



**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ  
ΕΛΛΑΔΑΣ  
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΤΕ  
(ΠΡΩΗΝ: ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ)**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ  
ΔΗΜΙΟΥΡΓΙΑ ΔΙΑΔΡΑΣΤΙΚΗΣ ΕΦΑΡΜΟΓΗΣ ΓΙΑ ΤΗΝ  
ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΗΧΗΤΙΚΩΝ ΣΗΜΑΤΩΝ  
ΑΠΟ ΑΝΑΛΟΓΙΚΕΣ ΚΑΙ ΨΗΦΙΑΚΕΣ ΠΗΓΕΣ  
ΣΗΜΑΤΟΣ**

**ΝΤΙΜΟ ΜΕΓΚΗ (ΑΜ 1653)  
ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ: ΑΘΑΝΑΣΙΟΣ ΚΟΥΤΡΑΣ**

**ΝΑΥΠΑΚΤΟΣ 2018**



## ΠΙΣΤΟΠΟΙΗΣΗ

Πιστοποιείται ότι η πτυχιακή εργασία με θέμα:

**«ΔΗΜΙΟΥΡΓΙΑ ΔΙΑΔΡΑΣΤΙΚΗΣ ΕΦΑΡΜΟΓΗΣ ΓΙΑ ΤΗΝ  
ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΗΧΗΤΙΚΩΝ ΣΗΜΑΤΩΝ ΑΠΟ  
ΑΝΑΛΟΓΙΚΕΣ ΚΑΙ ΨΗΦΙΑΚΕΣ ΠΗΓΕΣ ΣΗΜΑΤΟΣ»**

της φοιτήτριας του Τμήματος Μηχανικών Πληροφορικής ΤΕ

**ΝΤΙΜΟ ΜΕΓΚΗ**

**A.M.: 1653**

παρουσιάστηκε δημόσια και εξετάσθηκε στο Τμήμα Μηχανικών Πληροφορικής ΤΕ στις

\_\_\_\_ / \_\_\_\_ / \_\_\_\_

Ο ΕΠΙΒΛΕΠΩΝ

Ο ΠΡΟΕΔΡΟΣ ΤΟΥ ΤΜΗΜΑΤΟΣ

**ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΕΠΙΒΛΕΠΟΝΤΑ**

## ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΠΕΡΙ ΜΗ ΛΟΓΟΚΛΟΠΗΣ

Βεβαιώνω ότι είμαι συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Ακόμα δηλώνω ότι αυτή η γραπτή εργασία προετοιμάστηκε από εμένα προσωπικά και αποκλειστικά και ειδικά για την συγκεκριμένη πτυχιακή εργασία και ότι θα αναλάβω πλήρως τις συνέπειες εάν η εργασία αυτή αποδειχθεί ότι δεν μου ανήκει.

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ

ΑΜ

ΥΠΟΓΡΑΦΗ



.....

.....

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Επιθυμώ να εκφράσω τις ευχαριστίες μου στον Καθ.Αθανάσιο Κούτρα για την βοήθεια του ,τα σχόλια του και την προσεκτική εξέταση της εργασίας .Επιπλέον, επιθυμώ να ευχαριστώ τον Καθ.Γκοτσίνα Αντώνη για την αρχική επιλογή του θέματος της πτυχιακής, την βοήθεια του και την ενθάρρυνσή του.

# ΠΡΟΛΟΓΟΣ

Στόχος της πτυχιακής εργασίας είναι η παρουσίαση εννοιών της τεχνολογίας που σχετίζονται με την ψηφιακή επεξεργασία του ήχου αναλογικού και ψηφιακού και την υλοποίηση του σε ανοιχτού κώδικα λογισμικό , πιο συγκεκριμένα έχει γίνει προσπάθεια ανάλυσης του ηχητικού σήματος και σε γλώσσα προγραμματισμού Python με τη εισαγωγή βασικών βιβλιοθηκών και υλοποίηση με το XAMPP για τη δημιουργία διαδραστικής εφαρμογής.

# ΠΕΡΙΛΗΨΗ

Σε αυτή την εργασία , θα περιγράψουμε τη βασική διαδικασία πως από το αναλογικό ήχο φτάνουμε στο ψηφιακό σήμα και στη συνέχεια την επεξεργασία αυτού του σήματος χρησιμοποιώντας Python. Το πρώτο κεφάλαιο είναι εισαγωγικό και ανατρέπει σε βασικά σημεία που αφορούν στο φυσικό φαινόμενο του ήχου και στα βασικά χαρακτηριστικά. Το δεύτερο κεφάλαιο αναφέρεται στη μέτρηση του ήχου , στη συνέχεια θα συζητήσουμε εν συντομία τις βασικές έννοιες για το πώς μπορούμε από το αναλογικό να μεταβούμε στο ψηφιακό ,πρώτα πως μπορούμε να συλλέξουμε ήχο ,στην συνέχεια πως μπορούμε να το μετατρέψουμε μέσα από την βασική ιδέα της ψηφιοποίησης, ακόμα γίνεται μια θεωρητική μελέτη στην ανάλυση Fourier και στο τελευταίο κεφάλαιο πρόκειται να μελετήσουμε πάνω στο περιβάλλον εργασίας ώστε να αναμεταδώσουμε , φιλτράρουμε και αναπαράγουμε ήχο.

## ABSTRACT

In this thesis, we will describe the basic process how we get the digital signal from the analog signal and then process this signal using Python. The first chapter is introductory and refers to key points concerning the physical phenomenon of sound and basic features. The second chapter refers to the measurement of sound, then we will briefly discuss the basic concepts of how we can go from analog to digital, first how we can collect sound, then how can we convert it from the basic idea of digitization is still a theoretical study on Fourier analysis and in the last chapter we are going to study the working environment to relay, filter and reproduce sound.

## ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ

ήχος, αναλογικό, ψηφιακό, επεξεργασία ήχου, python

# ΠΕΡΙΕΧΟΜΕΝΑ

<b>ΕΥΧΑΡΙΣΤΙΕΣ</b> .....	5
<b>ΠΡΟΛΟΓΟΣ</b> .....	6
<b>ΠΕΡΙΛΗΨΗ</b> .....	7
<b>ABSTRACT</b> .....	7
<b>ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ</b> .....	7
<b>ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ</b> .....	10
<b>ΕΙΣΑΓΩΓΗ</b> .....	12
<b>1.ΦΥΣΗ ΤΟΥ ΉΧΟΥ</b> .....	12
Ορισμός.....	12
Διάδοση.....	12
Τύποι ηχητικών κυμάτων.....	13
Ταχύτητα ήχου.....	15
Χαρακτηριστικά των ηχητικών κυμάτων.....	17
Ένταση.....	19
<b>2.ΜΕΤΡΗΣΗ ΉΧΟΥ</b> .....	19
Συχνότητα.....	20
Ζώνες συχνοτήτων.....	20
Decibel.....	21
Ηχητικές στάθμες.....	21
<b>3.ΑΙΣΘΗΤΗΡΕΣ ΉΧΟΥ</b> .....	23
Μικρόφωνα.....	23
Ηχητικά συστήματα καναλιών.....	24
<b>4.ΑΠΟ ΑΝΑΛΟΓΙΚΟ ΣΕ ΨΗΦΙΑΚΟ</b> .....	25
Βασικά στάδια ψηφιοποίησης ήχου.....	26
Δειγματοληψία.....	27
Κβαντισμός.....	29
Κωδικοποίηση.....	31
<b>3.ΣΥΜΠΙΕΣΗ</b> .....	34
Μη απωλεστική συμπίεση (lossless compression).....	34
Απώλεια συμπίεσης(lossy).....	36
<b>5.ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΉΧΟΥ</b> .....	37
Τύποι σημάτων.....	40



Φάσμα συχνοτήτων και ανάλυση Fourier .....	41
Διακριτός Μετασχηματισμός Fourier (DFT) .....	44
Ταχύς Μετασχηματισμός Fourier(FFT) .....	45
Λευκός και ο ροζ θόρυβος.....	46
Φίλτρα .....	47
Χαρακτηριστικά των φίλτρων.....	48
Τύποι αναλογικών φίλτρων.....	51
Τύποι ψηφιακών φίλτρων.....	53
Λειτουργία ψηφιακού φίλτρου.....	54
<b>6.ΠΕΡΙΒΑΛΛΟΝ ΕΡΓΑΣΙΑΣ .....</b>	<b>55</b>
Λίγα λόγια για την Python.....	55
Python για την ψηφιακή επεξεργασία σημάτων.....	57
Εκτέλεση python στην index.php.....	61
PHP Ανέβασμα αρχείου .....	63
Index.php .....	65
Κώδικας για file upload & apply lowpass.....	74
Κώδικας για highpass filter .....	79
Κώδικας για recodering .....	80
Φόρτωση αρχείου WAV και εμφάνιση στο πεδίο συχνοτήτων ,Κώδικάς για FFT ( Fast Fourie Transform) .....	82
<b>ΒΙΒΛΙΟΓΡΑΦΙΑ .....</b>	<b>85</b>

# ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ

Σχήμα 1: Εκτροπή σωματιδίων από τη θέση ισορροπία.....	13
Σχήμα 2: Γραφικές παραστάσεις ενός απλού ήχου(I), ενός σύνθετου ήχου(II), ενός.....	13
Σχήμα 3: Εικόνα των συμπιεσμένων και αραιωμένων περιοχών ηχητικού κύματος στον αέρα για μια χρονική στιγμή και (β) αντίστοιχη διακύμανση της.....	14
Σχήμα 4: :Εγκάρσιο κύμα .....	14
Σχήμα 5: Η διάδοση ενός διαμήκους κύματος.....	15
Σχήμα 6: Στα αριστερά (α) ήχος με χαμηλό ύψος και στα δεξιά ήχος με υψηλό ύψος.....	17
Σχήμα 7: Διάγραμμα ακουστικών καμπυλών Των Fletcher-Munson ανά 10 phon για ημιτονοειδείς ήχους σε λογαριθμική χάραξη κατάλληλο για τη λήψη πληροφοριών που αφορούν συχνότητα (ύψος), στάθμη έντασης και στάθμη ακουστότητας .....	18
Σχήμα 8: Τμήμα της συνολικής περιοχής των ηχητικών κυμάτων που απλώνεται από μια πηγή που ηχητικά κύματα στην ταξιδεύουν μακριά από την πηγή προς όλες τις κατευθύνσεις....	19
Σχήμα 9:Συχνότητες διαφορετικών ημιτονοειδών κυμάτων: Στο πρώτο κύμα έχει ημιτονοειδή χαμηλή συχνότητα, υπάρχει μικρή επανάληψη κύκλων. Αντίθετα στο δεύτερο κύμα έχει υψηλή συχνότητα. Σημειώστε ότι ο χρόνος αυξάνεται κατά μήκος της οριζοντίου. ....	20
Σχήμα 10 : Μέτρηση ήχου σε db.....	21
Σχήμα 11 : Στο σχήμα προς τα αριστερά παίρνει τον ήχο εξίσου από όλες τις κατευθύνσεις. Στα δεξιά Παίρνει ήχο από μια καθορισμένη πλευρά .....	21
Σχήμα 12: : Μονοφωνικό- 1 κανάλι, stereo και surround .....	25
Σχήμα 31: Ψηφιοποίηση ήχου βήμα προς βήμα.....	26
Σχήμα 32: Η διαδικασία της μετατροπής ενός αναλογικού σήματος σε ψηφιακό .....	27
Σχήμα 33:Δειγματοληψία.....	27
Σχήμα 34: Ρυθμός δειγματοληψίας (samples/sec) .....	28
Σχήμα 35:Aliasing.....	29
Σχήμα 36:Κβαντισμός.....	29
Σχήμα 37: Ψαλιδισμός (Clipping).....	31
Σχήμα 38 Αναλογικό σήμα προς ψηφιοποίηση.....	32
Σχήμα 39: Δειγματοληψία αναλογικού σήματος με περίοδο T, 13 δείγματα. ....	32
Σχήμα 40: Τιμές δειγμάτων σήματος προς ψηφιοποίηση. ....	32
Σχήμα 41: Μέθοδοι συμπίεσης δεδομένων .....	34
Σχήμα 42: RLE για 2 σύμβολα .....	35
Σχήμα 43: Αλγόριθμος του Huffman .....	35
Σχήμα 13 : (επάνω) Σύστημα διακριτού χρόνου και (κάτω) σύστημα συνεχούς χρόνου .....	38
Σχήμα 14 : :Τύποι σημάτων: αναλογικό, διακριτό και ψηφιακό .....	39
Σχήμα 15: Αναλογικό ημιτονοειδές σήμα.....	39
Σχήμα 16 : Ημιτονοειδές σήμα διακριτού χρόνου .....	40
Σχήμα 17: Παράδειγμα (α) αιτιατού σήματος και (β) μη αιτιατού σήματος.....	40
Σχήμα 18: (α) Άρτια σήμα και (b) περιττό .....	41
Σχήμα 19: κυματομορφή του ημιτόνου στο πεδίο χρόνου και η απεικόνιση στο πεδίο της συχνότητας .....	41
Σχήμα 20: Σύνθετο σύνολο διαφορετικών ημιτονοειδών κυμάτων .Στην περιοχή συχνοτήτων μπορούμε να δούμε πιο καθαρά τι συμβαίνει και ορισμένες περιοχές συχνοτήτων. Αυτό το είδος απεικόνισης μας επιτρέπει να ελέγξουμε καλύτερα τον ήχο, ειδικά εάν βρισκόμαστε στη διαδικασία της διαμόρφωσής του.....	42
Σχήμα 21: Φάση.....	45
Σχήμα 22: Απόκριση συχνότητας σε bandpass φίλτρο .....	46
Σχήμα 23: Λευκός θόρυβος.....	47
Σχήμα 24: Ροζ θόρυβος.....	47

Σχήμα 25: Σε ένα lowpass filter βλέπουμε τη ζώνη διέλευσης (passband), ζώνη αποκοπής (stopband) και τη ζώνη μετάβασης (transition region).....	48
Σχήμα 26: Σε ένα band-pass φίλτρο βλέπουμε το εύρος ζώνης (Bandwidth).....	49
Σχήμα 27 Καμπύλη απόκρισης συχνότητας για low-pass,high-pass,band-pass,band-stop.....	49
Σχήμα 28:είσοδος-έξοδος περιεχομένου στα είδη 4 φίλτρου στο πεδίο της συχνότητας χρόνο.....	
Σχήμα 29:Αναλογικό -ψηφιακό φίλτρο.....	52
Σχήμα 30 Διάγραμμα ψηφιακού φίλτρου.....	53

# ΕΙΣΑΓΩΓΗ

Από τα μέσα του 20ου αιώνα η πρόοδος της ψηφιακής τεχνολογίας επέτρεψε σταδιακά την προσχώρηση του ήχου στον κόσμο των ψηφιακών μέσων και τη δυνατότητα μετατροπής και διαχείρισής του σε ψηφιακή μορφή . Η ψηφιακή επεξεργασία ήχου περιλαμβάνει όλες εκείνες τις διαδικασίες που σχετίζονται με την ηχητική μετατροπή από αναλογικό σε ψηφιακό , διαμόρφωση , αποθήκευση ,μετάδοση του ηχητικού σήματος με τη χρήση ψηφιακής ηχητικής τεχνολογίας.

Η ψηφιακή επεξεργασία ήχου βρίσκει σήμερα εφαρμογή σε πολλούς τομείς, όπως: στις τηλεπικοινωνίες ,συστήματα (κωδικοποίηση/συμπίεση του ήχου με στόχο την γρήγορη μετάδοση και λήψη ηχητικών σημάτων) .Για επιστημονική μελέτη και για την βελτιστοποίηση και την αποκατάσταση ηχητικού σήματος και αλλά πολλά .

Με βάση τα παραπάνω ο ρόλος που διαδραματίζει ο ήχος στην ψηφιακή τεχνολογία είναι αρκετά σημαντικός για το λόγο αυτό καλό είναι να έχουμε μια καλή κατανόηση του τρόπου παραγωγής, μέτρησης και τροποποίησης του ήχου.

## 1.ΦΥΣΗ ΤΟΥ ΉΧΟΥ

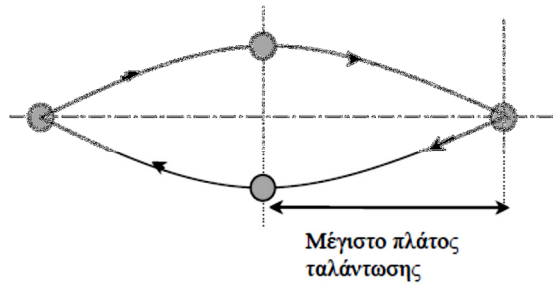
### Ορισμός

Ήχος ορίζεται ως η μηχανική διαταραχή που διαδίδεται με ορισμένη ταχύτητα μέσα σε ένα μέσο που μπορεί να αναπτύξει εσωτερικές δυνάμεις(π.χ ελαστικότητας) και έχει τέτοιο χαρακτήρα ώστε μπορεί να διεγείρει το αισθητήριο ακοής και να προκαλέσει ακουστικό αίσθημα.[1]

### Διάδοση

ταλάντωση ενός αερίου, υγρού ή ελαστικού μέσου η οποία μεταφέρει την ενέργεια της πηγής με μορφή ηχητικών κυμάτων. Σε όλες τις πρακτικές περιπτώσεις όπου ένα αντικείμενο κινείται ή δονείται, ένα μικρό ποσοστό της ενέργειάς του μεταφέρεται στα γειτονικά σωματίδια του ελαστικού μέσου που το περιβάλλει υπό μορφή ταλάντωσης η οποία με την σειρά της επενεργεί ως νέα πηγή, που μεταφέρει και διαδίδει το ηχητικό κύμα όπως περιγράφεται και από την αρχή του Huygens. Για να περιγραφεί η μετάδοση του ήχου χρειάζεται να προσδιοριστούν τα σωματίδια αυτά , δηλαδή τα μόρια ή τα στοιχεία του ελαστικού μέσου.

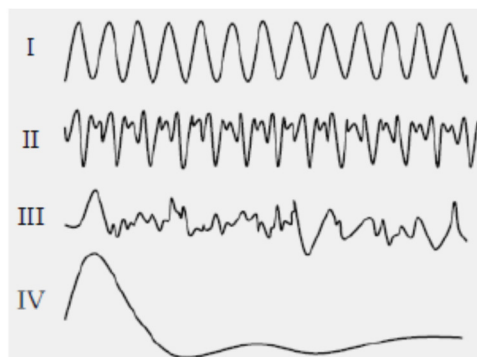
Με τον όρο «στοιχεία του ελαστικού μέσου» ορίζεται ένας στοιχειώδης όγκος του ρευστού, οποίος είναι αφ' ενός μεν τόσο μεγάλος ώστε να συμπεριλαμβάνει εκατομμύρια άτομα. Όταν ένα τέτοιο στοιχείο μετατοπιστεί από τη θέση ισορροπίας του η οποία αρχικά θεωρείται να είναι η ακινησία, τότε θα χτυπήσει γειτονικό του στοιχείο, αναγκάζοντάς το να μετακινηθεί κατά ίση απόσταση. Όταν το αρχικό στοιχείο επιστρέψει στην αρχική του θέση, θεωρώντας ότι εκτελεί μία αρμονική ταλάντωση (δηλαδή μία μαθηματικά ιδανική κίνηση), τότε θα συμβεί το ίδιο και με τα γειτονικά στοιχεία. Με τον τρόπο αυτό η διατάραξη του μέσου διαδίδεται με τη μεταφορά της ενέργειας και της μορφής της αρχικής ταλάντωσης και όχι με τη μεταφορά των ίδιων των στοιχείων του ελαστικού μέσου.



Σχήμα 1: Εκτροπή σωματιδίων από τη θέση ισορροπία

Ανάλογα με την κυματομορφή τους ή το φασματικό περιεχόμενο τους οι ήχοι κατατάσσονται σε διάφορες κατηγορίες. Οι ήχοι στους οποίους όμως ανταποκρίνονται το ανθρώπινο αυτί μπορούν να χωριστούν σε δύο κατηγορίες. Ήχοι βραχείας διάρκειας που αλλάζουν συνεχώς τον χαρακτήρα τους, όπως ο θόρυβος (λευκός, θόρυβοι περιβάλλοντος) που αντιστοιχούν σε μη περιοδικά ηχητικά κύματα, κρότος, απότομοι θόρυβοι μικρής χρονικής διάρκειας.

Ήχοι που χαρακτηρίζονται από την ομαλότητα και την κανονική ροή τους, απλός ήχος ή τόνους, στους οποίους η μορφή του ηχητικού κύματος (κυματομορφή) είναι περιοδική και ημιτονοειδής. Ακόμα υπάρχουν οι σύνθετοι ήχοι ή σύνθετοι τόνοι, που αποτελούνται από τη σύνθεση πολλών απλών ήχων και η κυματομορφή τους είναι μεν περιοδική, αλλά όχι ημιτονοειδής. Οι θόρυβοι και οι κρότοι διαφέρουν από τους απλούς ήχους, διακρίνονται από τη μη περιοδικότητά τους και είναι δυσάρεστοι και ανεπιθύμητοι ήχοι, που η στιγμιαία τους τιμή αυξομειώνεται γενικά με τυχαίο τρόπο.



Σχήμα 2: Γραφικές παραστάσεις ενός απλού ήχου(I), ενός σύνθετου ήχου(II), ενός θορύβου(III) και ενός κρότου(IV).

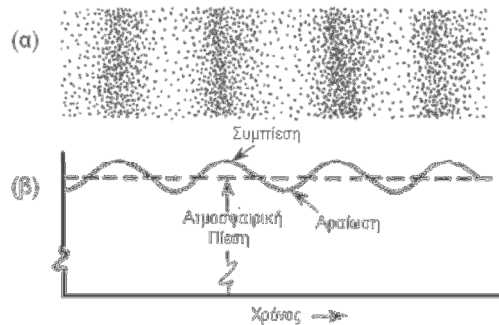
### Τύποι ηχητικών κυμάτων

Η διάδοση του ήχου γίνεται με τη μορφή ηχητικών κυμάτων που αναπτύσσονται, με τον τρόπο που αναφέραμε προηγουμένως, στα διάφορα μέσα. Τα ηχητικά αυτά κύματα ονομάζονται ελαστικά, αφού αναπτύσσονται στα διάφορα ελαστικά μέσα στερεά, υγρά και αέρια. Τα όρια συχνοτήτων που γίνονται αντιληπτά από τον άνθρωπο είναι από έως .Τα ηχητικά κύματα που διαδίδονται σε ελαστικό μέσο με συχνότητα μικρότερη των 20HZ καλούνται υπόηχοι ενώ με συχνότητα μεγαλύτερη των 20.000HZ υπέρηχοι. Τα σώματα τα οποία όταν ταλαντώνονται παράγουν ήχους, καλούνται ηχητικές πηγές. Μια ηχητική πηγή μπορεί για παράδειγμα να είναι:

- ένα διαπασών

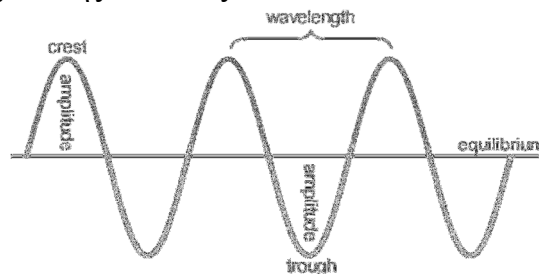
- μια τεντωμένη χορδή βιολιού
- μια μεταλλική πλάκα
- μια μεμβράνη μεγάλου ή ακουστικού τηλεφώνου ή ακόμη και
- μια αέρια μάζα σε ταλάντωση στο εσωτερικό ενός σωλήνα

Τα ηχητικά κύματα είναι διαμήκη και εγκάρσια κύματα τα οποία μπορούν να διαδοθούν σε όλα τα υλικά μέσα στερεά, υγρά και αέρια. Κατά την διάδοσή τους προκαλούν την σχετική μεταβολή της πίεσης του μέσου. Όταν διαδίδονται στον αέρα μεταβάλλεται η πίεση του ατμοσφαιρικού αέρα.



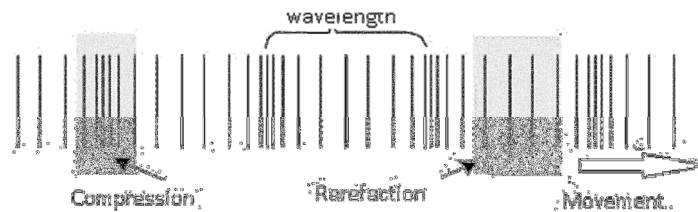
Σχήμα 3: Εικόνα των συμπιεσμένων και αραιωμένων περιοχών ηχητικού κύματος στον αέρα για μια χρονική στιγμή και (β) αντίστοιχη διακύμανση της πίεσης

Σε ένα εγκάρσιο κύμα, το μέσο έχει σωματίδια που δονείται σε κατεύθυνση κάθετη προς την κατεύθυνση της διάδοσης του κύματος. Εδώ γίνεται ο σχηματισμός της κορυφογραμμής και της κοιλάδας.



Σχήμα 4: Εγκάρσιο κύμα

Έτσι, κατά τη διάρκεια των ταλαντώσεων αυτών, τα σωματίδια μπορούν να κινηθούν προς τα πάνω ή προς τα κάτω από το επίπεδο που διέρχεται από τις μέσες θέσεις τους. Το ανώτατο σημείο του κύματος, δηλαδή η θέση της μέγιστης θετικής μετατόπισης, είναι η κορυφή και το χαμηλότερο σημείο, δηλαδή η θέση της μέγιστης μετατόπισης, ονομάζεται κατώτατο σημείο. Έτσι, σε ένα εγκάρσιο κύμα κορυφές και κοιλάδες εμφανίζονται εναλλακτικά. Σε αντίθεση με ένα εγκάρσιο κύμα τα διαμήκη κύματα είναι κύματα στα οποία η κίνηση των μεμονωμένων σωματιδίων του μέσου είναι σε μια κατεύθυνση που είναι παράλληλη προς την κατεύθυνση μεταφοράς ενέργειας. Τα σωματίδια του μέσου ταλαντώνται σε κατεύθυνση κάθετη προς την κατεύθυνση της διάδοσης όπως φαίνεται στο σχήμα.



Σχήμα 5: Η διάδοση ενός διαμήκους κύματος

Το βασικό χαρακτηριστικό ενός διαμήκους κύματος το οποίο το διακρίνει από άλλους τύπους κυμάτων είναι ότι τα σωματίδια του μέσου κινούνται προς μια κατεύθυνση παράλληλη προς την κατεύθυνση μεταφοράς ενέργειας .

Η ταχύτητα των σωματιδίων που περιγράφει την κίνηση-ταλάντωση των μορίων του αέρα ονομάζεται ταχύτητα των σωματιδίων ( $u$ ).

Μήκος κύματος που συχνά υποδηλώνεται με ( $\lambda$ ) ορίζεται ως η απόσταση μεταξύ δύο διαδοχικών σωματιδίων που βρίσκονται ακριβώς στο ίδιο σημείο της διαδρομής τους και κινούνται στην ίδια κατεύθυνση. Τα ζεύγη σωματιδίων βρίσκονται στην ίδια φάση. Στα εγκάρσια κύματα το μήκος κύματος μετράτε από την κορυφή έως την κορυφή ή από το κατώτατο σημείο στο κατώτατο σημείο. Το μήκος κύματος συσχετίζεται με τη συχνότητα ( $f$ ), η οποία αναφέρεται στον αριθμό κυκλικών κύκλων ανά δευτερόλεπτο. Όσο μεγαλύτερη είναι η συχνότητα ενός σήματος, τόσο μικρότερο είναι το μήκος κύματος. Ένα μήκος κύματος μπορεί να υπολογιστεί διαιρώντας την ταχύτητα ενός κύματος με τη συχνότητά του

$$(f): \lambda = u / f.$$

Και ο χρόνος που απαιτείται για ένα κύμα για να γίνει μια πλήρης ταλάντωση ονομάζεται περιοδικός χρόνος ( $T$ )

$$f = 1/T$$

### Ταχύτητα ήχου

Η ταχύτητα του ήχου είναι ένας όρος που χρησιμοποιείται για να περιγράψει την ταχύτητα των ηχητικών κυμάτων που περνούν μέσα από ένα ελαστικό μέσο. Η ταχύτητα ποικίλλει ανάλογα με το μέσο (για παράδειγμα, τα κύματα κινούνται ταχύτερα μέσα από το νερό παρά μέσω του αέρα), καθώς και με τις ιδιότητες του μέσου, ιδιαίτερα τη θερμοκρασία. Ο όρος χρησιμοποιείται συνήθως για να αναφέρεται ειδικά στην ταχύτητα του ήχου στον αέρα. Στο επίπεδο της θάλασσας, σε θερμοκρασία 21 βαθμών Κελσίου και υπό φυσιολογικές ατμοσφαιρικές συνθήκες, η ταχύτητα του ήχου είναι 344 m / s (1238 km / h). Ταχύτητα του ήχου στην περίπτωση των διαμηκών κυμάτων δίνεται από την σχέση:

$$c = \sqrt{\frac{\gamma P}{\rho}} = \sqrt{\frac{\gamma RT}{MB}}$$

$c$  - ταχύτητα του ήχου

$\rho$  - πυκνότητα του μέσου

$T$  - απόλυτη θερμοκρασία του μέσου

$P$  - ατμοσφαιρική πίεση

$R, \gamma$  - σταθερές ( $R=8317$  m/s/oK)

$MB$  - μοριακό βάρος του μέσου διάδοσης.

Για τον αέρα ισχύει  $\gamma=1,4$  οπότε από την προηγούμενη σχέση προκύπτει για τους 0°C η ταχύτητα  $c=331$ m/s [2].

Συνήθως όμως χρησιμοποιείται η παρακάτω προσεγγιστική σχέση για την ταχύτητα του ήχου σε μια θερμοκρασία  $T$ :

$$c=343,2\sqrt{\frac{T}{T_0}}$$

$T$  - θερμοκρασία σε βαθμούς Kelvin

$T_0=293,5\text{ K}$  - θερμοκρασία αναφοράς.

Στην περίπτωση των εγκάρσιων κυμάτων η ταχύτητα του ήχου είναι διαφορετική από αυτή για διαμήκη κύματα και δίνεται από την παρακάτω σχέση:

$$c=\sqrt{\frac{E}{2(1+\sigma)\rho}}$$

$E$  - μέτρο του Young σε  $\text{N/m}^2$

$\sigma$  - λόγος του Poisson

$\rho$  - πυκνότητα του μέσου.

Η σχέση αυτή δηλώνει ότι η ταχύτητα του ήχου στα εγκάρσια κύματα είναι μικρότερη από την ταχύτητα του ήχου στα διαμήκη.

Η ΤΑΧΥΤΗΤΑ ΤΟΥ ΗΧΟΥ ΣΕ ΔΙΑΦΟΡΑ ΥΛΙΚΑ ΣΕ m/s	
<b>Αέρια</b>	
Αέρας (20°C)	344
Ήλιο (20°C)	999
Υδρογόνο (20°C)	1.330
<b>Υγρά</b>	
Υγρό ήλιο (269°C)	211
Υδράργυρος (20°C)	1.451
Νερό (0°C)	1.402
Νερό (20°C)	1.482
Νερό (100°C)	1.543
<b>Στερεά</b>	
Κόκαλο	3.445
Ορείχαλκος	3.480
Γυαλί pyrex	5.170
Χαλύβας	5.790

Πίνακας 1: Ταχύτητες του ήχου σε διάφορα σώματα

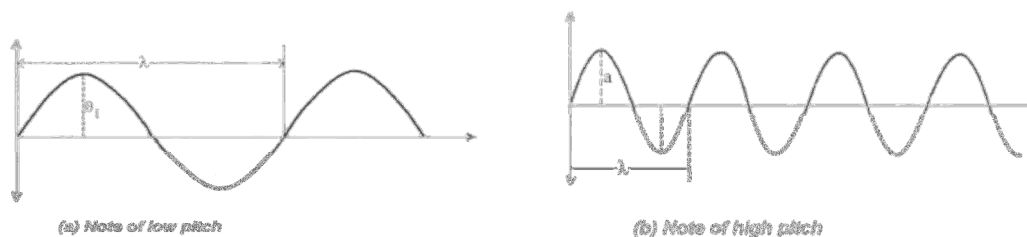
Η ταχύτητα ποικίλει ανάλογα με τις ατμοσφαιρικές συνθήκες. ο σημαντικότερος παράγοντας είναι η θερμοκρασία. Η υγρασία έχει μικρή επίδραση στην ταχύτητα του ήχου, ούτε και η πίεση του αέρα από μόνη της. Η πίεση του αέρα δεν επηρεάζει καθόλου την ιδανική προσέγγιση του αερίου. Αυτό οφείλεται στο γεγονός ότι η πίεση και η πυκνότητα συμβάλλουν εξίσου στην ταχύτητα του ήχου, και σε ένα ιδανικό αέριο τα δύο ακυρώνονται αφήνοντας μόνο την επίδραση της



θερμοκρασίας. Ο ήχος ταξιδεύει συνήθως πιο αργά με μεγαλύτερο υψόμετρο, λόγω μειωμένης θερμοκρασίας.

### Χαρακτηριστικά των ηχητικών κυμάτων

Τα φυσικά χαρακτηριστικά των ηχητικών κυμάτων προκαλούνται από συμπιεστικά κύματα, από τις δονήσεις. Όλα τα ηχητικά κύματα μπορούν να χαρακτηριστούν από το ύψος τους (pitch), την ένταση, την ακουστικότητα και από το φάσμα συχνοτήτων. Η αίσθηση μιας συχνότητας συνήθως αναφέρεται ως το ύψος ενός ήχου.



Σχήμα 6: Στα αριστερά (α) ήχος με χαμηλό ύψος και στα δεξιά ήχος με υψηλό ύψος

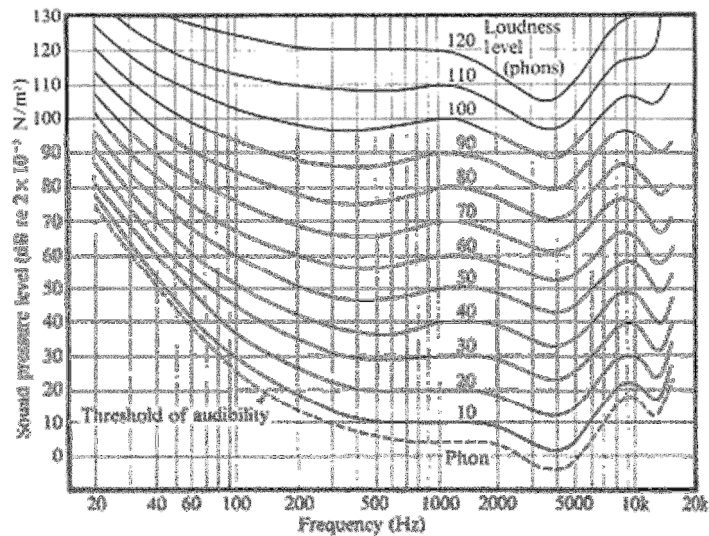
Ένας ήχος υψηλού βήματος αντιστοιχεί σε ένα ηχητικό κύμα υψηλής συχνότητας και ένας ήχος χαμηλού βήματος αντιστοιχεί σε ένα ηχητικό κύμα χαμηλής συχνότητας. Παράδειγμα, αν δύο όργανα μπορούν να παράγουν ήχους παρόμοιας συχνότητας, τότε ακούγονται διαφορετικά, δηλαδή έχουν διαφορετικό ύψος. Η αλλαγή ύψους στην περίπτωση παραγωγής ήχου γίνεται με μεταβολή του μήκους του υλικού που ταλαντώνεται ή μεταβολή του πάχους του υλικού που παράγει τον ήχο. Το ύψος μεταβάλλεται μη γραμμικά με τη συχνότητα, δηλαδή ίσες μεταβολές στη συχνότητα δεν προκαλούν ίδιες μεταβολές στην «αίσθηση» του ύψους.

Επίσης, αν αυξηθεί η ένταση του ήχου το ύψος μπορεί, ανάλογα με τη συχνότητα, να γίνει πιο μεγάλο ή πιο μικρό. Δηλαδή, να δίνεται η αίσθηση είτε οξύ ήχου είτε βαρύ ήχου. Η μονάδα μέτρησης του ύψους είναι η Mel (παράγεται από το Melody).

### Ακουστικότητα

Η ακουστικότητα (Loudness) είναι ένας υποκειμενικός όρος που περιγράφει τη δύναμη της αντίληψης του αυτιού για έναν ήχο, είναι στενά συνδεδεμένη με την ένταση του ήχου, αλλά δεν είναι ίδια. Η ένταση ήχου επίσης σχετίζεται με την ευαισθησία του αυτιού σε συγκεκριμένες συχνότητες που περιέχονται στον ήχο. Σύμφωνα με την αρχή Weber-Fechner, ο τρόπος με τον οποίο γίνονται αντιληπτές οι ακουστικές εντάσεις στην ανθρώπινη ακοή είναι λογαριθμικός και όχι γραμμικός, δηλαδή ανάλογος προς το λογάριθμο του λόγου των εντάσεων (ως προς μια ένταση αναφοράς). Αυτός είναι ο λόγος για τον οποίο η κλίμακα ντεσιμπέλ χρησιμοποιείται για τη μέτρηση της έντασης ήχου. Μια ευρέως όρος για την ένταση ενός συγκεκριμένου ήχου είναι ότι "ο ήχος πρέπει να αυξηθεί ένταση κατά δέκα φορές ώστε ο ήχος να γίνει αντιληπτός ως δύο φορές πιο δυνατός». Γιατί όμως ο διπλασιασμός της έντασης ήχου δεν παράγει διπλή ένταση; Ο λόγος είναι ο ακόλουθος. Τα νευρικά κύτταρα έχουν τα μέγιστα ποσοστά στα οποία μπορούν να πυροδοτήσουν, και φαίνεται ότι ο διπλασιασμός της ηχητικής ενέργειας στο

ευαίσθητο εσωτερικό αυτί δεν διπλασιάζει τη δύναμη του νευρικού σήματος στον εγκέφαλο. Αρα απαιτείται δέκα φορές αύξηση της έντασης ώστε να διπλασιαστεί το σήμα από το εσωτερικό αυτί, και ο ήχος που ακούγεται θα γίνει αισθητός με διπλή ένταση. Ως μονάδα μέτρησης της ακουστότητας ορίστηκε το phon. Η κλίμακα της στάθμης ακουστότητας σε μονάδες phon ταυτίζεται αριθμητικά με την κλίμακα της στάθμης έντασης σε dB (αναφορά  $10^{-12} \text{ W/m}^2$ ) μόνο στη συχνότητα αναφοράς, δηλαδή στη συχνότητα των 1000 Hz. Για παράδειγμα, λέμε ότι ένας ημιτονοειδής ήχος έχει στάθμη ακουστότητας 60 phon, εάν ηχεί εξίσου δυνατά με έναν ημιτονοειδή ήχο συχνότητας 1000 Hz και στάθμης έντασης 60 dB. Για όλες τις άλλες συχνότητες η μετάβαση από την κλίμακα της στάθμης ακουστότητας στην κλίμακα της στάθμης έντασης γίνεται με τη βοήθεια του διαγράμματος των ισοακουστικών καμπυλών, δημιουργήματα των H. Fletcher και W. A. Munson.

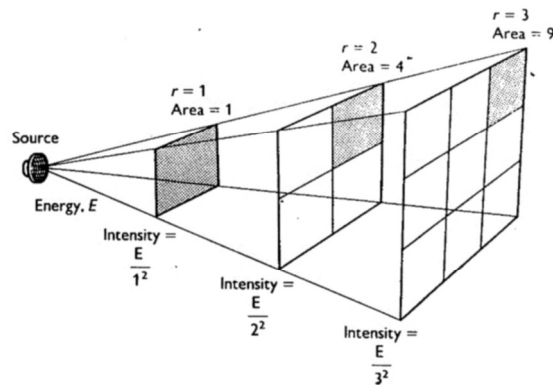


Σχήμα 7: Διάγραμμα ακουστικών καμπυλών των Fletcher-Munson ανά 10 phon για ημιτονοειδείς ήχους σε λογαριθμική χάραξη κατάλληλο για τη λήψη πληροφοριών που αφορούν συχνότητα (ύψος), στάθμη έντασης και στάθμη ακουστότητας

Η ακουστική καμπύλη των μηδέν (0) phon ονομάζεται «κατώφλι ακουστότητας» και είναι αυτή που περνά από το σημείο με συντεταγμένες 1000 Hz για τη συχνότητα και 0 dB SPL για τη στάθμη έντασης, όταν πρόκειται για ακούσματα από το ένα αυτί μέσω ακουστικού. Συμπερασματικά η ηχηρότητα-ακουστικότητα ήχου αναφέρεται στο πόσο δυνατός ή μαλακός ένας ήχος φαίνεται σε έναν ακροατή. Η ηχηρότητα του ήχου καθορίζεται με τη σειρά της από την ένταση των ηχητικών κυμάτων. Η ένταση είναι ένα μέτρο της ποσότητας ενέργειας σε ηχητικά κύματα. Καθώς τα επίπεδα των ντεσιμπέλ αυξάνονται, τα ηχητικά κύματα έχουν μεγαλύτερη ένταση και οι ήχοι είναι πιο δυνατοί.

## Ένταση

Η ένταση των ηχητικών κυμάτων καθορίζει την ένταση των ήχων. Η ένταση εξαρτάται από δύο παράγοντες: το πλάτος των ηχητικών κυμάτων και πόσο μακριά έχουν ταξιδέψει από την πηγή του ήχου. Το πλάτος είναι ένα μέτρο του μεγέθους των ηχητικών κυμάτων. Τα κύματα με μεγαλύτερο πλάτος παράγουν αισθήσεις πιο δυνατών ήχων.



Σχήμα 8: Τμήμα της συνολικής περιοχής των ηχητικών κυμάτων που απλώνεται από μια πηγή που ηχητικά κύματα στην ταξιδεύουν μακριά από την πηγή προς όλες τις κατευθύνσεις

Καθώς τα ηχητικά κύματα ταξιδεύουν μακρύτερα από την πηγή τους, τόσο πιο εκτεταμένη γίνεται η ενέργειά τους. Το ανθρώπινο αυτί μπορεί να ακούσει από 0dB έως 120dB. Ένα επίπεδο 120dB ή υψηλότερο μπορεί να είναι επικίνδυνο για το αυτί. Καθώς η απόσταση από την ηχητική πηγή αυξάνεται, η περιοχή που καλύπτεται από τα ηχητικά κύματα αυξάνεται. Η ίδια ποσότητα ενέργειας κατανέμεται σε μια μεγαλύτερη περιοχή, οπότε η ένταση του ήχου είναι μικρότερη. Αυτό εξηγεί γιατί ακόμα και δυνατοί ήχοι ξεθωριάζουν καθώς μετακινείτε μακρύτερα από την πηγή. Η ένταση εκφράζεται σε Watt/m<sup>2</sup>. Για λόγους απλότητας και συσχέτισης με την ανθρώπινη αντίληψη χρησιμοποιείται η μονάδα decibel (db).

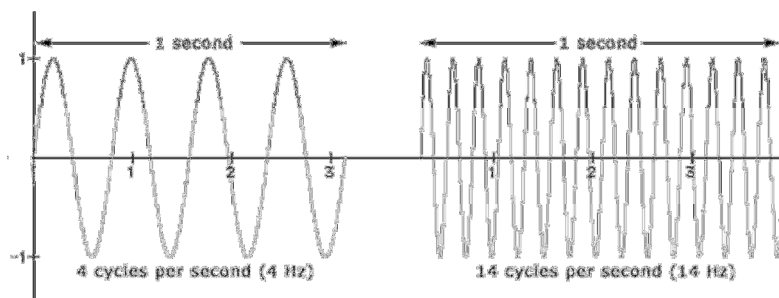
## Φάσμα συχνοτήτων

Ο συνολικός αριθμός συχνοτήτων, η ένταση της κάθε μιας, η κατανομή στο ακουστικό φάσμα και οι συσχετισμοί των φάσεων αποτελούν το φάσμα συχνοτήτων του ήχου και καθορίζουν το σχήμα της κυματομορφής του. Το φάσμα συχνοτήτων είναι εκείνο το χαρακτηριστικό που κάνει δυο ήχους ίδιας συχνότητας και έντασης να διαφέρουν. Οι σύνθετοι ήχοι διαφέρουν επειδή έχουν διαφορετικά φάσματα.

## 2.ΜΕΤΡΗΣΗ ΉΧΟΥ

## Συχνότητα

Η μονάδα μέτρησης συχνότητας είναι Hertz (Hz) και περιγράφει τον αριθμό των κύκλων, η περίοδος επαναλαμβάνεται ανά δευτερόλεπτο ( $1 \text{ Hz} = 1$  κύκλος ανά δευτερόλεπτο). Το ανθρώπινο αυτί είναι ικανό να ανιχνεύει ηχητικά κύματα με ευρύ φάσμα συχνοτήτων που κυμαίνονται μεταξύ περίπου 20 Hz έως 20 000 Hz. Οποιοσδήποτε ήχος με συχνότητα κάτω από το ακουστικό εύρος ακοής (δηλαδή μικρότερη από 20 Hz) είναι γνωστή ως υπερφόρτωση και κάθε ήχος με συχνότητα πάνω από το ακουστικό εύρος ακρόασης (δηλαδή πάνω από 20 000 Hz) είναι γνωστός ως υπέρηχος.



Σχήμα 9: Συχνότητες διαφορετικών ημιτονοειδών κυμάτων: Στο πρώτο κύμα έχει ημιτονοειδή χαμηλή συχνότητα, υπάρχει μικρή επανάληψη κύκλων. Αντίθετα στο δεύτερο κύμα έχει υψηλή συχνότητα. Σημειώστε ότι ο χρόνος αυξάνεται κατά μήκος της οριζοντίου.

## Ζώνες συχνοτήτων

Όλα τα ηχητικά σήματα έχουν στην πραγματικότητα και “ευρυζωνικά” χαρακτηριστικά, δηλαδή καλύπτουν ένα ευρύ φάσμα συχνοτήτων. Συνεπώς, όταν μελετάμε ένα ηχητικό σήμα, είναι προτιμότερο να το αναλύουμε στο πεδίο της συχνότητας. Το σύνολο των συχνοτήτων διαχωρίζεται σε συγκεκριμένες διαδοχικές περιοχές, που ονομάζονται ζώνες συχνοτήτων (bands) και η μελέτη γίνεται αναφορικά με μία συχνότητα κάθε ζώνης που ονομάζεται κεντρική.

Κάθε ζώνη αποτελεί ένα συνεχές, κλειστό διάστημα συχνοτήτων με άκρα τις πλευρικές  
 $f_1, f_2$

και περιγράφεται πλήρως από την κεντρική. Στις ζώνες μεταβλητού εύρους, η  $f_0$  ορίζεται ως ο γεωμετρικός μέσος των πλευρικών:

$$f_0 = \sqrt{f_1 f_2}$$

Οι βασικότερες ζώνες μεταβλητού εύρους είναι οι οκτάβες (octave), για τις οποίες

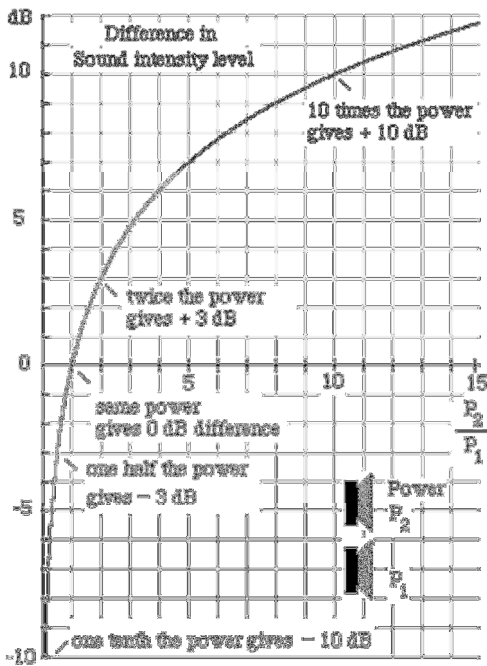
$$f_2 / f_1 = 2$$

Αρά οκτάβα είναι κάθε ζώνη συχνοτήτων της οποίας η τελική συχνότητα είναι διπλάσια της αρχικής. Έτσι η ζώνη συχνοτήτων μεταξύ 90-180Hz είναι μια οκτάβα, όπως το ίδιο ισχύει και για τη ζώνη 1400Hz-2800Hz. Οι κεντρικές συχνότητες των οκταβικών ζωνών κατά αύξουσα σειρά είναι οι: 63Hz, 125Hz, 250Hz, 500Hz, 1KHz, 2KHz, 4KHz, 8KHz. Τριτοοκτάβα είναι κάθε ζώνη συχνοτήτων με αρχική συχνότητα  $f_1$  και  $f_2$  που ικανοποιούν τη σχέση:

$$f_2 / f_1 = 2^{1/3}$$

## Decibel

Το decibel (**dB**) χρησιμοποιείται για τη μέτρηση του ηχητικού επιπέδου. Είναι μια λογαριθμική μονάδα μέτρησης και ουσιαστικά μετρά το λόγο μεταξύ δύο κυμάτων. Ο λόγος μπορεί να είναι η ισχύς, η ηχητική πίεση, η τάση ή η ένταση.



Το μέτρο και η ονομασία του έχει τις ρίζες του στη μονάδα μέτρησης bel, που ονομάστηκε έτσι προς τιμήν του Alexander Graham Bell.

Αν υποθέσουμε ότι έχουμε δύο μεγάφωνα, το πρώτο παίζει έναν ήχο με ισχύ  $P_1$  και το δεύτερο παίζει τον ίδιο ήχο, αλλά πιο δυνατά με ισχύ  $P_2$  τότε η διαφορά των decibel μεταξύ των δύο ορίζεται ως

$10 \log (P_2 / P_1)$  dB οπού log με βάση το 10.

Αν το δεύτερο παράγει διπλάσια ισχύ από την πρώτη, τότε η διαφορά σε dB είναι

$$10 \log (P_2 / P_1) = 10 \log 2 = 3 \text{ dB.}$$

Στο διπλανό γράφημα, το οποίο απεικονίζει  $10 \log (P_2 / P_1)$  έναντι του  $P_2 / P_1$ . Αν συνεχίσουμε το

παράδειγμα, αν το δεύτερο είχε 10 φορές την ισχύ του πρώτου, η διαφορά σε dB θα ήταν  $10 \log (P_2 / P_1) = 10 \log 10 = 10 \text{ dB}$ .

Σχήμα 10 : Μέτρηση ήχου σε db

## Ηχητικές στάθμες

### Στάθμη πίεσης ήχου, SPL

Ο ήχος μετριέται συνήθως με μικρόφωνα και ανταποκρίνονται (περίπου) ανάλογα με την ηχητική πίεση. Τα περισσότερα μικρόφωνα είναι ευαίσθητα στην πίεση του ήχου παρά στην έντασή του.

$$L_p = 10 \log p^2 / p_{ref}^2$$

Όταν τα μικρόφωνα χρησιμοποιούνται σε μετρήσεις ηχητικών σταθμών με ηχώμετρα, οι υπολογισμοί δεν βασίζονται στην ένταση του ήχου, αλλά στην πίεση του

ήχου. Η διαφορά στην στάθμη ηχητικής πίεσης μεταξύ δύο ήχων όπου  $p$  η πίεση του ήχου σε κάποιο σημείο και  $p_{ref}$  η πίεση αναφοράς. Η ακουστική πίεση είναι υψωμένη στο τετράγωνο γιατί η ενέργεια των ηχητικών κυμάτων είναι ανάλογη με το τετράγωνο τη πίεσης.

Στάθμη ισχύος ήχου, SWL

Προκειμένου να μετρήσουμε τη στάθμη ισχύος ενός σήματος, εννοούμε το δεκαπλάσιο του λογαρίθμου του λόγου των αριθμητικών τιμών των ισχύων δύο σημάτων, ένα εκ των οποίων εκλαμβάνεται ως σήμα ανα-

, δηλαδή:

$$L_W = 10 \log W/W_{ref}$$

όπου  $W$  η ακουστική ισχύς της πηγής σε Watt και  $W_{ref}$  η ισχύς αναφοράς.

Στάθμη έντασης ήχου, SIL

Όπου  $I$  το μέτρο της έντασης σε  $W/m^2$  και  $I_{ref}$  η ένταση αναφοράς που υπολογίζεται από τη σχέση

$$L_I = 10 \log I/I_{ref}$$

Όσον αφορά την στάθμη ηχητικής πίεσης και ηχητικής ισχύος, ενώ τα επίπεδα και των δυο εκφράζονται σε decibel αλλά τα πρότυπα για το καθένα είναι διαφορετικά.

Το επίπεδο ηχητικής ισχύος είναι η συνολική ακουστική ισχύς εξόδου ενός ήχου και είναι ανεξάρτητη από το περιβάλλον. Τα επίπεδα ηχητικής πίεσης εξαρτώνται από τους περιβαλλοντικούς παράγοντες, όπως την απόσταση από την πηγή, την παρουσία ανακλαστικών επιφανειών, και άλλων χαρακτηριστικών του χώρου.

Πρόσθεση σταθμών έντασης ήχου

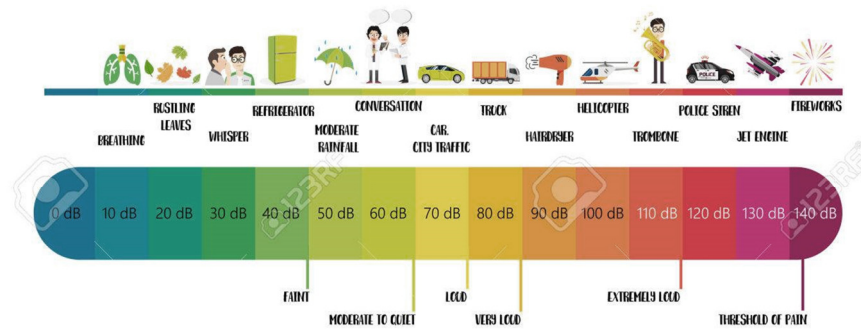
Έστω ότι μας δίδονται  $n$  πλήθους ηχητικές πηγές με στάθμες έντασης

$$IL_1, IL_2, IL_3, \dots, IL_n,$$

αντίστοιχα, οι οποίες συνηχούν σ' ένα χώρο, στην περίπτωση που όλες οι  $n$  ηχητικές πηγές έχουν την ίδια ένταση  $I$ , δηλαδή την ίδια στάθμη έντασης ( $IL$ ), τότε η παραπάνω σχέση παίρνει τη μορφή:

$$IL_{ol} = IL + 10 \times \log(n)$$

Πίνακας 2: Στον πίνακα παρατηρούμε μια κλίμακα από διάφορες τιμές στάθμης ηχητικής πίεσης σε σχέση με γνωστές πηγές που παράγουν ήχους με κάποιες από αυτές τις στάθμες, όρια της ανθρώπινης ακοής, και επιτρεπόμενες στάθμες θορύβου σε διάφορους εργασιακούς και οικιακούς χώρους



### 3. ΑΙΣΘΗΤΗΡΕΣ ΉΧΟΥ

#### Μικρόφωνα

Οι ηλεκτροακουστικές διατάξεις χρησιμοποιούνται για τη μετατροπή των ηχητικών κυμάτων σε ηλεκτρικά σήματα αλλά και την επεξεργασία των τελευταίων πριν από την αντίστροφη διαδικασία (μετατροπή των ηλεκτρικών σημάτων σε ακουστικά). Η μετατροπή της ηχητικής ενέργειας σε ηλεκτρική (και αντίστροφα) έγινε δυνατή με την ανάπτυξη των ηλεκτροακουστικών μετατροπέων και συνέβαλε καθοριστικά στην εξέλιξη των συστημάτων ήχου. Οι ηλεκτροακουστική μετατροπείς χωρίζονται σε δύο κατηγορίες:

1. Αυτούς που μετατρέπουν τα ηχητικά κύματα σε ηλεκτρικά σήματα, δηλαδή τα μικρόφωνα, και
2. Αυτούς που αντίστροφα μετατρέπουν τα ηλεκτρικά σήματα σε ηχητικά κύματα, δηλαδή τα μεγάφωνα.

Όπως είναι προφανές οι μετατροπείς είναι η αρχή και το τέλος κάθε διαδικασίας εγγραφής, επεξεργασίας και αναπαραγωγής του ήχου είτε αυτή γίνεται αναλογικά είτε ψηφιακά. Τα μικρόφωνα είναι ηλεκτρονικές διατάξεις που μετατρέπουν την ενέργεια των ηχητικών (ακουστικών) κυμάτων σε ηλεκτρική ενέργεια. Τα μεγάφωνα επιτελούν την αντίστροφη διαδικασία, δηλαδή, μετατρέπουν ηλεκτρική ενέργεια σε ενέργεια ακουστικών κυμάτων. Για το λόγο αυτό, τα μεγάφωνα και τα μικρόφωνα χαρακτηρίζονται και ως ηλεκτροακουστική μετατροπείς.

Η αρχή λειτουργίας του μικροφώνου βασίζεται στη μετατροπή των ηχητικών κυμάτων που αυτό συλλαμβάνει σε μηχανικές ταλαντώσεις ενός λεπτού ελαστικού διαφράγματος. Στη συνέχεια, οι ταλαντώσεις αυτές μετατρέπονται σε ηλεκτρικά σήματα με χαρακτηριστικά ανάλογα της έντασης και της συχνότητας των ηχητικών κυμάτων. Βασικό χαρακτηριστικό ενός μικροφώνου είναι η ευαισθησία, η οποία ορίζεται ως η ικανότητά του να μετατρέπει μεταβολές πίεσης του μέσου διάδοσης σε

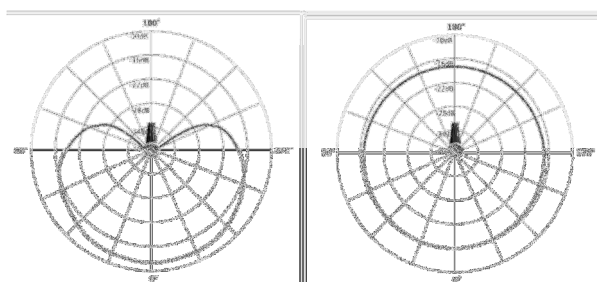
αντίστοιχες μεταβολές τάσης. Με βάση τη μέθοδο που χρησιμοποιείται για τη μετατροπή των ακουστικών σημάτων σε ηλεκτρικά, τα μικρόφωνα διακρίνονται στις εξής τρεις βασικές κατηγορίες:

*Ηλεκτροστατικά μικρόφωνα* : η μετατροπή βασίζεται σε φαινόμενα της ηλεκτροστατικής. Στην κατηγορία αυτή ανήκουν τα *χωρητικά* μικρόφωνα (στα οποία το διάφραγμα λειτουργεί ως ο ένας οπλισμός πυκνωτή οπότε ταλαντευμένος προκαλεί μεταβολές στην απόσταση των δύο οπλισμών και συνεπώς στην τάση του πυκνωτή) και τα *μικρόφωνα φύλλων ηλεκτρική* (στα οποία η ταλάντωση του ηλεκτρική προκαλεί μικρές μεταβολές της τάσης).

*Ηλεκτροδυναμικά μικρόφωνα* : η μετατροπή βασίζεται στο φαινόμενο της ηλεκτρομαγνητικής επαγωγής. Στην κατηγορία αυτή ανήκουν τα *μικρόφωνα κινητού πηνίου*, στα οποία το διάφραγμα είναι ενωμένο με ένα μικρό κινητό πηνίο τοποθετημένο μέσα στο πεδίο ενός μόνιμου μαγνήτη. Οι ταλαντώσεις του διαφράγματος προκαλούν μετακινήσεις του πηνίου μέσα στο μαγνητικό πεδίο με αποτέλεσμα την επαγωγή ρεύματος στο πηνίο. Με παρόμοιο τρόπο λειτουργούν και τα *μικρόφωνα ταινίας*, στα οποία όμως στη θέση του πηνίου υπάρχει μεταλλική ταινία, συνήθως από αλουμίνιο.

*Πιεζοηλεκτρικά μικρόφωνα* : η μετατροπή βασίζεται στο πιεζοηλεκτρικό φαινόμενο, σύμφωνα με το οποίο συγκεκριμένοι κρύσταλλοι παράγουν ηλεκτρική τάση όταν δεχθούν πίεση. Στα μικρόφωνα αυτά, η μηχανική ταλάντωση του διαφράγματος ασκεί την απαραίτητη πίεση στην επιφάνεια του πιεζοηλεκτρικού κρυστάλλου, με τελικό αποτέλεσμα την εμφάνιση ηλεκτρικής τάσης στα άκρα του κρυστάλλου.

Βασικό χαρακτηριστικό ενός μικροφώνου είναι η ευαισθησία στους ήχους υπό διάφορες γωνίες. Αυτή η ευαισθησία ονομάζεται *πολική*, ή *κατευθυντικό*, *μοτίβο*. Η *κατευθυντικότητα* δημιουργείται από αεραγωγούς που επιτρέπουν τον ήχο στο μικρόφωνο με τρόπο που μπορεί να απορρίπτει τον ήχο από ορισμένες γωνίες



Σχήμα 11 : Στο σχήμα προς τα αριστερά παίρνει τον ήχο εξίσου από όλες τις κατευθύνσεις. Στα δεξιά Παίρνει ήχο από μια καθορισμένη πλευρά.

## Ηχητικά συστήματα καναλιών

### Μονοφωνικό- 1 κανάλι

Το μονοφωνικό περιγράφει ένα σύστημα όπου όλα τα ηχητικά σήματα αναμιγνύονται μεταξύ τους και δρομολογούνται μέσω ενός μόνο καναλιού ήχου.

Τα μονοφωνικά συστήματα μπορούν να έχουν πολλαπλά ηχεία, ακόμα και πολλαπλά μεγάλα διαχωρισμένα μεγάφωνα. Το κλειδί είναι ότι το σήμα δεν περιέχει



πληροφορίες σχετικά με το επίπεδο και την ώρα άφιξης / φάσης που θα αναπαράγουν ή θα προσομοιώνουν κατευθυντήριες ενδείξεις.

Σχήμα 12: : Μονοφωνικό- 1 κανάλι, stereo και surround

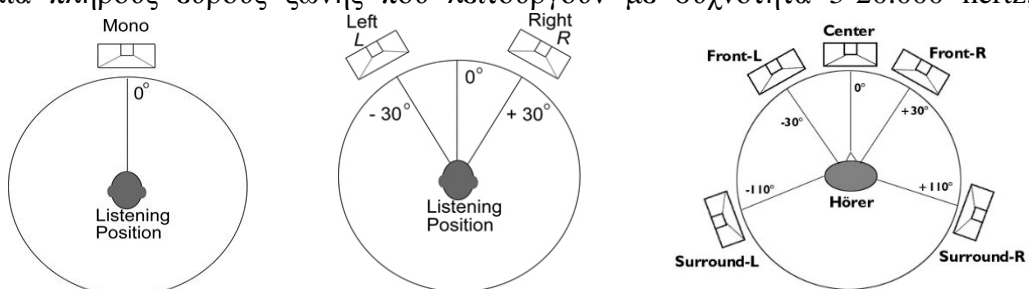
### Στερεοφωνικό- 2 κανάλια

Τα στερεοφωνικά συστήματα ήχου διαθέτουν δύο ανεξάρτητα κανάλια ήχου και τα αναπαραχθέντα σήματα έχουν ένα συγκεκριμένο επίπεδο και μία φάση μεταξύ τους έτσι ώστε όταν αναπαράγονται μέσω ενός κατάλληλου συστήματος αναπαραγωγής, θα υπάρξει μια εμφανής εικόνα της αρχικής πηγής ήχου. Μια βασική λειτουργία του στερεοφωνικού συστήματος αναπαραγωγής είναι ότι η συνολική περιοχή ακρόασης πρέπει να έχει ισότιμη κάλυψη τόσο του αριστερού όσο και του δεξιού καναλιού, σε ουσιαστικά ίσα επίπεδα. Επιπλέον, ένα στερεοφωνικό σύστημα αναπαραγωγής πρέπει να έχει τη σωστή απόκριση απόλυτης φάσης στη είσοδο και στην έξοδο και για τα δύο κανάλια.

### Surround 2+2 κανάλια

Το surround σύστημα είναι μια τεχνολογία πολυканαλου ήχου που χρησιμοποιεί την τεχνική surround έξι καναλιών. Αυτή η τεχνική χρησιμοποιεί πέντε κανάλια πλήρους εύρους ζώνης που λειτουργούν με συχνότητα 3-20.000 hertz,

στο  
χεύ  
οντ  
ας  
το  
μπρ  
οστι  
νό

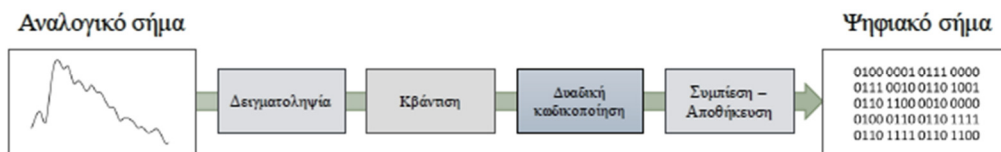


αριστερό, το δεξί, το κεντρικό και το δεξί και το αριστερό surround, καθώς και ένα κανάλι υπογούφερ που λειτουργεί με συχνότητα 3-120 Hertz για εφέ χαμηλής συχνότητας.

## 4.ΑΠΟ ΑΝΑΛΟΓΙΚΟ ΣΕ ΨΗΦΙΑΚΟ

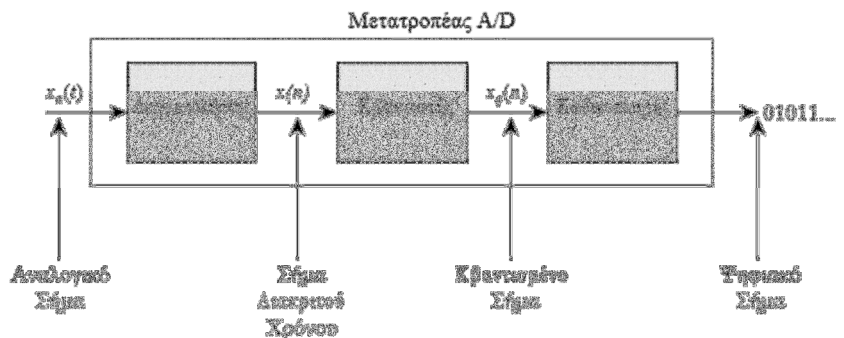
## Βασικά στάδια ψηφιοποίησης ήχου

Η ψηφιοποίηση του ήχου μπορεί να περιγράψει ως η διαδικασία κατά την οποία ένα συνεχές / αναλογικό ηχητικό σήμα μετατρέπεται σε κώδικα ψηφίων. Η μετατροπή της ηχητικής πληροφορίας σε κώδικα ψηφίων μας επιτρέπει να τη διαχειριστούμε με ψηφιακά μέσα, καθώς, ως γνωστό, τα ηλεκτρονικά κυκλώματα μικροεπεξεργαστών επικοινωνούν με μία γλώσσα που στη βάση της αποτελείται από συνδυασμούς ψηφίων με δύο πιθανές τιμές 0 και 1. Για να καταγράψουμε ψηφιακά οποιονδήποτε ήχο πρέπει αυτός πρώτα να μετατραπεί σε μεταβολές ηλεκτρικής τάσης. Για παράδειγμα, όταν ηχογραφούμε οι μεταβολές πυκνότητας του αέρα μετατρέπονται σε μεταβολές τάσης μέσω του μικροφώνου και έπειτα το ηχητικό σήμα μπορεί να ψηφιοποιηθεί από κάποιο ψηφιακό μέσο. Σε περιπτώσεις ήδη ηχογραφημένου ηχητικού υλικού σε αναλογικό μέσο (πχ. μαγνητοταινία) ο ήχος έχει ήδη αποθηκευτεί ως καταγραφές της μεταβολής τάσης οι οποίες μπορούν να σταλούν στο ψηφιακό σύστημα από τις εξόδους κάποιου αναλογικού μέσου αναπαραγωγής. Κατά την ψηφιοποίηση του ήχου αρχικά λαμβάνονται δείγματα ανά συγκεκριμένα χρονικά διαστήματα (δειγματοληψία). Στη συνέχεια τα δείγματα αντιστοιχούνται σε συγκεκριμένες τιμές πλάτους, ή 'στάθμες', ανάλογα με τη μεταβολή τάσης που έχει καταγραφεί στο κάθε δείγμα (κβαντισμός). Τέλος, οι τιμές που έχουν οριστεί κατά τον κβαντισμό οργανώνονται σε μία σειρά ψηφίων (κωδικοποίηση)



Σχήμα 13: Ψηφιοποίηση ήχου βήμα προς βήμα.

Η συσκευή που χρησιμοποιείται για την ψηφιοποίηση του αναλογικού σήματος λέγεται μετατροπέας ADC (Analog to Digital Converter), και συνήθως ενσωματώνεται σε πληρέστερα συστήματα όπως κάρτες ήχου H/Y, ψηφιακές κονσόλες μίξης κτλ. Αφού το ηχητικό σήμα ψηφιοποιηθεί οδηγείται σε κάποια μονάδα, όπου μπορεί να υποστεί ηχητική επεξεργασία, να αποθηκευτεί και να αναπαρασταθεί γραφικά με διάφορους τρόπους. Για να φτάσει τελικά στα αυτιά μας ο ήχος που έχει ήδη ψηφιοποιηθεί και υποστεί επεξεργασία πρέπει πρώτα να μετατραπεί και πάλι σε αναλογικό σήμα με τη χρήση μίας συσκευής που ονομάζεται μετατροπέας DAC (Digital to Analog Converter), και έπειτα το αναλογικό σήμα να σταλεί σε ένα σύστημα ηχείων, όπου τελικά θα μετατραπεί σε μεταβολές πυκνότητας του αέρα. Η λειτουργία δηλαδή ενός μετατροπέα ADC σε σχέση με αυτή ενός μετατροπέα DAC είναι ακριβώς αντίστροφη.



Σχήμα 14: Η διαδικασία της μετατροπής ενός αναλογικού σήματος σε ψηφιακό

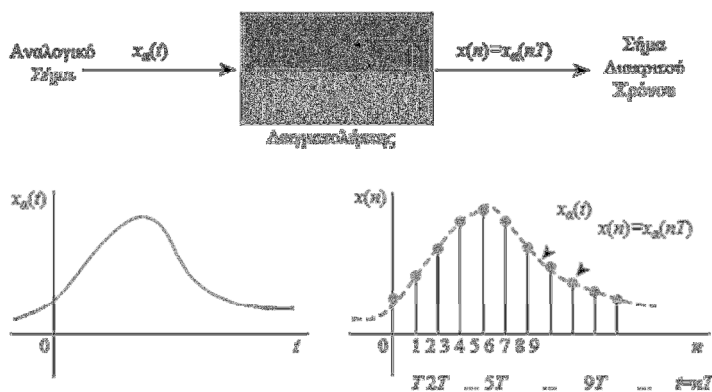
### Δειγματοληψία

Η δειγματοληψία είναι μια μέθοδος μετατροπής ενός αναλογικού συνεχούς σήματος σε ένα ψηφιακό σήμα. Έτσι, αν  $x_a(t)$  είναι η είσοδος στο δειγματολήπτη, τότε η έξοδος αυτού είναι  $x_a(nT)=x(n)$

Η διαδικασία αυτή περιγράφεται από τη σχέση:

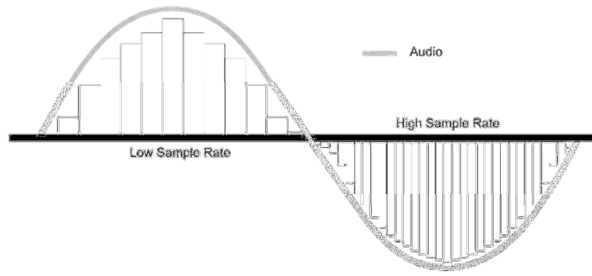
$$x(n) = x_a(nT), -\infty < n < \infty$$

όπου  $x(n)$  είναι το σήμα διακριτού χρόνου που προκύπτει. Το χρονικό διάστημα  $T$  μεταξύ των διαδοχικών δειγμάτων ονομάζεται περίοδος δειγματοληψίας και το αντίστροφο του  $1/T=Fs$  αποτελεί το ρυθμό δειγματοληψίας (sampling rate) σε δείγματα/sec. Οι μεταβλητές χρόνου  $t$  για τα σήματα συνεχούς και  $n$  χρόνου και διακριτού χρόνου συνδέονται γραμμικά μέσω της περιόδου δειγματοληψίας  $T$  και μέσω του ρυθμού της δειγματοληψίας  $Fs=1/T$ .



Σχήμα 15: Δειγματοληψία

Ο ρυθμός δειγματοληψίας (samples/sec) καθορίζει την ποιότητα του ψηφιοποιούμενου ήχου. Χαμηλός ρυθμός/συχνότητα δειγματοληψίας έχει ως αποτέλεσμα χαμηλή ποιότητα ήχου, καθώς η ψηφιακή κυματομορφή που δημιουργείται βάσει των τιμών των δειγμάτων δεν αναπαριστά πιστά το σχήμα της αρχικής αναλογικής κυματομορφής. Όσο περισσότερα δείγματα διαθέτουμε για την ψηφιοποίηση ενός σήματος τόσο πιο πιστά μπορούμε να αναπαραστήσουμε την αναλογική κυματομορφή, και άρα τόσο καλύτερο ηχητικό αποτέλεσμα έχουμε.



Σχήμα 16: Ρυθμός δειγματοληψίας (samples/sec)

Στις επαγγελματικές εφαρμογές ηχητικής επεξεργασίας χρησιμοποιούνται υψηλές συχνότητες δειγματοληψίας καθώς η ποιότητα του ψηφιοποιημένου ήχου καθορίζει την περαιτέρω ψηφιακή επεξεργασία του.

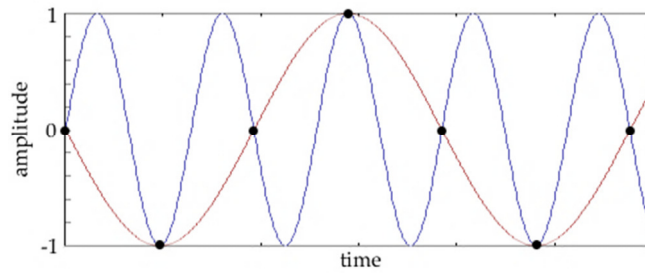
Το θεώρημα Nyquist δηλώνει ότι για να αναπαράγουμε με ακρίβεια ένα αναλογικό σήμα σε ψηφιακή μορφή η τιμή της συχνότητας δειγματοληψίας πρέπει να είναι διπλάσια της συχνότητας του αναλογικού σήματος .

$$f_s \geq 2f_c$$

όπου  $f_s$  είναι η συχνότητα δειγματοληψίας (πόσο συχνά λαμβάνονται δείγματα ανά μονάδα χρόνου) και  $f_c$  είναι η υψηλότερη συχνότητα που περιέχεται στο σήμα. Πιο απλά εάν έχουμε ένα ηχητικό αναλογικό σήμα το οποίο επαναλαμβάνεται 10.000 φορές το δευτερόλεπτο μπορούμε να το αναπαράγουμε με ακρίβεια σε ψηφιακή μορφή αν το αναπροσαρμόσουμε σε μια συχνότητα δειγματοληψίας 20.000 φορές το δευτερόλεπτο . Η τιμή  $f_s = 2f_c$  καλείται όριο Nyquist και αποτελεί τη μικρότερη δυνατή συχνότητα δειγματοληψίας που επιτρέπει την ακριβή ανακατασκευή ενός αναλογικού σήματος, πεπερασμένου εύρους ζώνης, από τα δείγματά του.

Το θεώρημα δειγματοληψίας αναφέρεται σε δύο βασικά αποτελέσματα. Πρώτον ότι είναι δυνατό με κατάλληλη δειγματοληψία να αποφύγουμε την αλλοίωση του συχνοτικού περιεχομένου ενός σήματος πεπερασμένου εύρους ζώνης και δεύτερον ότι είναι δυνατό να ανακατασκευάσουμε ακριβώς το αρχικό αναλογικό σήμα από τα δείγματά του.

Δειγματοληψία με συχνότητα χαμηλότερη από τη συχνότητα Nyquist παραμορφώνει το σήμα, χάνει τις υψηλότερες συχνότητες που τυχόν δεν καλύπτει η επιλεγμένη συχνότητα δειγματοληψίας  $f_s$  . Δημιουργούνται συχνότητες οι οποίες δεν υπήρχαν στο αρχικό σήμα (το φαινόμενο ονομάζεται αναδίπλωση “aliasing”. Ακόμα όταν η συχνότητα δειγματοληψίας  $f_c$  υψηλότερη από τη συχνότητα Nyquist ( $f_s$ ) αποτελεί σπατάλη αποθηκευτικού χώρου καθώς δημιουργεί πρόσθετα δείγματα χωρίς αυτά να είναι απαραίτητα. Παράδειγμα αν ψηφιοποιήσουμε ένα σήμα συχνότητας  $f_c = 40$  KHz με συχνότητα δειγματοληψίας  $f_s = 60$  kHz τότε θα έχουμε αναδίπλωση και το σήμα εξόδου θα έχει συχνότητα  $f_1 = 60 - 40 = 20$ kHz.

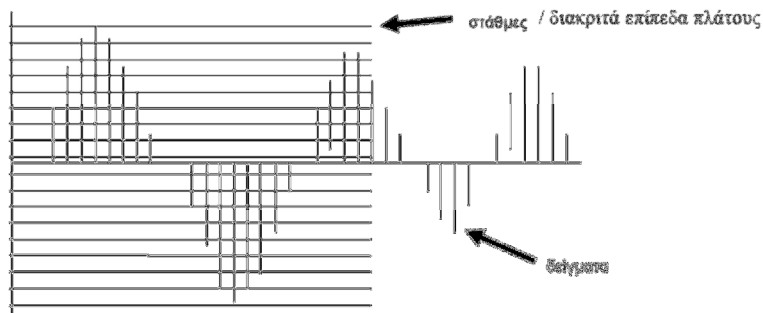


Σχήμα 17: Aliasing

## Κβαντισμός

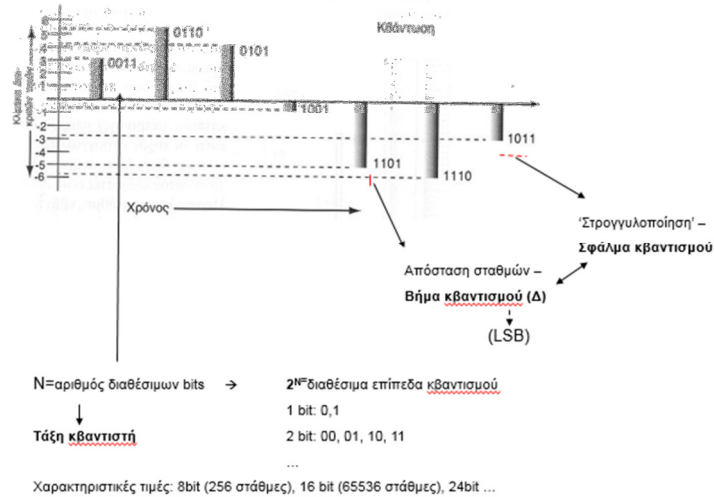
Τα αναλογικά δείγματα έχουν τιμές σε ένα συνεχές διάστημα τιμών και χρειάζεται άπειρος αριθμός bits για την παράστασή τους με τέλεια ακρίβεια.

Κβαντισμός (Quantization) είναι η διαδικασία της προσέγγισης ενός αναλογικού (συνεχούς) δείγματος με ένα πεπερασμένο αριθμό bits. Ο κβαντιστής είναι ένα μη γραμμικό και μη αντιστρέψιμο σύστημα. Το πλήθος αυτό των σταθμών περιγράφεται από τα "bit κωδικοποίησης". Χαρακτηριστικό παράδειγμα αριθμού σταθμών είναι τα "16 bit κωδικοποίησης (ή 16-άμπιτος ήχος)". Σε αυτό το σημείο χρειάζεται ιδιαίτερη προσοχή, καθώς ο αριθμός 16 δεν αναφέρεται στο πλήθος των σταθμών, αλλά στο πλήθος των ψηφίων που θα έχει ο κάθε αριθμός που θα αντιστοιχηθεί σε κάθε στάθμη (θα το δούμε αυτό στην επόμενη φάση της κωδικοποίησης). Τα 8 bit κωδικοποίησης σημαίνουν  $2^8=256$  στάθμες. Όσο μεγαλύτερο αριθμό bits διαθέτουμε, τόσο μεγαλύτερη ακρίβεια υπάρχει στην αποτύπωση του αρχικού σήματος. Αυτό είναι λογικό καθώς η ύπαρξη περισσότερων διακριτών επιπέδων ('στάθμες') επιτρέπει την αναλυτικότερη καταγραφή των τιμών του πλάτους του σήματος.



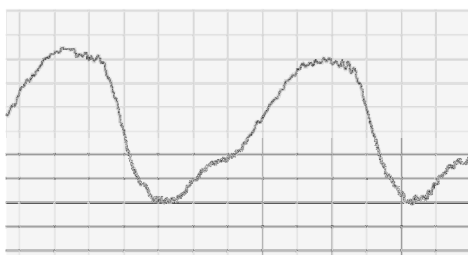
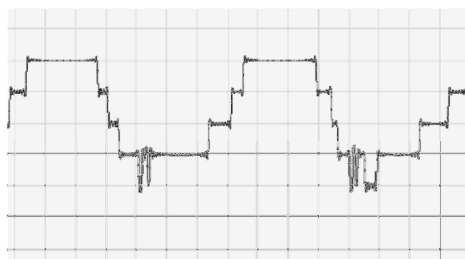
Σχήμα 18: Κβαντισμός

Η απόσταση μεταξύ των επιπέδων κβαντισμού (ή αλλιώς η απόσταση πλάτους ανάμεσα σε δύο στάθμες) ονομάζεται βήμα κβαντισμού ( $\Delta$ ) και συχνά αναφέρεται ως η τιμή του ελάχιστου σημαντικού ψηφίου. Για να έχουμε ένα ικανοποιητικό αποτέλεσμα το βήμα κβαντισμού πρέπει να είναι μικρότερο από τις ελάχιστες μεταβολές του πλάτους του αναλογικού σήματος.



Ακόμη λόγω της ‘στρογγυλοποίησης’ των συνεχών τιμών πλάτους στις κοντινότερες διαθέσιμες διακριτές τιμές, δημιουργείται σφάλμα(quantizing error) ακουστός ως θόρυβος. Το σφάλμα αυτό εκφράζεται από το λόγο σήματος προς θόρυβο, (Signal to Noise Ratio, SNR)  $S/E (dB)=6.02n+176$  όπου το n είναι ο αριθμός των bits της λέξης στο format του ψηφιακού ήχου. Όσο μεγαλύτερος είναι ο αριθμός των db τόσο μικρότερος είναι ο θόρυβος, ενώ τα περισσότερα συστήματα σήμερα κυμαίνονται στα 90db SNR, όπου η παραμόρφωση μετριέται επί τις εκατό του αρχικού σήματος. Η ευκρίνεια του κβαντισμού και του λόγου σήματος προς θόρυβο παρουσιάζεται στον παρακάτω πίνακα όπου ενδεικτικές τιμές SNR αντιστοιχούν σε διαφορετικές τάξεις μετατροπής. Συνοψίζοντας τα πιο πάνω, ανάγεται το συμπέρασμα ότι ο θόρυβος που παράγουν τα ψηφιακά ηχητικά συστήματα, εμφανίζεται κατά την διαδικασία της αναπαραγωγής του κβαντισμένου ηχητικού σήματος, έτσι ώστε να διαμορφώνεται ένας ανεπιθύμητος χαρακτήρας.

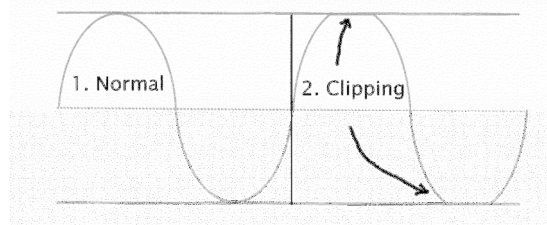
κβαντισμός με 8 bit (256 επίπεδα)



κβαντισμός με 16 bit (65536 επίπεδα)

Στην περίπτωση των 16bit συστημάτων, επιβάλλεται η χρήση του dither κατά την, μέθοδος κατά την οποία ενσωματώνεται μικρή ποσότητα αναλογικού θορύβου πριν από το στάδιο της δειγματοληψίας, εξασθενώντας πολλά από τα προβλήματα που συζητήθηκαν παραπάνω ούτως ώστε να εμφανίζεται ένα ικανοποιητικό αποτέλεσμα. Σε γενικές γραμμές, η προσθήκη του dither τροποποιεί την μορφή του σφάλματος και το καθιστά ανεξάρτητο από το σήμα εισόδου, αποκόπτει κάθε είδους αρμονικής παραμόρφωσης η οποία εμφανίζεται κατά την μετατροπή σήματος μικρού πλάτους και τέλος βελτιώνει την διακριτή ικανότητα του κβαντιστή αυξάνοντας κατά υποκειμενικό τρόπο την δυναμική του περιοχή.

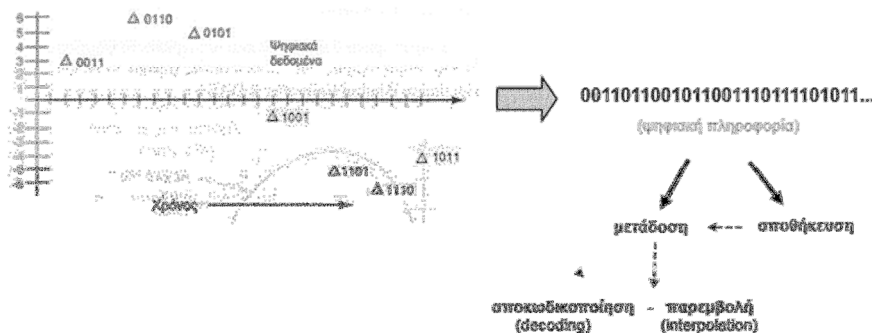
Ένας άλλος σημαντικός παράγοντας που επηρεάζει την ποιότητα του ήχου, είναι ο ψαλιδισμός (clipping) του σήματος, όπου κατά την εγγραφή του σήματος η μέγιστη στάθμη του σήματος εισόδου υπερβαίνει την μέγιστη στάθμη κβαντισμού διαμορφώνοντας κάποια διαστρέβλωση.



Σχήμα 19: Ψαλιδισμός (Clipping).

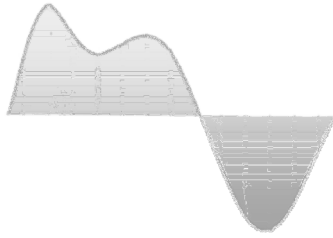
## Κωδικοποίηση

Η έξοδος του κβαντιστή οδηγεί σε ένα κωδικοποιητή, ο οποίος εκχωρεί ένα μοναδικό δυαδικό αριθμό (κώδικα λέξη) σε κάθε ένα επίπεδο κβαντισμού. Μπορεί να χρησιμοποιηθεί οποιαδήποτε εκχώρηση κωδικών λέξεων στα επίπεδα αυτά, καθώς υπάρχουν διάφορα πλαίσια κωδικοποίησης.



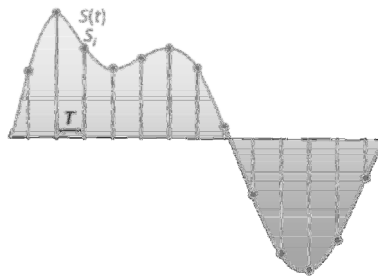
Σχήμα 5.8:Κωδικοποίηση

Παράδειγμα ψηφιοποίησης ήχου:



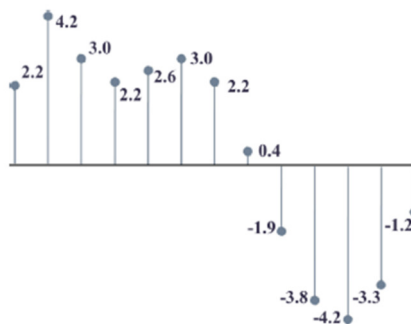
Σχήμα 20 Αναλογικό σήμα προς ψηφιοποίηση.

Αρχικά, πρέπει να αποφασιστεί η περίοδος δειγματοληψίας ( $T$ ), δηλαδή το σταθερό χρονικό διάστημα μεταξύ δύο λήψεων δειγμάτων από το αναλογικό σήμα, η οποία και θα καθορίσει και τη συχνότητα δειγματοληψίας ( $\nu$ , με  $\nu = 1/T$ ). Τα δείγματα που λαμβάνονται ανά χρονικό διάστημα  $T$  ανήκουν στο σύνολο  $S(t)$ , με  $t = 1T, 2T, 3T$  κ.λπ. (για λόγους απλότητας θα συμβολίζονται ως  $S_i$ , όπου  $i = 1, 2, 3, 4 \dots 13$ ) κ.λπ.), και αντιπροσωπεύουν μια τιμή έντασης του αναλογικού σήματος τη δεδομένη χρονική στιγμή.



Σχήμα 21: Δειγματοληψία αναλογικού σήματος με περίοδο  $T$ , 13 δείγματα.

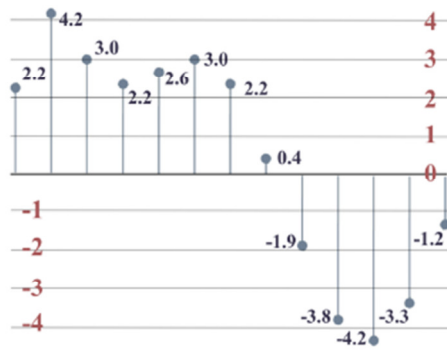
Με δεδομένες τις τιμές των δειγμάτων ξεκινά η διαδικασία κβάντωσης. Συγκεκριμένα, ορίζεται ως ελάχιστη τιμή  $-4$  και ως μέγιστη το  $4$ , με βήμα το  $1$ . Επομένως, το σύνολο των τιμών κβάντωσης είναι το εξής:  $-4, -3, -2, -1, 0, 1, 2, 3, 4$  (9 διακριτές τιμές).



Σχήμα 22: Τιμές δειγμάτων σήματος προς ψηφιοποίηση.

Έχοντας ορίσει τα επίπεδα κβάντωσης, στη συνέχεια γίνεται η αντιστοίχιση των δειγμάτων





Δείγμα	Τιμή δείγματος	Τιμή κβάντισης
S <sub>1</sub>	2.2	2
S <sub>2</sub>	4.2	4
S <sub>3</sub>	3.0	3
S <sub>4</sub>	2.2	2
S <sub>5</sub>	2.6	3
S <sub>6</sub>	3.0	3
S <sub>7</sub>	2.2	2
S <sub>8</sub>	0.4	0
S <sub>9</sub>	-1.9	-2
S <sub>10</sub>	-3.8	-4
S <sub>11</sub>	-4.2	-4
S <sub>12</sub>	-3.3	-3
S <sub>13</sub>	-1.2	-1

Πίνακας 3: Διαδικασία κβάντωσης. Τελικές τιμές κβάντωσης.

Στο επόμενο βήμα οι κβαντωμένες τιμές μετατρέπονται σε δυαδικούς αριθμούς. Οι αριθμοί αυτοί αποθηκεύονται στη μνήμη του υπολογιστή και μπορούν, στη συνέχεια, να χρησιμοποιηθούν για την ανασύνθεση του αρχικού αναλογικού ηχητικού σήματος μέσω της αντίστροφης διαδικασίας μετατροπής σήματος από ψηφιακό σε αναλογικό. Για την αναπαράσταση 9 διακριτών τιμών κβάντωσης απαιτούνται 4 bits ( $2^4 = 16$  διαφορετικές τιμές). Οι τιμές των επιπέδων κβάντωσης μετατρέπονται σε δυαδική μορφή πραγματοποιώντας την εξής παραδοχή: το πρώτο bit υποδηλώνει το πρόσημο (1 για αρνητικούς αριθμούς, 0 για θετικούς αριθμούς) και τα επόμενα 3 bits την τιμή. Άρα, στους πίνακες που ακολουθούν απεικονίζονται οι δυαδικές τιμές των επιπέδων κβάντισης, καθώς και η δυαδική μορφή (ψηφιακή μορφή) του αναλογικού σήματος.

Επίπεδο κβάντωσης	Δυαδική μορφή	Επίπεδο κβάντωσης	Δυαδική μορφή
0	0000	-1	1001
1	0001	-2	1010
2	0010	-3	1011
3	0011	-4	1100
4	0100		

Πίνακας 4: Δυαδική αναπαράσταση επιπέδων κβάντωσης.

Δείγμα	Τιμή κβάντωσης	Δυαδική μορφή
S <sub>1</sub>	2	0010
S <sub>2</sub>	4	0100
S <sub>3</sub>	3	0011
S <sub>4</sub>	2	0010
S <sub>5</sub>	3	0011
S <sub>6</sub>	3	0011
S <sub>7</sub>	2	0010
S <sub>8</sub>	0	0000
S <sub>9</sub>	-2	1010
S <sub>10</sub>	-4	1100
S <sub>11</sub>	-4	1100
S <sub>12</sub>	-3	1011
S <sub>13</sub>	-1	1001

Πίνακας 5: Δυαδική αναπαράσταση ψηφιοποιημένου σήματος (σύνολο bits: 52)

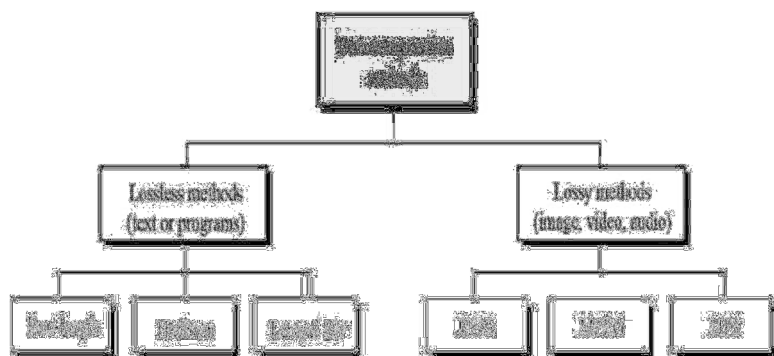
Πηγή:

Μάρκος Δεβρινός, Δημήτρης Κούης : Βασικές αρχές και Τεχνολογίες Επιστήμης της Πληροφορικής

## 4.ΣΥΜΠΙΕΣΗ

Η δημιουργία μιας εγγραφής ψηφιακού ήχου μπορεί να οδηγήσει σε πολύ μεγάλα μεγέθη αρχείων, πράγμα το οποίο περιορίζει τις πρακτικές χρήσεις της τεχνολογίας. Τα τελευταία χρόνια οι χρήστες θέλουν να έχουν ψηφιακό ήχο με βέλτιστη ποιότητα και να καταλαμβάνει όσο το δυνατόν λιγότερο χώρο. Γι' αυτό το λόγο, τα περισσότερα αρχεία ήχου χρησιμοποιούν κάποια μορφή συμπίεσης, αφαιρώντας ορισμένες πληροφορίες ήχου για να μειώσουν το μέγεθος του αποθηκευμένου αρχείου. Επιπλέον η συμπίεση είναι χρήσιμη για τη μεταφορά των δεδομένων μέσω του Διαδικτύου ή μεταξύ συσκευών. Έχουν αναπτυχθεί διάφοροι τύποι αλγορίθμων συμπίεσης που στοχεύουν στη μείωση των ψηφιακών δεδομένων ήχου για να μειώσουν τον αριθμό των bits που απαιτούνται για την αναπαραγωγή. Η διαδικασία συμπίεσης καλείται κωδικοποίηση ενώ το αντίστροφο καλείται αποκωδικοποίηση. Υπάρχουν δύο τύποι δεδομένων συμπίεση: lossless (χωρίς απώλειες) και lossy( απώλεια)

Η συμπίεση μπορεί να μειώσει τον όγκο των δεδομένων που στέλνονται ή αποθηκεύονται, με την ελαχιστοποίηση του ενυπάρχοντος πλεονασμού. Ο πλεονασμός παρουσιάζεται κατά τη δημιουργία των δεδομένων. Με τη διαδικασία της συμπίεσης η μεταφορά και η αποθήκευση γίνονται με πιο αποδοτικό τρόπο, ενώ παράλληλα διατηρείται η ακεραιότητα των δεδομένων.



Σχήμα 23: Μέθοδοι συμπίεσης δεδομένων

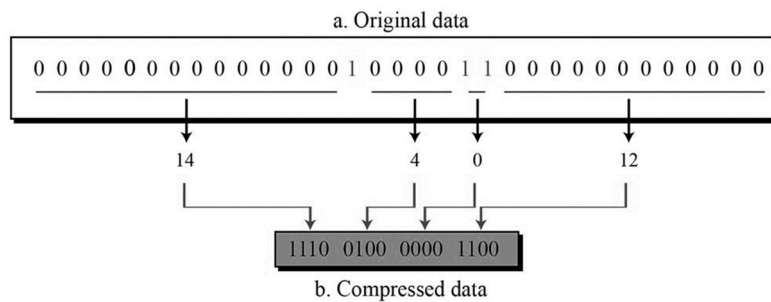
### Μη απωλεστική συμπίεση (lossless compression)

Στην μη απωλεστική συμπίεση (lossless compression) διατηρείται η ακεραιότητα των δεδομένων. Τα αρχικά δεδομένα και τα δεδομένα μετά τη συμπίεση και την αποσυμπίεση είναι ακριβώς τα ίδια. Δεν κρίνεται με βάση την ποιότητα της συμπίεσης, αλλά κυρίως για την αποτελεσματικότητα και την ταχύτητα (απόδοση) της. Ορισμένοι αλγόριθμοι βελτιστοποιούν περισσότερο το βαθμό συμπίεσης δηλαδή τη μείωση του μεγέθους του αρχείου, ενώ άλλοι βελτιστοποιούν περισσότερες την ταχύτητα συμπίεσης και αποσυμπίεσης.

### Αλγόριθμος RLE (Run Length Encoding)

Ο αλγόριθμος RLE είναι ίσως η απλούστερη μέθοδος συμπίεσης. Μπορεί να χρησιμοποιηθεί για τη συμπίεση δεδομένων από οποιαδήποτε συνδυασμός συμβόλων.

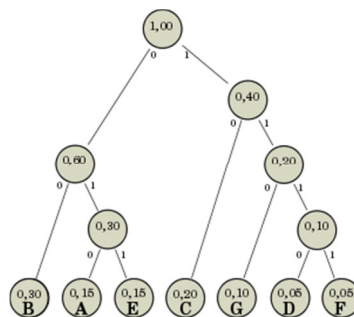
Δεν χρειάζεται να γνωρίζουμε τη συχνότητα εμφάνισης συμβόλων και μπορεί να είναι πολύ αποτελεσματικός εάν τα δεδομένα αντιπροσωπεύονται ως 0s και 1s. Σύμφωνα με τη μέθοδο αυτή, διατρέχεται η ακολουθία των bytes που αποτελούν τα δεδομένα προς συμπίεση και εντοπίζονται οι διαδοχικές επαναλήψεις του ίδιου χαρακτήρα. Στη συνέχεια αντικαθίστανται οι συνεχόμενες επαναλήψεις με το πλήθος τους, ακολουθούμενο από τον χαρακτήρα. Η μέθοδος μπορεί να είναι ακόμη πιο αποτελεσματική εάν τα δεδομένα που χρησιμοποιεί μόνο δύο σύμβολα (για παράδειγμα 0 και 1) στο bit του και ένα σύμβολο είναι συχνότερο από το άλλο.



Σχήμα 24: RLE για 2 σύμβολα

### Αλγόριθμος του Huffman

Η κωδικοποίηση Huffman είναι μια δημοφιλής μέθοδος για τη συμπίεση δεδομένων με κωδικούς μεταβλητού μήκους. Λαμβάνοντας υπόψη ένα σύνολο συμβόλων δεδομένων και τις συχνότητες εμφάνισής τους. Η μέθοδος κατασκευάζει ένα σύνολο κωδικών λέξεων μεταβλητού μήκους με το μικρότερο μέσο μήκος και τα εκχωρεί στα σύμβολα. Πιο αναλυτικά ο αλγόριθμος ξεκινά με την κατασκευή μιας λίστας με όλα τα σύμβολα κατά φθίνουσα σειρά της συχνότητάς τους. Στη συνέχεια κατασκευάζει, από κάτω προς τα πάνω ένα δέντρο. Σε κάθε βήμα δύο από τα σύμβολα με τις μικρότερες συχνότητες επιλέγονται, προστίθενται στην κορυφή του δέντρου, τα παλιά διαγράφονται από τη λίστα και έχουν αντικατασταθεί με το καινούριο σύμβολο. Όταν η λίστα μειώνεται σε ένα σύμβολο (που αντιπροσωπεύει όλα τα προηγούμενα), το δέντρο είναι πλήρες.



Σχήμα 25: Αλγόριθμος του Huffman

## Απώλεια συμπίεσης(lossy)

Η απώλεια συμπίεσης(lossy) διαφέρει από την αντίστοιχη, χωρίς απώλειες συμπίεσης, όπως υποδηλώνει το όνομά της ,κάποια ποσότητα δεδομένων μπορεί να χαθεί στη διαδικασία. Έτσι, μετά από έναν κύκλο συμπίεσης / αποσυμπίεσης, το σύνολο δεδομένων θα τροποποιηθεί από το ασυμπίεστο πρωτότυπο και ενδέχεται να χαθούν πληροφορίες, αυτό γίνεται γιατί εστιάζει περισσότερο στην εξοικονόμηση χώρου, διατηρώντας βέβαια την ακρίβεια των δεδομένων. Η απώλεια συμπίεσης χρησιμοποιείται συχνότερα σε βίντεο, ήχο και πολλούς τύπους αρχείων εικόνας. Για παράδειγμα, σε ένα MP3 , η συμπίεση απώλειας μπορεί να αφαιρέσει τμήματα του αρχείου ήχου που μόλις ακούγεται από το ανθρώπινο αυτί. Με αυτόν τον τρόπο, διατηρείται σχεδόν όλη η ποιότητα ήχου μειώνοντας παράλληλα το μέγεθος του αρχείου.

## MPEG (Moving Picture Experts Group) Audio

Το ηχητικό κομμάτι του MPEG αποτελεί μια οικογένεια τριών διαφορετικών τεχνικών κωδικοποίησης και συμπίεσης ήχου. Αυτές οι οικογένειες ονομάζονται MPEG-Audio Layer-1, Layer-2, Layer-3. Το επίπεδο III δίνει το γνωστό πρότυπο mp3 (MPEG-1, layerIII).

Το MPEG-1 Layer I ή II Audio είναι ένας γενικός κωδικοποιητής υποζώνης που λειτουργεί σε ρυθμούς δυαδικών ψηφίων στην περιοχή από 32 έως 448 kb / s και υποστηρίζει συχνότητες δειγματοληψίας 32, 44,1 και 48 kHz. Τα επίπεδα MPEG-1 I και II (MP1 ή MP2) είναι αισθητήρες κωδικοποιητή ήχου για περιεχόμενο ήχου 1 ή 2 καναλιών. Το Layer I έχει σχεδιαστεί για εφαρμογές που απαιτούν τόσο αποκωδικοποίηση χαμηλής πολυπλοκότητας όσο και κωδικοποίηση. Το επίπεδο II παρέχει υψηλότερη απόδοση συμπίεσης για ελαφρώς υψηλότερη πολυπλοκότητα.

Το MPEG-1 Layer 3 (ή MP3) είναι ένας κωδικοποιητής ήχου 1 ή 2 καναλιών ο οποίος παρέχει εξαιρετική συμπίεση μουσικών σημάτων. Σε σύγκριση με το Layer 1 και Layer 2, παρέχει μεγαλύτερη απόδοση συμπίεσης, μπορεί να συμπίεσει δεδομένα υψηλής ποιότητας ήχου κατά 12 φορές ενώ διατηρεί υψηλή ποιότητα ήχου. Γενικά, το MP3 είναι κατάλληλο για εφαρμογές που περιλαμβάνουν αποθήκευση ή μετάδοση μονοφωνικής ή στερεοφωνικής μουσικής ή άλλων σημάτων ήχου. Δεδομένου ότι εφαρμόζεται σχεδόν σε όλες τις ψηφιακές συσκευές ήχου, η αναπαραγωγή είναι πάντα εξασφαλισμένη. Το Layer 3, ή όπως ονομάζεται σήμερα "mp3", είναι η πιο διαδεδομένη μορφή κωδικοποίησης ήχου για την αποθήκευση μουσικής σε πλατφόρμες υπολογιστών και τη μετάδοση μουσικής μέσω του διαδικτύου.

## Waveform Audio File Format (WAV)

Είναι ένα πρότυπο για την αποθήκευση των bitstreams που έχει αναπτυχθεί από τη Microsoft .Χρησιμοποιείται κυρίως για την αποθήκευση ακατέργαστων και ασυμπίεστων δεδομένων ήχου με χρήση γραμμικής διαμόρφωσης παλμικού κώδικα (LPCM) ως κωδικοποίηση bitstream.

## Free Lossless Audio Codec (FLAC)

Είναι ένα ηχητικό σχήμα παρόμοιο με το MP3, αλλά χωρίς απώλειες, που σημαίνει ότι ο ήχος συμπιέζεται σε FLAC χωρίς απώλεια ποιότητας. Αυτό είναι παρόμοιο με τοπώς λειτουργεί το Zip. Η κωδικοποίηση δεδομένων ήχου (PCM) δεν προκαλεί απώλεια πληροφοριών και ο αποκωδικοποιημένος ήχος είναι bit-for-bit ομοιόμορφος με αυτό που εισέρχεται στον κωδικοποιητή.

#### Advanced Audio Coding (AAC)

Το AAC είναι ένα πρότυπο κωδικοποίησης για την απώλεια συμπίεσης ήχου που αναπτύχθηκε ως διάδοχος του MP3. Μερικές βελτιώσεις που περιλαμβάνει είναι περισσότερες συχνότητες δειγμάτων, αυθαίρετες ταχύτητες δυαδικών ψηφίων και μεταβλητά μήκη πλαισίων και υψηλότερα για τα στάσιμα και τα παροδικά σήματα. Λόγω μεγαλύτερης ευελιξίας από το MP3, AAC επιτυγχάνει μια πιο αποτελεσματική συμπίεση. Ο ρυθμός δυαδικών ψηφίων κυμαίνεται από 16 έως 320 kbit / s.

#### Windows Media Audio (WMA)

Το WMA είναι μια μορφή αρχείου που αναπτύχθηκε από τη Microsoft για την κωδικοποίηση ψηφιακών αρχείων ήχου. Δυνατότητα της ίδιας ποιότητας με ένα MP3, αλλά με ταχύτερο ρυθμό συμπίεσης, το WMA φτάνει σε ήχο υψηλής ποιότητας στα 64 kbps.

## 5.ΨΗΦΙΑΚΉ ΕΠΕΞΕΡΓΑΣΙΑ ΉΧΟΥ

Η ανταλλαγή της πληροφορίας στα ψηφιακά συστήματα πραγματοποιείται με την εκπομπή και τη λήψη σημάτων. Γενικά ως σήμα ορίζεται ένα φυσικό μέγεθος το οποίο μεταβάλλεται σε σχέση με το χρόνο ή το χώρο ή με οποιαδήποτε άλλη ανεξάρτητη μεταβλητή. Κάποια παραδείγματα: σήμα ομιλίας, ραδιοφωνικά σήματα κ.τ.λ .

Τα σήματα γίνονται αντικείμενο επεξεργασίας από τα συστήματα , τα οποία είναι τεχνικές οντότητες (π.χ αλγόριθμοι).Έτσι ένα σύστημα έχει ως είσοδο ένα η περισσότερα σήματα και ως έξοδο ένα η περισσότερα σήματα , τα οποία είναι αποτέλεσμα της επεξεργασίας που λαμβάνει χωρά στο εσωτερικό του .Στόχος της επεξεργασίας σήματος είναι η άμεσα αξιοποίηση του από τον τελικό χρήστη και η εφαρμογή του σε κάθε μορφής συστήματα ψηφιακής επικοινωνίας, ψηφιακής εγγραφής, εποχή μας βρίσκονται στο επίκεντρο των τεχνολογικών εξελίξεων για την βελτίωση της ποιότητας και εξάλειψη του θορύβου ,αποθήκευση και μετάδοση. Κάποιες χαρακτηριστικές εφαρμογές επεξεργασίας είναι η απομάκρυνση θορύβου είναι η απλούστερη και συχνότερη περίπτωση επεξεργασίας που εφαρμόζεται στην πράξη. Τεχνικές τύπου MPEG μετασχηματίζουν την αρχική πληροφορία με κατάλληλη επεξεργασία και κατόπιν συμπιέζουν σημαντικά τον όγκο της.Στην περίπτωση αυτή, το ζητούμενο δεν είναι να βελτιώσουμε την ποιότητα του σήματος απομακρύνοντας κάποιο θόρυβο, αλλά να μειώσουμε με τη βοήθεια μετασχηματισμού Fourier τον όγκο των δεδομένων ή αντίστροφα από το μετασχηματισμένο και συμπιεσμένο σήμα να ανακτήσουμε το αρχικό. Χωρίς

επεξεργασία, τα συστήματα επικοινωνίας, θα ήταν αδύνατο να έχουν ικανοποιητική απόδοση.

Από μαθηματική άποψη, ένα σήμα εκφράζεται ως συνάρτηση μιας ή περισσοτέρων ανεξαρτητών μεταβλητών.

$$t \rightarrow x(t)$$

Η ανεξάρτητη μεταβλητή  $t$  είναι συνήθως ο χρόνος, ή οποία μπορεί να έχει και άλλη φυσική σημασία. Με  $x(t)$  συμβολίζεται η τιμή του σήματος τη χρονική στιγμή  $t$ .

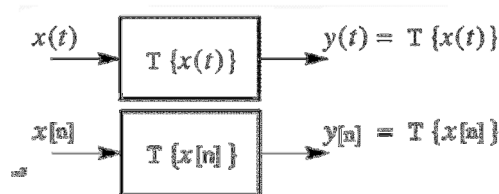
Σύστημα είναι ένας τελεστής που δρά σε σήματα και δημιουργεί νέα σήματα ή νέες αναπαραστάσεις σημάτων π.χ. Ένα σύστημα αναπαραγωγής CD μετατρέπει το σήμα της κωδικοποιημένης ομιλίας σε ήχο.

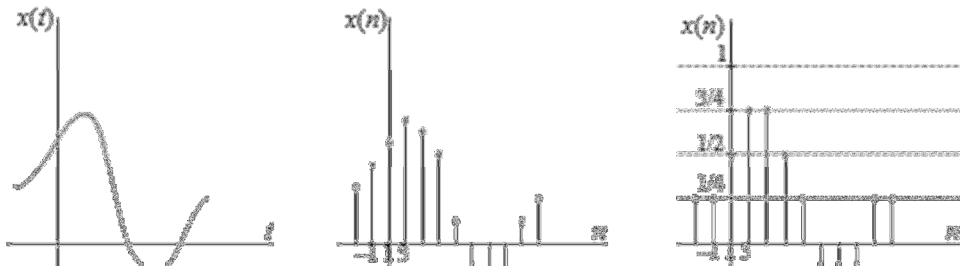
Σχήμα 26 : (επάνω) Σύστημα διακριτού χρόνου και (κάτω) σύστημα συνεχούς χρόνου

Τα σήματα χωρίζονται σε κατηγορίες ανάλογα με τον τύπο και το πλήθος των μεταβλητών. Έτσι, ανάλογα με το αν η ανεξάρτητη μεταβλητή είναι συνεχής ή διακριτή, έχουμε σήματα συνεχούς ή διακριτού χρόνου. Η ψηφιακή επεξεργασία σήματος ασχολείται με την ψηφιακή αναπαράσταση των σημάτων και την ανάλυση, τροποποίηση και εξαγωγή πληροφοριών από αυτά, με τη βοήθεια ψηφιακών επεξεργαστών. Τα σήματα ταξινομούνται ανά κατηγορίες: στα σήματα συνεχούς χρόνου και σήματα διακριτού χρόνου.

Σήματα συνεχούς χρόνου  $x(t)$  με  $t$  είναι η συνεχής μεταβλητή (continuous time), τα σήματα αυτά αναφέρονται και ως σήματα συνεχούς χρόνου συνεχούς πλάτους. Η ανεξάρτητη μεταβλητή είναι συνεχής, δηλαδή τα σήματα αυτά ορίζονται για οποιαδήποτε τιμή της ανεξάρτητης μεταβλητής. Η εξαρτημένη μεταβλητή, δηλαδή το πλάτος (amplitude) του σήματος, είναι και αυτή συνεχής.

Στα σήματα διακριτού χρόνου (discrete time) που συμβολίζονται με  $x[n]$  όπου  $n =$  ακέραιος αριθμός. Τα σήματα αυτά αναφέρονται και ως σήματα διακριτού χρόνου συνεχούς πλάτους. Η ανεξάρτητη μεταβλητή είναι διακριτή, δηλαδή τα ορίζονται μόνο για τιμές της ανεξάρτητης άλλα λόγια, η μεταβλητή παίρνει συγκεκριμένες μεταβλητής. Με ανεξάρτητη τιμές από ένα διακριτό σύνολο τιμών. Η εξαρτημένη μεταβλητή, δηλαδή το πλάτος του σήματος, είναι συνεχής.





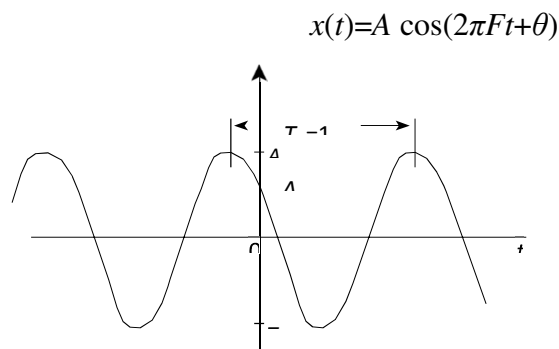
Σχήμα 27 : Τύποι σημάτων: αναλογικό, διακριτό και ψηφιακό

Σήμα συνεχούς χρόνου  $x(t)$  μπορεί να πάρει οπουδήποτε τιμή στο συνεχές διάστημα  $(a,b)$  οπου μπορεί να είναι  $-\infty$  και το  $b$  το  $+\infty$  , ονομάζεται αναλογικό .Αν σήμα διακριτού χρόνου  $x(n)$  μπορεί να πάρει μόνο πεπερασμένο πλήθος διακριτών τιμών , τότε ονομάζεται ψηφιακό σήμα. Ακόμα πολύ συχνά το σήμα διακριτού χρόνου  $x(n)$  σχηματίζεται με δειγματοληψία σήματος συνεχούς χρόνου έτσι ώστε ισχύει  $x(n)=x(nT_s)$  οπου  $T_s$  είναι το χρονικό διάστημα δειγματοληψίας . ( θα αναφερθούμε παρακάτω για τη δειγματοληψία).

Θα αναφερθούμε στην κυματομορφή ενός απλού ήχου , του ημίτονου, το οποίο αποτελείται από μία μόνο συχνότητα . Ένα ημιτόνιο σήμα περιγράφεται από την ακόλουθη εξίσωση:

$$x_a(t) = A \cos(2\pi f t + \theta), \quad -\infty < t < \infty$$

όπου  $A$  το πλάτος (amplitude) του ημιτονοειδούς,  $f$  η συχνότητα σε ακτίνια ανά δευτερόλεπτο (rad/s) και  $\theta$  η γωνία φάσης .



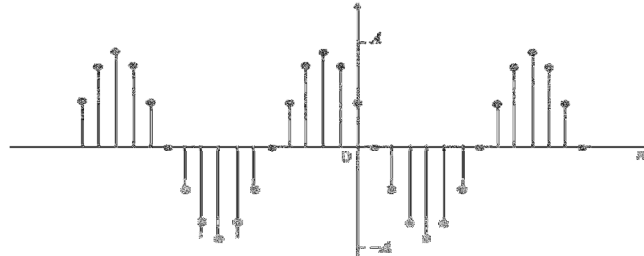
Σχήμα 28: Αναλογικό ημιτονοειδές σήμα

Ένα ημιτονοειδές σήμα διακριτού χρόνου μπορεί να εκφραστεί ως:

$$x(n) = A \cos(2\pi f n + \theta), \quad -\infty < n < \infty$$

όπου  $n$  ακέραιη μεταβλητή, η οποία αντιπροσωπεύει τον αριθμό του δείγματος,  $A$  το πλάτος του σήματος,  $f$  η συχνότητα του σήματος σε ακτίνια ανά δείγμα και  $\theta$  η φάση σε ακτίνια.

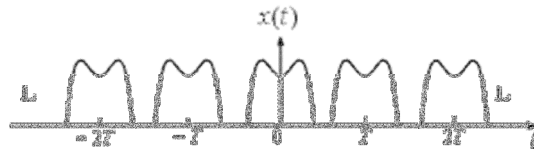
$$x(n) = A \cos(\omega n + \theta)$$



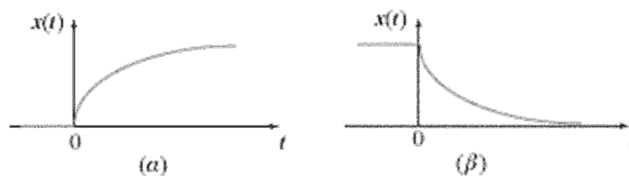
Σχήμα 29 : Ημιτονοειδές σήμα διακριτού χρόνου

### Τύποι σημάτων

Περιοδικό είναι ένα σήμα το οποίο ικανοποιεί τη συνθήκη  $x(t+T_0)=x(t)$ . Ο μικρότερος θετικός αριθμός  $T_0$  ονομάζεται περίοδος και το αντίστροφο της περιόδου θεμελιώδης συχνότητα  $f_0=1/T_0$  και από την παραπάνω συνθήκη προκύπτει  $x(t+nT_0)=x(t)$ . Τα σήματα που δεν ικανοποιούν τη συνθήκη ονομάζονται μη-περιοδικά



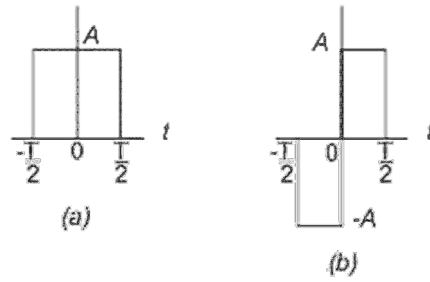
Ένα σήμα  $x(t)$  λέγεται αιτιατό, εάν είναι μηδενικό για αρνητικές τιμές του χρόνου  $t$ , δηλαδή,  $x(t) = 0 \quad t < 0$ . Στην αντίθετη περίπτωση, το σήμα λέγεται μη αιτιατό



Σχήμα 30: Παράδειγμα (α) αιτιατού σήματος και (β) μη αιτιατού σήματος.

Άρτια ονομάζεται ένα σήμα ένα σήμα  $x(t)$  για το οποίο ισχύει  $x(t)=x(-t)$  δηλαδή το σήμα είναι συμμετρικό ως προς κατακόρυφο άξονα και περιττό ονομάζεται ένα σήμα το οποίο είναι συμμετρικό ως προς την αρχή των αξόνων  $x(t)=-x(-t)$

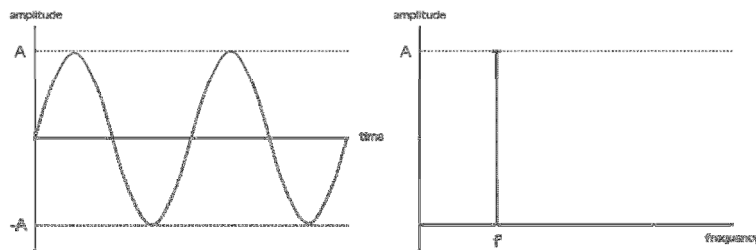




Σχήμα 31: (a) Άρτια σήμα και (b) περιττό

## Φάσμα συχνοτήτων και ανάλυση Fourier

Το σήμα που λαμβάνεται μπορεί να αναλυθεί προκειμένου να βρεθεί από πόσες συχνότητες αποτελείται και τη ισχύ έχει καθεμιά από αυτές. Αυτή η ανάλυση του φάσματος ονομάζεται ανάλυση Fourier. Η συμβολή του Fourier είναι ιδιαίτερα σημαντική αφού παρέχει δυνατότητα μελέτης και σχεδιασμού. Ορίζεται με διαφορετικό τρόπο ανάλογα αν το σήμα είναι περιοδικό ή όχι. Στην ουσία ο μετασχηματισμός Fourier είναι το εργαλείο για να μεταβούμε από την ανάλυση στο πεδίο του χρόνου σε ανάλυση στο πεδίο των συχνοτήτων και αντίστροφα.



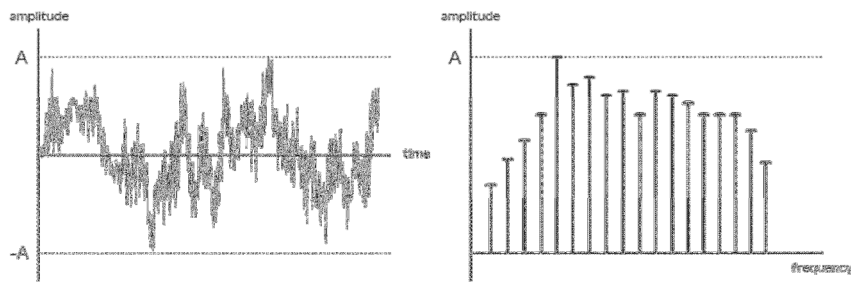
Σχήμα 32: κυματομορφή του ημιτόνου στο πεδίο χρόνου και η απεικόνιση στο πεδίο της συχνότητας

Η μετάβαση από το πεδίο του χρόνου στο πεδίο της συχνότητας επιτυγχάνεται με το μετασχηματισμό Fourier διακριτού χρόνου, ενώ η μετάβαση από το πεδίο της συχνότητας στο πεδίο του χρόνου γίνεται με τον αντίστροφο μετασχηματισμό Fourier διακριτού χρόνου. Κατά τη μετάβαση από το πεδίο του χρόνου στο πεδίο της συχνότητας η ενέργεια του σήματος διατηρείται (θεώρημα Parseval). Πρόκειται στην ουσία για την αναπαράσταση της ίδιας πληροφορίας με διαφορετικό τρόπο.

Ο DTFT ενός σήματος διακριτού χρόνου είναι μία συνεχής και περιοδική συνάρτηση.

Ο DFT ενός σήματος διακριτού χρόνου πεπερασμένης διάρκειας είναι μία περιοδική ακολουθία.

Ο ταχύς μετασχηματισμός Fourier (FFT) είναι κάθε αλγόριθμος ο οποίος έχει υπολογιστική πολυπλοκότητα μικρότερη από εκείνη του απευθείας υπολογισμού του DFT. Η συνήθης υπολογιστική πολυπλοκότητα του FFT είναι της τάξεως του  $M \log_2 N$ , ενώ εκείνη του DFT είναι της τάξεως του  $N^2$ .



Σχήμα 33: Σύνθετο σύνολο διαφορετικών ημιτονοειδών κυμάτων. Στην περιοχή συχνοτήτων μπορούμε να δούμε πιο καθαρά τι συμβαίνει και ορισμένες περιοχές συχνοτήτων. Αυτό το είδος απεικόνισης μας επιτρέπει να ελέγξουμε καλύτερα τον ήχο, ειδικά εάν βρισκόμαστε στη διαδικασία της διαμόρφωσής του

### Μετασχηματισμός Fourier διακριτού χρόνου

Ο μετασχηματισμός Fourier διακριτού χρόνου ενός σήματος διακριτού χρόνου  $x(n)$  είναι η αναπαράσταση του σήματος αυτού ως συνδυασμού μιγαδικών εκθετικών ακολουθιών της μορφής  $\{e^{-j\omega n}\}$ , όπου  $\omega$  μεταβλητή, γνωστή και ως συχνότητα. Ο μετασχηματισμός εφαρμόζεται γενικά σε μη περιοδικά σήματα διακριτού χρόνου άπειρου μήκους. Ο DTFT τέτοιων σημάτων είναι συνεχής και περιοδικός.

Ο DTFT  $X(e^{j\omega})$  της ακολουθίας  $x(n)$  ορίζεται

$$X(e^{j\omega}) = F\{x(n)\} = \sum_{-\infty}^{\infty} x(n)e^{-j\omega n}$$

Ενώ ο αντίστροφος μετασχηματισμός Fourier διακριτού χρόνου ορίζεται ως

$$x(n) = F^{-1}\{X(e^{j\omega})\} = \frac{1}{2\pi} \int X(e^{j\omega})e^{j\omega n} d\omega$$

Η συνάρτηση  $X(e^{j\omega})$  είναι μιγαδική και μπορεί να γραφεί

$$X(e^{j\omega}) = X_r(e^{j\omega}) + jX_i(e^{j\omega})$$

όπου  $X_r(e^{j\omega})$  και  $X_i(e^{j\omega})$  είναι πραγματικές συναρτήσεις ως προς  $\omega$ , και αποτελούν το πραγματικό και φανταστικό μέρος της  $X(e^{j\omega})$

Ιδιότητες του μετασχηματισμού Fourier διακριτού χρόνου είναι:

Γραμμικότητα, ο μετασχηματισμός Fourier είναι ένας γραμμικός μετασχηματισμός, κατάλληλος για τη μελέτη γραμμικών συστημάτων. Έτσι, ο DTFT του γραμμικού συνδυασμού δύο ή περισσότερων σημάτων ισούται με το γραμμικό συνδυασμό του κάθε σήματος. Ένα μη περιοδικό σήμα  $a_1x_1(n) + a_2x_2(n)$  στο μετασχηματισμό είναι  $a_1X_1(e^{j\omega}) + a_2X_2(e^{j\omega})$

Μετατόπιση στο χρόνο και συχνότητα

Συνέλιξη , η οποία μας λέει ότι η συνέλιξη δύο ακολουθιών στο πεδίο του χρόνου, ισοδυναμεί με το γινόμενο των φασμάτων τους στο πεδίο της συχνότητας.  $x_1(n)*x_2(n)$  σε  $X_1(e^{j\omega})X_2(e^{j\omega})$

Θεώρημα του Parseval , που αναφέρεται στη διατήρηση της ενέργειας κατά τη μεταβίβαση από το πεδίο του χρόνου στο πεδίο της συχνότητας.

Στη προηγούμενη ενότητα μιλήσαμε για τη δειγματοληψία και είδαμε ότι έχει να κάνει με την μετατροπή ενός αναλογικού σήματος σε ψηφιακό .Γνωρίζοντας πλέον κάποια πράγματα για το μετασχηματισμό Fourier και το πεδίο της συχνότητας θα επανέλθουμε στο θέμα της δειγματοληψίας και θα εξετάσουμε τα σήματα στο πεδίο της συχνότητας.

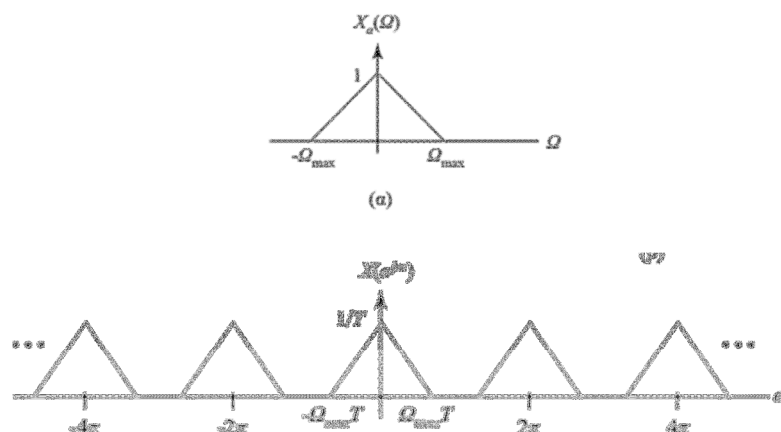
Έστω λοιπόν, ότι το αναλογικό σήμα  $x(t)$  τροφοδοτείται στην είσοδο ενός ιδανικού δειγματολήπτη. Η έξοδος αυτού θα είναι η ακολουθία  $x(n)$ , τα στοιχεία της οποίας αντιστοιχούν στις μετρήσεις του πλάτους της  $x(t)$  ανά χρονικά διαστήματα  $T$  δευτερολέπτων

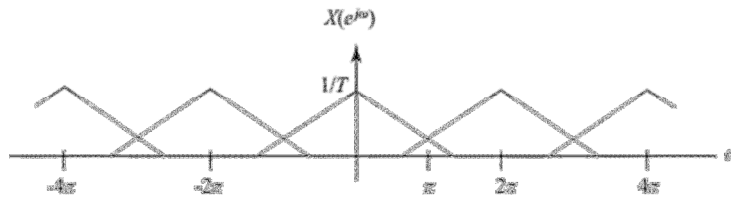
Η παράμετρος  $T$  αποτελεί την περίοδο δειγματοληψίας και το αντίστροφο αυτής τη συχνότητα ή το ρυθμό δειγματοληψίας ( $F_s = 1/T$ ), δηλαδή, το πλήθος των δειγμάτων που λαμβάνονται στη μονάδα του χρόνου.

Για να ανακατασκευάσουμε το αναλογικό σήμα  $x(t)$  από την ακολουθία  $x(n)$  θα πρέπει τα φασματικά χαρακτηριστικά του αναλογικού σήματος να διατηρούνται μετά τη δειγματοληψία. Έστω  $X_a(j\Omega)$  ο μετασχηματισμός Fourier του σήματος  $x(t)$  και  $X(e^{j\omega})$  ο μετασχηματισμός Fourier διακριτού χρόνου της ακολουθίας  $x(n)$ . Αποδεικνύεται ότι

$$X(e^{j\omega}) = \frac{1}{T} \sum_{\kappa=-\infty}^{\infty} X_a(j\frac{\omega-2\pi\kappa}{T})$$

Η σχέση αυτή δηλώνει ότι το φάσμα του σήματος που προέκυψε από τη δειγματοληψία, αποτελείται από περιοδικές επαναλήψεις του φάσματος του αναλογικού σήματος.





Έστω ότι έχουμε το παραπάνω σχήμα, όλες οι συχνότητες κυμαίνονται μεταξύ  $-\Omega_{\max}$  και  $\Omega_{\max}$ . Για να ανακατασκευάσουμε το αρχικό σήμα από τα δείγματά του, θα πρέπει οι περιοδικές επαναλήψεις να μην καλύπτουν η μία την άλλη. Θα πρέπει να ικανοποιείται η συνθήκη  $F_s < 2F_{\max}$  ( $f_s \geq 2f_c$ ) που είναι το γνωστό μας θεώρημα δειγματοληψίας. Και για να διασφαλίσουμε το γεγονός ότι το σήμα εισόδου είναι περιορισμένου εύρους (band-limited) δηλαδή όλες οι συχνότητες κυμαίνονται μεταξύ  $-\Omega_{\max}$  και  $\Omega_{\max}$  ( $-\Omega_{\max} \leq \Omega \leq \Omega_{\max}$ ) τοποθετούμε πάντοτε ένα αναλογικό βαθυπερατό φίλτρο (analog lowpass filter) πριν από το σύστημα ψηφιακής επεξεργασίας σημάτων, για να αποκόψει τις συχνότητες που είναι μεγαλύτερες από  $F_s/2$ . Ένα παρόμοιο αναλογικό φίλτρο τοποθετείται και στην έξοδο του ψηφιακού συστήματος για τη σωστή ανακατασκευή του σήματος εξόδου, δηλαδή για την επιλογή μόνο του βασικού φασματικού περιεχομένου μεταξύ  $-F_s/2$  και  $F_s/2$ .

Στην περίπτωση που η συνθήκη αυτή δεν ικανοποιείται, όπως φαίνεται στο σχήμα τότε λέμε ότι έχουμε το φαινόμενο της φασματικής επικάλυψης για το οποίο αναφερθήκαμε στην παραπάνω ενότητα.

#### Διακριτός Μετασχηματισμός Fourier (DFT)

Ο διακριτός μετασχηματισμός Fourier χρησιμοποιείται στην επεξεργασία σημάτων με υπολογιστή καθώς ο μετασχηματισμός Fourier ενός σήματος διακριτού χρόνου  $x(n)$  είναι η συνεχής συνάρτηση  $X(e^{j\omega})$  και είναι δύσκολο η συνάρτηση αυτή να υπολογιστεί με τη χρήση ενός ψηφιακού επεξεργαστή-υπολογιστή. Αυτό που είναι εύκολο να υπολογιστεί, είναι δείγματα του φάσματος. Στην ουσία υπολογισμός του φάσματος σε διακριτές συχνότητες. Μπορούμε από ένα σήμα  $N$  δειγμάτων, να προκύψουν  $N$  δείγματα στη συχνότητα.

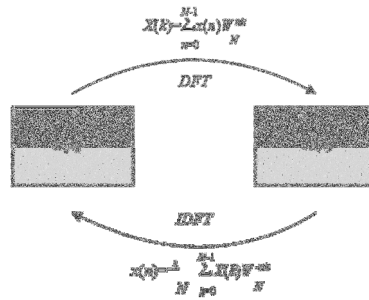
Το ζεύγος του DFT  $N$ -σημείων μπορεί να εκφραστεί ως  $x(n) \rightarrow x(k)$

$$x(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N} \quad n=0, 1, \dots, N-1$$

αντίστροφος διακριτός μετασχηματισμός Fourier (inverse DFT, IDFT)

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} x(k) e^{j2\pi kn/N} \quad n=0, 1, \dots, N-1$$

οι ακολουθίες  $x(n)$  και  $X(k)$  είναι ίδιου μήκους  $N$ .



Σχήμα33: Μετάβασης από το πεδίο του χρόνου στο πεδίο της συχνότητας και αντιστρόφως με χρήση του ζεύγους DFT.

### Ταχύς Μετασχηματισμός Fourier(FFT)

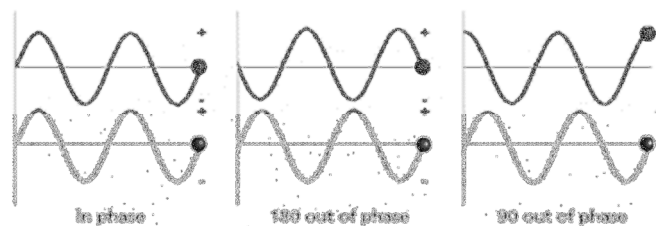
Ο ταχύς μετασχηματισμός Fourier (fast Fourier transform, FFT) δεν είναι τίποτε άλλο παρά ένας αποδοτικός αλγόριθμος για τον υπολογισμό του DFT. Στην πράξη, δεν υπάρχει μόνο ένας αλγόριθμος, αλλά πλήθος από διαφορετικούς αλγόριθμους που επιτυγχάνουν το σκοπό αυτό. Οι διαφορές τους βρίσκονται κυρίως στο πλήθος και στο είδος των πράξεων καθώς και στο μέγεθος της απαιτούμενης μνήμης. Όλοι όμως, έχουν ένα κοινό χαρακτηριστικό: χρειάζονται μόνο  $(N/2) \log_2 N$  μιγαδικούς πολλαπλασιασμούς για τον υπολογισμό ενός DFT  $N$ -σημείων.

Ουσιαστικό στοιχείο του FFT είναι η μειωμένη υπολογιστική πολυπλοκότητά του, δηλαδή, το συνολικό πλήθος πράξεων, δηλαδή από  $N^2$  σε  $N/2 \log_2 N = \frac{N \cdot v}{2}$  όπου  $N=2^v$

Για παράδειγμα αν το  $N=1024=2^{10}$  τότε για τον υπολογισμό του DFT χρειάζονται  $N^2=2^{20}=1.048.576$  μιγαδικοί πολλαπλασιασμοί ενώ ο FFT απαιτεί  $N=N \cdot v/2=1024 \cdot 10/2=5120$  πολλαπλασιασμούς, ο χρόνος υπολογισμού μειώνεται τουλάχιστον κατά δύο τάξεις μεγέθους.

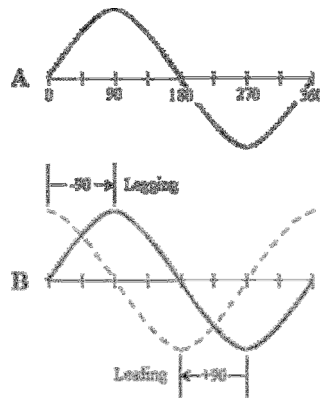
### Φάση

Η φάση είναι η στιγμιαία κατάσταση στην οποία βρίσκεται μια κυματομορφή και μετράται σε μοίρες .



Σχήμα 34: Φάση

Για να δούμε την έννοια της φάσης ας δούμε το σχήμα ,περιγράφει την σχέση φάσης ενός σήματος εισόδου και ενός σήματος εξόδου από επεξεργασία σήματος.Στο πρώτο σήμα εξόδου είναι σε φάση με το σήμα εισόδου. Και τα δύο σήματα περνούν από την τιμή μηδέν του χρόνου την ίδια στιγμή. Το δεύτερο σήμα εξόδου βλέπουμε ότι είναι εκτός φάσης  $180^\circ$  .Στο τρίτο σήμα εξόδου έχουμε διαφορά φάσης  $90^\circ$  . Και τα δύο σήματα περνούν από το μηδέν την ίδια χρονική στιγμή, αλλά με διαφορετικές κατευθύνσεις.

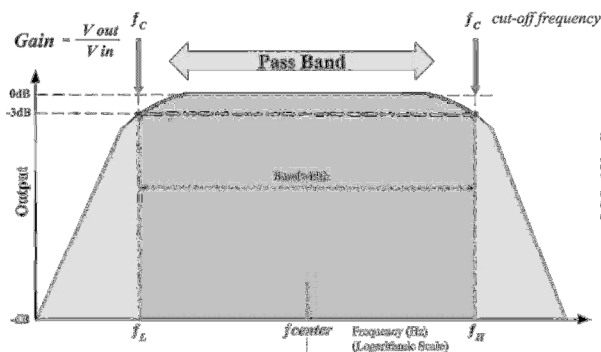


Ενώ στη μεταβολή της συχνότητας ενός ήχου παρατηρούμε μεταβολή του ύψους, στη μεταβολή της φάσης δεν παρατηρείται καμία αλλαγή στα χαρακτηριστικά ενός ήχου. Όταν όμως μεταβάλλονται οι φάσεις ενός σύνθετου ήχου κάθε φορά με διαφορετικό τρόπο, τότε μπορούμε να αντιληφθούμε την αλλαγή αυτή.

### Απόκριση συχνότητας

Σε όλες τις συσκευές στις οποίες παρατηρείται κέρδος δηλαδή εξασθένιση σήματος συναντάμε τον όρο απόκριση συχνότητας, που μας πληροφορεί για το πως μεταβάλλεται το κέρδος με τη μεταβολή της συχνότητας και επίσης ποιες είναι οι περιοχές συχνοτήτων που επιτρέπει να περάσουν ομοιόμορφα. Για παράδειγμα έχουμε έναν ενισχυτή που ενισχύει σταθερά από 50 έως 10000Hz. Παρατηρούμε ότι δεν επιτρέπει κάτω τον 50 Hz η πάνω από 10kHz και ενισχύει ομοιόμορφα όλες τις συχνότητες από 50 ως 10000Hz.

Ορισμένα χαρακτηριστικά της απόκρισης συχνότητας μέσα από ένα band-pass φίλτρο



Σχήμα 35: Απόκριση συχνότητας σε bandpass φίλτρο

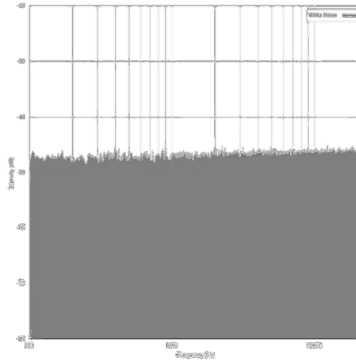
Γιαν είναι το κέρδος, δίδεται σε decibel. Το σημείο  $f_c$  λέγεται η συχνότητα αποκοπής χαμηλή συχνότητα συμβολίζεται με  $f_L$  και βρίσκεται στη κάτω συχνότητα όπου το κέρδος έχει μειωθεί  $-3\text{db}$  και  $f_H$  υψηλή συχνότητα αποκοπής. Πάνω από την  $f_L$  και την  $f_H$  το κέρδος μηδενίζεται. Εύρος ζώνης είναι η περιοχή μεταξύ των συχνοτήτων αποκοπής και η περιοχή διέλευσης έχει σταθερό κέρδος.

### Λευκός και ο ροζ θόρυβος

Στον ήχο, ο θόρυβος είναι γενικά οποιοσδήποτε δυσάρεστος ήχος και, οποιοσδήποτε ανεπιθύμητος ήχος που προστίθεται ακούσια σε έναν επιθυμητό ήχο. , πιο τεχνικά ο θόρυβος είναι ένα σύνολο τυχαίων σημάτων με ανεξάρτητες φάσεις και συχνότητες.

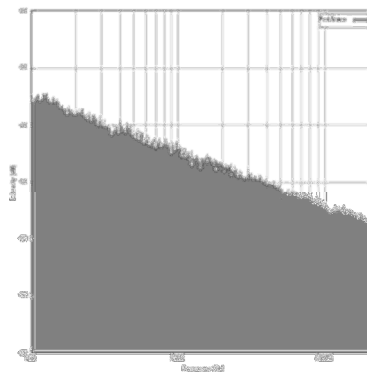
Ιδιαίτερο ενδιαφέρον παρουσιάζουν δύο συγκεκριμένοι τύποι θορύβου: ο λευκός και ο ροζ θόρυβος.

Λευκός θόρυβος (white noise) είναι ένα τυχαίο σήμα με σταθερή όμως φασματική πυκνότητα ισχύος. Δηλαδή το φάσμα του λευκού θορύβου είναι μία γραμμή παράλληλη στον άξονα των συχνοτήτων και η στάθμη ζώνης του παρουσιάζει μία αύξηση +3dB ανά οκτάβα.



Σχήμα 36: Λευκός θόρυβος

Ροζ θόρυβος (pink noise) είναι ένα σήμα που έχει συνεχές φάσμα με σταθερή ακουστική ισχύ ανά ζώνη μεταβλητού εύρους (οκτάβα, τριτοοκτάβα κτλ.). Η φασματική πυκνότητα ισχύος του ροζ θορύβου είναι αντιστρόφως ανάλογη της συχνότητας του σήματος.



Σχήμα 37: Ροζ θόρυβος

## Φίλτρα

Το ηχητικό φιλτράρισμα παίζει κεντρικό ρόλο στην επεξεργασία του ήχου. Η ανθρώπινη φωνητική οδός χρησιμεύει επίσης ως ηχητικό φίλτρο που τροποποιεί και διαμορφώνει τους ήχους που δημιουργούνται από τον λάρυγγα και άλλους αρθρωτές.

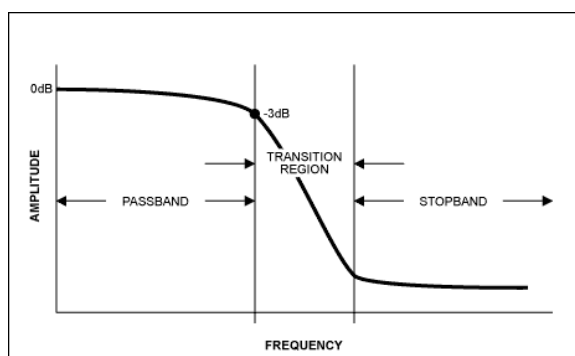
Για το λόγο αυτό, είναι πολύ σημαντικό να αναφέρουμε πώς λειτουργούν τα ηχητικά φίλτρα. Με την ευρύτερη έννοια, ο όρος φίλτρο αναφέρεται σε μια συσκευή ή ένα σύστημα που είναι επιλεκτικό για τα είδη των πραγμάτων που επιτρέπεται να περάσουν μέσα από τα είδη των πραγμάτων που εμποδίζονται. Στην επιστήμη της ομιλίας και ακοής έχουμε φίλτρα επιλεκτικής συχνότητας, συσκευές που επιτρέπουν τη διέλευση ορισμένων συχνοτήτων κατά την παρεμπόδιση ή την εξασθένηση άλλων συχνοτήτων. Με τον όρο εξασθένηση ,εννοούμε την εξασθένηση ή μείωση του πλάτους.

#### Χαρακτηριστικά των φίλτρων

Ζώνη διέλευσης (passband) είναι το διάστημα στο οποίο το φίλτρο αφήνει τις συχνοτήτες να περνούν ω<sub>p</sub> είναι η συχνότητα αποκοπής στη ζώνη διέλευσης σε rad/sec και A<sub>max</sub>(maximum passband ripple) που συμβολίζει το πλάτος της κυμάτωσης της ζώνης διέλευσης.

Ζώνη αποκοπής ή απόρριψης (stopband) είναι το διάστημα στο οποίο το φίλτρο δεν επιτρέπει της συχνοτήτες να περάσουν ω<sub>s</sub> είναι η συχνότητα αποκοπής στη ζώνη απόρριψης σε rad/sec Η μέγγιστη τιμή της κυμάτωσης στη ζώνη αποκοπής (stopband gain) δίνεται από τη σχέση 1/A ενώ η μικρότερη τιμή είναι ίση με το μηδέν 0 και A<sub>min</sub>. (minimum stopband ripple). Που συμβολίζει το πλάτος της κυμάτωσης στη ζώνη αποκοπής.

Ζώνη μετάβασης (transition region) είναι το διάστημα όπου ισχύει: Δω = ω<sub>s</sub> - ω<sub>p</sub> και είναι η περιοχή ανάμεσα στη ζώνη διέλευσης και τη ζώνη αποκοπής. (στο πεδίο της συχνότητας).

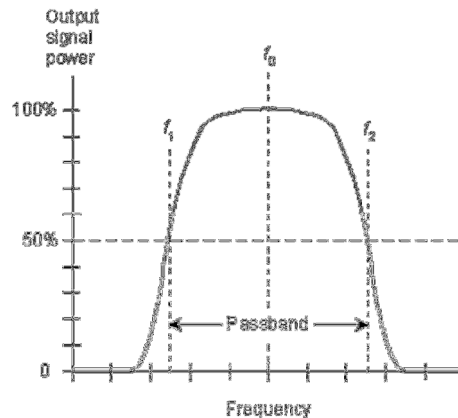


Σχήμα 38: Σε ένα lowpass filter βλέπουμε τη ζώνη διέλευσης (passband), ζώνη αποκοπής (stopband) και τη ζώνη μετάβασης (transition region)

Εύρος ζώνης - Bandwidth (BW) είναι το διάστημα ανάμεσα στη μικρότερη συχνότητα  $f_1$  ή  $\omega_1$  ή  $f_{\min}$  και τη μεγαλύτερη συχνότητα  $f_2$  ή  $\omega_2$  ή  $f_{\max}$  που περνάνε



ανεπηρέαστες μέσα από ένα φίλτρο. Όπου  $f_1$  και  $f_2$  είναι οι συχνότητες αποκοπής του φίλτρου και  $f_0$  ή  $f_{center}$  η κεντρική συχνότητα .

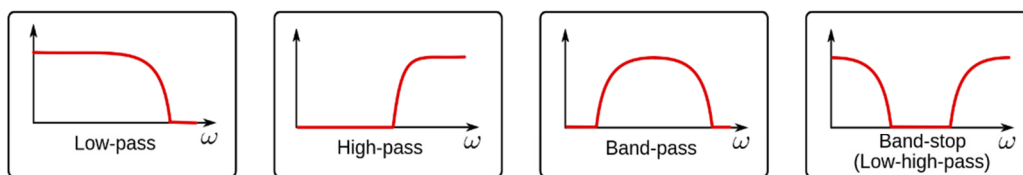


Σχήμα 39: Σε ένα band-pass φίλτρου βλέπουμε το εύρος ζώνης ( Bandwidth)

Τα φίλτρα ανάλογα με το είδος του σήματος που έχουν κατασκευαστεί να δέχονται και να επεξεργάζονται, χωρίζονται σε αναλογικά και σε ψηφιακά. Τα αναλογικά δέχονται αναλογικό σήμα, δηλαδή σήμα συνεχούς χρόνου και τα ψηφιακά, δέχονται ψηφιακό δηλαδή σήμα πεπερασμένου αριθμού δειγμάτων. Φίλτρα συνεχούς χρόνου και φίλτρα δειγματοσιμμένων δεδομένων αναφέρονται ως αναλογικά και φίλτρα διακριτού χρόνου ως ψηφιακά.

Τα ψηφιακά φίλτρα χωρίζονται σε δύο κατηγορίες ανάλογα με το μήκος της κρουστικής απόκρισης τους. Στα άπειρης Κρουστικής απόκρισης IIR και στα πεπερασμένης Κρουστικής απόκρισης FIR. Ανάλογα με τις συχνότητες που επιτρέπουν ή απορρίπτουν, χωρίζονται στα εξής είδη: Low-Pass, High-Pass, Band-Pass, Band-Stop και All-Pass.

Μια καμπύλη απόκρισης συχνότητας είναι ένα γράφημα που δείχνει πως διαφορετικές συχνότητες θα επηρεαστούν από το φίλτρο , όπως βλέπουμε στο παρακάτω σχήμα.



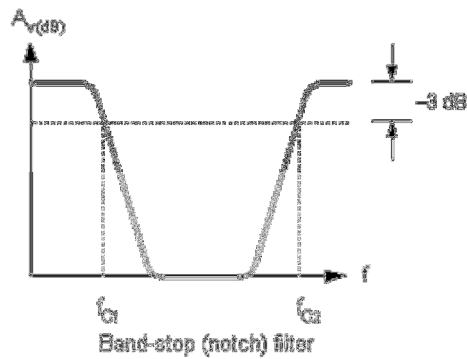
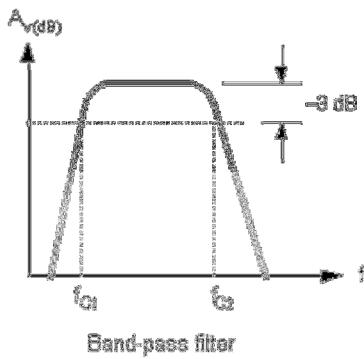
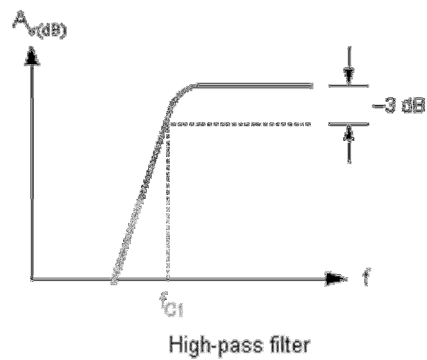
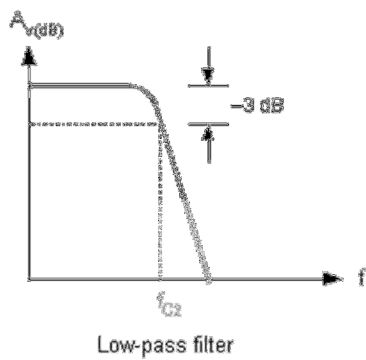
Σχήμα 40 Καμπύλη απόκρισης συχνότητας για low-pass,high-pass,band-pass,band-stop

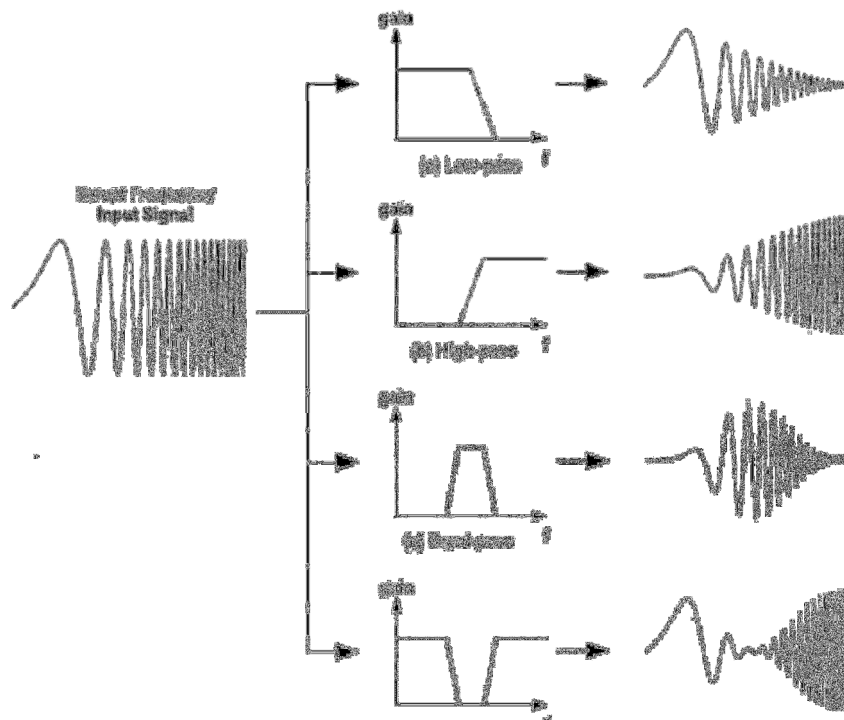
Το φίλτρο lowpass(χαμηλής διέλευσης) στα αριστερά επιτρέπει χαμηλές συχνότητες για να περάσουν, ενώ εξασθενεί ή αποκλείει υψηλότερες συχνότητες.

Το φίλτρο highpass (υψηλής διέλευσης) στη μέση έχει το αντίθετο αποτέλεσμα, επιτρέπει τη διέλευση υψηλών συχνοτήτων, ενώ εξασθενεί ή αποκλείει χαμηλότερες συχνότητες..

Το Band-Pass ή αλλιώς ζωνοδιαβατό, ζωνοπερατό, ή διέλευσης ζώνης φίλτρο επιτρέπει να διέρχονται μόνο οι συχνότητες μεταξύ των δύο συχνοτήτων αποκοπής  $f_{c1}$  και  $f_{c2}$  ώστε να υπάρχει μια ζώνη συχνοτήτων που θα περάσει (Bandwidth) ενώ οι εκατέρωθεν συχνότητες απορρίπτονται.

Το band-stop ή αλλιώς band-reject ή απόρριψης ζώνης, κάνει ακριβώς το αντίθετο: απορρίπτει μια ζώνη συχνοτήτων και επιτρέπει σε όλες τις συχνότητες κάτω από την  $f_{c1}$  και πάνω από την  $f_{c2}$  να περνούν, αποκόπτοντας μόνο το τμήμα μεταξύ  $f_{c1}$  και  $f_{c2}$ . Χωρίζονται σε ευρείας ζώνης απόρριψης και σε στενής ζώνης απόρριψης ή εγκοπής επειδή μπορούν να επιλέγουν μία συγκεκριμένη συχνότητα και να την απορρίπτουν.





Σχήμα 41: είσοδος-έξοδος περιεχομένου στα είδη 4 φίλτρου στο πεδίο της συχνότητας χρόνο

## Αναλογικά και ψηφιακά φίλτρα

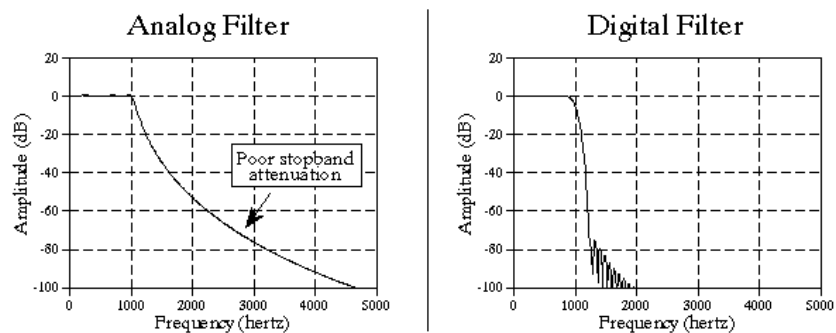
Τα αναλογικά φίλτρα ή αλλιώς φίλτρα συνεχούς χρόνου, είναι τα φίλτρα εκείνα που στη είσοδό τους δέχονται αναλογικό σήμα και που χρησιμοποιούν συγκεκριμένες μεθόδους για να το φιλτράρουν. Το σήμα που εισέρχεται στο σύστημα(φίλτρο) έχει συνεχόμενες τιμές άρα και το σήμα που εξέρχεται θα είναι συνεχούς χρόνου. Αντιστοίχως, τα ψηφιακά φίλτρα που ονομάζονται αλλιώς και φίλτρα διακριτού χρόνου, χρησιμοποιούν άλλες μεθόδους επεξεργασίας και δέχονται ψηφιακό σήμα στη είσοδό τους. Το σήμα εισόδου έχει διακριτές τιμές οπότε και το σήμα εξόδου θα είναι ανάλογης μορφής. Τις περισσότερες φορές που γίνεται χρήση ψηφιακών φίλτρων απαιτείται η μετατροπή του σε αναλογικό στο τέλος της διάταξης. Στα ψηφιακά φίλτρα μετά την επεξεργασία, προστίθεται ένα κύκλωμά που μετατρέπει το ψηφιακό σήμα με αναλογικό. Το κύκλωμά αυτό ονομάζεται Μετατροπέας Αναλογικού σήματος σε Ψηφιακό AD Converter. Αντίθετα σε ένα κύκλωμά με αναλογικό φίλτρο δεν χρειάζεται τέτοιο είδος μετατροπής, αφού το σήμα παραμένει στην καθαρή αναλογική μορφή του καθ' όλη τη διάρκεια της επεξεργασίας του.

## Τύποι αναλογικών φίλτρων

Υπάρχουν πολλοί τύποι αναλογικών φίλτρων ανάλογα με την μορφή της συνοπτικής απόκρισης του σήματος εξόδου.

- Butterworth: δίνει σχεδόν είδη απόκριση και στη ζώνη διέλευσης και στη ζώνη αποκοπής χωρίς κυματώσεις αλλά έχει μέτρια μετάβαση από τη μία περιοχή στην άλλη.
- Chebyshev :παρουσιάζει κυματώση στη ζώνη διέλευσης, ενώ η μετάβαση από τη ζώνη διέλευσης στη ζώνη αποκοπής είναι γρήγορη.
- Elliptic: εμφανίζει κυματώσεις και στη ζώνη διέλευσης και στη ζώνη αποκοπής. Έχει όμως τη στενότερη ζώνη μετάβασης από όλα τα υπόλοιπα είδη.
- Bessel: έχει την πιο αργή μετάβαση από τη ζώνη διέλευσης στη ζώνη αποκοπής αλλά παρουσιάζει γραμμική φάση στη ζώνη διέλευσης συγκριτικά με τα άλλα φίλτρα.

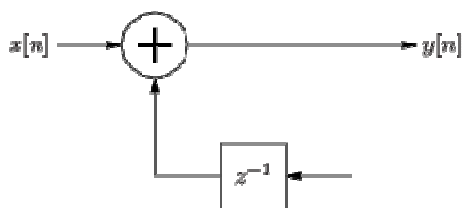
Ένας ακόμη διαχωρισμός των αναλογικών φίλτρων γίνεται με βάση τα διακριτά στοιχεία που χρησιμοποιούνται για την κατασκευή τους. Διακρίνονται σε παθητικά και ενεργά. Τα παθητικά φίλτρα αποτελούνται μόνο από αντιστάσεις, πυκνωτές και πηνία τα οποία απλά αποκόπτουν τις ανεπιθύμητες συχνότητες. Τα ενεργά φίλτρα κατασκευάζονται με τη χρήση τελεστικών ενισχυτών.



Σχήμα 42: Αναλογικό -ψηφιακό φίλτρο

Ένα ψηφιακό φίλτρο λειτουργεί ως εξής: δέχεται στην είσοδο του ένα σήμα διακριτού χρόνου, εκτελεί την προκαθορισμένη επεξεργασία η οποία βασίζεται στη χρήση αλγορίθμων και υπολογισμών, και στην έξοδο του δίνει ένα νέο σήμα διακριτού χρόνου (ψηφιακό).

$$y[n] = x[n] + x[n - 1]$$



Σχήμα 43 Διάγραμμα ψηφιακού φίλτρου

### Τύποι ψηφιακών φίλτρων

Τα φίλτρα μπορούν να ταξινομηθούν σε δύο βασικές κατηγορίες. Στα φίλτρα πεπερασμένης κρουστικής απόκρισης (FIR) και τα φίλτρα άπειρης κρουστικής απόκρισης (IIR). Λόγω της βασικής αρχής λειτουργίας τους τα IIR φίλτρα ονομάζονται και επαναληπτικά ή αναδρομικά ενώ τα FIR ονομάζονται μη αναδρομικά

Τα βασικά χαρακτηριστικά των FIR ψηφιακών φίλτρων είναι :

- γραμμική φάση,
- είναι ευσταθή.

Τα βασικά χαρακτηριστικά των IIR ψηφιακών φίλτρων είναι:

- μη γραμμική φάση
- χαμηλότερης τάξης φίλτρο από ένα αντίστοιχο FIR
- ασταθές.

### Πλεονεκτήματα της χρήσης ψηφιακών φίλτρων

Η παρακάτω λίστα παρουσιάζει μερικά από τα κύρια πλεονεκτήματα των ψηφιακών φίλτρων από αναλογικά.

1. Ένα ψηφιακό φίλτρο είναι προγραμματιζόμενο, δηλαδή η λειτουργία του καθορίζεται από ένα πρόγραμμα που είναι αποθηκευμένο στη μνήμη του επεξεργαστή. Αυτό σημαίνει ότι το ψηφιακό φίλτρο μπορεί εύκολα να αλλάξει χωρίς να επηρεαστεί το κύκλωμα (υλικό). Ένα αναλογικό φίλτρο μπορεί να αλλάξει μόνο με επανασχεδιασμό του κυκλώματος φίλτρου. Τα ψηφιακά φίλτρα είναι εύκολα σχεδιασμένα, δοκιμασμένα και υλοποιούνται σε έναν υπολογιστή γενικής χρήσης ή έναν σταθμό εργασίας.

2. Σε αντίθεση με τα αναλογικά τους αντίστοιχα, τα ψηφιακά φίλτρα μπορούν να χειριστούν με ακρίβεια τα σήματα χαμηλής συχνότητας. Καθώς η ταχύτητα της τεχνολογίας συνεχίζει να αυξάνεται, τα ψηφιακά φίλτρα εφαρμόζονται σε σήματα

υψηλής συχνότητας στην περιοχή RF (ραδιοσυχνότητα), η οποία στο παρελθόν αποτελούσε αποκλειστική συντήρηση της αναλογικής τεχνολογίας.

Λειτουργία ψηφιακού φίλτρου

Έχουμε "ακατέργαστο" σήμα που πρόκειται να ψηφιοποιηθεί ψηφιακά είναι υπό μορφή κυματομορφή που περιγράφεται από τη λειτουργία  $V = x(t)$  όπου  $t$  είναι χρόνος. Αυτό το σήμα λαμβάνεται δειγματοληπτικά σε χρονικά διαστήματα  $h$  (το διάστημα δειγματοληψίας). Η τιμή δειγματοληψίας στο χρόνο  $t = ih$  είναι

$$x_i = x(ih)$$

Έτσι, οι ψηφιακές τιμές που μεταφέρονται από το ADC στον επεξεργαστή μπορούν να αναπαρασταθούν από την ακολουθία

$$x_0, x_1, x_2, x_3, \dots$$

που αντιστοιχεί στις τιμές της κυματομορφής σήματος στο

$$t = 0, h, 2h, \dots$$

και  $t = 0$  είναι η στιγμή στην οποία αρχίζει η δειγματοληψία.

Στο χρόνο  $t = nh$  (όπου  $n$  είναι κάποιος θετικός ακέραιος αριθμός), οι τιμές που είναι διαθέσιμες στον επεξεργαστή, αποθηκευμένες στη μνήμη, είναι

$$x_0, x_1, x_2, x_3, \dots, x_n$$

οι τιμές δειγματοληψίας  $x_{n+1}, x_{n+2}$  κλπ. δεν είναι διαθέσιμες, καθώς δεν έχουν συμβεί ακόμα)

Η ψηφιακή έξοδος από τον επεξεργαστή στο DAC αποτελείται από την ακολουθία τιμών

$$y_0, y_1, y_2, y_3, \dots, y_n$$

Γενικά, η τιμή του  $y_n$  υπολογίζεται από τις τιμές  $x_0, x_1, x_2, x_3, \dots, x_n$ . Ο τρόπος με τον οποίο υπολογίζονται τα  $y$  από τα  $x$  καθορίζει τη δράση φιλτραρίσματος του ψηφιακού φίλτρου.

## 6.ΠΕΡΙΒΑΛΛΟΝ ΕΡΓΑΣΙΑΣ

Χρησιμοποιώντας ανοιχτού κώδικα εργαλεία (opensource) μπορούμε να αποκωδικοποιήσουμε/κωδικοποιήσουμε ,αναμεταδώσουμε να φιλτράρουμε και να αναπαράγουμε ήχο χρησιμοποιώντας διαθέσιμες βιβλιοθήκες.Για την ανάλυση ήχου χρησιμοποιήσαμε γλώσσα προγραμματισμού python λόγω της εύκολης εγκατάστασης των βιβλιοθηκών της και του πλουσίου documentation και της απλότητας στην χρήση μιας και ενδείκνυται για γρήγορες υλοποιήσεις. Η python είναι προεγκατεστημένη σε όλα τα λειτουργικά συστήματα linux ,στα windows η εγκατάσταση είναι πολύ εύκολη θα αναφερθούμε παρακάτω επίσης έχουμε εγκαταστήσει το XAMPP για την διαδραστική εφαρμογή .Η εγκατάσταση των επιπλέον βιβλιοθηκών της γίνεται μέσω της pip .Για να κάνουμε install την pip τρέχουμε την εντολή:

```
pip install [όνομα βιβλιοθήκης που θα χρησιμοποιήσουμε]
```

από το τερματικό όπως μας ορίζει το documentation.

```
C:\Python37\Scripts>python -m pip install --upgrade pip
Collecting pip
  Downloading https://files.pythonhosted.org/packages/c2/d7/90f34cb0d83a6c5631cf71df64cc1054598c843a92b400e55675cc2ac37/pip-18.1-py2.py3-none-any.whl (1.3MB)
    100% |#####|: 1.3MB 476kB/s
Installing collected packages: pip
  Found existing installation: pip 18.0.1
  Uninstalling pip-18.0.1:
    Successfully uninstalled pip-18.0.1
  Successfully installed pip-18.1
```

Λίγα λόγια για την Python

Η Python είναι μια γλώσσα προγραμματισμού γενικής χρήσης, απλή και εύκολη στην εκμάθηση, αποδοτική, παραγωγική και επεκτάσιμη. Μπορεί να χρησιμοποιηθεί τόσο για εκπαιδευτικούς σκοπούς όσο και για την ανάπτυξη ολοκληρωμένων εφαρμογών. Διαθέτει πληθώρα έτοιμων βιβλιοθηκών που μπορούν να χρησιμοποιηθούν εύκολα και άμεσα. Η φιλοσοφία της Python συνοψίζεται κυρίως στην απλότητα και την αναγνωσιμότητα του κώδικα.

Η Python δεν προσφέρεται προεγκατεστημένη στα Windows . Η εγκατάσταση της Python είναι πολύ εύκολη από το ισότοπο της <https://www.python.org/>, επιλέγουμε τον σύνδεσμο Downloads και κατεβάζουμε το νεότερο αρχείο εγκατάστασης της σειράς 3ή 2 για Windows.



Ακόμα εγκαταστήσαμε το Sublime (editor), το οποίο υποστηρίζει πολλές δυνατότητες, όπως χρωματική επισήμανση των εντολών, πολλαπλές αναιρέσεις, κατάλληλη στοίχιση των εντολών και αυτόματη συμπλήρωση εντολών. Σε ένα τέτοιο παράθυρο γράφουμε πρώτα ολόκληρο το πρόγραμμά μας ,στη συνέχεια το αποθηκεύουμε σε αρχείο .py και στο τέλος το εκτελούμε.

Ο πιο απλός τρόπος εκτέλεσης του προγράμματος python είναι η πληκτρολόγηση του "python example.py"

```
1 import sounddevice as sd
2 import numpy as np
3 import scipy.io.wavfile as wav
4 import sys
5 from struct import pack
6 from matplotlib import pyplot as plt
7 #import pyaudio
8 import wave
9
10 fs=44100
11 duration = 5 # seconds
12 myrecording = sd.rec(duration * fs, samplerate=fs, channels=2, dtype='float64')
13 print(type(myrecording))
14 data=myrecording
15 print("Recording Audio")
16 sd.wait()
17 print( "Audio recording complete , Play Audio")
18 sd.play(myrecording, fs)
19 sd.wait()
20 print ("Play Audio Complete")
21
22 #high filter
23
24
25 path = 'C:\\xampp\\htdocs\\wave.wav'
26 T = 5.0 # seconds
27 n = int(T * fs) # total number of samples
28 t = np.linspace(0, T, n, endpoint=False)
29
30 wav.write(path,fs,myrecording)
```



Python για την ψηφιακή επεξεργασία σημάτων

Η βασική λειτουργικότητα της Python επεκτείνεται με χιλιάδες διαθέσιμες δωρεάν βιβλιοθήκες όπως αναφερθήκαμε παραπάνω , πολλές από τις οποίες είναι απίστευτα πρακτικές.

Οι βιβλιοθήκες που χρησιμοποιήθηκαν στην εργασία είναι :

Numpy : Είναι μια βιβλιοθήκη που υποστηρίζει πολυδιάστατους πίνακες (array) όπου χρειάζονται για τα δεδομένα του ψηφιακού σήματος. Αυτή η βιβλιοθήκη χρειάζεται για να δημιουργήσουμε ένα σήμα όπως παρακάτω

```
fs = 30.0      #δείγματα ανά δευτερόλεπτο
T = 5.0       # διάρκεια σήματος
n = int(T * fs) # συνολικά δείγματα
t = np.linspace(0, T, n, endpoint=False) Τυχαία δεδομένα
signal = np.sin(1.2*2*np.pi*t) μετατροπή δεδομένων σε ημίτονο.
```

```
C:\Python37\Scripts>pip install numpy
Requirement already satisfied: numpy in c:\python37\lib\site-packages (1.15.4)
C:\Python37\Scripts>
```

Scipy: Η scipy είναι μια βιβλιοθήκη που περιέχει πολλές επιστημονικές εφαρμογές – συναρτήσεις επεξεργασίας – προβολής πολύπλοκων δεδομένων όπως για τις δικές μας ανάγκες θα χρησιμοποιήσουμε την βιβλιοθήκη scipy.signal όπου περιέχει τις παρακάτω έτοιμα υλοποιημένες συναρτήσεις όπως το φίλτρο του Butterworth που ορίζεται όπως παρακάτω:

```
scipy.signal.butter(N, Wn, btype='low', analog=False, output='ba')
```

Αναλυτικά η συνάρτηση παρακάτω:

N : int

Wn : πίνακας

btype : {'lowpass', 'highpass', 'bandpass', 'bandstop'}, Ο τύπος φίλτρου. Η προεπιλογή είναι "lowpass".

analog=False, : Όταν είναι αληθές, επιστρέψτε ένα αναλογικό φίλτρο, διαφορετικά επιστρέφει ένα ψηφιακό φίλτρο.

output='ba' : έξοδος : {'ba' 'zpk'}

Είδος εξόδου: αριθμητής / παρονομαστής («ba»). Η προεπιλογή είναι "ba".

Επιστρέφει

**b, a** : ndarray, ndarray

Αριθμητής ( $\beta$ ) και παρονομαστής ( $\alpha$ ) πολυώνυμα του φίλτρου IIR. Επιστρέφεται μόνο αν το output = 'ba' .

**z, p, k** : ndarray, ndarray, float

Μηδενικά, πόλους και κέρδος συστήματος της λειτουργίας μεταφοράς φίλτρου IIR. Επιστρέφεται μόνο αν το output = 'zpk' .

Matplotlib: είναι μια βιβλιοθήκη οπτικοποίησης δεδομένων στη Python. είναι ευέλικτη και έχει πολλές πρακτικές, και ενσωματωμένες προεπιλογές . Για αρχή σχεδιάζουμε με τη βοήθεια της plot() και τα εμφανίζουμε χρησιμοποιώντας την show() Το pyplot είναι ένα module στη matplotlib. Γι 'αυτό συχνά βλέπουμε το matplotlib.pyplot στον κώδικα.

```
from matplotlib import pyplot as plt
```

 ( as plt είναι ένα ψευδώνυμο )

Το module επιτρέπει να δημιουργήσουμε αυτόματα αριθμητικά στοιχεία και άξονες για να επιτύχουμε το επιθυμητό πλοταρισμα (plot) .Το pylab είναι ένα άλλο module,το οποίο εγκαθίσταται και αυτό μαζί με το πακέτο matplotlib.

pip install --user matplotlib

```
C:\Users\user>pip install --user matplotlib
Collecting matplotlib
  Using cached https://files.pythonhosted.org/packages/3f/16/4500e22ea8d11f4946b
d902695d0113f82a0aaca45f352478f157ea6623d/matplotlib-3.0.2-cp37-cp37m-win32.whl
Requirement already satisfied: numpy>=1.10.0 in c:\python37\lib\site-packages (f
rom matplotlib) (1.15.2)
Requirement already satisfied: python-dateutil>=2.1 in c:\python37\lib\site-packa
ges (from matplotlib) (2.7.5)
Requirement already satisfied: cycler>=0.10 in c:\python37\lib\site-packages (fr
om matplotlib) (0.10.0)
Requirement already satisfied: pyparsing<=2.0.4,!=2.1.2,!=2.1.6,!=2.0.1 in c:\py
thon37\lib\site-packages (from matplotlib) (2.3.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\python37\lib\site-packag
e (from matplotlib) (1.0.1)
Requirement already satisfied: six>=1.5 in c:\python37\lib\site-packages (from p
ython-dateutil>=2.1->matplotlib) (1.11.0)
Requirement already satisfied: astunparse in c:\python37\lib\site-packages (from
kiwisolver>=1.0.1->matplotlib) (3.6.1)
Installing collected packages: matplotlib
Successfully installed matplotlib-3.0.2
```

Ακόμα έχουν εγκατασταθεί βιβλιοθήκες όπως :

Pip install sounddevice

Sounddevice : Βιβλιοθήκη για την αναπαραγωγή και εγγραφή σημάτων ήχου.

Pip install wavefile

Wavefile : Για την ανάγνωση και την εγγραφή αρχείων ήχου.

```

C:\Python37\Scripts>pip install sounddevice
Collecting sounddevice
  Downloading https://files.pythonhosted.org/packages/15/d3/23552ac2935ed61044b4
55283416856fd2ccfc05947b3fa92f17a4bdf0e/sounddevice-0.3.12-py2.py3.cp26.cp27.cu
32.cp33.cp34.cp35.cp36.cp37.pp27.pp32.pp33.pp34.pp35.pp36-none-win32.whl (163kB)
  100% |#####| 163kB 853kB/s
Collecting CFFI>=1.0 (from sounddevice)
  Downloading https://files.pythonhosted.org/packages/2d/3e/9204a72d4e2d08e494f1
30e49cc20265709312749f40f365326f0f520e/cffi-1.11.5-cp37-cp37w-win32.whl (154kB)
  100% |#####| 154kB 334kB/s
Collecting pyparser (from CFFI>=1.0->sounddevice)
  Downloading https://files.pythonhosted.org/packages/6d/9e/40196946aa21fbaad12
95e88d1e72da2657703e83e1b92a5582e6220a/pyparser-2.19.tar.gz (150kB)
  100% |#####| 150kB 471kB/s
Installing collected packages: pyparser, CFFI, sounddevice
  Running setup.py install for pyparser ... done
  Successfully installed CFFI-1.11.5 pyparser-2.19 sounddevice-0.3.12

C:\Python37\Scripts>pip install wavefile
Collecting wavefile
  Downloading https://files.pythonhosted.org/packages/a8/41/35-d7f95f87017021ba
6524977ad352a77079243b2a0c4577e052e/wavefile-1.5.tar.gz
  Downloading already satisfied: numpy in c:\python37\lib\site-packages (from wave
file) (1.15.4)
Installing collected packages: wavefile
  Running setup.py install for wavefile ... done
  Successfully installed wavefile-1.5

C:\Python37\Scripts>pip install pyaudio
Collecting pyaudio
  Downloading https://files.pythonhosted.org/packages/ab/42/12f04721c055bf41760a
1538e79998980e0590290502490ab/PyAudio-0.2.11.tar.gz
Installing collected packages: pyaudio
  Running setup.py install for pyaudio ... error
Complete output from command c:\python37\python.exe -u -c "import setuptools
; tokenize;__file__='C:\Users\user\AppData\Local\Temp\pip-install-11lu37rn
\pyaudio\setup.py';if __getattr__({'tokenize','open','open'}):code=f.read().r
eplace('\n','\r\n');if __close__:{exec(compile(code,__file__,'exec'))}" install --
single-version-externally-managed --compile:
  running install
  running build

```

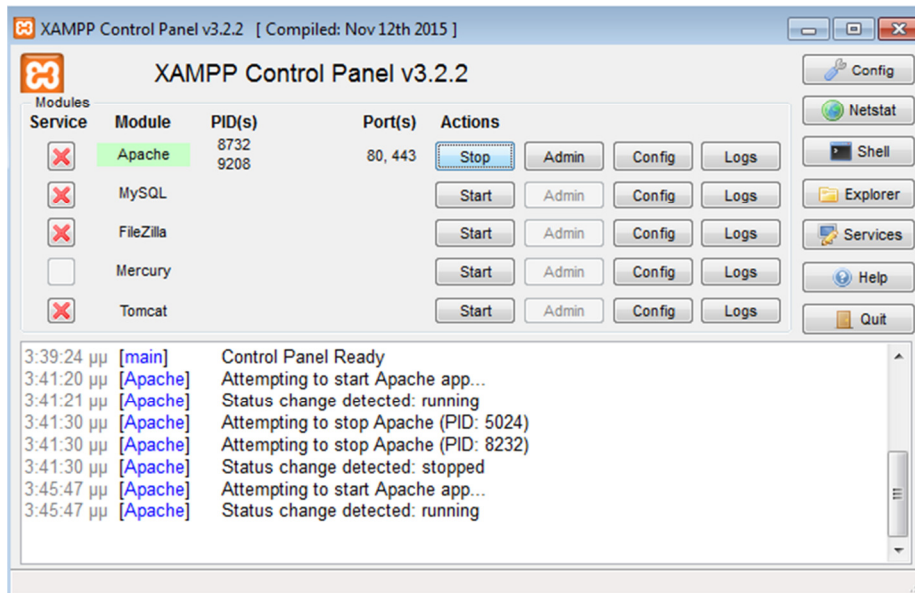
Πηγή:

<https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.signal.butter.html#scipy.signal.butter>

Πηγή: <https://matplotlib.org/>

<https://matplotlib.org/users/installing.html><https://www.datacamp.com/community/tutorials/matplotlib-tutorial-python/>

Για να μπορούν να δουλέψει ο σερβερ της apache εγκαταστήσαμε το XAMPP και εκκινήσαμε τον σερβερ όπως παρακάτω



Χρειαστήκαμε μόνο τον apache που τρέχει την php\ . Τοποθετήσαμε τα αρχεία μας PHP στο φάκελο htdocs που βρίσκεται κάτω από το φάκελο "XAMMP" στη μονάδα δίσκου C: Η διαδρομή αρχείου είναι "C: \ xampp \ htdocs" για τον διακομιστή web . Μεταφέρθηκαν και τα αρχεία .py . Θέσαμε το index.php και μπορούμε να έχουμε πρόσβαση μέσα από το url: <http://localhost/index.php> . Ακόμα δημιουργήσαμε ένα φάκελο img και ένα φάκελο uploads που αποθηκεύουμε τα αρχεία .wav .

Στο αρχείο php.ini , αφαιρέσουμε τα error που εμφανίζονταν πάνω στη οθόνη αναζητήσαμε τα παρακάτω και γυρίσαμε τα display\_errors από on σε off

```
Off = Do not display any errors
; stderr = Display errors to STDERR (affects only CGI/CLI binaries!)
; On or stdout = Display errors to STDOUT
; Default Value: On
; Development Value: On
; Production Value: Off
; http://php.net/display-errors
display_errors=off
; The display of errors which occur during PHP's startup sequence are handled
; separately from display_errors. PHP's default behavior is to suppress those
; errors from clients. Turning the display of startup errors on can be useful in
; debugging configuration problems. We strongly recommend you
; set this to 'off' for production servers.
; Default Value: Off
; Development Value: On
; Production Value: Off
; http://php.net/display-startup-errors
display_startup_errors=off
```

Αφού βρήκαμε μια εύκολη σχετικά και γρήγορη υλοποίηση Low pass φίλτρου σε python, μπορούσαμε να φτιάξουμε μια online διαδραστική εφαρμογή με την χρήση XAMPP ως πρόγραμμα server και με την χρήση της PHP μπορούμε να εκτελέσουμε το πρόγραμμα lowpassfilter.py με την εντολή exec(). στο αρχείο index.php. Σε αυτό το αρχείο μπορούμε και έχουμε την βασική έκδοση -εμφάνιση του προγράμματος, και με την php μπορούμε να προσφέρουμε διαδραστικότητα στο χρήστη με την βασική λειτουργία φόρμας html, περνάμε την είσοδο του χρήστη στην php η οποία εκτελεί το πρόγραμμα της python με ορίσματα. Αυτά τα ορίσματα στην Python ορίζονται με την βιβλιοθήκη sys η οποία είναι προ εγκατεστημένη και μπορεί να χρησιμοποιηθεί με την γραμμή import sys, και η μεταβλητή sys.argv είναι ένας μονοδιάστατος πίνακας με sys.argv[0] = όνομα του προγράμματος, με sys.argv[1] το πρώτο ορίσματα κοκ. Στην δική μας εφαρμογή έχουμε μία είσοδο χρήστη αυτή της τιμής cutoff. Η τιμή από την φόρμα με το πάτημα του κουμπιού «cutoff» ξαναφορτώνει η σελίδα μαζί με την τιμή της φόρμας της μορφής GET['cutoff'], η οποία αν δεν είναι κενή τότε εισάγεται ως όρισμα με την παρακάτω μορφή

Python lowpassfilter.py 5 (όπου 5 μια τυχαία τιμή του χρήστη). Στη συνέχεια το πρόγραμμα lowpassfilter.py αποθηκεύει την φωτογραφία στο φάκελο img και αφού ολοκληρωθεί την εμφανίζει στον χρήστη. Ουσιαστικά είναι ένας απλός αλλά βασικός σχεδιασμός και μπορεί πολύ εύκολα να αναπτυχθεί με παραπάνω λειτουργίες. Όπως να υπάρχει δυναμικά η δημιουργία του σήματος καθώς επίσης και να υπάρχουν περισσότερα μοντέλα φίλτρων με τα ανάλογα πλεονεκτήματα και μειονεκτήματα τους.

Εκτέλεση python στην index.php

Γενικά η Php μας παρέχει web-based λειτουργίες για την ανάπτυξη web εφαρμογών και κάποια χαρακτηριστικά την εκτέλεση εξωτερικών προγραμμάτων ή εντολών στο τερματικό- terminal(γραμμή εντολών).

Η sessions start () μπορεί και ανοίγει μια συνεδρία ,αυτή η συνεδρία τερματίζεται όταν κλείσει ο browser, σβήνονται οι μεταβλητές session, οι οποίες ορίζονται \$\_SESSION['ονομα μεταβλητης'] = 5 .Σε κάθε σελίδα php στην αρχή θα πρέπει να την γραφουμε στην αρχή του αρχείου για να μπορέσει να κρατηθεί η συνεδρια-session . Έτσι θα μπορούμε να ορίσουμε μεταβλητές πχ να ξέρουμε τί πατήσαμε πριν. Στο πρόγραμμά μας δεν τη χρησιμοποιήσαμε.

```
<?php
session_start();
```

Η συνάρτηση exec()

```
string exec ( string $command [, array &$output [, int &$return_var ] ]
```

χρησιμοποιείται για την εκτέλεση ενός εξωτερικού προγράμματος συστήματος(τερματικού) μέσω της php. Χρησιμοποιήσαμε την echo για να εκτυπώσουμε την έξοδο της συνάρτησης exec [true ή false.] .Το δεύτερο όρισμα της συνάρτησης ορίζει την μεταβλητή που θα έχει την επιστροφή το retur, την έξοδο της συνάρτησης που τρέξαμε στο τερματικό.

```
echo exec("$cmd", $output);
//echo $cmd;
echo print_r($output);|
```

Η print\_r () χρησιμοποιείται για να εκτυπώσει array (πίνακα) ,δηλαδή πίνακας ή array είναι τύπος μεταβλητής της μορφής \$pinakas[1] = '32' μπορεί να έχει 1- πολλές διαστάσεις.

Γράψαμε html φόρμες, για να μπορέσουμε να στείλουμε τα δεδομένα στην php, όταν πατάμε submit σε μια φόρμα τα δεδομένα αυτής εμπεριέχονται στο αίτημα post/get .

```

<form>
<label>File to analyze<input type="text" name="file" value="<?php echo $_GET['file'];?>" >/label>
<label> cutoff<input type="text" name="cutoff" >/label>
<input type="submit" value="apply low pass to <?php echo $_GET['file'];?> ">
</form>

```

Οι μεταβλητές μας έχουν τη μορφή `$_GET["name formas"]` της action της φόρμας(στη δική μας περίπτωση στην ίδια σελίδα, `if (get && get)` ουσιαστικά ελέγχει ποια φόρμα χρησιμοποιήσαμε)

```
if($_GET['file'] && $_GET['cutoff'])
```

```
$_GET['name']
```

Η μεταβλητή `$_GET` χρησιμοποιείται για τη συλλογή τιμών από μια φόρμα με τη μέθοδο = "get". Όταν χρησιμοποιούμε τη μεταβλητή `$_GET`, όλα τα ονόματα μεταβλητών και οι τιμές εμφανίζονται στη διεύθυνση URL.

<http://localhost/index.php?file=nto.wav&lowpass=1> (από το παράδειγμα μας)

Η `pyscript` έχει την εντολή σε μορφή string, η `cmd` ολοκληρώνει την εντολή με τα ορίσματα της python.

```

if($_GET['file'] && $_GET['cutoff'])
{
    $pyscript = 'C:\\xampp\\htdocs\\filelowpass.py';
    $cmd = "$python $pyscript ".$_GET['file']." ".$_GET['cutoff'];
    echo exec("$cmd", $output);
    //echo $cmd;
    echo print_r($output);
}

```

Στο βασικό κορμό μας που είναι οι html φόρμες έχουμε βάλει 2 βιβλιοθήκες jQuery και Bootstrap

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other head content must come *after* these tags -->
    <title>Διαδραστική εφαρμογή </title>

    <!-- Latest compiled and minified CSS -->
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAKycuHARg320mUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">

```

Η jQuery είναι βιβλιοθήκη JavaScript συναρτήσεων συμβατή με browsers όπως chrome, mozilla κ.τ.λ. Για να δημιουργήσουμε όμως τις εναλλασσόμενες καρτέλες (tabs) χρησιμοποιήσαμε ως βάση τη bootstrap, που έχει μια βιβλιοθήκη CSS, jQuery και JS.

Η χρησιμότητά της είναι ότι όταν κλικάρει ο χρήστης στο πρώτο tab ,το bootstrap.min.js

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg320mUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">
```

Αναγνωρίζει το αντικείμενο του πρώτου link και μας εμφανίζει το περιεχόμενο. Όταν ο χρήστης θα πατήσει το δεύτερο , μέσω των id υπάρχει μια συνάρτηση που χρησιμοποιεί jQuery για να μας κρύψει το πρώτο και να ανοίξει το δεύτερο tab.

```
<ul class="nav nav-tabs">
  <li class="active"><a data-toggle="tab" href="#home">File upload & apply low pass</a></li>
  <li><a data-toggle="tab" href="#menu1">Low pass filter</a></li>
  <li><a data-toggle="tab" href="#menu2">High pass</a></li>
  <li><a data-toggle="tab" href="#menu4">Bandpass</a></li>
  <li><a data-toggle="tab" href="#menu3">Recmic </a></li>
</ul>
```

Για να κάνουμε τις καρτέλες εναλλασσόμενες, προσθέσαμε το χαρακτηριστικό εναλλαγής δεδομένων = "καρτέλα" σε κάθε σύνδεσμο. Στη συνέχεια, προσθέσαμε μια κλάση .tab-pane με ένα μοναδικό αναγνωριστικό για κάθε καρτέλα σε ένα στοιχείο div με το class .tab-content.

```
<div class="tab-content">
  <div id="home" class="tab-pane fade in active">
    upload
    <form action="upload.php" method="post" enctype="multipart/form-data">
      Select to upload:
      <input type="file" name="fileToUpload" id="fileToUpload">
      <input type="submit" value="Upload " name="submit">
    </form>
```

## PHP Ανέβασμα αρχείου

Έχουμε δημιουργήσει την HTML φόρμα που επιτρέπει στους χρήστες να κάνουν upload ένα αρχείο

```
upload
<form action="upload.php" method="post" enctype="multipart/form-data">
  Select to upload:
  <input type="file" name="fileToUpload" id="fileToUpload">
  <input type="submit" value="Upload " name="submit">
</form>
```

Σε ένα POST method πρέπει να κωδικοποιήσουμε τα δεδομένα που περιέχει η φόρμα μας. Οι HTML φόρμες παρέχουν τρεις μεθόδους κωδικοποίησης.

Στο upload χρησιμοποιήσαμε `enctype="multipart/form-data"`. Το enctype είναι το encoding type δηλαδή πως κωδικοποιούμε τα δεδομένα στη φόρμα. Συνήθως τα αρχεία στην αρχή έχουν κάποια περιέργα δεδομένα π.χ κωδικα ASCII , με το multipart/form-data επιτρέπουμε την ενσωμάτωση ολόκληρων αρχείων ή μεταβλητών στα δεδομένα . Πάντα με upload file χρησιμοποιούμε multipart/form-data.

```
upload
<form action="upload.php" method="post" enctype="multipart/form-data">
```

Με τη `foreach ($files1 as $thefile)` διαβάζουμε έναν πίνακα (array) και χρησιμοποιούμε τα περιεχόμενά του με το όνομα της μεταβλητής , στη δική μας περίπτωση το File.

```
files we upload
<?php
//syndesmoi ton arxeion pou exoume anevasei
foreach ($files1 as $thefile){
    print '<a href="index.php?file='.$thefile.'&lowpass=1">'.$thefile.'</a><br />';
}
?>
```

Στην εφαρμογή μας αυτό φαίνεται στην εικόνα παρακάτω .

```
files we upload .
..
heavy-rain-daniel_simon.wav
meg.wav
nto.wav
File to analyze 
```

Μετάπειτα δημιουργήσαμε ένα upload.php αρχείο.

```
<?php
session_start();
$target_dir = 'C:\\xampp\\htdocs\\uploads\\';
$target_file = $target_dir . basename($_FILES["fileToUpload"]["name"]);
$uploadOk = 1;
$imageFileType = strtolower(pathinfo($target_file,PATHINFO_EXTENSION));
if(isset($_POST["submit"])) {
    if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $target_file)) {
        echo "The file ". basename( $_FILES["fileToUpload"]["name"]). " has been uploaded.";
    }
}
header('Location: index.php');
$_SESSION['filename'] = $_FILES["fileToUpload"]["name"];
exit;
?>
```



Ορίσαμε μια μεταβλητή την `$target_dir = 'C:\\xampp\\htdocs\\uploads\\'`; Η οποία καθορίζει τον κατάλογο στον οποίο θα αποθηκεύτε το αρχείο μας. Και την `$target_file = $target_dir .`

`basename($_FILES["fileToUpload"]["name"]);` που είναι η διαδρομή του αρχείου (dir)μαζί με το όνομα του αρχείου . Η `basename` είναι μια συνάρτηση της php που από το `$_FILES["fileToUpload"]["name"]` μας επιστρέφει μόνο το όνομα του αρχείου που ανεβάσαμε .

Το `$uploadOk = 1;` Είναι μια μεταβλητή True ή False

`$imageFileType = strtolower(pathinfo($target_file,PATHINFO_EXTENSION));` η μεταβλητή μας επιστρέφει το τύπο του αρχείου .

`if(isset($_POST["submit"]))` Ελέγχουμε εάν ο χρήστης έχει πατήσει το κουμπί 'submit' , εάν η εντολή δηλαδή έχει εκτελεστεί επιτυχώς .

`if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $target_file))` Αν ανέβηκε το αρχείο μετέφερε το αρχείο tmp\_name του fileToUpload στο target\_file (που είναι η διαδρομή μαζί με το όνομα του αρχείου) , if true τότε εμφάνισε το αρχείο ανέβηκε.

`{echo "The file ". basename( $_FILES["fileToUpload"]["name"]). " has been uploaded.";`

Επέστρεψε στο index.php `header('Location: index.php');`

## Index.php

```
<?php
```

```
session_start();
$recmic='';
$cutoff='';
$highcutoff='';
$python = 'python.exe';
if($_GET['file'] && $_GET['cutoff'])
{
    $pyscript = 'C:\\xampp\\htdocs\\filelowpass.py';
    $cmd = "$python $pyscript ".$_GET['file']." ".$_GET['cutoff'];
```

```

echo exec("$cmd", $output);
//echo $cmd;
echo print_r($output);
}
if($_GET['recmic'])
{
    $pyscript = 'C:\\xampp\\htdocs\\recmic.py';
    $cmd = "$python $pyscript ".$_GET['recmic'];
    echo exec("$cmd", $output);
    //echo $cmd;
    //echo print_r($output);
}
if($_GET['cutoff'])
{
    $pyscript = 'C:\\xampp\\htdocs\\lowpass.py';
    $cmd = "$python $pyscript ".$_GET['cutoff'];
    echo exec("$cmd", $output);
    // echo $cmd;
    //echo print_r($output);
}
if($_GET['highcutoff'])
{
    $cmd = "$python $pyscript ".$_GET['highcutoff'];
    echo exec("$cmd", $output);
    // echo $cmd;
    // echo print_r($output);
}
if($_GET['file'] && $_GET['fft'] )
{
    $pyscript = 'C:\\xampp\\htdocs\\wavfilefft.py';

```

```

    $cmd = "$python $pyscript ".$_GET['file'];
echo  exec("$cmd", $output);
//echo $cmd;
echo print_r($output);
}
//diavazoume to fakelo uploads
$dir    = 'C:\\xampp\\htdocs\\uploads\\';
$files1 = scandir($dir);
?>
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other
head content must come *after* these tags -->
    <title>Διαδραστική εφαρμογή </title>
    <!-- Latest compiled and minified CSS -->
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.mi
n.css" integrity="sha384-
BVYiiSIFeK1dGmJRAkyCuHAHRg320mUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"
crossorigin="anonymous">
  </head>
  <body>
    <div class="container">
      <h2>Διαδραστική εφαρμογή για την επεξεργασία ηχητικών σημάτων</h2>
      <p>Στην παρούσα εφαρμογή ο χρήστης μπορεί να ανεβάσει ένα αρχείο
ήχου .wav και στα πεδία "cutoff" να επιλέξει τη συχνότητα αποκοπής σε
ένα lowpass/high pass φίλτρο ,και στο πεδίο remic εγγραφή από το
μικρόφωνο , ως επιστροφή λαμβάνει το τη κυματομορφή του ηχογραφημένου
ήχου </p>

```

```

    <ul class="nav nav-tabs">
      <li class="active"><a data-toggle="tab" href="#home">File upload
& apply low pass</a></li>
      <li><a data-toggle="tab" href="#menu1">Low pass filter</a></li>
      <li><a data-toggle="tab" href="#menu2">High pass</a></li>
      <li><a data-toggle="tab" href="#menu4">FFT</a></li>
      <li><a data-toggle="tab" href="#menu3">Recmic </a></li>
    </ul>
    <div class="tab-content">
      <div id="home" class="tab-pane fade in active">
        upload
        <form action="upload.php" method="post" enctype="multipart/form-
data">
Select to upload:
      <input type="file" name="fileToUpload" id="fileToUpload">
      <input type="submit" value="Upload " name="submit">
      </form>
files we upload
      <?php
      //syndesmoi ton arxeion pou exoume anevasei
      foreach ($files1 as $thefile){
        print '<a
href="index.php?file='.$thefile.'&lowpass=1">'.$thefile.'</a><br />';
      }
    ?>
    <form>
    <label>File to analyze<input type="text" name="file" value="<?php echo
$_GET['file'];?>"> </label>
    <label> cutoff<input type="text" name="cutoff"> </label>
    <input type="submit" value="apply low pass to <?php echo
$_GET['file'];?> ">
    </form>
    

```

```

    </div>
    <div id="menu1" class="tab-pane fade">
        <h1>Low pass filter</h1>
        <form action="" method="get">
<label>
Cutoff:
<input type="number" min="0" max="1000" step="0.1" name="cutoff" >
    </label>
    <input type="submit" value="update cutoff">
</form>

    </div>
    <div id="menu2" class="tab-pane fade">
        <h1>high pass filter</h1>
        <form action="" method="get">
<label>
Cutoff:
<input type="number" min="0" max="200" step="0.1" name="highcutoff" >
    </label>
    <input type="submit" value="update highcutoff">
</form>

    </div>
    <div id="menu3" class="tab-pane fade">
        <h1>Record from mic</h1>
        <form action="" method="get">
<label>
filename:
<input type="text" min="0" max="10" step="0.1" name="recmic" >
</label>
    <input type="submit" value="recmic">

```

```

</form>
  </div>
  <div id="menu4" class="tab-pane fade">
    <h1>FFT</h1>
    <form action="uploadfft.php" method="post"
enctype="multipart/form-data">
      Select to upload:
      <input type="file" name="fileToUpload" id="fileToUpload">
      <input type="submit" value="Upload " name="submit">
</form>
<?php
  //syndesmoi ton arxeion pou exoume anevasei
  foreach ($files1 as $thefile){
    print '<a
href="index.php?file='.$thefile.'&fft=1">'.$thefile.'</a><br />';
  }
?>
  </div>
</div>
</div>
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
</body>
</html>

```

Πηγή : [https://www.w3schools.com/bootstrap/tryit.asp?filename=trybs\\_tabs\\_dynamic&stacked=h](https://www.w3schools.com/bootstrap/tryit.asp?filename=trybs_tabs_dynamic&stacked=h)  
[https://www.w3schools.com/php/php\\_file\\_upload.asp](https://www.w3schools.com/php/php_file_upload.asp)

Παρακάτω βλέπουμε το τελικό στάδιο της εφαρμογής με

- πεδίο file upload & apply lowpass , ο χρήστης μπορεί να εισάγει ένα αρχείο ήχου .wav και να το περάσει μέσα από ένα lowpass φίλτρο .Στο πεδίο file

analyze διαβάζει το αρχείο(στην περίπτωση μας nto.wav) και στο πεδίο cutoff επιλέγει τη συχνότητα αποκοπής .

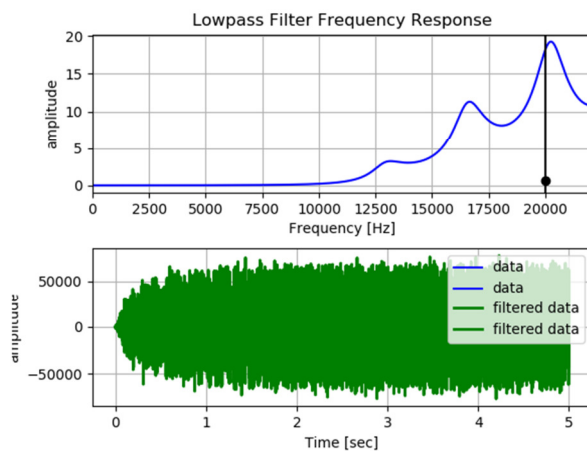
- πεδίο lowpass filter .Εδώ υπάρχουν από προεπιλογή 3 σήματα , ο χρήστης επιλέγει στο πεδίο cutoff τη συχνότητα αποκοπής και μπορεί να δει τη λειτουργία του φίλτρου
- πεδίο highpass filter .Υπάρχουν από προεπιλογή σήματα , ο χρήστης επιλέγει στο πεδίο cutoff τη συχνότητα αποκοπής και μπορεί να δει τη λειτουργία του highpass φίλτρου
- FFT ,ταχύς μετασχηματισμός Fourier
- πεδίο remic(record from microphone) εγγραφή από το μικρόφωνο , ως επιστροφή λαμβάνει το τη κυματομορφή του ηχογραφημένου ήχου.

## Διαδραστική εφαρμογή για την επεξεργασία ηχητικών σημάτων

Στην παρούσα εφαρμογή ο χρήστης μπορεί να ανεβάσει ένα αρχείο ήχου .wav και στα πεδία "cutoff" να επιλέξει τη συχνότητα αποκοπής σε ένα lowpass/highpass φίλτρο και στο πεδίο remic εγγραφή από το μικρόφωνο , ως επιστροφή λαμβάνει το τη κυματομορφή του ηχογραφημένου ήχου

File upload & apply low pass    Low pass filter    High pass    FFT    Remic

upload  
Select to upload:  
Επιλογή αρχείου    Δεν επιλέχθηκ... ανάνα αρχείο.  
Upload  
files we upload .  
heavy-rain-daniel\_simon.wav  
meg.wav  
nto.wav  
File to analyze  cutoff  apply low pass to



## Επεξεργασία ήχου σε Python

Σε αυτό το σημείο οφείλουμε να περιγράψουμε με ένα παράδειγμα τη βασική διαδικασία για την εξομοίωση ενός δειγματοληπτικού σήματος τη οποία συναντάμε σε όλα μας τα προγράμματα στη Python

Δειγματοληψία κυματοειδούς κύματος

Παρακάτω είναι η δημιουργία ενός ημιτονοειδούς κύματος στην Python χρησιμοποιώντας κριτήρια δειγματοληψίας που εξομοιώνουν ένα πραγματικό σήμα. Ο παρακάτω κώδικας "δειγματοληψία" ένα ημιτονοειδές κύμα στα 41,0 kHz για 0,1 δευτερόλεπτα. Έχουμε χρησιμοποιήσει ημιτονοειδές κύμα 200 Hz.

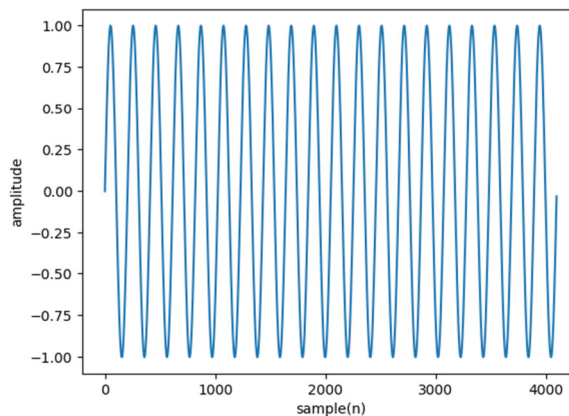
```
import matplotlib.pyplot as plt
import numpy as np

Fs = 4100 # sample rate
T = 1/Fs # sampling period
t = 0.1 # seconds of sampling
sample=4100 # total points in signal
f = 200*t # the frequency of the signal

x = np.arange(sample)
y = np.sin(2 * np.pi * f * x / Fs)
plt.plot(x, y)
plt.xlabel('sample(n)')
plt.ylabel('amplitude')
plt.show()
```

$F \cdot t = 200 \text{ Hz} \cdot 0.1 \text{ s} = 20 \text{ cycles}$

Αυτό σημαίνει ότι θα έχουμε 20 κύκλους από το ημιτονοειδές κύμα 100 Hz σε 0,1 δευτερόλεπτα.



Πηγή : <https://engineersportal.com/blog/2018/9/13/audio-processing-in-python-part-i-sampling-and-the-fast-fourier-transform>

Παράμετροι συναρτήσεων της Python



## scipy.io.wavfile.read

**filename** συμβολοσειρά. Εισάγουμε αρχείο wav.

**mmap** πρόκειται για ανάγνωση δεδομένων (Προεπιλογή: False)

**Επιστρέφει :**

**rate** *int*

Ποσοστό δειγματοληψίας του αρχείου wav

**data** πίνακας

Τα δεδομένα διαβάζονται από το αρχείο wav

Δημιουργούμε δεδομένα για ημίτονο, συνημίτονο

```
t = np.linspace(0, T, n, endpoint=False)
```

```
# "Noisy" data. We want to recover the 1.2 Hz signal from this.
```

```
data = np.sin(1.2*2*np.pi*t) + 1.5*np.cos(9*2*np.pi*t) \
```

```
      + 0.5*np.sin(12.0*2*np.pi*t)
```

**return** : δήλωση επιστροφής

**print** : δήλωση εμφάνισης

Η μεταβλητή **sys argv** είναι μια λίστα συμβολοσειρών που αντιπροσωπεύουν κάποια argument . στον κώδικα Python, μπορούμε να χρησιμοποιήσουμε αυτήν τη λίστα συμβολοσειρών ως είσοδο στο πρόγραμμά μας. Δεδομένου ότι οι λίστες είναι ευρετηριασμένες με ακέραιους αριθμούς μηδέν, μπορούμε να πάρουμε τα μεμονωμένα στοιχεία χρησιμοποιώντας τη σύνταξη λίστας [0]. Για παράδειγμα:

```
script_name = sys.argv[0]
```

στη συγκεκριμένη περίπτωση η `sys.argv[1]` αντιπροσωπεύει το πρώτο argument

```
thefile = 'uploads/'+str(sys.argv[1])
cutoff = float(sys.argv[2])
```

thefile = 'uploads/'+str(sys.argv[1]) και ως δεύτερο είναι η cutoff = float(sys.argv[2])

**fs** : Η συχνότητα δειγματοληψίας του σήματος. Κάθε συχνότητα

στην αποκοπή πρέπει να είναι μεταξύ 0 και fs/2. Η προεπιλογή είναι 2. (**float**)

**nyq** : Αυτή είναι η συχνότητα Nyquist. Κάθε συχνότητα στο *freq* πρέπει να είναι μεταξύ 0 και *nyq* . Η προεπιλογή είναι 1. (**float**)

**def** : εισάγει ορισμό λειτουργίας. Πρέπει να ακολουθείται από το όνομα της λειτουργίας και από τη λίστα των τυπικών παραμέτρων με παρενθέσεις.

**freq** : Τα σημεία δειγματοληψίας συχνότητας. Συνήθως 0.0 έως 1.0 με το 1.0 να είναι Nyquist. Η πρώτη τιμή στο *freq* πρέπει να είναι 0 και η τελευταία τιμή πρέπει να είναι fs/2.

**nfreqs** : Το μέγεθος του πλέγματος παρεμβολής που χρησιμοποιήθηκε για την κατασκευή του φίλτρου(**int**)

**y = lfilter(b,a,data):** φιλτράρει δεδομένα κατά μήκος μονοδιάστατου με φίλτρο IIR ή FIR.(γραμμικό).με **b : array** , **a : array** . Η έξοδος του ψηφιακού φίλτρου.  
**y : array**

**b, a = butter(order, normal\_cutoff, btype='low', analog=True)**

**btype :** {'lowpass', 'highpass', 'bandpass', 'bandstop'}

Ο τύπος φίλτρου. Η προεπιλογή είναι "lowpass"

**analog :** Όταν είναι αληθές, επιστρέφει ένα αναλογικό φίλτρο, διαφορετικά επιστρέφει ένα ψηφιακό φίλτρο.

**έξοδος :** 'ba'.Είδος εξόδου: αριθμητής / παρονομαστής («ba»),

**return b, a :** **b, a :** *ndarray, ndarray* Αριθμητής ( $\beta$ ) και παρονομαστής ( $\alpha$ ) πολυώνυμα του φίλτρου IIR. Επιστρέφεται μόνο εάν `output='ba'`.

**w, h = freqz(b, a, worN=8000)**

**b :** *array* **a :** *array* αριθμητής ενός γραμμικού φίλτρου

**worN :** {None, int, array\_like}

Κανονικά, οι συχνότητες υπολογίζονται από το 0 έως τη συχνότητα Nyquist,  $\pi$  radians / δείγμα (άνω μισό του κύκλου μονάδας). Αν το *σύνολο* είναι True,

υπολογίστε τις συχνότητες από 0 έως  $2 * \pi$  radians / δείγμα.

**w :** *ndarray*. Οι κανονικοποιημένες συχνότητες στις οποίες υπολογίστηκε *h* , σε ακτίνια / δείγμα.

**h :** *ndarray* Η απόκριση συχνότητας

Κώδικας για file upload & apply lowpass

Ένα φίλτρο χαμηλής διέλευσης χρησιμοποιείται για την αφαίρεση των υψηλότερων συχνοτήτων σε ένα σήμα δεδομένων.

Στο πρώτο κατά σειρά πρόγραμμα πρόκειται να σχεδιάσουμε ένα lowpass filter εισάγοντας τις βιβλιοθήκες Matplotlib ,Numpy, scipy και sys.

```
import numpy as np
from scipy.signal import butter, lfilter, freqz
from matplotlib import pyplot as plt
import sys
import scipy.io.wavfile
```

Για να φορτώσουμε ένα αρχείο wav στη Python:

Φορτώνουμε τη βιβλιοθήκη scipy για τη χρήση του module io.wavfile.

```
import scipy.io.wavfile

thefile = 'uploads/'+str(sys.argv[1])

fs_rate, signal = wavfile.read(thefile)
```

Φορτώνουμε το ρυθμό δειγματοληψίας (bps) και τα δεδομένα σήματος (wav)

```
#read wav file
therate,thedata = scipy.io.wavfile.read(thefile, mmap=False)
print(thedata[1:2000])
print(therate)
```

Η έξοδος από το `wavfile.read` είναι ο ρυθμός δειγματοληψίας στο κομμάτι και τα δεδομένα του ακουστικού κύματος. Ο ρυθμός δειγματοληψίας αντιπροσωπεύει τον αριθμό των σημείων δεδομένων που ελήφθησαν ανά δευτερόλεπτο στο αρχείο ήχου.

Η συχνότητα Nyquist είναι το μισό από το ρυθμό δειγματοληψίας. Επειδή τα δεδομένα μας υποβάλλονται σε δειγματοληψία, θέλουμε ένα ψηφιακό φίλτρο, όχι ένα αναλογικό φίλτρο αρά στο `analog=False` βάλουμε `false` και όχι `true`.

```
def butter_lowpass(cutoff, fs, order=5):
    nyq = 0.5 * fs
    normal_cutoff = cutoff / nyq
    b, a = butter(order, normal_cutoff, btype='low', analog=False)
    return b, a
```

Χρησιμοποιήσαμε `scipy.signal.freqz` (όχι τα `freqs`) για να δημιουργήσετε την απόκριση συχνότητας.

```
# Plot the frequency response.
w, h = freqz(b, a, worN=8000)
plt.subplot(2, 1, 1)
plt.plot(0.5*fs*w/np.pi, np.abs(h), 'b')
plt.plot(cutoff, 0.5*np.sqrt(2), 'ko')
plt.axvline(cutoff, color='k')
plt.xlim(0, 0.5*fs)
plt.title("Lowpass Filter Frequency Response")
plt.xlabel('Frequency [Hz]')
plt.grid()
```

Η απόκριση συχνότητας έχει δύο μέρη: απόκριση πλάτους και απόκριση φάσης. Και οι δύο εκπροσωπούνται ως ένα πολύπλοκο σήμα όταν λαμβάνετε την απάντηση από `freqz`. Για να σχεδιάσουμε την απόκριση πλάτους χρησιμοποιήσαμε το `abs`.

```
#Εισαγάγουμε τις βιβλιοθήκες
import numpy as np
from scipy.signal import butter, lfilter, freqz
from matplotlib import pyplot as plt
import sys
import scipy.io.wavfile

thefile = 'uploads/'+str(sys.argv[1])
cutoff = float(sys.argv[2])
```

```

print(cutoff)
print(thefil
#διαβάζουμε το .wav αρχείο
therate,thedata = scipy.io.wavfile.read(thefile, mmap=False)
print(thedata[1:2000])
print(therate)

def butter_lowpass(cutoff, fs, order=5):
    nyq = 0.5 * fs
    normal_cutoff = cutoff / nyq
    b, a = butter(order, normal_cutoff, btype='low', analog=True)
    return b, a

def butter_lowpass_filter(data, cutoff, fs, order=5):
    b, a = butter_lowpass(cutoff, fs, order=order)
    y = lfilter(b, a, data)
    return y

# φίλτρο
order = 6
fs = therate      # sample rate, Hz
#cutoff = 6.667 # desired cutoff frequency of the filter, Hz

# Get the filter coefficients so we can check its frequency response.
b, a = butter_lowpass(cutoff, fs, order)

# Γράφημα για συχνότητα απόκρισης
w, h = freqz(b, a, worN=8000)
plt.subplot(2, 1, 1)
plt.plot(0.5*fs*w/np.pi, np.abs(h), 'b')
plt.plot(cutoff, 0.5*np.sqrt(2), 'ko')
plt.axvline(cutoff, color='k')
plt.xlim(0, 0.5*fs)
plt.title("Lowpass Filter Frequency Response")
plt.xlabel('Frequency [Hz]')
plt.grid()

#δεδομένα
#χρήση του φίλτρου.
# δεδομένα για φιλτράρισμα.
T = 5.0          # seconds
n = int(T * fs)  # total number of samples
t = np.linspace(0, T, n, endpoint=False)
# "Noisy" data. We want to recover the 1.2 Hz signal from this.
data = np.sin(1.2*2*np.pi*t) + 1.5*np.cos(9*2*np.pi*t) \
        + 0.5*np.sin(12.0*2*np.pi*t)
data = thedata[0:220500]
# Filter the data, and plot both the original and filtered signals.
y = butter_lowpass_filter(data, cutoff, fs, order)

#σχεδιάζουμε τα δεδομένα

```

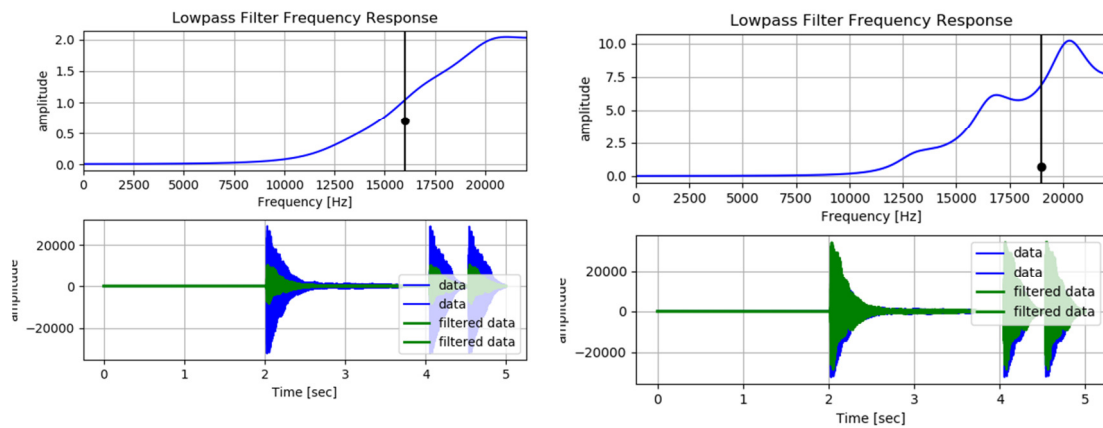
```

plt.subplot(2, 1, 2)
plt.plot(t, data, 'b-', label='data')
plt.plot(t, y, 'g-', linewidth=2, label='filtered data')
plt.xlabel('Time [sec]')
plt.grid()
plt.legend()

plt.subplots_adjust(hspace=0.35)
plt.savefig('C:\\xampp\\htdocs\\img\\filefiltered.png')

# εμφάνιση
plt.show()

```



Στα 2 παραπάνω σχήματα (στα αριστερά) βλέπουμε τη συχνότητα απόκρισης στα 16000 Hz , τα αρχικά δεδομένα του σήματος απεικονίζεται με το μπλε χρώμα και το φιλτραρισμένο με το πράσινο. Στα δεξιά βλέπουμε τη συχνότητα απόκρισης στα 19000Hz . Το lowpass φίλτρο έχει εξασθενήσει τις συχνότητες από 16000Hz ή 19000Hz αντίστοιχα .

### Κώδικας για Lowpass filter

```

import numpy as np
from scipy.signal import butter, lfilter, freqz
from matplotlib import pyplot as plt
import sys

#διαβάζουμε την argument(sys.argv)
cutoff = float(sys.argv[1])
print(cutoff)
def butter_lowpass(cutoff, fs, order=5):
    nyq = 0.5 * fs
    normal_cutoff = cutoff / nyq

    b, a = butter(order, normal_cutoff, btype='low', analog=False)

```

```

    return b, a

def butter_lowpass_filter(data, cutoff, fs, order=5):
    b, a = butter_lowpass(cutoff, fs, order=order)
    y = lfilter(b, a, data)
    return y

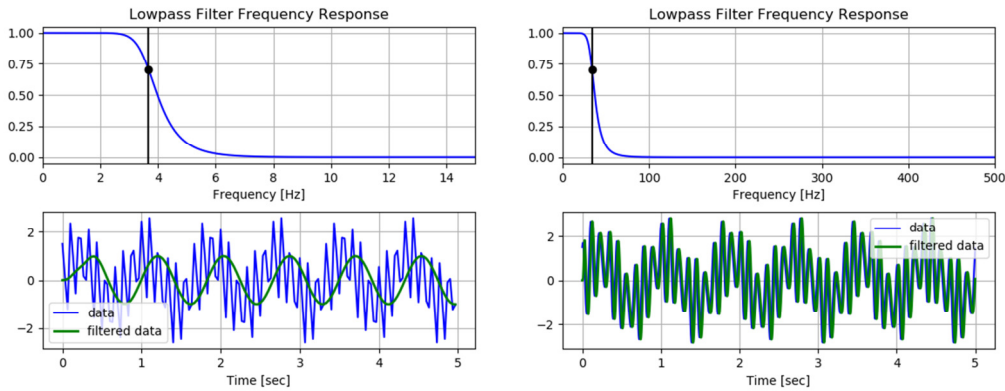
# Filter requirements
order = 6
fs = 30.0          # sample rate, Hz
#cutoff = 6.667   # desired cutoff frequency of the filter, Hz
# συντελεστές φίλτρου για να μπορέσουμε να ελέγξουμε την απόκριση
# συχνότητας.
b, a = butter_lowpass(cutoff, fs, order)

# Γράφημα για συχνότητα απόκρισης
w, h = freqz(b, a, worN=8000)
plt.subplot(2, 1, 1)
plt.plot(0.5*fs*w/np.pi, np.abs(h), 'b')
plt.plot(cutoff, 0.5*np.sqrt(2), 'ko')
plt.axvline(cutoff, color='k')
plt.xlim(0, 0.5*fs)
plt.title("Lowpass Filter Frequency Response")
plt.xlabel('Frequency [Hz]')
plt.grid()
#δεδομένα
#χρήση του φίλτρου.
# δεδομένα για φιλτράρισμα.
T = 5.0           # seconds
n = int(T * fs)   # total number of samples
t = np.linspace(0, T, n, endpoint=False)
# "Noisy" data. Θέλουμε να ανακτήσουμε το σήμα 1,2 Hz
#3 σήματα
data = np.sin(1.2*2*np.pi*t) + 1.5*np.cos(9*2*np.pi*t) \
      + 0.5*np.sin(12.0*2*np.pi*t)

# Filter the data, and plot both the original and filtered signals.
y = butter_lowpass_filter(data, cutoff, fs, order)

plt.subplot(2, 1, 2)
plt.plot(t, data, 'b-', label='data')
plt.plot(t, y, 'g-', linewidth=2, label='filtered data')
plt.xlabel('Time [sec]')
plt.grid()
plt.legend()
plt.subplots_adjust(hspace=0.35)
plt.savefig('C:\\xampp\\htdocs\\img\\filtered.png')
plt.show()

```



## Κώδικας για highpass filter

Για να τρέξουμε το highpass filter εγκαταστήσαμε την βιβλιοθήκη pandas `Pip install pandas`

```

C:\xampp\htdocs>pip install pandas
Collecting pandas
  Downloading https://files.pythonhosted.org/packages/26/fc/d0509d445d2724fbc5f9
  9a6fc9ce7da794873469739b6c94afc166ac2a2/pandas-0.23.4-cp37-cp37m-win32.whl (6.8
  MB)
    100% |#####| 6.8MB 781kB/s
Collecting pytz<2018.7 (from pandas)
  Downloading https://files.pythonhosted.org/packages/f8/0e/2365ddc010afb3d79147
  1dd544e5ee24bf4ece58ab99b16fb465ce6dc0/pytz-2018.7-py2.py3-none-any.whl (506kB
  )
    100% |#####| 512kB 1.6MB/s
Requirement already satisfied: python-dateutil<2.8.0 in c:\python37\lib\site-pa
  ckages (from pandas) (2.7.5)
Requirement already satisfied: numpy>=1.9.0 in c:\python37\lib\site-packages (fr
  om pandas) (1.16.4)
Requirement already satisfied: six>=1.5 in c:\python37\lib\site-packages (from p
  ython-dateutil<2.8.0>pandas) (1.11.0)
Installing collected packages: pytz, pandas
  
```

```

import numpy as np
import pandas as pd
from scipy import signal
import matplotlib.pyplot as plt

def sine_generator(fs, sinefreq, duration):
    T = duration
    nsamples = fs * T
    w = 2. * np.pi * sinefreq
    t_sine = np.linspace(0, T, nsamples, endpoint=False)
    y_sine = np.sin(w * t_sine)
    result = pd.DataFrame({
        'data' : y_sine}, index=t_sine)
    return result

def butter_highpass(cutoff, fs, order=5):
    nyq = 0.5 * fs
    normal_cutoff = cutoff / nyq
  
```

```

    b, a = signal.butter(order, normal_cutoff, btype='high',
analog=False)
    return b, a

def butter_highpass_filter(data, cutoff, fs, order=5):
    b, a = butter_highpass(cutoff, fs, order=order)
    y = signal.filtfilt(b, a, data)
    return y

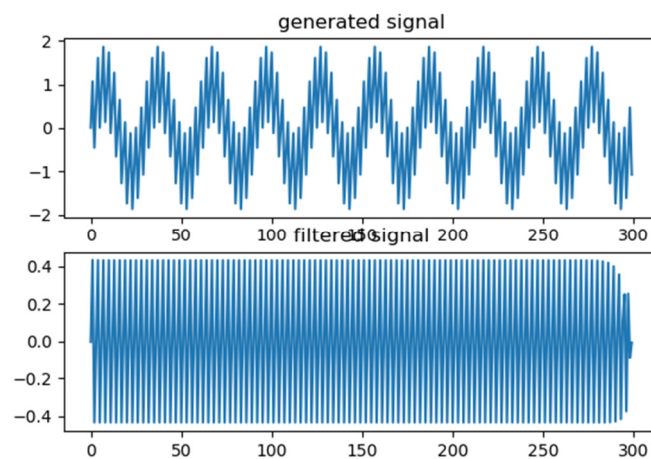
fps = 30
sine_fq = 10 #Hz
duration = 10 #seconds
sine_5Hz = sine_generator(fps, sine_fq, duration)
sine_fq = 1 #Hz
duration = 10 #seconds
sine_1Hz = sine_generator(fps, sine_fq, duration)

sine = sine_5Hz + sine_1Hz

filtered_sine = butter_highpass_filter(sine.data, 10, fps)
plt.figure(figsize=(20,10))
plt.subplot(211)
plt.plot(range(len(sine)), sine)
plt.title('generated signal')
plt.subplot(212)
plt.plot(range(len(filtered_sine)), filtered_sine)
plt.title('filtered signal')
plt.savefig('C:\\xampp\\htdocs\\img\\highfiltered.png')

plt.show()

```



### Κώδικας για recodering

```
import sounddevice as sd
```



```

import numpy as np
import scipy.io.wavfile as wav
import sys
from struct import pack
from matplotlib import pyplot as plt
#import pyaudio
import wave
fs=44100
duration = 5 # seconds
myrecording = sd.rec(duration * fs, samplerate=fs,
channels=2,dtype='float64')
print(type(myrecording))
data=myrecording
print("Recording Audio")
sd.wait()
print( "Audio recording complete , Play Audio")
sd.play(myrecording, fs)
sd.wait()
print ("Play Audio Complete")

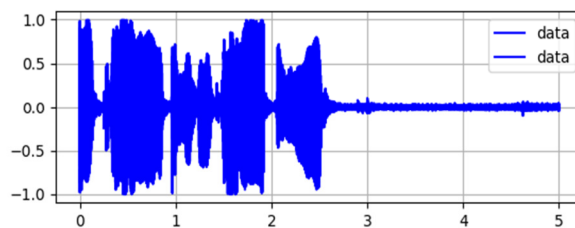
#high filter
path ='C:\\xampp\\htdocs\\wave.wav'
T = 5.0 # seconds
n = int(T * fs) # total number of samples
t = np.linspace(0, T, n, endpoint=False)
#για να το εμφανίσουμε
wav.write(path,fs,myrecording)
plt.subplot(2, 1, 2)
plt.plot(t, myrecording, 'b-', label='data')
#plt.plot(t, y, 'g-', linewidth=2, label='filtered data')
plt.xlabel('Time [sec]')
plt.grid()
plt.legend()
plt.show()

```

```

C:\xampp\htdocs>python recmic.py onoma
<class 'numpy.ndarray'>
Recording Audio
Audio recording complete , Play Audio
Play Audio Complete

```



Φόρτωση αρχείου WAV και εμφάνιση στο πεδίο συχνοτήτων .

Κώδικας για FFT ( Fast Fourier Transform)

Ο μετασχηματισμός Fourier υπολογίζεται σε ένα σήμα χρόνου για να ελέγξει τη συμπεριφορά του στον τομέα συχνοτήτων όπως έχουμε προαναφερθεί. Ο ταχύς μετασχηματισμός Fourier (fast Fourier transform, FFT) είναι ένας αποδοτικός αλγόριθμος για τον υπολογισμό του DFT. Στην πράξη, δεν υπάρχει μόνο ένας αλγόριθμος, αλλά πλήθος από διαφορετικούς αλγόριθμους που επιτυγχάνουν το σκοπό αυτό. Όλοι όμως, έχουν ένα κοινό χαρακτηριστικό: χρειάζονται μόνο  $(N/2) \log_2 N$  μιγαδικούς πολλαπλασιασμούς για τον υπολογισμό ενός DFT N-σημείων.

Στη Python οι μετασχηματισμοί μπορούν να υπολογιστούν με τη βοήθεια της `fftpack`, μια σειρά από λειτουργίες της βιβλιοθήκης `SciPy`. Υπάρχουν δύο χρήσιμες λειτουργίες για τον υπολογισμό και τη λήψη του μετασχηματισμού Fourier από μια διάταξη δειγμάτων:

Η συνάρτηση `scipy.fftpack.fftfreq()` που δημιουργεί τις συχνότητες δειγματοληψίας και το `scipy.fftpack.fft()` υπολογίζει τον γρήγορο μετασχηματισμό Fourier.

Παρακάτω διαβάσαμε ένα αρχείο WAV που επιλέχθηκε και θα τρέξουμε βασικές FFTs σε αυτό

```
import scipy.fftpack
```

Για να φορτώσουμε ένα αρχείο wav στη Python:

Φορτώνουμε τη βιβλιοθήκη `scipy` για τη χρήση του module `io.wavfile`.

```
import scipy.io.wavfile as wavfile
```

```
thefile = 'uploads/'+str(sys.argv[1])
```

```
fs_rate, signal = wavfile.read(thefile)
```

Το ακόλουθο αρχείο που επιλέξαμε είναι ένα σήμα 500 Hz , (η νότα ντο)

Στο επόμενο βήμα είναι η εκτέλεση του FFT καλώντας το `fft()` με τα δεδομένα

```
FFT = abs(scipy.fft(signal))
```

```
freqs = scipy.fftpack.fftfreq(signal.size, t[1]-t[0])
```

`fftfreq` - Επιστρέφει τη συχνότητα που αντιστοιχεί σε κάθε δείγμα `xi` από το αρχείο δείγματος `x [n]` δεδομένων. Αυτή είναι η συχνότητα στην οποία αντιστοιχεί κάθε στοιχείο `fft`.

`fft` - Επιστρέφει τα δεδομένα μετασχηματισμού Fourier από το αρχείο δείγματος. Η θέση των στοιχείων που επιστρέφονται αντιστοιχεί στη θέση του `fftfreq`.

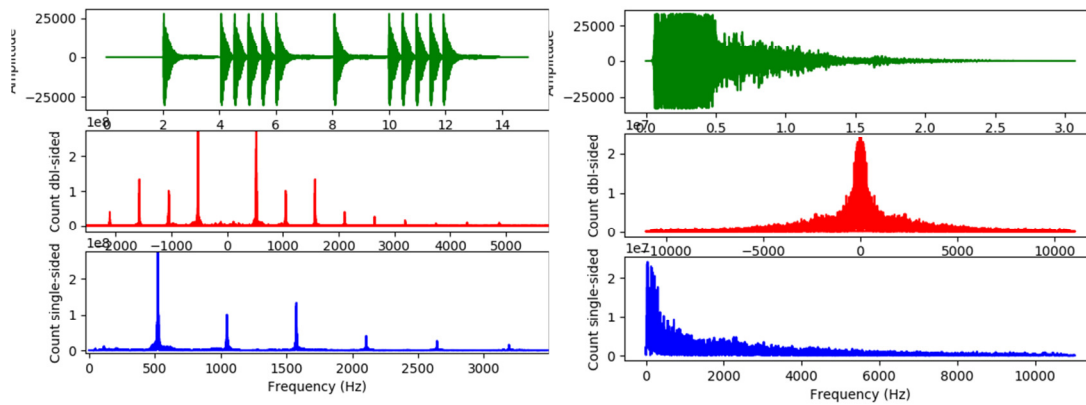
Για παράδειγμα, εάν η συνάρτηση μετασχηματισμού Fourier επιστρέφει `fft = {0,0,5,1}` και `fftfreq = {100,200,300}`, σημαίνει ότι το σήμα έχει ισχύ 0 για συχνότητα 100Hz, ισχύ 0,5 για 200Hz και ισχύ 1 εντός 300Hz. με συχνότητα συχνότητας 300 Hz.

```

from __future__ import print_function
import scipy.io.wavfile as wavfile
import scipy
import sys
import scipy.fftpack
import numpy as np
import os
from matplotlib import pyplot as plt
thefile = 'uploads/'+str(sys.argv[1])
fs_rate, signal = wavfile.read(thefile)
print ("Frequency sampling", fs_rate)
l_audio = len(signal.shape)
print ("Channels", l_audio)
if l_audio == 2:
    signal = signal.sum(axis=1) / 2
N = signal.shape[0]
print ("Complete Samplings N", N)
secs = N / float(fs_rate)
print ("secs", secs)
Ts = 1.0/fs_rate # sampling interval in time
print ("Timestep between samples Ts", Ts)
t = scipy.arange(0, secs, Ts) # time vector as scipy arange field /
numpy.ndarray
FFT = abs(scipy.fft(signal))
FFT_side = FFT[range(int(N/2))] # one side FFT range
freqs = scipy.fftpack.fftfreq(signal.size, t[1]-t[0])
print (freqs)
fft_freqs = np.array(freqs)
freqs_side = freqs[range(int(N/2))] # one side frequency range
fft_freqs_side = np.array(freqs_side)
plt.subplot(311)
p1 = plt.plot(t, signal, "g") # plotting the signal
plt.xlabel('Time')
plt.ylabel('Amplitude')
plt.subplot(312)
p2 = plt.plot(freqs, FFT, "r") # plotting the complete fft spectrum
plt.xlabel('Frequency (Hz)')
plt.ylabel('Count dbl-sided')
plt.subplot(313)
p3 = plt.plot(freqs_side, abs(FFT_side), "b") # plotting the positive
fft spectrum
plt.xlabel('Frequency (Hz)')
plt.ylabel('Count single-sided')

playwav = 'powershell -c (New-Object Media.SoundPlayer
"C:/xampp/htdocs/uploads/'+str(sys.argv[1] )+'").PlaySync();'
print (playwav)
os.system(playwav)
plt.show()

```



Πηγές : <https://stackoverflow.com/questions/25191620/creating-lowpass-filter-in-scipy-understanding-methods-and-units>  
<https://stackoverflow.com/questions/39032325/python-high-pass-filter>  
<https://stackoverflow.com/questions/35344649/reading-input-sound-signal-using-python>  
<https://engineersportal.com/blog/2018/9/13/audio-processing-in-python-part-i-sampling-and-the-fast-fourier-transform>

## **ΒΙΒΛΙΟΓΡΑΦΙΑ**

Δημήτρης Σκαρλάτος : *Εφαρμοσμένη ακουστική (3<sup>η</sup> έκδοση)*[1]

Λουκάς Χαδέλλης : *Τεχνολογία Ήχου*

HWEI P.HSU : *Αναλογικές και Ψηφιακές Τεχνολογίες*

Σημειώσεις Μαθήματος: *Φλώρος Αντρέας, Τεχνολογία Ψηφιακού Ήχου.*

Νικόλαος Α. Αγγελιδάκης : *Εισαγωγή στον προγραμματισμό με την Python 1η Έκδοση*

Γεωργιος Κ.Καραγιαννίδης : *Τελεπικοινωνιακα Συστήματα 2<sup>η</sup> εκδοση*

Α. Βαρδουλάκης : *Εισαγωγή στη θεωρία σημάτων και συστημάτων*

Καλουμπιτσίδης Νίκος: *Σήματα Συστήματα και Αλγόριθμοι (5<sup>η</sup> έκδοση).*

Γεώργιος Β. Μουστακίδης: *Βασικές Τεχνικές Ψηφιακής Επεξεργασίας Σημάτων*

Σημειώσεις : *The Physics of Sound*

Professor J. Malchaire : *Sound measuring instruments*

Γεράσιμος Ποταμιάνος : *Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών Πανεπιστήμιο Θεσσαλίας <http://www.inf.uth.gr/~gpotamianos>*

Behrouz A. Forouzan : *Foundations of Computer Science*

Jean Brainard : *Intensity and Loudness of Sound*

Σημειώσεις : *Ηχος : πολικό διάγραμμα μεγάφωνου , μετρησεις – προσδιορισμος*

Σημειώσεις : *Κεφάλαιο 4 – Συμπύεση Ήχου*

Στεφανακης Νικολαος : *Σημειώσεις για το μάθημα Επεξεργασία Ήχου Φωνής*

Απόστολος Λουφόπουλος : *Σημειώσεις για το μάθημα:Ψηφιακη Επεξεργασια Ηχου*

(Θεωρια) (εκδοχή 1.0)

Νικόλας Τσαπατσούλης: *Συμπίεση και Μετάδοση Πολυμέσων*

Σημειώσεις : *Ψηφιακές Τηλεπικοινωνίες: Εισαγωγή στους Προθεματικούς Κώδικες και περιγραφή του αλγορίθμου Huffman.*

<https://eclass.upatras.gr/modules/units/?course=CEID1110&id=6448>

Ν.Σγούρος : *Συστήματα Πολυμέσων*

Βασίλειος Δ. Ανδριτσάνος: *Εισαγωγή στην ανάλυση Fourier ,Σειρές Fourier*

<https://www.w3schools.com/default.asp>

<https://www.python.org/>

<https://stackoverflow.com/questions/39032325/python-high-pass-filter>

<http://catalysis.gr/mathematics/applied%20mathematics/Fourier/body.html>

<https://matplotlib.org/>

<https://matplotlib.org/users/installing.html>

[https://docs.scipy.org/doc/scipy-](https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.signal.butter.html#scipy.signal.butter)

[0.14.0/reference/generated/scipy.signal.butter.html#scipy.signal.butter](https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.signal.butter.html#scipy.signal.butter)

<https://python-sounddevice.readthedocs.io/en/0.3.12/installation.html>

<https://www.scipy.org/>

<https://pandas.pydata.org/pandas-docs/stable/install.html>

<http://www.technologyuk.net/telecommunications/telecom-principles/pulse-code-modulation.shtml>

<https://audioguy.gr/gr/courses/3/>

<http://ebooks.edu.gr/modules/ebook/show.php/DSB103/173/1206,4408/>

<http://ebooks.edu.gr/modules/ebook/show.php/DSB103/173/1206,4409/>

[http://www.cslab.ece.ntua.gr/~ekall/Science/personal\\_docs/ixos.htm](http://www.cslab.ece.ntua.gr/~ekall/Science/personal_docs/ixos.htm)

<http://www.it.uom.gr/project/MultimediaTechnologyNotes/chap2b-3.htm>

<https://www.archives.gov/preservation/products/definitions/filetypes.html>

<https://www.cl.cam.ac.uk/~jac22/books/mm/book/node96.html>

[https://en.wikipedia.org/wiki/Analog-to-digital\\_converter](https://en.wikipedia.org/wiki/Analog-to-digital_converter)

<https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.io.wavfile.read.html>

[https://www.w3schools.com/bootstrap/tryit.asp?filename=trybs\\_tabs\\_dynamic&stacked=h](https://www.w3schools.com/bootstrap/tryit.asp?filename=trybs_tabs_dynamic&stacked=h)

[https://www.w3schools.com/bootstrap/bootstrap\\_tabs\\_pills.asp](https://www.w3schools.com/bootstrap/bootstrap_tabs_pills.asp)

<https://docs.python.org/2/tutorial/controlflow.html#tut-docstrings>

<http://aggelid.mysch.gr/pythonbook/>

<https://el.wikipedia.org/wiki/Python>

<https://www.sublimetext.com/>

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.butter.html>

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.firwin2.html>

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.lfilter.html>

<https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.io.wavfile.read.html>

[https://www.w3schools.com/tags/att\\_form\\_enctype.asp](https://www.w3schools.com/tags/att_form_enctype.asp)