# Εθνικό Μετσόβιο Πολυτεχνείο

# Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

## 1η Εργαστηριακή Άσκηση

**Μάθημα:** Ψηφιακή Επεξεργασία Σήματος

**Εξάμηνο:** 6ο

**Ονοματεπώνυμα:** Αλεξοπούλου Γεωργία (ΑΜ: 03120164), Γκενάκου Ζωή (ΑΜ: 03120015)

**Θέμα:** Εισαγωγή στην Ψηφιακή Επεξεργασία Σημάτων με Python και Εφαρμογές σε Ακουστικά Σήματα

## Μέρος 1ο - Φασματική Ανάλυση και Ανίχνευση Ημιτονοειδών με τον Διακριτό Μετ/σμό Fourier (DFT)

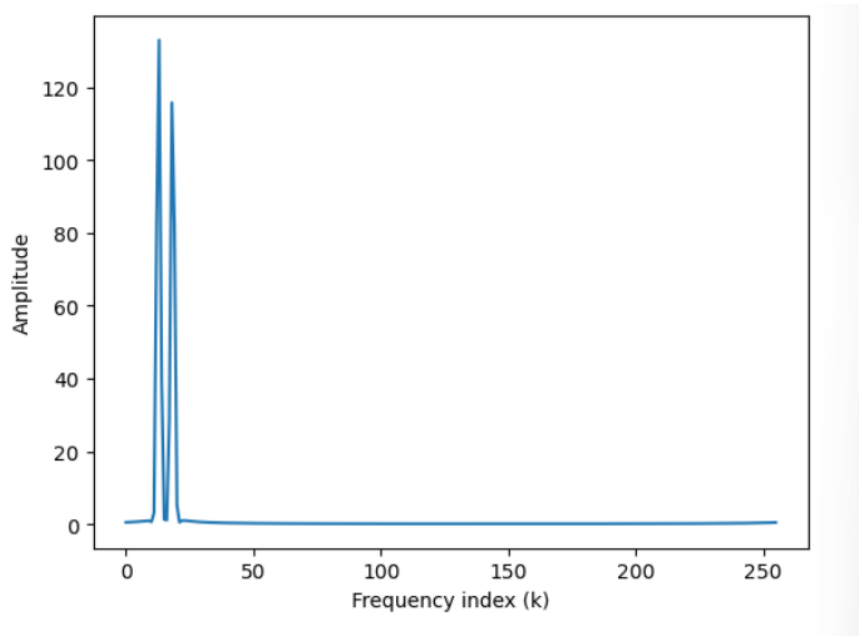
*Ερώτημα 1.1*

Στο πρώτο μέρος της άσκησης μας δίνονται τα παρακάτω δύο σήματα προς επεξεργασία:

$$x_1[n] = A_1 e^{j(\omega_1 n + \varphi_1)} \text{ και } x_2[n] = A_2 e^{j(\omega_2 n + \varphi_2)}$$

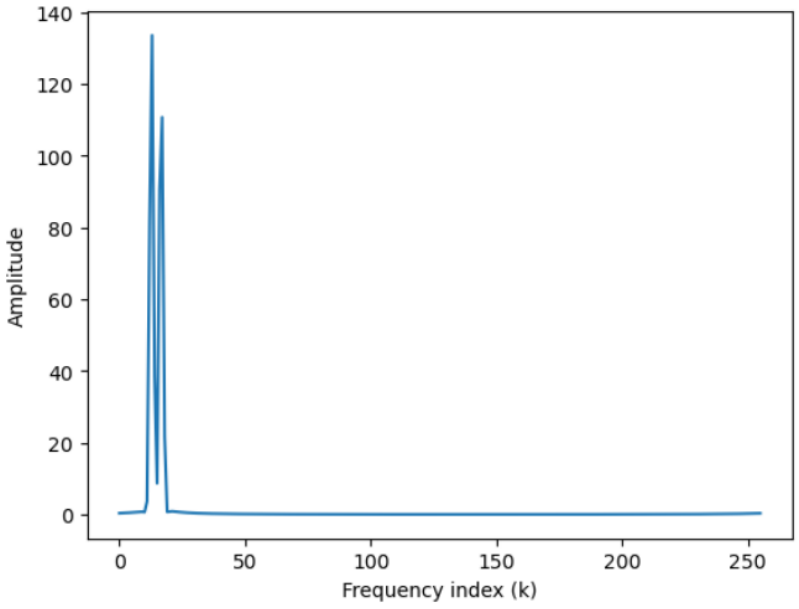
Το σήμα ανάλυσης y[n] προκύπτει από το άθροισμα των παραπάνω δύο σημάτων, πολλαπλασιασμένα με το παράθυρο Hamming.

Η άσκηση, αρχικά, μας ζητάει να υπολογίσουμε τον DFT μήκους N=256 δειγμάτων του σήματος και να σχεδιάσουμε το πλάτος του.

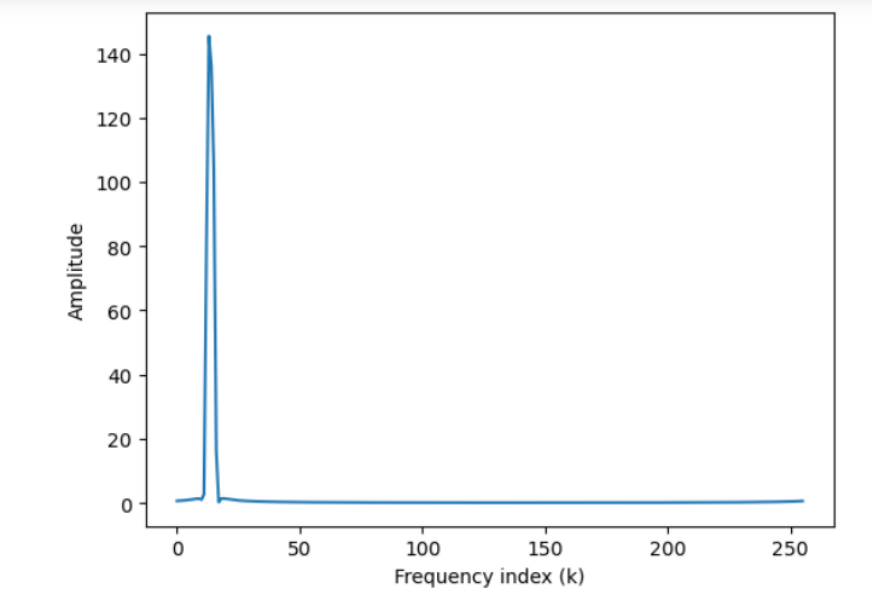



Για να βρούμε πόσο μικρή είναι η διαφορά Δω ώστε να ξεχωρίζουν οι κορυφές, χειροκίνητα, αλλάζουμε την συχνότητα $\omega_2$, μέχρι να δούμε τις δύο κορυφές να ενώνονται.

Αυτό συμβαίνει πρώτη φορά για $\omega_2 = \pi/7.7$



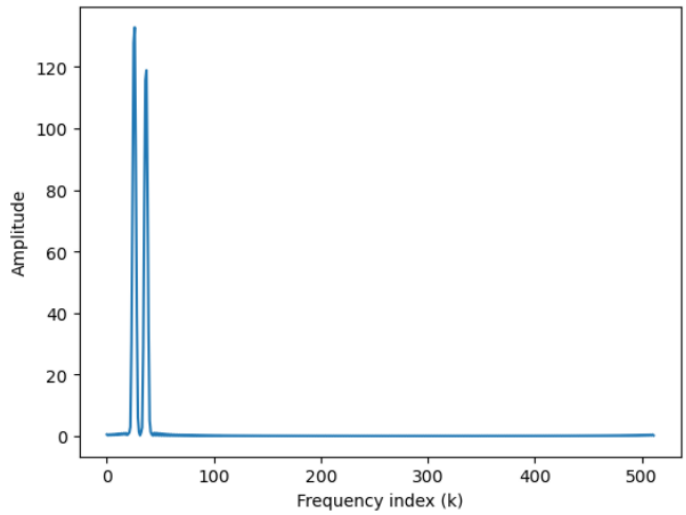

Και όταν $\omega_2 = \pi/8.8$ οι δύο κορυφές δεν είναι καν ευδιάκριτες



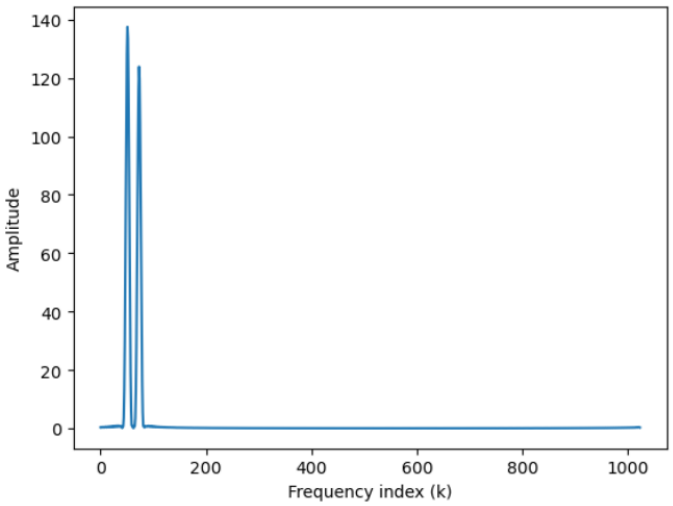

Με βάση τα αποτελέσματα, μπορούμε να παρατηρήσουμε ότι καθώς η διαφορά συχνότητας μεταξύ των δύο σημάτων εισόδου γίνεται μικρότερη, οι κορυφές στο DFT του παραθύρου σήματος γίνονται όλο και λιγότερο διακριτές, υποδηλώνοντας μείωση της διακριτικής ικανότητας του συστήματος. Αυτό οφείλεται στο γεγονός ότι όσο στενότερος είναι ο διαχωρισμός συχνοτήτων μεταξύ των σημάτων, τόσο περισσότερο επικαλύπτονται στον τομέα της συχνότητας, καθιστώντας πιο δύσκολο για το σύστημα να διακρίνει μεταξύ τους. Επομένως, είναι απαραίτητο να επιλέγουμε προσεκτικά τη λειτουργία παραθύρου για να μεγιστοποιείται η διακριτική ικανότητα του συστήματος για ένα δεδομένο σήμα εισόδου. Επιπλέον, η αύξηση του μήκους του σήματος εισόδου και του αριθμού των σημείων στο DFT μπορεί επίσης να βελτιώσει τη διακριτική ικανότητα του συστήματος, αν και με το κόστος της αυξημένης υπολογιστικής πολυπλοκότητας.

## *Ερώτημα 1.2*

Επαναλαμβάνοντας το παραπάνω πείραμα για διαφορετικά μήκη DFT, καταλήγουμε με τις εξης γραφικές:




Για N = 512




Για N = 1024

Συνοπτικά, η αύξηση του μήκους DFT από N=256 σε N=512 και N=1024 αυξάνει τη φασματική ανάλυση του DFT, γεγονός που καθιστά ευκολότερη τη διάκριση μεταξύ των συνιστωσών συχνότητας σε κοντινή απόσταση. Αυτό συμβαίνει επειδή η αύξηση του μήκους DFT αυξάνει την ανάλυση συχνότητας του DFT.

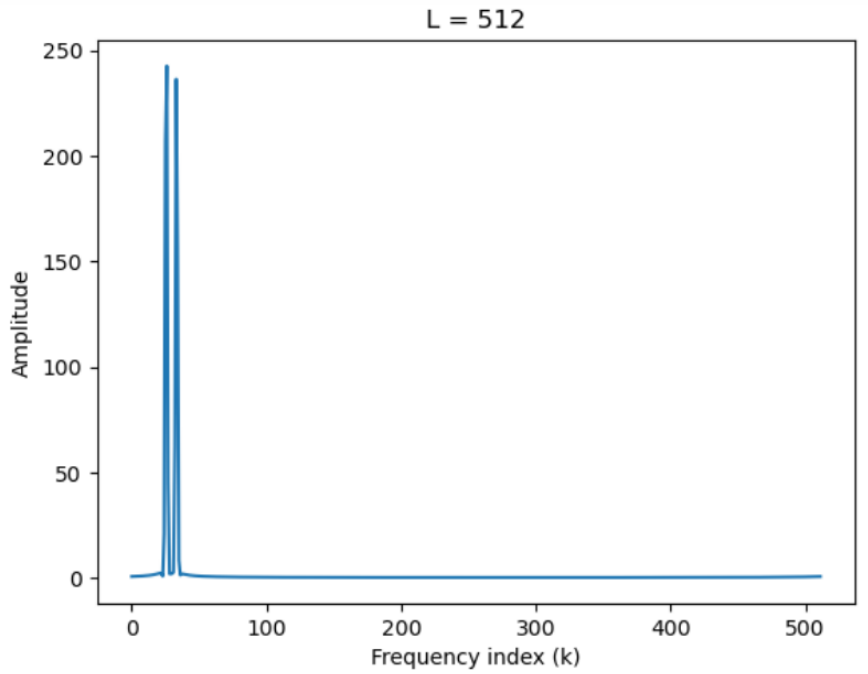
Ωστόσο, η φασματική διάκριση των δύο ημιτονοειδών, η οποία αναφέρεται στην ικανότητα του DFT να διακρίνει μεταξύ των φασματικών συνιστωσών των δύο ημιτονοειδών, καθορίζεται κυρίως από τον διαχωρισμό συχνότητας των δύο ημιτονοειδών και δεν επηρεάζεται από το μήκος DFT ή από το zero-padding.

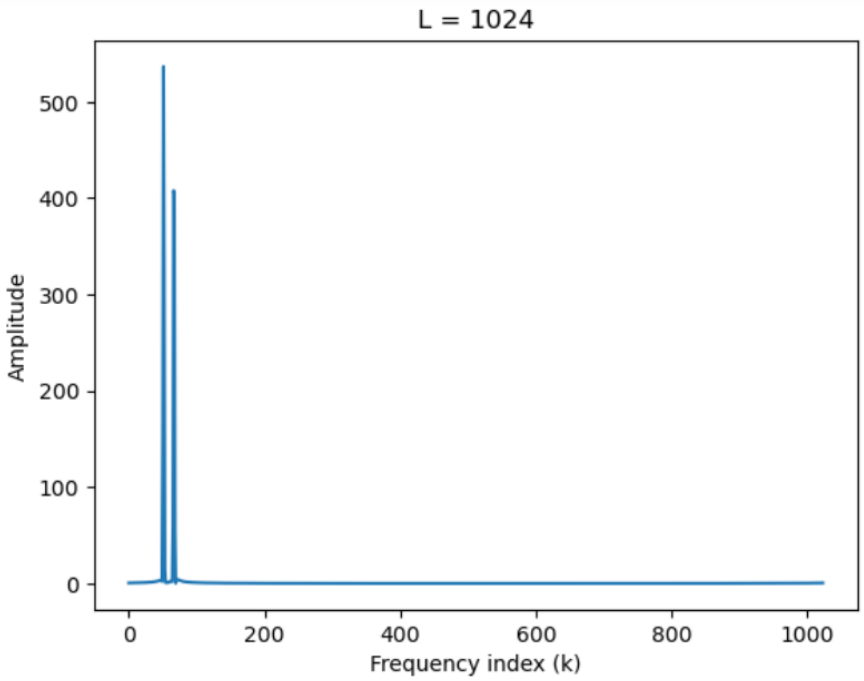
Επομένως, η αύξηση του μήκους DFT και το zero-padding μπορούν να βελτιώσουν την ποιότητα του φάσματος DFT όσον αφορά τη φασματική ανάλυση, αλλά η ικανότητα διάκρισης μεταξύ διαφορετικών στοιχείων συχνότητας καθορίζεται κυρίως από τον διαχωρισμό συχνότητας μεταξύ αυτών των στοιχείων και δεν επηρεάζεται σημαντικά από το μήκος DFT ή το zero-padding.  Αυτό συμβαίνει καθώς στο μήκος σήματος προσθέτονται μηδενικά και ως αποτέλεσμα δεν λαμβάνεται περαιτέρω χρήσιμη πληροφορία από το σήμα. . Αυτό που αλλάζει είναι το σχήμα των peaks στο φάσμα των συχνοτήτων τα οποία για μεγαλύτερα μήκη DFT πλησιάζουν καλυτέρα την συνάρτηση sinc.

## *Ερώτημα 1.3*

Για να παρατηρήσουμε την επίδραση της αύξησης του μήκους του σήματος στη φασματική διάκριση, θα επαναλάβουμε το ίδιο πείραμα όπως στην ερώτηση 1.1, αλλά αυτή τη φορά με μεγαλύτερα μήκη σήματος L=512 και L=1024. Θα επικεντρωθούμε στις συχνότητες στις οποίες παρατηρήσαμε προηγουμένως οριακή φασματική διάκριση (δηλαδή, $\omega_2$ κοντά στο $\omega_1$).

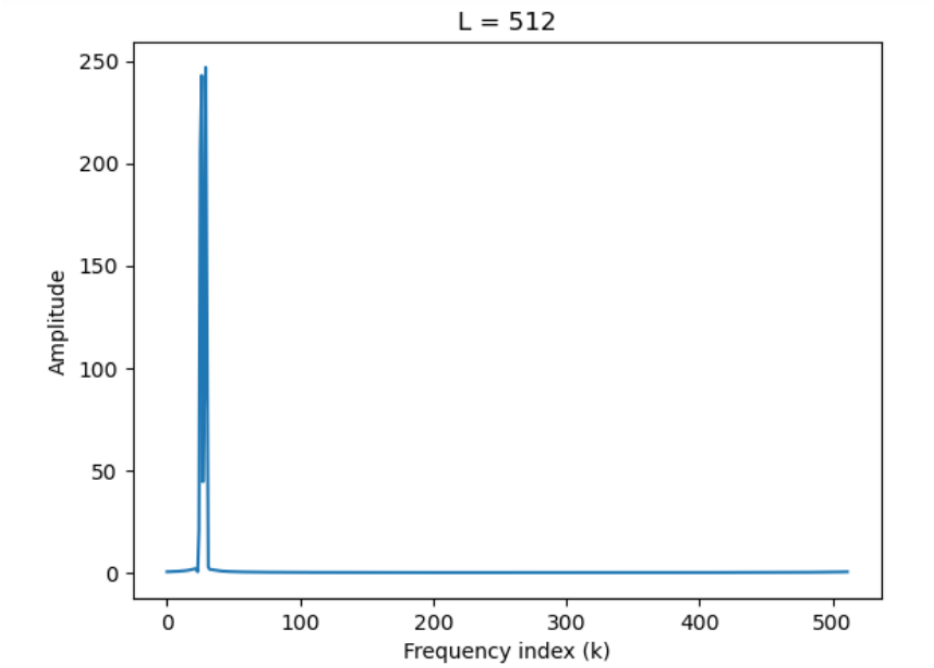
Για την οριακή τιμή από το ερώτημα 1.1 της $\omega_2 = \pi/7.7$ παίρνουμε τις παρακάτω γραφικές:

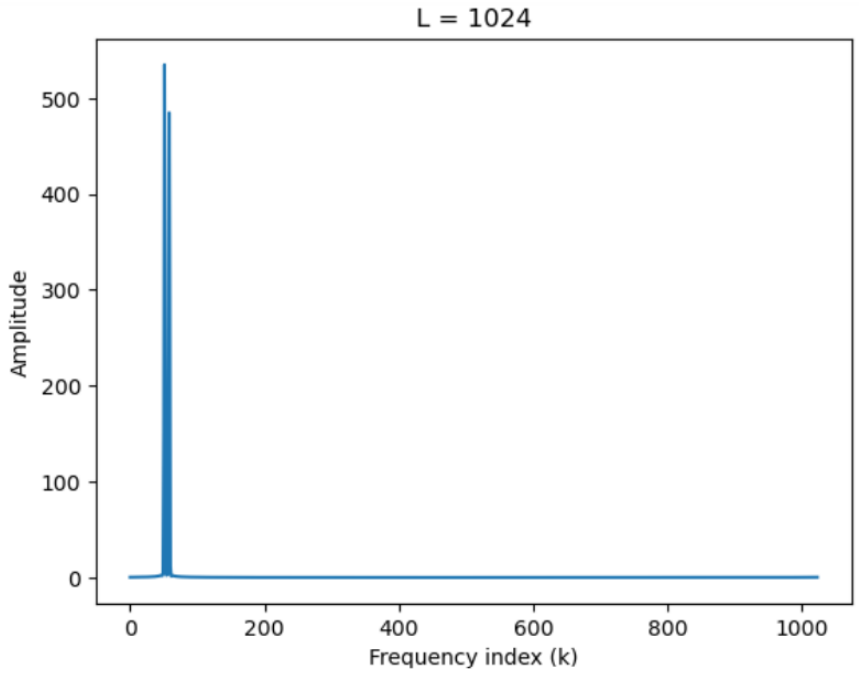



Εκεί που οι συχνότητες προηγουμένως επικαλύπτονταν, πλέον οι κορυφές ξεχωρίζουν τελείως.

Και αντίστοιχα για $\omega_2 = \pi/8.8$, όπου για L = 256 οι δύο κορυφές δεν ήταν ευδιάκριτες, τώρα μπορούμε να ξεχωρίσουμε τις δύο κορυφές, ακόμα και αν υπάρχει μερική επικάλυψη.

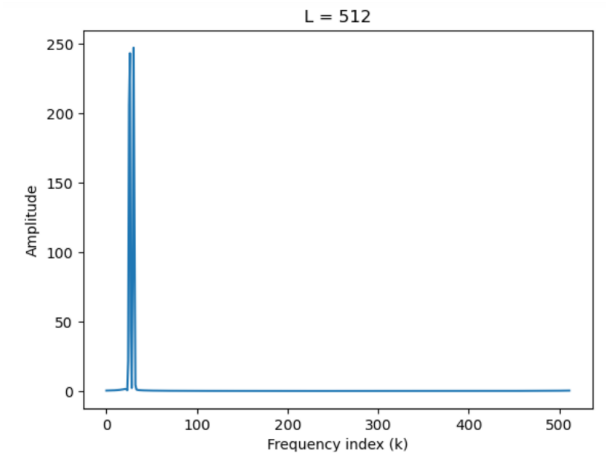


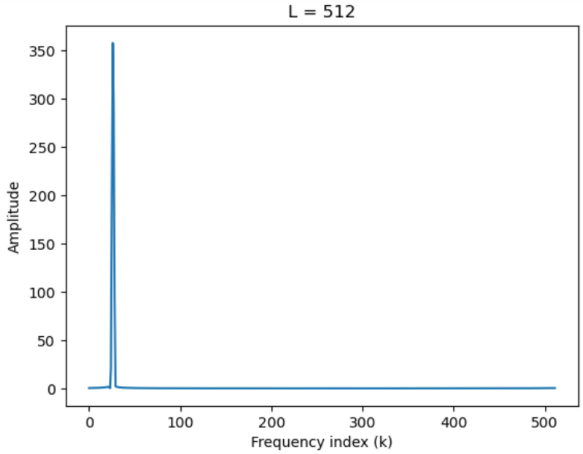




Όταν αυξάνουμε το μήκος του σήματος από L=256 σε L=512 και L=1024, παρατηρούμε ότι η ανάλυση συχνότητας του φάσματος βελτιώνεται, πράγμα που σημαίνει ότι μπορούμε να διακρίνουμε καλύτερα τις κοντινές συχνότητες.

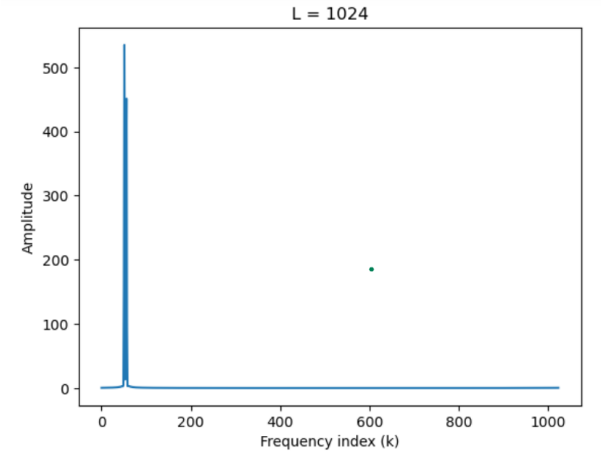
Συγκεκριμένα, με L=512 και L=1024, μπορούμε να δούμε περισσότερες λεπτομέρειες στο φάσμα και οι κορυφές είναι πιο στενές και καλύτερα διαχωρισμένες. Επιπλέον, ο κύριος λοβός του παραθύρου Hamming γίνεται στενότερος, γεγονός που μειώνει τη φασματική διαρροή και βελτιώνει την ανάλυση συχνότητας.

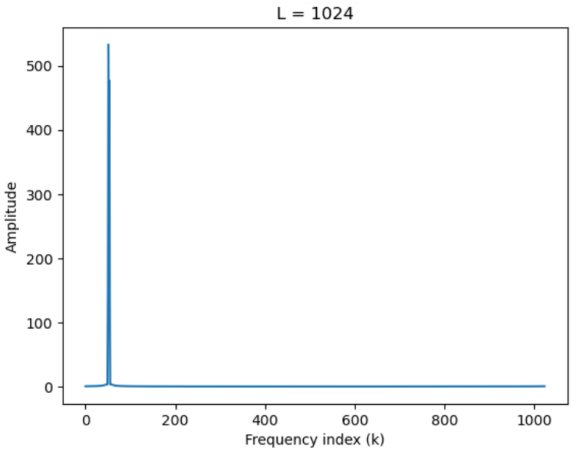
Επιπλέον για το νέο όριο φασματικής διάκρισης, για L = 512, βλέπουμε ότι η επικάλυψη ξεκινάει για $\omega_2 = \pi/8.5$ και για $\omega_2 = \pi/9.5$ οι δύο κορυφές δεν είναι ευδιάκριτες.



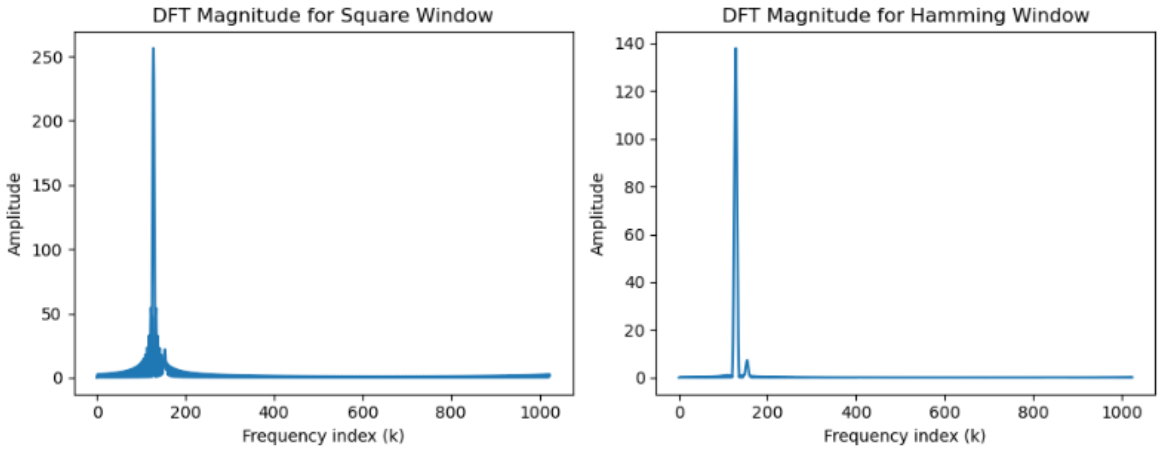




Και αντίστοιχα για L=1024, η επικάλυψη των κορυφών ξεκινάει για $\omega_2 = \pi/9.2$ και για $\omega_2 = \pi/9.7$ οι δύο κορυφές σχεδόν ταυτίζονται.



*Ερώτημα 1.4*




Όταν χρησιμοποιούμε το ορθογώνιο παράθυρο, παρατηρούμε τη φασματική διαρροή, καθώς οι αιχμηρές φασματικές κορυφές των μεμονωμένων ημιτονοειδών στο πεδίο συχνοτήτων δεν είναι ορατές λόγω των πλευρικών λοβών που προκαλούνται από το ορθογώνιο παράθυρο. Αυτό έχει ως αποτέλεσμα ένα θολό φάσμα με ενέργεια να εξαπλώνεται σε παρακείμενες συχνότητες.

Από την άλλη, όταν χρησιμοποιούμε το παράθυρο Hamming, παρατηρούμε σημαντική μείωση της φασματικής διαρροής, καθώς οι πλευρικοί λοβοί είναι πολύ εξασθενημένοι σε σύγκριση με το ορθογώνιο παράθυρο. Αυτό οδηγεί σε μια πιο ευκρινή και ακριβέστερη αναπαράσταση των υποκείμενων συχνοτήτων στον τομέα συχνοτήτων, με αποτέλεσμα μια σαφέστερη φασματική κορυφή για κάθε ημιτονοειδή.

Στο τρέχον παράδειγμα, τα σήματα x1[n] και x2[n] έχουν συχνότητες ω1 = 0,25π και ω2 = 0,3π αντίστοιχα, και πλάτη A1 = 1 και A2 = 0,05. Το σήμα y[n] προκύπτει προσθέτοντας x1[n] και x2[n] και στη συνέχεια πολλαπλασιάζοντας με ένα ορθογώνιο ή ένα παράθυρο Hamming. Το μήκος του σήματος είναι 256 και το μέγεθος DFT είναι 1024.

Επομένως, η χρήση μιας συνάρτησης παραθύρου όπως το Hamming μπορεί να βελτιώσει την ανάλυση συχνότητας και την ακρίβεια στη φασματική ανάλυση των σημάτων, καθώς μπορεί να μειώσει τη φασματική διαρροή που προκαλείται από το παράθυρο.

## Μέρος 2ο - Σύστημα Εντοπισμού Τηλεφωνικών Τόνων (Telephone Touch – Tones)

*Ερώτημα 2.1*

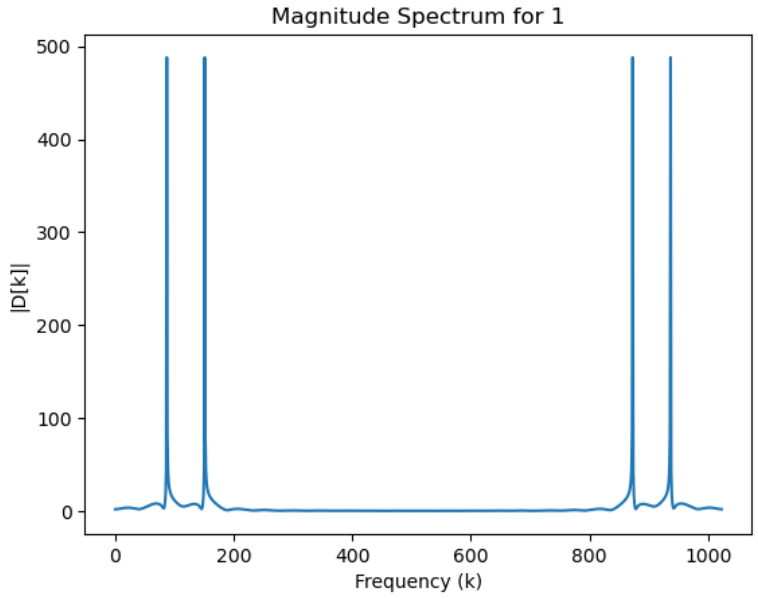
Για να δημιουργήσουμε τους 10 διαφορετικούς τόνους, θα ορίσουμε κάθε τόνο ως άθροισμα δύο ημιτονοειδών κυμάτων με τις αντίστοιχες συχνότητες με βάση τον πίνακα που δίνεται.

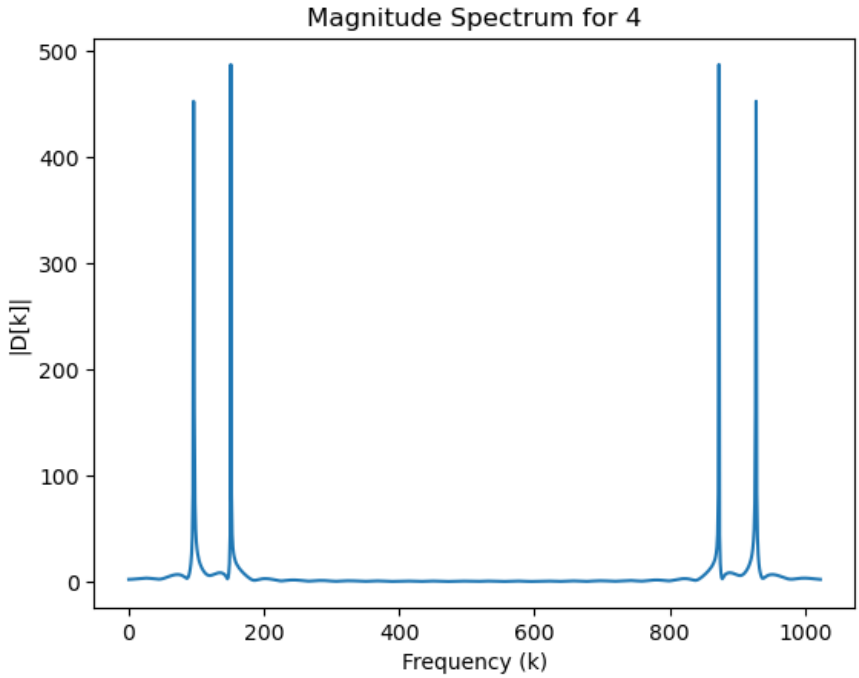
| | $\Omega_{column}$ | | |
|---|---|---|---|
| $\Omega_{row}$ | 0.9273 | 1.0247 | 1.1328 |
| 0.5346 | 1 | 2 | 3 |
| 0.5906 | 4 | 5 | 6 |
| 0.6535 | 7 | 8 | 9 |
| 0.7217 | | 0 | |

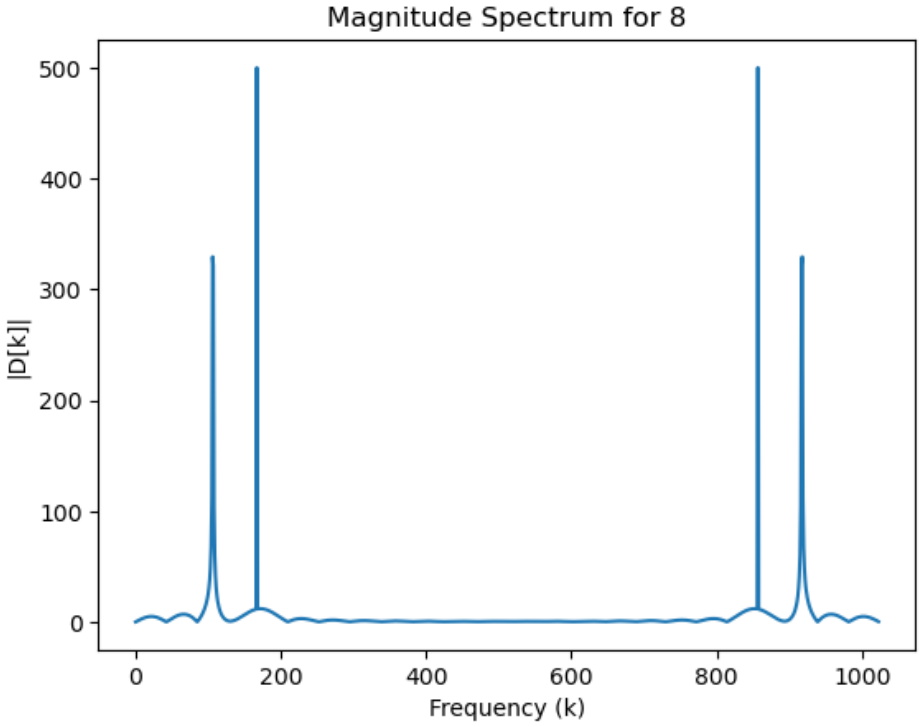
Θα χρησιμοποιήσουμε ρυθμό δειγματοληψίας 8192 Hz και κάθε τόνος θα έχει μήκος 1000 δειγμάτων.

*Ερώτημα 2.2*

Υπολογίζουμε τις γραφικές παραστάσεις των |D1[k]|, |D4[k]| και |D8[k]|.






Αφού σχεδιάσουμε το φάσμα μεγέθους κάθε σήματος, θα πρέπει να παρατηρήσουμε την παρουσία δύο κυρίαρχων κορυφών στον τομέα συχνότητας, που αντιστοιχούν στις δύο συχνότητες ημιτονοειδών κυμάτων που χρησιμοποιούνται για τη δημιουργία του τόνου.

Συγκεκριμένα, θα πρέπει να παρατηρήσουμε μια κορυφή στο φάσμα μεγέθους σε καθεμία από τις δύο συχνότητες που στον πίνακα για κάθε τόνο.

Τα φάσματα μεγέθους και των τριών σημάτων θα έχουν αιχμές στις αντίστοιχες συχνότητές τους (δηλαδή, για d1[n] στα 697 Hz και 1209 Hz, για d4[n] στα 770 Hz και 1209 Hz, και για d8[n] στα 852 Hz και 1336 Hz). Αυτές στον x άξονα, υπολογίζονται ως f = k*fs/N, όπου fs είναι η συχνότητα δειγματολειψίας και N είναι το μήκος FFT.

Τα μεγέθη των αιχμών θα εξαρτηθούν από το πλάτος της αντίστοιχης συνιστώσας συχνότητας στα αρχικά σήματα. Για παράδειγμα, δεδομένου ότι το d8[n] έχει μεγαλύτερο πλάτος στις δύο συνιστώσες συχνότητάς του σε σύγκριση με τα d1[n] και d4[n], θα πρέπει να περιμένουμε να δούμε μεγαλύτερα μεγέθη στις αιχμές του |D8[k]| σε σύγκριση με |D1[k]| και |D4[k]|.

Αξίζει επίσης να σημειωθεί ότι η ανάλυση συχνότητας του φάσματος μεγέθους καθορίζεται από το μήκος FFT που χρησιμοποιείται στον κώδικα. Τα μεγαλύτερα μήκη FFT παρέχουν καλύτερη ανάλυση συχνότητας, επιτρέποντάς μας να διακρίνουμε με μεγαλύτερη ακρίβεια μεταξύ των στοιχείων συχνότητας που βρίσκονται σε κοντινή απόσταση.

Τέλος, οι αιχμές στα φάσματα μεγέθους θα είναι συμμετρικές ως προς την κεντρική συχνότητα (δηλαδή, N/2) λόγω της πραγματικής αξίας φύσης των σημάτων.

## *Ερώτημα 2.3*

Για αυτό το ερώτημα, δημιουργείται μια ακολουθία ήχων με βάση το άθροισμα των αριθμών μητρώου μας και αποθηκεύεται ο ήχος που προκύπτει σε ένα αρχείο με το όνομα 'tone_sequence.wav'.  AM1 = 03120015 και AM2 = 03120164 τότε τα ζητούμενα ψηφία είναι:0 6 2 4 0 1 7 9 (= 03120015 + 03120164)

Οι τόνοι παράγονται συνδυάζοντας δύο ημιτονοειδή κύματα, ένα για τη συχνότητα γραμμής και ένα για τη συχνότητα της στήλης, που αντιστοιχεί σε κάθε ψηφίο του αθροίσματος. Οι ήχοι που δημιουργούνται συνδέονται με τη σιωπή για να δημιουργηθεί η πλήρης ακολουθία και ο ήχος που προκύπτει αποθηκεύεται σε ένα αρχείο χρησιμοποιώντας τη συνάρτηση scipy.io.wavfile.write.

Επίσης, μέσα στον κώδικα μας, έχουμε σε σχόλια την εντολή για την αναπαραγωγή του αρχείου μας.

*Ερώτημα 2.4*

Έχοντας το σήμα από το προηγούμενο ερώτημα, φαίνεται να υπάρχουν 7 διαφορετικοί μεταξύ τους αριθμοί ο καθένας εκ των οποίων αποτελείται από δύο ημίτονα.
Τα διαγράμματα δείχνουν το φάσμα ισχύος των αποτελεσμάτων FFT για κάθε ομάδα σημάτων με παράθυρο. Κάθε γραμμή στο διάγραμμα αντιπροσωπεύει το φάσμα ισχύος του αποτελέσματος FFT για ένα σήμα με παράθυρο και το χρώμα της γραμμής αντιστοιχεί στο ψηφίο που αντιπροσωπεύεται από το σήμα με παράθυρο.

Παρατηρούμε ότι το φάσμα ισχύος του αποτελέσματος FFT για κάθε σήμα με παράθυρο έχει ένα διακριτικό μοτίβο κορυφών και κοιλάδων. Η θέση και η απόσταση αυτών των κορυφών και κοιλάδων στον τομέα συχνότητας αντιστοιχεί στη θεμελιώδη συχνότητα και τις αρμονικές του υποκείμενου σήματος. Το ύψος κάθε κορυφής αντιπροσωπεύει την ποσότητα ισχύος σε αυτό το στοιχείο συχνότητας του σήματος.

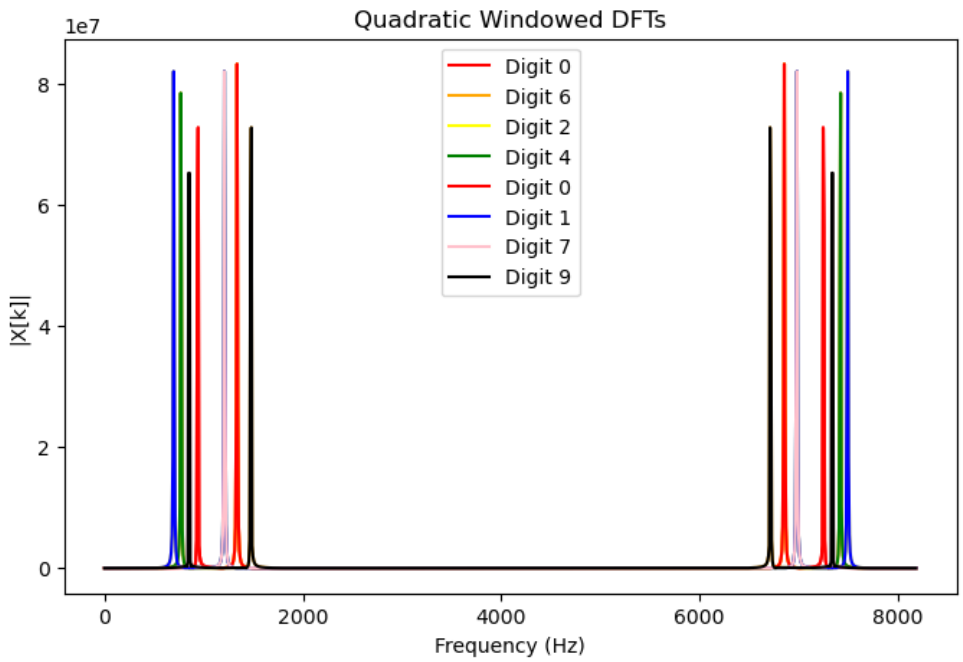
Στην γραφική παράσταση Quadratic Windowed DFTs, θα πρέπει να παρατηρήσετε ότι το φάσμα ισχύος για κάθε ψηφίο είναι λιγότερο ευδιάκριτο από το διάγραμμα Hamming Windowed DFTs. Αυτό συμβαίνει επειδή η συνάρτηση τετραγωνικού παραθύρου είναι λιγότερο αποτελεσματική στη μείωση της φασματικής διαρροής σε σύγκριση με τη συνάρτηση παραθύρου Hamming. Η φασματική διαρροή συμβαίνει όταν η λειτουργία παραθύρου παραμορφώνει το σήμα, προκαλώντας διαρροή ενέργειας σε παρακείμενες συχνότητες στο αποτέλεσμα FFT.

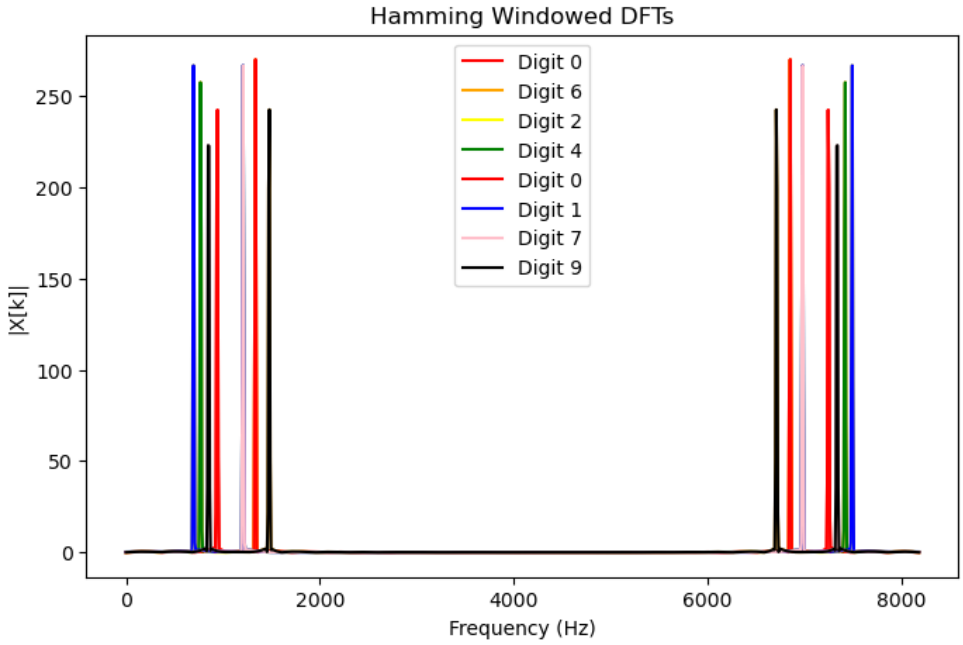
Συνολικά, οι γραφικές παραστάσεις δίνουν μια οπτική αναπαράσταση του περιεχομένου συχνότητας των παραθυρωμένων σημάτων και μπορούν να βοηθήσουν στον εντοπισμό των στοιχείων συχνότητας που είναι σημαντικά για την αναγνώριση κάθε ψηφίου.

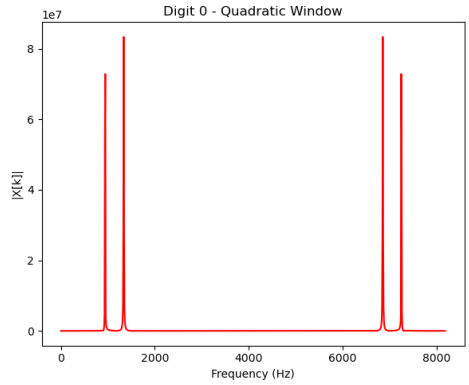
Επιπλέον σχεδιάζουμε και κάθε τόνο μόνο του, για να παρατηρήσουμε καλύτερα την μορφή τους και μετασχηματίζουμε τον άξονα x για να φαίνονται οι πραγματικές συχνότητες των σημάτων.

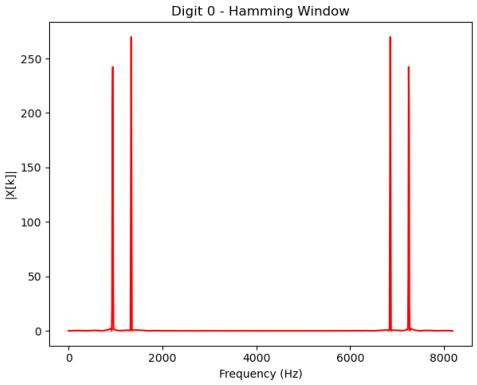
Από τα διαγράμματα, μπορούμε να παρατηρήσουμε ότι το DFT κάθε σήματος ψηφίου είναι μοναδικό και μπορεί να χρησιμοποιηθεί για τη διάκριση μεταξύ διαφορετικών ψηφίων. Μπορούμε να δούμε ότι διαφορετικά ψηφία έχουν διαφορετικές κορυφές στο DFT τους, υποδεικνύοντας διαφορετικές συχνότητες που υπάρχουν στο σήμα.

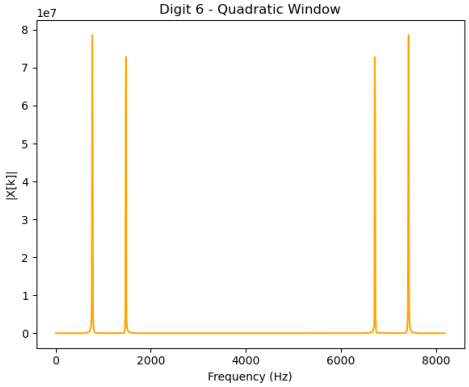
Όμως στα διαγράμματα όπου υπάρχουν όλα τα ψηφία μαζί, υπάρχουν κάποιες επικαλύψεις, διότι τα ψηφία μεταξύ τους μπορεί να έχουν κοινή είτε την χαμηλή έιτε την υψηλή τους συχνότητα.

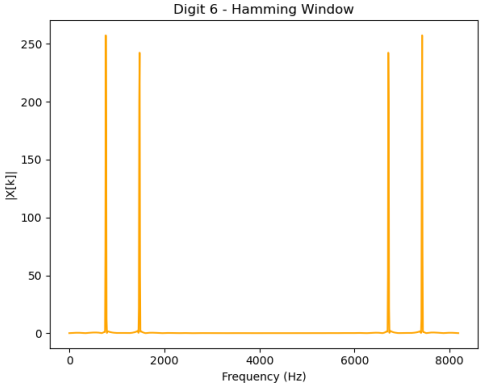

Quadratic Windowed DFTs
1e7
Digit 0
Digit 6
Digit 2
Digit 4
Digit 0
Digit 1
Digit 7
Digit 9
|X[k]|
Frequency (Hz)
0
2
4
6
8
0
2000
4000
6000
8000

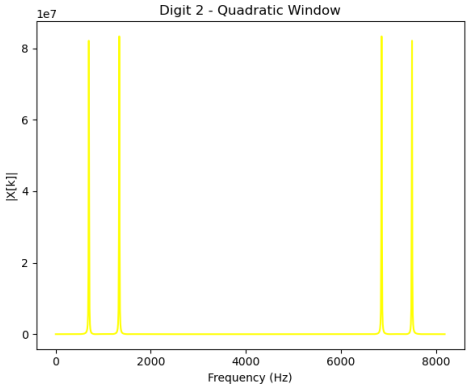

Hamming Windowed DFTs
Digit 0
Digit 6
Digit 2
Digit 4
Digit 0
Digit 1
Digit 7
Digit 9
|X[k]|
Frequency (Hz)
0
50
100
150
200
250
0
2000
4000
6000
8000

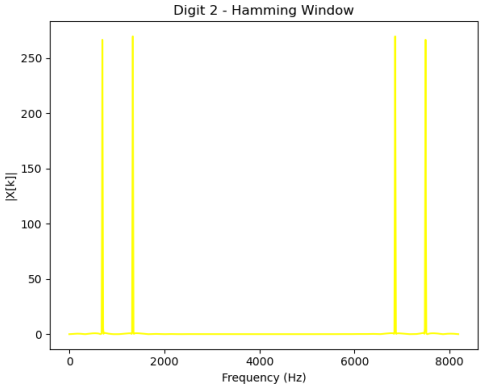

Digit 0 - Quadratic Window
1e7
|X[k]|
Frequency (Hz)

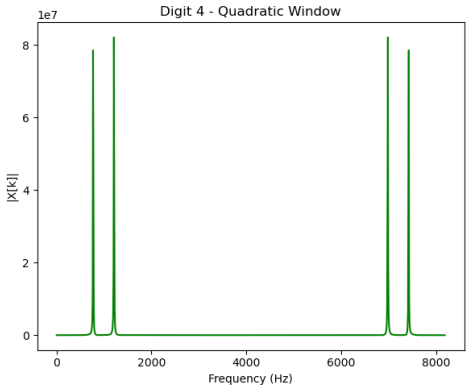

Digit 0 - Hamming Window
|X[k]|
Frequency (Hz)

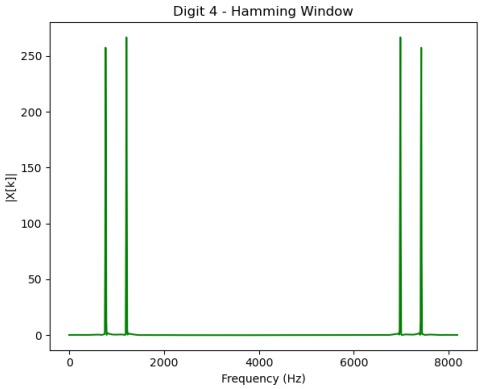

Digit 6 - Quadratic Window
1e7
|X[k]|
Frequency (Hz)

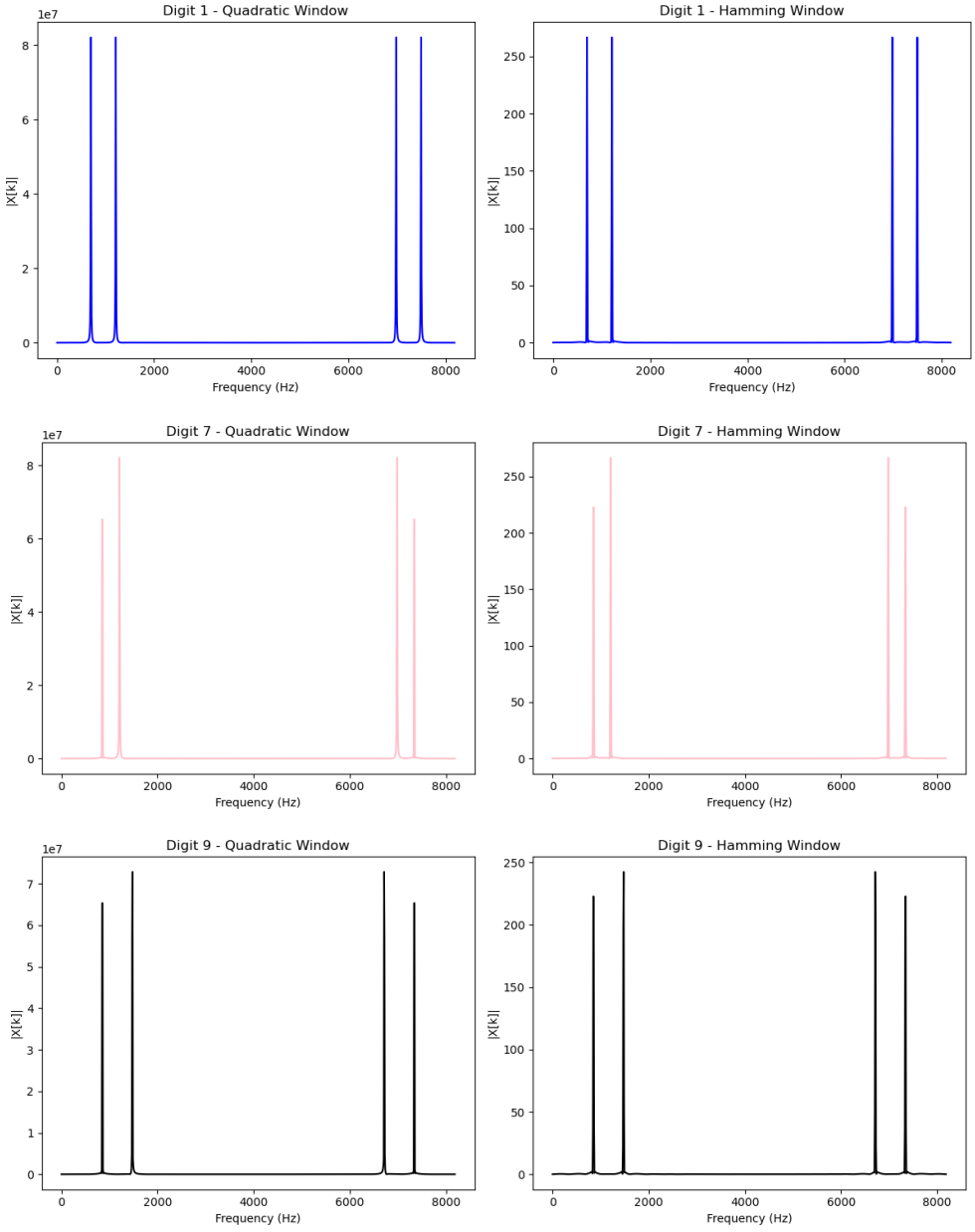

Digit 6 - Hamming Window
|X[k]|
Frequency (Hz)


Digit 2 - Quadratic Window
1e7
|X[k]|
Frequency (Hz)


Digit 2 - Hamming Window
|X[k]|
Frequency (Hz)


Digit 4 - Quadratic Window
1e7
|X[k]|
Frequency (Hz)


Digit 4 - Hamming Window
|X[k]|
Frequency (Hz)




*Ερώτημα 2.5*

Στο ερώτημα αυτό καλούμαστε να υπολογίσουμε τον δείκτη k για καθεμία από τις touch-tone frequencies. Για τον σκοπό αυτό, δημιουργούμε εκ νέου τη λίστα frequencies, η οποία αντιστοιχίζει κάθε ψηφίο στο ζεύγος των συχνοτήτων του. Ο δείκτης k αντιστοιχίζεται σε αυτό το ζεύγος συχνοτήτων και υπολογίζεται από τη σχέση

$$k = \frac{\omega N}{2\pi}$$, όπου N = 1000 (το μήκος κάθε τόνου)

Αξιοποιώντας τα σημεία στα οποία βρίσκονται οι κορυφές του DFT, καταλήγουμε στις touch-tone συχνότητες που μας ενδιαφέρουν.

*Ερώτημα 2.6*

Στο ερώτημα αυτό, ζητείται κατ' ουσίαν το deconstruction του σήματος 'tone_sequence.wav' που δημιουργήσαμε στο Ερώτημα 2.3. Για τον σκοπό αυτό, θα αξιοποιήσουμε και τη λίστα τιμών του δείκτη k που σχηματίσαμε στο προηγούμενο ερώτημα. Στην πραγματικότητα, η συνάρτηση ttdecode() που κατασκευάζουμε είναι ένας αλγόριθμος αναγνώρισης ακουστικών σημάτων (DTMF), ο οποίος αναγνωρίζει τις συχνότητες που αντιστοιχίζονται σε κάθε πλήκτρο τηλεφώνου και εκχωρεί το κάθε ψηφίο σε μια συγκεκριμένη συχνότητα. Η λειτουργία της ttdecode() στηρίζεται στην ανάλυση των φασματικών περιεχομένων του ήχου, μέσω του μετασχηματισμού Fourier. Συνεπώς, για είσοδο 'tone_sequence.wav', λαμβάνουμε το ζητούμενο αποτέλεσμα, δηλαδή:
[0, 6, 2, 4, 0, 1, 7, 9]  (= 03120015 + 03120164)

*Ερώτημα 2.7*

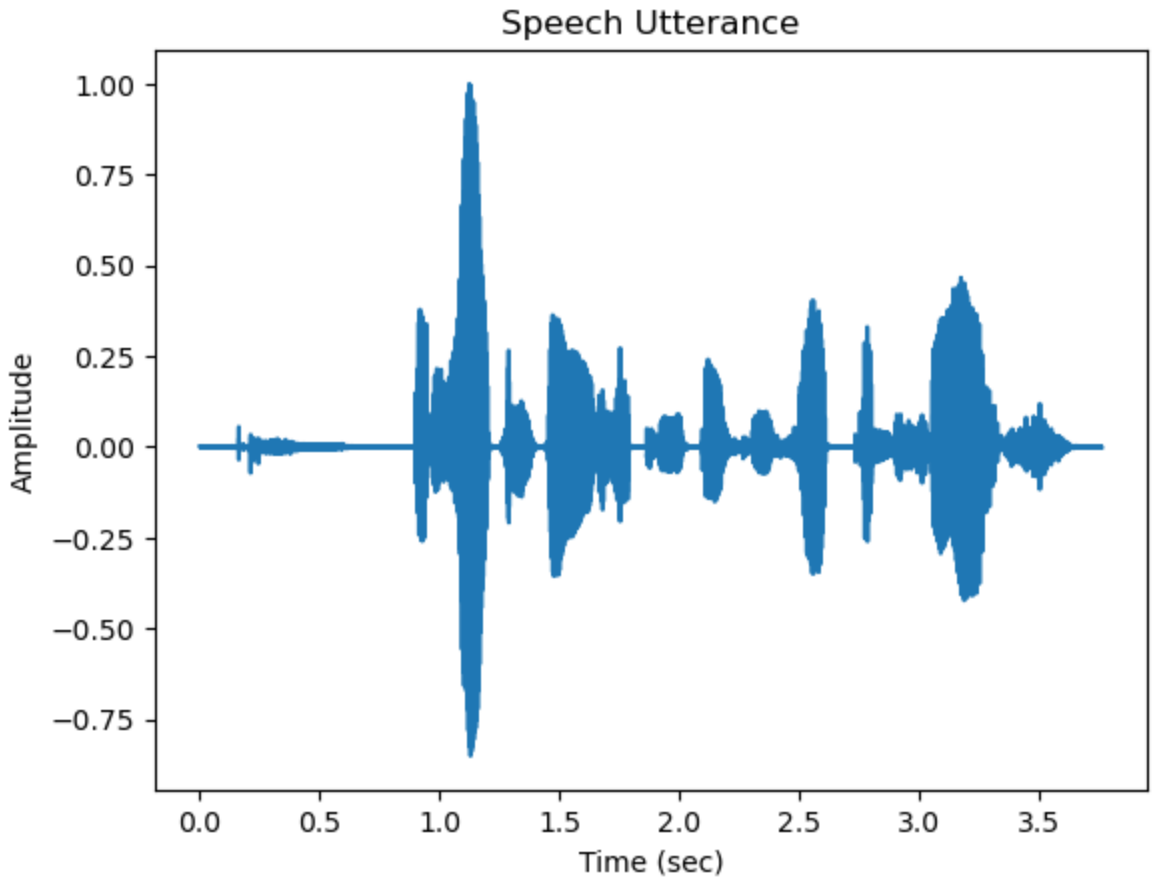
Η αποκωδικοποίηση των ζητούμενων σημάτων δεν συνιστά παρά την εφαρμογή της συνάρτησης ttdecode() σε καθένα από τα 3 σήματα, με μόνη διαφορά την ανάγκη φόρτωσης των δεδομένων του αρχείου .npy, μέσω της εντολής numpy.load. Έτσι, για κάθε σήμα λαμβάνουμε τον εξής Vector():

- Easy_sig : [3, 1, 4, 0, 4, 8, 1, 5]
- Medium_sig: [3, 2, 4, 8, 8, 2, 1, 0, 9, 6]
- Hard_sig: [2, 0, 4, 4, 9, 6, 3, 7, 6, 4]

## Μέρος 3ο - Χαρακτηριστικά Βραχέος Χρόνος Σημάτων Φωνής και Μουσικής

*Ερώτημα 3.1*

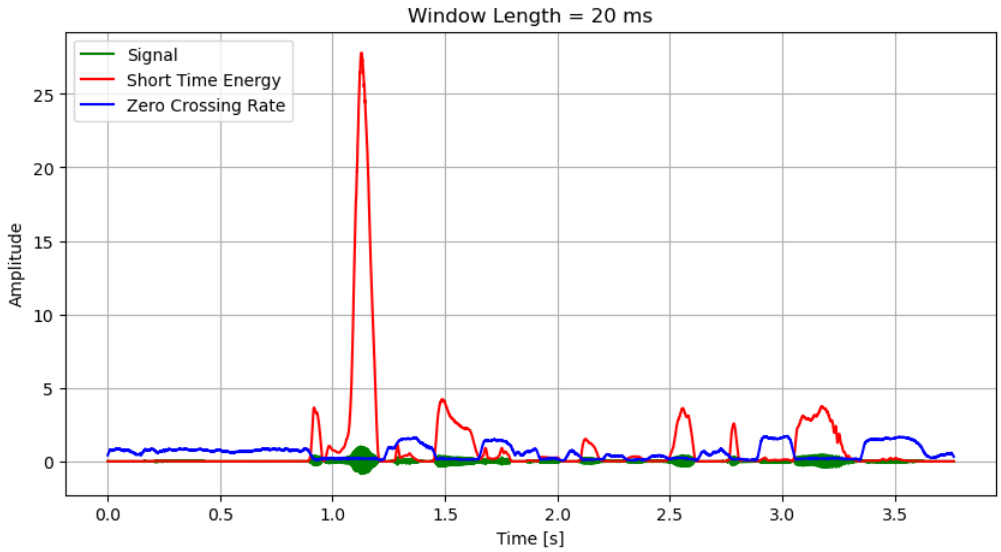
Για να απεικονίσουμε το δοθέν σήμα, απαιτείται πρωτίστως να διαβάσουμε το αρχείο ήχου, αξιοποιώντας τη συνάρτηση wavfile.read(). Έπειτα, μετατρέπουμε τα δεδομένα ήχου σε ένα numpy array και τα κανονικοποιούμε στην περιοχή [-1, 1]. Για τη γραφική απεικόνιση δημιουργούμε μια μεταβλητή time που αντιστοιχεί στον άξονα χρόνου του σήματος ήχου και, τέλος, καλούμε τη συνάρτηση plt.plot(). Από την παραπάνω διαδικασία, λαμβάνουμε το εξής αποτέλεσμα:



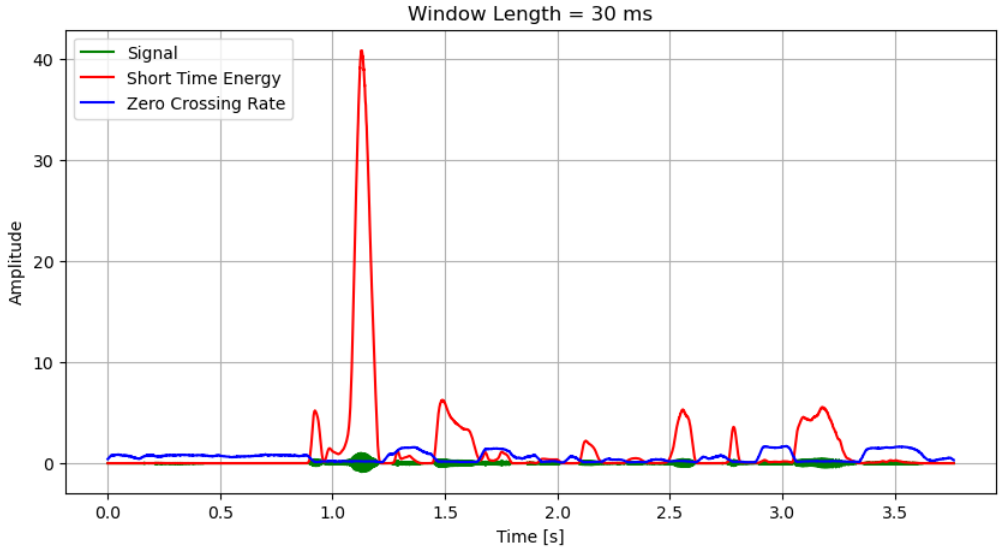

*Ερώτημα 3.2*

Στο ερώτημα αυτό, ζητείται η απεικόνιση της Ενέργειας Βραχέος Χρόνου (Short-Time Energy) και του Ρυθμού Εναλλαγής Προσήμου (Zero Crossing Rate) του σήματος του αρχείου ήχου ‘speech_utterance.wav’. Επιλέγουμε να απεικονίσουμε τα δύο αυτά γραφήματα ταυτόχρονα με το ίδιο το σήμα, ώστε η άντληση των πορισμάτων μας να διευκολύνεται μέσα από τη σύγκριση των γραφικών παραστάσεων. Έτσι, για καθένα από τα διαφορετικά μεγέθη παραθύρου Hamming, λαμβάνουμε τα παρακάτω αποτελέσματα:

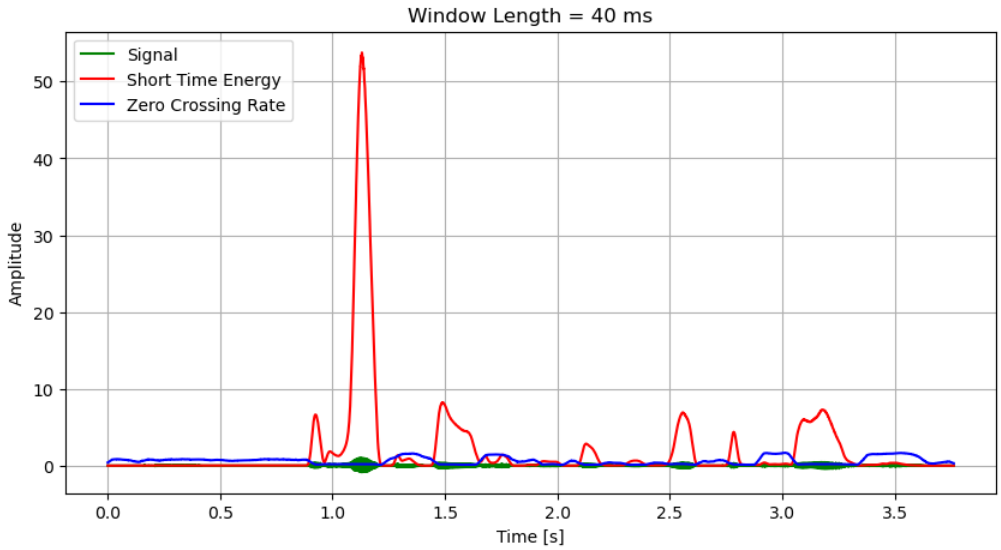
- Window length = 20ms:



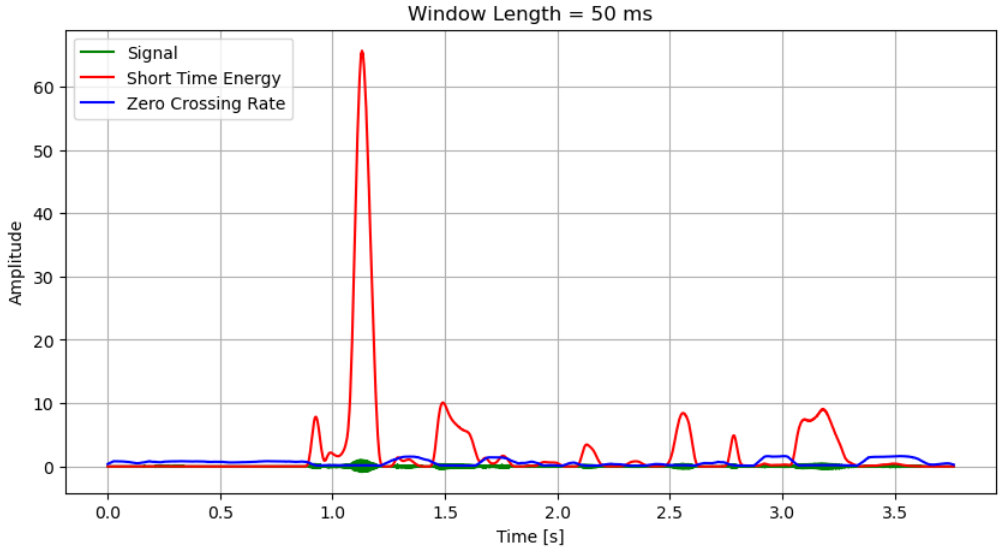

- Window length = 30ms:



- Window length = 40ms:

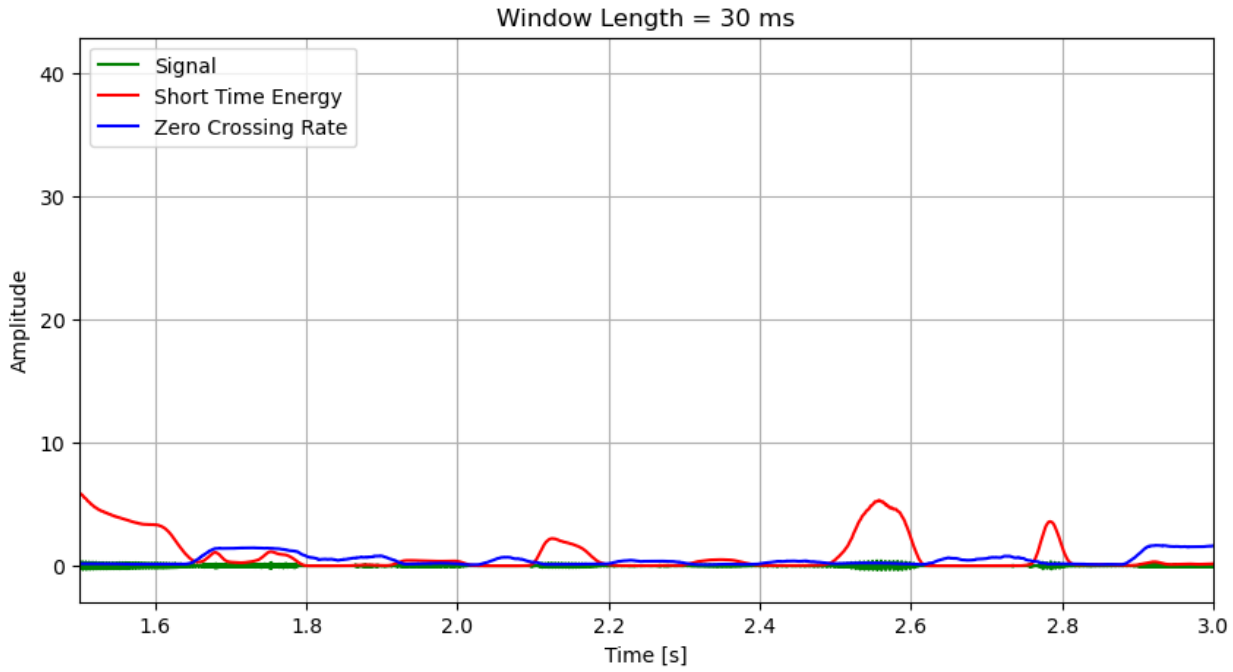


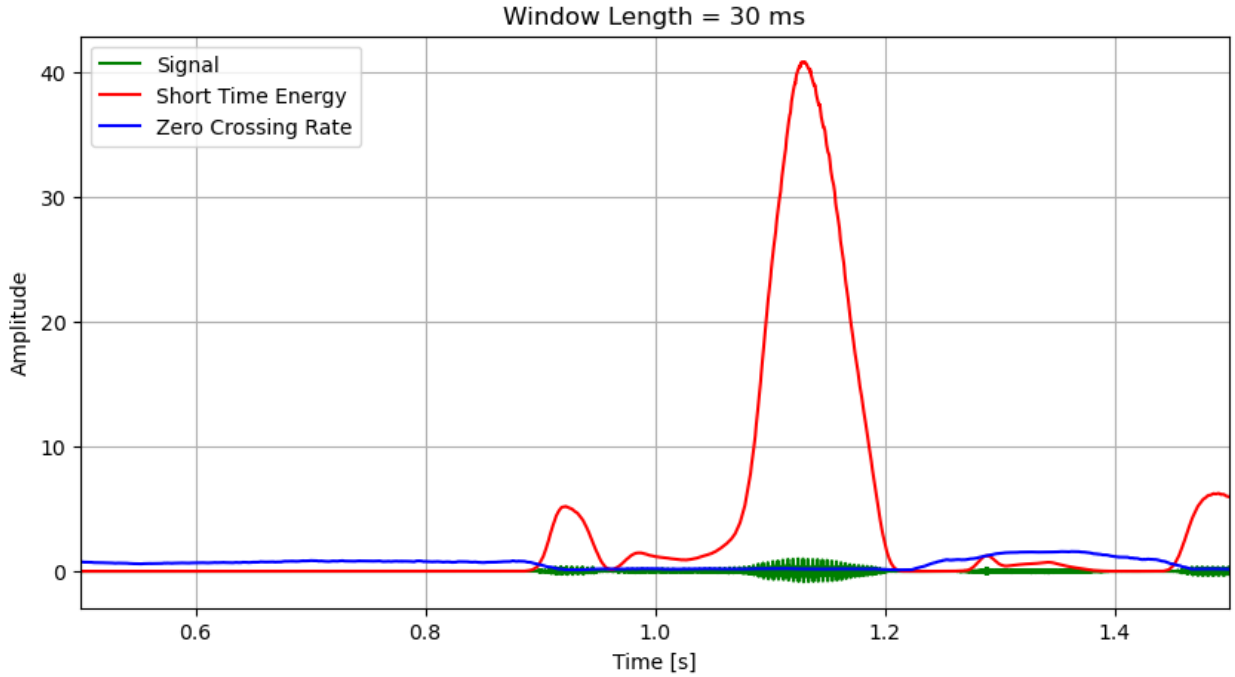

- Window length = 50ms:




Από τα παραπάνω διαγράμματα γίνεται εμφανές πως, όσο αυξάνεται το μήκος του Hamming Window, τόσο αυξάνεται και η Ενέργεια Βραχέος Χρόνου (STE). Πιο συγκεκριμένα, για window length 40ms, το μέγιστο πλάτος του STE κυμαίνεται στο 50-60, ενώ για window length 50ms το μέγιστο πλάτος ξεπερνά το 60. Το γεγονός αυτό, ωστόσο, δυσχεραίνει την ανάγνωση του σήματος, γι' αυτό κρίνεται σκόπιμη η επιλογή μιας μέσης τιμής παραθύρου (π.χ. 30 ms).

Επιλέγοντας την τιμή window length = 30 ms, κάνουμε zoom στα χρονικά διαστήματα 0.5-1.5 sec και 1.5-3 sec. Με τον τρόπο αυτό διευκολύνεται η ανάλυση του σήματος. Το φωνητικό σήμα είναι αρκετά "καθαρό", ώστε να μπορούμε να διακρίνουμε τις παύσεις στη φωνή καθώς έχουν πρακτικά μηδενική ενέργεια -η ενέργεια έχει πολύ μικρή τιμή λόγω θορύβου και γι' αυτό μπορεί να θεωρηθεί αμελητέα. Έτσι, λαμβάνουμε το παρακάτω διάγραμμα:




Ο Ρυθμός Εναλλαγής Προσήμου αποτελεί εργαλείο για τον διαχωρισμό των έμφωνων (voiced) ήχων, -*π.χ. /aa/,/ih/*- από τους άφωνους (unvoiced) ήχους -*π.χ. /f/,/p/*. Πιο συγκεκριμένα, οι έμφωνοι ήχοι έχουν μεγαλύτερη ενέργεια από τους άφωνους, γεγονός που αντικατοπτρίζεται και στο μεγαλύτερο πλάτος σήματος. Αντιθέτως, οι άφωνοι ήχοι έχουν μεγαλύτερο ZCR, λόγω της μεγαλύτερης συχνότητάς τους. Όμοια, και ο θόρυβος έχει μεγαλύτερη συχνότητα, άρα και μεγαλύτερο ZCR.

Στην περιοχές του φωνητικού σήματος όπου έχουμε έμφωνους ήχους, έχουμε μεγάλο STE και μικρό ZCR. Αυτό αποδίδεται στο γεγονός ότι οι έμφωνοι ήχοι, εκτός του ότι έχουν μικρή ενέργεια, περιλαμβάνουν και τα φωνήεντα τα οποία εμφανίζουν περιοδικότητα και μικρή συχνότητα, περνώντας πολύ αραιά από το 0.
Συνεπώς, αξιοποιώντας τη συχνότητα και το ZCR μπορούμε να διαχωρίσουμε τους έμφωνους ήχους από τους άφωνους, καθώς και να εντοπίσουμε τα φωνήεντα. Επιπλέον, το ZCR είναι ενδεικτικό για την αναγνώριση της περιοδικότητας και των άφωνων ήχων. Όλα τα προαναφερθέντα είναι χρήσιμα στην επεξεργασία του ήχου, στην αναγνώριση φωνητικών εντολών, ακόμα και στην ανίχνευση παθολογικών καταστάσεων στη φωνή.

*Ερώτημα 3.3*

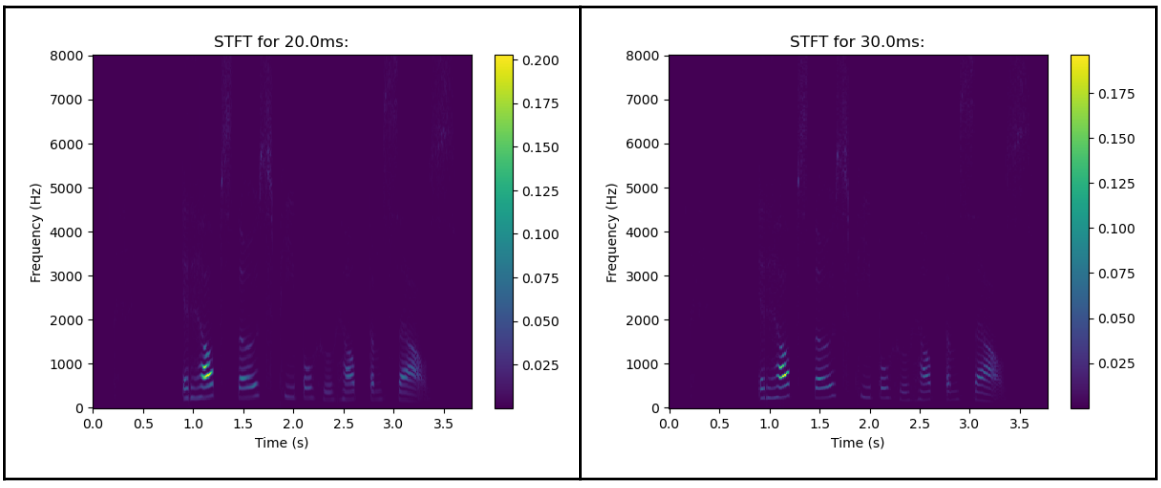
Σε αυτό το ερώτημα ζητείται η απεικόνιση του μετασχηματισμού Fourier βραχέος χρόνου του φωνητικού σήματος 'speech_utterance.wav'. Για τον σκοπό αυτό δημιουργούμε τη βοηθητική συνάρτηση stft, η οποία, κάνοντας χρήση της εντολής scipy.signal.stft(), επιστρέφει τον STFT του σήματος. Η συνάρτηση επίσης υπολογίζει το φασματικό κέντρο (SC) και τη φασματική ροή (SF) του STFT. Ο μετασχηματισμού Fourier βραχέος χρόνου του σήματος υπολογίζεται για διαφορετικά window lengths (20, 30, 40 και 50 ms), ενώ τα μεγέθη SF και SC υπολογίζονται από τον ίδιο τον μετασχηματισμό Fourier, αξιοποιώντας τις παρακάτω δοθείσες σχέσεις:

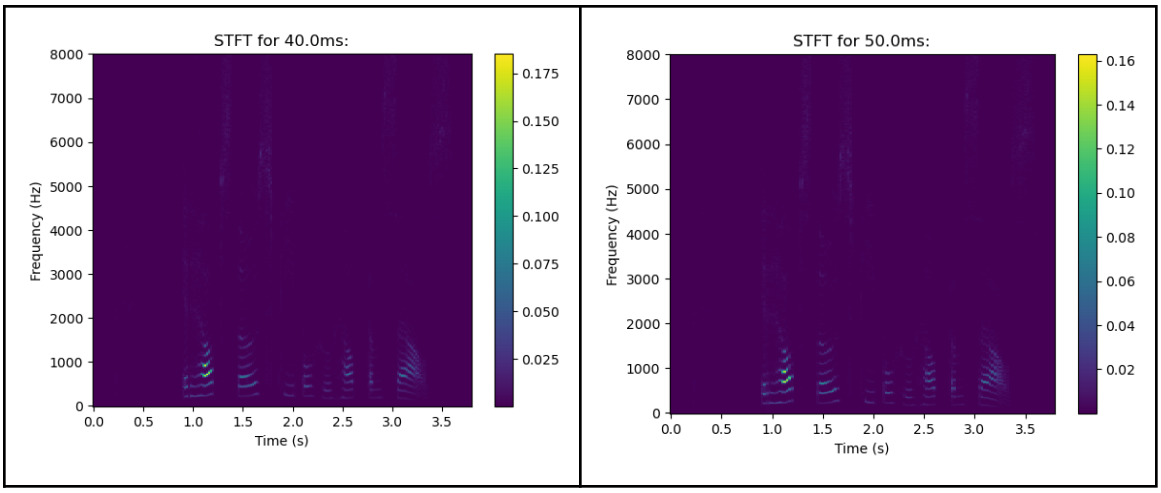
$$SCm = \frac{\sum_{k=0}^{N/2} k|X[k.m]|^2}{\sum_{k=0}^{N/2} |X[k.m]|^2} * \frac{f_s}{N} \text{ και}$$

$$SF = \left\| \frac{|X[k.m+1]|^2}{\sum_{k=0}^{N/2} |X[k.m+1]|^2} - \frac{|X[k.m]|^2}{\sum_{k=0}^{N/2} |X[k.m]|^2} \right\|_2, k = 0, 1, \dots, N/2$$

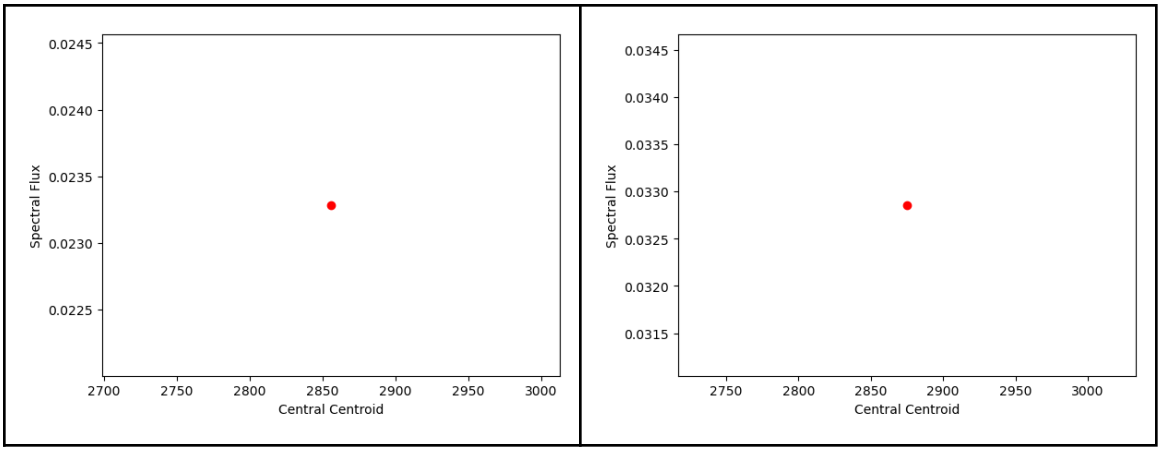
Έτσι, για διαφορετικά μεγέθη παραθύρου Hamming, λαμβάνουμε τα εξής αποτελέσματα:

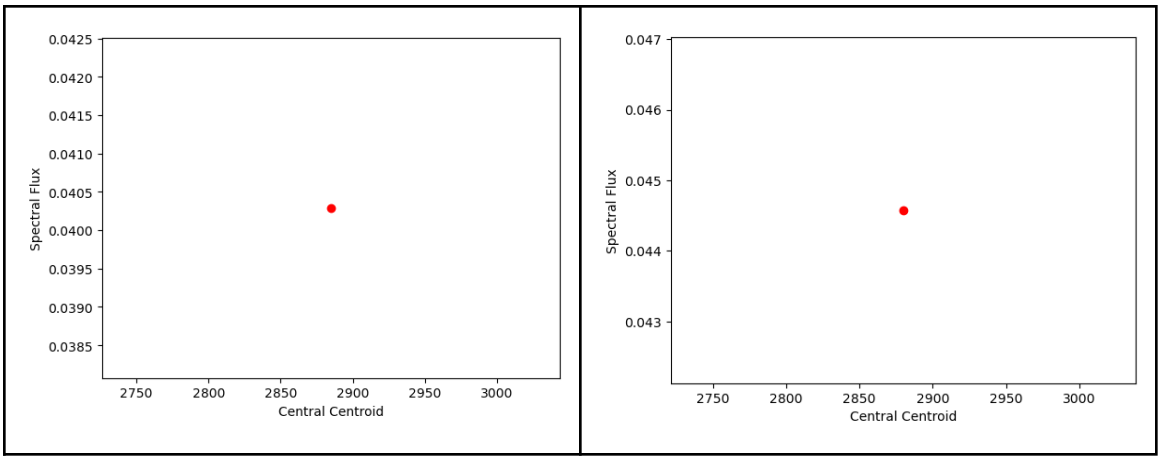
- Διαγράμματα Short-Time Fourier Transform of Audio Sample:






- Αποτελέσματα μετρήσεων Spectral Centroid και Spectral Flux από τους παραπάνω STFTs:
    - Central Centroid for 20.0 ms: 2855.725719381927
      Spectral Flux for 20.0 ms : 0.023283642781695316
    - Central Centroid for 30.0 ms: 2875.082429942771
      Spectral Flux for 30.0 ms : 0.032855073465938106
    - Central Centroid for 40.0 ms: 2884.888531169433
      Spectral Flux for 40.0 ms : 0.04028933780781899
    - Central Centroid for 50.0 ms: 2880.0713458515584
      Spectral Flux for 50.0 ms : 0.04457624950592827
- Διαγράμματα SC-SF:



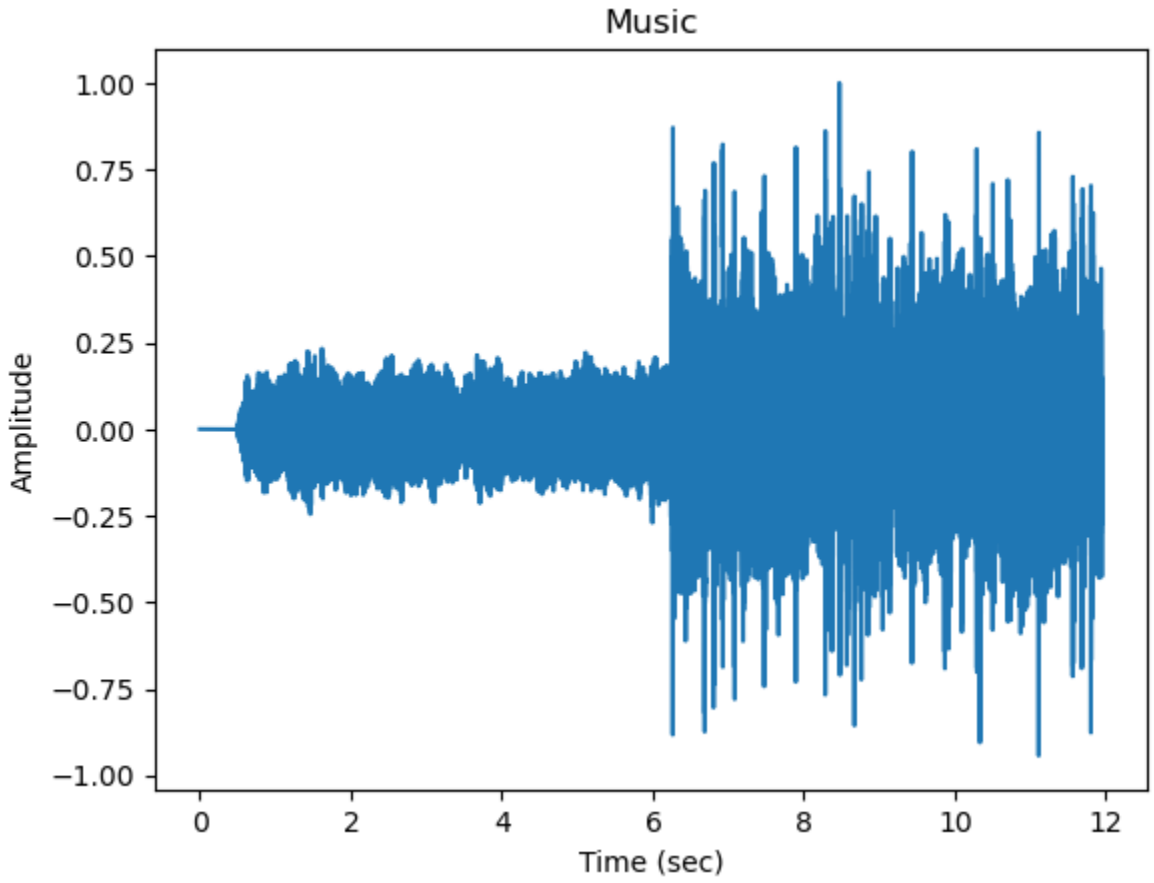



Παρατηρούμε πως οι εναλλαγές του μήκους του Hamming Window επηρεάζουν το διάγραμμα STFT με διάφορους τρόπους. Αρχικά, με την αύξηση του window length αυξάνεται και η ανάλυση των συχνοτήτων, δηλαδή οι εγγύτερες συχνότητες γίνονται πιο ευδιάκριτες. Παράλληλα, η αύξηση αυτή συμβάλλει στην καλύτερη εξομάλυνση της συχνότητας του STFT, δηλαδή ο STFT θα επηρεάζεται σε μικρότερο βαθμό από τυχόν θορύβους ή εξωτερικούς παράγοντες. Η μεγαλύτερη ομαλότητα των συχνοτήτων του STFT έχει ως αποτέλεσμα τη μείωση της τιμής του SC, καθώς η ομαλοποίηση μειώνει την επίδραση των υψηλών συχνοτήτων, οι οποίες τείνουν να αυξάνουν την τιμή του φασματικού κέντρου. Τέλος, αξίζει να σημειωθεί πως η αύξηση του μήκους παραθύρου μειώνει την ικανότητα του STFT να επεξεργαστεί γρήγορες αλλαγές στο σήμα ήχου, καθώς μειώνεται η χρονική του ανάλυση. Το γεγονός αυτό επηρεάζει το SF, μειώνοντας την τιμή του, καθώς η φασματική ροή μετρά το ποσό της φασματικής αλλαγής μεταξύ διαδοχικών χρονικών πλαισίων και μεγαλύτερα παράθυρα μπορεί να μην αποτυπώνουν τις γρήγορες αλλαγές του σήματος.

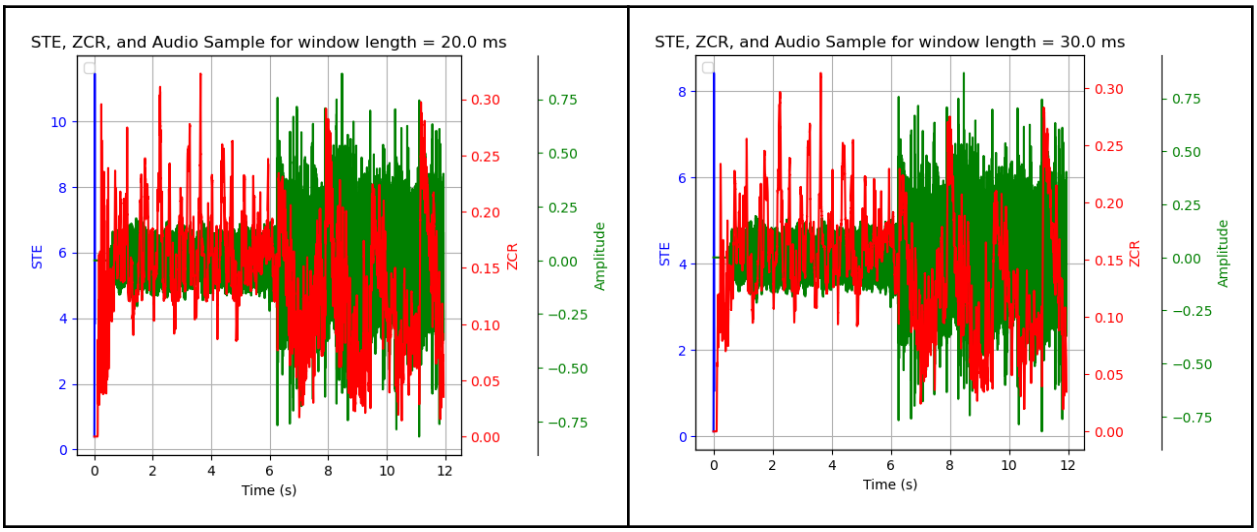
*Ερώτημα 3.4*

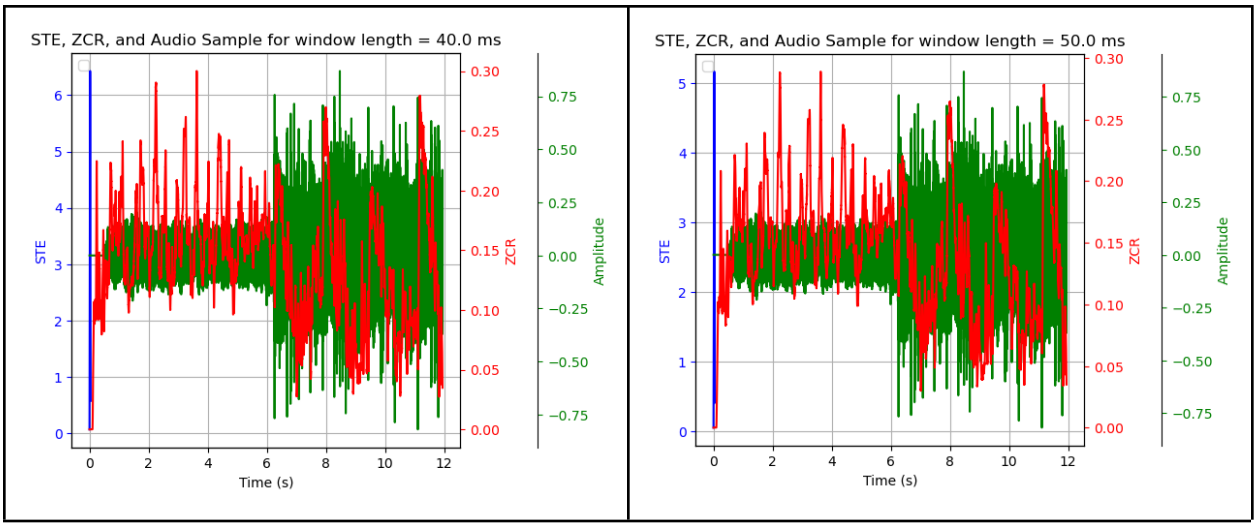
Εισάγουμε στον κώδικά μας το σήμα ήχου ‘music.wav’. Προκειμένου να επεξεργαστούμε το σήμα αυτό, θα πρέπει πρώτα να το μετατρέψουμε σε mono μορφή. Για τον σκοπό αυτό δημιουργούμε τη συνάρτηση stereo_to_mono, η οποία επιστρέφει το mono file ‘mono_audio.wav’. Στη συνέχεια, εκτελούμε την ίδια διαδικασία για καθένα από τα παραπάνω

ερωτήματα. Πρώτα, αποτυπώνουμε το σήμα ήχου ως διάγραμμα:




Έπειτα, όπως και στο ερώτημα 3.2, απεικονίζουμε στο ίδιο διάγραμμα τα μεγέθη Short-Time Energy και Zero Rate Crossing ταυτόχρονα με το ίδιο σήμα ήχου, για διαφορετικά μεγέθη Hamming Window:






Για τις ίδιες τιμές Hamming Window length, απεικονίζουμε και τον Short-Time Fourier Transform, με την ίδια διαδικασία που ακολουθήσαμε και στο ερώτημα 3.3: