

Τμήμα Τεχνών Ήχου και Εικόνας
Ιόνιο Πανεπιστήμιο

Μάθημα: Ψηφιακή Επεξεργασία Ήχου

Εργαστηριακή Άσκηση 2
«Εμβάθυνση σε θέματα ψηφιοποίησης του ήχου»

Διδάσκων: Φλώρος Ανδρέας
Δρ. Ηλ/γος Μηχ/κός & Τεχνολογίας Υπολογιστών

Κέρκυρα, 2013

Πίνακας περιεχομένων

1. Εισαγωγή.....	3
2. Θεωρία ψηφιοποίησης αναλογικών σημάτων.....	3
2.1. Δειγματοληψία – Διακριτοποίηση στο πεδίο του χρόνου	3
2.2. Κβαντισμός.....	3
2.3. Προβλήματα κατά την δειγματοληψία και τον κβαντισμό	4
3. Θέματα δειγματοληψίας.....	6
3.1. Επίδραση του sampling jitter στη βαθμίδα sample and hold	6
3.2. Δειγματοληψία και αναδίπλωση (aliasing)	7
3.3. Επίδραση της αναδίπλωσης στην Ψηφιακό-σε-Αναλογικό (Ψ/Α) μετατροπή	8
4. Θέματα κβαντισμού	9
4.1. Ο θόρυβος κβαντισμού.....	9
4.2. Η επίδραση του dither στον κβαντισμό.....	11
4.3. Κβαντισμός και ψαλιδισμός	12
5. Θέματα υπερδειγματοληψίας.....	13
5.1. Η επίδραση της υπερδειγματοληψίας στο φάσμα	13
5.2. Η εφαρμογή της μεθόδου της υπερδειγματοληψίας σε ψηφιακά σήματα	14
6. Υλοποίηση της άσκησης	15
7. Παράδοση της εργασίας.....	16
8. Σημαντικές παρατηρήσεις.....	17

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 2

«Εμβάθυνση σε θέματα ψηφιοποίησης του ήχου»

1. Εισαγωγή

Οι διαδικασίες που συνιστούν την ψηφιοποίηση των ηχητικών σημάτων, ήτοι η δειγματοληψία και ο κβαντισμός, εισάγουν υπό συνθήκες παραμορφώσεις οι οποίες σε αρκετές περιπτώσεις είναι ακουστές.

Σκοπός της παρούσας εργαστηριακής άσκησης είναι η επίδειξη της εμφάνισης και της ακουστότητας των παραπάνω παραμορφώσεων. Η άσκηση οργανώνεται σε δύο υποενότητες, εκ των οποίων η μία ασχολείται με τα φαινόμενα που σχετίζονται με την δειγματοληψία και η άλλη με τον κβαντισμό. Για τις ανάγκες της εργαστηριακής άσκησης θα χρησιμοποιηθεί ειδικό λογισμικό, συγκεκριμένα το περιβάλλον της *MATLAB*.

2. Θεωρία ψηφιοποίησης αναλογικών σημάτων

Όπως αναφέρθηκε και στην προηγούμενη εργαστηριακή άσκηση, η διαδικασία μετατροπής ενός αναλογικού σήματος σε ψηφιακό, καλείται ψηφιοποίηση. Κατά την ψηφιοποίηση, το συνεχούς χρόνου και πλάτους ακουστικό σήμα μετατρέπεται σε σήμα διακριτού χρόνου (έχει δηλαδή πεπερασμένο πλήθος τιμών στην μονάδα του χρόνου) και ταυτόχρονα το πλάτος του αντιστοιχίζεται σε πεπερασμένου πλήθους στάθμες. Η ανωτέρω μετατροπή πραγματοποιείται μέσω των διαδικασιών της δειγματοληψίας και του κβαντισμού αντίστοιχα.

2.1. Δειγματοληψία – Διακριτοποίηση στο πεδίο του χρόνου

Κατά τη διαδικασία της δειγματοληψίας το αρχικό αναλογικό σήμα μετατρέπεται σε σήμα διάκριτου χρόνου $s_d(nT_s)$, το οποίο ορίζεται σε πεπερασμένο πλήθος χρονικών στιγμών στην μονάδα του χρόνου (nT_s). Τα δείγματα του σήματος στις παραπάνω χρονικές στιγμές απέχουν ιδανικά μεταξύ τους ακριβώς τον ίδιο χρόνο (T_s). Υπό την προϋπόθεση τήρησης του θεωρήματος του Nyquist, το οποίο συζητήθηκε εκτενώς στην πρώτη εργαστηριακή άσκηση, δεν εμφανίζονται παραμορφώσεις στο σήμα λόγω της δειγματοληψίας.

2.2. Κβαντισμός

Όπως συζητήθηκε και στην προηγούμενη εργαστηριακή άσκηση, κβαντισμός είναι η διαδικασία κατά την οποία οι διαδοχικές τιμές της στάθμης (πλάτους) του σήματος διάκριτου χρόνου $s_d(nT_s)$ που προκύπτει κατά την δειγματοληψία μετατρέπονται σε διάκριτες (ψηφιακές) τιμές, πραγματοποιείται δηλαδή μια απεικόνιση της μορφής $s(nT_s) = Q[s_d(nT_s)]$, όπου $Q[s_d(nT_s)]$ είναι η κβαντισμένη τιμή στάθμης του σήματος. Κατά την παραπάνω μετατροπή προκύπτει ένα σφάλμα $e(n)$ το οποίο καλείται σφάλμα κβαντισμού. Το τελικό σήμα $s(nT_s)$, είναι πλέον διάκριτο τόσο ως προς τον χρόνο, όσο και ως προς το πλάτος και καλείται ψηφιακό.

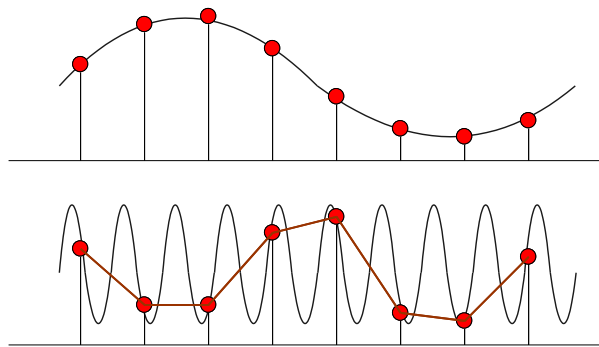
Είναι ευρύτερα γνωστό ότι τα αναλογικά σήματα της ακουστικής πίεσης ή αντίστοιχα του ηλεκτρικού σήματος που αντιστοιχεί στην πίεση αυτή, μετατρέπονται σε ψηφιακά με τη χρήση υποσυστημάτων μετατροπής Αναλογικού σε Ψηφιακό - Α/Ψ (Analog-to-Digital Converters, ADC's). Σε τέτοια υποσυστήματα μετατροπών, οι διαδικασίες κβαντισμού και δειγματοληψίας πραγματοποιούνται ταυτόχρονα. Το αντίστροφο επιτυγχάνεται με τη χρήση Ψηφιακού-σε-Αναλογικό - Ψ/Α μετατροπών (Digital-to-Analog Converters, DAC's), όπου χρονικά διάκριτα δείγματα και οι κβαντισμένες στάθμες μετατρέπονται (κατά προσέγγιση) σε συνεχείς αναλογικές

τιμές. Πρακτικά και θεωρητικά προβλήματα σε σχέση με την πιστότητα αναπαραγωγής των ακουστικών σημάτων εμφανίζονται υπό συνθήκες και στις δύο περιπτώσεις και συζητούνται στην επόμενη ενότητα.

2.3. Προβλήματα κατά την δειγματοληψία και τον κβαντισμό

Το διακοπτικό στοιχείο που χρησιμοποιείται για την υλοποίηση της διαδικασίας της δειγματοληψίας πρέπει να λειτουργεί χωρίς σφάλματα χρονισμού, ανοιγοκλείνοντας αυστηρά σε χρονικές στιγμές που είναι ακέραια πολλαπλάσια της περιόδου δειγματοληψίας T_s . Σε αντίθετη περίπτωση προκύπτει ακουστή παραμόρφωση. Τα παραπάνω σφάλματα χρονισμού ονομάζονται σφάλματα ολίσθησης δειγματοληψίας (jitter) και για τα ηχητικά συστήματα πρέπει να παραμένουν κάτω από το όριο του 1nsec. Στην πράξη, τα σφάλματα αυτά είναι υπαρκτά και δημιουργούν υπό συνθήκες ακουστές παραμορφώσεις, των οποίων τα χαρακτηριστικά εξαρτώνται από τα στατιστικά χαρακτηριστικά της χρονικής ολίσθησης jitter.

Κατά την δειγματοληψία ενός αναλογικού ακουστικού σήματος ενδέχεται επίσης η μέγιστη συχνότητα του καταγραφόμενου ήχου να είναι μεγαλύτερη από το μισό της εφαρμοζόμενης συχνότητας δειγματοληψίας. Στην περίπτωση αυτή, τα δείγματα διακριτού χρόνου που λαμβάνονται περιοδικά κατά τη φάση της δειγματοληψίας, δεν επαρκούν ώστε να «περιγράψουν» με ακρίβεια το αρχικό σήμα. Αντίθετα, το σήμα που προκύπτει μετά τη δειγματοληψία στην περίπτωση αυτή αντιστοιχεί σε ένα σήμα μικρότερης συχνότητας (Εικόνα 1). Το φαινόμενο αυτό του ελλιπούς ρυθμού δειγματοληψίας ονομάζεται αναδίπλωση (aliasing).



Εικόνα 1: Δειγματοληψία σύμφωνα με το θεώρημα του Nyquist (πάνω) και με συχνότητα δειγματοληψίας μικρότερη από το διπλάσιο της μέγιστης συχνότητας του καταγραφόμενου ήχου (κάτω)

Στην πράξη τα ακουστικά σήματα δεν περιορίζονται στη μέγιστη τιμή των 22.05kHz που απαιτεί η παραπάνω συχνότητα δειγματοληψίας, γι' αυτό και απαιτείται προ-φιλτράρισμα του αρχικού αναλογικού σήματος πριν οδηγηθεί στη βαθμίδα δειγματοληψίας, με στόχο την αποκοπή συχνοτήτων μεγαλύτερων από τη μέγιστη συχνότητα που στο συγκεκριμένο σύστημα ικανοποιεί τη συχνότητα δειγματοληψίας. Αυτό επιτυγχάνεται με τη χρήση χαμηλοδιαβατού (low-pass) φίλτρου, που συχνά ονομάζεται και φίλτρο anti-aliasing.

Από την άλλη μεριά, ο κβαντισμός είναι γενικά μία μη γραμμική διαδικασία καθώς, εάν τα δείγματα ενός σήματος διάκριτου χρόνου κβαντιστούν, η εύρεση των αρχικών τιμών πλάτους του σήματος είναι αδύνατη. Επιπλέον, είναι προφανές ότι η μετατροπή αυτή εμφανίζει σφάλμα το οποίο είναι δυνατόν να μετρηθεί, αλλά αποδεικνύεται ότι είναι και ακουστό. Αν θεωρήσουμε ότι κατά τον ιδανικό κβαντισμό ενός αναλογικού σήματος διάκριτου χρόνου $s_d(n)$ (θεωρούμε για

λόγους απλοποίησης ότι $T_s=1$) παράγεται ένα σήμα $s(n)$ και εμφανίζεται σφάλμα κβαντισμού $e(n)$, τότε:

$$s(n)=s_d(n)+e(n) \quad (1)$$

Η μορφή της συνάρτησης $e(n)$ έχει μεγάλη σημασία στην πιστότητα αναπαραγωγής ενός ψηφιακού ηχητικού συστήματος και γενικά εξαρτάται από την στιγμιαία τιμή πλάτους του σήματος $s_d(n)$ σε σχέση με το βήμα κβαντισμού Δ (άρα και την ευκρίνεια) του μετατροπέα. Ο θόρυβος κβαντισμού που παράγουν τα ψηφιακά ηχητικά έχει ιδιόζοντα και ανεπιθύμητο χαρακτήρα. Μια λύση που ακολουθείται για την αντιμετώπισή του κατά την αρχική Α/Ψ μετατροπή του μουσικού σήματος (ηχογράφηση) βασίζεται στην προσθήκη θορύβου χαμηλής στάθμης (dither), που έχει τιμές πλάτους μεγαλύτερες της στάθμης κβαντισμού Δ . Το αποτέλεσμα της προσθήκης αυτής, αναιρεί πολλά από τα προβλήματα που συζητήθηκαν παραπάνω, έτσι ώστε να εμφανίζεται ένα περισσότερο ικανοποιητικό ακουστικό αποτέλεσμα, αυξάνει όμως τη στάθμη του θορύβου, ο οποίος έχει πλέον τον επιθυμητό χαρακτήρα του λευκού θορύβου.

Συγκεκριμένα, η προσθήκη του dither στην είσοδο ενός κβαντιστή προκαλεί μια σειρά από σημαντικές βελτιώσεις στη συνολική λειτουργία του: (α) αλλάζει την μορφή του σφάλματος κβαντισμού και το καθιστά ανεξάρτητο από το σήμα εισόδου, (β) αφαιρεί κάθε είδους αρμονικής παραμόρφωσης η οποία εμφανίζεται κατά την μετατροπή σημάτων πολύ μικρού πλάτους, δίνοντας στον θόρυβο κβαντισμού χαρακτηριστικά όμοια με τα χαρακτηριστικά λευκού θορύβου και (γ) βελτιώνει τη διακριτική ικανότητα του κβαντιστή, αυξάνοντας κατά υποκειμενικό (κι όχι ποσοτικό) τρόπο τη δυναμική του περιοχής. Γενικά, η εξομάλυνση των παραμορφώσεων λόγω κβαντισμού με προσθήκη dither θα πρέπει να υλοποιείται σε όλες τις επεξεργασίες του ψηφιακού σήματος όπου απαιτείται επανακβαντισμός ή και αριθμητική αποκοπή του (truncation), όπως για παράδειγμα κατά την μεταβολή της στάθμης ή το ψηφιακό φιλτράρισμα του σήματος σε μία ψηφιακή συσκευή.

Με βάση τα παραπάνω, η χρήση dither έχει σαν αποτέλεσμα την αποδιαμόρφωση του σφάλματος κβαντισμού από το σήμα εισόδου και την μείωση της αρμονικής παραμόρφωσης που δημιουργείται κατά τον κβαντισμό σημάτων μικρού πλάτους, με ταυτόχρονη αύξηση το επιπέδου του θορύβου κβαντισμού. Παρόλο που σε τυπικές ηχητικές εφαρμογές (κβαντισμός 16bit, συχνότητα δειγματοληψίας 44.1kHz) η στάθμη θορύβου είναι αρκετά χαμηλή (περίπου ίση με -124dB) κι έχει χαρακτηριστικά λευκού θορύβου, εντούτοις για σήματα χαμηλής έντασης, το επίπεδο του θορύβου ξεπερνά το ελάχιστο κατώφλι ακοής και ο θόρυβος κβαντισμού καθίσταται ακουστός. Η ακουστότητα του θορύβου αυξάνει λόγω της μεταβλητής ευαισθησίας του ανθρώπινου ακουστικού οργάνου με τη συχνότητα, με αποτέλεσμα ο θόρυβος κβαντισμού να είναι περισσότερο ακουστός από τη συχνότητα των 4kHz και πάνω. Για το λόγο αυτό, σε τυπικές ηχητικές εφαρμογές, ο θόρυβος κβαντισμού μορφοποιείται κατάλληλα στο πεδίο της συχνότητας, χρησιμοποιώντας τεχνικές μορφοποίησης θορύβου (Noise-Shaping). Οι τεχνικές αυτές υλοποιούνται μέσω ανατροφοδοτούμενων ψηφιακών φίλτρων που λειτουργούν ως κλάδοι ανατροφοδότησης γύρω από έναν κβαντιστή.

Πολλά τεχνικά προβλήματα κατά την μετατροπή του σήματος αντιμετωπίζονται με τη μέθοδο της υπερδειγματοληψίας (oversampling), η οποία, επιτρέπει τη χρήση κβαντιστών χαμηλότερης τάξης (π.χ. $N=2\text{bit}$) χωρίς αντίστοιχη ελάττωση της δυναμικής περιοχής λειτουργίας που συνεπάγεται μια τέτοια μείωση τάξης. Αποδεικνύεται θεωρητικά ότι διπλασιασμός του παράγοντα υπερδειγματοληψίας R ισοδυναμεί με αύξηση της τάξης του αντίστοιχου κβαντιστή κατά μισό bit. Η υλοποίηση της υπερδειγματοληψίας επιτυγχάνεται πάντοτε με χρήση ψηφιακών φίλτρων, τα οποία πρέπει να εμφανίζουν μικρές παραμορφώσεις φάσης (συχνά είναι φίλτρα γραμμικής

φάσης - linear phase filters), όροι που συχνά εμφανίζονται ακόμη και σε οικιακά συστήματα CD. Τα ψηφιακά αυτά φίλτρα είναι πεπερασμένης απόκρισης (Finite Impulse Response - FIR), τάξης από 10 μέχρι 256.

3. Θέματα δειγματοληψίας

Στόχος της ενότητας αυτής είναι η παρουσίαση και η κατανόηση των εννοιών που σχετίζονται με τη δειγματοληψία αναλογικών ηχητικών σημάτων, όπως του sampling jitter (ολίσθησης χρονικής στιγμής δειγματοληψίας), της αναδίπλωσης (aliasing) κ.λπ. καθώς και της πρακτικής εφαρμογής του θεωρήματος δειγματοληψίας του Nyquist. Επιπλέον, το φαινόμενο της αναδίπλωσης εξετάζεται και στην περίπτωση της μετατροπής από ψηφιακό-σε-αναλογικό (Ψ/Α), με στόχο την κατανόηση της αναγκαιότητας ύπαρξης reconstruction φίλτρων σε τέτοιες διατάξεις.

3.1. Επίδραση του sampling jitter στη βαθμίδα sample and hold

Βήμα 1: Δημιουργείστε μια ημιτονική κυματομορφή διάκριτου χρόνου συχνότητας 1kHz και πλάτους ίσου με 1, αφού πρώτα ορίσετε τη συχνότητα δειγματοληψίας ίση προς 44.1kHz. Ο συνολικός αριθμός των σημείων δειγματοληψίας θα πρέπει να είναι ίσος προς 441. Σημειώστε ότι θα πρέπει πρώτα να ορίσετε τις χρονικές τιμές δειγματοληψίας σε ένα διάνυσμα χρόνου και στη συνέχεια να παράγετε το ημιτονικό σήμα με βάση το χρόνο αυτό. Επίσης, προσέξτε ότι εφόσον δεν κβαντίζουμε το σήμα, η δυναμική του περιοχή θα είναι ίση προς τη δυναμική περιοχή της αριθμητικής αναπαράστασης της MATLAB.

```
» npoints=441;           % Define the total number of sampling points
» fs=44100;             % Define the sampling frequency (in Hz)
» t=[0:npoints-1]/fs;  % Jitter-free sampling time values
» s=sin(2*pi*t*1000);   % Create the 1kHz sinewave signal s
```

Βήμα 2: Δημιουργείστε και πάλι μια ημιτονική κυματομορφή διάκριτου χρόνου συχνότητας 1kHz και πλάτους ίσου με 1, αυτή τη φορά εισάγοντας τυχαίο (random) jitter στις χρονικές στιγμές δειγματοληψίας μέγιστου πλάτους $\pm T_s/2$, όπου T_s η περίοδος δειγματοληψίας.

```
» Ts=1/fs;             % The sampling period (in seconds)
» MaxJitter=Ts/2;      % The maximum jitter size
» tj=t+(2*rand(1,npoints)-1)*MaxJitter;
» sjitter=sin(2*pi*tj*1000);
```

Βήμα 3: Τυπώστε σε κοινό διάγραμμα τη μορφή του καθενός από τα δύο παραπάνω ημιτονικά σήματα ως συνάρτηση του χρόνου. Για λόγους καλύτερης απεικόνισης, επιλέξτε να τυπώσετε μόνο τα πρώτα 50 δείγματα από το κάθε σήμα. Σημειώστε ότι με τη χρήση της εντολής hold on μπορείτε να τυπώσετε περισσότερες από μια κυματομορφές στο ίδιο διάγραμμα της MATLAB.

```
» nplot=50;           % The number of points to plot
» plot(t(1:nplot),s(1:nplot),'b-')
» hold on
» plot(t(1:nplot),sjitter(1:nplot),'-r+')
» xlabel('Time (seconds)')
```

```

» ylabel('Normalized amplitude')
» legend('No timing errors','With timing errors')

```

Βήμα 4: Υπολογίστε τα φάσματα που αντιστοιχούν σε κάθε ένα από τα παραπάνω σήματα χρησιμοποιώντας την έτοιμη συνάρτηση `CalculateSpectrum`. Στο παράθυρο διαλόγου με τίτλο “Digital Signal Spectrum Calculation” που θα εμφανιστεί, δώστε την τιμή 1000 στο πεδίο “Enter the Input sinewave frequency in Hz (0 for non-tone input):”. Με τον τρόπο αυτό θα υπολογιστεί η αρμονική παραμόρφωση για κάθε περίπτωση ημιτονικού σήματος ξεχωριστά στο εμφανιζόμενο διάγραμμα. Τι συμπεράσματα εξαγάγετε από τη μορφή των δύο φασμάτων και από την τιμή της αρμονικής παραμόρφωσης;

```

» [NoJitterSpectrum,f]=CalculateSpectrum(s);
» [JitteredSpectrum,f]=CalculateSpectrum(sjitter);

```

Βήμα 5: Εκτελέστε ξανά τα παραπάνω βήματα δημιουργώντας όχι τυχαίο, αλλά περιοδικό χρονικό jitter συχνότητας 500Hz και πλάτους $\pm T_s/2$. Αλλάξτε τον τρόπο υπολογισμού του χρονικού jitter όπως παρακάτω, και εκτελέστε πάλι τον κώδικα που δόθηκε στα προηγούμενα βήματα. Ποιες οι διαφορές στο χρόνο και στη συχνότητα σε σχέση με την περίπτωση του τυχαίου jitter;

```

» jitter_frequency=500;
» tj=t+MaxJitter*sin(2*pi*jitter_frequency*t);

```

Βήμα 6: Εκτελέστε ξανά τα παραπάνω βήματα (για τυχαίο και περιοδικό jitter με τα χαρακτηριστικά που αναφέρθηκαν προηγουμένως) για συνολική διάρκεια ημιτονικού σήματος ίση προς 5 δευτερόλεπτα (`npoints=220500`). Μην υπολογίσετε τις φασματικές εκφράσεις των φασμάτων, διότι λόγω πλήθους δειγμάτων, η διαδικασία ενδέχεται να μην ολοκληρωθεί εξαιτίας ανεπάρκειας πόρων του συστήματος. Αποθηκεύστε το σήμα διάκριτου χρόνου `sjitter` με τυχαίο και περιοδικό `sampling jitter` σε αρχείο τύπου `.wav` με ακρίβεια κβαντισμού 16bit. (Σημείωση: μην επανακβαντίσετε το σήμα, ο κβαντισμός θα γίνει αυτόματα από την MATLAB κατά την αποθήκευση του αρχείου ήχου).

```

» wavwrite(sjitter,44100,16,'d:\JitteredSignal.wav');

```

3.2. Δειγματοληψία και αναδίπλωση (aliasing)

Βήμα 1: Δημιουργήστε ένα ημιτονικό σήμα διάκριτου χρόνου με συχνότητα `fin=5000Hz` και πλάτος 1 (peak-to-peak). Ορίστε τη συχνότητα δειγματοληψίας ίση προς 44.1kHz και το πλήθος των δειγμάτων ίσο προς τον αριθμό των δειγμάτων που εμπεριέχονται σε μια θεμελιώδη περίοδο.

```

» fs=44100; % The sampling frequency (in Hz)
» fin=5000; % The sinewave frequency (in Hz)
» [c,npoints,fo]=find_fo(fs,fin); % Calculate the number of samples
» t=[0:npoints-1]/fs; % Create the time sampling vector
» s=cos(2*pi*fin*t); % Create the sinusoidal wave

```

Βήμα 2: Τυπώστε τη μορφή του ημιτονικού σήματος που δημιουργήσατε ως συνάρτηση του χρόνου. Για λόγους ευκρίνειας της αναπαράστασης, τυπώστε τα πρώτα 64 σημεία μόνο.


```

» plot(t(1:64),s(1:64),'b-')           % Plot the first 64 samples
» xlabel('Time (seconds)')
» ylabel('Amplitude')
» title('Sinewave plot')

```

Βήμα 3: Υπολογίστε και απεικονίστε το φάσμα του ημιτονικού σήματος χρησιμοποιώντας την έτοιμη συνάρτηση `CalculateSpectrum`.

```

» [SignalSpectrum,f]=CalculateSpectrum(s);

```

Βήμα 4: Επαναλάβετε τα παραπάνω βήματα αλλάζοντας κάθε φορά τη συχνότητα του ημιτονικού σήματος εξόδου. Χρησιμοποιήστε τις τιμές `fin=15000, 20000, 25000` Hz. Τι παρατηρείτε (ειδικά στην περίπτωση όπου `fin=25kHz`); Σημειώστε ότι για λόγους ευκολίας, μπορείτε να αποθηκεύσετε τον κώδικά σας σε ένα αρχείο με κατάληξη `.m` σε κάποιον φάκελο που έχει δηλωθεί στο `path` της MATLAB. Με τον τρόπο αυτό, αρκεί κάθε φορά να αλλάζετε την τιμή της παραμέτρου `fin` πριν την εκτέλεση του συγκεκριμένου αρχείου. Π.χ., εάν αποθηκεύσετε τον κώδικά σας στο αρχείο `TestAliasing.m`, αρκεί να εκτελέσετε την εντολή:

```

» TestAliasing;

```

3.3. Επίδραση της αναδίπλωσης στην Ψηφιακό-σε-Αναλογικό (Ψ/Α) μετατροπή

Βήμα 1: Ορίστε τη βασική συχνότητα δειγματοληψίας ίση προς 16000Hz.

```

» fs=16000;

```

Βήμα 2: Δημιουργείστε ένα διάνυσμα τιμών συχνότητας και ένα δεύτερο χρονικής διάρκειας που αντιστοιχούν στις νότες ενός μουσικού κομματιού.

```

» f=[1319 1319 1397 1568 1568 1397 1319 1175 1047 1047 1175 1319 1319
1175 1175];
» dur=[0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.4 0.6 0.2 0.5];

```

Βήμα 3: Δημιουργείστε ένα στιγμιότυπο του μουσικού κομματιού που αντιστοιχεί στις νότες που ορίσατε προηγουμένως, αποθηκεύοντάς το σε ένα νέο διάνυσμα με όνομα `music`.

```

» music=[];
» for i=1:length(f),
»     new_note=note(f(i), dur(i), fs);
»     music=[music new_note];
» end

```

Βήμα 4: Αναπαράγετε το μουσικό κομμάτι που δημιουργήσατε από την κάρτα ήχου του υπολογιστή, κάνοντας χρήση της συνάρτησης `soundsc` της MATLAB. Προσέξτε ότι το δεύτερο όρισμα εισόδου της συνάρτησης αυτής είναι η συχνότητα δειγματοληψίας, η οποία για τη συγκεκριμένη περίπτωση είναι ίση προς 16kHz.

```

» soundsc(music, fs);

```

Βήμα 5: Υποδειγματοληπτίστε (`downsample`) το μουσικό κομμάτι `music` με παράγοντα `dsr=3` (σημείωση: η υποδειγματοληψία να γίνει χωρίς τη χρήση χαμηλοδιαβατού φίλτρου, απλά

δημιουργώντας ένα νέο σήμα `music_` με δείγματα ίδια με τα δείγματα του αρχικού σήματος `music` ανά 3).

```
» dsr=3;
» music_ = music(1:dsr:length(music));
```

Βήμα 6: Αναπαράγετε το νέο μουσικό κομμάτι `music_` με χρήση της συνάρτησης `soundsc` της MATLAB. Προσέξτε ότι η τιμή του δεύτερου ορίσματος εισόδου (συχνότητας δειγματοληψίας) είναι ίση προς fs/dsr , καθώς η συχνότητα δειγματοληψίας του σήματος `music_` είναι ίση με το $1/dsr$ της αρχικής. Πώς ακούγεται το νέο σήμα σε σχέση με το αρχικό; Είναι ίδια ή διαφορετικά; Μπορείτε να προσδιορίσετε «ακουστικά» την επίδραση του aliasing;

```
» soundsc(music_, fs/dsr);
```

Βήμα 7: Επαναλάβετε τη διαδικασία της υποδειγματοληψίας του αρχικού μουσικού κομματιού `music`, φιλτράροντας προηγουμένως το σήμα με χρήση ενός κατάλληλου χαμηλοδιαβατού FIR φίλτρου. Για το σκοπό αυτό, δημιουργήστε πρώτα ένα τέτοιο φίλτρο κάνοντας χρήση της συνάρτησης `fir1` της MATLAB. Σημειώνεται ότι περισσότερες λεπτομέρειες σχετικά με FIR φίλτρα θα διδαχθούν αργότερα στο πλαίσιο του μαθήματος.

```
» filter_coeff=fir1(64,1/dsr);
```

Βήμα 8: Φιλτράρετε το αρχικό σήμα `music` κάνοντας με το παραπάνω φίλτρο.

```
» music_filtered=filter(filter_coeff,1,music);
```

Βήμα 9: Υποδειγματοληψίστε το μουσικό κομμάτι `music_filtered` με παράγοντα `dsr=3`.

```
» music_filtered_ = music_filtered(1:dsr:length(music_filtered));
```

Βήμα 10: Αναπαράγετε το μουσικό κομμάτι `music_filtered_` και εντοπίστε τις ακουστές διαφορές σε σχέση με το μουσικό κομμάτι `music_`.

```
» soundsc(music_filtered_, fs/dsr);
```

4. Θέματα κβαντισμού

Στόχος της ενότητας αυτής είναι η κατανόηση εννοιών που σχετίζονται με τον κβαντισμό ηχητικών σημάτων, όπως η ευκρίνεια κβαντισμού, το σφάλμα κβαντισμού, η δυναμική περιοχή κ.λ.π., καθώς επίσης και η κατανόηση της επίδρασης του θορύβου dither και των τεχνικών μορφοποίησης θορύβου (noise-shaping) τόσο στο πεδίο του χρόνου, όσο και στο πεδίο της συχνότητας.

4.1. Ο θόρυβος κβαντισμού

Βήμα 1: Διαβάστε από το δίσκο το αρχείο `speech.wav` που βρίσκεται στο φάκελο `DATToolbox`. Αλλάξτε τη διαδρομή του αρχείου εάν αυτό είναι απαραίτητο, ανάλογα με τη διαδρομή στην οποία το έχετε αποθηκεύσει στον υπολογιστή σας.

```
» [speech, fs, N]=wavread('d:\DATToolbox\speech.wav');
```

Βήμα 2: Δείτε τη μορφή του σήματος που φορτώσατε στο χρόνο (waveform plot).

```
» plot(speech)
```

Βήμα 3: Παρατηρείστε ότι ο άξονας x του διαγράμματος που εμφανίστηκε έχει μονάδες δείγματα (samples). Δημιουργήστε το κατάλληλο χρονικό «διάνυσμα» το οποίο περιέχει τις ακριβείς χρονικές στιγμές δειγματοληψίας του ψηφιακού σήματος, λαμβάνοντας υπόψιν ότι η συχνότητα δειγματοληψίας είναι ίση με f_s (Hz) και θεωρώντας αρχική τιμή χρόνου την $t=0$.

```
» t=[0:1:length(speech)-1]/fs;
```

Βήμα 4: Τυπώστε ξανά το σήμα που φορτώσατε ως συνάρτηση του χρόνου.

```
» plot(t, speech)
```

Βήμα 5: Ελέγξτε την ελάχιστη και μέγιστη τιμή του σήματος εισόδου. Θυμηθείτε ότι η απουσία ελληνικού ερωτηματικού στο τέλος της κάθε δήλωσης, σημαίνει άμεση εκτύπωση στην οθόνη του αντίστοιχου αποτελέσματος/τιμής.

```
» max(speech)
```

```
» min(speech)
```

Βήμα 6: Μετατρέψτε τις τιμές του σήματος εισόδου σε ακέραιες. Επειδή το σήμα ήταν αποθηκευμένο σε αρχείο .wav, είναι ήδη κβαντισμένο (με ευκρίνεια κβαντισμού 16bit), αρκεί ο πολλαπλασιασμός των τιμών του με την μέγιστη (απολύτως) τιμή της αναπαράστασης προσημασμένων ακεραίων μήκους 2byte.

```
» speech=speech*32768;
```

Βήμα 7: Ελέξτε και πάλι την ελάχιστη και τη μέγιστη τιμή του σήματος εισόδου.

```
» max(speech)
```

```
» min(speech)
```

Βήμα 8: Επανακβαντίστε το σήμα εισόδου `speech` σε ανάλυση 8bit, χωρίς χρήση dither και τεχνικής μορφοποίησης θορύβου.

```
» RequantisedSpeech=QuantiseAudio(speech, 8, 32767, -32768, 0, 0);
```

Βήμα 9: Τυπώστε το επανακβαντισμένο σήμα συνάρτηση του χρόνου.

```
» plot(t, RequantisedSpeech)
```

Βήμα 10: Συγκρίνετε τις κυματομορφές των δύο σημάτων (16 και 8bit). Τυπώστε το αρχικό (16bit) σήμα ως συνάρτηση του χρόνου με διαφορετικό χρώμα, διατηρώντας το διάγραμμα του 8bit σήματος.

```
» hold on
```

```
» plot(t, speech, 'c-')
```

Βήμα 11: Υπολογίστε το θόρυβο κβαντισμού ως τη διαφορά μεταξύ των σημάτων 8 και 16 bit.

```
» e= RequantisedSpeech-speech;
```

Βήμα 12: Τυπώστε το θόρυβο κβαντισμού σα συνάρτηση του χρόνου.

```
» hold off
```

```
» plot(t, e);
```

Βήμα 13: Αποθηκεύστε το θόρυβο κβαντισμού σε αρχείο τύπου .wav με 16bit ακρίβεια. Αλλάξτε τη διαδρομή αποθήκευσης εάν το επιθυμείτε.

```
» wavwrite(e/32768,44100,16,'d:\QuantError8bit.wav');
```

Βήμα 14: Τέλος, αποθηκεύστε και το επανακβαντισμένο σήμα σε αρχείο τύπου .wav. Η ακρίβεια αποθήκευσης μπορεί να είναι 16 ή 8 bit, χωρίς αλλοίωση του ακουστού αποτελέσματος. Μπορείτε να εξηγήσετε το γιατί;

```
» wavwrite(RequantisedSpeech/32768,44100,16,'d:\Requantised8bit.wav');
```

Βήμα 15: Επαναλάβετε όλα τα παραπάνω στάδια: α) κάνοντας χρήση RPDF dither και noise-shaping 3ης τάξης κατά τον επανακβαντισμό από 16 σε 8bit β) για ευκρίνεια επανακβαντισμού ίση με 4bit με και χωρίς χρήση dither και noise shaping 3ης τάξης. Αποθηκεύστε τα αρχεία τύπου .wav που προκύπτουν σε όλες τις παραπάνω περιπτώσεις με διαφορετικά, κατάλληλα ονόματα. Για την εφαρμογή dither και noise-shaping, θέστε κατάλληλα τις παραμέτρους NSFlag και DitherFlag κατά την κλήση της συνάρτησης QuantiseAudio που υλοποιεί τον επανακβαντισμό του σήματος.

4.2. Η επίδραση του dither στον κβαντισμό

Βήμα 1: Δημιουργείστε ένα ημιτονικό σήμα πλάτους -60dBFS και συχνότητας 1kHz. Η συχνότητα δειγματοληψίας να επιλεγεί ίση προς 44.1kHz και η ευκρίνεια κβαντισμού 16bit.

```
» Sine16=DigitalSineWave(0.001,1000,1);
```

Βήμα 2: Υπολογίστε το φάσμα του ψηφιακού ημιτονικού σήματος που δημιουργήσατε.

```
» [Spectrum16,f]=CalculateSpectrum(Sine16);
```

Βήμα 3: Επανακβαντίστε το ψηφιακό σήμα στα 8bit χωρίς τη χρήση dither.

```
» Sine8NoDither=QuantiseAudio(Sine16,8,32767,-32768,0,0);
```

Βήμα 4: Δείτε την έξοδο του 8bit κβαντιστή σε διάγραμμα σε συνάρτηση του χρόνου. Ερώτηση: γιατί η έξοδος του κβαντιστή είναι μηδενική;

```
» t=[0:1:length(Sine8NoDither)-1]/44100;
```

```
» figure
```

```
» plot(t,Sine8NoDither)
```

Βήμα 5: Επανακβαντίστε και πάλι το 16bit ψηφιακό ημιτονικό σήμα στα 8bit κάνοντας χρήση TPDF dither (χωρίς noise-shaping).

```
» Sine8WithDither=QuantiseAudio(Sine16,8,32767,-32768,0,2);
```

Βήμα 6: Συγκρίνετε τη νέα έξοδο του 8bit κβαντιστή με την προηγούμενη.

```
» hold on
```

```
» plot(t, Sine8WithDither, 'c-')
```

Βήμα 7: Επανακβαντίστε και πάλι το αρχικό 16bit ψηφιακό σήμα στα 8bit κάνοντας χρήση TPDF dither με χρήση noise-shaping 3^{ης} τάξης.

```
» Sine8WithDitherNS=QuantiseAudio(Sine16,8,32767,-32768,3,2);
```

Βήμα 8: Απεικονίστε τη νέα έξοδο του κβαντιστή σε διάγραμμα σε συνάρτηση του χρόνου.

```
» plot(t, Sine8WithDitherNS, 'm-')
```

Βήμα 9: Δείτε το φάσμα των εξόδων του κβαντιστή σε 8bit στις δύο περιπτώσεις εφαρμογής του dither.

```
» [Spectrum8WithDither, f]=CalculateSpectrum(Sine8WithDither);
» [Spectrum8WithDitherNS, f]=CalculateSpectrum(Sine8WithDitherNS);
```

Βήμα 10: Επαναλάβετε τα προηγούμενα στάδια της δραστηριότητας, δημιουργώντας ένα 16bit ημιτονικό σήμα χρονικής διάρκειας ίσης με 5 δευτερόλεπτα, αλλάζοντας την τιμή της παραμέτρου `more_periods` στην κλίση της συνάρτησης `DigitalSineWave`. Μην απεικονίσετε τις χρονικές εξόδους του επανακβαντιστή στα 8bit, γιατί λόγω μεγέθους των σημάτων, η απεικόνιση μπορεί να μην πραγματοποιηθεί. Επίσης, για τους ίδιους λόγους μην εκτελέσετε τους φασματικούς υπολογισμούς και απεικονίσεις των σημάτων που προκύπτουν στα στάδια που περιγράφηκαν παραπάνω. Σημειώστε ότι η εκτέλεση της διαδικασίας επανακβαντισμού του σήματος ενδέχεται να διαρκέσει αρκετά, λόγω του σειριακού τρόπου υπολογισμού του dither από τις αντίστοιχες συναρτήσεις.

```
» clear
» Sine16=DigitalSineWave(0.001, 1000, 500);
» Sine8NoDither=QuantiseAudio(Sine16, 8, 32767, -32768, 0, 0);
» Sine8WithDither=QuantiseAudio(Sine16, 8, 32767, -32768, 0, 2);
» Sine8WithDitherNS=QuantiseAudio(Sine16, 8, 32767, -32768, 3, 2);
```

Βήμα 11: Αποθηκεύστε το αρχικό 16bit και όλα τα επανακβαντισμένα ημιτονικά σήματα σε αρχεία τύπου `.wav` με συχνότητα δειγματοληψίας 44.1kHz και ευκρίνεια κβαντισμού 16bit. Αλλάξτε τη διαδρομή αποθήκευσης εάν το επιθυμείτε. Στη συνέχεια προχωρήστε στην αναπαραγωγή τους. Τι παρατηρείτε;

```
» wavwrite(Sine8NoDither/32768, 44100, 16, 'Sine8NoDither.wav');
» wavwrite(Sine8WithDither/32768, 44100, 16, 'Sine8WithDither.wav');
» wavwrite(Sine8WithDitherNS/32768, 44100, 16, 'Sine8WithDitherNS.wav');
```

4.3. Κβαντισμός και ψαλιδισμός

Βήμα 1: Δημιουργείστε ένα ημιτονικό σήμα κανονικοποιημένου πλάτους 1 (0dBFS) και συχνότητας 1kHz. Η συχνότητα δειγματοληψίας να επιλεγεί ίση προς 44.1kHz και η ευκρίνεια κβαντισμού 16bit.

```
» Signal16=DigitalSinewave(1, 1000, 1);
```

Βήμα 2: Υπολογίστε και απεικονίστε το φάσμα του σήματος που δημιουργήσατε.

```
» [Spectrum16, f]=CalculateSpectrum(Signal16);
```

Βήμα 3: Πολλαπλασιάστε το ημιτονικό σήμα με έναν σταθερό παράγοντα κέρδους ίσο προς 1.2

```
» GSignal16=1.2*Signal16;
```

Βήμα 4: Το νέο σήμα έχει δυναμική περιοχή μεγαλύτερη των 16bit (μπορείτε να εξηγήσετε το γιατί;). Επανακβαντίστε το ώστε να μπορεί και πάλι να αναπαρασταθεί από 16bit ακρίβεια.

```
» GSignal16=QuantiseAudio(GSignal16,16,32767,-32768,0,0);
```

Βήμα 5: Υπολογίστε και απεικονίστε το φάσμα του νέου σήματος. Που οφείλεται η ύπαρξη της παραμόρφωσης;

```
» [GSpectrum16,f]=CalculateSpectrum(GSignal16);
```

Βήμα 6: Απεικονίστε και τα δύο παραπάνω σήματα στο πεδίο του χρόνου. Τι παρατηρείτε;

```
» t=[0:1:length(Signal16)-1]/44100;
» figure
» plot(t,Signal16)
» hold on
» plot(t,GSignal16,'m-')
```

5. Θέματα υπερδειγματοληψίας

Στόχος της ενότητας αυτής είναι η κατανόηση της επίδρασης της υπερδειγματοληψίας στα χαρακτηριστικά του ψηφιακού ηχητικού σήματος, εστιάζοντας στα πλεονεκτήματα που εμφανίζονται (π.χ. αύξηση της δυναμικής περιοχής του συστήματος).

5.1. Η επίδραση της υπερδειγματοληψίας στο φάσμα

Βήμα 1: Δημιουργείστε ένα ημιτονικό σήμα κανονικοποιημένου πλάτους 1 (0dBFS) και συχνότητας 1kHz. Η συχνότητα δειγματοληψίας να επιλεγεί ίση προς 44.1kHz και η ευκρίνεια κβαντισμού 16bit.

```
» Signal=DigitalSinewave(1,1000,1);
```

Βήμα 2: Δημιουργείστε ένα ακόμα ημιτονικό σήμα κανονικοποιημένου πλάτους 1 (0dBFS) και συχνότητας 1kHz με συχνότητα δειγματοληψίας ίση προς 88.2kHz και ευκρίνεια κβαντισμού 16bit. Το σήμα αυτό αποτελεί ουσιαστικά το x2 υπερδειγματοληπτημένο αντίγραφο του ημιτονικού σήματος που δημιουργήσατε στο προηγούμενο βήμα.

```
» SignalOS=DigitalSinewave(1,1000,1);
```

Βήμα 3: Επανακβαντίστε τα δύο ψηφιακά σήματα στα 8bit χωρίς προσθήκη dither και με χρήση noise-shaping 3^{ης} τάξης.

```
» Signal8=QuantiseAudio(Signal,8,32767,-32768,3,0);
» Signal8OS=QuantiseAudio(SignalOS,8,32767,-32768,3,0);
```

Βήμα 4: Υπολογίστε και απεικονίστε τα φάσματα των δύο παραπάνω 8bit σημάτων. Υπολογίστε τις τιμές της αρμονικής παραμόρφωσης+θόρυβο (THD+N) και στις δύο περιπτώσεις και συγκρίνετέ τις.

```
» [Spectrum8,f]=CalculateSpectrum(Signal8);
» [Spectrum8OS,fos]=CalculateSpectrum(Signal8OS);
```

Βήμα 5: Για λόγους σύγκρισης, δημιουργήστε ένα νέο διάγραμμα και απεικονίστε σε αυτό και τα δύο φάσματα των 8bit σημάτων. Τι παρατηρείτε; Ποιό σήμα έχει μεγαλύτερη δυναμική περιοχή;

```

» figure
» plot(f,Spectrum8,'b-');
» hold on
» plot(fos(1:length(f)),Spectrum8OS(1:length(f)), 'm-');
» xlabel('Frequency (Hz)')
» ylabel('Amplitude (dB)')
» title('Spectrum plot of oversampled and normal signals')

```

Βήμα 6: Επαναλάβετε την προηγούμενη διαδικασία, δημιουργώντας ψηφιακές ημιτονικές κυματομορφές μεγαλύτερης διάρκειας. Όπως αναφέρθηκε και στη Δραστηριότητα 2(β), αυτό επιτυγχάνεται αλλάζοντας κατάλληλα την τιμή της παραμέτρου `more_periods` στην κλίση της συνάρτησης `DigitalSineWave`. Επανακβαντίστε στα 8bit τα δύο σήματα με χρήση `noise-shaping` τρίτης τάξης και (χωρίς να πραγματοποιήσετε φασματική ανάλυσή τους) αποθηκεύστε τα σε αρχεία τύπου `.wav` με ευκρίνεια κβαντισμού 16bit και την αντίστοιχη τιμή της συχνότητας δειγματοληψίας. Αλλάξτε τη διαδρομή αποθήκευσης εάν το επιθυμείτε. Σημειώστε ότι η διάρκεια εκτέλεσης του κβαντισμού σε 8bit μπορεί να είναι αρκετά μεγάλη, λόγω της εφαρμογής του `noise-shaping`.

```

» Signal=DigitalSinewave(1,1000,500);
» SignalOS=DigitalSinewave(1,1000,500);
» Signal8=QuantiseAudio(Signal,8,32767,-32768,3,0);
» Signal8OS=QuantiseAudio(SignalOS,8,32767,-32768,3,0);
» wavwrite(Signal8/32768,44100,16,'d:\SineNoOS.wav');
» wavwrite(Signal8OS/32768,88200,16,'d:\SineWithOS.wav');

```

Βήμα 7: Αναπαράγετε τα παραπάνω αρχεία ήχου, εστιάζοντας κυρίως στην ακουστότητα του θορύβου κβαντισμού. Τι παρατηρείτε;

Βήμα 8: Επαναλάβετε την προηγούμενη διαδικασία, χωρίς τη χρήση `noise-shaping` κατά τον επανακβαντισμό σε 8bit και ακούστε τη διαφορά. Τι παρατηρείτε;

5.2. Η εφαρμογή της μεθόδου της υπερδειγματοληψίας σε ψηφιακά σήματα

Βήμα 1: Διαβάστε από το δίσκο το αρχείο `speech.wav` που βρίσκεται στο φάκελο `DATToolbox`. Αλλάξτε τη διαδρομή του αρχείου εάν αυτό είναι απαραίτητο, ανάλογα με τη διαδρομή στην οποία το έχετε αποθηκεύσει στον υπολογιστή σας.

```

» [speech,fs,N]=wavread('d:\DATToolbox\speech.wav');

```

Βήμα 2: Μετατρέψτε τις τιμές του σήματος `speech` σε ακέραιες. Όπως αναφέρθηκε και προηγουμένως στη Δραστηριότητα 2(α), εφόσον το σήμα ήταν αποθηκευμένο σε αρχείο `.wav`, είναι ήδη κβαντισμένο (με ευκρίνεια κβαντισμού 16bit), οπότε αρκεί ο πολλαπλασιασμός των τιμών του με την μέγιστη (απολύτως) τιμή της αναπαράστασης προσημασμένων ακεραίων μήκους 2byte.

```

» speech=speech*32768;

```

Βήμα 3: Εφαρμόστε υπερδειγματοληψία παράγοντα $\times 2$ στο σήμα `speech` σε διαδοχικά στάδια. Αρχικά προσθέστε τον απαραίτητο αριθμό μηδενικών μεταξύ των αρχικών δειγμάτων. Η διαδικασία αυτή πραγματοποιείται με χρήση της συνάρτησης `InterpolateZeros` του `DATToolbox`. Ελέγξτε το μέγεθος του νέου σήματος που προκύπτει κάνοντας χρήση της συνάρτησης `length` της `MATLAB`.

```
» R=2; % The oversampling factor
» speech_zeroOS=InterpolateZeros(speech,R);
» length(speech)
» length(speech_zeroOS)
```

Βήμα 4: Στη συνέχεια, όπως προβλέπεται από τη θεωρία της υπερδειγματοληψίας, φιλτράρετε το σήμα που προέκυψε με την προσθήκη των μηδενικών κάνοντας χρήση ενός FIR φίλτρου υπερδειγματοληψίας. (Σημείωση: η δήλωση των συντελεστών του φίλτρου `firx2` πρέπει να γίνει σε μια σειρά, αλλιώς η `MATLAB` θα εμφανίσει σφάλμα).

```
» firx2=[-850 0 245 0 -541 0 1041 0 -1865 0 3303 0 -6400 0 20670 32767
20670 0 -6400 0 3303 0 -1865 0 1041 0 -541 0 245 0 -85];
» speechOS=filter(firx2/32767,1,speech_zeroOS);
```

Βήμα 5: Επανακβαντίστε το αρχικό και το υπερδειγματοληπτημένο σήμα στα 8bit χωρίς προσθήκη `dither` και με χρήση `noise-shaping` 3^{ης} τάξης.

```
» speech8=QuantiseAudio(speech,8,32767,-32768,3,0);
» speech8OS=QuantiseAudio(speechOS,8,32767,-32768,3,0);
```

Βήμα 6: Αποθηκεύστε τα 8bit ψηφιακά σήματα σε αρχεία τύπου `.wav` με ευκρίνεια κβαντισμού 16bit και κατάλληλη τιμή συχνότητας δειγματοληψίας. Αλλάξτε τη διαδρομή αποθήκευσης εάν το επιθυμείτε.

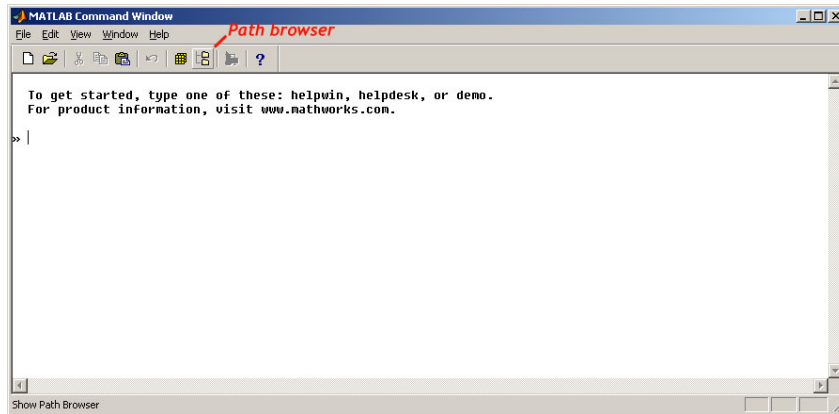
```
» wavwrite(speech8/32768,fs,16,'d:\SpeechNoOS.wav');
» wavwrite(speech8OS/32768,R*fs,16,'d:\SpeechWithOS.wav');
```

Βήμα 7: Αναπαράγετε τα αρχεία ήχου που αποθηκεύσατε, εστιάζοντας κυρίως στην ακουστότητα του θορύβου κβαντισμού. Τι παρατηρείτε;

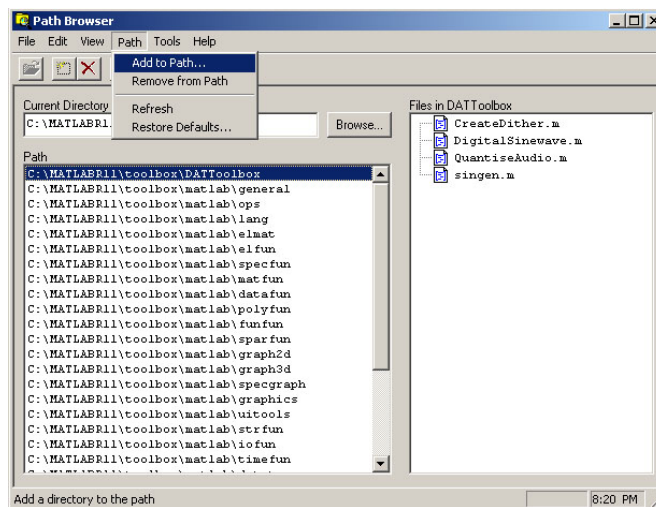
6. Υλοποίηση της άσκησης

Πριν την εκτέλεση των δραστηριοτήτων που περιγράφηκαν προηγουμένως, θα πρέπει:

1. Να έχετε αντιγράψει το φάκελο `DATToolbox` σε κάποια διαδρομή στο δίσκο σας, κατά προτίμηση στο εσωτερικό του φακέλου `toolbox` της εγκατεστημένης `MATLAB`. Τον εν λόγω φάκελο μπορείτε να τον ανακτήσετε από το e-class.
2. Να έχετε προσθέσει το φάκελο `DATToolbox` στα `default paths` της `MATLAB`. Αυτό γίνεται επιλέγοντας το κουμπί “Path Browser” στην εργαλειοθήκη της `MATLAB` (Εικόνα 2) και στη συνέχεια πατώντας στο `menu` του εμφανιζόμενου παραθύρου “Path” → “Add to path” κι επιλέγοντας τον φάκελο `DATToolbox` (Εικόνες 3 και 4).



Εικόνα 2: Πρόσβαση στον path Browser της MATLAB



Εικόνα 3: Προσθήκη στο path



Εικόνα 4: Προσθήκη του DATToolbox

7. Παράδοση της εργασίας

Παραδοτέο της συγκεκριμένης άσκησης αποτελεί αναφορά σε ηλεκτρονική μορφή (τύπου .doc ή .pdf), στην οποία απαραίτητως θα περιλαμβάνονται οι παρακάτω ενότητες:

1. Εξώφυλλο με τα στοιχεία της εργασίας, τα προσωπικά σας στοιχεία (ονοματεπώνυμο, έτος σπουδών και αριθμός μητρώου) και η ομάδα / ημερομηνία εκτέλεσης της άσκησης.
2. Ενότητα «Εισαγωγή» στην οποία θα εξηγήτε επιγραμματικά τους στόχους της άσκησης.
3. Ενότητα «Θεωρία» στην οποία θα αναπτύσσετε το θεωρητικό υπόβαθρο των αποτελεσμάτων της επεξεργασίας και της ανάλυσής σας.
4. Ενότητα «Υλοποίηση της άσκησης», στην οποία θα περιγράψετε τη διαδικασία εκτέλεσης της άσκησης που πραγματοποιήσατε.

5. Ενότητα «Αποτελέσματα» στην οποία θα δίνετε τα ζητούμενα από την άσκηση αποτελέσματα υπό μορφή πινάκων ή / και διαγραμμάτων συχνότητας ή / και διαγραμμάτων χρόνου.
6. Ενότητα «Συμπεράσματα» στην οποία θα συνοψίζετε σε μία-δύο παραγράφους τις εργασίες που εκτελέσατε και θα ερμηνεύετε ποιοτικά και ποσοτικά τα αποτελέσματα που λάβατε.

Η κατάθεση της ηλεκτρονικής αναφοράς θα γίνει ηλεκτρονικά μέσω του e-class. Η προθεσμία υποβολής είναι αυστηρή και ορίζεται ως μία εβδομάδα μετά την εκτέλεση της άσκησης στο εργαστήριο. Υπενθυμίζεται ότι η εργασία είναι ατομική.

8. Σημαντικές παρατηρήσεις

1. Δεδομένου ότι τα βήματα που πρέπει να ακολουθηθούν για την εκτέλεση της εργαστηριακής άσκησης είναι καθορισμένα, παρακαλούνται οι συμμετέχοντες να προσέλθουν χωρίς καθυστέρηση την προβλεπόμενη ώρα, καθώς θα είναι αδύνατη η προς τα πίσω εξήγηση, η ερμηνεία επιμέρους λεπτομερειών και η παροχή πρόσθετων διευκρινίσεων.
2. Για την εκτέλεση της άσκησης στις συνθήκες του εργαστηρίου, απαραίτητη κρίνεται η χρήση ακουστικών. Για το λόγο αυτό παρακαλούνται όλοι οι φοιτητές να φέρουν μαζί τους το προσωπικό τους ζευγάρι ακουστικών.