινωνίας.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 – Add noise**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/main/theory_components/Chapter%201/add_noise.ipynb)

import numpy as np

import matplotlib.pyplot as plt

# Create a Sinusoidal Signal

Fs = 2000                  # Sampling frequency in Hz

Ts = 1 / Fs                # Sampling period in seconds

T = 0.1                    # Signal duration in seconds

t = np.arange(0, T, Ts)    # Time vector for signal

A = 1                      # Signal amplitude

x = A \* np.sin(2 \* np.pi \* 100 \* t)  # Generate sinusoidal signal

L = len(x)

# Add noise to x

rand\_n = np.random.randn(len(x))

s = x + rand\_n

N = 1 \* L                  # Length of Fourier Transform

Fo = Fs / N                # Frequency resolution

Fx = np.fft.fft(s, N)      # DFT of the signal

freq = np.arange(0, N) \* Fo  # Frequency vector

# Plot the sinusoidal signal in time domain

plt.figure(figsize=(12, 10))

# Time-domain plot

plt.subplot(2, 1, 1)

plt.plot(t, s)

plt.title('Sinusoidal Signal in Time Domain')

plt.xlabel('Time (s)')

plt.ylabel('Amplitude')

# Frequency-domain (FFT) plot

plt.subplot(2, 1, 2)

plt.plot(freq, np.abs(Fx))  # Plot only the positive frequencies

plt.title('FFT of Sinusoidal Signal')

plt.xlabel('Frequency (Hz)')

plt.ylabel('Magnitude')

# Display the plots

plt.tight\_layout()

plt.show()

#### Διαδραστική επεξεργασία μεταβλητών σήματος:

Με τη χρήση διαδραστικών εργαλίων, να μπορούν οι χρήστες να τροποποιούν τις παραμέτρους του σήματος και να παρατηρούν σε πραγματικό χρόνο τις αλλαγές.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 – Interactive Signal**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/main/theory_components/Chapter%201/interactive_signal.ipynb)

import numpy as np

import matplotlib.pyplot as plt

from ipywidgets import interact, FloatSlider

import ipywidgets as widgets

# Αρχικές παράμετροι σήματος

def signal\_generator(amplitude, frequency, phase, time\_duration):

    # Χρονικό διάστημα

    t = np.linspace(0, time\_duration, 1000)

    # Γεννήτρια σήματος (ημιτονοειδής)

    signal = amplitude \* np.sin(2 \* np.pi \* frequency \* t + phase)

    # Γράφημα σήματος

    plt.figure(figsize=(10, 5))

    plt.plot(t, signal, label=f'Signal (A={amplitude}, f={frequency}, φ={phase})')

    plt.title('Signal in Time Domain')

    plt.xlabel('Time (s)')

    plt.ylabel('Amplitude')

    plt.grid(True)

    plt.legend()

    plt.show()

# Δημιουργία διαδραστικών widgets

amplitude\_slider = FloatSlider(min=0, max=5, step=0.1, value=1, description='Amplitude')

frequency\_slider = FloatSlider(min=0.1, max=10, step=0.1, value=1, description='Frequency (Hz)')

phase\_slider = FloatSlider(min=0, max=2\*np.pi, step=0.1, value=0, description='Phase (rad)')

time\_slider = FloatSlider(min=0.5, max=5, step=0.1, value=1, description='Time Duration (s)')

# Διαδραστική εκτέλεση

interact(signal\_generator, amplitude=amplitude\_slider, frequency=frequency\_slider, phase=phase\_slider, time\_duration=time\_slider)

#### Ανάλυση Φάσματος:

Δημιουργία γραφημάτων φασματικής ανάλυσης.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 - Spectrum Analysis**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/main/theory_components/Chapter%201/spectrum_analysis.ipynb)

#### Απεικόνιση Φίλτρου (Filter Visualization):

Δυνατότητα επιλογής διαφορετικών φίλτρων και οπτική αναπαράσταση της απόκρισης τους με μορφή stem plot.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 - Filter Visualization**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/filter_visualization.ipynb)

#### Ανάλυση Συχνοτήτων Φίλτρων (Filter Frequency Response):

Διαδραστική απεικόνιση της φασματικής απόκρισης φίλτρων για διαφορετικά μήκη φίλτρων και παραθύρων.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 - Filter Frequency Response**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/filter_frequency_response.ipynb)

#### Ανάλυση Equiripple Φίλτρων:

Δυνατότητα επιλογής equiripple φίλτρων διαφορετικών μηκών και σύγκριση της φασματικής τους απόκρισης.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 - Equiripple Filter Frequency Response**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/equiripple_filter_frequency_response.ipynb)

#### Εφαρμογή Χαμηλοπερατού Φίλτρου (Low Pass Filter Application):

Εφαρμογή φίλτρων χαμηλής διέλευσης και απεικόνιση της απόκρισης συχνοτήτων του φιλτραρισμένου σήματος.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 - Low Pass Filter Application**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/low_pass_filter_application.ipynb)

#### Φασματική Πυκνότητα Ισχύος (Power Spectral Density):

Διαδραστική σύγκριση φίλτρων equiripple και ανάλυση της φασματικής πυκνότητας ισχύος των φιλτραρισμένων σημάτων.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 - Power Spectral Density**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/power_spectral_density.ipynb)

#### Ανάλυση Απόκρισης Bandpass Φίλτρων:

Εφαρμογή φίλτρων bandpass και οπτική απεικόνιση της απόκρισης τους τόσο στην συχνότητα όσο και στην φασματική πυκνότητα του φιλτραρισμένου σήματος.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 - Ανάλυση Απόκρισης Bandpass Φίλτρων**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/bandpass_filter_response.ipynb)

#### Εφαρμογή Parks-McClellan Φίλτρου:

Σχεδιασμός φίλτρου bandpass με τη μέθοδο Parks-McClellan για επίτευξη συγκεκριμένης εξασθένισης στα stop bands.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 - Εφαρμογή Parks-McClellan Φίλτρου**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/parks-McClellan_filter_response.ipynb)

#### Εφαρμογή Φίλτρου με 2 passbands:

Σχεδιασμός και υλοποίηση φίλτρου με δύο passbands στο MATLAB, με απεικόνιση της φασματικής του απόκρισης.

**Παράδειγμα:**

[**Κώδικας – Chapter 1 - Εφαρμογή Φίλτρου με 2 passbands**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/2_bandpass_filter.ipynb)

### Chapter 3: Optimal digital detection – Matched filters

#### Δημιουργία τυχαίων σημάτων:

Δημιουργία τυχαίων σημάτων και παρουσίαση του ιστογράμματός τους.

**Παράδειγμα:**

[**Κώδικας – Chapter 3 - Δημιουργία τυχαίων σημάτων**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%203/random_signal_histogram.ipynb)

#### Εφαρμογή θορύβου:

Επίδραση του θορύβου στο σήμα με τη ρύθμιση του λόγου Eb/No μέσω ενός διαδραστικού slider και τον υπολογισμό του ιστόγραμματος για διάφορες τιμές.

**Παράδειγμα:**

[**Κώδικας – Chapter 3 - Εφαρμογή θορύβου**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%203/random_noisy_signal_histogram.ipynb)

#### ASK Bit Error Rate (BER) Visualization:

Η διαδραστική δυνατότητα σύγκρισης διαμορφώσεων Μ-ASK με επιλογές για πειραματική και θεωρητική τιμή του BER. Να μπορούν οι χρήστες να επιλέξουν μεταξύ διαφορετικών διαμορφώσεων, δειγμάτων ανά σύμβολο, και τύπου matched filter για να δουν τα αποτελέσματα σε πραγματικό χρόνο.

**Παράδειγμα:**

[**Κώδικας – Chapter 3 - ASK Bit Error Rate (BER) Visualization**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%203/ASK_BER_tool.ipynb)

#### Ανάλυση με φίλτρα:

Οπτικοποιήσεις απόκρισης φίλτρων πάνω σε ορθογώνιο και ημιτονοειδές σήμα.

**Παράδειγμα:**

[**Κώδικας – Chapter 3 - Ανάλυση με φίλτρα**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%203/filter_analysis.ipynb)

### Chapter 4: Spectral Characteristics of Digital Waveforms & Nyquist signaling

#### Ρυθμιζόμενες παράμετροι:

Εισαγωγή διαδραστικού τρόπου ρύθμισης παραμέτρων μήκους bitstream, roll-off factor, nsamp (δείγματα ανά σύμβολο), group delay, και σειρά του φίλτρου.

**Παράδειγμα:**

[**Κώδικας – Chapter 4 - Ρυθμιζόμενες παράμετροι**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%204/interactive_filter_response.ipynb)

#### Οπτικοποίηση σημάτων και φίλτρων:

Δημιουργία γραφημάτων για την παρουσίαση του φίλτρου, του φιλτραρισμένου σήματος, τη σύγκριση μεταξύ των αρχικών και φιλτραρισμένων σημάτων και την ανάλυση της φασματικής πυκνότητας ισχύος του ληφθέντος σήματος.

**Παράδειγμα:**

[**Κώδικας – Chapter 4 - Οπτικοποίηση σημάτων και φίλτρων**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%204/interactive_filter_response.ipynb)

#### BER καμπύλες:

Με δυνατότητα επιλογής επιπέδου M-ASK, εμφάνισης των θεωρητικών και πειραματικών καμπυλών σε σχέση με το Eb/N0 και σύγκριση μεταξύ κωδικοποίησης Gray και Natural.

**Παράδειγμα:**

[**Κώδικας – Chapter 4 - BER καμπύλες**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%204/ASK_BER_tool(with%20gray).ipynb)

### Chapter 5: Digital Modulation QAM and PSK

#### QAM και PSK Constellations:

Διαδραστικά διαγράμματα αστερισμών για L-QAM και M-PSK. Οι χρήστες να μπορούν να επιλέξουν το επίπεδο διαμόρφωσης και να δουν τις αντίστοιχες θέσεις των συμβόλων στον αστερισμό.

**Παράδειγμα:**

[**Κώδικας – Chapter 5 - QAM Constellation**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/QAM_constellation.ipynb)

[**Κώδικας – Chapter 5 - PSK Constellation**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/PSK_constellation.ipynb)

#### QAM και PSK Καμπύλες BER:

Δημιουργία γραφημάτων για τις καμπύλες BER τόσο για QAM όσο και για PSK διαμορφώσεις, όπου οι χρήστες μπορούν να συγκρίνουν την θεωρητική και πειραματική καμπύλη και να ρυθμίσουν παραμέτρους όπως το roll-off, τις συχνότητες f1 και f2, και το bitrate για την ανάλυση της πιθανότητας σφάλματος.

**Παράδειγμα:**

[**Κώδικας – Chapter 5 - QAM BER**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/QAM_BER_tool.ipynb)

[**Κώδικας – Chapter 5 - PSK BER**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/PSK_BER_tool.ipynb)

#### Διαδραστική Φασματική Πυκνότητα Ισχύος:

Δυνατότητα οπτικοποίησης της φασματικής πυκνότητας ισχύος για διάφορες διαμορφώσεις QAM και συχνοτήτων f1 και f2.

**Παράδειγμα:**

[**Κώδικας – Chapter 5 - Διαδραστική Φασματική Πυκνότητα Ισχύος**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/power_welch_spectral_density.ipynb)

#### Διαδραστικός Υπολογισμός Παραμέτρων:

Δυνατότητα υπολογισμού παραμέτρων, με βάση τις παραμέτρους (R, M και roll-off) που επιλέγει ο χρήστης, όπως ο ρυθμός.

**Παράδειγμα:**

[**Κώδικας – Chapter 5 - Διαδραστικός Υπολογισμός Παραμέτρων**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/interactive_parameters.ipynb)

### Chapter 6: Digital Modulation FSK and MSK

#### Αριθμός σφαλμάτων για Eb/No:

Γράφημα στο οποίο ο χρήστης ρυθμίζει ένα εύρος του Eb/No, και στη συνέχεια εμφανίζεται η καμπύλη που απεικονίζει τον αριθμό των σφαλμάτων σε σχέση με το Eb/No για συστήματα FSK.

**Παράδειγμα:**

[**Κώδικας – Chapter 6 - Αριθμός σφαλμάτων για Eb/No**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%206/error_number_for_EbNo.ipynb)

#### FSK Bit Error Rate:

Διαδραστική σύγκριση των καμπυλών BER για coherent και non coherent συστήματα FSK. Οι χρήστες μπορούν να δουν τις καμπύλες που αντιπροσωπεύουν την θεωρητική και πειραματική απόδοση.

**Παράδειγμα:**

[**Κώδικας – Chapter 6 - FSK Bit Error Rate**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%206/FSK_BER_tool.ipynb)

#### Ανάλυση Φάσματος coherent και non coherent FSK:

Διαδραστικό γράφημα ανάλυσης της φασματικής πυκνότητας ισχύος τόσο για coherent όσο και για non coherent FSK συστήματα, όπου οι χρήστες μπορούν να προσαρμόσουν τις παραμέτρους όπως ο αριθμός συμβόλων και το Eb/No.

**Παράδειγμα:**

[**Κώδικας – Chapter 6 - Analysis coherent και non coherent FSK**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%206/FSK_frequency_analysis.ipynb)

#### MSK Bit Error Rate:

Διαδραστικό γράφημα για σύγκριση μεταξύ των καμπυλών BER με και χωρίς precoding για τη διαμόρφωση MSK. Οι χρήστες μπορούν να δουν τις θεωρητικές και πειραματικές καμπύλες, συγκρίνοντας τις επιδόσεις για διαφορετικές τιμές Eb/No.

**Παράδειγμα:**

[**Κώδικας – Chapter 6 - MSK Bit Error Rate**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%206/MSK_BER_tool.ipynb)

## Οδηγίες για τα βήματα 5-9

Λεπτομερείς οδηγίες για κάθε βήμα μπορούν να βρεθούν στο παράρτημα. Για γρήγορη περιήγηση μπορείτε να πατήσετε τους παρακάτω συνδέσμους:

* Τροποποίηση του αρχείου config.yml για την ενσωμάτωση της επιλογής binder, επιτρέποντας την εκτέλεση κώδικα σε πραγματικό χρόνο μέσω ενός εξωτερικού kernel.

**Οδηγίες - Βήμα 5**

* Δημιουργία του αρχείου requirements.txt, περιλαμβάνοντας όλες τις απαιτούμενες βιβλιοθήκες Python για την εξασφάλιση της λειτουργικότητας του kernel μέσω του MyBinder.

**Οδηγίες - Βήμα 6**

* Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για τη δημιουργία των αρχείων της ιστοσελίδας.

**Οδηγίες - Βήμα 7**

* Χρήση του GitHub Pages για την ανάπτυξη και φιλοξενία της διαδικτυακής εφαρμογής.

**Οδηγίες - Βήμα 8**

* Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για την δημοσίευση της ιστοσελίδας στο GitHub Page που δημιουργήσαμε.

**Οδηγίες - Βήμα 9**

Κεφάλαιο 4

# Αποτελέσματα

## Εισαγωγή

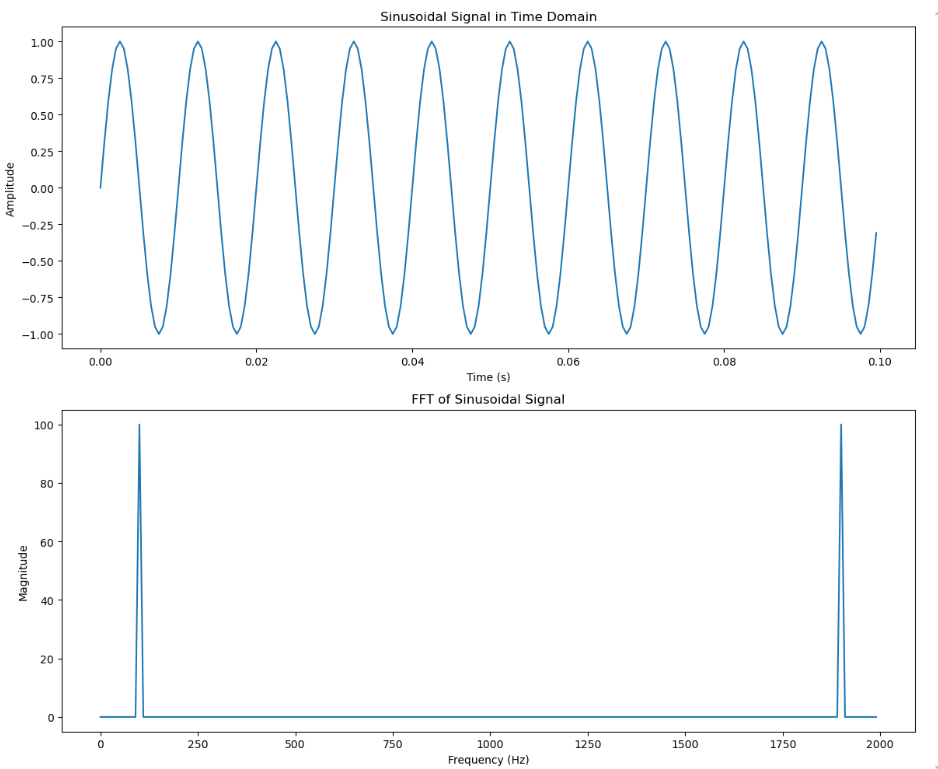
Σε αυτό το κεφάλαιο παρουσιάζονται τα αποτελέσματα των προσομοιώσεων που προέκυψαν από την ανάπτυξη των προηγούμενων τμημάτων κώδικα. Στη συνέχεια, αναλύονται οι διαφορές μεταξύ των προσομοιωμένων αποτελεσμάτων και των θεωρητικά αναμενόμενων γραφικών παραστάσεων, καθώς και οι χρόνοι εκτέλεσης. Επιπλέον, καταγράφονται οι προκλήσεις που προέκυψαν κατά τη διαδικασία ανάπτυξης των προσομοιώσεων και η επίδρασή τους στην ακρίβεια και την απόδοση των αποτελεσμάτων.

## Αποτελέσματα Προσομοιώσεων

### Chapter 1: Digital Signal Processing in Telecommunications

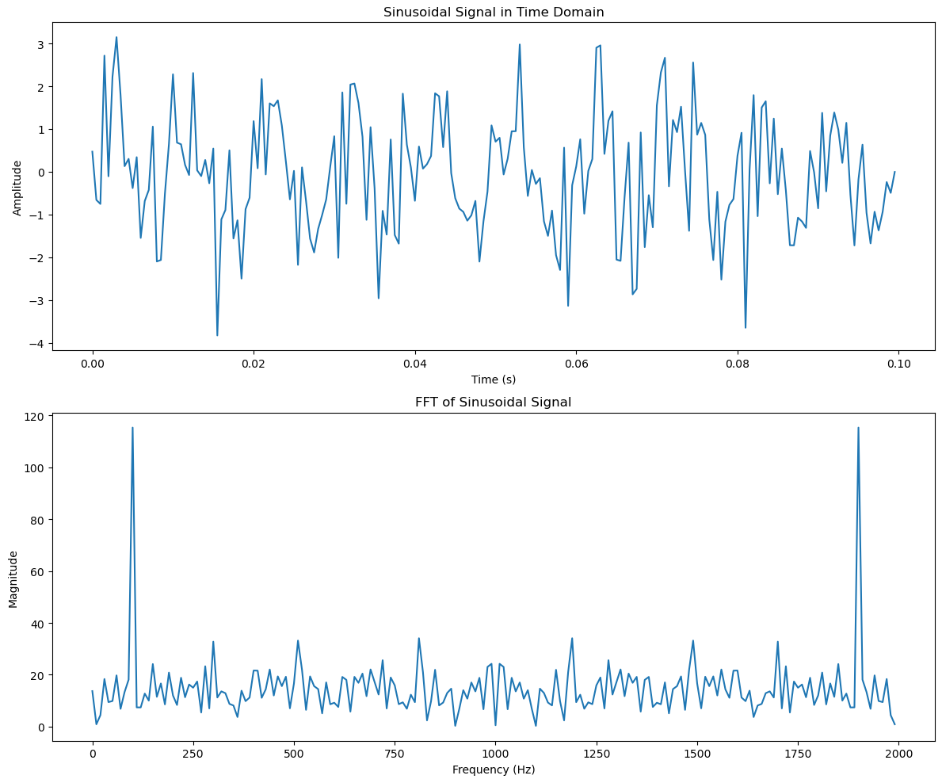
#### Δημιουργία & Οπτικοποίηση σημάτων:

[**Κώδικας – Chapter 1 – Signal Creation & Visualization**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/main/theory_components/Chapter%201/signal_creation_and_visualization.ipynb)



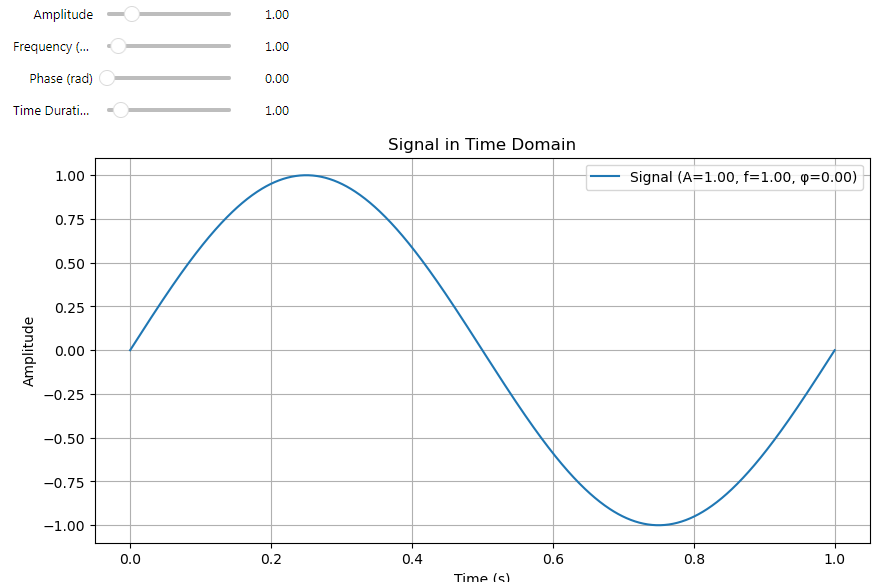
#### Προσθήκη θορύβου:

[**Κώδικας – Chapter 1 – Add noise**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/main/theory_components/Chapter%201/add_noise.ipynb)



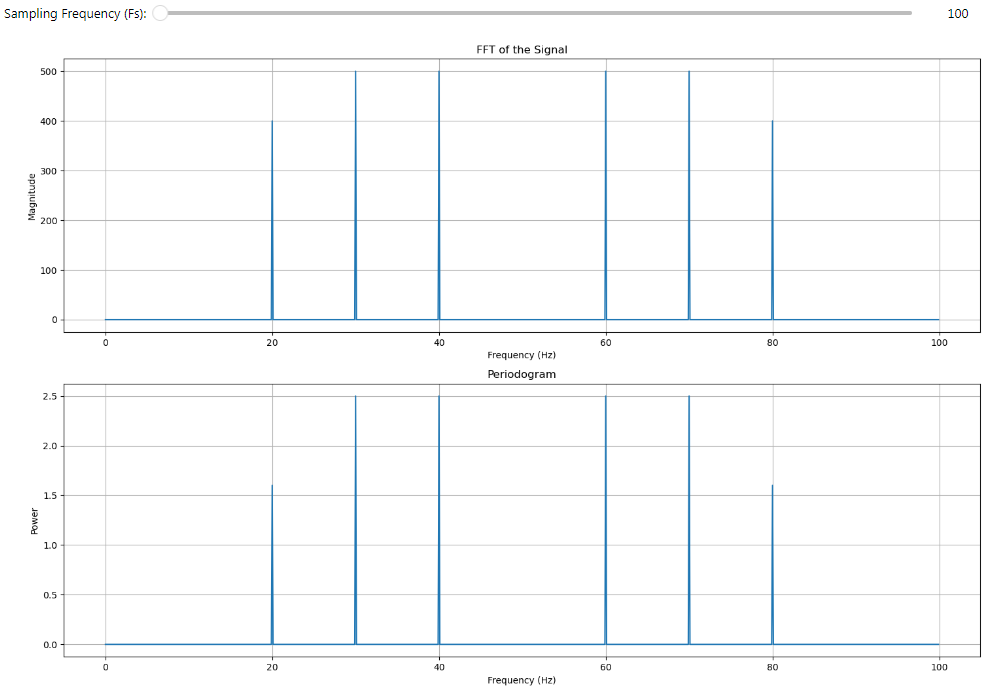
#### Διαδραστική επεξεργασία μεταβλητών σήματος:

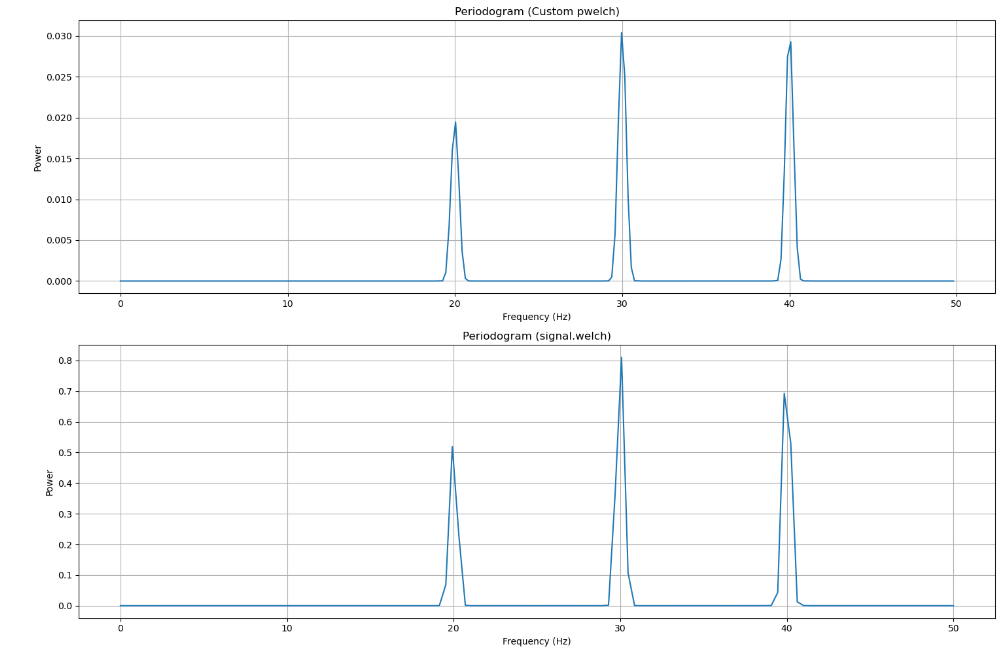
[**Κώδικας – Chapter 1 – Interactive Signal**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/main/theory_components/Chapter%201/interactive_signal.ipynb)



#### Ανάλυση Φάσματος:

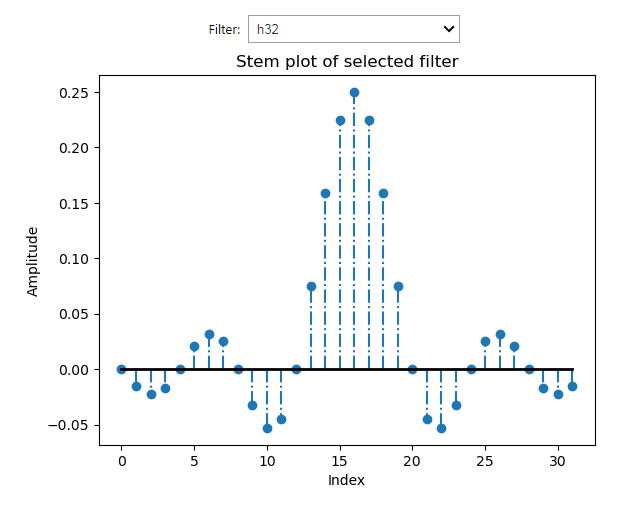
[**Κώδικας – Chapter 1 - Spectrum Analysis**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/main/theory_components/Chapter%201/spectrum_analysis.ipynb)





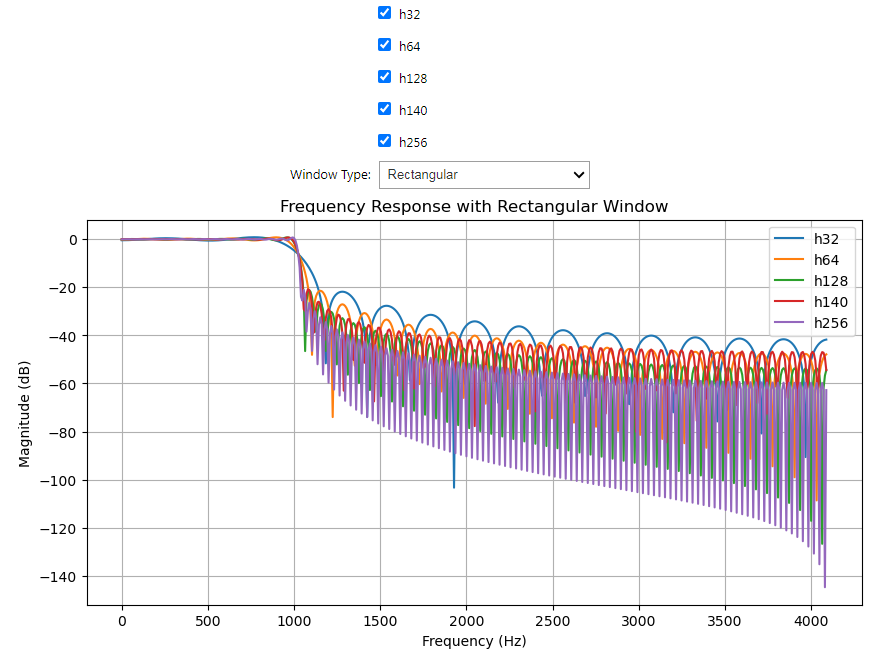
#### Απεικόνιση Φίλτρου (Filter Visualization):

[**Κώδικας – Chapter 1 - Filter Visualization**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/filter_visualization.ipynb)



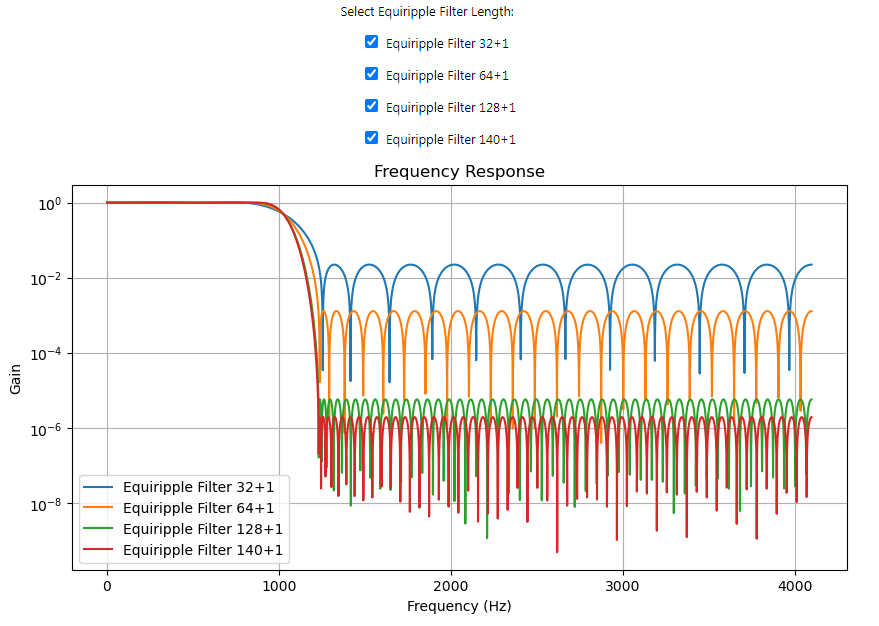
#### Ανάλυση Συχνοτήτων Φίλτρων (Filter Frequency Response):

[**Κώδικας – Chapter 1 - Filter Frequency Response**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/filter_frequency_response.ipynb)



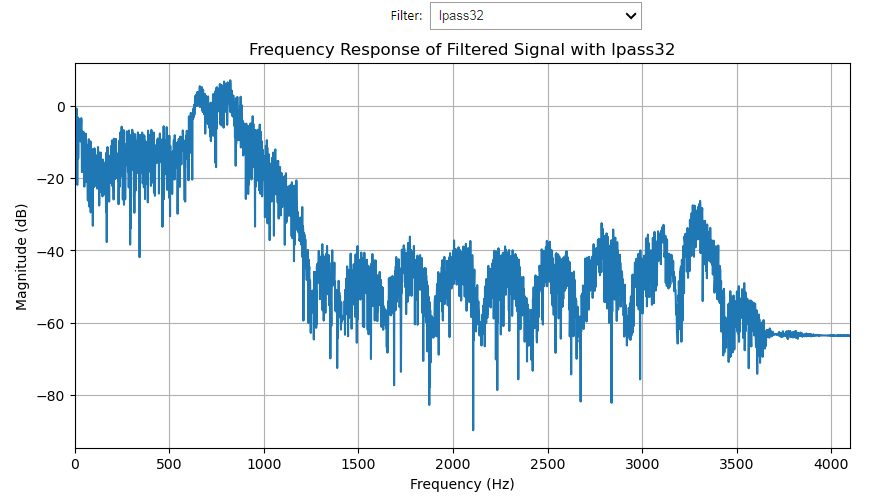
#### Ανάλυση Equiripple Φίλτρων:

[**Κώδικας – Chapter 1 - Equiripple Filter Frequency Response**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/equiripple_filter_frequency_response.ipynb)



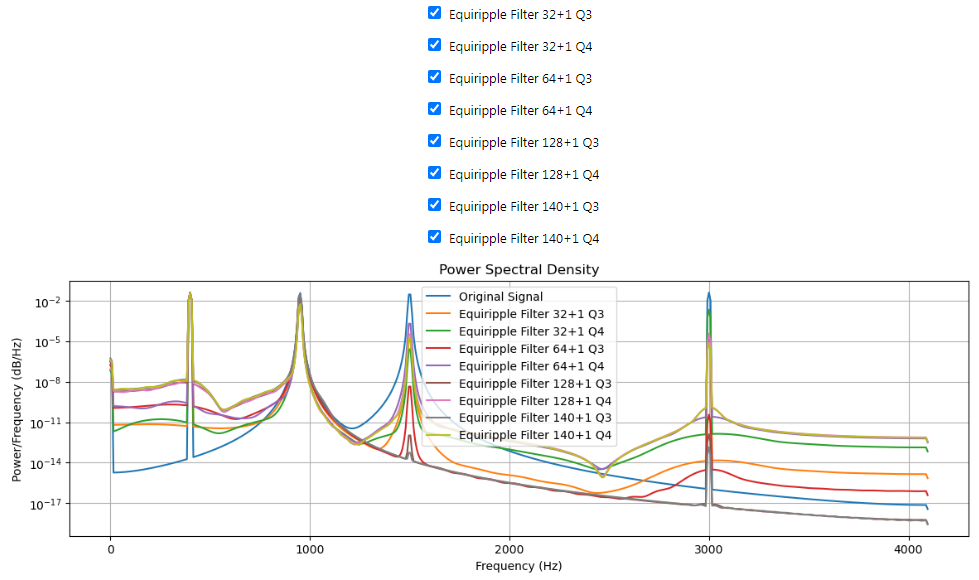
#### Εφαρμογή Χαμηλοπερατού Φίλτρου (Low Pass Filter Application):

[**Κώδικας – Chapter 1 - Low Pass Filter Application**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/low_pass_filter_application.ipynb)



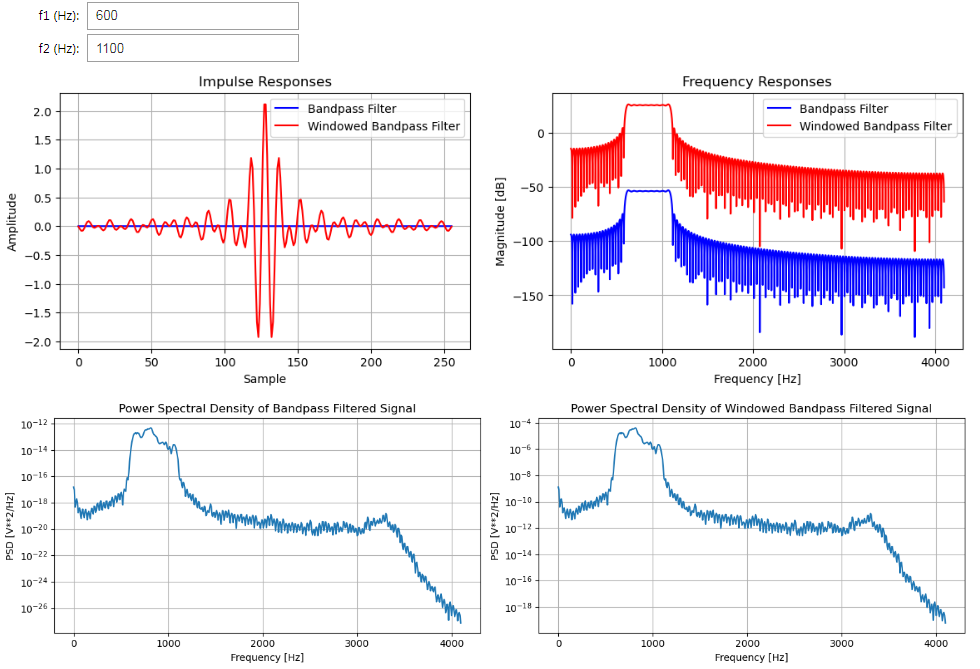
#### Φασματική Πυκνότητα Ισχύος (Power Spectral Density):

[**Κώδικας – Chapter 1 - Power Spectral Density**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/power_spectral_density.ipynb)



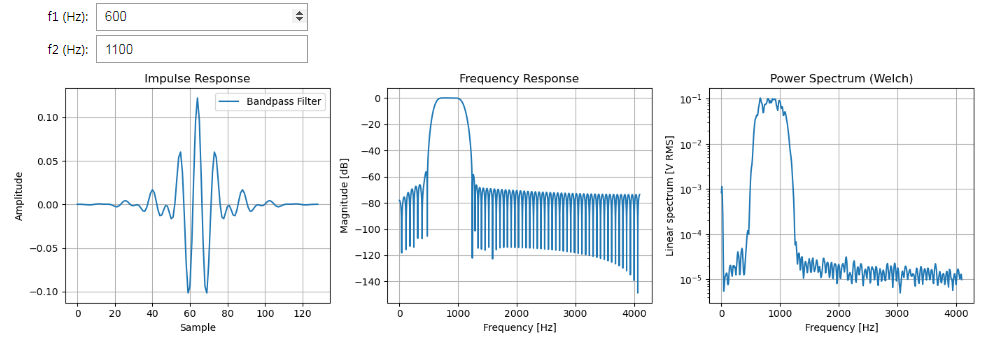
#### Ανάλυση Απόκρισης Bandpass Φίλτρων:

[**Κώδικας – Chapter 1 - Ανάλυση Απόκρισης Bandpass Φίλτρων**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/bandpass_filter_response.ipynb)



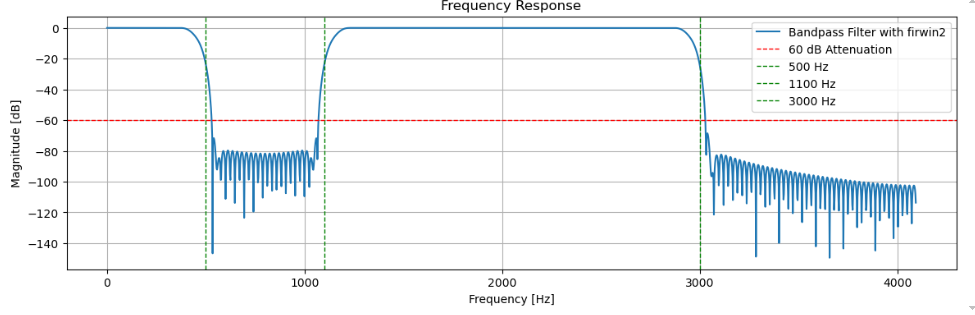
#### Εφαρμογή Parks-McClellan Φίλτρου:

[**Κώδικας – Chapter 1 - Εφαρμογή Parks-McClellan Φίλτρου**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/parks-McClellan_filter_response.ipynb)



#### Εφαρμογή Φίλτρου με 2 passbands:

[**Κώδικας – Chapter 1 - Εφαρμογή Φίλτρου με 2 passbands**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%201/2_bandpass_filter.ipynb)



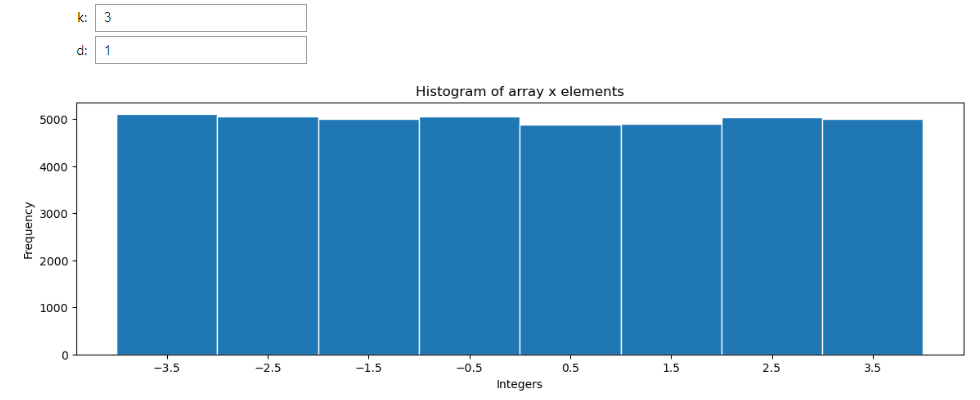
#### Συμπεράσματα

dfgaf

### Chapter 3: Optimal digital detection – Matched filters

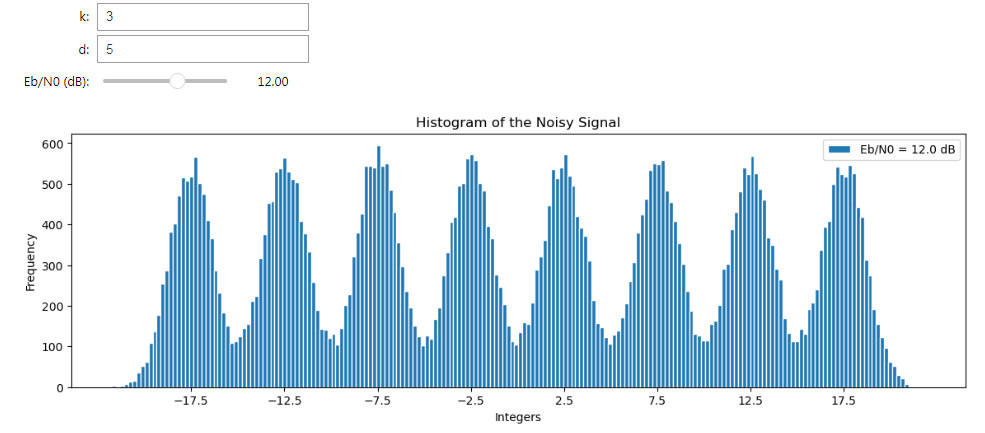
#### Δημιουργία τυχαίων σημάτων:

[**Κώδικας – Chapter 3 - Δημιουργία τυχαίων σημάτων**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%203/random_signal_histogram.ipynb)



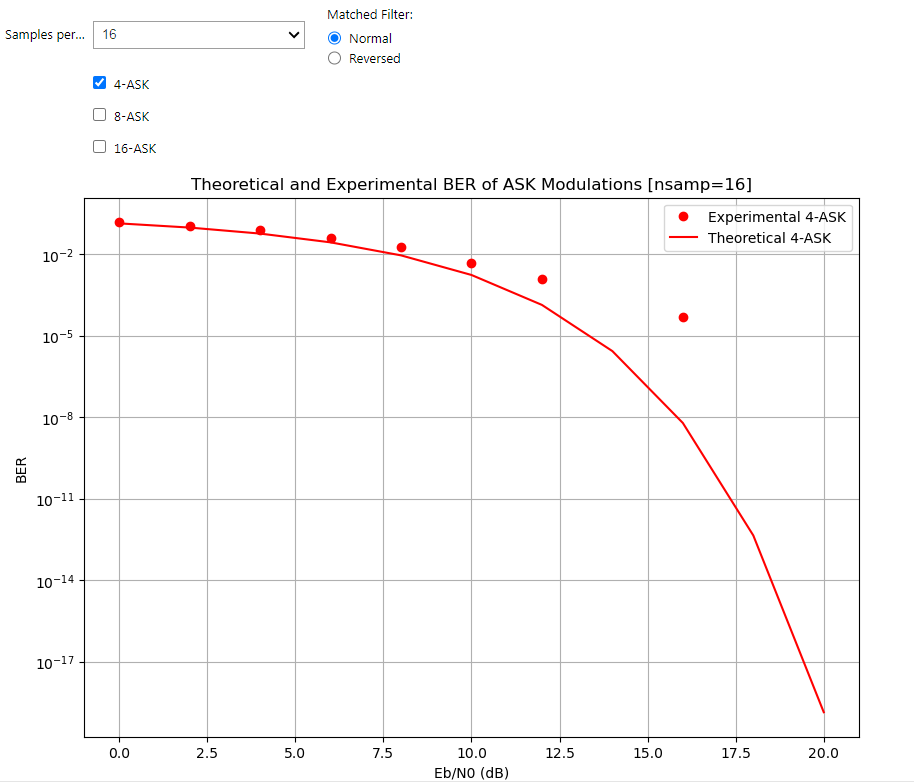
#### Εφαρμογή θορύβου:

[**Κώδικας – Chapter 3 - Εφαρμογή θορύβου**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%203/random_noisy_signal_histogram.ipynb)



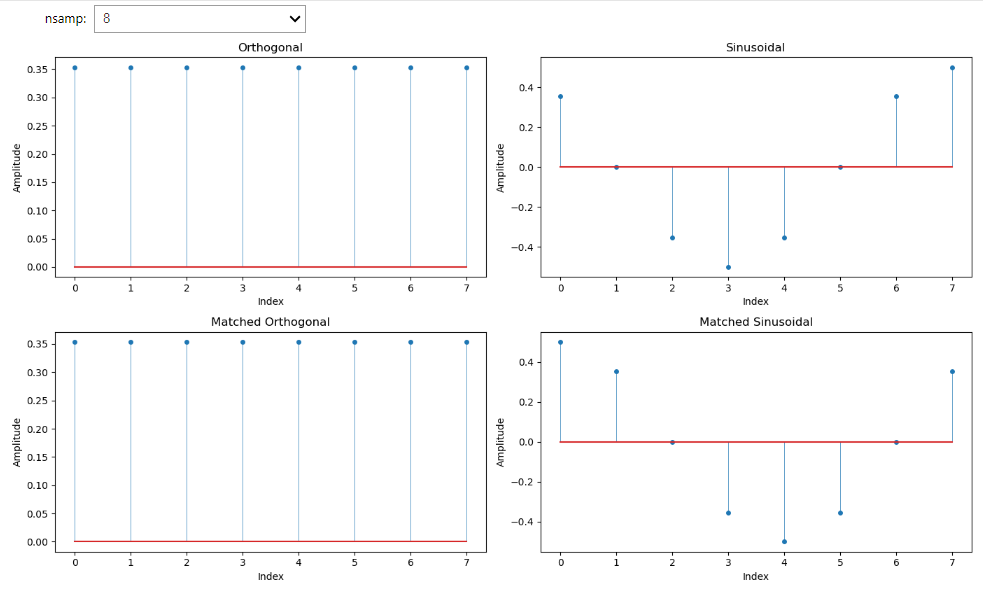
#### ASK Bit Error Rate (BER) Visualization:

[**Κώδικας – Chapter 3 - ASK Bit Error Rate (BER) Visualization**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%203/ASK_BER_tool.ipynb)



#### Ανάλυση με φίλτρα:

[**Κώδικας – Chapter 3 - Ανάλυση με φίλτρα**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%203/filter_analysis.ipynb)



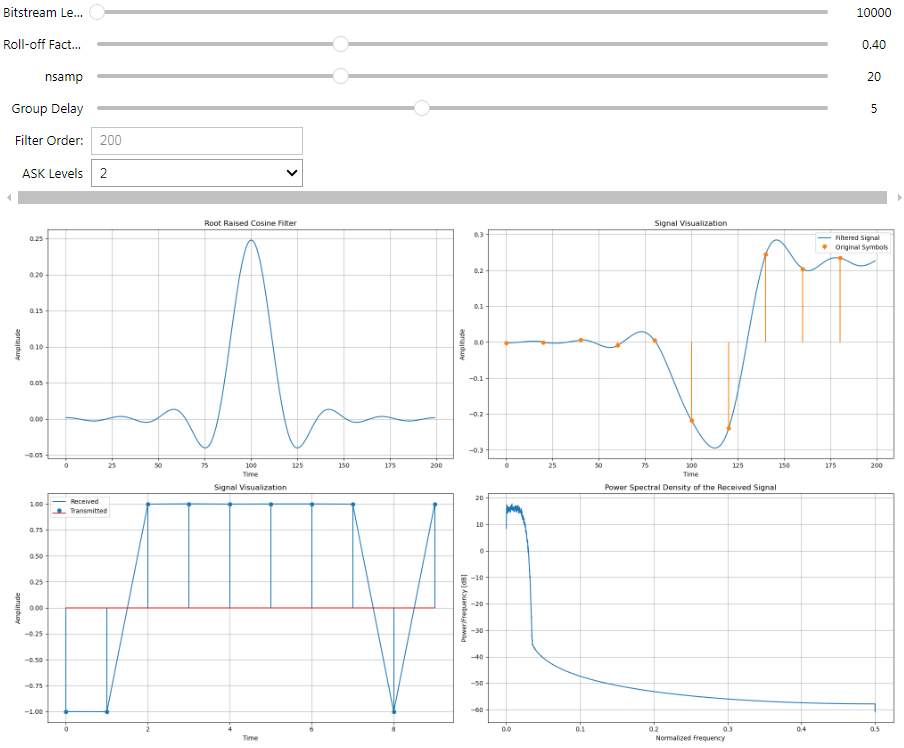
#### Συμπεράσματα

dfgaf

### Chapter 4: Spectral Characteristics of Digital Waveforms & Nyquist signaling

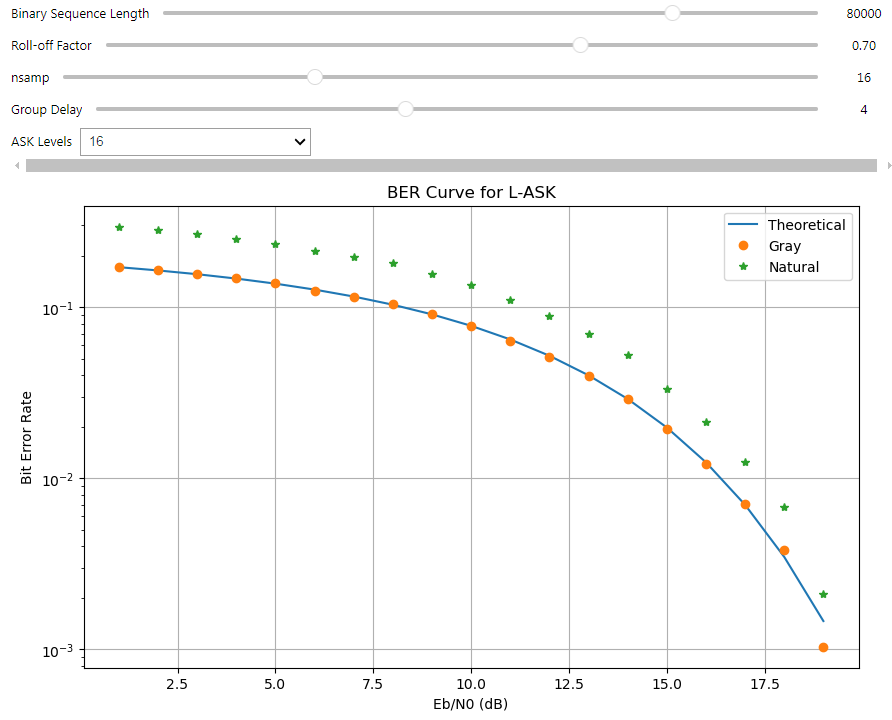
#### Ρυθμιζόμενες παράμετροι & Οπτικοποίηση σημάτων και φίλτρων:

[**Κώδικας – Chapter 4 – Interactive Filter Response**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%204/interactive_filter_response.ipynb)



#### BER καμπύλες:

[**Κώδικας – Chapter 4 - BER καμπύλες**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%204/ASK_BER_tool(with%20gray).ipynb)



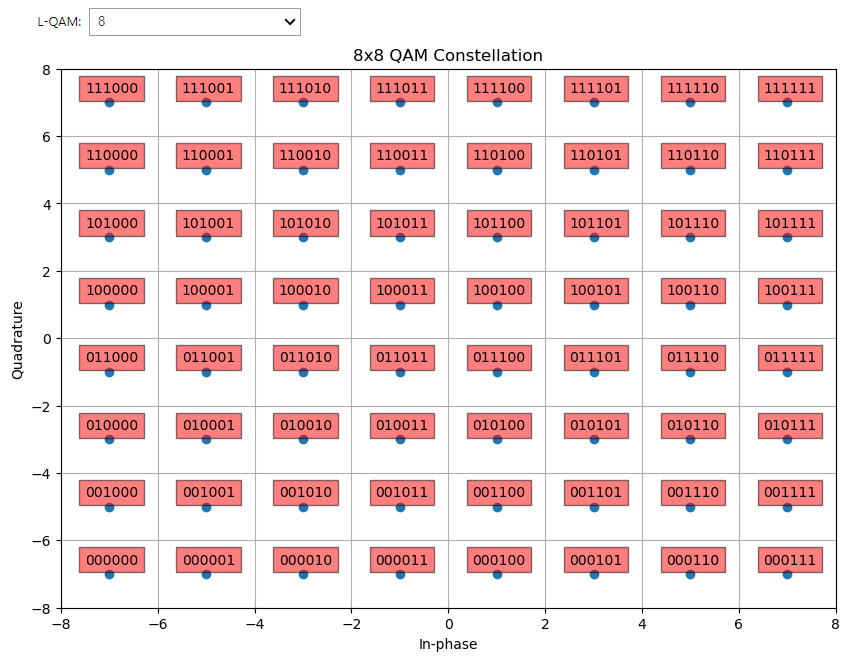
#### Συμπεράσματα

fasdfasdf

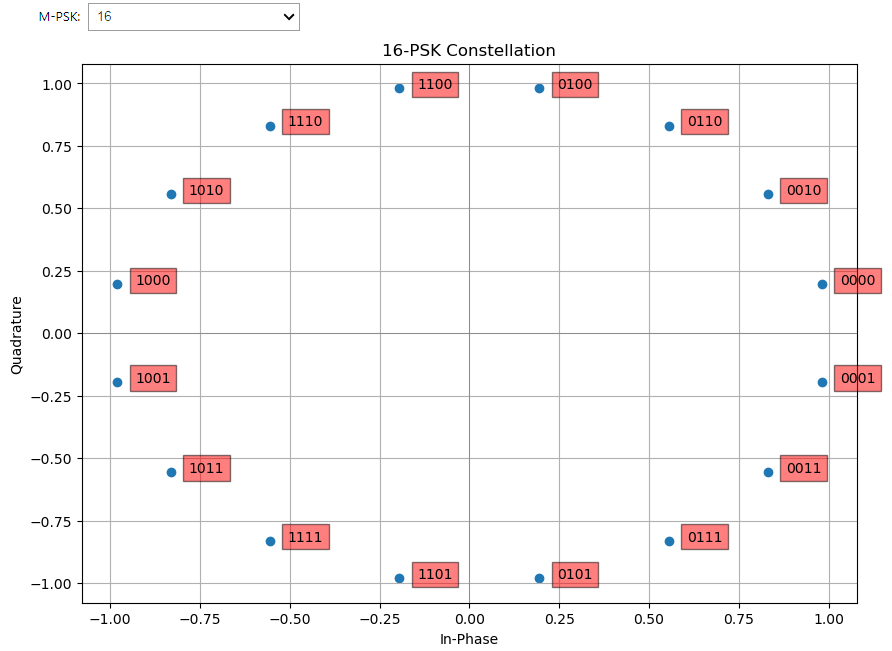
### Chapter 5: Digital Modulation QAM and PSK

#### QAM και PSK Constellations:

[**Κώδικας – Chapter 5 - QAM Constellation**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/QAM_constellation.ipynb)

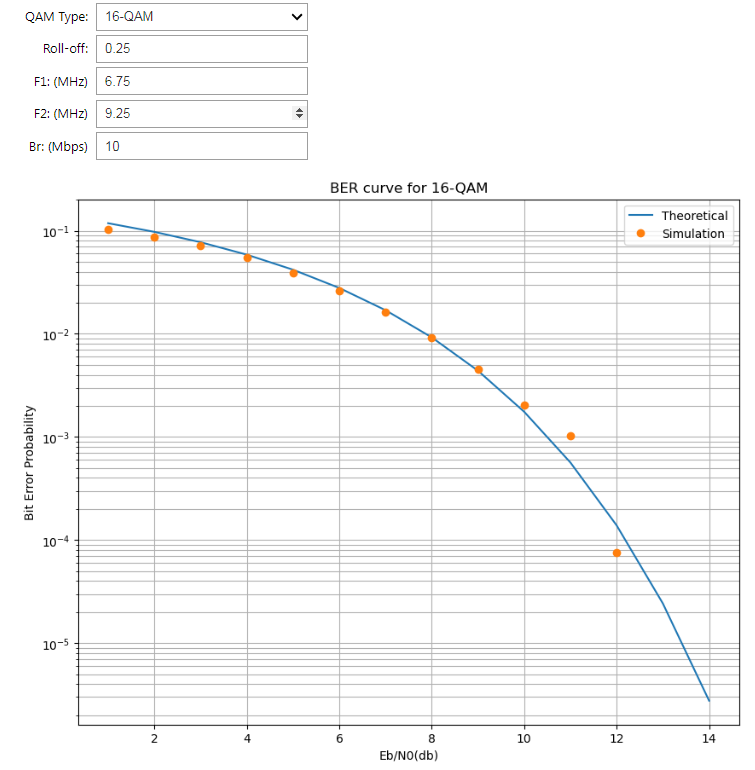


[**Κώδικας – Chapter 5 - PSK Constellation**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/PSK_constellation.ipynb)

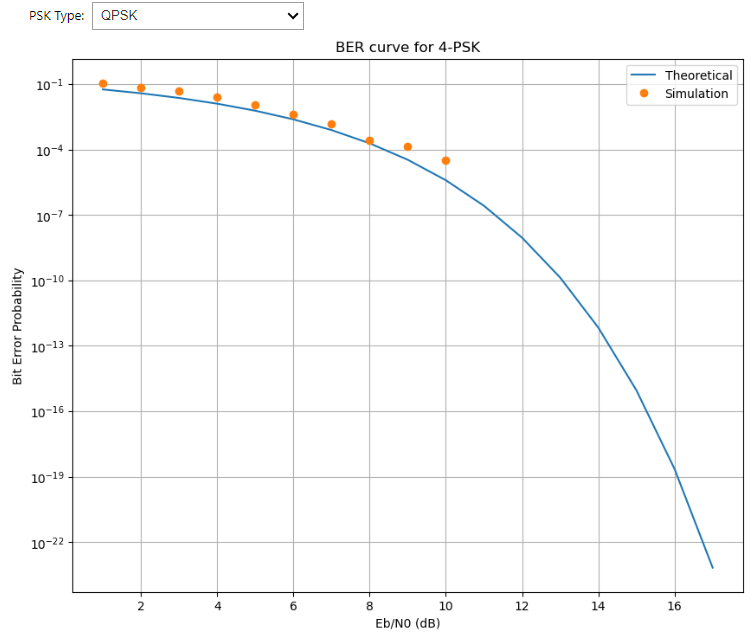


#### QAM και PSK Καμπύλες BER:

[**Κώδικας – Chapter 5 - QAM BER**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/QAM_BER_tool.ipynb)

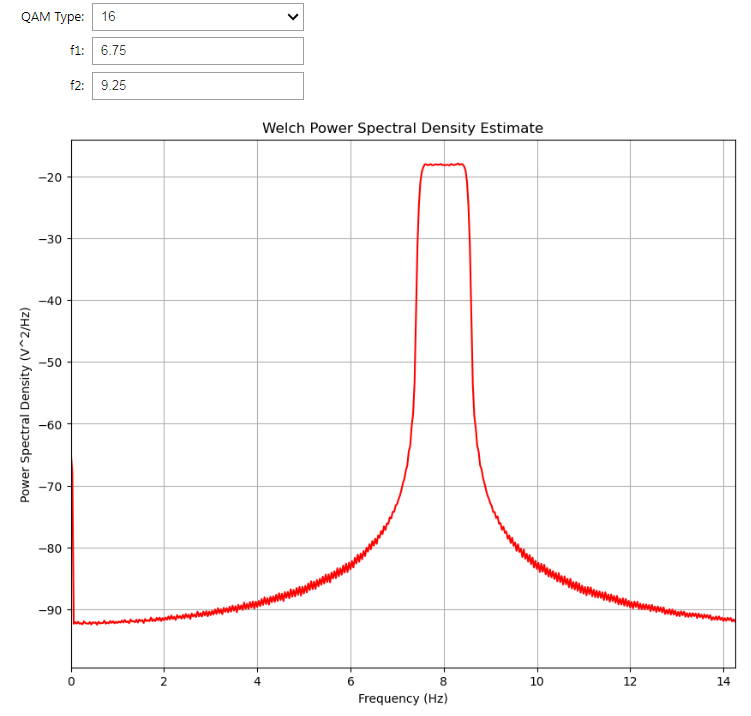


[**Κώδικας – Chapter 5 - PSK BER**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/PSK_BER_tool.ipynb)



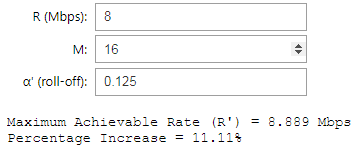
#### Διαδραστική Φασματική Πυκνότητα Ισχύος:

[**Κώδικας – Chapter 5 - Διαδραστική Φασματική Πυκνότητα Ισχύος**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/power_welch_spectral_density.ipynb)



#### Διαδραστικός Υπολογισμός Παραμέτρων:

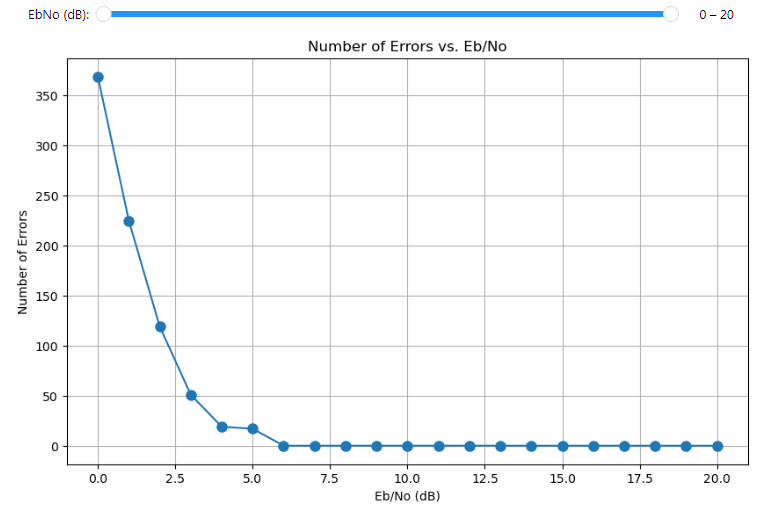
[**Κώδικας – Chapter 5 - Διαδραστικός Υπολογισμός Παραμέτρων**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%205/interactive_parameters.ipynb)



### Chapter 6: Digital Modulation FSK and MSK

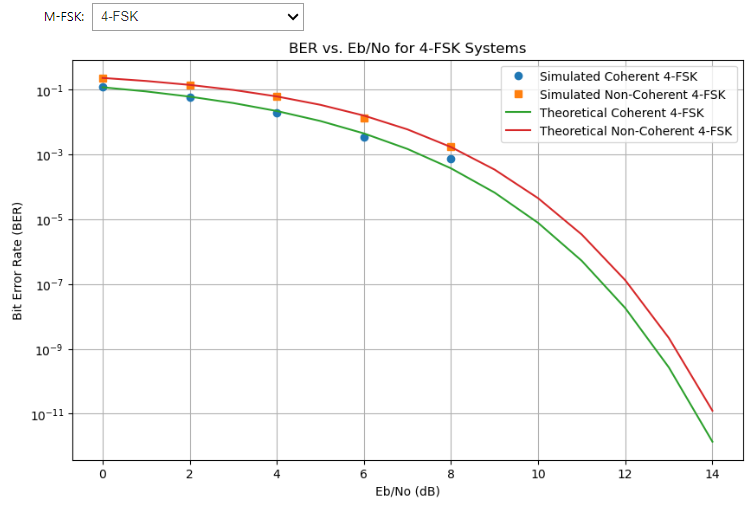
#### Αριθμός σφαλμάτων για Eb/No:

[**Κώδικας – Chapter 6 - Αριθμός σφαλμάτων για Eb/No**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%206/error_number_for_EbNo.ipynb)



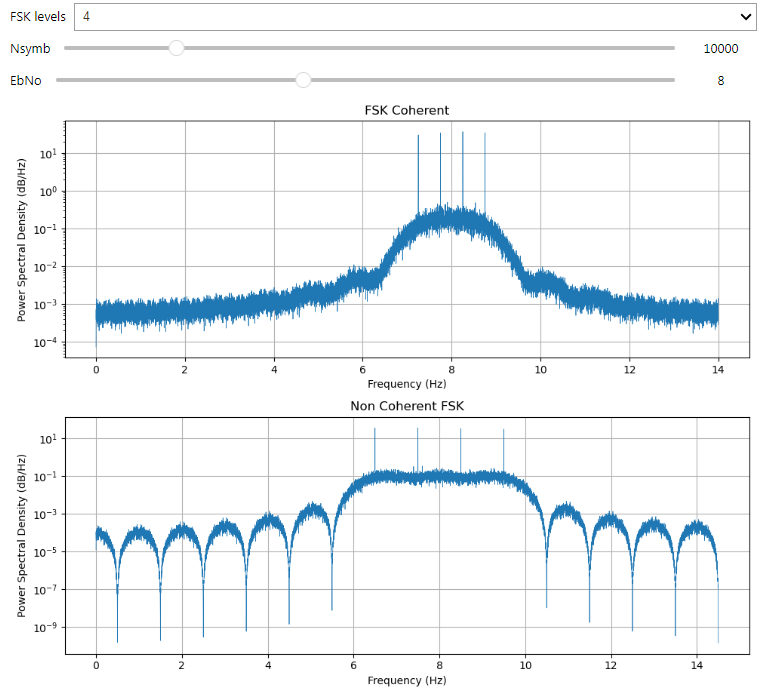
#### FSK Bit Error Rate:

[**Κώδικας – Chapter 6 - FSK Bit Error Rate**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%206/FSK_BER_tool.ipynb)



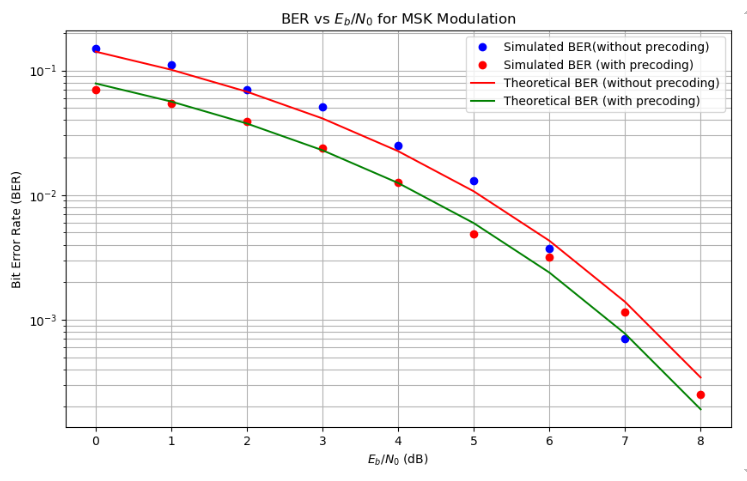
#### Ανάλυση Φάσματος coherent και non coherent FSK:

[**Κώδικας – Chapter 6 - Analysis coherent και non coherent FSK**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%206/FSK_frequency_analysis.ipynb)



#### MSK Bit Error Rate:

[**Κώδικας – Chapter 6 - MSK Bit Error Rate**](https://github.com/ntua-el17840/Interactive-Digital-Communications/blob/f0fdd47bbaf43c531ceec69d6a3ab1a271069233/theory_components/Chapter%206/MSK_BER_tool.ipynb)



#### Συμπεράσματα

ΛΑλσδφ

## Γενικά Συμπεράσματα

Γενικά, παρατηρείται ότι ο κώδικας σε MATLAB είναι ταχύτερος σε σύγκριση με τον αντίστοιχο κώδικα σε Python, ακόμη και με τη χρήση των βιβλιοθηκών που αναφέρονται. Η διαφορά αυτή οφείλεται στο γεγονός ότι το MATLAB είναι σχεδιασμένο ειδικά για μαθηματικούς και αριθμητικούς υπολογισμούς, καθιστώντας το πιο αποδοτικό για προσομοιώσεις συστημάτων ψηφιακών επικοινωνιών. Επιπλέον, οι ενσωματωμένες βιβλιοθήκες του MATLAB, όπως το Communications Toolbox, είναι εξαιρετικά βελτιστοποιημένες για συγκεκριμένες εφαρμογές, επιτρέποντας την εκμετάλλευση των δυνατοτήτων πολυνηματικής επεξεργασίας και χρήσης GPU. Παρόλο που η Python διαθέτει ισχυρές βιβλιοθήκες, όπως η NumPy και η SciPy, συχνά απαιτούν παραπάνω βελτιστοποίηση για να επιτύχουν ανάλογες επιδόσεις.

Ωστόσο, η σύγκριση δεν είναι απόλυτα ισοδύναμη, καθώς ο κώδικας MATLAB δεν περιλαμβάνει τις διαδραστικές δυνατότητες που προστέθηκαν στην υλοποίηση με Python. Η Python προσφέρει ευελιξία και πρόσβαση σε πληθώρα βιβλιοθηκών ανοιχτού κώδικα, καθιστώντας την δημοφιλή στην ακαδημαϊκή κοινότητα, ιδιαίτερα για την ανάπτυξη διαδραστικών εφαρμογών.

Στις περισσότερες περιπτώσεις, δεδομένου ότι ο χρόνος εκτέλεσης είναι εξαιρετικά μικρός και στις δύο υλοποιήσεις, η διαφορά μπορεί να θεωρηθεί αμελητέα. Ωστόσο, στα γραφήματα του Bit Error Rate (BER) παρατηρείται σημαντική απόκλιση στον χρόνο εκτέλεσης, με τον κώδικα Python να παρουσιάζει καθυστέρηση σε σχέση με τον αντίστοιχο σε MATLAB. Αυτό οφείλεται κυρίως στην εξειδίκευση του MATLAB στους αριθμητικούς υπολογισμούς, καθώς και στην ικανότητά του να εκμεταλλεύεται καλύτερα το διαθέσιμο υλικό για απαιτητικές υπολογιστικές εργασίες.

Κεφάλαιο 5

# Μελλοντικές επεκτάσεις

## Επέκταση σε Εργαστηριακές Ασκήσεις

Με βάση τα παραπάνω κομμάτια κώδικα, στο τεύχος [Εργασία Λάμπρου] περιγράφεται πως υλοποιήθηκε η ένταξη των εργαστηριακών ασκήσεων στο Jupyter Book για τη επίτευξη της τελικής διαδικτυακής εφαρμογής.

## Εισαγωγή νέων εργαλείων

Με βάση τα εργαλεία που χρησιμοποιήσαμε στο project, μπορούμε να εξετάσουμε τις ακόλουθες επεκτάσεις και προσθήκες στο μέλλον:

1. **Plotly ή Bokeh**: Ενσωμάτωση προηγμένων διαδραστικών γραφημάτων για καλύτερη οπτικοποίηση δεδομένων.
2. **Ενσωμάτωση Machine Learning**: Χρήση βιβλιοθηκών όπως το scikit-learn για προχωρημένες αναλύσεις.
3. **Ανάπτυξη Mobile Εφαρμογής**: Χρήση πλαισίων όπως το Kivy για πρόσβαση από κινητές συσκευές.
4. **Διεθνοποίηση (i18n)**: Υποστήριξη πολλαπλών γλωσσών για ευρύτερο κοινό.
5. **Αυτοματισμός με CI/CD**: Ενσωμάτωση εργαλείων όπως το GitHub Actions για συνεχή ανάπτυξη και δοκιμές.
6. **Ενσωμάτωση Βάσεων Δεδομένων**: Χρήση SQL ή NoSQL βάσεων για αποθήκευση και διαχείριση δεδομένων.
7. **Cloud Deployment**: Φιλοξενία της εφαρμογής σε πλατφόρμες όπως AWS ή Azure για καλύτερη κλιμάκωση.
8. **Ασφάλεια και Ταυτοποίηση**: Προσθήκη μηχανισμών πιστοποίησης χρηστών για προστασία δεδομένων.
9. **Ενσωμάτωση Realtime Συνεργασίας**: Χρήση εργαλείων όπως το Google Colab για συνεργατική εργασία.
10. **Προηγμένες Οπτικοποιήσεις 3D**: Χρήση βιβλιοθηκών όπως το VisPy για τρισδιάστατες αναπαραστάσεις.
11. **Χρήση Docker Containers**: Για ευκολότερη ανάπτυξη και μεταφορά της εφαρμογής.

Κεφάλαιο 6

# Σύνοψη

## Στόχοι της Εργασίας

Οι στόχοι της παρούσας εργασίας επικεντρώνονται στην αναζήτηση, αξιολόγηση, επιλογή και χρήση κατάλληλων εργαλείων που θα υποστηρίξουν τη δημιουργία μιας διαδραστικής ιστοσελίδας για την προσομοίωση συστημάτων ψηφιακών επικοινωνιών. Η πλατφόρμα αυτή στοχεύει να προσφέρει στους φοιτητές τη δυνατότητα να εξερευνούν και να κατανοούν καλύτερα τις αρχές των ψηφιακών επικοινωνιών, μέσα από πρακτικές εφαρμογές και πειραματισμό.

Ένας βασικός στόχος της εργασίας είναι να εντοπιστούν εργαλεία που προσφέρουν επαρκή διαδραστικότητα και ευκολία χρήσης, διατηρώντας παράλληλα την ευελιξία που απαιτείται για την προσομοίωση και ανάλυση συστημάτων σε πραγματικό χρόνο. Η έρευνα επικεντρώνεται σε εργαλεία ανοιχτού κώδικα, τα οποία επιτρέπουν την ενσωμάτωση διαδραστικών στοιχείων σε εκπαιδευτικά περιβάλλοντα.

Η γλώσσα προγραμματισμού Python, λόγω της ευρείας χρήσης της στις επιστήμες των δεδομένων και της ανάλυσης, αποτελεί κεντρικό εργαλείο στην υλοποίηση των στόχων αυτών. Τέλος, απαραίτητο στοιχείο θα αποτελέσει η ενσωμάτωση kernel στην ιστοσελίδα για τη ζωντανή εκτέλεση του κώδικα Python απευθείας στον φυλλομετρητή, εξασφαλίζοντας, έτσι, για τους χρήστες, έναν εύκολο και άμεσο τρόπο αλληλεπίδρασης με τις προσομοιώσεις σε πραγματικό χρόνο.

## Περίληψη κύριων βημάτων και μεθοδολογίας

Η παρούσα εργασία περιγράφει αναλυτικά τα πρώτα στάδια υλοποίησης της διαδικτυακής εφαρμογής, η οποία αναπτύχθηκε σε συνεργασία με τον Λάμπρο Φραγκουλόπουλο. Ειδικότερα, στο παρόν τεύχος παρατίθεται η ανάλυση των βημάτων 1-9 και , τα οποία αποτελούν το θεμέλιο για την ανάπτυξη του συνολικού έργου. Για περαιτέρω πληροφορίες σχετικά με τα επόμενα βήματα και την εξέλιξη του έργου, ο αναγνώστης παραπέμπεται στην αντίστοιχη εργασία του Λάμπρου Φραγκουλόπουλου [Εργασία Λάμπρου].

1. **Αναζήτηση, δοκιμή και τελική επιλογή των κατάλληλων εργαλείων για την ολοκλήρωση του project.**
2. **Δημιουργία αποθετηρίου στο GitHub για τη διαχείριση των αρχείων του έργου.**
3. **Κατασκευή βασικών κομματιών κώδικα python γύρω από την θεωρία και τις εργαστηριακές ασκήσεις του μαθήματος "Ψηφιακές Επικοινωνίες Ι", με διαδραστικές δυνατότητες. (Βάση μας ο κώδικας matlab του μαθήματος)**
4. **Ανάπτυξη και διαμόρφωση του Jupyter Book ως εργαλείο παρουσίασης και εκτέλεσης του εκπαιδευτικού υλικού.**
5. **Τροποποίηση του αρχείου config.yml για την ενσωμάτωση της επιλογής binder, επιτρέποντας την εκτέλεση κώδικα σε πραγματικό χρόνο μέσω ενός εξωτερικού kernel.**
6. **Δημιουργία του αρχείου requirements.txt, περιλαμβάνοντας όλες τις απαιτούμενες βιβλιοθήκες Python για την εξασφάλιση της λειτουργικότητας του kernel μέσω του MyBinder.**
7. **Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για τη δημιουργία των αρχείων της ιστοσελίδας.**
8. **Χρήση του GitHub Pages για την ανάπτυξη και φιλοξενία της διαδικτυακής εφαρμογής.**
9. **Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για την δημοσίευση της ιστοσελίδας στο GitHub Page που δημιουργήσαμε.**
10. Αναδιάρθρωση της δομής του Jupyter Book μέσω αλλαγών στο αρχείο toc.yml, ώστε να ευθυγραμμίζεται με τη διάρθρωση των εργαστηριακών ασκήσεων του μαθήματος "Ψηφιακές Επικοινωνίες Ι".
11. Μετάφραση του περιεχομένου των εργαστηριακών ασκήσεων στα Αγγλικά.
12. Ενσωμάτωση του μεταφρασμένου περιεχομένου στα Jupyter Notebooks που δημιουργήθηκαν. Ένα για κάθε άσκηση.
13. Εντοπισμός των σημείων που απαιτούν διαδραστικότητα και ενσωμάτωση από τον υπάρχον κώδικα, ή και εκ νέου σχεδιασμός, αντίστοιχων λειτουργιών.
14. Μετατροπή του υπάρχοντος βοηθητικού κώδικα από MATLAB σε Python, με χρήση των βιβλιοθηκών ψηφιακής επικοινωνίας και γραφικής απεικόνισης δεδομένων. (Με την βοήθεια των κομματιών κώδικα από το βήμα 3)
15. Προσθήκη διαδραστικών λειτουργιών στον κώδικα Python μέσω των κατάλληλων βιβλιοθηκών. (Με την βοήθεια των κομματιών κώδικα από το βήμα 3)
16. Ανανέωση του αρχείου requirements.txt, ώστε να περιλαμβάνει τις τελικές βιβλιοθήκες python που χρησιμοποιήθηκαν.
17. Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για τη δημιουργία των αρχείων της ιστοσελίδας.
18. Εκτέλεση των κατάλληλων εντολών μέσω του Jupyter Book για την δημοσίευση της ιστοσελίδας στο GitHub Page που δημιουργήσαμε.

## Βασικά αποτελέσματα

Γενικά, παρατηρείται ότι ο κώδικας σε MATLAB είναι ταχύτερος σε σύγκριση με τον αντίστοιχο κώδικα σε Python, ακόμη και με τη χρήση των βιβλιοθηκών που αναφέρονται. Ωστόσο, η σύγκριση δεν είναι απόλυτα ισοδύναμη, καθώς ο κώδικας MATLAB δεν περιλαμβάνει τις διαδραστικές δυνατότητες που προστέθηκαν στην υλοποίηση με Python.

Στις περισσότερες περιπτώσεις, δεδομένου ότι ο χρόνος εκτέλεσης είναι εξαιρετικά μικρός και στις δύο υλοποιήσεις, η διαφορά μπορεί να θεωρηθεί αμελητέα. Ωστόσο, στα γραφήματα του Bit Error Rate (BER) παρατηρείται σημαντική απόκλιση στον χρόνο εκτέλεσης, με τον κώδικα Python να παρουσιάζει καθυστέρηση σε σχέση με τον αντίστοιχο σε MATLAB.

## Συμπεράσματα

# Πηγές

*Altair - Documentation*. (n.d.). Ανάκτηση από https://altair-viz.github.io/

*Bokeh - Documentation*. (n.d.). Ανάκτηση από https://docs.bokeh.org/en/latest/

*CommPy - Documentation*. (n.d.). Ανάκτηση από https://github.com/veeresht/CommPy

*Dash - Documentation*. (n.d.). Ανάκτηση από https://dash.plotly.com/

*ggplot - Documentation*. (n.d.). Ανάκτηση από http://ggplot.yhathq.com/

*GitHub - Documentation*. (n.d.). Ανάκτηση από https://docs.github.com/en

*Google Colab - Documentation*. (n.d.). Ανάκτηση από https://colab.research.google.com/notebooks/intro.ipynb

*Gradio - Documentation*. (n.d.). Ανάκτηση από https://gradio.app/docs/

*Holoviews - Documentation*. (n.d.). Ανάκτηση από https://holoviews.org/

*IPython - Documentation*. (n.d.). Ανάκτηση από https://ipython.org/documentation.html

*ipywidgets - Documentation*. (n.d.). Ανάκτηση από https://ipywidgets.readthedocs.io/en/stable/

*Jupyter Book - Binder*. (n.d.). Ανάκτηση από https://jupyterbook.org/en/stable/interactive/launchbuttons.html#add-a-launch-on-binder-button

*Jupyter Book - Documentation*. (n.d.). Ανάκτηση από https://jupyterbook.org/intro.html

*Jupyter Book – Google Colab*. (n.d.). Ανάκτηση από https://jupyterbook.org/en/stable/interactive/launchbuttons.html#add-a-launch-on-google-colab-button

*Jupyter Book - JupyterHub*. (n.d.). Ανάκτηση από https://jupyterbook.org/en/stable/interactive/launchbuttons.html#add-a-launch-on-jupyterhub-button

*Jupyter Book - Notebook Interface*. (n.d.). Ανάκτηση από https://jupyterbook.org/en/stable/interactive/launchbuttons.html#control-the-notebook-interface-that-opens

*Jupyter Book - Thebe*. (n.d.). Ανάκτηση από https://jupyterbook.org/en/stable/interactive/thebe.html

*Math Module - Documentation*. (n.d.). Ανάκτηση από https://docs.python.org/3/library/math.html

*Matplotlib - Documentation*. (n.d.). Ανάκτηση από https://matplotlib.org/stable/contents.html

*MkDocs - Documentation*. (n.d.). Ανάκτηση από (https://www.mkdocs.org/)

*NumPy - Documentation*. (n.d.). Ανάκτηση από https://numpy.org/doc/

*Pandas Visualization - Documentation*. (n.d.). Ανάκτηση από https://pandas.pydata.org/docs/user\_guide/visualization.html

*Panel - Documentation*. (n.d.). Ανάκτηση από https://panel.holoviz.org/

*Plotly - Documentation*. (n.d.). Ανάκτηση από https://plotly.com/python/

*Plotly Dash - Documentation*. (n.d.). Ανάκτηση από https://dash.plotly.com/

*PyPlot - Documentation*. (n.d.). Ανάκτηση από https://matplotlib.org/stable/api/pyplot\_summary.html

*PySDR GitHub*. (n.d.). Ανάκτηση από https://github.com/ve3wwg/pySDR

*PySimpleGUI - Documentation*. (n.d.). Ανάκτηση από https://pysimplegui.readthedocs.io/en/latest/

*PyWavelets - Documentation*. (n.d.). Ανάκτηση από https://pywavelets.readthedocs.io/en/latest/

*scikit-dsp-comm - Documentation*. (n.d.). Ανάκτηση από https://scikit-dsp-comm.readthedocs.io/en/latest/

*SciPy - Documentation*. (n.d.). Ανάκτηση από https://docs.scipy.org/doc/scipy/reference/

*scipy.signal - Documentation*. (n.d.). Ανάκτηση από https://docs.scipy.org/doc/scipy/reference/signal.html

*Seaborn - Documentation*. (n.d.). Ανάκτηση από https://seaborn.pydata.org/

*SimPy - Documentation*. (n.d.). Ανάκτηση από https://simpy.readthedocs.io/en/latest/

*Streamlit - Documentation*. (n.d.). Ανάκτηση από https://docs.streamlit.io/

*Tkinter - Documentation*. (n.d.). Ανάκτηση από https://docs.python.org/3/library/tkinter.html

*VisPy - Documentation*. (n.d.). Ανάκτηση από http://vispy.org/

*Voila - Documentation*. (n.d.). Ανάκτηση από https://voila.readthedocs.io/en/stable/

# Παράρτημα Ι

## Πλήρης Οδηγός Εγκατάστασης

Η δημιουργία και δημοσίευση ενός Jupyter Book είναι μια αποτελεσματική μέθοδος για τη συγγραφή και διανομή διαδραστικών βιβλίων που περιλαμβάνουν κείμενο και εκτελέσιμο κώδικα. Σε αυτόν τον οδηγό, περιγράφεται η διαδικασία εγκατάστασης της Python και των απαραίτητων βιβλιοθηκών, η δημιουργία ενός Jupyter Book και η δημοσίευσή του μέσω GitHub.

#### Εγκατάσταση της Python στον Υπολογιστή

Για να ξεκινήσεις με το Jupyter Book, η πρώτη κίνηση είναι η εγκατάσταση της Python. Ακολούθησε τα παρακάτω βήματα:

1. Κατέβασε το εκτελέσιμο αρχείο της Python από την επίσημη ιστοσελίδα.
2. Κατά την εγκατάσταση, βεβαιώσου ότι έχεις επιλέξει την επιλογή "Add Python to PATH".
3. Ολοκλήρωσε την εγκατάσταση ακολουθώντας τις οδηγίες.

Μετά την εγκατάσταση, άνοιξε ένα τερματικό (Command Prompt ή PowerShell) και επιβεβαίωσε την επιτυχημένη εγκατάσταση με την εντολή:

python –version

#### Εγκατάσταση των Βιβλιοθηκών με Conda

Η χρήση του Conda, ενός διαχειριστή πακέτων και περιβαλλόντων για Python, διευκολύνει την εγκατάσταση των απαραίτητων βιβλιοθηκών για το Jupyter Book. Ακολούθησε τις παρακάτω εντολές για να εγκαταστήσεις τις βιβλιοθήκες μέσω Conda:

Δημιουργία περιβάλλοντος Conda:

conda create --name myenv python=3.9

conda activate myenv

Εγκατάσταση του **Notebook** και **Kernels**:

conda install -c conda-forge notebook

conda install -c conda-forge nb\_conda\_kernels

Εγκατάσταση βιβλιοθηκών για επιστημονικούς υπολογισμούς:

conda install -c anaconda scipy

conda install -c anaconda numpy

conda install anaconda::pandas

Εγκατάσταση βιβλιοθηκών για γραφήματα:

conda install -c conda-forge matplotlib

conda install -c plotly plotly

Εγκατάσταση του **Dash** για διαδραστικές εφαρμογές:

conda install -c conda-forge dash

conda install -c conda-forge dash-bootstrap-components

Εγκατάσταση βιβλιοθηκών ήχου και άλλων απαραίτητων εργαλείων:

conda install conda-forge::python-sounddevice

conda install nbconvert

Εγκατάσταση του **MATLAB Kernel** για την υποστήριξη εκτέλεσης κώδικα MATLAB μέσα σε Jupyter Notebooks:

python -m pip install matlab\_kernel

python -m matlab\_kernel install –user

#### Εγκατάσταση Jupyter Book

Για τη δημιουργία του Jupyter Book, πρέπει πρώτα να εγκαταστήσουμε το πακέτο του:

pip install -U jupyter-book

#### Δημιουργία Jupyter Book

Η δημιουργία του Jupyter Book περιλαμβάνει τη χρήση αρχείων όπως τα **\_toc.yml** και **\_config.yml** για τη διαχείριση της δομής και των ρυθμίσεων του βιβλίου. Ακολουθεί η διαδικασία:

1. **Δημιουργία νέου Jupyter Book**: Για να δημιουργήσεις ένα νέο βιβλίο, χρησιμοποίησε την εντολή:

jupyter-book create mybook

Αυτό θα δημιουργήσει έναν φάκελο με όλα τα απαραίτητα αρχεία, συμπεριλαμβανομένων των \_toc.yml και \_config.yml κ.α. Πρέπει να αντικαταστάσεις τα αρχεία \_config.yml, \_toc.yml, intro.ipynb με τα αντίστοιχα τους και να αντιγράψεις στο ίδιο directive τους φακέλους content, files με τα περιεχόμενα τους και το αρχείο digital\_communications\_intro.ipynb. Το documentation των αρχείων και των φακέλων του jyputer book περιλαμβανέται στο δεύτερο παράρτημα για την πλήρη κατανόηση του.

#### Εντολές για Δημιουργία και Προβολή του Βιβλίου

Για να χτίσεις και να δεις το βιβλίο σου τοπικά:

1. Χτίσε το βιβλίο:

jupyter-book build mybook/

1. Για να δεις το βιβλίο στον τοπικό server, πήγαινε στον φάκελο \_build/html και άνοιξε το αρχείο index.html. Εναλλακτικά, μπορείς να χρησιμοποιήσεις τον ακόλουθο server:

python -m http.server --directory mybook/\_build/html

Αυτό θα ανοίξει το βιβλίο στο localhost και θα μπορείς να το δεις τοπικά στον browser σου.

#### Ανέβασμα στο GitHub και Δημοσίευση στο Διαδίκτυο

Για να ανεβάσεις το βιβλίο στο GitHub και να το δημοσιεύσεις, ακολούθησε τα παρακάτω βήματα:

* Δημιουργία GitHub Repository:

1. Συνδέσου στον λογαριασμό σου στο GitHub και δημιούργησε ένα νέο repository.
2. Ακολούθως, στο τερματικό του τοπικού σου συστήματος:

git init  
git add .  
git commit -m "Initial commit"  
git remote add origin <URL\_του\_Repository>  
git push -u origin master

* Δημοσίευση του Jupyter Book: Για τη δημοσίευση του βιβλίου μέσω GitHub Pages, χρησιμοποίησε την εντολή:

jupyter-book publish gh-pages mybook/ --repository <URL\_του\_Repository>

Η εντολή αυτή θα δημιουργήσει ένα branch gh-pages και θα δημοσιεύσει το βιβλίο στο διαδίκτυο μέσω της υπηρεσίας GitHub Pages.

* Διανομή του Βιβλίου:

1. Μετά τη δημοσίευση, το βιβλίο θα είναι διαθέσιμο στη διεύθυνση https://<όνομα\_χρήστη>.github.io/<repository\_name>.
2. Κοινοποίησε αυτή τη διεύθυνση για να επιτρέψεις σε άλλους να δουν το διαδραστικό βιβλίο σου.

Συμπεράσματα

Η εγκατάσταση και η δημιουργία ενός Jupyter Book είναι μια διαδικασία που προσφέρει μια ισχυρή πλατφόρμα για τη συγγραφή και δημοσίευση διαδραστικών εγγράφων που ενσωματώνουν κώδικα και θεωρία. Χρησιμοποιώντας τα εργαλεία που παρέχονται μέσω της Python και του Conda, μπορείς να δημιουργήσεις μια πλήρως παραμετροποιήσιμη και επεκτάσιμη μαθησιακή πλατφόρμα. Η διαδικασία αυτή περιλαμβάνει όλα τα στάδια από την εγκατάσταση βιβλιοθηκών μέχρι τη δημοσίευση του έργου στο διαδίκτυο μέσω του GitHub, καθιστώντας τη γνώση προσβάσιμη σε παγκόσμιο επίπεδο.

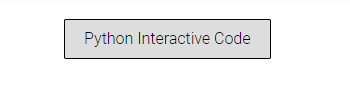
# Παράρτημα ΙΙ

## User Guide for Jupyter Book with Sphinx-Thebe

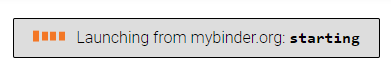
Αυτός ο οδηγός περιγράφει τη χρήση διαδραστικού κώδικα σε ένα Jupyter Book που χρησιμοποιεί το Sphinx-Thebe για εκτέλεση κώδικα. Ακολουθήστε τα παρακάτω βήματα για να εξασφαλίσετε ομαλή λειτουργία του διαδραστικού περιβάλλοντος.

### Εκκίνηση Διαδραστικού Κώδικα

* Αρχικά, πατήστε το κουμπί Interactive Code στην κορυφή της σελίδας.



Σχήμα : Κουμπί Interactive code

* Περιμένετε όσο η διαδικασία δείχνει μηνύματα (όπως launching και building κ.α.). Όταν το σύστημα εμφανίσει ότι είναι Ready, η σελίδα είναι έτοιμη για εκτέλεση διαδραστικού κώδικα.



Σχήμα 54: Αναμονή για έτοιμο kernel

* Αν δείτε ότι η διαδικασία καθυστερεί για πάνω από 5 λεπτά, δοκιμάστε να κάνετε refresh (ανανέωση) στη σελίδα και επαναλάβετε τα βήματα.

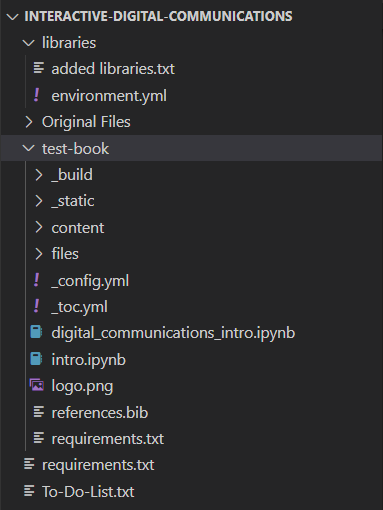
### Επανεκκίνηση και Εκτέλεση Όλων

* Αν θέλετε να επανεκκινήσετε όλο το σύστημα και να εκτελέσετε όλους τους κώδικες εκ νέου, πατήστε το κουμπί Restart and Run All. Αυτή η λειτουργία θα τρέξει όλους τους κώδικες από την αρχή, καθαρίζοντας προηγούμενες εκτελέσεις.

Με αυτά τα βήματα, το Jupyter Book σας θα λειτουργεί διαδραστικά και ομαλά, διευκολύνοντας την εκτέλεση και ανάλυση των εργαστηριακών ασκήσεων.

## Documentation: Basic Elements of a Jupyter Book

Το **Jupyter Book** είναι ένα ισχυρό εργαλείο για τη δημιουργία αλληλεπιδραστικών βιβλίων, εργαστηρίων και τεκμηρίωσης, τα οποία μπορούν να περιλαμβάνουν κώδικα, υπολογισμούς, γραφήματα και πολλά άλλα. Αυτή η τεκμηρίωση περιγράφει τα βασικά στοιχεία και τη δομή ενός Jupyter Book, προσφέροντας έναν οδηγό για τη δημιουργία περιεχομένου.



Σχήμα : Απαραίτητα αρχεία του jyputer book

### Δομή του Jupyter Book

Ένα Jupyter Book αποτελείται από μια συλλογή αρχείων που οργανώνονται σε μια συγκεκριμένη δομή για να δημιουργηθεί το τελικό βιβλίο. Βασίζεται στο Sphinx για την τεκμηρίωση και χρησιμοποιεί MyST Markdown για τη συγγραφή κειμένων και τη διαχείριση κώδικα.

#### Βασικά Στοιχεία:

* **\_config.yml:** Το αρχείο ρυθμίσεων του Jupyter Book. Περιέχει παραμέτρους για το πώς θα δημιουργηθεί και θα εμφανιστεί το βιβλίο, όπως τον τίτλο, το λογότυπο, τις επεκτάσεις και άλλες παραμέτρους διαμόρφωσης.
* **\_toc.yml:** Το αρχείο Table of Contents (Πίνακας Περιεχομένων), το οποίο καθορίζει την ιεραρχία των κεφαλαίων και των ενοτήτων στο βιβλίο σας.
* **Αρχεία περιεχομένου:** Αυτά είναι τα αρχεία σε μορφή Markdown (.md) ή Jupyter Notebooks (.ipynb) που αποτελούν το περιεχόμενο του βιβλίου.

### Αρχεία Περιεχομένου

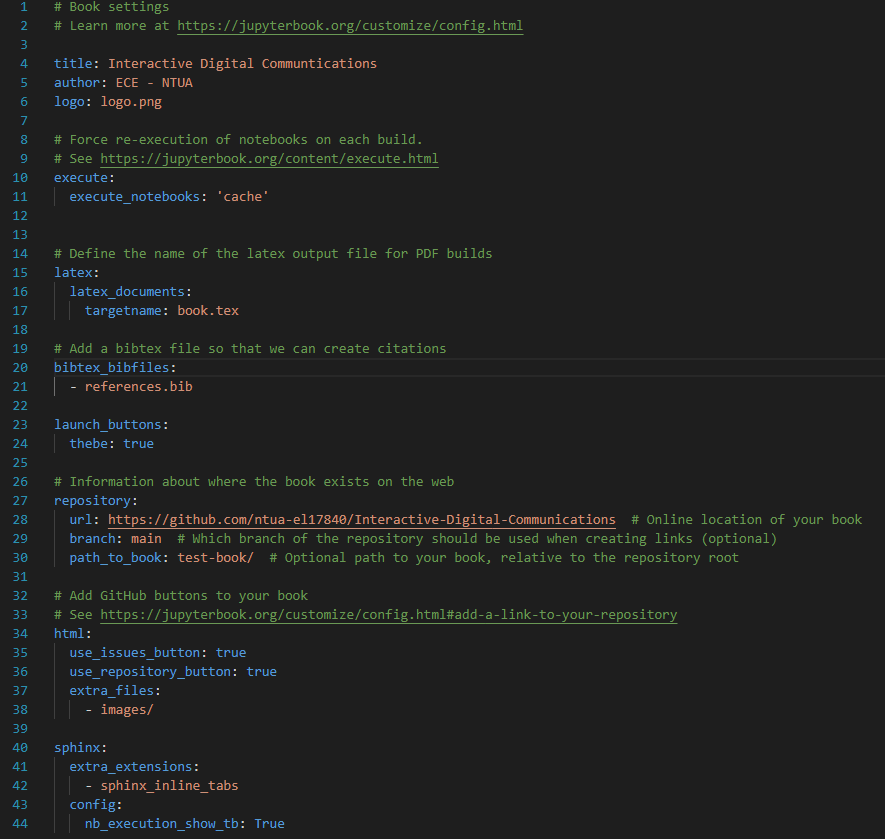
* Markdown Files (.md)
* Το MyST Markdown είναι μια επέκταση του Markdown που προσθέτει επιπλέον λειτουργίες.
* Jupyter Notebooks (.ipynb)
* Τα **Jupyter Notebooks** μπορούν να περιλαμβάνουν κείμενο, κώδικα και αποτελέσματα, όλα ενσωματωμένα σε ένα έγγραφο. Αυτά τα αρχεία επιτρέπουν τη διαδραστικότητα, καθώς ο αναγνώστης μπορεί να εκτελεί τα μπλοκ κώδικα κατά την ανάγνωση του βιβλίου.

### Ρυθμίσεις του Jupyter Book

\_config.yml

Το αρχείο **\_config.yml** καθορίζει τη γενική διαμόρφωση του βιβλίου σας. Κάποιες βασικές ρυθμίσεις περιλαμβάνουν:

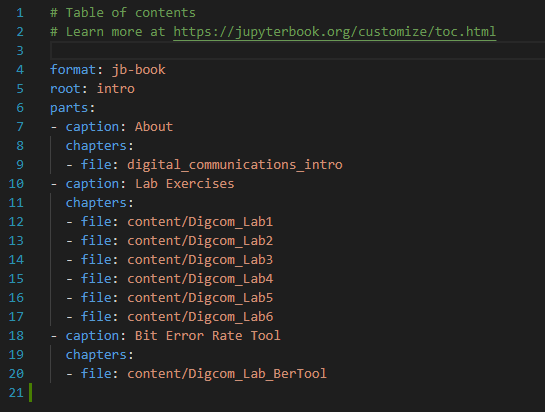
* **title**: Τίτλος του βιβλίου.
* **author**: Συγγραφέας ή ομάδα συγγραφέων.
* **logo**: Τοποθεσία του λογοτύπου που θα εμφανίζεται στο βιβλίο.
* **launch buttons**: Επιλογές για να επιτρέψετε στους χρήστες να τρέξουν το βιβλίο σε πλατφόρμες όπως το **Binder** ή το **Google Colab**.



Σχήμα : \_config.yml

\_toc.yml

Το αρχείο **\_toc.yml** οργανώνει το περιεχόμενο του βιβλίου σε κεφάλαια και υποκεφάλαια.

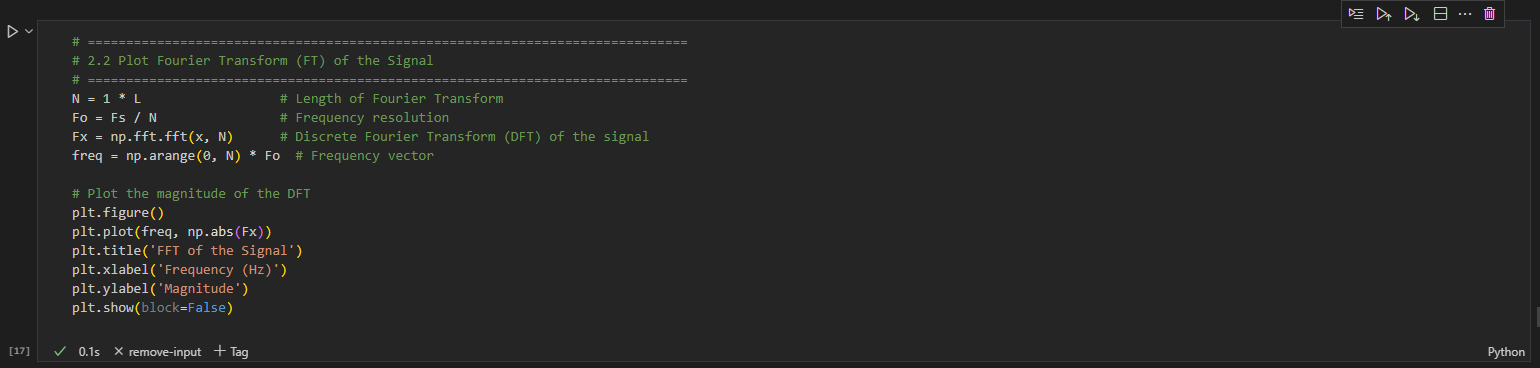


Σχήμα : \_toc.yml

Αυτό το αρχείο καθορίζει τη σειρά με την οποία τα αρχεία Markdown και Jupyter Notebooks θα εμφανιστούν στο βιβλίο.

### Προσθήκη Κώδικα

Το Jupyter Book υποστηρίζει την εκτέλεση κώδικα μέσω ενσωματωμένων Jupyter Notebooks ή MyST Markdown. Ο κώδικας μπορεί να τρέχει σε γλώσσες όπως Python, R, Julia, κ.λπ. Μπορείτε να δημιουργήσετε διαδραστικά κομμάτια με την παρακάτω σύνταξη:



Σχήμα : Εκτελέσιμο αρχείο κώδικα

### Διαδραστικότητα με το Thebe

Μια από τις πιο ισχυρές λειτουργίες του Jupyter Book είναι η χρήση του Thebe για την ενσωμάτωση διαδραστικού κώδικα απευθείας στην ιστοσελίδα. Με αυτόν τον τρόπο, οι χρήστες μπορούν να εκτελούν τα παραδείγματα κώδικα σε πραγματικό χρόνο χωρίς να χρειάζεται να κατεβάσουν τα αρχεία.

Για να ενεργοποιήσετε το Thebe, πρέπει να προσθέσετε τις παρακάτω ρυθμίσεις στο αρχείο \_config.yml:

launch\_buttons: thebe: true

### Δημιουργία και Ανάπτυξη του Βιβλίου

Για να δημιουργήσετε το Jupyter Book:

1. Βεβαιωθείτε ότι έχετε εγκαταστήσει το Jupyter Book. Μπορείτε να το εγκαταστήσετε με την ακόλουθη εντολή:

pip install jupyter-book

1. Για να δημιουργήσετε το βιβλίο, πλοηγηθείτε στο φάκελο του βιβλίου και εκτελέστε:

jupyter-book build mybookname/

Αυτό θα δημιουργήσει το βιβλίο σας σε μορφή HTML, το οποίο μπορείτε να ανοίξετε τοπικά ή να το ανεβάσετε σε μια ιστοσελίδα.

1. Για να αναπτύξετε το βιβλίο σας στο διαδίκτυο, μπορείτε να χρησιμοποιήσετε το GitHub Pages, Netlify, ή άλλες πλατφόρμες φιλοξενίας ιστοσελίδων.

### Συμπέρασμα

Το Jupyter Book προσφέρει ένα ευέλικτο περιβάλλον για τη δημιουργία αλληλεπιδραστικών και πλούσιων σε περιεχόμενο εγγράφων, επιτρέποντας τη σύνθεση κώδικα, κειμένου και γραφημάτων. Με τη χρήση MyST Markdown, Jupyter Notebooks, και τη διαδραστικότητα που προσφέρει το Thebe, το Jupyter Book καθιστά τη δημιουργία εκπαιδευτικών εγχειριδίων και τεκμηρίωσης απλή και ισχυρή ταυτόχρονα.

Αυτή η τεκμηρίωση παρέχει έναν συνοπτικό οδηγό για τα βασικά στοιχεία και τη δομή ενός Jupyter Book, προσφέροντας τις απαραίτητες πληροφορίες για να ξεκινήσετε τη δημιουργία και τη χρήση του.