

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ

ΣΧΟΛΗ:ΔΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ

Τμήμα: ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ

**ΜΕΛΕΤΗ ΤΟΥ ΣΧΕΔΙΑΣΜΟΥ ΚΑΙ
ΜΟΝΤΕΛΟΠΟΙΗΣΗΣ ΨΗΦΙΑΚΩΝ
ΙΣΟΣΤΑΘΜΙΣΤΩΝ ΓΙΑ
ΕΝΣΩΜΑΤΩΜΕΝΑ ΣΥΣΤΗΜΑΤΑ ΗΧΟΥ**

Όνοματεπώνυμο: Γεωρογλίδης Χριστόφορος

Επιβλέπων : Κούτρας Αθανάσιος

ΠΥΡΓΟΣ 2016

ΠΕΡΙΕΧΟΜΕΝΑ

Περίληψη	6
1 Τι είναι φίλτρο.....	7
1.1 Τα χαρακτηριστικά του φίλτρου.....	7
1.2 Ιδανικά φίλτρα	8
1.3 Εξίσωση διαφορών.....	9
2. Απειρης κρουστικής απόκρισης φίλτρα(IIR).....	10
2.1 Φίλτρα άπειρης κρουστικής απόκρισης(IIR).....	10
2.2 Αναλογικά IIR φίλτρα.....	11
2.2.1 Butterworth.....	11
2.2.2 Chybychev I.....	12
2.2.3 Chybychev II.....	14
2.2.4 Elliptic.....	15
2.3 Σύγκριση αναλογικών IIR φίλτρων.....	16
2.4 Μετατροπή αναλογικών IIR σε ψηφιακά.....	17
2.4.1 Μέθοδος Αμετάβλητης κρουστικής.....	17
2.4.2 Διγραμμικός μετασχηματισμός.....	18
2.5 Γενικά χαρακτηριστικά IIR φίλτρων.....	20
3 .Μοντελοποίηση φίλτρων Simulink matlab.....	21
3.1 2ος τρόπος σχεδίασης φίλτρου.....	23
3.2 Δομές συστημάτων άπειρης κρουστικής απόκρισης (IIR).....	25
3.2.1 Direct Form.....	25
3.2.2 Direct Form II.....	26
3.2.3 Δομή καταρράκτη (σύνδεση σε σειρά cascaded).....	27
3.2.4 Παράλληλη σύνδεση(σύνδεση παράλληλα parallel).....	27
3.3 Σύνδεση φίλτρων στο simulink.....	28
3.4 Ηχητικό σύστημα(only cut equalizer).....	30
4 Equalizers(μεταβλητού κερδους).....	33
4.1 Ολοπερατο φίλτρο(all pass).....	33
4.2 low pass/high pass bandpass/Bandreject.....	35
4.2.1 low pass.....	35
4.2.2 high pass.....	35
4.2.3 bandpass/bandreject.....	36
4.3 Shelving φίλτρα	37
4.3.1 Low Frequency shelving φίλτρο	38
4.3.1.1 Πρώτη μέθοδος σχεδιασμού (απευθείας μέθοδος σχεδιασμού).....	38
4.3.1.2 Δεύτερη μέθοδος σχεδιασμού low frequency shelving φίλτρου (ημιπαραμετρική).....	40
4.3.1.3 Τρίτη μέθοδος σχεδιασμού low frequency shelving φίλτρου.....	43
4.3.2 High Frequency shelving φίλτρο	46
4.3.2.1 Πρώτη μέθοδος(απευθείας μέθοδος σχεδιασμού)	46
4.3.2.2 Δεύτερη μέθοδος σχεδιασμού High frequency shelving φίλτρου(ημιπαραμετρική).....	47
4.3.2.3 Τρίτη μέθοδος σχεδιασμού high frequency shelving φίλτρου	50

4.4 Peaking/notch φίλτρο.....	52
4.4.1 Πρώτη μέθοδος peaking φίλτρο(απευθείας μέθοδος σχεδιασμού).....	53
4.4.2 Δεύτερη μέθοδος σχεδιασμού peak φίλτρου(full παραμετρική)	57
4.3 Χρήση τριών μπαντών (3bands) ισοσταθμιστής (full parametric)	60
4.5 Σφάλματα κβαντίσου κατά την υλοποίηση.....	62
4.5.1 Δυαδική αναπαράσταση των συντελεστών	63
4.5.2 Σφάλματα κβαντίσου IIR φίλτρα	62
4.6 Audio equalizer(guide Matlab(θεωρητικός σχεδιασμός εφαρμογής).....	64
4.7 Ενδεικτική εξήγηση του κώδικα (GUI Matlab)	66
4.7.1 Guide Matlab.....	70
4.7.2 Κώδικας.....	70
4.7.3 Slider Callback.....	70
4.7.4 Edit box Callback.....	71
4.7.5 Κουμπια.....	72
4.7.6 Απόκριση συχνότητας.....	74
4.7.7 Φάσμα.....	75
4.8 Συμπεράσματα.....	76
Βιβλιογραφία.....	78
Παραρτήματα.....	80

Εικόνες

Εικόνα 1.1 Ψηφιακό φίλτρο	7
Εικόνα 1.2 Αποκρίσεις ιδανικών φίλτρων	8
Εικόνα 1.3 Χαρακτηριστικά μη ιδανικού φίλτρου	9
Εικόνα 2.1 Μεταφορά από το splane στο zplane(μοναδιασος κύκλος)	11
Εικόνα 2.2 αποκρίσεις συχνότητας IIR butter	12
Εικόνα 2.3 Πόλοι αναλογικού φίλτρου Butterworth τάξη(N=12)	12
Εικόνα 2.4 Αποκρίσεις συχνότητας chybyshev1	13
Εικόνα 2.5 Θέσεις των πόλων chybyshev1	13
Εικόνα 2.6 Αποκρίσεις συχνότητας chybyshev2	14
Εικόνα 2.7 Μηδενικά και πόλοι chybyshev2	14
Εικόνα 2.8 Αποκρίσεις συχνότητας ελλειπτικού.....	15
Εικόνα 2.9 Μηδενικά και πολοι ελλειπτικού φίλτρου	15
Εικόνα 2.10 Σύγκριση Butter,Chybushev1,ελλειπτικο	16
Εικόνα 2.11 Απεικόνιση των σημείων splane($\sigma < 0$) στο zplane μέθοδος αμετάβλητης κρουστικής απόκρισης. ενα προς πολλά απεικόνιση	18
Εικόνα 2.12	18
Εικόνα 2.13 Σχέση αναλογικής και ψηφιακής συχνότητας διγραμμικού μετασχηματισμού.....	19
Εικόνα 2.14 IIR ελλειπτικό φίλτρο 8 τάξης ,κυμάτωση στην ζώνη διέλευσης 2 db. και -80 db. απόσβεση στις ζώνες αποκοπής ψηφιακό με την μέθοδο του διγραμμικού μετασχηματισμού	20
Εικόνα 3.1 Στοιχεία βιβλιοθήκης Simulink	21
Εικόνα 3.2 Μοντέλο IIR φίλτρου Butterworth είσοδος στο φίλτρο δυο ημίτονα 100 και 120 Hz με πλάτος .22	
Εικόνα 3.3 IIR Butterworth bandpass $f_{c1}=250$ και $f_{c2}=400$ Hz 6ης τάξης $f_s=2000$ Hz	22
Εικόνα 3.4 Φάσμα πεδίο συχνότητας είσοδος 2 ημίτονων 0.1 και 0.25 KHz πριν το φιλτράρισμα $f_s/2=1000$	22
Εικόνα 3.5 Φάσμα πεδίο συχνότητας μετά το φιλτράρισμα $F_s/2=1000$	23
Εικόνα 3.6 Bandpass Butterworth $N=16$ $f_s=2000/2=1000$	23
Εικόνα 3.7 Συντελεστές	23
Εικόνα 3.8 Απόκριση συχνότητας IIR butter bandpass Direct form II.....	24
Εικόνα 3.9 Απόκριση συχνότητας	24
Εικόνα 3.10 Δομή υλοποίησης Direct Form II	24
Εικόνα 3.11 IIR Direct Form I	25
Εικόνα 3.12 IIR Direct form II	26
Εικόνα 3.13 2ης τάξης IIR φίλτρο δομή καταρράκτη	27
Εικόνα 3.14 Σύνδεση σε σειρά τριών φίλτρων 2ης τάξης. Αρα φίλτρο 6 τάξης σύνδεση σε σειρά ευθεία μορφή II. Η έξοδος ενός φίλτρου αποτελεί την είσοδο στο επόμενη cascade(z)=H1(z)H2(z)H3(z)	27
Εικόνα 3.15 Παράλληλη σύνδεση	28
Εικόνα 3.16 3ης τάξης φίλτρο σύνδεση παράλληλα ενός φίλτρου 1ης τάξης και ένα 2 τάξης .Η σύνδεση φίλτρων παράλληλα είναι η πρόσθεση των εξόδων των δυο φίλτρων	28
Εικόνα 3.17 Φιλτράρισμα της εισόδου με δυο IIR butterworth bandpass και bandstop κερδους σε db και συνδεση τους παραλληλα	28
Εικόνα 3.18 IIR butterworth bandpass $f_{c1}=8000$ $f_{c2}=9000$ Hz.....	29
Εικόνα 3.19 IIR butterworth bandstop $f_{c1}=2000$ $f_{c2}=2500$ Hz	29
Εικόνα 3.20 Φάσμα συχνοτήτων της εισόδου μετα το φιλτράρισμα	30
Εικόνα 3.21 Προσομοίωση στο Simulink του ματλαμπ ενός only cut equalizer	30
Εικόνα 3.22 IIR Butterworth bandstop φίλτρο με κεντρική συχνότητα στα 250 hz(0.25 kHz)	31
Εικόνα 3.23 Διάγραμμα πόλων μηδενικών του ιδίου φίλτρου bandstop f_c 250Hz 4 ^{ης} τάξης	32
Εικόνα 3.24 Καθυστέρηση ομάδας IIR φίλτρου bandstop 0.25 KHz	32
Εικόνα 3.25 Συνάρτηση μεταφοράς 4ης τάξης bandstop φίλτρου	32
Εικόνα 4.1 Απόκριση συχνότητας ολοπερατού φίλτρου 1ης τάξης	34
Εικόνα 4.2 Απόκριση φάσης πρώτης τάξης ολοπερατού φίλτρου $f_c=1000$ άρα $c=-0.8677$ -90 μοίρες.....	34
Εικόνα 4.3 Δομή κατασκευής 1ης τάξης χαμηλοπερατό φίλτρο	35
Εικόνα 4.4 Δομή κατασκευής 1ης τάξης high pass filter.....	36
Εικόνα 4.5 2ης τάξης bandpass φίλτρο	36
Εικόνα 4.6 2ης τάξης bandreject φίλτρο.....	37
Εικόνα 4.7 Low shelving φίλτρο συχνότητα αποκοπής 0.06kHz	38
Εικόνα 4.8 Μοντελοποίηση στο simulink lowfrequency shelving φίλτρο	39
Εικόνα 4.9 Direct form II	39

Εικόνα 4.10 Feed Forward Feed Backward μέθοδος για boost και cut	41
Εικόνα 4.11 Feed Forward Feed Backward μέθοδος στο Simulink lowshelving	41
Εικόνα 4.12 Απόκριση συχνότητας lowfrequency shelving boost	42
Εικόνα 4.13 Απόκριση συχνότητας lowfrequency shelving cut	42
Εικόνα 4.14 Low frequency shelving φίλτρο $y(n)=0.5*H0*(x(n)+A(z))+x(n)$	44
Εικόνα 4.15 Πρώτης τάξης allpass φίλτρου	45
Εικόνα 4.16 lowshelving cut	45
Εικόνα 4.17 highshelving cut	46
Εικόνα 4.18 High frequency shelving φίλτρο με $f_c=16000$	47
Εικόνα 4.19 Μοντελοποίηση στο Simulink highfrequency shelving φίλτρο	47
Εικόνα 4.20 Feed Forward Feed Backward μέθοδος για boost και cut	48
Εικόνα 4.21 Feed Forward Feed Backward μέθοδος στο Simulink highshelving	49
Εικόνα 4.22 Απόκριση συχνότητας highfrequency shelving boost	49
Εικόνα 4.23 Απόκριση συχνότητας highfrequency shelving cut	50
Εικόνα 4.24 High frequency shelving filter $y(n)=0.5*H0*(x(n)-A(z))+x(n)$	51
Εικόνα 4.25 Πρώτης τάξης allpass φίλτρο	52
Εικόνα 4.26 Αποκρίσεις συχνότητας 2 peaking φίλτρων με $g=15$ db και $Q=4$ $g=10$ db και $Q=8$	53
Εικόνα 4.27 peaking φίλτρο	54
Εικόνα 4.28 Απόκριση συχνότητας peaking $g=15$ $Q=4$ $f_0=1000$ Hz και notch $g=15$ $Q=8$ ΚΑΙ $f_0=5000$ Hz σύνδεση σε σειρά(cascade)	54
Εικόνα 4.29 Φάσμα σήματος μετά την εφαρμογή των φίλτρων στο σήμα εισόδου(peaking 1000Hz και notch 5000Hz $fft(y)$ όπου $y=filter(hcas,y)$	55
Εικόνα 4.30 Δομή υλοποίησης peak φίλτρου Direct form II	55
Εικόνα 4.31 Μοντελοποίηση peaking φίλτρο	56
Εικόνα 4.32 Απόκριση συχνότητας peak φίλτρου $G=15$ db	56
Εικόνα 4.33 Δομή υλοποίησης 2ης τάξης allpass φίλτρο	57
Εικόνα 4.34 Peaking/Notch φίλτρο 2ης τάξης $y(n)=H0/2*(x(n)-A2(z))+x(n)$	58
Εικόνα 4.35 Peaking/Notch φίλτρο 2ης τάξης $y(n)=H0/2*(x(n)+A2(z))+x(n)$	59
Εικόνα 4.36 Peaking φίλτρο 5000Hz	59
Εικόνα 4.37 Notch 5000Hz simulink	60
Εικόνα 4.38 Notch φίλτρο	60
Εικόνα 4.39 Full parametric σχεδιασμός φίλτρων	61
Εικόνα 4.40 Σύνδεση σε σειρά full parametric eq	61
Εικόνα 4.41 Απόκριση συχνότητας παραμετρικού ισοσθμιστή στο Simulink	59
Εικόνα 4.42 Equalizer	64
Εικόνα 4.43 Equalizer	64
Εικόνα 4.44 Βιβλιοθήκη GUI Matlab	66
Εικόνα 4.45 παράθυρο αναζήτησης αρχείο. Fig	66
Εικόνα 4.46 Εργαλεία σχεδίασης GUI	67
Εικόνα 4.47 Επιθεωρητής ιδιότητας αντικειμένου	68

Πίνακες

Πίνακας 4.1 Βασικά αντικείμενα του GUI

Πίνακας 4.2 Ιδιότητες των αντικειμένων

Ευχαριστίες

Ευχαριστώ τον κ Μουρτζόπουλο για την βοήθειά του ,και τον κ Βαρσάμη. Επίσης τον Μάνο από το DareStudio Recordings.



ΠΕΡΙΛΗΨΗ

Η πτυχιακή έχει να κάνει με την μελέτη του σχεδιασμού και την μοντελοποίηση των φίλτρων άπειρης κρουστικής καθώς και των ισοσταθμιστών ως προς τους μεθόδους σχεδίασης ,τα χαρακτηριστικά τους την δομή κατασκευής και την σύγκριση ως προς τα πλεονεκτήματα τους και τα μειονεκτήματα τους. Αναλύεται το θεωρητικό μέρος για τον σχεδιασμό και την μοντελοποίηση των φίλτρων και στην συνέχεια γίνεται η πρακτική εφαρμογή όλων αυτών των θεωριών για την δημιουργία ενός ηχητικού συστήματος για περαιτέρω πειραματισμό . Ο σχεδιασμός και η μοντελοποίηση τους σε πρακτικό επίπεδο έγινε σε περιβάλλον Matlab.Μέσω της βιβλιοθήκης Simulink του Matlab έγινε η προσπάθεια σχεδιασμού ενός συστήματος ήχου με την σύνδεση ψηφιακών συγκεκριμένων περιοχών-φίλτρων με σκοπό την προσομοίωση της λειτουργίας ενός ισοσταθμιστή κατηγορίας only cut. Τέλος ενδιαφέρον παρουσιάζουν οι ισοσταθμιστές μεταβλητού κέρδους καθώς και οι διάφοροι μέθοδοι σχεδιασμών τους όπου χρησιμοποιούνται σε διαφορά ενσωματωμένα συστήματα επεξεργασίας ήχου(ισοσταθμιστές ήχου-equalizers).Έτσι μέσω του guide(GUI=graphical user interface=περιβάλλον διεπαφής χρήστη) ο χρήστης μπορεί να αλληλοεπιδρά με το ηχητικό σύστημα και να καθορίζει εκείνος, στην προκειμένη περίπτωση μόνο το κέρδος(\pm db.) ενίσχυση ή εξασθένηση μιας συγκεκριμένη περιοχής στο φάσμα συχνοτήτων, εφόσον μιλάμε για γραφικό ισοσταθμιστή

Abstract

This document has been produced in order to provide information regarding the design and the modeling of filters. It also refers to equalizers with variable gain.

The following stages of this document provide an overall analysis to the subject matter through a wide range of experimentation and continuous testing, utilizing various methods and tools (Matlab, Simulink, G.U.I.).

1.ΦΙΛΤΡΟ

Αρχικά θα πρέπει να πούμε ότι τα φίλτρα στα οποία θα αναφερθούμε είναι IIR (άπειρης κρουστικής απόκρισης. Είναι μια από τις δυο κατηγορίες των ΓΧΑ φίλτρων όπου αυτού του είδους τα συστήματα μπορούν να περιγράψουν από την κρουστική απόκριση(η έξοδος του συστήματος όταν αυτό έχει ως είσοδο την διακριτή ώση) και η κρουστική τους απόκριση έχει άπειρους ορούς δηλαδή μη μηδενικές τιμές. Αλλά ας δούμε κάποια εισαγωγικά πράγματα για αυτού του είδους τα φίλτρα που θα αποτελέσουν βάση και για τους ισοσταθμιστές

Ένα ψηφιακό φίλτρο είναι ένα σύστημα διακριτού χρόνου που πραγματοποιεί κάποιο μετασχηματισμό σε ένα ψηφιακό σήμα $x(n)$ (έχει υποστεί δειγματοληψία η είσοδος)και παράγει μια έξοδο $y(n)$ (απόκριση)



Εικόνα 1.1 Ψηφιακό φίλτρο

Το φίλτρο είναι ένας τελεστής T δηλαδή η απεικόνιση μιας συνάρτησης σε μια άλλη. Τα χαρακτηριστικά του τελεστή είναι τα χαρακτηριστικά του φίλτρου. Η έξοδος $y(n)$ είναι αποτέλεσμα της εισόδου και του μετασχηματισμού T . Τα ψηφιακά φίλτρα ονομάζονται και φίλτρα διακριτού χρόνου, δέχονται ψηφιακό σήμα στη είσοδό τους. Το σήμα εισόδου είναι διακριτού χρόνου(έχει υποστεί δειγματοληψία) και το σήμα εξόδου είναι επίσης διακριτού χρόνου

Τις περισσότερες φορές που χρησιμοποιούμε ψηφιακά φίλτρα πρέπει η έξοδος του να μετατραπεί σε αναλογική δηλαδή μετά την επεξεργασία, πρέπει να προσθέσουμε ένα κύκλωμα για να μετατρέψουμε το ψηφιακό σε αναλογικό. Αυτό το κύκλωμα λέγεται Μετατροπέας Ψηφιακού σε Αναλογικό (Digital to Analog converter). Στην πτυχιακή θα αναφερθούμε στα ενεργά φίλτρα (δηλαδή έχουν ενεργά στοιχεία αλλά και παθητικά στοιχεία) έτσι μπορούν να ενισχύουν και να αποκόπτουν συχνότητες ενώ τα παθητικά μόνο αποκόπτουν συχνότητες

1.1 ΤΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΟΥ ΦΙΛΤΡΟΥ

Τα χαρακτηριστικά ενός φίλτρου είναι:

Γραμμικό: είναι το φίλτρο ή σύστημα για το οποίο ισχύει η αρχή της υπέρθεσης δηλαδή αν η είσοδος στο φίλτρο αποτελείται από ένα γραμμικό συνδυασμό σημάτων τότε η έξοδος του συστήματος θα είναι ίση με τον γραμμικό συνδυασμό των εξόδων του συστήματος δηλαδή αν

$Y1=T[x1(n)]$ και $y2=T[x2(n)]$ δηλαδή αν $y1$ η έξοδος του φίλτρου σε είσοδο $x1$ και $y2$ η έξοδος του ίδιου φίλτρου σε είσοδο $x2$ τότε η έξοδος του φίλτρου με είσοδο $ax1(n)+bx2(n)$ θα είναι ίση με $ay1(n)+by2(n)$ a και b πραγματικοί αριθμοί

$$y(n)=T[ax1(n)+bx2(n)]=aT[x1(n)]+bT[x2(n)]=ay1(n)+ay2(n) \quad (\text{Sophocles Orfanidis 2010})$$

(1.1.1)

Χρονικά αμετάβλητο: είναι το φίλτρο στο οποίο μια χρονική μετατόπιση της εισόδου του αντιστοιχεί στην ίδια χρονική μετατόπιση της εξόδου του δηλαδή είναι ένα φίλτρο σύστημα που η συμπεριφορά του και οι ιδιότητες του δεν αλλάζουν με το χρόνο δηλαδή αν $y(n)$ η έξοδος του συστήματος με $x(n)$ τότε μια χρονική μετατόπιση ίση με $n-1$ ή έξοδος του ίδιου συστήματος $y(n-1)$ αντιστοιχεί σε είσοδο $x(n-1)$

Τα συστήματα που ικανοποιούν τα δυο αυτά χαρακτηριστικά λέγονται Γραμμικά Χρονικά Αμετάβλητα (Linear Time invariant)(LTI)

Ευσταθές είναι το φίλτρο που κάθε περιορισμένη είσοδος παράγει μια περιορισμένη έξοδο (BIBO=bounded input bounded output)

Αν $|x(n)| < \infty$ τότε $|y(n)| < \infty$
 (1.1.2)

(Sophocles Orfanidis 2010)

Αιτιατό (causal) σύστημα είναι εκείνο του οποίου η έξοδος, σε κάθε χρονική στιγμή, εξαρτάται μόνο από τις τιμές του σήματος εισόδου στην παρούσα χρονική στιγμή και σε προηγούμενες χρονικές στιγμές. Μπορούμε να πούμε ότι το φίλτρο δεν παράγει καμία έξοδος αν δεν εφαρμοστεί σε αυτό μια είσοδος ένα φίλτρο είναι αιτιατό αν η κρουστική του απόκριση είναι $\varepsilon h(n)=0$ για $n<0$
 (1.1.3) (Sophocles Orfanidis 2010)

Ένα φίλτρο περιγράφεται σε δύο διαφορετικά πεδία, στο πεδίο του χρόνου και στο πεδίο της συχνότητας. Ο σχεδιασμός και η μελέτη των φίλτρων στο πεδίο του χρόνου γίνεται με την βοήθεια της κρουστικής απόκρισης Στο πεδίο της συχνότητας περιγράφεται με τη συνάρτηση μεταφοράς $H(s)$,

$$H(s) = \frac{Y(s)}{X(s)} = \frac{N(s)}{D(s)} \quad (1.1.4)$$

οπου $Y(s)$ η κατά Laplace έξοδος $N(s)$ (μηδενικά) , και $X(s)$ η κατά Laplace είσοδος $D(s)$ (πόλοι) η οποία προκύπτει από τον μετασχηματισμό Fourier της κρουστικής απόκρισης για αναλογικά φίλτρα ή τον μετασχηματισμό (z) της κρουστικής απόκρισης για τα ψηφιακά. Από τη συνάρτηση μεταφοράς βρίσκουμε την απόκριση πλάτους και φάσης. Με τη βοήθεια της συνάρτησης μεταφοράς μπορούμε να υπολογίσουμε την απόκριση του φίλτρου για οποιαδήποτε σήμα εισόδου

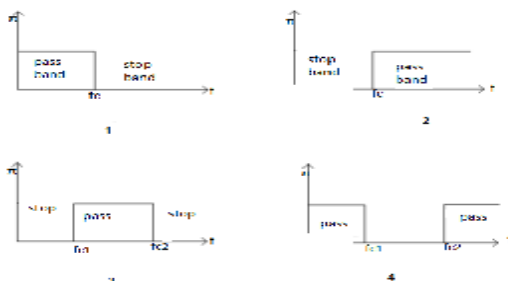
Η απόκριση πλάτους είναι η γραφική παράσταση που μπορούμε να βγάλουμε χρήσιμα συμπεράσματα για τα χαρακτηριστικά και την συμπεριφορά του φίλτρου πχ το είδος του, την προσέγγισή του, τη συχνότητα αποκοπής, την τάξη του κ.α.

Η απόκριση φάσης μας δίνει το ποσό μετατοπίζεται η φάση του σήματος στην έξοδο σε συνάρτηση με την συχνότητα. Μια αλλαγή στη φάση ενός σήματος, σημαίνει καθυστέρηση(delay) οπότε το σήμα παραμορφώνεται αλλά επειδή το αυτί δεν αντιλαμβάνεται τις μεταβολές-παραμορφώσεις της φάσης όσο τις μεταβολές στο πλάτος ενός σήματος .Οπότε δεν μας ενδιαφέρει τόσο η απόκριση φάσης πχ στην επεξεργασία μουσικής ήχου κ.α. .Ωστόσο σε κάποιες εφαρμογές μας νοιάζει η φάση ενός σήματος γιατί μας νοιάζει η μορφή του σήματος όπως στις τηλεπικοινωνίες η σε ένα καρδιογράφημα

1.2 Ιδανικά φίλτρα

Η καμπύλη απόκρισης των ιδανικών φίλτρων δίνεται στο παρακάτω σχήμα. Από αυτές τις αποκρίσεις καταλαβαίνουμε ότι το ιδανικό για ένα φίλτρο είναι η επίπεδη διέλευση όλων των συχνοτήτων, η άμεση αποκοπή από το σημείο που ορίσαμε (δηλαδή την συχνότητα αποκοπής f_c), και η αποκοπή όλων των συχνοτήτων που δεν θέλουμε . Το πλάτος των συχνοτήτων στη ζώνη διέλευσης είναι ίσο με 1 και παραμένει σταθερό, ενώ το πλάτος στη ζώνη αποκοπής είναι πάντα 0. Τα φίλτρα ταξινομούνται σε

1) βαθυπερατά(low pass),2)υψηπερατά(high pass),3)ζωνοπερατά(bandpass) και 4)απόρριψης ζώνης (band stop)



Εικόνα 1.2 Αποκρίσεις ιδανικών φίλτρων

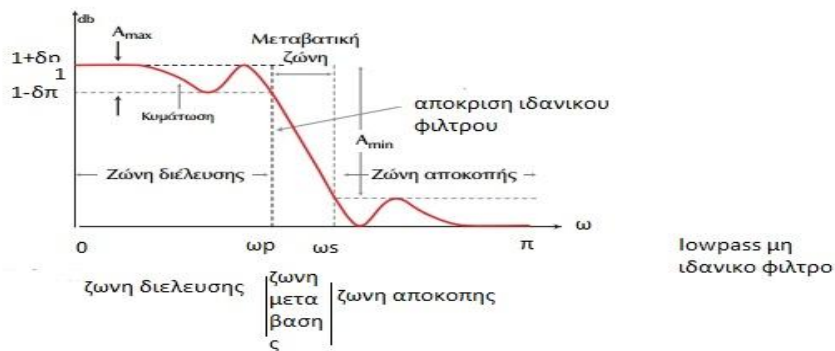
Όπου 1 low pass $\pi=1$ για $f \leq f_c$ αλλού $\pi=0$

Όπου 2 high pass $\pi=1$ για $f \geq f_c$ αλλού $\pi=0$

Όπου 3 bandpass $\pi=1$ για $f > fc1$ και $f < fc2$

Όπου 4 band stop $\pi=0$ για $f > fc1$ και $f < fc2$ αλλού 1

Ωστόσο δεν είναι δυνατή η υλοποίηση τέτοιων φίλτρων από πραγματικά φίλτρα (αιτιατά). Μιας και αυτά έχουν αμφίπλευρη κρουστική απόκριση (δεν είναι αιτιατά). (Sophocles Orfanidis 2010) Τα πραγματικά φίλτρα δεν έχουν απότομη αποκοπή από τη ζώνη διέλευσης στη ζώνη αποκοπής αλλά χρειάζονται κάποιον χρόνο για τη μετάβαση. Η περιοχή στην οποία γίνεται η μετάβαση αυτή, ονομάζεται ζώνη μετάβασης και μειώνεται όσο αυξάνεται η τάξη του πραγματικού φίλτρου. Η περιοχή συχνοτήτων από όπου επιτρέπεται η διέλευση ονομάζεται ζώνη διέλευσης (passband) ενώ η περιοχή που δεν επιτρέπεται η διέλευση ονομάζεται ζώνη αποκοπής (stopband). Επίσης το κέρδος στην ζώνη διέλευσης και στη ζώνη αποκοπής δεν είναι σταθερό αλλά μεταβάλλεται, μπορεί να έχει μια κυμάτωση ± 1 και 0 για την ζώνη αποκοπής



Εικόνα 1.3 Χαρακτηριστικά μη ιδανικού φίλτρου

1.3 Εξίσωση διαφορών

Μια κατηγορία συστημάτων-φίλτρων είναι αυτή που τα σήματα εισόδου-εξόδου συνδέονται με μια εξίσωση διαφορών (difference equation) της μορφής

$$y(n) = \sum_{k=0}^M b_k x(n-k) - \sum_{k=1}^N a_k y(n-k) \quad (\text{John Proakis Dimitris \& G Manolakis 2007}) \quad (1.3.1)$$

όπου $x(n)$ η ακολουθία εισόδου, $y(n)$ η ακολουθία εξόδου και a_k b_k οι συντελεστές του συστήματος. Η εξίσωση αυτή μας λέει ότι η τρέχουσα τιμή του δείγματος εξόδου προκύπτει ως γραμμικός συνδυασμός του παρόντος και των προηγούμενων δειγμάτων της εισόδου, καθώς και από προηγούμενες τιμές της εξόδου. Η σύνδεση μεταξύ της εξίσωσης αυτής και της συνάρτησης μεταφοράς του συστήματος μπορεί εύκολα να γίνει με τη βοήθεια των ιδιοτήτων της γραμμικότητας και της ολίσθησης στο χρόνο. Έτσι αν πάρουμε το M.Z και των δυο μελών της τότε

$$Y(z) \sum_{k=0}^M b_k z^{-k} - \sum_{k=1}^N a_k z^{-k} Y(z) \quad (1.3.2)$$

Άρα

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad (\text{John Proakis Dimitris \& G Manolakis 2007}) \quad (1.3.3)$$

όπου ο αριθμητής είναι τα μηδενικά και ο παρονομαστής πόλοι

Τα συστήματα με συνάρτηση μεταφοράς της μορφής (1.3.3) στην οποία τουλάχιστον ένας από τους συντελεστές του παρονομαστή είναι διάφορος του μηδενός ονομάζονται συστήματα άπειρης κρουστικής απόκρισης (IIR, Infinite Impulse Response), επειδή η αντίστοιχη κρουστική απόκριση $h(n)$ αποτελείται από άπειρους όρους (John Proakis & Dimitris G Manolakis 2007)

Μια άλλη κατηγοριοποίηση των συστημάτων αυτών είναι σε επαναληπτικά. Επαναληπτικά είναι τα συστήματα στα οποία η εξίσωση διαφορών έχει την μορφή (1.3.2) με $b_i \neq 0$. Σ' αυτά τα συστήματα η έξοδος $y(n)$ σε κάθε χρονική στιγμή εξαρτάται και από τιμές της εξόδου σε

προηγούμενες χρονικές στιγμές $y(n-1), y(n-2), \dots$. Έχουμε δηλαδή, ανατροφοδότηση (feedback) τιμών της εξόδου στο σύστημα .

2. Άπειρης κρουστικής απόκρισης φίλτρα(IIR)

Ένα γραμμικό χρονικά αναλλοίωτο IIR φίλτρο περιγράφεται από την εξίσωση διαφορών

$$y(n) = \sum_{k=0}^M b_k x(n-k) - \sum_{k=1}^N a_k y(n-k) \quad (2.1)$$

Εάν σε αυτήν την εξίσωση διαφορών $a_k \neq 0$ για $k=0,1,2$ τότε καταλαβαίνουμε ότι εξόδος σε κάθε χρονική στιγμή προκύπτει ως γραμμικός συνδυασμός της τρέχουσας και των προηγούμενων τιμών των εισόδων καθώς και από τις προηγούμενες τιμές της εξόδου ($y(n-1), y(n-2), \dots$). Από αυτό συμπεραίνουμε ότι έχουμε ανατροφοδότηση τιμών εξόδων στο σύστημα δηλαδή αν ακόμα και η είσοδος x μηδενιστεί τότε οι οροί $y(n-1)$ και $y(n-2)$ θα δημιουργούν νέους ορούς μη μηδενικούς . Τέτοια φίλτρα ονομάζονται επαναληπτικά ή άπειρης κρουστικής απόκρισης (δηλαδή η κρουστική απόκριση θα αποτελείται από άπειρους όρους μη μηδενικών τιμών)

Παίρνοντας τον μετασχηματισμό z της εξόδου δηλαδή

$$Yz = \sum_{k=0}^M b_k z^{-k} X(z) - \sum_{k=1}^N a_k z^{-k} Y(z) \quad (2.2)$$

Οπότε

$$Y(z) \left(1 + \sum_{k=1}^N a_k z^{-k} \right) = X(z) \sum_{k=0}^M b_k z^{-k} \quad (2.3)$$

Αρα η συνάρτηση μεταφοράς $H(z)$ θα είναι

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad (2.4)$$

όπου b_k και a_k είναι οι συντελεστές του φίλτρου και N είναι η τάξη του φίλτρου όπου το N ακέραιος Έτσι λοιπόν αν θέσουμε την $\delta(n)$ σαν είσοδο σε ΓΧΑ σύστημα τότε η έξοδος ενός IIR φίλτρου θα είναι η συνέλιξη της εισόδου με την κρουστική απόκριση δηλαδή

$y(n) = \sum_{k=0}^{\infty} h(k) x(n-k)$ ωστόσο δεν μπορούμε να υλοποιήσουμε κάτι που τείνει στο άπειρο οπότε η έξοδος ενός IIR φίλτρου είναι

$$y(n) \sum_{k=0}^{\infty} h(k) x(n-k) = \sum_{k=0}^M b_k x(n-k) - \sum_{k=1}^N a_k y(n-k) \quad (2.5)$$

(ΑΘΑΝΑΣΙΟΣ ΣΚΟΔΡΑΣ & ΒΑΣΙΛΕΙΟΣ ΑΝΑΣΤΑΣΟΠΟΥΛΟΣ 2003)

Εάν σε αυτήν την εξίσωση διαφορών $a_k \neq 0$ για $k=0,1,2$ (δηλαδή τα a_k (οι ρίζες του παρονομαστή ,οι πόλοι της συνάρτησης μεταφοράς($H(z)$), δεν αντιστοιχούν σε μηδενικές τιμές) έτσι λοιπόν η κρουστική απόκριση ενός IIR φίλτρου θα τείνει στο άπειρο δηλαδή θα είναι $h(n)=0$ για $n < 0$ (αιτιατό) άρα 1 από 0 έως ∞ δηλαδή θα είναι όπως η $u(n)$ οπότε η κρουστική απόκριση θα είναι άπειρης διάρκειας και για αυτό τον λόγο ονομάζετε φίλτρο άπειρης κρουστικής απόκρισης δηλαδή "θεωρητικά" το σήμα δεν θα έχει απόσβεση στο 0 πάντα θα έχει μη μηδενικές τιμές

Επειδή υπάρχει συνεχής ανατροφοδότηση δεν έχουν γραμμική φάση αυτό σημαίνει όχι σταθερή καθυστέρηση ομάδας σε όλες τις συχνότητες.

Μπορεί να είναι ασταθή φίλτρα εφόσον έχουν πόλους.

(ΑΘΑΝΑΣΙΟΣ ΣΚΟΔΡΑΣ & ΒΑΣΙΛΕΙΟΣ ΑΝΑΣΤΑΣΟΠΟΥΛΟΣ 2003)

2.1 Φίλτρα άπειρης κρουστικής απόκρισης(IIR)

Τα ψηφιακά IIR φίλτρα προέρχονται από τα αναλογικά φίλτρα. Από την θεωρία γνωρίζουμε ότι η συνάρτηση μεταφοράς ενός αναλογικού φίλτρου είναι:

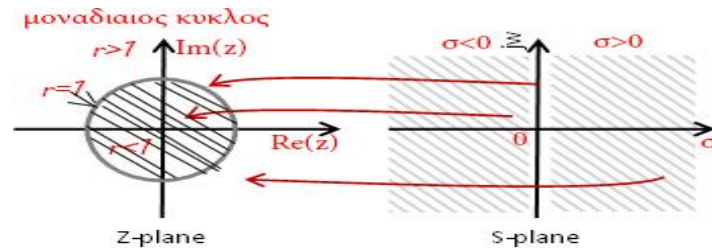
$$H(s) = \frac{B_s - \sum_{k=0}^M b_k s^k}{A_s - \sum_{k=0}^N a_k s^k} \quad (3.1.1) \text{ όπου } b_k \text{ και } a_k \text{ είναι οι συντελεστές του φίλτρου}$$

Εναλλακτικά μπορούμε να πάρουμε την συνάρτηση μεταφοράς του αναλογικού φίλτρου με τον μετασχηματισμό Laplace της κρουστικής απόκρισης συνεχούς χρόνου $h(t)$

$$H(s) = \int_{-\infty}^{\infty} h(t) e^{-st} dt \quad (2.1.2)$$

1) Γνωρίζουμε επίσης από την θεωρία ότι για να είναι ένα αναλογικό φίλτρο ευσταθές θα πρέπει οι πόλοι του να βρίσκονται στην αριστερή πλευρά του πεδίου s δηλαδή αν $\sigma < 0$ τότε $r < 1$. Δηλαδή οι πόλοι του αναλογικού φίλτρου θα βρίσκονται μέσα στον μοναδιαίο κύκλο οπότε ένα ευσταθές αναλογικό φίλτρο θα μας δώσει ένα ευσταθές ψηφιακό φίλτρο

2) Και η απεικόνιση του φανταστικού άξονα $j\omega$ του s επιπέδου θα πρέπει να γίνεται πάνω ακριβώς στο μοναδιαίο κύκλο ένα προς ένα έτσι ώστε να διατηρούνται τα χαρακτηριστικά της απόκρισης συχνότητας και στα δυο πεδία (s και z)



Εικόνα 2.1 Μεταφορά από το splane στο zplane(μοναδιαιος κύκλος)

2.2 Αναλογικά IIR φίλτρα

Υπάρχουν τέσσερα είδη IIR φίλτρων αυτά είναι: Butterworth, chybyshev 1, chybyshev2, elliptic

2.2.1 Butterworth

Σε ένα φίλτρο Butterworth η απόκριση του χαρακτηρίζετε 'maximally flat'. Αυτό έχει να κάνει με το ότι η απόκριση πλάτους του (amplitude response) είναι επίπεδη (flat) στην συχνότητα 0. Κυριο χαρακτηριστικό του είναι ότι είναι ένα all pole φίλτρο δηλαδή έχει μόνο πόλους. Αυτό έχει να κάνει με το ότι η συνάρτηση μεταφοράς $H_a(s)$ έχει μηδενικά τα οποία τείνουν στο άπειρο και πόλους που βρίσκονται στην αριστερή πλευρά του s (domain) οπότε στο εσωτερικό του μοναδιαιου κύκλου Ένα βασικό χαρακτηριστικό είναι ότι δεν παρουσιάζει κυμάτωση στην ζώνη διέλευσης η στην ζώνη αποκοπής. Για να είναι ευσταθές ένα Butterworth φίλτρο θα πρέπει όπως προ είπα οι πόλοι να βρίσκονται στην αριστερή πλευρά του s (πεδίου)

Το τετράγωνο του πλάτους της απόκρισης συχνότητας είναι

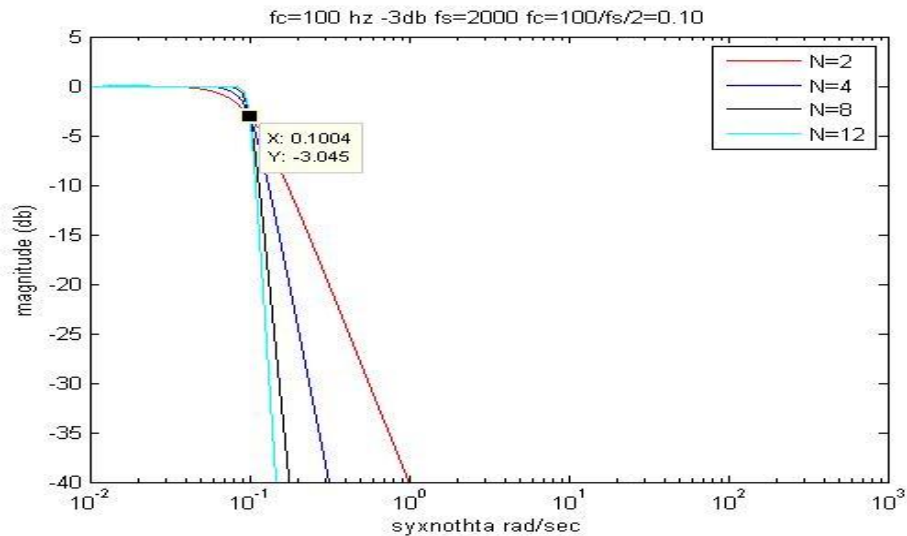
$$[H_a(j\Omega)]^2 = \frac{1}{1 + (\frac{j\Omega}{\Omega_c})^{2N}} \quad (2.2.1.1)$$

Όπου N η τάξη του φίλτρου, Ω_c είναι η συχνότητα αποκοπής των 3db. σε λογαριθμική κλίμακα δηλαδή το κέρδος στην ζώνη αποκοπής είναι 3db. λιγότερο από ότι στην ζώνη διέλευσης ή $2/\sqrt{2}=0,707$ φορές λιγότερο αντίστοιχα εάν μιλάμε για voltage κέρδος όταν $H_a(\Omega=0)=1$ και $H_a(\Omega=\Omega_c)=1/\sqrt{2}=0,707$. Επίσης όσο αυξάνετε η τάξη του φίλτρου δηλαδή ο (αριθμός των πόλων του φίλτρου) τόσο πιο στενή γίνετε η ζώνη μετάβασης, τόσο πιο γρηγορά γίνετε η μετάβαση από την ζώνη διέλευσης στην ζώνη αποκοπής, έτσι απόκριση ενός χαμηλοπερατού Butterworth θα πλησιάζει με αυτήν του ιδανικού χαμηλοπερατού περισσότερο ωστόσο αυτό δεν το θέλουμε γιατί όσο αυξάνει η τάξη του φίλτρου άρα οι συντελεστές του φίλτρου και έτσι χρειαζόμαστε περισσότερη μνήμη για την αποθήκευση των συντελεστών=μεγαλύτερο κόστος. Η τάξη του φίλτρου βρίσκετε ως εξής

$$N \geq \frac{\log\left(\frac{A_s}{A_p}\right)}{2 \log\left(\frac{\omega_s P}{\omega_p P}\right)} \quad (2.2.1.2)$$

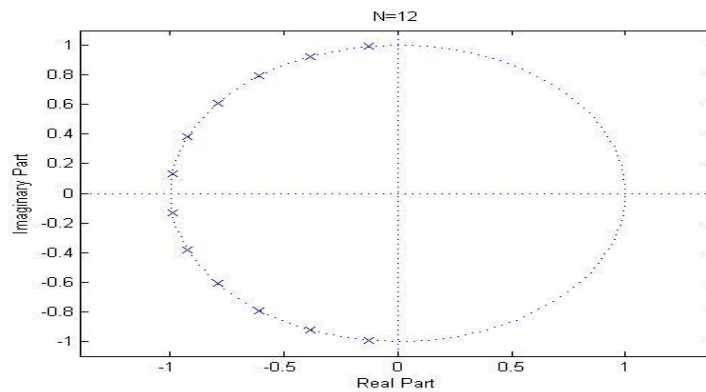
όπου ω_s είναι συχνότητα αποκοπής και A_p είναι η κυμάτωση στην ζώνη διέλευσης και A_s η απόσβεση στην ζώνη αποκοπής σε db.

(Γεώργιος Μουστακίδης 2003)



Εικόνα 2.2 αποκρίσεις συχνότητας IIR butter

Επίσης ένα αναλογικό φίλτρο Butterworth έχει μόνο πόλους που οι θέσεις τους θα είναι πάνω και γύρω από έναν κύκλο ακτίνας Ω_c και σημεία που θα ισαπέχουν το ένα με το άλλο $[Ha(j\Omega)]^2 = Ha(s)Ha(-s)$ για $s=j\Omega$ από εδώ καταλαβαίνουμε ότι $Ha(s)$ αντιστοιχούν στους πόλους της αριστερής πλευράς και $Ha(-s)$ αντιστοιχούν στους πόλους της δεξιάς πλευράς. Οι πόλοι της αριστερής πλευράς είναι $Ha(s)$ είναι και οι "ευσταθείς πόλοι" του φίλτρου και οι δεξιοί είναι "ασταθείς πόλοι" που αντιστοιχούν στους αντίθετους πόλους της αριστερής πλευράς του φίλτρου. Εμείς κρατάμε τους αριστερούς πόλους γιατί ένα αναλογικό φίλτρο για να είναι ευσταθές θα πρέπει οι πόλοι του να είναι στην αριστερή πλευρά του s πεδίου (> 0 τιμες)



Εικόνα 2.3 Πόλοι αναλογικού φίλτρου Butterworth τάξη(N=12)

2.2.2 Chebyshev I

Το φίλτρο ChybshevI είναι και αυτό όπως το Butterworth ένα (all pole) φίλτρο δηλαδή έχει μόνο πόλους, έχουν ομοιόμορφη (ισοκυματική = ίδια μέγιστα κυμάτωσης) κυμάτωση στην ζώνη διέλευσης ενώ δεν έχουν κυμάτωση στην ζώνη αποκοπής

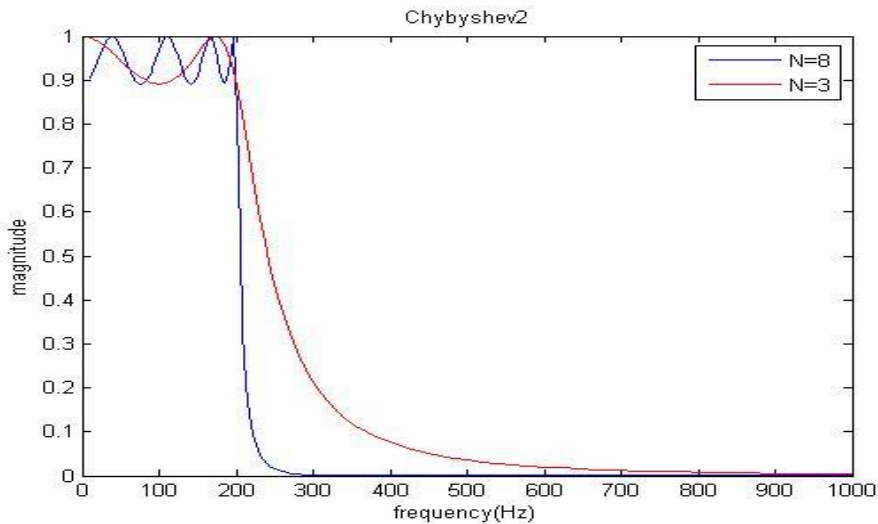
Το πλάτος της απόκρισής συχνότητας είναι

$$|Ha(j\Omega)|^2 = \frac{1}{1 + \varepsilon^2 T_N^2(\frac{\Omega}{\Omega_p})} \quad (2.2.2.1)$$

όπου N είναι η τάξη του φίλτρου, ε^2 είναι μια παράμετρος του φίλτρου που σχετίζεται με το πλάτος της κυμάτωσης στην ζώνη διέλευσης και Ω_p είναι η συχνότητα αποκοπής στο όριο της ζώνης διέλευσης. $T_N(x)$ είναι το πολυώνυμο Chebyshev τάξης N

$$T_N(x) = \begin{cases} \cos(N \cos^{-1} x) & |x| \leq 1 \\ \cosh(N \cosh^{-1} x) & |x| > 1 \end{cases} \quad (2.2.2.2)$$

Όσο αυξάνει η τάξη του φίλτρου αυξάνεται και αριθμός των ταλαντώσεων(με σταθερό ϵ) στην ζώνη διέλευσης ενώ το εύρος της ζώνης μετάβασης μεταξύ της ζώνης διέλευσης και της ζώνης αποκοπής μικραίνει. Την τάξη του φίλτρου μπορούμε να την καταλάβουμε κοιτώντας την κυμάτωση όταν $N=3$ τότε θα έχουμε 3 ημικύκλια κυμάτωσης στην ζώνη διέλευσης η $N=8$ αντίστοιχα (Γεώργιος Μουστακίδης 2003)

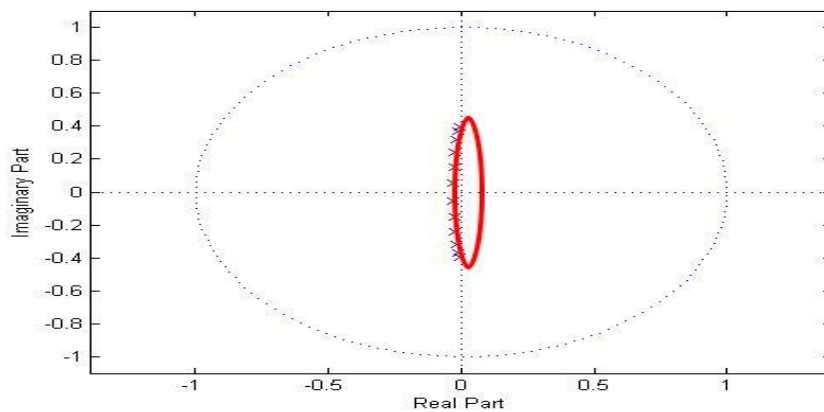


Εικόνα 2.4 Αποκρίσεις συχνότητας Chebyshev1

Επίσης στο σχήμα καταλαβαίνουμε ότι εάν το N =περιττός δηλαδή $N=3$ αριθμός τότε τα πολυώνυμο $T_N(0)=0$ και έτσι $|\mathbf{H}(0)|^2=1$.Εάν το N =αρτιος δηλαδή $N=8$ τότε $T_N(0)=1$ και έτσι $|\mathbf{H}(0)|^2=1/\epsilon^2$.Επίσης βλέπουμε ότι το φίλτρο έχει κυμάτωση από 0 έως $\Omega_c(200\text{Hz})$ ή για τιμές $-\Omega_c$ δηλαδή σύμφωνα με τις ιδιότητες του πολυωνύμου Chebyshev το T_{N^2} θα είναι $x \leq 1$ Για $x \geq 1$ δηλαδή για $\Omega > \Omega_c$ η απόκριση πλάτους βλέπουμε ότι φθίνει μονοτονικά προς το 0,οπότε η ζώνη διάβασης είναι όσο είναι και η κυμάτωση. Επίσης εάν $\Omega = \Omega_c$ τότε η απόκριση πλάτους

$$|\mathbf{H}_a(j\Omega)|^2 = \frac{1}{\sqrt{1+\epsilon^2}} \quad (2.2.2.3)$$

που εκεί θα έχει η κυμάτωση το μικρότερο πλάτος. Οι πόλοι του φίλτρου Chebyshev και αυτοί βρίσκονται στο αριστερό μιγαδικό επίπεδο για να είναι ευσταθές το φίλτρο σε ένα σχήμα έλλειψης γύρω από φανταστικό άξονα ($J\Omega$) (Γεώργιος Μουστακίδης 2003)



Εικόνα 2.5 Θέσεις των πόλων Chebyshev1

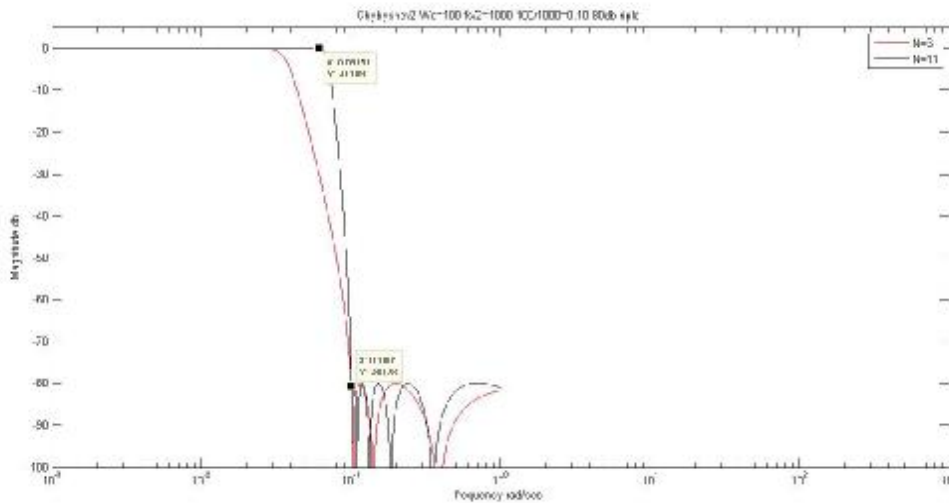
2.2.3 Chebyshev II

Τα ChebyshevII έχουν και μηδενικά και πόλους και παρουσιάζουν κυμάτωση στην ζώνη αποκοπής $[\Omega_c, \infty]$ οπότε για $\Omega > \Omega_c$ η αποκρίση πλάτους θα έχει κυματισμούς στο διαστημα $[0, \frac{1}{\sqrt{1+\varepsilon^2}}]$ ενώ φθίνουν μονοτονικά για τιμές του Ω στην ζώνη διέλευσης $[0, \Omega_c]$ $0 \leq \Omega \leq \Omega_c$ όπου $(\Omega_c = \Omega_s)$ Τα μηδενικά βρίσκονται πάνω στο φανταστικό άξονα $j\omega$ οπότε όλα τα μηδενικά του φίλτρου είναι φανταστικοί αριθμοί. Όταν η τάξη του φίλτρου είναι περιττός τότε στην συνάρτηση μεταφοράς τα μηδενικά θα είναι $z=p-1$ δηλαδή κατά 1 λιγότερα από τους πόλους αυτό γίνεται γιατί το κεντρικό μηδέν απειρίζεται και αποκλείεται. Εδώ τα μηδενικά έχουν τους συζυγείς τους όπως και οι πόλοι

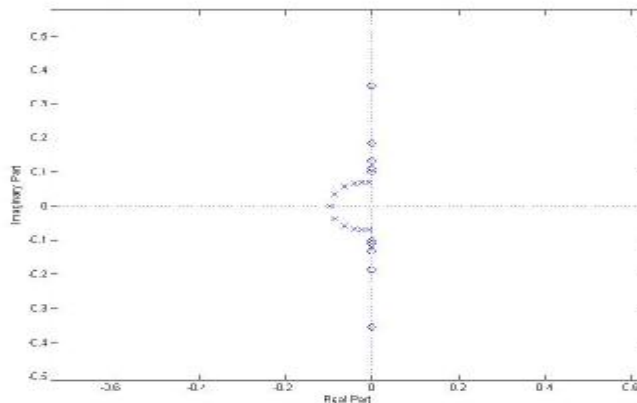
Το πλάτος της απόκρισής συχνότητάς είναι

$$|Ha(j\Omega)|^2 = \frac{1}{1 + \varepsilon^2 [T_N(\frac{\Omega_s}{\Omega p}) / T_N(\frac{\Omega_s}{\Omega})]^2} \quad (\text{Γεώργιος Μουστακίδης 2003}) \quad (2.2.3.1)$$

Όπου N είναι η τάξη του φίλτρου Ω_p είναι η συχνότητα αποκοπής στην ζώνη διέλευσης Ω_s είναι η συχνότητα αποκοπής στην ζώνη αποκοπής και ε είναι η παράμετρος που ελέγχει την κυμάτωση στην ζώνη αποκοπής. Και εδώ ισχύει ότι όσο αυξάνετε η τάξη του φίλτρου τόσο αυξάνεται ο αριθμός των κυματώσεων και η ζώνη μετάβασης μικραίνει



Εικόνα 2.6 Αποκρίσεις συχνότητάς Chybyshev2



Εικόνα 2.7 Μηδενικά και πόλοι Chybyshev2

2.2.4 Ελλειπτικά φίλτρα

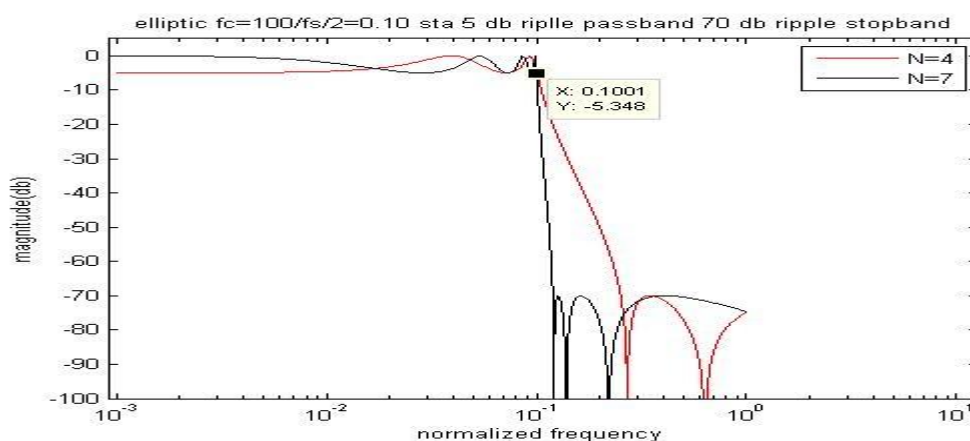
Τα Ελλειπτικά φίλτρα παρουσιάζουν κυμάτωση και στην ζώνη διέλευσης και στην ζώνη αποκοπής Η συναρτηση μεταφοράς του φίλτρου έχει και μηδενικά και πολλούς

Το πλάτος της αποκρίσης συχνοτητας είναι

$$|H\alpha(\Omega)|^2 = \frac{1}{1 + \varepsilon^2 U_{N^2}\left(\frac{\Omega}{\Omega_p}\right)} \quad (2.2.4.1)$$

Όπου ε είναι η παράμετρος που σχετίζεται με την κυμάτωση στην ζώνη διέλευσης, Ω_p είναι η συχνότητα αποκοπής στην ζώνη διέλευσης και U_{N^2} είναι η Ιακωβιανή ελλειπτική συναρτηση που έχει την μορφή κλασματος $U_N\left(\frac{1}{\Omega}\right) = \frac{1}{U_N(\Omega)}$ (Γεώργιος Μουστακίδης 2003) (3.2.4.2)

και είναι αυτή που δίνει τα χαρακτηριστικά της ισοκυμάτωσης και στην ζώνη διέλευσης και στην ζώνη αποκοπής Από το σχήμα μπορούμε να καταλάβουμε ότι το ελλειπτικό φίλτρο είναι ουσιαστικά η διασταύρωση η ένωση ενός Chebyshev1 (στη ζώνη διέλευσης) και Chybyshen2 (στη ζώνη αποκοπής)

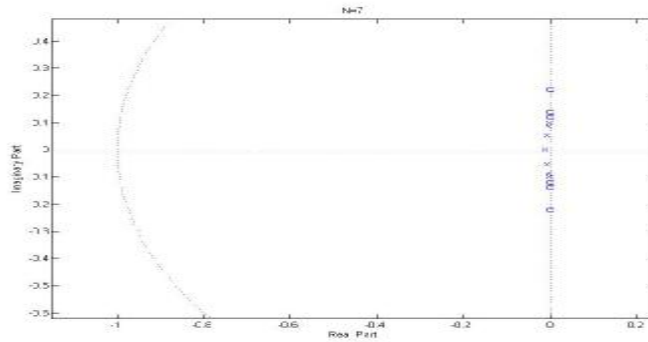


Εικόνα 2.8 Αποκρίσεις συχνοτητας ελλειπτικού

Από το σχήμα καταλαβαίνουμε ότι όπως στο φίλτρο Chebyshev1 στην συχνότητα 0 το πλάτος της το πλάτος της συνάρτησης μεταφοράς είναι μηδέν μόνο όταν N =περιττός αλλιώς N =άρτιος τότε

$$|H(0)|^2 = \frac{1}{1 + \varepsilon^2} \quad (2.2.4.3)$$

Από την φύση του (από το σχεδιασμό του, και για αυτό λεμέ ότι είναι πολύ πιο δύσκολο στο σχεδιασμό σε σχέση με τα άλλα φίλτρα) το ελλειπτικό φίλτρο λόγω του ότι έχει μια πιο απότομη απόκριση στην ζώνη μετάβασης ($\Omega_p < \Omega < \Omega_s$) Έτσι μπορούμε να καλύψουμε καλύτερα για συγκεκριμένες προδιαγραφές του φίλτρου Αυτό που θα πρέπει να πούμε είναι ότι όταν ένα ελλειπτικό φίλτρο στην συνάρτησης μεταφοράς του έχει ένα μηδενικό λιγότερο από τους πόλους του (γιατί το ένα μηδενικό απειρίζεται) δηλαδή όταν N =περιττός τότε δεν θεωρείται βέλτιστο (Γεώργιος Μουστακίδης 2003)



Εικόνα 2.9 Μηδενικά και πόλοι ελλειπτικού φίλτρου

2.3 Σύγκριση αναλογικών ΠΡ φίλτρων

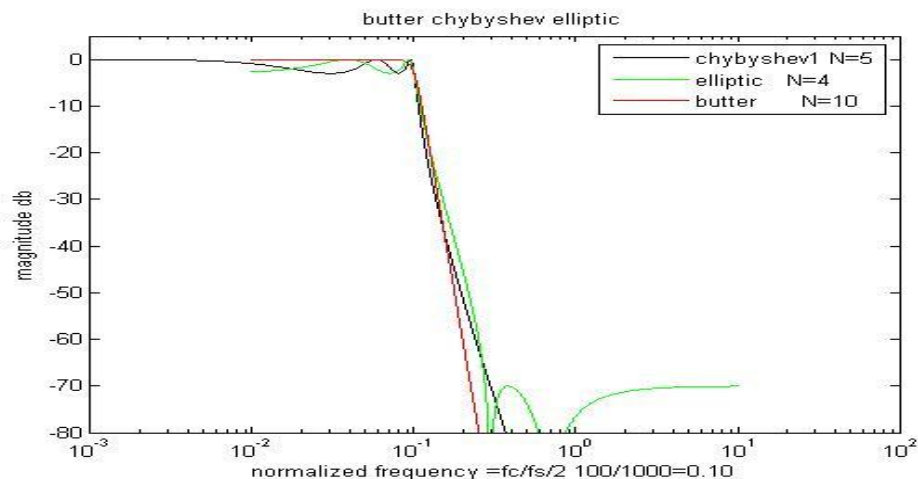
Το Butterworth φίλτρο θα λέγαμε ότι είναι εύκολο να υλοποιηθεί (χρειάζεται λιγότερες απαιτήσεις). Ωστόσο έχει αργή μετάβαση από την ζώνη διέλευσης στην ζώνη αποκοπής δηλαδή ο ρυθμός αποκοπής ενός σήματος σαν είσοδος δεν είναι και αυτός που θέλουμε έτσι ώστε να σβήσει το εισερχόμενο σήμα πιο γρήγορα, κάτι που το θέλουμε. Αναγκαστικά για να πέτυχουμε μικρότερη ζώνη μετάβασης (έτσι θα πλησιάζει στο ιδανικό φίλτρο) θα πρέπει να αυξήσουμε την τάξη του κάτι που σημαίνει (μεγαλύτερη πολυπλοκότητα) περισσότεροι συντελεστές περισσότερη μνήμη άρα μεγαλύτερο κόστος. Τα φίλτρα αυτά είναι χρήσιμα σε εφαρμογές στην επεξεργασία ήχου

Τα Chebyshev1 σε σχέση με τα Butterworth έχουν μεγαλύτερη κλίση στην ζώνη μετάβασης σε σχέση με ένα Butterworth (ίδιες τάξεις πάντα) Αυτό γίνεται γιατί επιτρέπει κυμάτωση στην ζώνη διέλευσης. Άρα για συγκεκριμένες προδιαγραφές και στα δυο φίλτρα το Chebyshev δεν θα είναι ποτέ μεγαλύτερης τάξης από το Butterworth (μικρότερη πολυπλοκότητα) έτσι κάνουμε οικονομία στα στοιχεία του φίλτρου οπότε μικρότερο κόστος κάτι που είναι πολύ σημαντικό στο σχεδιασμό και πιο ευκολά υλοποιήσιμο. Αυτά δεν χρησιμοποιούνται σε ακουστικές εφαρμογές γιατί έχουν κυμάτωση στην ζώνη διέλευσης

Τα Chebyshev2 και αυτά σε σχέση με τα Butterworth παρουσιάζουν και αυτά μεγαλύτερη κλίση στην ζώνη μετάβασης από ότι ένα Butterworth η ένα Chebyshev1. Αυτά έχουν και πόλους και μηδενικά, που είναι πάνω στον φανταστικό άξονα, οπότε είναι πιο δύσκολο να υλοποιηθούν. Σε σχέση με τα Chebyshev1 παρουσιάζουν κυμάτωση στην ζώνη αποκοπής και μπορούμε να ορίσουμε εμείς στα πόσα db.

Τα Ελλειπτικά φίλτρα είναι αυτά τα οποία έχοντας να καλύψουμε συγκεκριμένες προδιαγραφές μπορούμε να τις προσεγγίσουμε καλύτερα με μικρότερη τάξη άρα λιγότεροι συντελεστές (μικρότερη πολυπλοκότητα) σε σχέση με τα άλλα φίλτρα, μικρότερο κόστος. Παρουσιάζουν κυμάτωση και στην ζώνη διέλευσης και στην ζώνη αποκοπής (συνδυασμός Chebyshev1 και Chebyshev2)

Μπορούμε να πέτυχουμε μεγαλύτερη κλίση στην ζώνη μετάβασης σε σχέση με τα άλλα τρία φίλτρα, δηλαδή η απόκριση στην ζώνη μετάβασης είναι πολύ πιο απότομη. Ωστόσο είναι πολύ πιο δύσκολη μέθοδος (υπολογιστικά) Προτιμάτε σαν φίλτρο όταν θέλουμε η ζώνη μετάβασης να είναι όσο το δυνατόν πιο απότομη (John Proakis Dimitris G Manolakis 2007)



Εικόνα 2.10 Σύγκριση Butter,Chebyshev I,ελλειπτικό

Από το σχήμα καταλαβαίνουμε ότι για να ικανοποιήσουν τις ίδιες προδιαγραφές τα φίλτρα(Butter ,Chebyshev,elliptic) πχ αν δούμε στα 50 db. (που παρουσιάζουν μηδαμινή διαφορά) θα πρέπει το butter να έχει σχεδόν διπλάσια τάξη(N=10) από το ChebyshevI Το ελλειπτικό φίλτρο μπορεί να καλύψει τις ίδιες προδιαγραφές με τα άλλα με μια τάξη λιγότερο σε σχέση με το Chebyshev και 6 λιγότερες από το butter.Το ChebyshevI μπορεί να καλύψει τις ίδιες προδιαγραφές 5 τάξεις μικρότερο σε σχέση με το butter.

Το Ελλειπτικό έχει σχεδόν την ίδια ζώνη μετάβασης με τα άλλα φίλτρα αλλά συγχρόνως με μικρότερη τάξη σε σχέση με τα άλλα (λιγότεροι συντελεστές) δηλαδή έχει πιο στενή ζώνη μετάβασης και μικρότερη τάξη άρα και μικρότερο κόστος κατασκευής του φίλτρου κάτι που στον σχεδιασμό παίζει μεγάλο ρολό Από αυτήν την πλευρά το ελλειπτικό θεωρείται καλύτερο σε σχέση με τα άλλα.

Πάντως η επιλογή του φίλτρου εξαρτάται από την εφαρμογή στην οποία θα χρησιμοποιηθούν και πολλές φορές εξαρτάται και υποκειμενικά κυρίως όταν πρόκειται για ήχο για το πόσο καλά μας ακούγετε σε εμάς τους ίδιους ωστόσο τα butter θεωρούνται ικανοποιητικά για επεξεργασία ήχου γιατί θεωρούνται λιγότερο πολύπλοκα σε σχέση με τα άλλα (για να τα προσδιορίσουμε χρειάζεται να ξέρουμε την τάξη τους N και την συχνότητα αποκοπής f_c)

2.4 Μετατροπή αναλογικών IIR σε ψηφιακά

Υπάρχουν διαφορετικές μέθοδοι μετατροπής ενός αναλογικού φίλτρου(IIR) σε ψηφιακό φίλτρο. Οι δυο πιο γνωστές μέθοδοι μετατροπής είναι α)η μέθοδος της αμετάβλητης κρουστικής απόκρισης και β)η μέθοδος του διγραμμικού μετασχηματισμού.

2.4.1 Μέθοδος Αμετάβλητης κρουστικής

Σύμφωνα με αυτήν την μέθοδο μπορούμε να σχεδιάσουμε ένα ψηφιακό IIR φίλτρο όπου η κρουστική του απόκριση να είναι ίση με δειγματολημμένη κρουστική απόκριση ενός αναλογικού φίλτρου $h(n)=h(nT)=ht$

$$(2.4.1.1)$$

όπου T η περίοδος δειγματοληψίας .Η συνάρτηση μεταφοράς του ψηφιακού φίλτρου είναι ίση με με τον μετασχηματισμό z της δειγματολημμένης κρουστικής απόκρισης

$$(z)=Z\{h(nT)\} \quad (2.4.1.2)$$

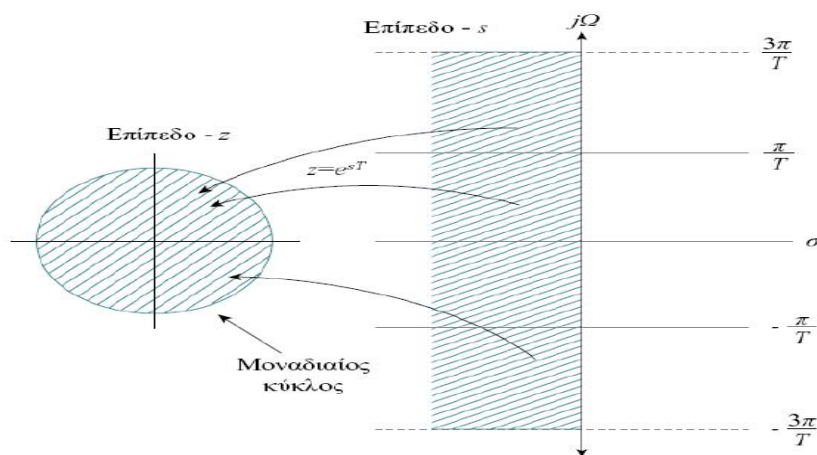
)

και H(s) είναι ο μετασχηματισμός Laplace της κρουστικής απόκρισης h(t) Εφόσον $f_s=1/T$ δείγματα ανά δευτερόλεπτα

Αν ο ρυθμός δειγματοληψίας είναι μικρότερος από το διπλάσιο της μέγιστης συχνότητάς του αναλογικού φίλτρου ($f_s < 2f_{max}$) τότε έχουμε το φαινόμενο (aliasing). Οπότε καταλαβαίνουμε ότι το T θα πρέπει να είναι αρκετά μικρό για να μην έχουμε το aliasing. Έτσι η μέθοδος αυτή δεν κάνει για τον σχεδιασμό high pass φίλτρων αλλά μόνο για bandpass και low pass. Η απεικόνιση των σημείων από το επίπεδο s στο επίπεδο z με την αμετάβλητη κρουστική απόκριση γίνεται ως εξής $z = e^{sT}$ και η ψηφιακή γωνιακή συχνότητα θα είναι $\omega = \Omega T$

Για $s = \sigma + j\Omega$ και $z = re^{j\omega}$ (2.4.1.3)
δηλαδή $(z = \text{Re}(z) + j\text{Im}(z))$

Αυτό που παρατηρούμε είναι ότι εάν για $\sigma < 0$ έχουμε $0 < r < 1$, ενώ για $\sigma > 0$ έχουμε $r > 1$. Όταν $\sigma = 0$, τότε $r = 1$. Αυτό σημαίνει ότι το αριστερό ημιεπίπεδο του s απεικονίζεται στο εσωτερικό του μοναδιαίου κύκλου στο z , ενώ το δεξί ημιεπίπεδο του s απεικονίζεται στο εξωτερικό του μοναδιαίου κύκλου στο z . Επίσης, ο άξονας $j\Omega$ απεικονίζεται πάνω στο μοναδιαίο κύκλο του επιπέδου z . Η απεικόνιση αυτή όμως δεν είναι απεικόνιση ένα προς ένα. Αφού το ω είναι μοναδικό στην περιοχή $(-\pi, \pi)$, η απεικόνιση $\omega = \Omega T$ συνεπάγεται ότι η περιοχή των συχνοτήτων $-\pi/T \leq \Omega \leq \pi/T$ απεικονίζεται στις αντίστοιχες τιμές $-\pi \leq \omega \leq \pi$. Επίσης και η περιοχή των συχνοτήτων $\pi/T \leq \Omega \leq 3\pi/T$ απεικονίζεται επίσης στην ίδια περιοχή $-\pi \leq \omega \leq \pi$. Εδώ φαίνεται και το φαινόμενο aliasing (Vinay K. Ingle, John G. Proakis 2007)

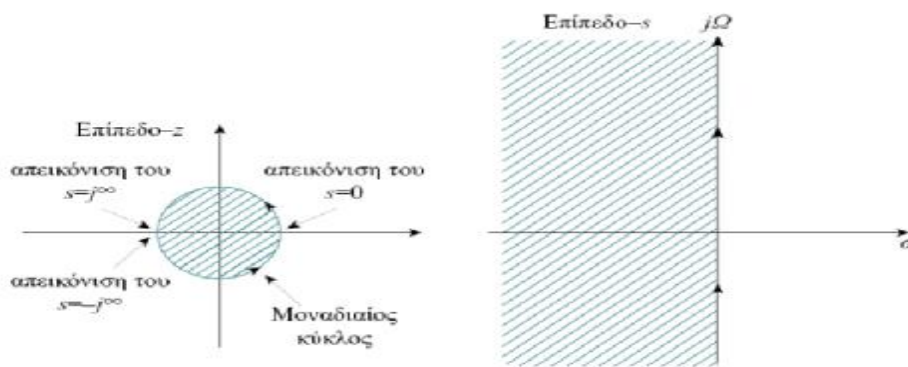


Εικόνα 2.11 Απεικόνιση των σημείων s plane ($\sigma < 0$) στο z plane μέθοδος αμετάβλητης κρουστικής απόκρισης. Ένα προς πολλά απεικόνιση

2.4.2 Διγραμμικός μετασχηματισμός

Πρόκειται για έναν μαθηματικό μετασχηματισμό του πεδίου s (αναλογικό) στο πεδίο z (ψηφιακό). Σύμφωνα με τον μετασχηματισμό αυτό μπορούμε να υλοποιήσουμε την συνάρτηση μεταφοράς $H(z)$ του ψηφιακού φίλτρου αντικαθιστώντας το s με $\frac{z-1}{z+1}$ δηλαδή $H(z) = H(s)$ όπου η απεικόνιση από το πεδίο s στο πεδίο z γίνεται σύμφωνα με την σχέση

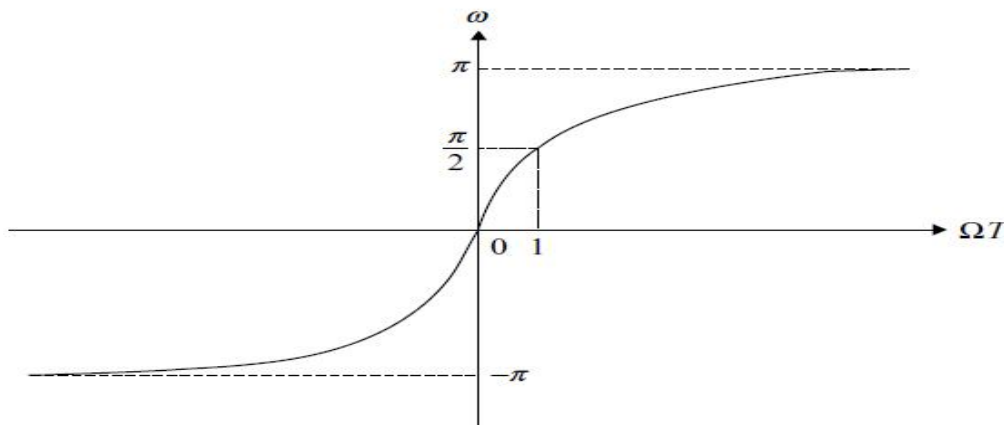
$$s = \frac{z-1}{z+1} \quad (2.4.2.1)$$



Εικόνα 2.12

Τα χαρακτηριστικά του διγραμμικού μετασχηματισμού είναι τα εξής αν $z=re^{j\omega}$ και $s = \sigma + j\Omega$ τότε και από το σχήμα καταλαβαίνουμε ότι

- 1) $\sigma < 0$ τότε $r < 1$ και $z < 1$ δηλαδή το αριστερό ημιπίπεδο του s (άρα ευσταθές αναλογικό φίλτρο) απεικονίζεται στο εσωτερικό του μοναδιαίου κύκλου στο επίπεδο- z οπότε ένα ευσταθές αναλογικό φίλτρο θα μας δώσει ένα ευσταθές ψηφιακό φίλτρο
- 2) $\sigma > 0$ τότε $r > 1$ και $z > 1$ δηλαδή ενώ το δεξί ημιπίπεδο του s απεικονίζεται στο εξωτερικό του μοναδιαίου κύκλου άρα ένα ασταθές αναλογικό φίλτρο (για να είναι ευσταθές ένα αναλογικό φίλτρο οι πόλοι του θα πρέπει να βρίσκονται στην αριστερή πλευρά του μοναδιαίου κύκλου)
- 3) εάν $\sigma = 0$ τότε $r = 1$ και $z = 1$ τότε ο φανταστικός άξονας $j\Omega$ του επιπέδου s θα πέσει πάνω στον κάθετο άξονα του μοναδιαίου κύκλου δηλαδή υπάρχει μια απευθείας σχέση της αναλογικής συχνότητάς Ω του s επιπέδου και της ψηφιακής συχνότητάς ω του z επιπέδου οπότε δεν υπάρχει το φαινόμενο aliasing γιατί είναι για μια απεικόνιση 1 προς 1.
(Vinay K. Ingle John G. Proakis 2007)



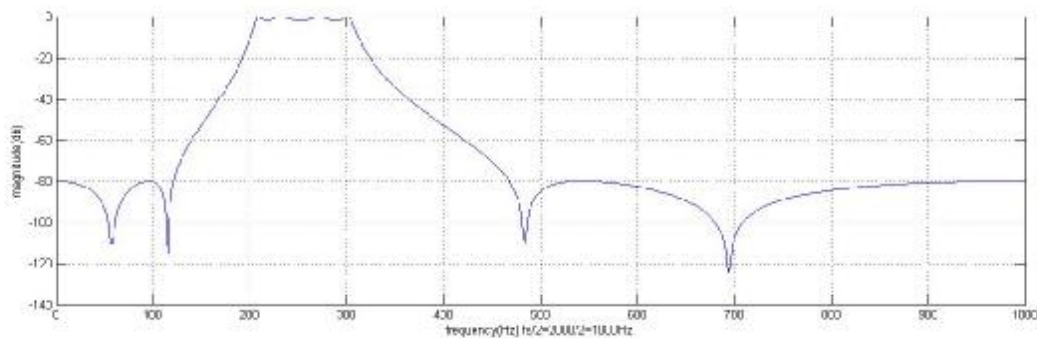
Εικόνα 2.13 Σχέση αναλογικής και ψηφιακής συχνότητάς διγραμμικού μετασχηματισμού

Στο παραπάνω σχήμα βλέπουμε την σχέση της αναλογικής συχνότητάς Ω και της ψηφιακής συχνότητάς ω όπου $\Omega = \tan(\frac{\omega}{2})$ και $\omega = 2 \tan^{-1}(\frac{\Omega T}{2})$. Εδώ βλέπουμε ότι οι συχνότητες Ω απεικονίζονται μια φορά μόνο στο διάστημα από $(-\pi$ έως $\pi)$ του ω (η αντίστοιχη ψηφιακή συχνότητα). Επίσης αυτό που βλέπουμε από το σχήμα είναι ότι η περιοχή $-\infty$ έως 0 για Ω η απεικόνιση τους θα γίνει από $\omega = -\pi$ έως $\omega = 0$ ($z = -1$ έως $z = 1$) και από 0 έως $+\infty$ για Ω θα γίνει από $\omega = 0$ έως π ($z = 1$ έως $z = -1$). Τέλος αυτό που βλέπουμε είναι ότι η απεικόνιση της αναλογικής συχνότητάς Ω στην αντίστοιχη ψηφιακή συχνότητα ω είναι μη γραμμική για συχνότητες $\Omega > 1$ οι οποίες συμπίπτουν στις

αντίστοιχες ψηφιακές συχνότητες στο διάστημα $\pi/2$ έως π . Αυτό λέγεται στρέβλωση της συχνότητάς(frequency warping).

Ουσιαστικά η στρέβλωση της συχνότητάς είναι προφανής εφόσον η αναλογική συχνότητα έχει εύρος $-\infty$ έως ∞ ενώ η ψηφιακή συχνότητα έχει πεπερασμένο εύρος ($-\pi$ έως π). Αυτό θα πρέπει να λαμβάνετε υπόψιν όταν σχεδιάζουμε ένα φίλτρο και αυτό που θα πρέπει να κάνουμε είναι αποστρέβλωση της συχνότητάς (prewarp frequency). Αυτό γίνεται με τον τύπο $\Omega_c = \frac{2}{T} \tan \frac{\omega_c T}{2}$ όπου Ω_c είναι η συχνότητα αποκοπής του αναλογικού φίλτρου και ω_c είναι η επιθυμητή συχνότητα αποκοπής ψηφιακού φίλτρου $\omega_c = 2\pi f$ και $T=1$

Οπότε αυτό που κάνουμε είναι να 1)βρούμε την κανονικοποιημένη συχνότητα αποκοπής ω_c 2)Να κάνουμε prewarp frequency 3)Διαλέγουμε την συνάρτηση μεταφοράς συγκεκριμένου φίλτρου πχ butter, Chybychev κλπ. 4)αντικαθιστούμε το s της $H_a(s)$ με $s = \frac{2}{T} \frac{Z-1}{Z+1}$



Εικόνα 2.14 IIR ελλειπτικό φίλτρο 8 τάξης ,κυμάτωση στην ζώνη διέλευσης 2 db. και -80 db. απόσβεση στις ζώνες αποκοπής ψηφιακό με την μέθοδο του διγραμικού μετασχηματισμού

2.5 Γενικά χαρακτηριστικά IIR φίλτρων

Τα IIR φίλτρα άπειρης κρουστικής απόκρισης(infinite impulse response) αυτό γιατί η συνάρτηση μεταφοράς έχει και μηδενικά και πόλους αυτό σημαίνει ότι η έξοδος εξαρτάται όχι μόνο από τις τρέχουσα και τις προηγούμενες τιμές της εισόδου αλλά και από προηγούμενες τιμές της εξόδου οπότε γίνεται μια συνεχής ανατροφοδότηση δηλαδή οι συντελεστές του φίλτρου είναι "άπειροι" τότε και η κρουστική απόκριση θα έχει διάρκεια άπειρη $h(n)=0 \ n < 0$ αιτιατο και θα είναι από το 0 έως ∞

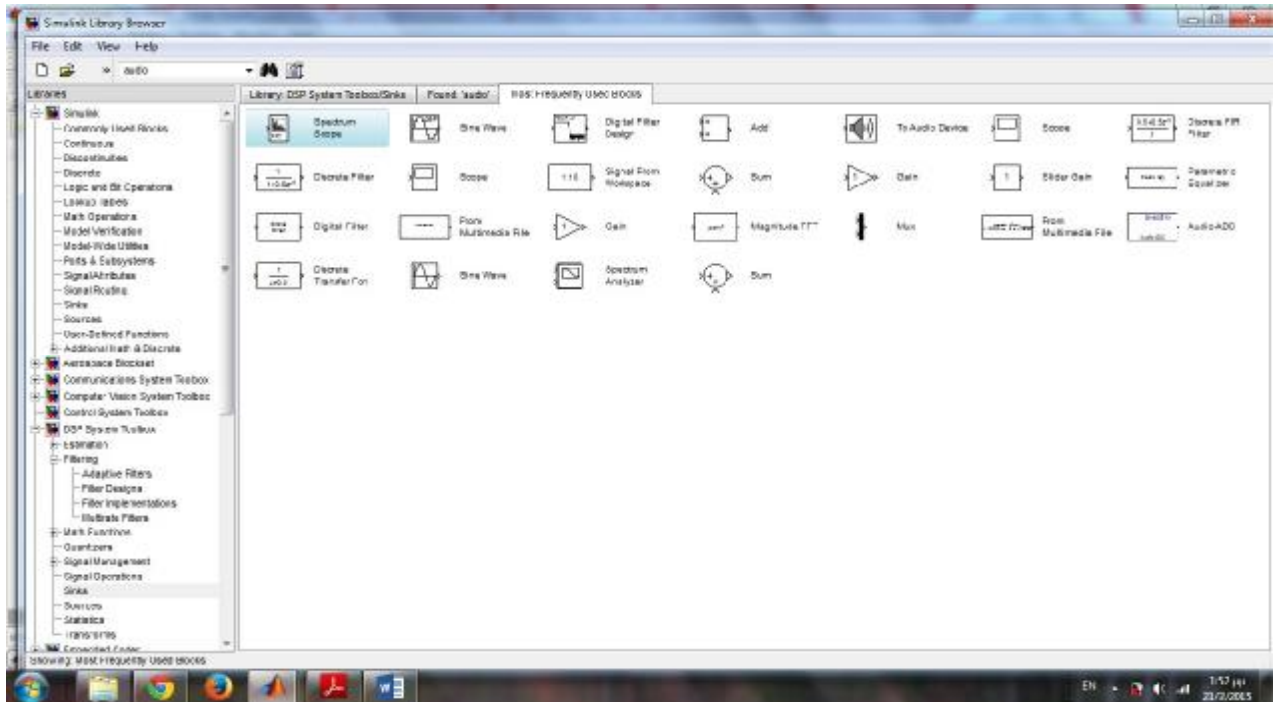
Τα IIR φίλτρα μπορεί να είναι ασταθή, εφόσον έχουν και πόλους όπου οι θέσεις των πόλων καθορίζει την ευστάθειά του φίλτρου ,αν αυτοί είναι έξω από μοναδιαίο κύκλο τότε το φίλτρο είναι ασταθές .Οπότε σημαντικό ρολό παίζει η επιλογή των καταλλήλων συντελεστών του φίλτρου

Τα IIR φίλτρα δεν έχουν γραμμική φάση γιατί υπάρχει μια συνεχής ανατροφοδότηση οπότε δεν έχουν και σταθερή καθυστέρηση ομάδας σε όλες τις συχνότητες οπότε υπάρχει παραμόρφωση στην μορφή του σήματος. Τα IIR φίλτρα μπορούν να πέτυχουν πολύ πιο καλύτερη απόκριση(εφαρμογή, προσέγγιση) με λιγότερους συντελεστές δηλαδή μικρότερης τάξης σε σχέση με τα FIR. Αυτό σημαίνει λιγότερες πράξεις για να υλοποιηθούν, μικρότερη πολυπλοκότητα λιγότερη μνήμη για την αποθήκευση των συντελεστών των φίλτρων .Δεν μας νοιάζει τόσο που δεν έχουν γραμμική φάση κυρίως όταν πρόκειται για χρήση σε εφαρμογές ήχου. Το αυτί δεν αντιλαμβάνεται τις διαφορές φάσης μεταξύ των συχνοτήτων

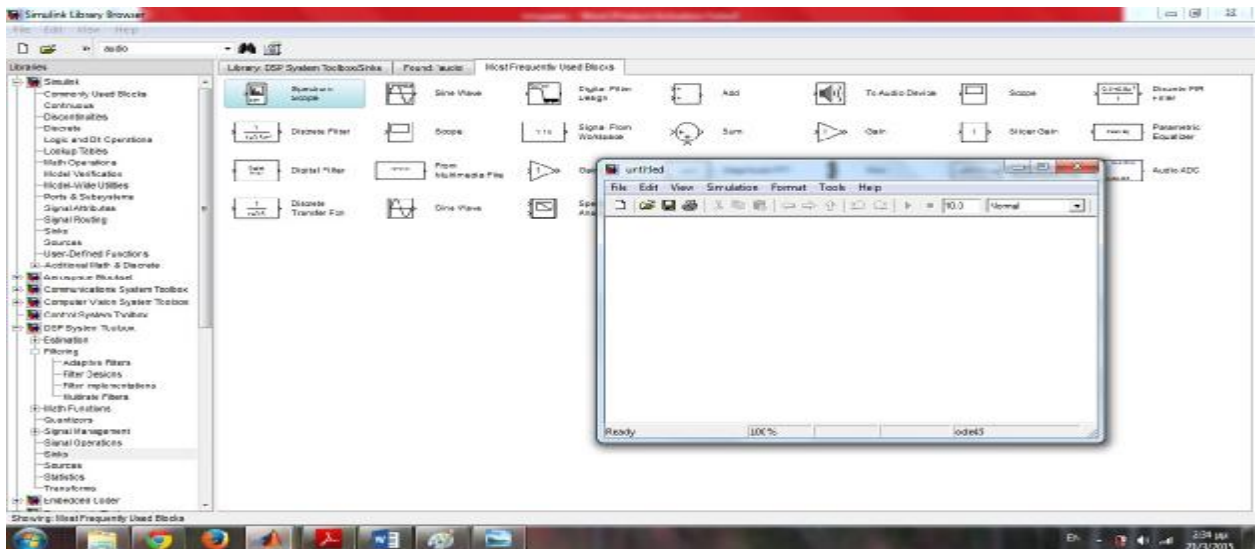
Οπότε το εάν χρησιμοποιήσουμε ένα IIR ή FIR εξαρτάται από την εφαρμογή .Αν θέλουμε την μορφή του σήματος αναλλοίωτη, (γραμμική φάση) τότε FIR πχ (επεξεργασία σήματος ,εικόνας ,βιολογικό σήμα) εάν η γραμμική φάση δεν παίζει ρολό και θέλουμε λιγότερους συντελεστές δηλαδή λιγότερες πράξεις άρα οικονομία στην χρήση υλικού άρα μικρότερο κόστος τότε IIR (equalizer) (Vinay K. Ingle,John G. Proakis 2007)

3 Μοντελοποίηση φίλτρων Simulink Matlab

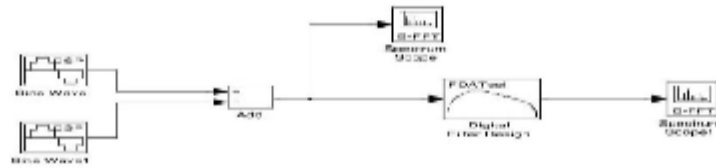
Το Simulink είναι ένα λογισμικό που το χρησιμοποιούμε για μοντελοποίηση, προσομοίωση την ανάλυση και την εφαρμογή πολλών συστημάτων. Το Simulink έχει έναν μεγάλο αριθμό βιβλιοθηκών οι οποίες περιέχουν στοιχεία των συστημάτων έτσι ώστε να μοντελοποιήσουμε ένα σύστημα και να κάνουμε την προσομοίωση τους. Αφού καθορίσουμε το αντίστοιχο μοντέλο μπορούμε μετά να προχωρήσουμε στην προσομοίωση



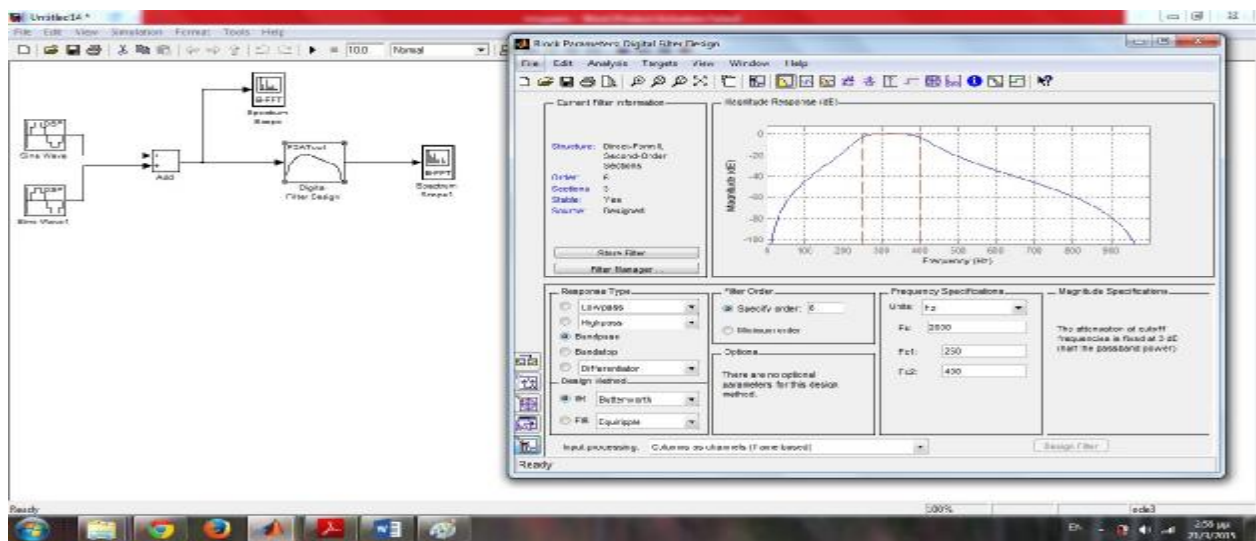
Εικόνα 3.1 Στοιχεία βιβλιοθήκης Simulink



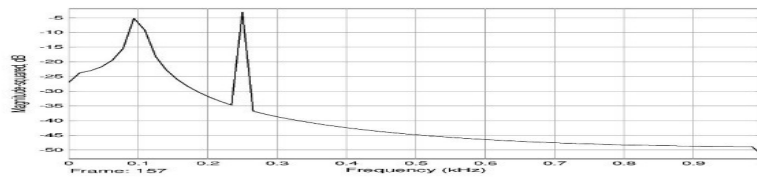
Ανοίγει ένα παράθυρο που μπορούμε να εισάγουμε τα διάφορα στοιχεία της βιβλιοθήκης και σχηματίζουμε το μοντέλο που θέλουμε. Μετά μπορούμε να πατήσουμε run και να τρέξει η προσομοίωση του μοντέλου



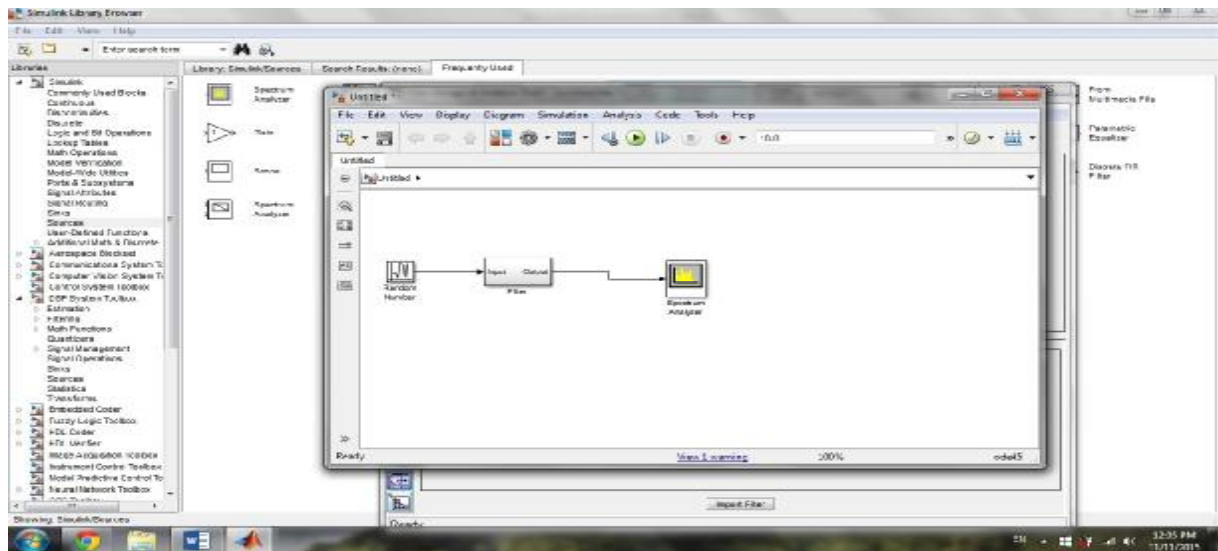
Εικόνα 3.2 Μοντέλο IIR φίλτρου Butterworth είσοδος στο φίλτρο δυο ημίτονα 100 και 120 Hz με πλάτος 1



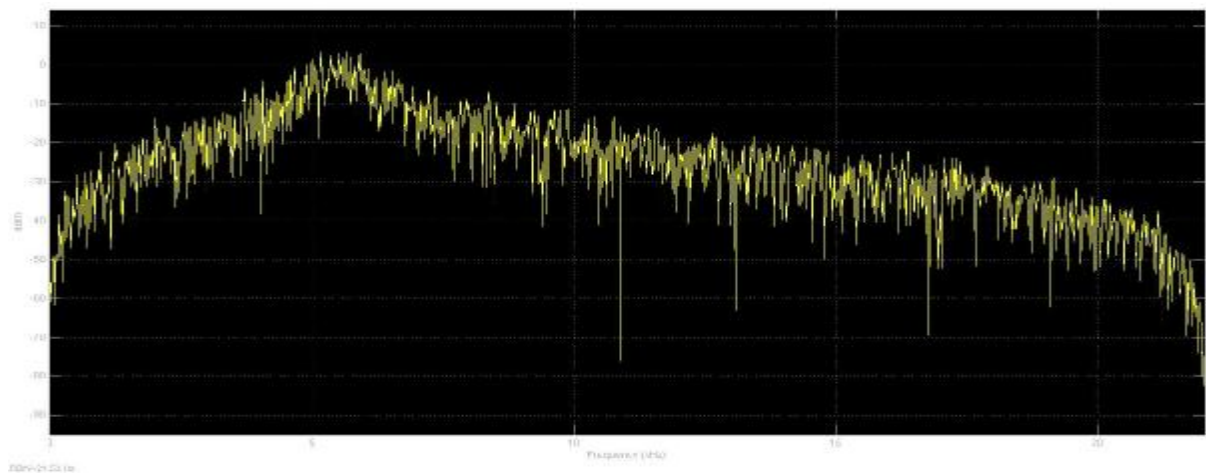
Εικόνα 3.3 IIR Butterworth bandpass $f_{c1}=250$ και $f_{c2}=400$ Hz 6ης τάξης $f_s=2000$ Hz



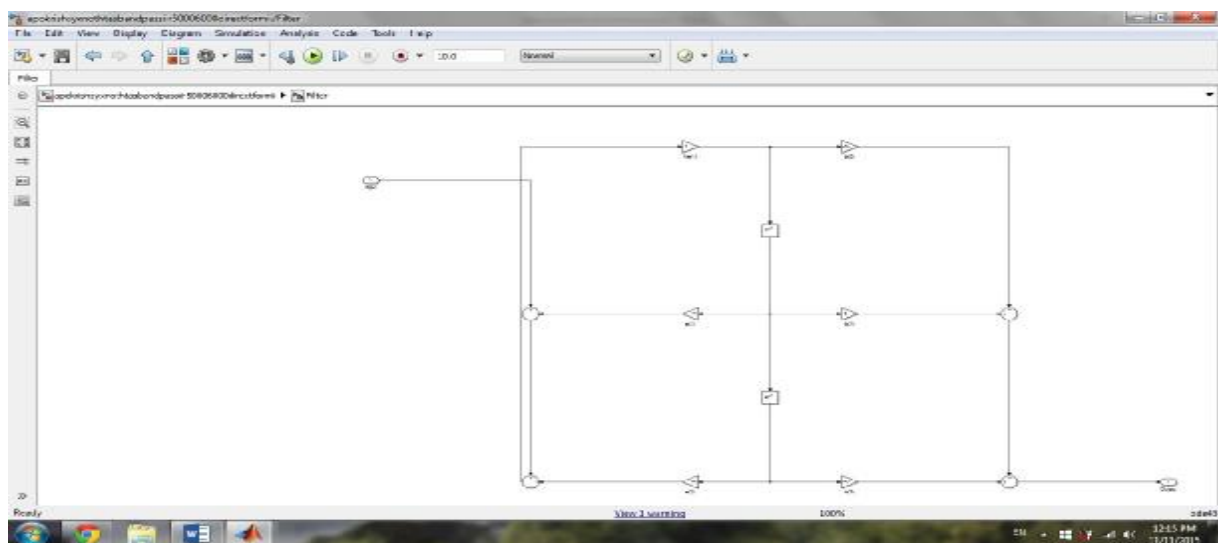
Εικόνα 3.4 Φάσμα πεδίο συχνότητας είσοδος 2 ημίτονων 0.1 και 0.25 KHz πριν το φιλτράρισμα $f_s/2=1000$



Εικόνα 3.8 Απόκριση συχνότητας IIR butter bandpass Direct form II



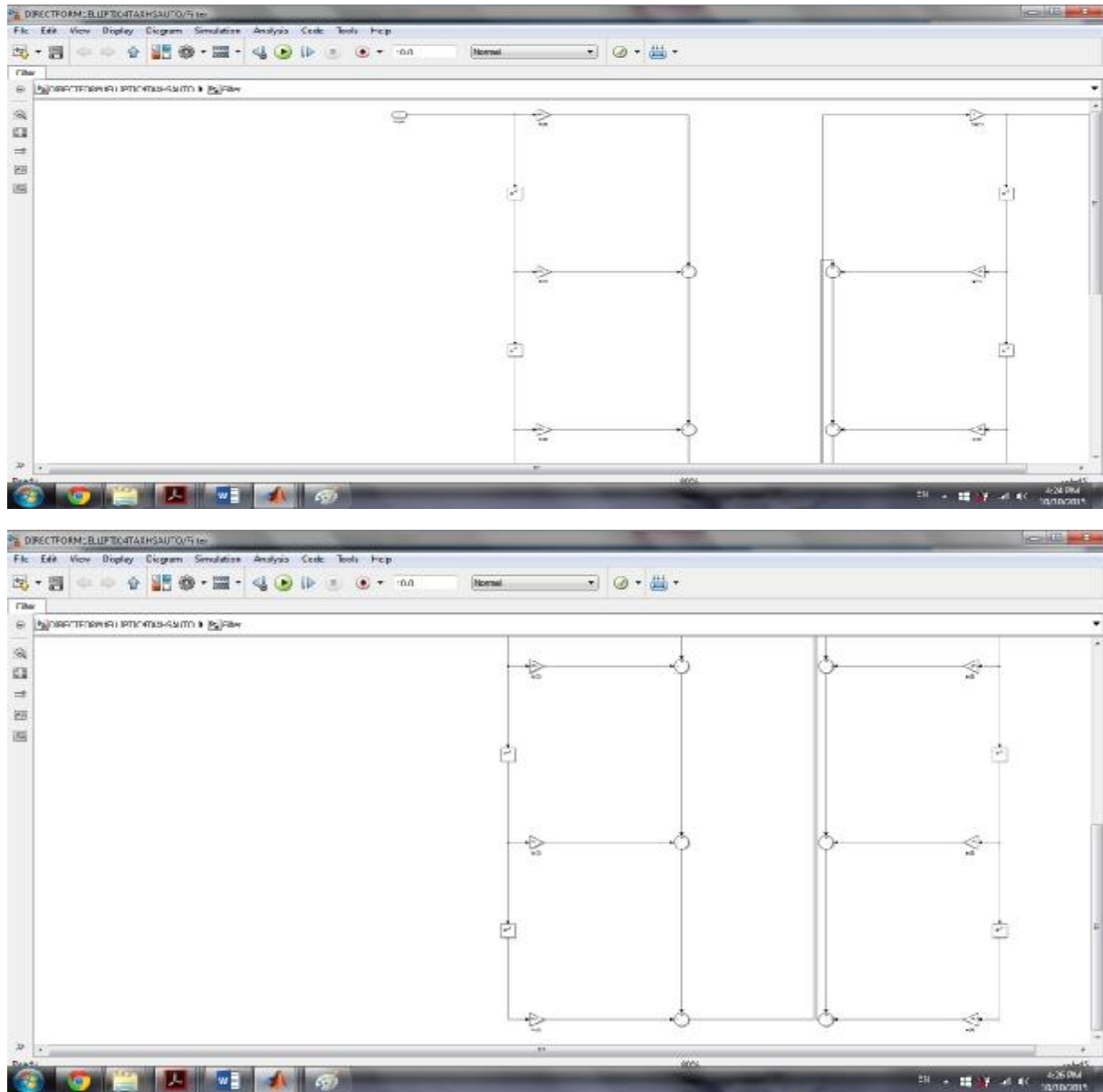
Εικόνα 3.9 Απόκριση συχνότητας



Εικόνα 3.10 Δομή υλοποίησης Direct Form II

3.2 Δομές συστημάτων άπειρης κρουστικής απόκρισης (IIR)

3.2.1 Direct Form I



Εικόνα 3.11 IIR Direct Form I

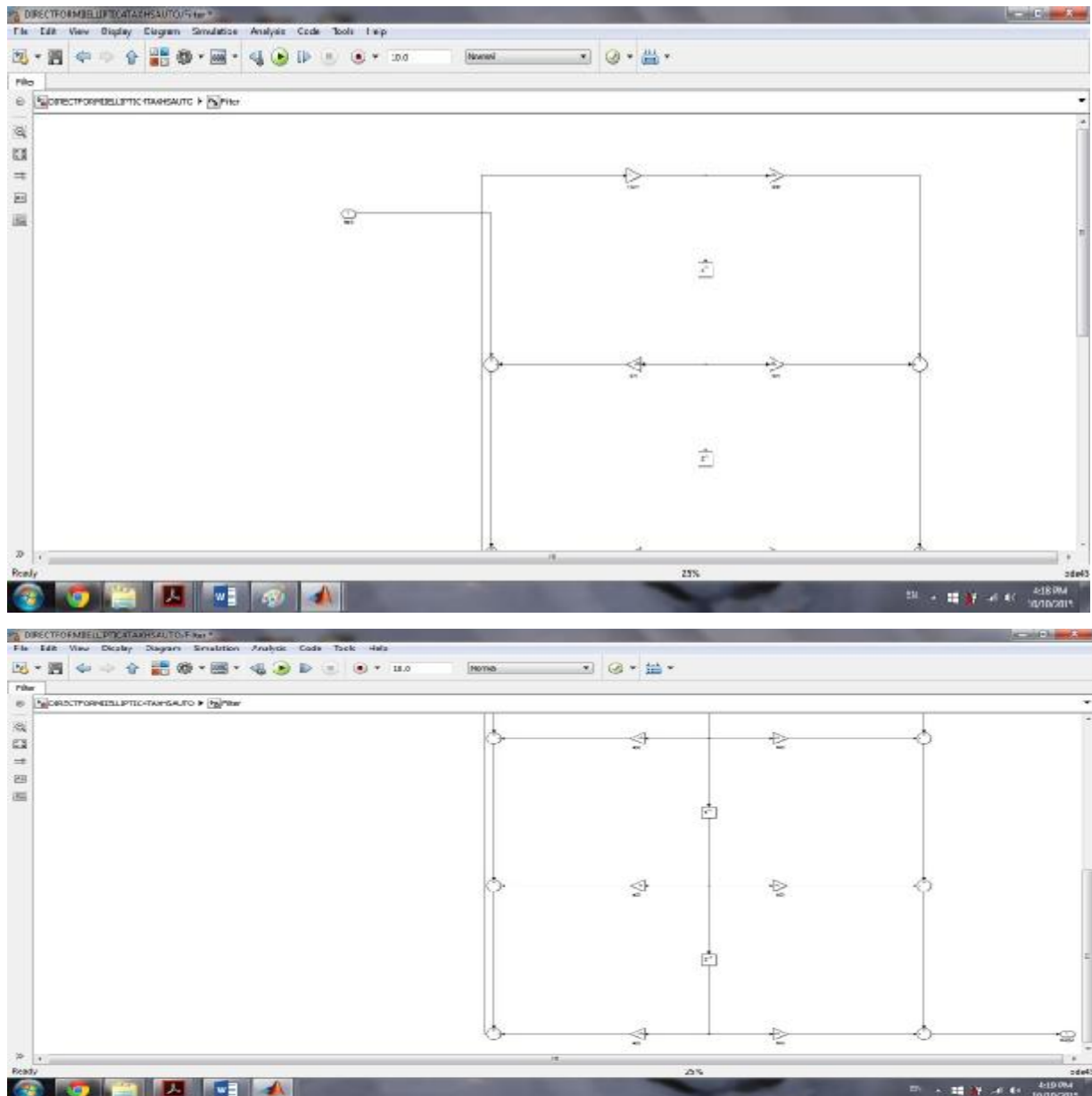
Η δομή κατασκευής IIR filter αποτελείται από πολλαπλασιαστές, αθροιστές και καθυστερητές οι καθυστερητές στην Direct form I είναι $2N$ όπου N η τάξη του φίλτρου. Στην αριστερή πλευρά της δομής σχεδιασμού του φίλτρου αντιστοιχούν τα μηδενικά και στην δεξιά οι πόλοι του φίλτρου

Το σύστημα είναι σαν την εξίσωση διαφορών

$$y(n) = \sum_{k=0}^M b_k x(n-k) - \sum_{k=1}^N a_k y(n-k) \quad (3.2.1.1)$$

$$\text{οπότε η συνάρτηση μεταφοράς του φίλτρου θα είναι } H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad (3.2.1.2)$$

3.2.2 Direct Form II



Εικόνα 3.12 IIR Direct form II

Ο προτιμότερος σχεδιασμός της δομής ενός IIR φίλτρου είναι η Direct form II γιατί βλέπουμε ότι χρησιμοποιεί N καθυστερητες δηλαδή 4 όσο και η τάξη του φίλτρου. Όταν μιλάμε για καθυστέρηση εννοούμε μνήμη άρα χρειαζόμαστε την μισή μνήμη άρα 50% λιγότερη μνήμη για αποθηκεύσουμε τους συντελεστές(προηγούμενων τιμών εισόδου εξόδου IIR φίλτρο). Για αυτό λέγετε και κανονικοποιημενη μορφή γιατί χρησιμοποιεί το λιγότερο αριθμό καθυστερητων. Επίσης βλέπουμε ότι στην Direct form II ουσιαστικά αλλάξαμε(αντιμετάθεση) τους συντελεστές τα a_k που είναι αριστερά και b_k είναι δεξιά δηλαδή πρώτα οι πόλοι και μετά μηδενικά Αυτό έχει ισχύ χωρίς να δημιουργείτε πρόβλημα εφόσον στα γραμμικά χρονικά αμετάβλητα δεν έχει σημασία ποιοι όροι ή πιο υποσύστημα θα μπει πρώτο οπότε η γενική συνάρτηση μεταφοράς του συστήματος παραμένει ίδια

3.2.3 Δομή καταρράκτη (σύνδεση σε σειρά cascaded)

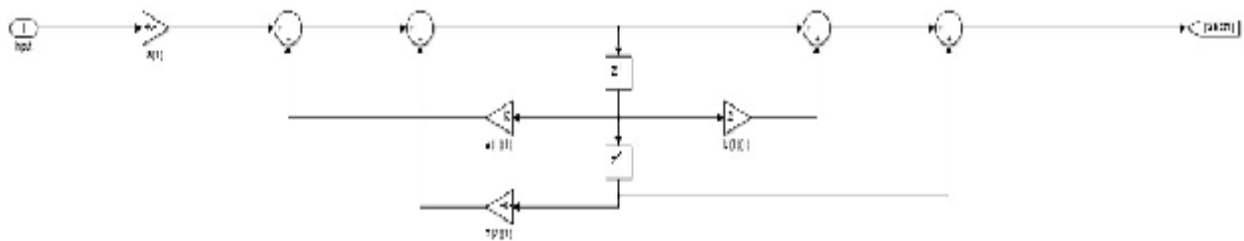
Η συνάρτηση μεταφοράς ενός IIR φίλτρου θα είναι

$$H(z) = \frac{\sum_{k=0}^{M-1} b_k z^{-k}}{\sum_{k=1}^{N-1} a_k z^{-k}} = H_0 \frac{\prod_{k=0}^{M-1} (1 - q_k z^{-k})}{\prod_{k=1}^{N-1} (1 - p_k z^{-k})} \quad (3.2.3.1)$$

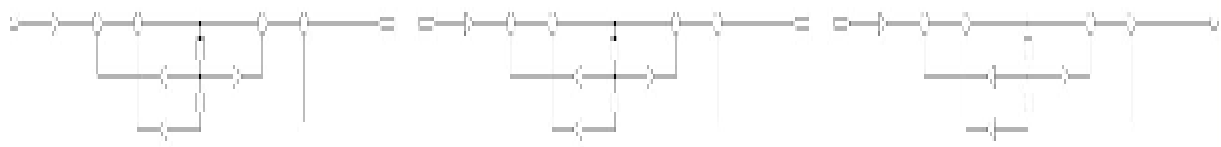
Όπου b_k και a_k οι συντελεστές του φίλτρου και q_k p_k τα μηδενικά και οι πόλοι αντίστοιχα. Η συνάρτηση μεταφοράς ενός δεύτερης τάξης IIR φίλτρου είναι

$$H_k(z) = \frac{b_0 k + b_1 k z^{-1} + b_2 k z^{-2}}{1 + a_1 k z^{-1} + a_2 k z^{-2}} \quad (3.2.3.2)$$

Δηλαδή θα είναι $H(z) = H_1(z)H_2(z)H_3(z) = H_0 \prod_{k=1}^3 H_k(z)$. Αυτό που καταλαβαίνουμε είναι ότι η έξοδος κάθε τμήματος(section) είναι η είσοδος του επομένου τμήματος



Εικόνα 3.13 2ης τάξης IIR φίλτρο δομή καταρράκτη



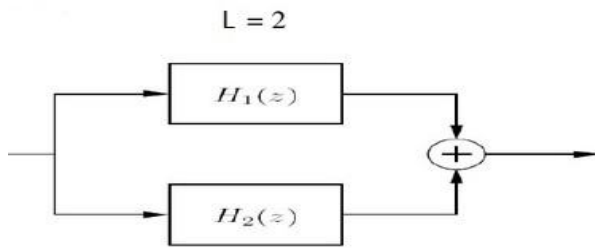
Εικόνα 3.14 Σύνδεση σε σειρά τριών φίλτρων 2ης τάξης. Άρα φίλτρο 6 τάξης σύνδεση σε σειρά ευθεία μορφή II. Η έξοδος ενός φίλτρου αποτελεί την είσοδο στο επόμενο $cascade(z) = H_1(z)H_2(z)H_3(z)$

3.2.4 Παράλληλη δομή

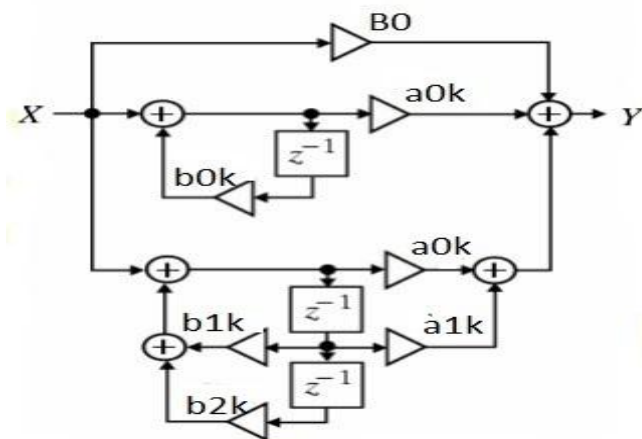
Η συνάρτηση μεταφοράς σύνδεση παράλληλα σε Direct Form I θα είναι θα είναι

$$H(z) = B_0 + \sum_{k=1}^L H_k(z) \quad (3.2.4.1)$$

$$H_k(z) = \frac{a_0 k + a_1 k}{1 + b_1 k z^{-1} + b_2 k z^{-2}} \text{ και 1 πρωτης ταξης } H_k(z) = \frac{a_0 k}{1 + b_0 k z^{-1}} \quad (3.2.4.2)$$

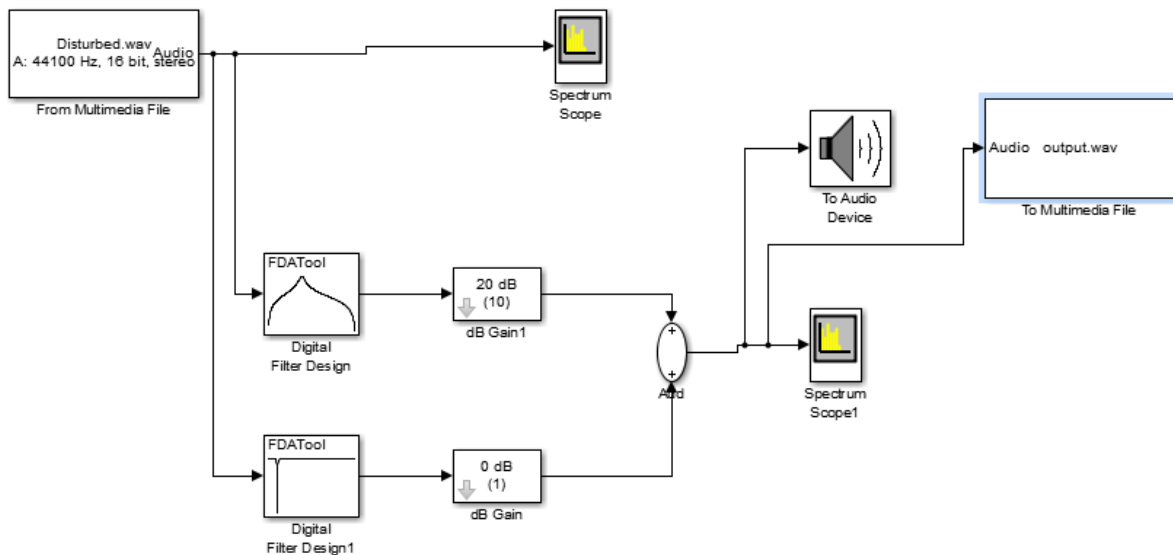


Εικόνα 3.15 Παράλληλη συνδεση



Εικόνα 3.16 3ης τάξης φίλτρο σύνδεση παράλληλα ενός φίλτρου 1ης τάξης και ένα 2 τάξης .Η σύνδεση φίλτρων παράλληλα είναι η πρόσθεση των εξόδων των δυο φίλτρων

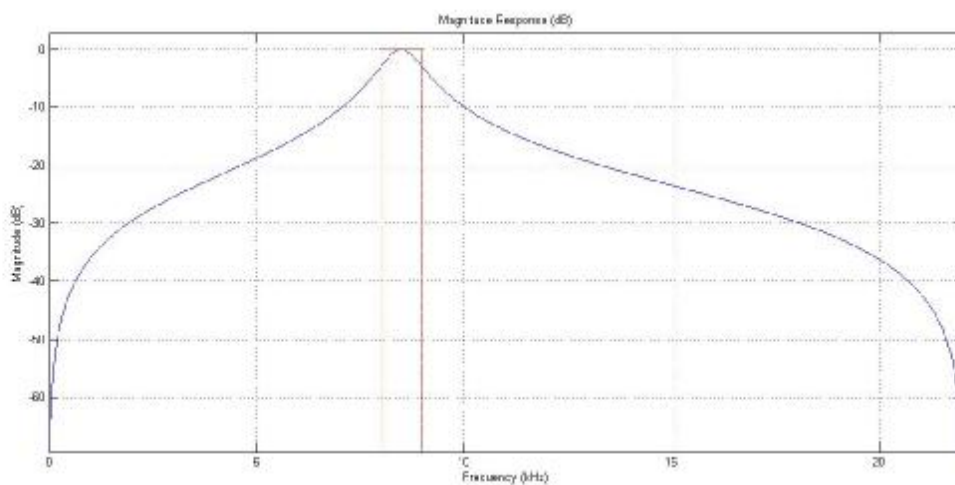
3.3 Συνδεση φιλτρων στο simulink



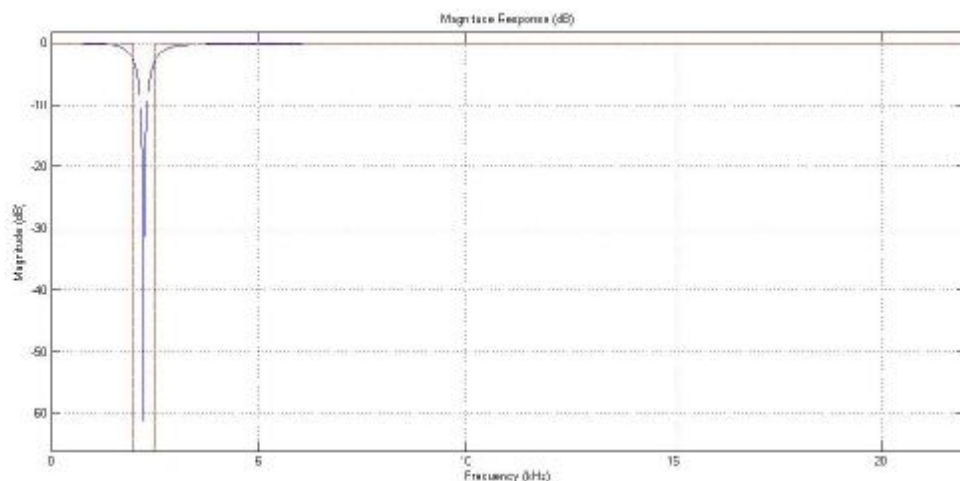
Εικόνα 3.17 Φιλτράρισμα της εισόδου με δυο IIR butterworth bandpass και bandstop κερδους σε db και συνδεση τους παραλληλα

Στην εικόνα 3.17 βλέπουμε την σύνδεση δυο φίλτρων παράλληλα με κεντρική συχνότητα το πρώτο $f_0=8485(f_0=\sqrt{8000 \times 9000})$ και το δευτερο $f_0=2236(f_0=\sqrt{2000 \times 2500})$ Η εισοδος στα φίλτρα

ένα αρχείο ηχου τυπου wav μέσω ενός block του Simulink (from multimedia file) με $f_s=44100$ Hz η F_n (συχνότητα Nyquist)=22050 Hz(η μέγιστη τιμή- περιοχή ακουστών συχνοτήτων για τον άνθρωπο) 16 bit βάθος ανάγνωσης (αριθμός bits για την αποθήκευση των δειγμάτων του αναλογικού ήχου που παίρνουμε όταν ψηφιοποιούμε τον ήχο .πχ $44100\text{Hz} \cdot 16\text{bit} = 705600 \cdot 2\text{καναλια} = 1411200$ bits /8=176400bytes ή 172 kilobyte. Γενικά αυτά τα δυο χαρακτηριστικά καθορίζουν την ποιότητα όταν ηχογραφούμε η όταν αναπαράγουμε ένα σήμα Το παραπάνω σύστημα αποτελείται από ένα στοιχείο της βιβλιοθήκης του Simulink “from multimedia file” με το οποίο μπορούμε να αναζητήσουμε ένα αρχείο ,από κάποιον φάκελο τύπου wav. Μετά επιλέγουμε από την βιβλιοθήκη του Simulink το στοιχείο “digital filter”με διπλό κλικ πάνω τους μπορούμε να σχεδιάσουμε τα φίλτρα μας .Τα δυο στοιχεία “Spectrum scope” βλέπουμε το φάσμα της εισόδου πριν το φιλτράρισμα και μετά βάζουμε ένα στοιχείο add και έτσι συνδέουμε τα φίλτρα μας παράλληλα .Για σύνδεση σε σειρά το ένα φίλτρο πίσω από το άλλο και η έξοδος του πρώτου φίλτρου αποτελεί είσοδο του δεύτερου. Τέλος έχουμε το στοιχείο ‘to multimedia file’ με το οποίο ,μπορούμε να αποθηκεύσουμε την είσοδο μετά το φιλτράρισμα.



Εικόνα 3.18 IIR Butterworth bandpass $f_{c1}=8000$ $f_{c2}=9000\text{Hz}$



Εικόνα 3.19 IIR Butterworth bandstop $f_{c1}=2000$ $f_{c2}=2500\text{Hz}$

Στην παραπάνω εικόνα βλέπουμε έναν equalizer που η μοντελοποίηση του έγινε στο Simulink του Matlab. Αποτελείται από ένα στοιχείο της βιβλιοθήκης του Simulink “from multimedia file” με το οποίο μπορούμε να εισάγουμε ένα αρχείο wav σαν είσοδο όπου και θα αποτελέσει την είσοδο στις 10 μπάντες -περιοχές φίλτρων. Τα φίλτρα η οι μπάντες συχνοτήτων είναι μπάντες οκτάβας αυτό σημαίνει ότι η χαμηλότερη ακραία-πλευρική συχνότητα του φίλτρου f_1 είναι το μισό της υψηλότερης πλευρικής f_2 συχνότητάς δηλαδή εάν έχουμε μια κεντρική συχνότητα $f_0=1000\text{Hz}$ τότε $f_1=1000/\sqrt{2}=707\text{ Hz}$ και $f_2=1000\sqrt{2}=1414$ και $f_0=\sqrt{f_1f_2} = 1000\text{Hz}$. Το εύρος ζώνης δηλαδή οι συχνότητες που βρίσκονται μεταξύ των δυο ακραίων συχνοτήτων f_1 και f_2 είναι το εύρος ζώνης της μπάντας η του φίλτρου $\Delta f=1/1.4142f_0=707$ η $f_2-f_1=707$ για ένα bandpass φίλτρο, για ένα low pass η high pass $\Delta f=fc$.

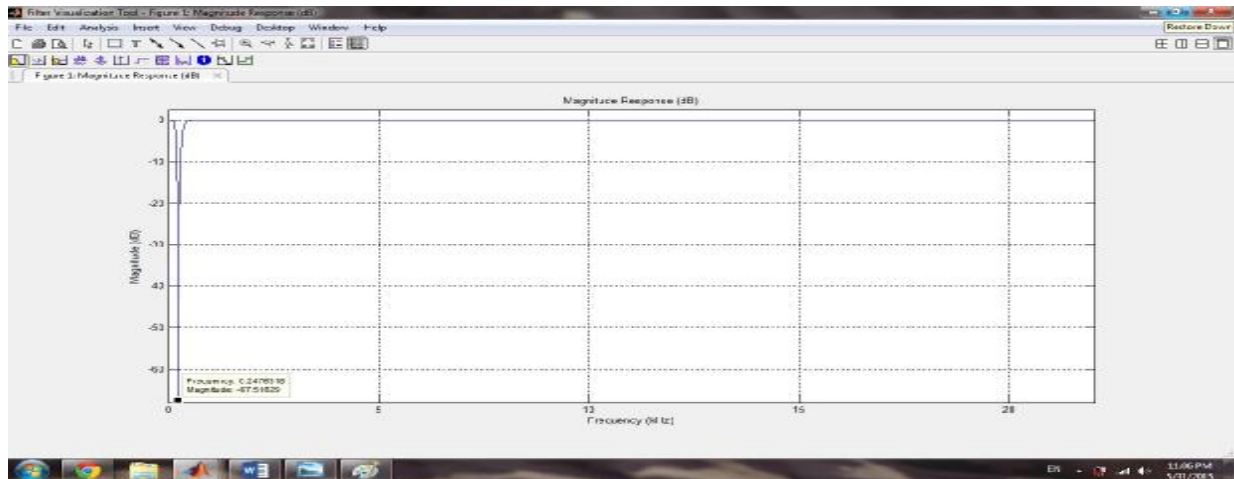
Τα φίλτρα συνδέονται παράλληλα δηλαδή η κρουστική απόκριση του συστήματος θα είναι $h(n)=h_1(n)+h_2(n)+\dots+h_{10}(n)$ με απόκριση συχνοτήτάς $H(e^{j\omega})=H_1(e^{j\omega})+H_2(e^{j\omega})+H_{10}(e^{j\omega})$ Το πρώτο φίλτρο είναι ένα low pass τα άλλα οκτώ είναι bandstop η bandreject και το τελευταίο ένα high pass με συχνότητες αποκοπής 31,62,125,250,500,1000,2000,4000,8000,16000Hz.

Τα φίλτρα υλοποιήθηκαν με το fdatool (filter design analysis tool) στοιχείο της βιβλιοθήκης Matlab .

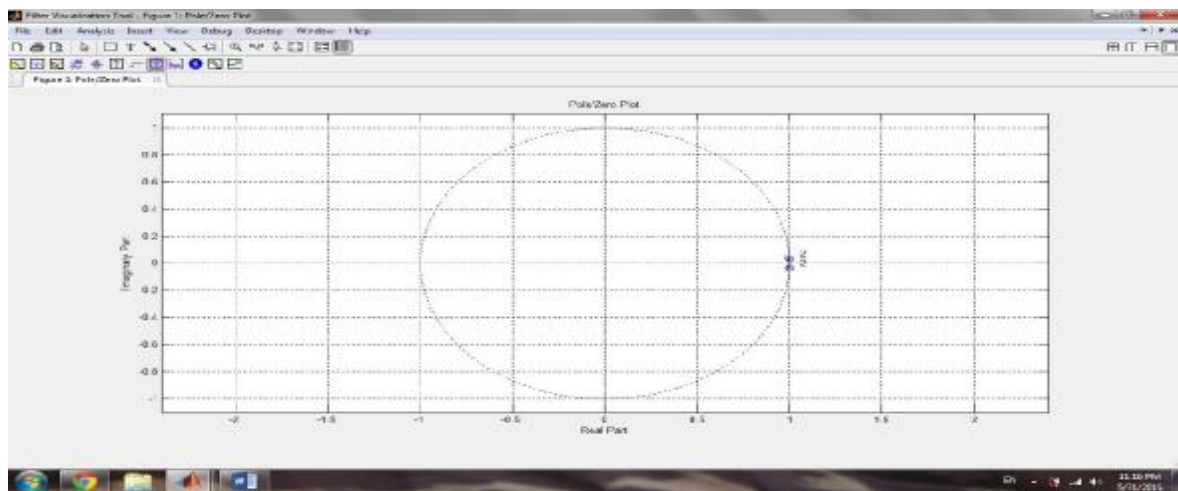
Τα φίλτρα είναι άπειρης κρουστικής απόκρισης(IIR) γιατί μπορούμε να πέτυχουμε καλύτερη προσέγγιση- σχεδιασμό του φίλτρου έτσι ώστε να πλησιάζει στο ιδανικό φίλτρο και να είναι μικρότερης τάξης (κάτι που σημαίνει λιγότεροι συντελεστές λιγότερες πράξεις λιγότερη μνήμη) σε σχέση με ένα FIR φίλτρο και τύπου Butterworth τα οποία είναι καταλληλά για ακουστικές εφαρμογές γιατί δεν έχουν καθόλου κυμάτωση τόσο στην ζώνη διέλευσης και στην ζώνη αποκοπής. Τα φίλτρα είναι δεύτερης τάξης ο ρυθμός δειγματοληψίας τους είναι 44100Hz.

Η έξοδος-απόκριση κάθε φίλτρου ενισχύεται με το αντίστοιχο κέρδος(gain db.) block της βιβλιοθήκης του Simulink.Εφόσον τα φίλτρα είναι only cut τότε προφανώς μέγιστο κέρδος θα έχουν τα 0 db. και ελάχιστο στην προκειμένη περίπτωση $g \leq -40\text{db}$. Τέλος το φιλτραρισμένο σήμα μέσω του block του Simulink(to multimedia file) μπορούμε να το εξάγουμε και να το αποθηκεύσουμε με τις αλλαγές τις οποίες έχουμε κάνει. Επίσης μπορούμε να ακούσουμε το τελικό φιλτραρισμένο σήμα με το στοιχείο (to audio device).Τα δυο block(fft spectrum) μπορούμε να δούμε το φάσμα συχνοτήτων του αρχικό σήματος και του τελικού φιλτραρισμένου σήματος.

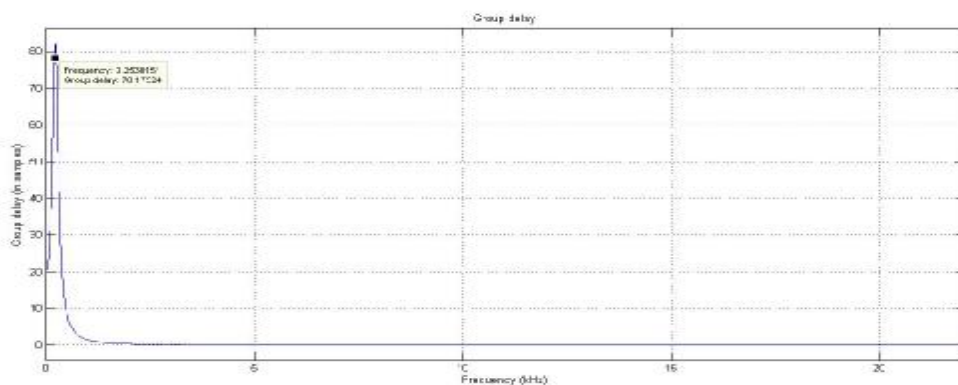
Η παραπάνω προσομοίωση ενός γραφικού ισοσταθμιστή θα μπορούσε να χρησιμοποιηθεί σαν equalizer PA (public address) δηλαδή σε χώρους που συγκεντρώνονται πολλά άτομα πχ συναυλίες θέατρα όπου αυτός ο ισοσταθμιστής θα συνδέεται με κάποιο ηχείο το οποίο εκ κατασκευής θα δίνει μια ενίσχυση. Οπότε δεν χρειαζόμαστε ενίσχυση(θετικό κέρδος) αλλά μέχρι 0 db. δηλαδή ανώτερο όριο κέρδους 0db., τα φίλτρα θα κάνουν μόνο cut και όχι boost (εξασθένιση και όχι ενίσχυση) Έτσι ένας τέτοιος equalizer θα μπορούσε να έχει διορθωτικό χαρακτήρα δηλαδή να διορθώνουμε κάποια περιοχή συχνοτήτων αλλά όχι να την ενισχύουμε κάτι που μας θα μας δώσει παραδείγματος χάρη ένα ηχείο.



Εικόνα 3.22 IIR Butterworth bandstop φίλτρο με κεντρική συχνότητα στα 250 hz(0.25 kHz)



Εικόνα 3.23 Διάγραμμα πόλων μηδενικών του ίδιου φίλτρου bandstop f_c 250Hz 4^{ης} τάξης



Εικόνα 3.24 Καθυστέρηση ομάδας IIR φίλτρου bandstop 0.25 KHz

Καθυστέρηση ομάδας IIR φίλτρου. Βλέπουμε ότι η συχνότητα 0.25kHz θα καθυστερεί περίπου 79 δείγματα όχι σταθερή καθυστέρηση ομάδας σε όλες τις συχνότητες (μη γραμμικής φάσης φίλτρο),αλλοιώνετε η μορφή του σήματος

```

>> sys=filt(b,a)

sys =

    0.9824 - 3.927 z^-1 + 5.889 z^-2 - 3.927 z^-3 + 0.9824 z^-4
-----
    1 - 3.962 z^-1 + 5.889 z^-2 - 3.892 z^-3 + 0.965 z^-4

Sample time: unspecified
Discrete-time transfer function.

>>

```

Εικόνα 3.25 Συνάρτηση μεταφοράς 4ης τάξης bandstop φίλτρου

4. Equalizers(μεταβλητού κέρδους)

4.1. Ολοπεράτο φίλτρο (allpass)

Ο ρόλος ενός allpass φίλτρου είναι ιδιαίτερα σημαντικός για την πραγματοποίηση των low pass,high pass,bandpass και bandreject

Ένα allpass φίλτρο επιτρέπει να περνάνε όλες οι συχνότητες με το ίδιο κέρδος=1v=0db.Ωστόσο η σημασία του είναι ότι μετατοπίζει την φάση του και έτσι σε συνδυασμό με μια είσοδο μπορεί να δώσει διαφορετικές υλοποιήσεις φίλτρων. Ουσιαστικά το μέγιστο της μετατόπιση φάσης γίνεται στις ψηλές συχνότητες -180 μοίρες που αυτό είναι κάτι σαν αλλαγή πρόσημου. Εφόσον θα έχω για τις χαμηλές συχνότητες + από το allpass φίλτρο λόγω της μηδενικής μετατόπισης φάσης και προσθέσω την είσοδο οπότε (+ και +) με αυτό τον τρόπο κρατάω τις χαμηλές, και φεύγουν οι ψηλές γιατί εκεί έχω (-)οπότε θα έχω ένα low pass.(Udo Zolzer 2002)

Για high pass εάν έχω εάν (-) για τις ψηλές συχνότητες από το allpass εφόσον έχω αντιστροφή φάσης και αφαιρέσω την είσοδο τότε (- και -)οπότε έτσι θα κρατήσω τις ψηλές συχνότητες και εφόσον θα έχω + για τις χαμηλές συχνότητες αυτές είναι τώρα που θα φύγουν οπότε θα έχω ένα high pass. Για πρώτης τάξης ολοπερατό η συνάρτηση μεταφοράς

$$A(z)=\frac{z^{-1}+c}{1+cz^{-1}} \quad (4.1.1)$$

Από εδώ καταλαβαίνουμε ότι ένα allpass φίλτρο έχει ένα πόλο στο c και ένα μηδενικό στο 1/c και

$$c=\frac{\tan\left(\frac{\pi fc}{fs}\right)-1}{\tan\left(\frac{\pi fc}{fd}\right)+1}$$

(4.1.2) Το c εδώ ελέγχει την συχνότητα αποκοπής του ολοπερατού φίλτρου .Αρα ένα allpass θα έχει τόσους πόλους όσα και μηδενικά. Οπότε για 1^{ης} τάξης στην συχνότητα αποκοπής θα έχω ένα πόλο και ένα μηδενικό οπότε -45 μοίρες για πόλο και +45 μοίρες για μηδενικό οπότε πάντα για 1^{ης} ταξης φίλτρο πάντα στις 90 μοίρες η συχνοτητα αποκοπης .

Σε ένα πρώτης τάξης ολοπερατό φίλτρο(allpass) οι ψηλές συχνότητες φθίνουν στις -180 μοίρες και πάνω από την συχνότητα αποκοπής -90 μοίρες. (Udo Zolzer 2002)

Ένα δεύτερης τάξης ολοπερατό (allpass) που θα χρησιμοποιηθεί για bandpass η για bandstop η συνάρτηση μεταφοράς είναι

$$A(z)=\frac{-c+d(1-c)z^{-1}+z^{-2}}{1+d(1-c)z^{-1}-cz^{-2}} \quad (4.1.3)$$

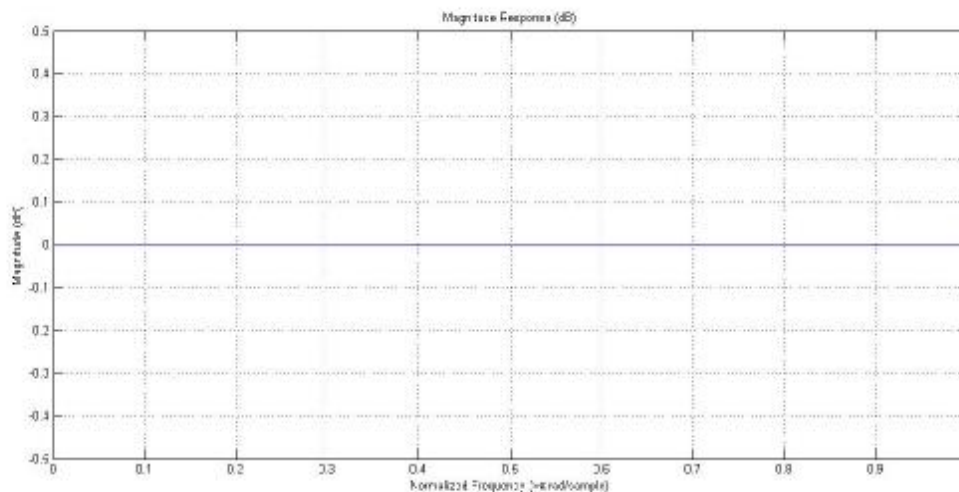
$$c=\frac{\tan\left(\frac{\pi fb}{fs}\right)-1}{\tan\left(\frac{\pi fb}{fd}\right)+1} \quad (4.1.4)$$

το c ρυθμίζει το bandwidth και το d την συχνότητα αποκοπής $d=-\cos(2*\pi*fc/fs)$ (4.1.5)

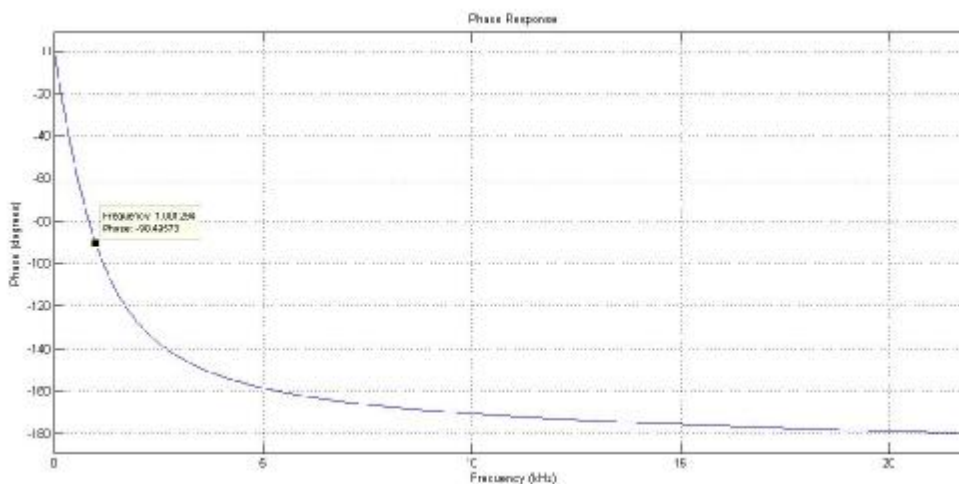
Σε ένα 2ταξης ολοπερατό φίλτρο οι ψηλές συχνότητες φθίνουν στις -360 μοίρες και πάνω από την κεντρική συχνότητα (εκεί που γίνεται η καμπύλη στην φάση του φίλτρου) είναι -180 μοίρες

(Udo Zolzer 2002). Η σημασία στην κατασκευή ενός low pass/high pass και bandpass/bandstop φίλτρου μέσω ενός allpass έχει να κάνει ότι στο allpass χρειαζόμαστε να υπολογίσουμε μόνο έναν συντελεστή, το c. Οπότε εάν θέλουμε να αλλάζουμε την γωνιακή συχνότητα ,δηλαδή την συχνότητα αποκοπής($\omega_c=2*\pi*fc/fs$) τότε χρειαζόμαστε να υπολογίσουμε μόνο μια παράμετρο ενώ σε άλλες περιπτώσεις χρειαζόμαστε να υπολογίσουμε όλους τους άλλους συντελεστές, δηλαδή ολόκληρη την συνάρτηση μεταφοράς του φίλτρου. Οπότε κατασκευάζοντας ένα low pass φίλτρο μέσω ενός allpass τότε έχουμε την δυνατότητα και στο low pass να υπολογίσουμε μόνο ένα συντελεστή το c εάν αλλάξει η συχνότητα αποκοπής. Τέλος κοιτώντας την παράμετρο c για 1^{ης} και 2^{ης} τάξης ολοπερατό φίλτρο βλέπουμε ότι για το πρώτης τάξης η συχνότητα αποκοπής έχει διαφορετικό εύρος ορισμού δηλαδή $[0 \pi/2]$ με $\pi/2$ (συχνότητα Nyquist) ενώ για 2^{ης} τάξης $d = -\cos(2*\pi*fc/fs)$ άρα $[0 \pi]$ για μονόπλευρη απόκριση συχνότητάς με π (Nyquist συχνότητα).

Αυτή η διαφορά έχει να κάνει με το ότι το 2^{ης} τάξης allpass φίλτρο η κεντρική γωνιακή συχνότητα του φίλτρου σε rad/sample είναι στις -180 μοίρες μετατόπιση φάσης άρα 0 έως π για ω ενώ στο 1^{ης} τάξης έχει μετατόπιση φάσης στην συχνότητα αποκοπής -90 μοίρες οπότε $0 \pi/2$ για ω .



Εικόνα 4.1 Απόκριση συχνότητάς ολοπερατού φίλτρου 1ης τάξης



Εικόνα 4.2 Απόκριση φάσης πρώτης τάξης ολοπερατού φίλτρου $fc=1000$ άρα $c=-0.8677$ -90 μοίρες

4.2 Low /High pass Bandpass Bandreject

4.2.1 Low pass

Αρχικά θα μπορούσαμε να αναφέρουμε ότι ένα πρώτης τάξης low pass η ένα high pass μπορεί να πραγματοποιηθεί ως εξής direct path+allpass αλλά κλιμακούμενο με $\frac{1}{2}$

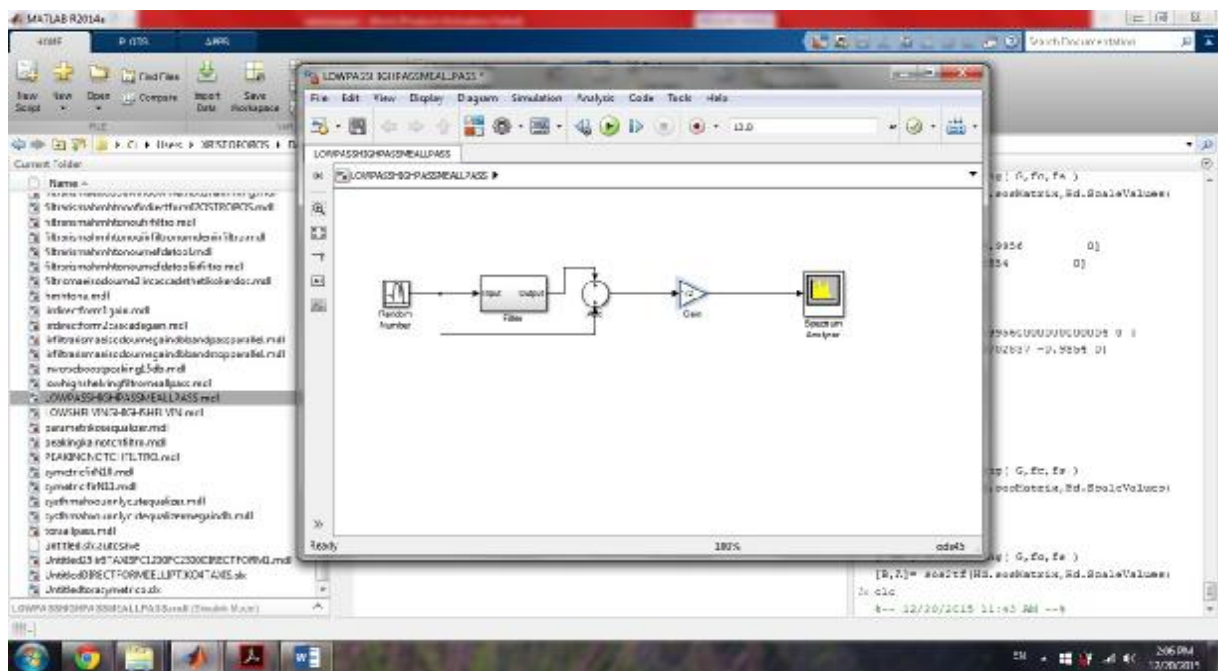
Direct path σημαίνει: είναι το ένα(1) μπροστά από την εξίσωση πιο κάτω είναι το αρχικό σήμα εισόδου το οποίο προστίθεται στην έξοδο του ενός allpass φίλτρου.

Έτσι ένα ένα low pass φίλτρο μπορεί να επιτευχθεί εάν προσθέσουμε το σήμα εισόδου στο σήμα εξόδου ενός allpass φίλτρου .Η συνάρτηση μεταφοράς ενός ψηφιακού low pass φίλτρου είναι

$$H(z)=\frac{1}{2}[1+A(z)] \quad (4.2.1.1)$$

όπου $A(z)$ είναι η συνάρτηση μεταφοράς ενός πρώτης τάξης ολοπερατού φίλτρου (4.1.1)

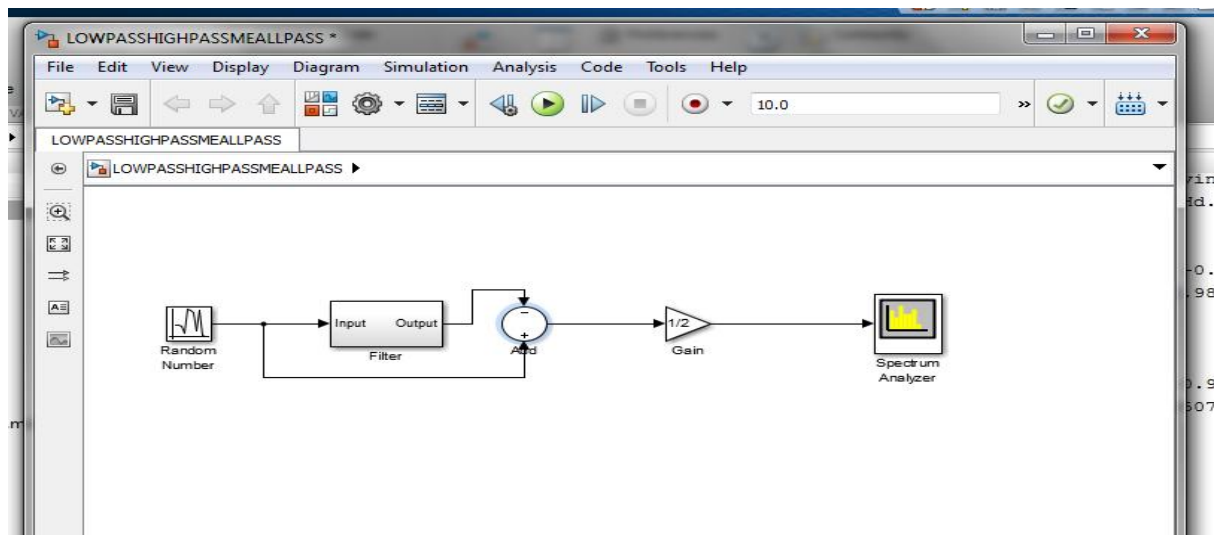
(Udo Zolzer 2002)



Εικόνα 4.3 Δομή κατασκευής 1ης τάξης χαμηλοπερατό φίλτρο

4.2.2 Highpass

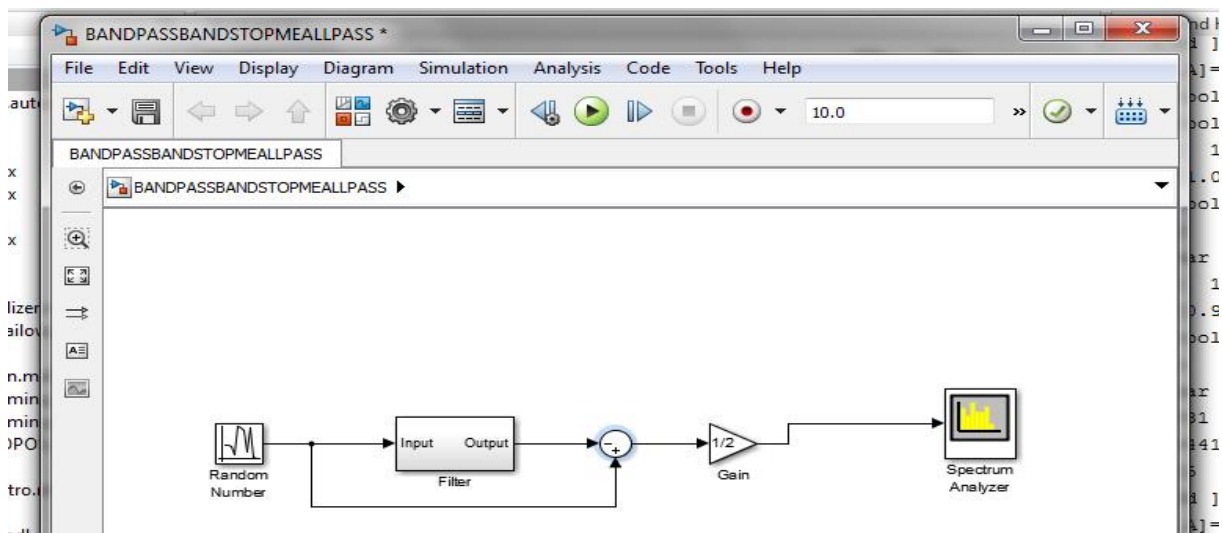
Ένα high pass μπορεί να πραγματοποιηθεί ως εξής: direct path-allpass αλλά κλιμακούμενο με $\frac{1}{2}$.Έτσι ένα high pass φίλτρο μπορεί να επιτευχθεί εάν αφαιρέσουμε το σήμα εισόδου από την έξοδο ενός ολοπερατού φίλτρου ουσιαστικά αυτό σημαίνει μετατόπιση φάσης δηλαδή εφόσον ένα πρώτης τάξης allpass παρουσιάζει μετατόπιση φάσης στην έξοδο του -180 μοιρών μπορεί να μας δώσει σε μια υλοποίηση low pass/high pass .Ένα high pass φίλτρο θα έχει μια συνάρτηση μεταφοράς $H(z)=\frac{1}{2}[1-A(z)]$ (Udo Zolzer 2002) (4.2.2.1)



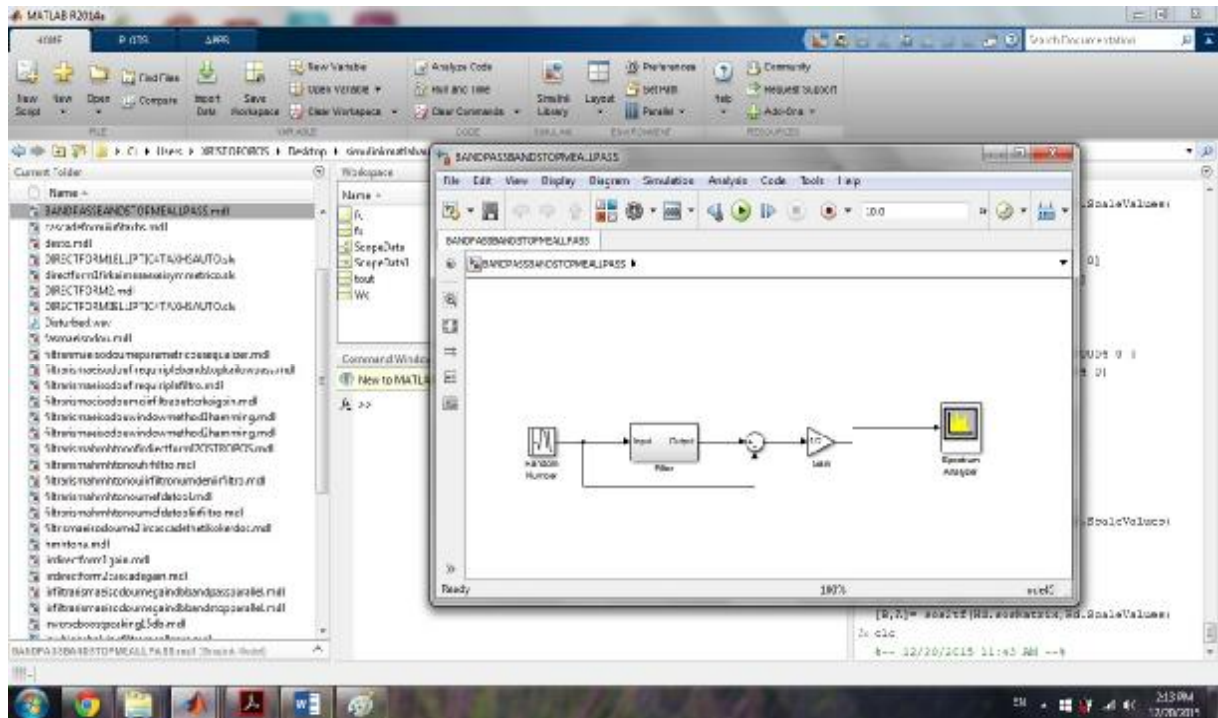
Εικόνα 4.4 Δομή κατασκευής 1ης τάξης high pass filter

4.2.3 Bandpass/bandstop

Ένα bandpass και ένα bandstop υλοποιείται με ένα $2^{\text{ης}}$ τάξης ολοπερατό φίλτρο-direct path κλιμακούμενο με $\frac{1}{2}$. Έτσι ένα bandpass υλοποιείται εάν αφαιρέσουμε την είσοδο από την έξοδο ενός 2 τάξης ολοπερατού φίλτρου, ενώ ένα bandreject υλοποιείται εάν προσθέσουμε το σήμα εισόδου στην έξοδο ενός $2^{\text{ης}}$ τάξης ολοπερατού φίλτρου. Η συνάρτηση μεταφοράς ενός bandpass φίλτρο θα είναι $H(z)=\frac{1}{2}[1-A_2(z)]$ (4.2.3.1) και ενός bandreject είναι $H(z)=\frac{1}{2}[1+A_2(z)]$ όπου $A_2(z)$ είναι η εξίσωση (4.1.3) και (4.1.4) (Udo Zolzer 2002)



Εικόνα 4.5 2ης τάξης bandpass φίλτρο



Εικόνα 4.6 2ης τάξης bandreject φίλτρο

4.3 Shelving φίλτρο

Ενώ ένα low pass η high pass έχουν την δυνατότητα να αφαιρούν ουσιαστικά συχνότητες η καλύτερα χρησιμεύουν στο να κάνουν ‘cut’ εφόσον είναι φίλτρα με κέρδος 0 db, δεν χρησιμοποιούνται συχνά σε equalizers και ακουστικές εφαρμογές

Τα φίλτρα που χρησιμοποιούνται ως ισοσταθμιστές (equalizers) σε ακουστικές εφαρμογές συνήθως είναι τα shelving (low shelving/high shelving) και τα (peaking/notch) 2 τάξης άπειρης κρουστικής απόκρισης (IIR)

Να αναφέρουμε ότι με τον όρο equalizers εννοούμε φίλτρα με τα οποία μπορούμε να αυξομειώνουμε την στάθμη (το κέρδος) ενός φίλτρου προκαλώντας ενίσχυση η εξασθένιση μιας συγκεκριμένης περιοχής στο φάσμα συχνοτήτων μιας εισόδου. Το εργαλείο που χρησιμοποιείται λέγεται equalizer (ισοσταθμιστής) και αποτελείται από ένα σύνολο φίλτρων με βασικό στόχο του χρήστη να αλλάξει το χρώμα των μουσικών οργάνων και να ακούσει όπως θέλει αυτός ένα μουσικό κομμάτι (υποκειμενικά) η για διορθώσεις σε ηχογραφήσεις η studio κ.α

Αρχικά τα πρώτα λέγονται shelving γιατί ουσιαστικά υπάρχει σαν ένα ράφι (shelf) με το αντίστοιχο κέρδος πάνω (boost) η κάτω (για cut) από την συχνότητα αποκοπής. Ένα low shelving μπορεί να ενισχύσει (boost) ένα εύρος συχνοτήτων με θετικό κέρδος (+gain db) αλλά και να εξασθενήσει (cut) ένα εύρος συχνοτήτων με αρνητικό κέρδος (-gain db) δηλαδή να έχει μεταβλητό κέρδος (variable gain) ενώ ένα low pass κάνει μόνο ‘cut’. Τα ίδια ισχύουν και για ένα high frequency shelving αλλά εκείνο χρησιμοποιείται για φιλτράρισμα η εξομάλυνση υψηλών συχνοτήτων.

Τα low/high frequency shelving φίλτρα χρησιμοποιούνται όταν θέλουμε να «μαλακώσουμε η να δώσουμε έμφαση σε ένα εύρος συχνοτήτων ανάλογα με την συχνότητα αποκοπής του φίλτρου. Το εύρος ζώνης ενός low/high shelving είναι ίσο με την συχνότητα αποκοπής.

Τα shelving φίλτρα είναι σχεδιασμένα για να χρησιμοποιούνται στα άκρα του φάσματος. Παρακάτω παρουσιάζονται τρεις τρόποι σχεδιασμού που ο καθένας έχει τα δικά του πλεονεκτήματα και ανάλογα με αυτά χρησιμοποιούνται σε ανάλογους σχεδιασμούς ηχητικών συστημάτων

4.3.1 Low frequency shelving φίλτρο

4.3.1.1 Πρώτη μέθοδος σχεδιασμού (απευθείας σχεδιασμός)

Η αρχική συνάρτηση μεταφοράς ενός low shelving φίλτρου για την περίπτωση boost είναι

$$\frac{s+V_0\omega_c}{s+\omega_c} \quad (\text{Udo Zolzer :2}^{\text{η}} \text{ έκδοση 2008}) \quad (4.3.1.1)$$

και για cut $\frac{s+\omega_c}{s+\omega_c/V_0}$ (4.3.1.2)

Προφανώς καταλαβαίνουμε από τις παραπάνω εξισώσεις ότι το low frequency shelving φίλτρο προκύπτει από ένα low pass $\frac{\omega_c}{s+\omega_c}$ με το Dc κέρδος πάνω από την συχνότητα $\omega=0$ και πάνω από την συχνότητα αποκοπής κατά -3db.

Εδώ το κέρδος ορίζετε κατευθείαν για boost είναι $V_0>1$ και για cut ορίζετε $V_0\text{cut}<1$

Με $V_0=10^{G/20}$ (boost) και $V_0\text{cut}=10^{-G/20}$ (4.3.1.3)

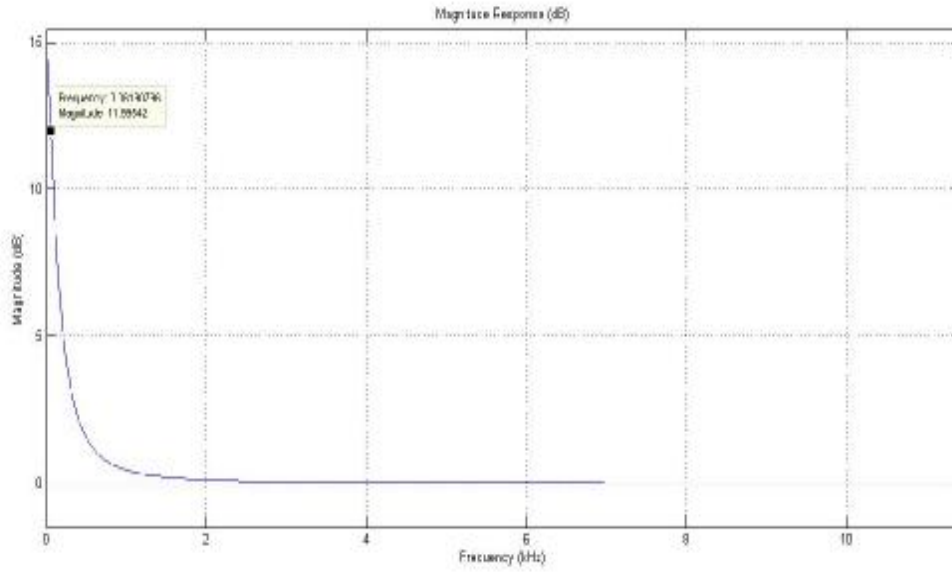
Η περίπτωση cut είναι η ανεστραμμένη μορφή της περίπτωσης boost προφανώς για φίλτρο με σταθερή συχνότητα αποκοπής(fixed ω_c). Αυτό ουσιαστικά επιτυγχάνεται με μια εναλλαγή των πόλων και των μηδενικών στην συνάρτηση μεταφοράς στο s πεδίο. Στο αναλογικό φίλτρο οι πόλοι ως γνωστό από πριν πρέπει να βρίσκονται στην αριστερή πλευρά του s πεδίου έτσι ώστε να βρίσκονται μετά τον διγραμμικό μετασχηματισμό μέσα στα όρια του μοναδιαίου κύκλου ($z=-1$ $z=1$). (Udo Zolzer :2^η έκδοση 2008)

Η ευστάθειά των φίλτρων και για τις δυο περιπτώσεις boost και cut επιτυγχάνεται με το ότι τα φίλτρα πρέπει να είναι ελάχιστης φάσης δηλαδή και οι πόλοι και κυρίως τα μηδενικά του φίλτρου πρέπει να βρίσκονται εντός του μοναδιαίου κύκλου έτσι ώστε μια εναλλαγή αυτών των δυο να με δώσει επίσης ένα ευσταθές φίλτρο και στις δυο περιπτώσεις

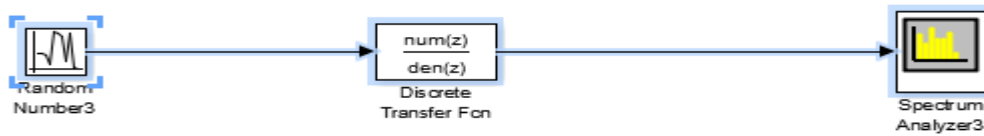
Ο κώδικας για την υλοποίηση 1^{ης} τάξης low shelving φίλτρου που προκύπτει από την (4.3.1.1) και την (4.3.1.2) είναι(στο παράρτημα κώδικας 1)

Μετά τον διγραμμικό μετασχηματισμό το 1^{ης} τάξης φίλτρου περιγράφετε στο πεδίο της συχνότητάς με την συνάρτηση μεταφοράς $H(z) \frac{b_0+b_1z^{-1}}{1+a_1z^{-1}}$ (Udo Zolzer :2^η έκδοση 2008) (4.3.1.4)

Και στο χρόνο με μια εξίσωση διαφορών $y(n)=b_0x(n)+b_1x(n-1)-a_1y(n-1)$

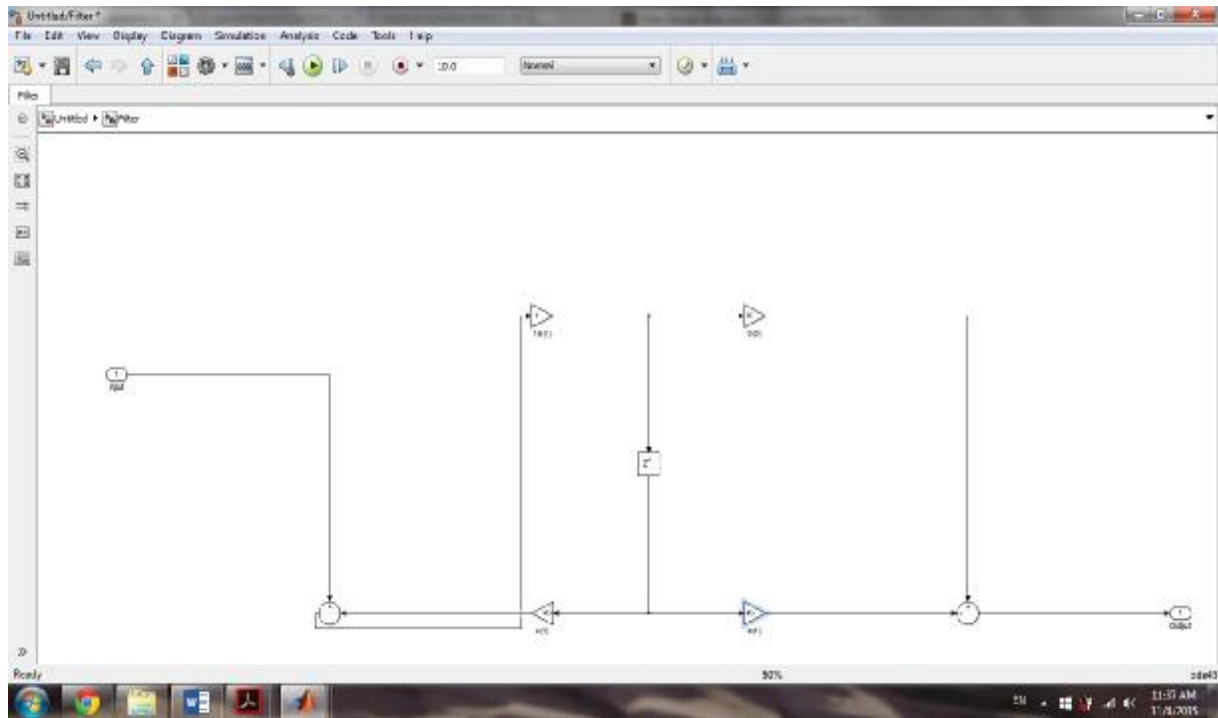


Εικόνα 4.7 Low shelving φίλτρο συχνότητα αποκοπής 0.06kHz



Εικόνα 4.8 Μοντελοποίηση στο simulink lowfrequency shelving φίλτρο

Και δομή υλοποίησης Direct form II



Εικόνα 4.9 Direct form II

4.3.1.2 Δεύτερη μέθοδος σχεδιασμού low frequency shelving φίλτρου (ημιπαραμετρική)

Ένα low frequency shelving φίλτρο για την περίπτωση boost έχει συνάρτηση μεταφοράς στο s πεδίο(αναλογικό)

$$H(s)=1+H_0\frac{\omega_c}{s+\omega_c} \quad (\text{Udo Zolzer :2}^{\text{η}} \text{ έκδοση } 2008) \quad (4.3.1.2.1)$$

η οποία προκύπτει από την έξοδο ενός low pass φίλτρου (το οποίο πολλαπλασιάζεται- σταθμίζεται με μια σταθερά=κέρδος ενός low pass)+directpath

Η (4.3.1.2.1) είναι ιδίως(ισοδύναμες) με την (4.3.1.1) αλλά στην (4.3.1.2.1) εφαρμόζεται η ημιπαραμετρική δομή του φίλτρου: Εδώ το κέρδος και η συχνότητα αποκοπής μπορεί να είναι ανεξάρτητα μεταξύ τους. Σύμφωνα με αυτήν μπορούμε να αλλάξουμε το κέρδος που θέλουμε να δώσουμε σε ένα εύρος- μάντα συχνότητων ανεξάρτητα από την συχνότητα αποκοπής. Αυτό γίνεται εμφανές στην (4.3.1.2.1) όπου ουσιαστικά πετάει το κέρδος $V_0=1+H_0$ έξω έτσι ώστε εάν θέλω να αλλάξω το V_0 που θα αλλάξει και το $H_0(H_0=V_0-1)$ για να πάρω το σωστό φίλτρο εφόσον $V_0=1+H_0$.Ετσι τώρα μόνο το V_0 θα αλλάξει. Γι αυτό οι άλλοι συντελεστές θα πρέπει να είναι ανεξάρτητοι από το κέρδος έτσι ώστε εάν θέλω να αλλάξω το κέρδος δεν είναι απαραίτητο να κάτσω να υπολογίσω όλους τους συντελεστές της συνάρτησης μεταφοράς αλλά μόνο έναν το H_0 .

Direct path : είναι το 1 μπροστά από το H_0 .Ουσιαστικά είναι το αρχικό αυθεντικό σήμα εισόδου το οποίο αντιγράφεται στην έξοδο ενός low pass φίλτρου. Εάν προσθέσουμε στο direct path+low pass φίλτρο τότε θα πάρουμε ένα σήμα στην έξοδο στο οποίο υπάρχει ενίσχυση στις χαμηλές συχνότητες με $V_0=1+H_0$.Οπου $1+H_0$ είναι το DC gain ενός low frequency shelving φίλτρου και

$$V_0=1+H_0 \text{ το DC κέρδος όταν } \omega=0 \quad (4.3.1.2.2)$$

$$H_0=V_0-1 \text{ είναι το κέρδος ενός low pass} \quad (4.3.1.2.3)$$

$$V_0=10^{G/20} \text{ G(κέρδος σε db.) και } V_0 \text{ (κέρδος σε voltage)}$$

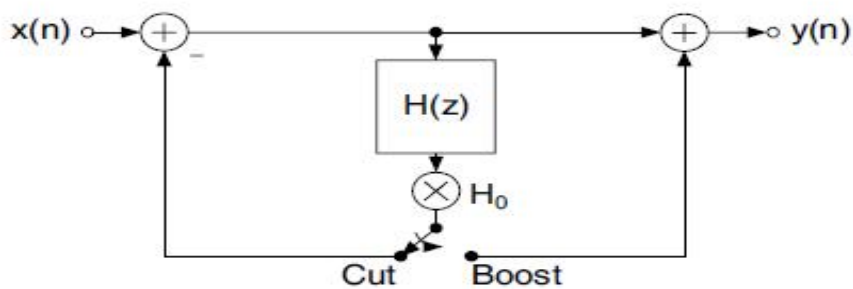
$$(4.3.1.2.4) V_0=H(s=0) \text{ δηλαδή } V_0 \text{ είναι η ενίσχυση στην θέση } \omega=0 \text{ ή } H(z=1)=V_0$$

έτσι μπορούμε να λύσουμε την συνάρτηση μεταφοράς για το H_0 και έτσι καθορίζετε η συμπεριφορά του, όταν το ω μεγαλώνει δηλαδή $\omega>\omega_c$ τότε $H(s=\infty) = 1$ ή $H(z=-1)=1$ voltage $1=0$ db.,εκεί που το lowpass φθίνει και τείνει προς το 1 voltage=0db.

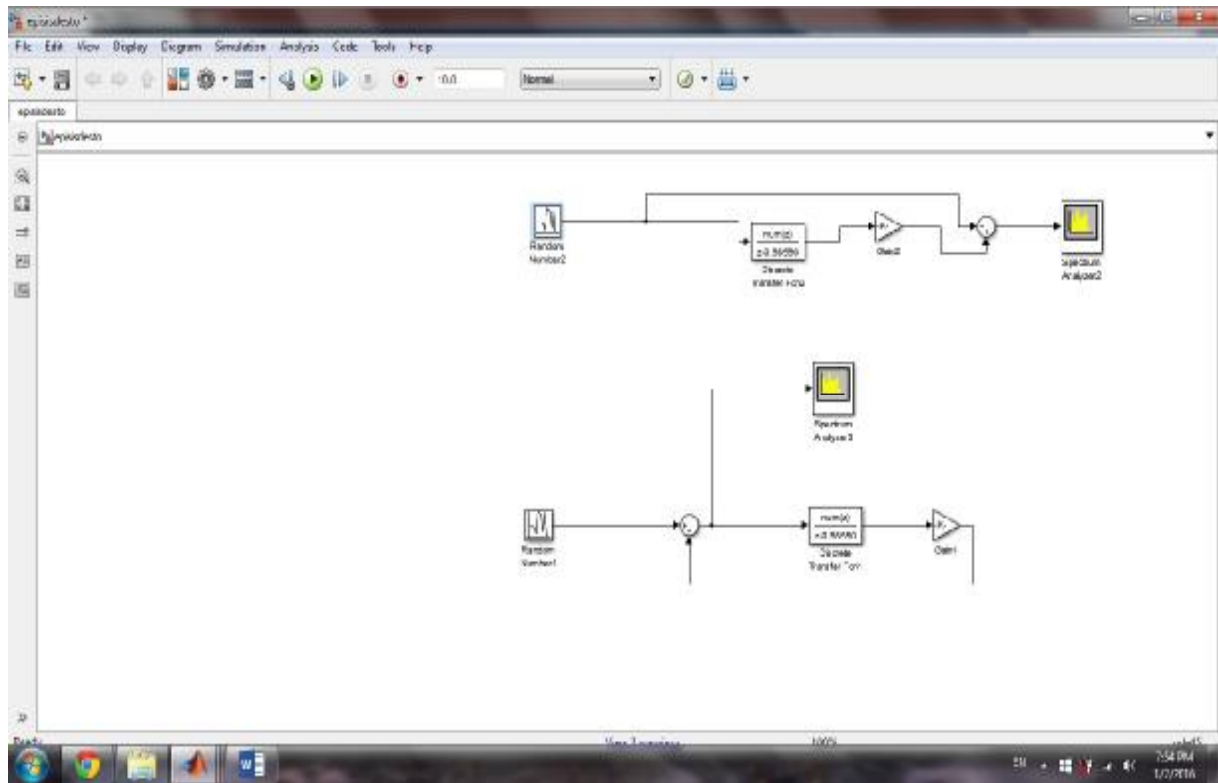
Για την περίπτωση cut(προφανώς άλλη απόκριση συχνότητάς του φίλτρου σε σχέση με την περίπτωση boost θα έχει η περίπτωση cut) Οπότε στην περίπτωση cut θα έχουμε διαφορετική συνάρτηση μεταφοράς σε σχέση με την περίπτωση boost για την ίδια συχνότητα αποκοπής. Πρακτικά είναι μια εναλλαγή των πολλών και το μηδενικών του φίλτρου η εξίσου μια εναλλαγή των συντελεστών της συνάρτησης μεταφοράς.

Πιο συγκεκριμένα για την περίπτωση cut η συνάρτηση μεταφοράς θα είναι η ανεστραμμένη συνάρτηση μεταφοράς της περίπτωσης boost δηλαδή η περίπτωση cut είναι ο "καθρέπτης" της περίπτωσης boost οπότε για να μπορέσουμε να αλλάξουμε η να ελέγξουμε το κέρδος ανεξάρτητα από τους άλλους συντελεστές του φίλτρου τότε η δομή που χρησιμοποιείτε είναι η FF και FB (Feed Forward για ενίσχυση και Feed Backward για εξασθένηση)δηλαδή από ότι καταλαβαίνουμε αν τροφοδοτήσουμε την έξοδο του φίλτρου ξανά στην είσοδο του τότε με αυτόν τον τρόπο πετυχαίνουμε την περίπτωση cut του φίλτρου. Έτσι ουσιαστικά αλλάζουν οι θέσεις των πόλων με τα μηδενικά(δηλαδή μια αντιστροφή της συνάρτησης μεταφοράς) και έτσι έχουμε την περίπτωση cut.Από εδώ καταλαβαίνουμε ότι το $V0=1+H0>1$ και στις δυο περιπτώσεις boost και cut. Προφανώς και στις δυο περιπτώσεις καταλαβαίνουμε ότι το κέρδος θα είναι ανεξάρτητο από την συνάρτηση μεταφοράς του φίλτρου (Udo Zolzer :2^η έκδοση 2008)

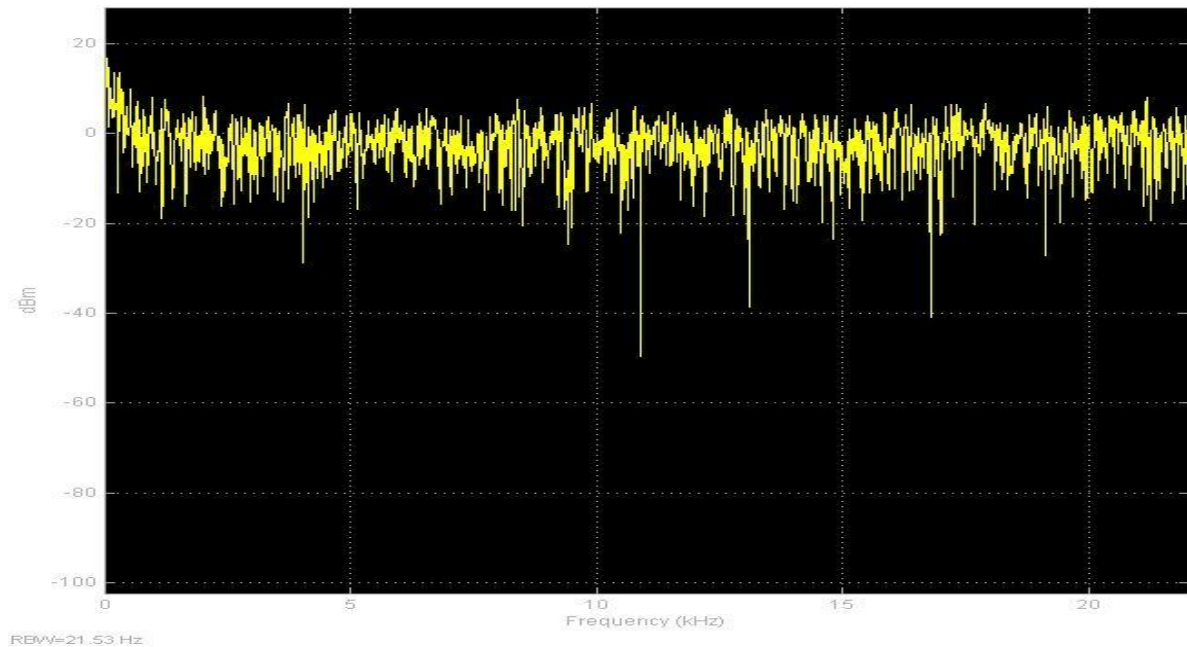
Κοιτώντας το σχήμα μιλάμε για μια αρνητική ανατροφοδότηση . Ουσιαστικά είναι η αρχική έξοδος του φίλτρου αφαιρείται από την είσοδο που εφαρμόζεται και αυτό που προκύπτει εφαρμόζεται ξανά σαν είσοδος στο φίλτρο . Βέβαια θα πρέπει να εξετάσουμε την ευστάθεια του ανεστραμμένου συστήματος δηλαδή για την περίπτωση cut .Αυτό επιτυγχάνεται με το να είναι ελάχιστης φάσης το φίλτρο της περίπτωσης boost οπότε η αντιστροφή του θα με δώσει επίσης ένα ευσταθές φίλτρο. Σίγουρα χρειάζεται περισσότερο έρευνα για την ανάδραση ενός ψηφιακού συστήματος αλλά εδώ ας κρατήσουμε ότι μας χρησιμεύει για να πάρουμε την περίπτωση cut κάνοντας μια αρνητική ανατροφοδότηση



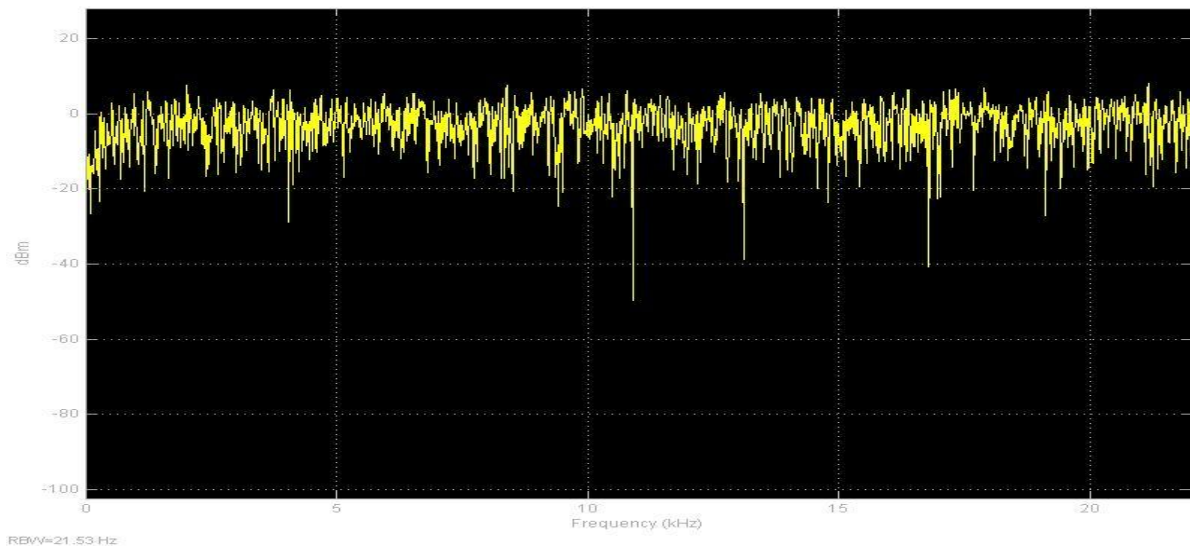
Εικόνα 4.10 Feed Forward Feed Backward μέθοδος για boost και cut



Εικόνα 4.11 Feed Forward Feed Backward μέθοδος στο Simulink lowshelving



Εικόνα 4.12 Απόκριση συχνότητας lowfrequency shelving boost



Εικόνα 4.13 Απόκριση συχνότητας lowfrequency shelving cut

Οπότε εάν $H(s)_{boost} = H(s) = 1 + H_0 \frac{\omega c}{s + \omega c}$ τότε και η $H(s)_{cut}$ θα είναι $\frac{1}{H(s)_{boost}}$

Δηλαδή $FF(z) = 1 + H(z) * H_0$ (4.3.1.2.5)

$FB = 1 / (1 + H(z) * H_0)$ (4.3.1.2.6)

δηλαδή $H(s)_{cut} = 1 + H_0 \frac{\omega c / V_0}{s + \omega c / V_0}$ (Udo Zolzer :2^η έκδοση 2008) (4.3.1.2.7)

εδώ το κέρδος για την cut περίπτωση επίσης θα είναι $v_0 > 1$ γιατί ουσιαστικά χρησιμοποιούμε το $V_0 = 1 + H_0$ της περίπτωσης boost. Δηλαδή εάν θέλαμε για την περίπτωση cut το κέρδος $Cut(G) = -6$ db τότε θα κάνουμε $Boost(G) = Cut(-G) = 6$ db. άρα πάλι $V_0 = 1 + H_0$ και μετα αντιστρέφουμε την συνάρτηση μεταφοράς.

Οπότε εφόσον χρησιμοποιούμε για την περίπτωση cut την ανεστραμμένη συνάρτηση μεταφοράς της περίπτωσης boost to V_0 και στις δυο περιπτώσεις $V_0 > 1$. Αλλιώς ορίζουμε το $V_0 < 1$ η αλλιώς $V_0_{cut} = 1 / V_0_{boost}$ αλλά τότε δεν θα έχουμε ανεστραμμένη την συναρτά μεταφοράς της περίπτωσης Boost.

4.3.1.3 Τρίτη μέθοδος σχεδιασμού low frequency shelving φίλτρου

Μια άλλη μέθοδος σχεδιασμού ενός low frequency shelving φίλτρου είναι αυτή με συνάρτηση μεταφοράς στο z πεδίο $H(z) = 1 + H_0 * 1/2 * (1 + A(z))$ (Udo Zolzer :2^η έκδοση 2008) (4.3.1.3.1) εκφράζοντας ένα low pass με ένα allpass. Όπως ξέρουμε η συμπεριφορά ενός low shelving φίλτρου καθορίζεται όταν $\omega = 0$ ή $H(z=1) = V_0$ $H(1) = 1 + H_0/2 * (1 + (1)) = 1 + H_0/2 * 2 = 1 + H_0$ και για $Z = -1$ τότε $H(-1) = 1 + H_0/2 * (1 + (-1)) = 1 + H_0/2 * 0 = 1$ voltage = 0 db. για τις ψηλές συχνότητες όσο πάμε προς το $fs/2$

Με την (4.3.1.3.1) περιγράφουμε το ίδιο φίλτρο σε σχέση με την (4.3.1.2.1). Ωστόσο τώρα το φίλτρο γίνεται τελείως-εντελώς παραμετρικό. Εδώ εφόσον μπορώ να εκφράσω ένα low pass μέσω ενός allpass(που με δίνει την δυνατότητα να αλλάζω την συχνότητα αποκοπής χωρίς να αλλάξω όλη την συνάρτηση μεταφοράς του φίλτρου αλλά υπολογίζοντας μόνο το α_B και α_C , επίσης το κέρδος

είναι ανεξάρτητο ούτως η αλλιώς από το low pass, οπότε εδώ έχω την δυνατότητα να αλλάξω δυο παραμέτρους (το κέρδος και την συχνότητα αποκοπής) και να υπολογίσω μόνο αυτούς τους δυο και όχι τους άλλους τρεις χωρίς φυσικά να αλλάξω όλη την συνάρτηση μεταφοράς. Στην (4.3.1.2.1) αν αλλάξω την συχνότητα αποκοπής θα άλλαζε όλη η συνάρτηση μεταφοράς του φίλτρου.

Οπότε στο s πεδίο (συνεχές πεδίο) ή κατά Laplace ένα low shelving φίλτρο μπορεί να γραφεί για την περίπτωση ενισχύσεις (boost)

$$H(s) = 1 + H_0/2 \left[1 + \frac{s - \omega_c}{s + \omega_c} \right] \quad (4.3.1.3.2)$$

$$\text{με } a_B(s) = \frac{s - \omega_c}{s + \omega_c} \quad (4.3.1.3.3)$$

(συνάρτηση μεταφοράς ολοπερατό πριν τον διγραμμικό μετασχηματισμό για περίπτωση boost)

$$\text{και στο πεδίο } z \text{ είναι } AB(z) = \frac{z^{-1} + ab}{1 + abz^{-1}} \quad (4.3.1.3.4)$$

$$\text{με } a_b = \frac{\tan\left(\pi \frac{f_c}{f_s}\right) - 1}{\tan\left(\pi \frac{f_c}{f_s}\right) + 1} \quad (4.3.1.3.5)$$

(η παράμετρος ελέγχου της συχνότητας αποκοπής για την περίπτωση της ενίσχυσης) οπότε εδώ αν για την περίπτωση boost θέλουμε να αλλάξουμε το ω_c τότε μπορούμε να το κάνουμε χωρίς να υπολογίσουμε όλους τους άλλους συντελεστές του φίλτρου αλλά μόνο ένα παράγοντα το a_b

$$\text{Για την περίπτωση cut } 1 + H_0/2 \left[1 + \frac{s - \omega_c/V_0}{s + \omega_c/V_0} \right] \quad (4.3.1.3.6)$$

$$\text{με } a_c(s) = \frac{s - \omega_c/V_0}{s + \omega_c/V_0} \quad (4.3.1.3.7)$$

(συνάρτηση μεταφοράς ολοπερατό πριν τον διγραμμικό μετασχηματισμό για περίπτωση cut) και μετά τον διγραμμικό

$$a_c(z) = \frac{z^{-1} + ac}{1 + acz^{-1}} \quad (4.3.1.3.8)$$

$$\text{με } a_C = \frac{\tan\left(\pi \frac{f_c}{f_s}\right) - V_0}{\tan\left(\pi \frac{f_c}{f_s}\right) + V_0} \quad (4.3.1.3.9)$$

(η παράμετρος ελέγχου της συχνότητάς αποκοπής για την περίπτωση της εξασθένησης cut).

(Udo Zolzer :2^η έκδοση 2008)

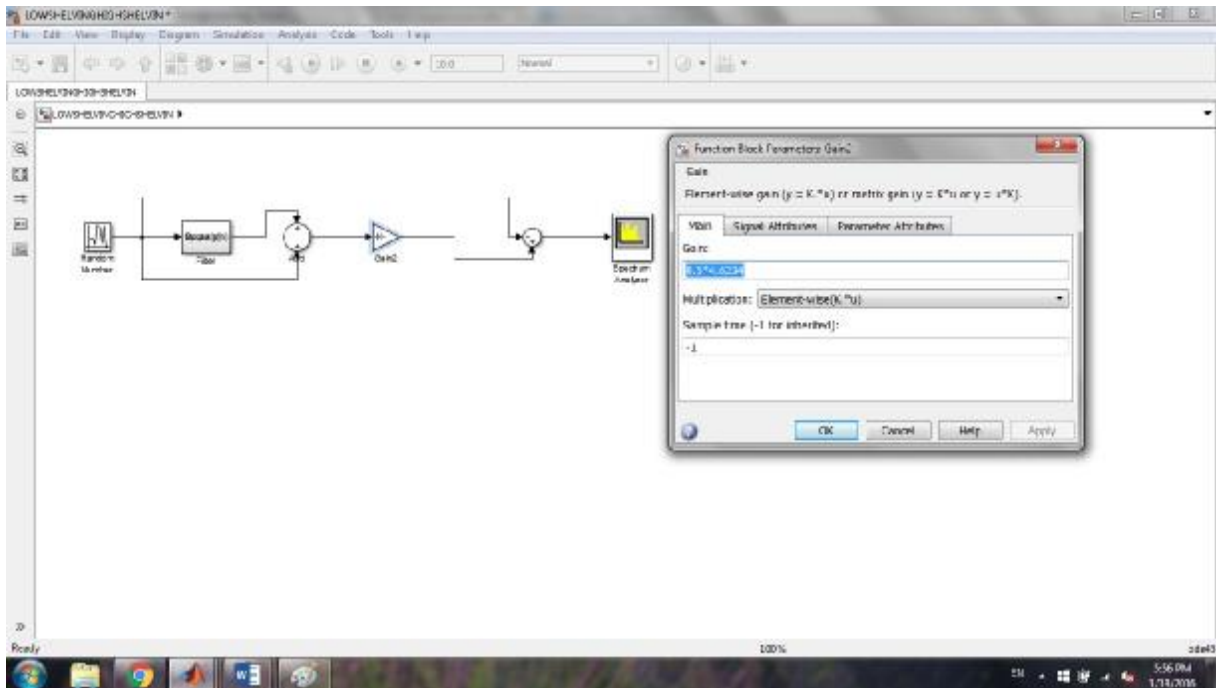
Οπότε εδώ αν για την περίπτωση cut θέλουμε να αλλάξουμε το f_c τότε μπορούμε να το κάνουμε χωρίς να υπολογίσουμε όλους τους άλλους συντελεστές του φίλτρου αλλά μόνο ένα παράγοντα το a_c . Εδώ πρέπει να πούμε ότι υπάρχει ένα πρόβλημα με την περίπτωση cut. Το κέρδος πάνω και κάτω από την συχνότητα αποκοπής στην περίπτωση boost δεν είναι ίδια με την περίπτωση cut. Δηλαδή εάν για την περίπτωση boost το μέγιστο κέρδος είναι στην συχνότητα αποκοπής (10 log(0.5) = -3db. και για την περίπτωση cut εφόσον αυτή είναι η αντιστροφή της περίπτωσης boost τότε για cut $G_{ut} = +3db$. και αλλά το low pass θα έχει συχνότητα αποκοπής πάλι -3db. δηλαδή για την περίπτωση boost η συχνότητα αποκοπής θα είναι ίδια με του low pass (-3 db) αλλά για περίπτωση cut το low pass θα με αλλάξει το κέρδος της συχνότητα αποκοπής του low shelving φίλτρου (+3db)

Για αυτό για να έχω ίδια συχνότητα αποκοπής και στις δυο περιπτώσεις (boost, cut) για low shelving φίλτρο θα πρέπει να υπολογίσω την παράμετρο της συχνότητάς αποκοπής για cut βάζοντας και το κέρδος (V_0)

Τέλος όσο αναφορά τον σχεδιασμό και όπως βλέπουμε και στον κώδικα για την υλοποίηση του φίλτρου (απόκριση συχνότητας) χρειαζόμαστε ένα σήμα εισόδου $x = [\text{zeros}(1,0), 1, \text{zeros}(1,22050)]$; Είναι η διακριτή ώση που έχει πεπερασμένο μήκος άρα και η κρουστική του φίλτρου θα έχει πεπερασμένη διάρκεια. Άρα με την (5.3.1.3.1) θα με δώσει ένα FIR φίλτρο δηλαδή μόνο μηδενικά. Οπότε θα χρησιμοποιήσουμε την συνάρτηση μεταφοράς για πρώτης τάξης low shelving φίλτρο αποτελούμενη από μηδενικά και πόλους (IIR) και αυτή θα είναι

$$\frac{1 + (1 + abc)H_0/2 + (abc + (1 + abc) \frac{H_0}{2})z^{-1}}{1 + abc z^{-1}} = H(z) \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1}} = \frac{1.308 + 0.558z^{-1}}{1 - 0.866z^{-1}} \quad H(z) = \frac{bz}{az} \quad (4.3.1.3.10)$$

Για την περίπτωση $\text{cut } V0_{\text{cut}}=10^{-(G/20)}$ και χρησιμοποιούμε το a_C (Udo Zolzer :2^η έκδοση 2008)

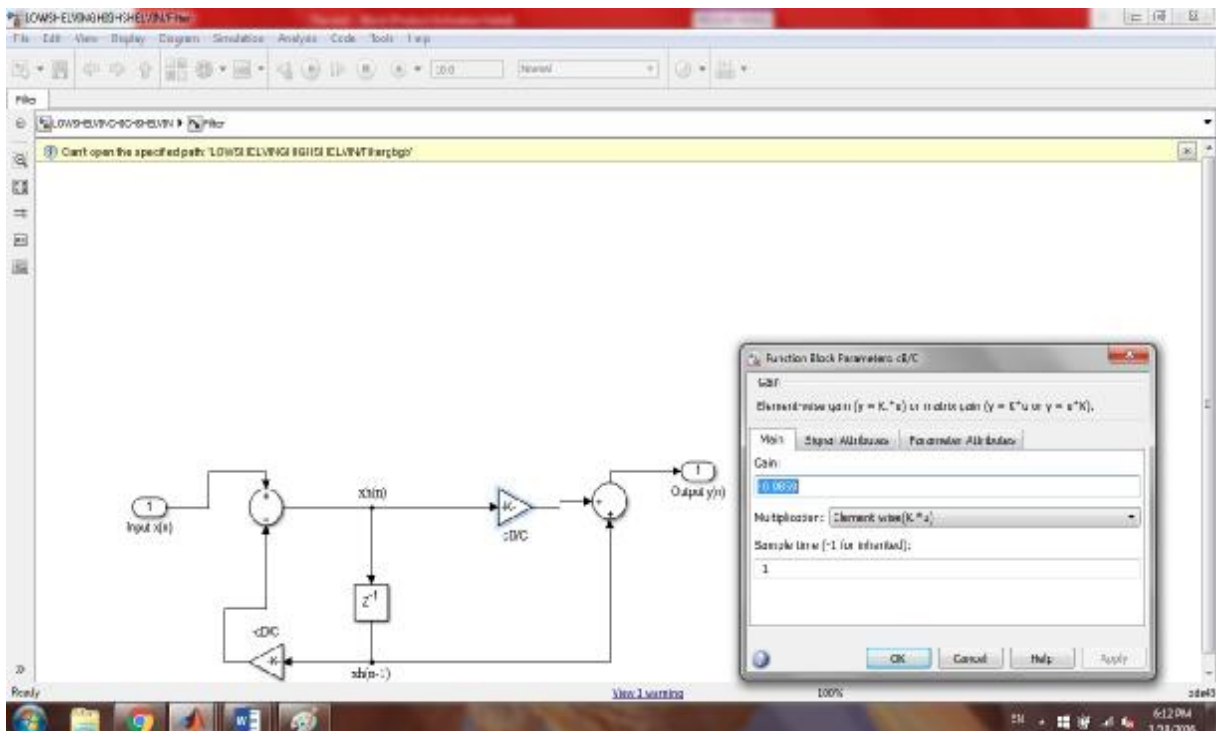


Εικόνα 4.14 Low frequency shelving φίλτρο $y(n)=0.5*H0*(x(n)+A(z))+x(n)$

Η συνάρτηση μεταφοράς 1^{ης} τάξης ενός allpass είναι

$$A(z) = \frac{z^{-1} + a_B/C}{1 + a_B/Cz^{-1}} \quad (4.3.1.3.10)$$

Η εξίσωση διαφορών (στο χρόνο) είναι: $xh(n)=x(n)-aB/Cxh(n-1)$ και η τελική έξοδος θα είναι $y1(n)=a_B/Cxh(n)+xh(n-1)$. Προφανώς $A(z)=y1(n)$



Εικόνα 4.15 Πρώτης τάξης allpass φίλτρου

Ωστόσο μια εναλλακτική μοντελοποίηση ενός full parametric φίλτρου εκμεταλλευόμενοι τις διαφορετικές υλοποιήσεις που με δίνει το ολοπερατό φίλτρο λόγω της μετατόπισης φάσης έτσι ώστε να έχω την shelving συμπεριφορά και για low/high shelving για την περίπτωση boost (4.3.1.3.5) και για την περίπτωση cut θα είναι (4.3.1.3.9)

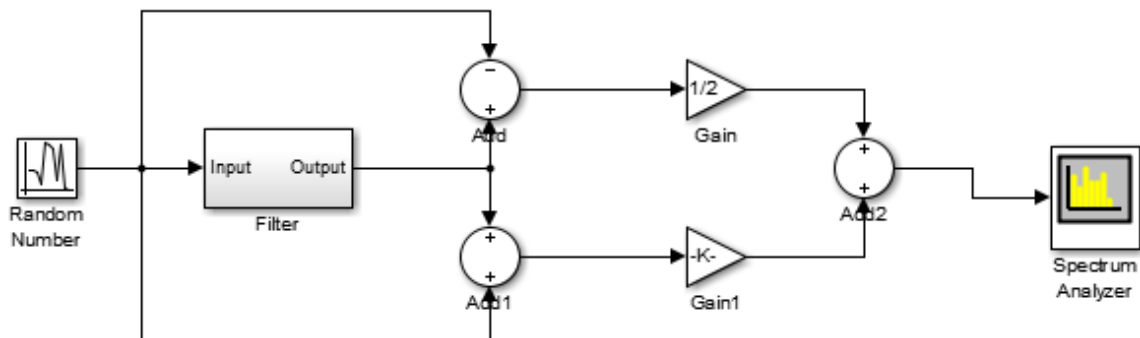
Η συνολική συνάρτηση μεταφοράς του συστήματος θα είναι

$$H_z = \frac{1}{2}[1 - Az] + \frac{V_0}{2}[1 + Az] \quad (4.3.1.3.11)$$

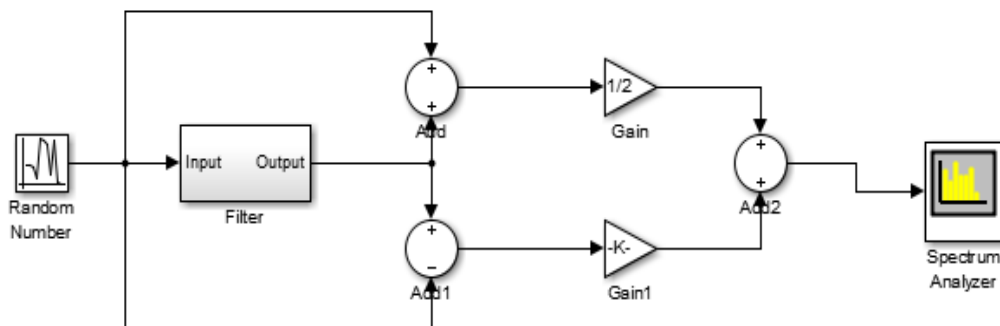
Δηλαδή $\frac{1}{2}[1 - Az]$ είναι ένα highpass που υλοποιείτε με ένα allpass και $\frac{V_0}{2}[1 + Az]$ είναι ένα lowpass

Το $V_0/2$ (Gain 1) στο σχήμα ρυθμίζει το κέρδος και των δυο φίλτρων (low/high shelving)

(PHILLIP A. REGALIA & SANJIT K. MITRA 1987)



Εικόνα 4.16 lowshelving cut



Εικόνα 4.17 highshelving cut

4.3.2 High Frequency Shelving φίλτρο

4.3.2.1 Πρώτη μέθοδος(απευθείας μέθοδος σχεδιασμού)

Η αρχική συνάρτηση μεταφοράς (boost) ενός high shelving φίλτρου είναι

$$H(s) = \frac{sV_0 + \omega_c}{s + \omega_c} \quad (4.3.2.1)$$

και για cut είναι $H(s) = \frac{s + \omega_c}{s/V_0 + \omega_c}$ (Udo Zolzer :2^η έκδοση 2008)

(4.3.2.2)

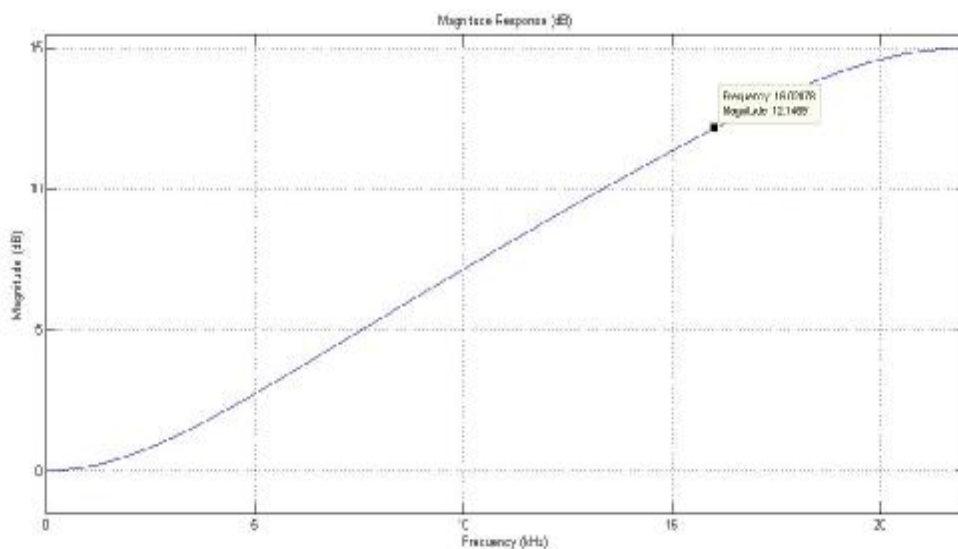
Καταλαβαίνουμε από τις παραπάνω εξισώσεις ότι το high frequency shelving φίλτρο προκύπτει από ένα high pass $\frac{s}{s + \omega_c}$ με κέρδος πάνω από την συχνότητα $\omega = \pi$ και όσο τείνει προς το 0 φθίνει. Πάνω από την συχνότητα αποκοπής ω_c κατά $-3d$. Για την περίπτωση cut ισχύει ότι και το low shelving. Επίσης και εδώ η ευστάθειά και για την περίπτωση cut μιας και αυτή είναι η ανεστραμμένη περίπτωση boost, υφίσταται μόνο εάν το φίλτρο της περίπτωσης boost είναι ελάχιστης φάσης δηλαδή ($ak < 1$) αλλιώς το φίλτρο για την περίπτωση cut θα είναι ασταθές

Μετά τον διγραμμικό μετασχηματισμό το 1^{ης} τάξης φίλτρο περιγράφεται στο πεδίο της συχνότητάς με την συνάρτηση μεταφοράς $H(z) = \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1}}$ και η δομή υλοποίησης του φίλτρου είναι εξίσου η ίδια με την εικόνα 4.9.

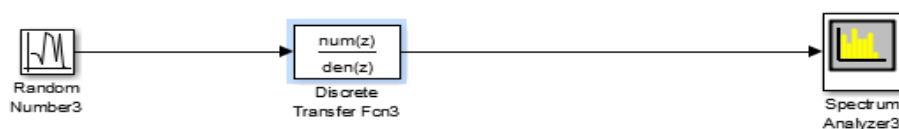
Ο κώδικας για την υλοποίηση 1^{ης} τάξης high shelving φίλτρου που προκύπτει από την (4.3.2.1) για την περίπτωση boost και από την (4.3.2.2) για cut είναι ο (κώδικας 2 παράρτημα).

Εδώ το κέρδος για boost είναι $V_0 > 1$ και για cut ορίζετε $V_0 \text{cut} < 1$

$$V_0 = 10^{G/20} \quad V_0 \text{cut} = 10^{-G/20} \quad (\text{Udo Zolzer :2^η έκδοση 2008}) \quad (4.3.2.3)$$



Εικόνα 4.18 High frequency shelving φίλτρο με $f_c = 16000$



Εικόνα 4.19 Μοντελοποίηση στο Simulink high frequency shelving φίλτρο

4.3.2.2 Δεύτερη μέθοδος σχεδιασμού high frequency shelving φίλτρου(ημιπαραμετρική)

Για ένα πρώτης τάξης high shelving φίλτρο η συνάρτηση μεταφοράς είναι

$$H(s)=1+H_0\frac{s}{s+\omega c} \quad (4.3.2.2.1)$$

Από την εξίσωση (4.3.2.2.1) καταλαβαίνουμε ότι ένα high frequency shelving φίλτρο προκύπτει από την έξοδο ενός high pass φίλτρο (το οποίο πολλαπλασιάζεται- σταθμίζεται με μια σταθερά=κέρδος ενός high pass) +direct path

Η (4.3.2.2.1) είναι ιδίες(ισοδύναμες) με την (4.3.2.1) αλλά στην πρώτη εφαρμόζεται η ημιπαραμετρική δομή του φίλτρου. Ισχύει ότι ισχύει για την ημιπαραμετρική μέθοδο για low shelving φίλτρο.

Direct path: είναι το 1 μπροστά από το H0. Ουσιαστικά είναι το αρχικό αυθεντικό σήμα εισόδου το οποίο αντιγράφεται στην έξοδο ενός high pass φίλτρου. Εάν προσθέσουμε στο direct path+high pass φίλτρο τότε θα πάρουμε ένα σήμα στην έξοδο στο οποίο υπάρχει ενίσχυση στις ψηλές συχνότητες με $V_0=1+H_0$

(4.3.2.2.2) Όπου $V_0=1+H_0$ (η ενίσχυση για τις ψηλές συχνότητες)
 (4.3.2.2.3) $H_0=V_0-1$

(4.3.2.2.4) $V_0=10^{G/20}$ G(κέρδος σε db.) και V_0 (κέρδος σε voltage)

(4.3.2.2.5) (Udo Zolzer :2^η έκδοση 2008)

όταν $V_0=H(s=\infty)$ η $\omega=\pi$ η $H(z=-1)=V_0$ και $H(z=1)=1$ voltage 1=0 db. δηλαδή έχει μεταβλητό κέρδος όσο η συχνότητα τείνει προς το ∞ και φθινηή όσο η συχνότητα πάει προς το μηδέν δηλαδή 0 db.

Για την περίπτωση cut ισχύει ότι ισχύει για το low frequency shelving φίλτρο δηλαδή εάν $H(s)_{boost}=1+H_0\frac{s}{s+\omega c}$ τότε για την περίπτωση cut για high shelving θα έχουμε

$H(s)_{cut}=\frac{1}{H(s)_{boost}}$ δηλαδή τότε $H(s)=1+H_0\frac{s}{s+v_0\omega c}$ οπότε και εδώ $V_0>1$. (4.3.2.2.6)

Κοιτώντας το σχημα 4.18

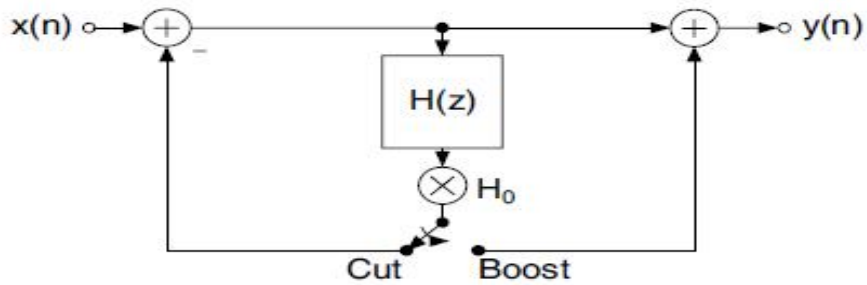
(4.3.1.2.7) $FF(z) = 1+H(z)*H_0$

(4.3.1.2.7) $FB=1/(1+H(z)*H_0)$ (Udo Zolzer :2^η έκδοση 2008)

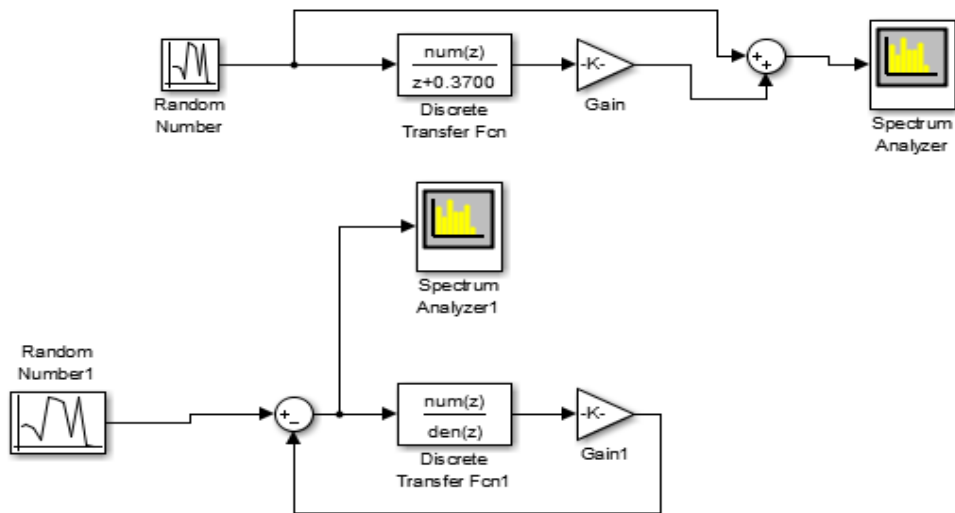
Δηλαδή εάν θέλαμε για high frequency shelving , για την περίπτωση cut to κέρδος $Cut(G)=-6db$ τότε θα κάνουμε $Boost(G)=Cut(-G)=6 db$. άρα πάλι $V_0=1+H_0$ και μετά αντιστρέφουμε την συνάρτηση μεταφοράς.Οπότε εφόσον χρησιμοποιούμε για την περίπτωση cut την ανεστραμμένη συνάρτηση μεταφοράς την περίπτωσης boost to V_0 και στις δυο περιπτώσεις $V_0>1$. Σε διαφορετική περίπτωση ορίζουμε το $V_0<1$ η αλλιώς $V_0_{cut}=1/V_0_{boost}$ αλλά τότε δεν θα έχουμε ανεστραμμένη την συνάρτηση μεταφοράς της περίπτωσης boost.

Πιο συγκεκριμένα για την περίπτωση cut η συνάρτηση μεταφοράς θα είναι η ανεστραμμένη συνάρτηση μεταφοράς της περίπτωσης boost δηλαδή η περίπτωση cut είναι ο "καθρέπτης" της περίπτωσης boost οπότε για να μπορέσουμε να αλλάξουμε η να ελέγξουμε το κέρδος ανεξάρτητα από τους άλλους συντελεστές του φίλτρου τότε η δομή που χρησιμοποιείτε είναι η FF και FB (Feed Forward για ενίσχυση και Feed Backward για εξασθένηση)δηλαδή από ότι καταλαβαίνουμε αν τροφοδοτήσουμε την έξοδο του φίλτρου ξανά στην είσοδο του ,τότε με αυτόν τον τρόπο πετυχαίνουμε την περίπτωση cut του φίλτρου. Έτσι ουσιαστικά αλλάζουν οι θέσεις των πόλων με τα μηδενικά (δηλαδή μια αντιστροφή της συνάρτησης μεταφοράς) και έτσι έχουμε την περίπτωση cut. Από εδώ καταλαβαίνουμε ότι το $V_0=1+H_0>1$ και στις δυο περιπτώσεις boost και cut. Προφανώς

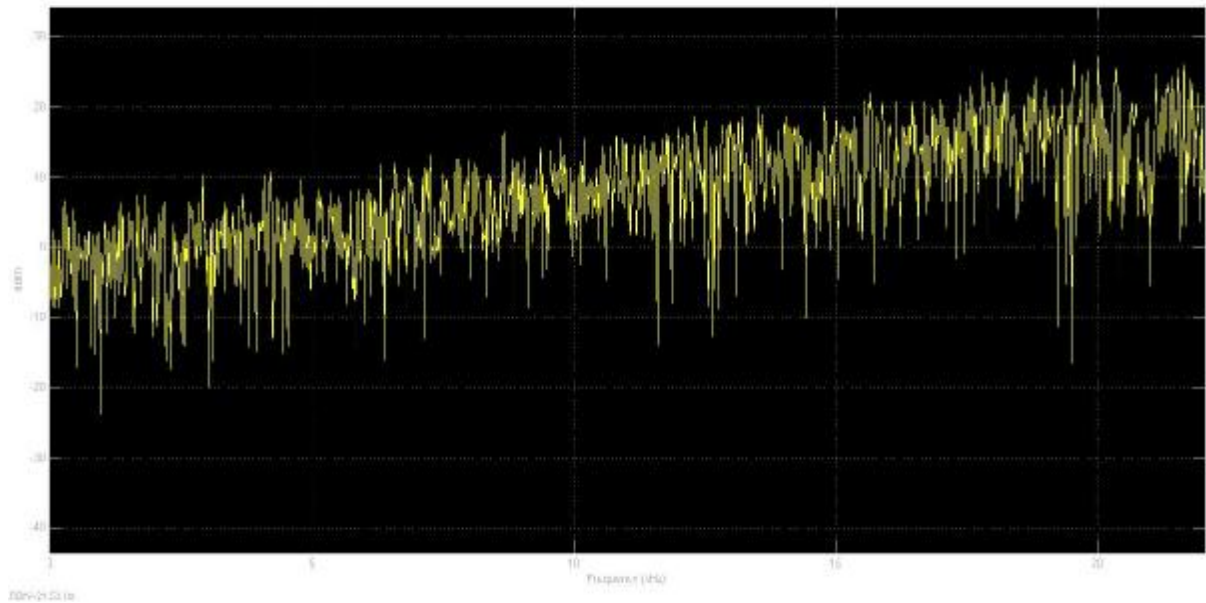
και στις δυο περιπτώσεις καταλαβαίνουμε ότι το κέρδος θα είναι ανεξάρτητο από την συνάρτηση μεταφοράς του φίλτρου



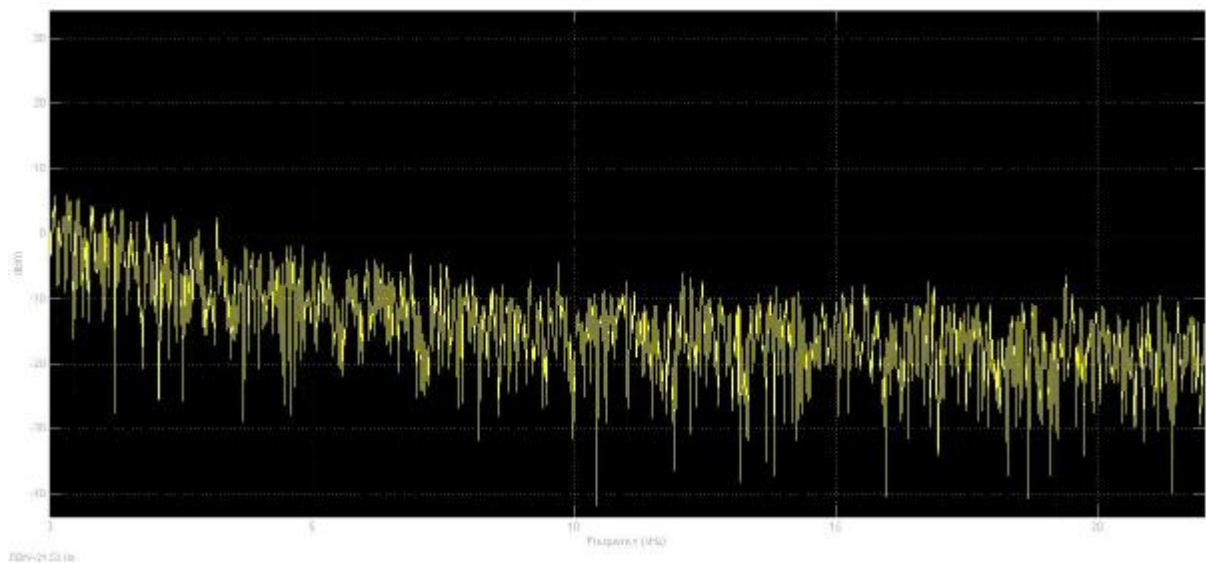
Εικόνα 4.20 Feed Forward Feed Backward μέθοδος για boost και cut



Εικόνα 4.21 Feed Forward Feed Backward μέθοδος στο Simulink highshelving



Εικόνα 4.22 Απόκριση συχνότητας highfrequency shelving boost



Εικόνα 4.23 Απόκριση συχνότητας highfrequency shelving cut

4.3.2.3 Τρίτη μέθοδος σχεδιασμού high frequency shelving φίλτρου

Μια άλλη μέθοδος σχεδιασμού ενός high frequency shelving φίλτρου είναι αυτή με συνάρτηση μεταφοράς στο z πεδίο

$$H(z)=1+H_0*1/2*(1-A(z)) \quad (\text{Udo Zolzer :2}^{\text{η}} \text{ εκδοση 2008}) \quad (4.3.2.3.1)$$

εκφράζοντας ένα low pass με ένα allpass . Και εδώ η συμπεριφορά ενός highshelving φίλτρου καθορίζεται όταν $\omega=\pi$ ή $H(z=-1)=V_0$ $H(-1)=1+H_0/2*(1-(-1))=1+H_0/2*2=1+H_0$ και για $Z=1$ τότε $H(1)=1+\text{voltage}= 0 \text{ db.}$ για τις χαμηλές συχνότητες όσο πάμε προς το 0. Με την (5.3.2.3.1) περιγράφουμε το ίδιο φίλτρο σε σχέση με την (5.3.2.1) αλλά με διαφορετικό σχεδιασμό .

Εδώ το φίλτρο γίνεται τελείως-εντελώς παραμετρικό. Εδώ εφόσον μπορώ να εκφράσω ένα high pass μέσω ενός allpass(που δίνει την δυνατότητα να αλλάζω την συχνότητα αποκοπής χωρίς να αλλάξω όλη την συνάρτηση μεταφοράς του φίλτρου αλλά υπολογίζοντας μόνο το a_B και a_C)

επίσης το κέρδος είναι ανεξάρτητο ούτως η αλλιώς από το high pass, οπότε εδώ έχω την δυνατότητα να αλλάξω δυο παραμέτρους (το κέρδος V_0 και την συχνότητα αποκοπής ω_c) και να υπολογίσω μόνο αυτούς τους δυο και όχι τους άλλους τρεις χωρίς φυσικά να αλλάξω όλη την 2^{ης} τάξης συνάρτηση μεταφοράς φίλτρου IIR. Στην (5.3.2.1) αν αλλάζα την συχνότητα αποκοπής θα άλλαζε όλη η συνάρτηση μεταφοράς του φίλτρου. Οπότε στο s πεδίο (συνεχές πεδίο) ή κατά Laplace ένα low shelving φίλτρο μπορεί να γραφεί για την περίπτωση ενίσχυσης (boost)

$$H(s) = 1 + H_0/2 \left[1 - \frac{s - \omega_c}{s + \omega_c} \right] \quad (4.3.2.3.2)$$

εκφράζοντας ένα high pass μέσω ενός allpass. Το $H_0/2$ προκύπτει από το $1/2$ το οποίο έχει το high pass φίλτρο μπροστά του και έτσι ουσιαστικά το H_0 γίνεται $H_0/2$ δηλαδή η εξίσωση (5.3.2.3.3) μπορεί να διαβαστεί

$$H(z) = 1 + H_0 \cdot 1/2 \cdot (1 - A(z)) \quad \text{με} \quad A(s) = \frac{s - \omega_c}{s + \omega_c} \quad (4.3.2.3.4)$$

(συνάρτηση μεταφοράς ολοπερατό πριν τον διγραμμικό μετασχηματισμό) και μετά τον διγραμμικό

$$A(z) = \frac{z^{-1} + ab}{1 + abz^{-1}} \quad (4.3.2.3.5) \quad \text{με} \quad ab = \frac{\tan\left(\pi \frac{f_c}{f_s}\right) - 1}{\tan\left(\pi \frac{f_c}{f_s}\right) + 1} \quad (4.3.2.3.6)$$

(η παράμετρος ελέγχου της συχνότητάς για την περίπτωση της ενίσχυσης (boost) (Udo Zolzer : 2^η έκδοση 2008)

Οπότε εδώ αν για την περίπτωση boost θέλουμε να αλλάξουμε το f_c τότε μπορούμε να το κάνουμε χωρίς να υπολογίσουμε όλους τους άλλους συντελεστές του φίλτρου αλλά μόνο ένα παράγοντα ab

Για την περίπτωση cut

$$1 + H_0/2 \left[1 + \frac{s - V_0 \omega_c}{s + V_0 \omega_c} \right] \quad (4.3.2.3.7)$$

$$\text{με} \quad ac(s) = \frac{s - V_0 \omega_c}{s + \omega_c / V_0} \quad (4.3.2.3.8)$$

(συνάρτηση μεταφοράς ολοπερατό πριν τον διγραμμικό μετασχηματισμό για περίπτωση cut) και μετά τον διγραμμικό

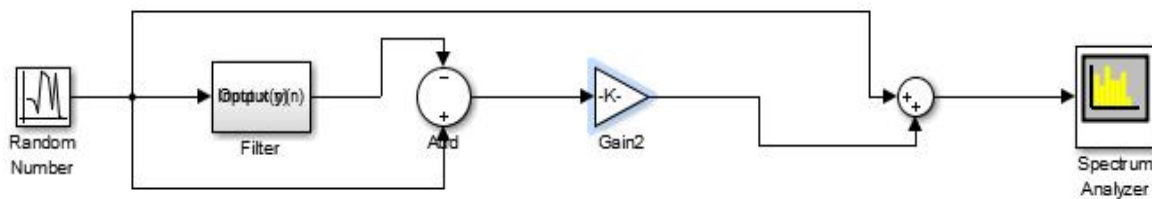
$$ac(z) = \frac{z^{-1} + ac}{1 + acz^{-1}} \quad (4.3.2.3.9) \quad \text{με} \quad ac(s) = \frac{V_0 \tan\left(\pi \frac{f_c}{f_s}\right) - 1}{V_0 \tan\left(\pi \frac{f_c}{f_s}\right) + 1} \quad (4.3.2.3.10)$$

(η παράμετρος ελέγχου της συχνότητάς αποκοπής για την περίπτωση της εξασθένισης cut). Για τον ίδιο λόγο με την περίπτωση cut του low shelving φίλτρου και εδώ θα πρέπει να υπολογίσω την παράμετρο της συχνότητάς αποκοπής για cut βάζοντας και το κέρδος (V_0) έτσι ώστε να έχουμε την ίδια γωνιακή συχνότητα αποκοπής για boost και για cut.

Τέλος όσο αναφορά και στον σχεδιασμό high shelving και όπως βλέπουμε και στον κώδικα θα χρησιμοποιήσουμε την συνάρτηση μεταφοράς για πρώτης τάξης high shelving φίλτρο αποτελούμενη από μηδενικά και πόλους (IIR) για την περίπτωση boost και αυτή θα είναι

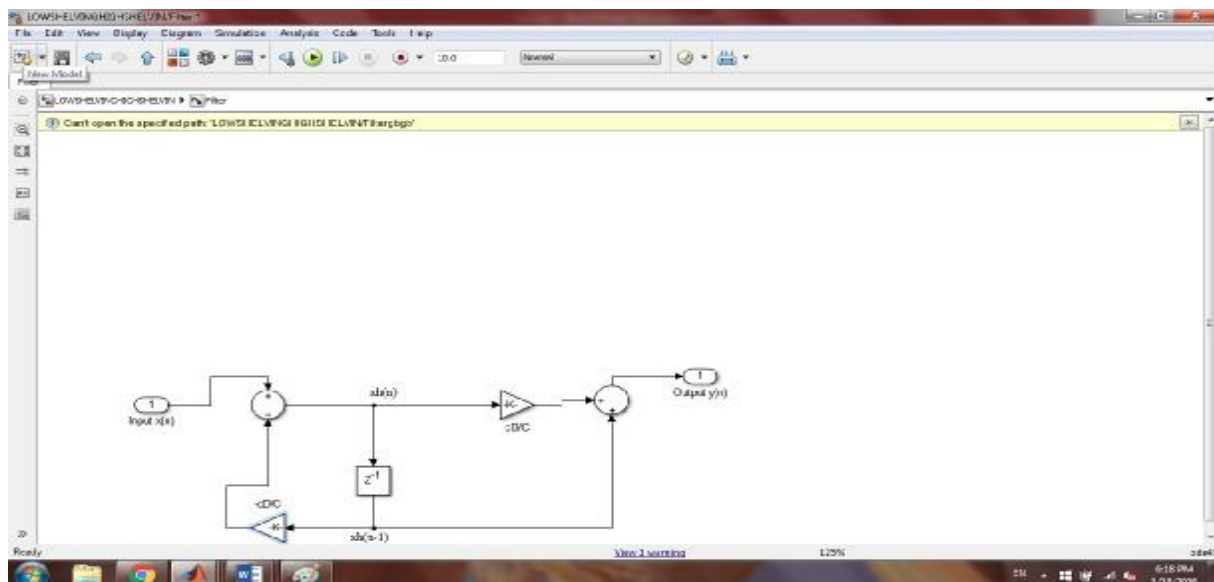
$$\frac{1 + (1 - abc)H_0/2 + (abc + abc - 1)H_0/2 z^{-1}}{1 + abc z^{-1}} = H(z) \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1}} = \frac{2.45 - 1.0863 z^{-1}}{1 + 0.37 z^{-1}} \quad H(z) = \frac{bz}{az} \quad (4.3.2.3.11)$$

Για την περίπτωση cut $V_0 \text{cut} = 10^{-(G/20)}$ και χρησιμοποιούμε το a_c (Udo Zolzer : 2^η έκδοση 2008)



Εικόνα 4.24 High frequency shelving filter $y(n)=0.5*H0*(xn-A(z))+x(n)$

Η συνάρτηση μεταφοράς 1ης τάξης ενός allpass είναι $A(z)=\frac{z^{-1}+a_B/C}{1+a_B/Cz^{-1}}$
 Η εξίσωση διαφορών (στο χρόνο) είναι: $xh(n)=x(n)-a_B/Cxh(n-1)$ και η τελική έξοδος θα είναι $y1(n)=a_B/Cxh(n)+xh(n-1)$. Προφανώς $A(z)=y1(n)$



Εικόνα 4.25 Πρώτης τάξης allpass φίλτρου

4.4 Peaking/Notch

Όπως καταλαβαίνουμε από τα σχήματα ένα bandpass είναι το ίδιο σχηματικά με ένα peaking Ωστόσο ένα bandpass επηρεάζει συνήθως εξασθενεί, εφόσον μιλάμε για bandpass με μέγιστο κέρδος 0 db και τις συχνότητες που βρίσκονται έξω από το εύρος του φίλτρου, ενώ ένα peaking τις συχνότητες έξω από το φίλτρο τις αφήνει ανεπηρέαστες

Τα peaking η τα notch φίλτρα είναι ισοσταθμιστές-φίλτρα τα οποία χρησιμοποιούνται όταν θέλουμε να στοχεύσουμε σε μια συγκεκριμένη κεντρική συχνότητα η καλύτερα σε ένα συγκεκριμένο εύρος συχνοτήτων με μεγάλη ακρίβεια. Και οι δυο ισοσταθμιστές peaking/notch μπορούν να έχουν μεταβλητό κέρδος \pm gain σε db. (boost η cut αντίστοιχα) και αυτό σημαίνει θετικό κέρδος =peaking αρνητικό =notch. Το εύρος ζώνης(Band Width ή BW) καθορίζετε από το $Q=Quality$, με το Q μπορούμε να ελέγξουμε το εύρος ζώνης του φίλτρου όπου $Q=F0/BW$

.Γενικά μεγάλο Q στενό εύρος ζώνης, το χρησιμοποιούμε όταν θέλουμε μια συχνότητα να την κάνουμε πιο ηχηρή ,μικρό Q μεγάλο εύρος ζώνης το χρησιμοποιούμε όταν θέλουμε να κάνουμε τον ήχο πιο φυσικό. Επίσης Για ενίσχυση(boost) θέλουμε ένα μικρό Q , για εξασθένηση(cut) θέλουμε ένα μεγάλο Q . Το εύρος ζώνης μπούμε να το βρούμε $BW=f_2-f_1$ δηλαδή η αφαίρεση της μεγαλύτερης ακραίας συχνότητάς μείον την μικρότερη ακραία συχνότητα και την κεντρική συχνότητα $f_0=\sqrt{f_1f_2}$ σε Hz(cycles/sec) κύκλους ανά δευτερόλεπτο.Το notch είναι το αντίστροφο του peaking ο καθρέπτης του ουσιαστικά και λόγω της ελάχιστης φάσης του peak φίλτρου αντιστροφή του(η εναλλαγή των θέσεων των πόλων και των μηδενικών του)θα με δώσει εξίσου ένα ευσταθές notch φίλτρο.

4.4.1 Πρώτη μέθοδος peaking φίλτρο(απευθείας μέθοδος σχεδιασμού)

Ο σχεδιασμός ενός peak φίλτρου έχει ως βάση του ένα bandpass.Προφανώς μιας και μιλάμε για peak φίλτρο θα πρέπει να συμπεριλάβουμε στον αρχικό σχεδιασμό του bandpass και το κέρδος πάνω από την συχνότητα αποκοπής. Οπότε εάν η συνάρτηση μεταφοράς ενός bandpass στο s πεδίο

είναι
$$H(s) = \frac{\frac{w_c}{q}s}{s^2 + \frac{w_c}{q}s + \omega_c^2}$$
 (4.4.1.1)

οπότε ένα peaking φίλτρο θα είναι
$$H(s) = \frac{s^2 \frac{w_c + v_0}{q} + \omega_c^2}{s^2 \frac{w_c}{q} + \omega_c^2}$$
 (Udo Zolzer :2^η εκδοση 2008) (4.4.1.2)

Εδώ βλέπουμε ότι για να πέτυχουμε το peaking φίλτρο διαιρούμε το κέρδος V_0 με το $q(V_0/q)$ και έτσι ορίζεται η μέγιστη ενίσχυση πάνω στην ω_0 (κεντρική συχνότητα). Εδώ ορίζουμε και το q το οποίο καθορίζει το εύρος ζώνης του φίλτρου.

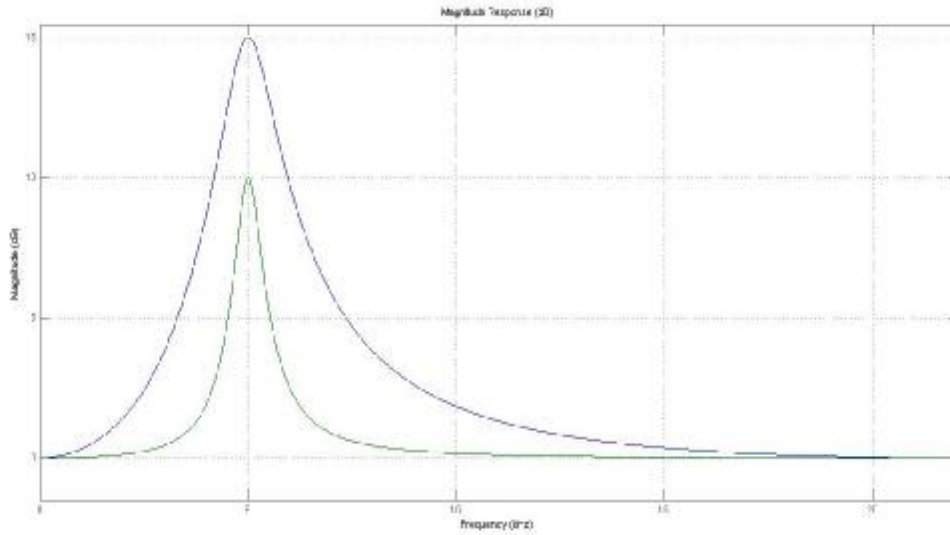
Η συνάρτηση μεταφοράς της περίπτωσης cut είναι η αντιστροφή της συνάρτησης μεταφοράς της περίπτωσης boost (notch φίλτρο) (5.4.1.2) δηλαδή είναι μια εναλλαγή των θέσεων των πολλών και των μηδενικών και έτσι θα πάρουμε ένα notch φίλτρο $V_0cut=10^{-G/20}$ με την ίδια

κεντρική συχνότητα του peak
$$H(s) = \frac{s^2 \frac{\omega_c}{q} + \omega_c^2}{s^2 \frac{w_c/V_0}{q} + \omega_c^2}$$
 (4.4.1.3)

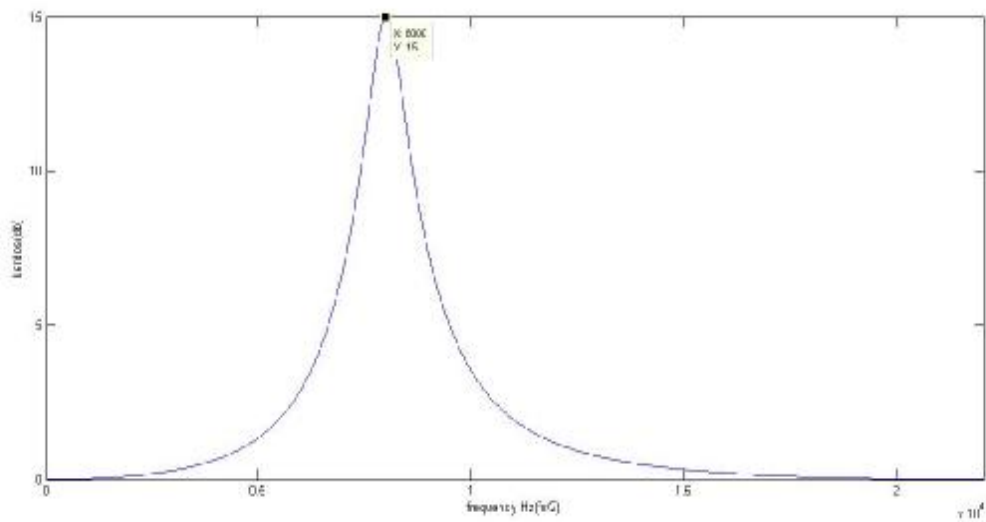
Ο κώδικας για την υλοποίηση 2^{ης} τάξης peaking φίλτρου που προκύπτει από την (4.4.1.2) είναι(στο παράρτημα κώδικας 3)

Μετά τον διγραμμικό μετασχηματισμό το 2^{ης} τάξης φίλτρου περιγράφεται στο πεδίο της συχνότητάς με την συνάρτηση μεταφοράς
$$H(z) = \frac{b_0 + b_1z^{-1} + b_2z^{-2}}{1 + a_1z^{-1} + a_2z^{-2}}$$
 (4.4.1.4)

Και στο χρόνο με μια εξίσωση διαφορών $y(n) = b_0x(n) + b_1x(n-1) + b_2x(n-2) - a_1y(n-1) - a_2y(n-2)$

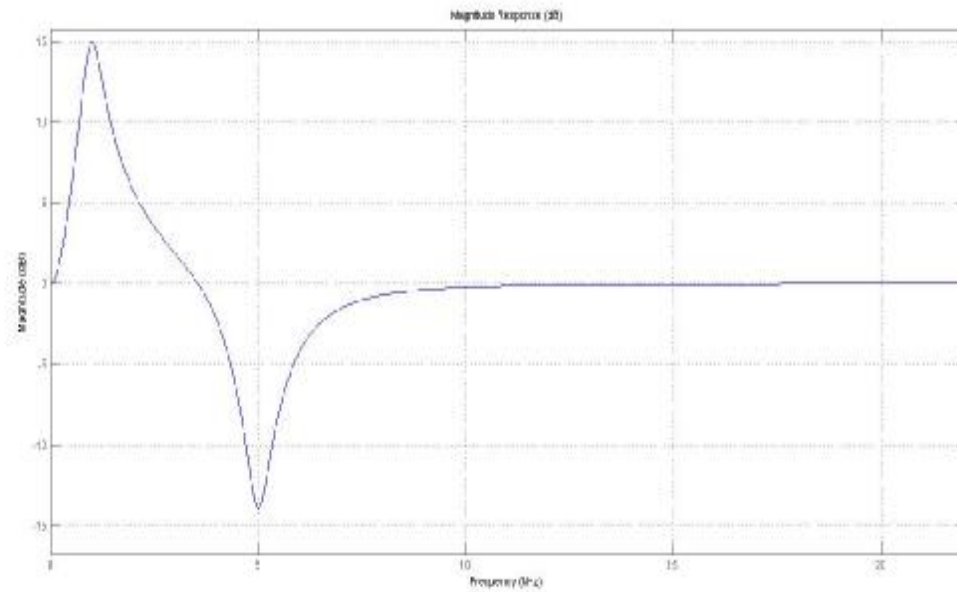


Εικόνα 4.26 Αποκρίσεις συχνότητας 2 peaking φίλτρων με $g=15$ db και $Q=4$ $g=10$ db και $Q=8$

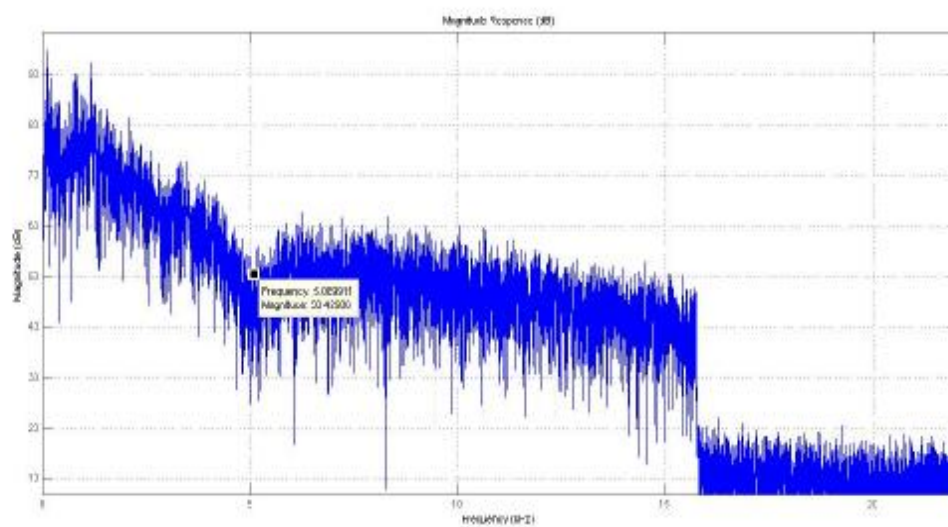


Εικόνα 4.27 peaking φίλτρο

Απόκριση συχνότητας peaking φίλτρου με $f_c=8000$ Hz και κέρδος 15 db. Σύμφωνα με την θεωρία βάζουμε είσοδο στο φίλτρο μια $\delta(n)$ τότε η έξοδος σε ένα ΓΧΑ φίλτρο είναι η κρουστική απόκριση του, κάνοντας fft στην κρουστική απόκριση του φίλτρου παίρνουμε την απόκριση συχνότητας

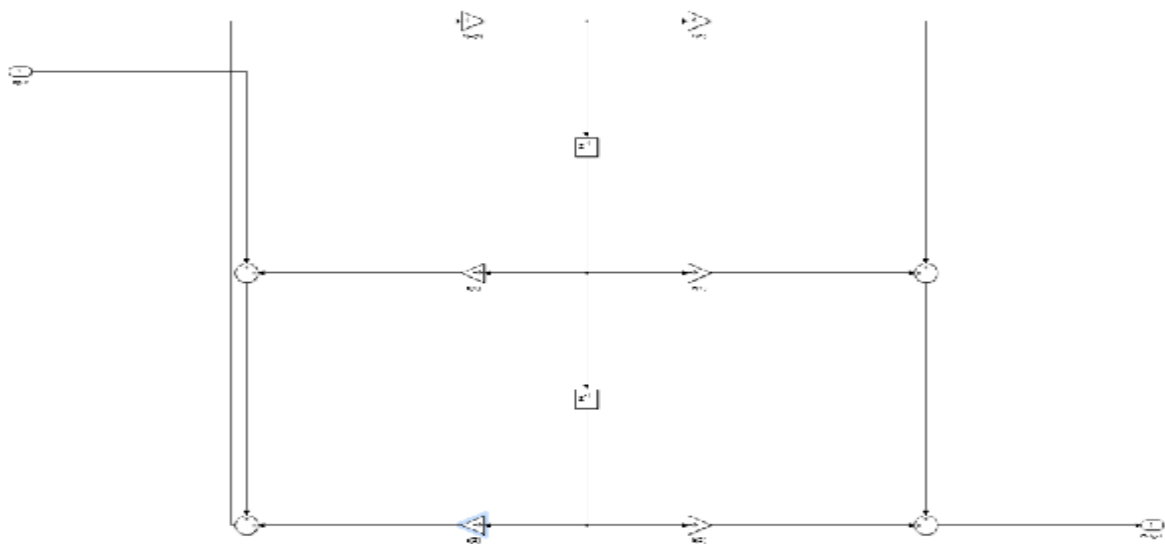


Εικόνα 4.28 Απόκριση συχνότητας peaking $g=15$ $Q=4$ $f_0=1000\text{Hz}$ και notch $g=15$ $Q=8$ ΚΑΙ $f_0=5000\text{Hz}$ σύνδεση σε σειρά(cascade)

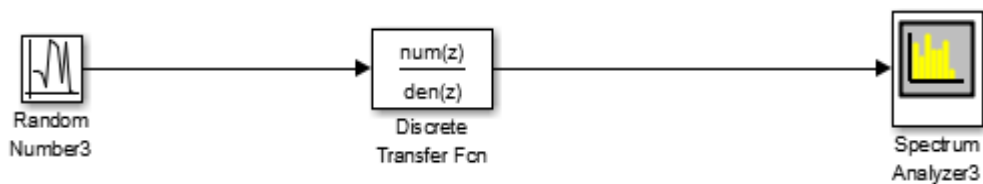


Εικόνα 4.29 Φάσμα σήματος μετά την εφαρμογή των φίλτρων στο σήμα εισόδου(peaking 1000Hz και notch 5000Hz $\text{fft}(y)$ όπου $y=\text{filter}(hcas,y)$)

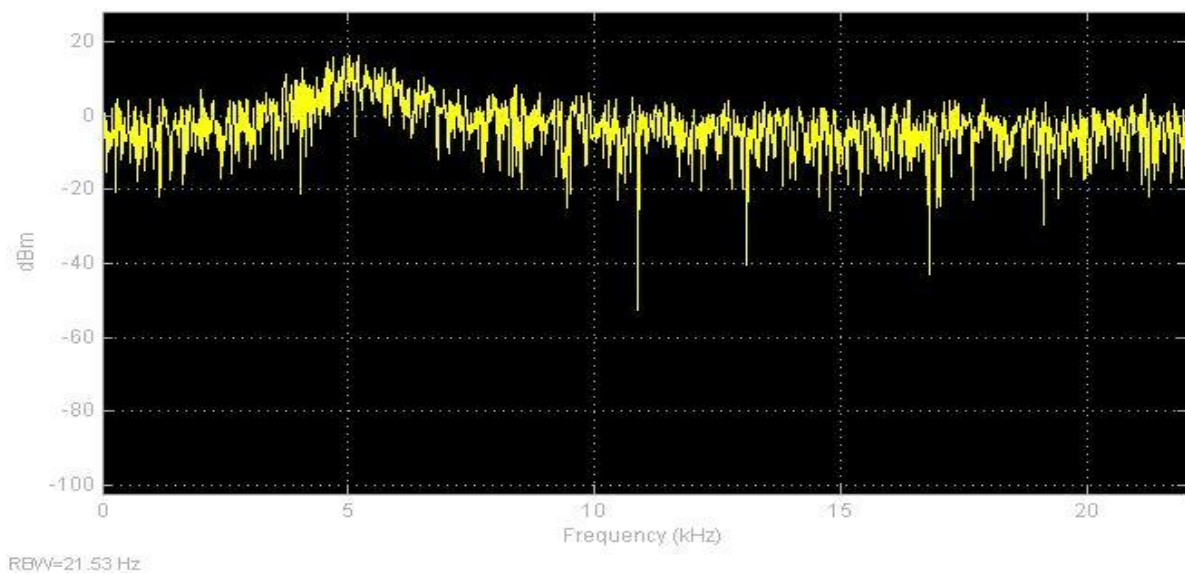
Και η δομή υλοποίησης είναι Direct form II



Εικόνα 4.30 Δομή υλοποίησης peak φίλτρου Direct form II



Εικόνα 4.31 Μοντελοποίηση peaking φίλτρο



Εικόνα 4.32 Απόκριση συχνότητας peak φίλτρου G=15db

4.4.2 Δεύτερη μέθοδος σχεδιασμού peak φίλτρου(full παραμετρική)

Χρησιμοποιώντας την (4.2.3.1) καταλαβαίνουμε ότι ένα peaking φίλτρο μπορεί να εξαχθεί από ένα bandpass+direct path.

Η συνάρτηση μεταφοράς ενός peaking/notch φίλτρου είναι

$$H(z)=1+\frac{H_0}{2}[1\mp A_2(z)] \quad (4.4.2.1)$$

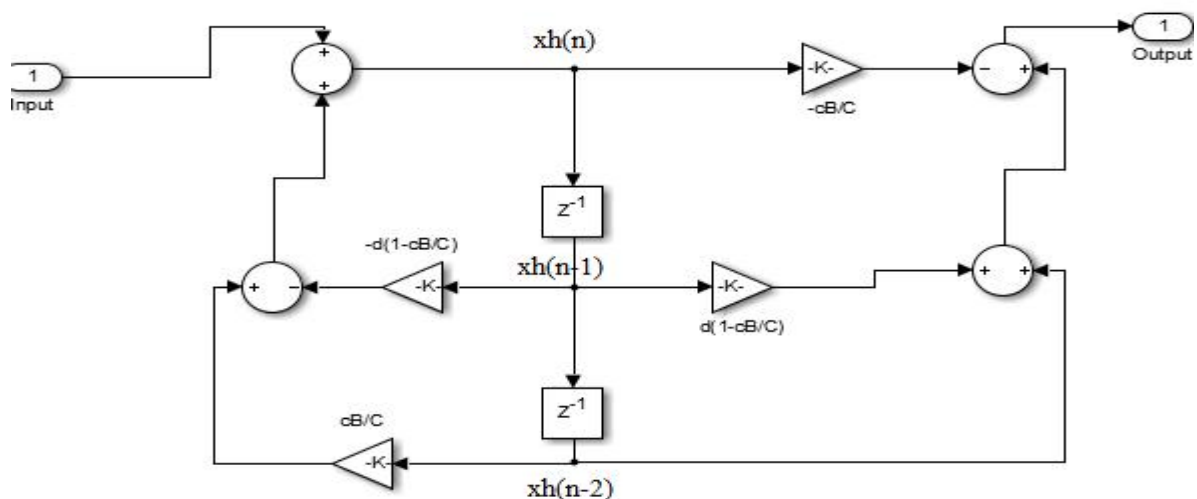
όπου $A_2(z)$ η συνάρτηση μεταφοράς ενός 2 τάξης allpass φίλτρου

Επίσης βλέπω ότι το $A_2(z)$ για να σχεδιάσω το bandpass φίλτρο και κάτ.' επέκταση το peak θέλω μόνο 3 παράγοντες να υπολογίσω αυτοί είναι το a_B και a_C για το εύρος ζώνης του φίλτρου $f_b(\text{Hz})$, το d για την κεντρική συχνότητα του φίλτρου και το κέρδος $V_0=10^{G/20}$ πάνω από την ω_0 χωρίς να υπολογίσω ξανά από την αρχή και τους 5 συντελεστές αλλά μόνο αυτούς τους τρεις. Έτσι στο peaking φίλτρο μπορώ να ελέγγω αυτούς τους τρεις προφανώς σημαντικούς παραμέτρους του φίλτρου υπολογίζοντας μόνο τρεις συντελεστές. Προφανώς η περίπτωση cut επιτυγχάνεται με τον παράγοντα a_C και έτσι θα πάρουμε ένα notch φίλτρο που είναι το αντίστροφο της περίπτωσης boost. Επίσης στην παράμετρο a_C πάλι χρειάζεται το V_0 έτσι ώστε η γωνιακή συχνότητα της περίπτωσης boost να είναι ίδια με της περίπτωσης cut

$$\text{Όπου } A_2(z) = \frac{-aB/C + d(1-aB/C)z^{-1} + z^{-2}}{1 + d(1-aB/C)z^{-1} - aB/Cz^{-2}} \quad (\text{Udo Zolzer :2}^{\text{η}} \text{ εκδοση 2008}) \quad (4.4.2.2)$$

Η εξίσωση διαφορών (στο χρόνο) είναι $xh(n)=x(n)-d(1-aB/C)xh(n-1)+aB/C)xh(n-2)$ και η τελική έξοδος $y1(n)$ θα είναι $y1(n)=-a B/C xh(n)+d(1-aB/C)xh(n-1)+xh(n-2)$

και δομή υλοποίησης



Εικόνα 4.33 Δομή υλοποίησης 2ης τάξης allpass φίλτρο

$$\alpha = \begin{cases} \alpha_B = \frac{1 - \tan\left(\frac{\pi \cdot fb}{fs}\right)}{1 + \tan\left(\frac{\pi \cdot fb}{fs}\right)} & \text{για } v0 > 1 \text{ (ενισχυση)} \\ \alpha_C = \frac{v0 - \tan\left(\frac{\pi \cdot fb}{fs}\right)}{v0 + \tan\left(\frac{\pi \cdot fb}{fs}\right)} & \text{για } v0 < 1 \text{ (εξασθενιση)} \end{cases} \quad (4.4.2.3)$$

$$V0 = 10^{G/20} \quad G(\text{κέρδος σε db.}) \text{ και } V0(\text{κέρδος σε voltage}) \quad (4.4.2.4)$$

$$H0 = V0 - 1 \quad (4.4.2.5)$$

Όπου fb είναι το εύρος ζώνης του φίλτρου το οποίο καθορίζετε από τις παραμέτρους α_B και α_C και για τις δυο περιπτώσεις boost/cut.

$$d \text{ η παράμετρος ελέγχου της κεντρικής συχνότητας } d = -\cos(\omega_0) \quad (4.4.2.6)$$

ω_0 είναι η κανονικοποιημένη κεντρική συχνότητα σε radians/sample (ακτίνια ανά δείγμα) κυκλική συχνότητα

και $\omega_0 = 2\pi f_0 / fs$ rad/samples όπου f_0 η κεντρική συχνότητα του φίλτρου σε Hz

Και εδώ όπως στην περίπτωση του low/high frequency shelving έτσι στον σχεδιασμό του peak και όπως βλέπουμε και στον κώδικα του για την υλοποίηση του φίλτρου (απόκριση συχνότητας) χρειαζόμαστε ένα σήμα εισόδου $x = [\text{zeros}(1,0), 1, \text{zeros}(1,22050)]$; Είναι η διακριτή ώση που έχει πεπερασμένο μήκος άρα και η κρουστική του φίλτρου θα έχει πεπερασμένη διάρκεια.

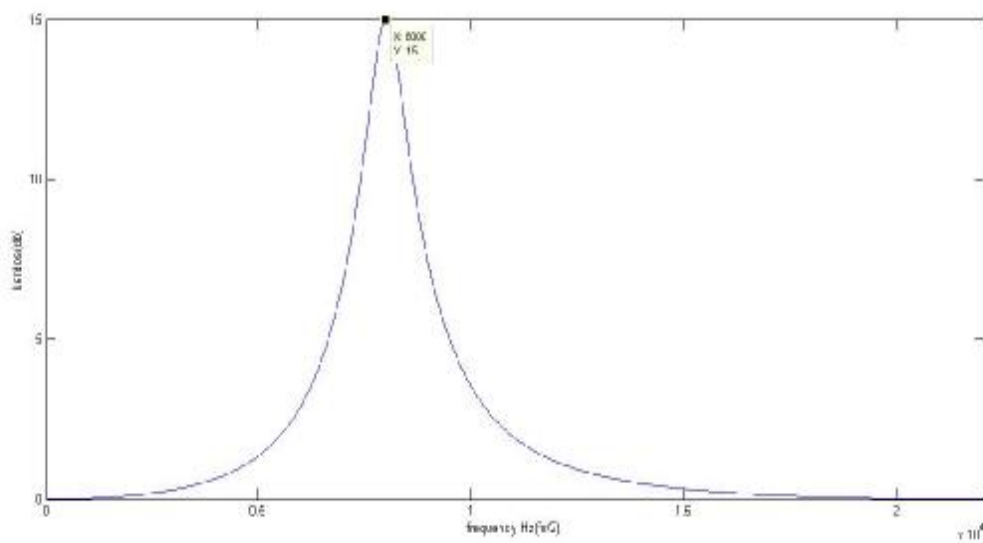
Άρα με την (5.4.2.1) θα με δώσει ένα FIR φίλτρο δηλαδή μόνο μηδενικά.

Οπότε θα χρησιμοποιήσουμε την συνάρτηση μεταφοράς για δεύτερης τάξης peaking φίλτρο αποτελούμενη από μηδενικά και πόλους(IIR) και αυτή θα είναι

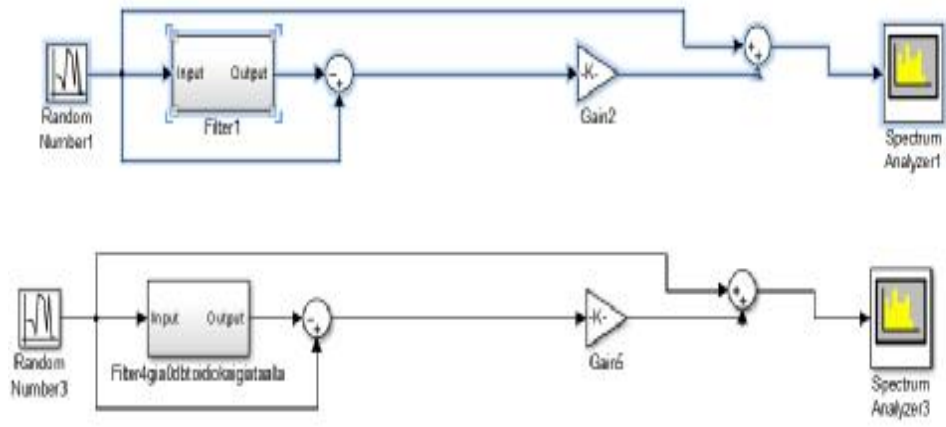
$$H(z) = \frac{1 + (1+abc) \cdot \frac{H0}{2} + d(1-c)z^{-1} + (-c - (1+abc) \cdot \frac{H0}{2})z^{-2}}{1 + d(1-abc)z^{-1} - abc z^{-2}} = H(z) \frac{b0 + b1z^{-1} + b2z^{-2}}{1 + a1z^{-1} + a2z^{-2}} = \frac{1.1591 - 1.4615z^{-1} + 0.7721z^{-2}}{1 - 1.4615z^{-1} + 0.9312z^{-2}}$$

$$H(z) = \frac{bz}{az} \quad (4.4.2.7)$$

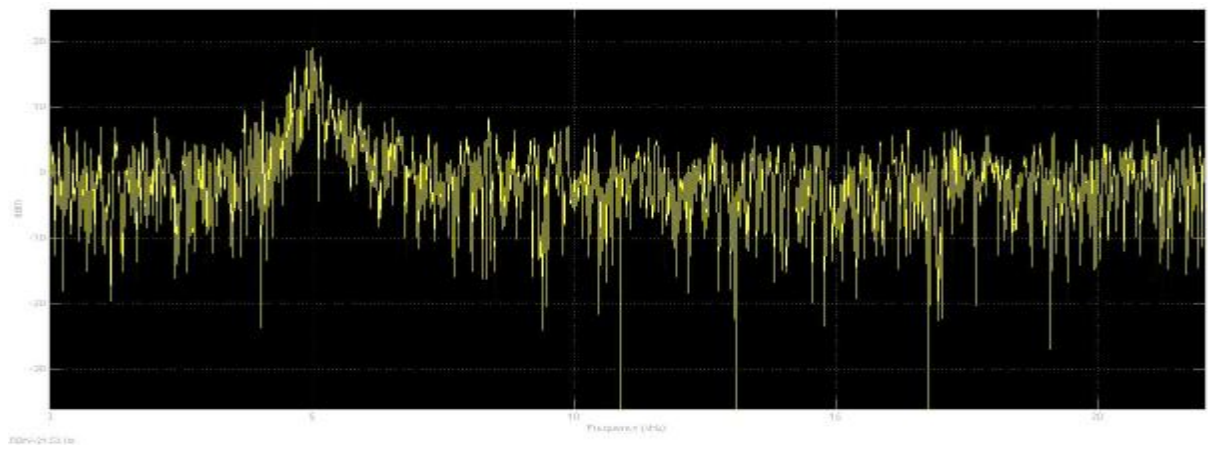
Για την περίπτωση cut $V0_{cut} = 10^{-(G/20)}$ και χρησιμοποιούμε το α_C (Udo Zolzer :2^η έκδοση 2008)



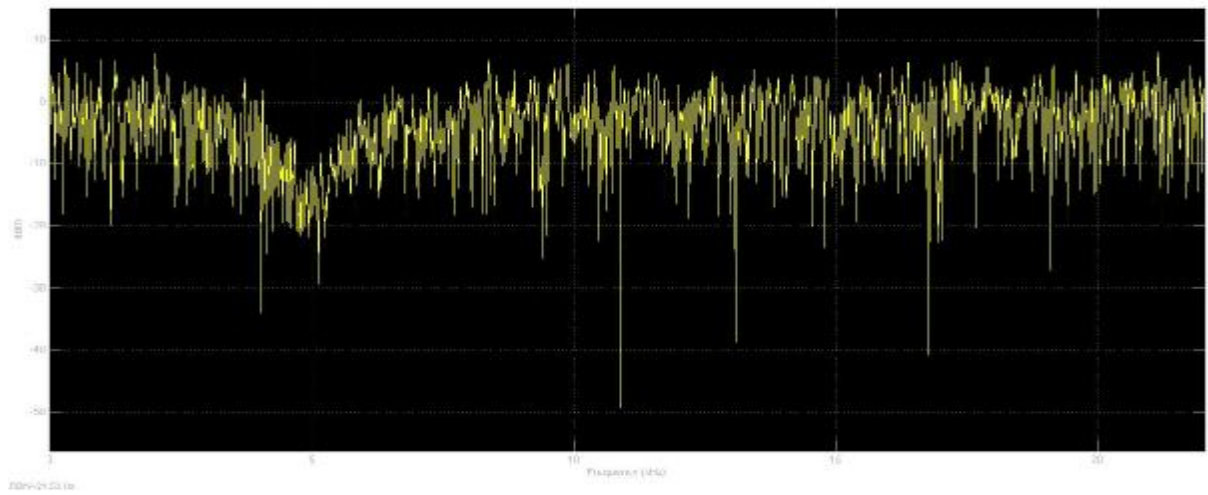
Εικόνα 4.34 Peaking/Notch φίλτρο 2ης τάξης $y(n) = H0/2 * (x(n) - A2(z)) + x(n)$



Εικόνα 4.35 Peaking/Notch φίλτρο 2ης τάξης($y(n)=H0/2*(x(n)-A2(z))+x(n)$)



Εικόνα 4.36 Peaking φίλτρο 5000Hz



Εικόνα 4.37 Notch 5000Hz Simulink

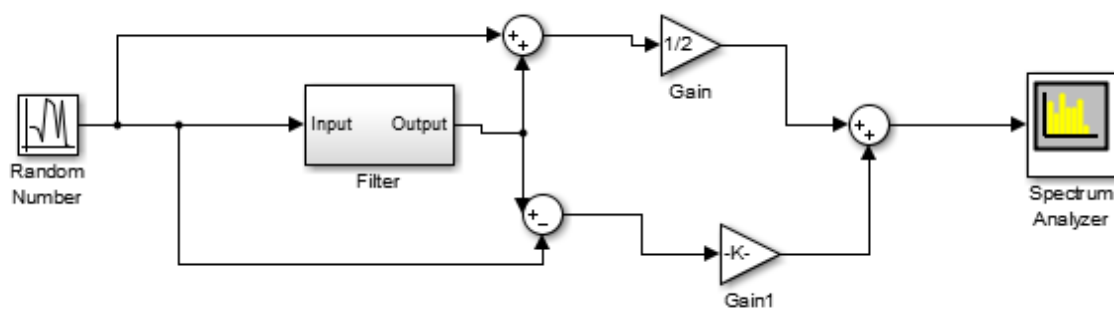
Όπως και στο low/high shelving φίλτρο έτσι και εδώ στην εικόνα 4.38 μια εναλλακτική μοντελοποίηση ενός full parametric φίλτρου εκμεταλλευόμενοι τις διαφορετικές υλοποιήσεις που με δίνει το ολοπερατό φίλτρο λόγω της μετατόπισης φάσης έτσι ώστε να έχω peaking/notch για την περίπτωση boost και για την περίπτωση cut (4.4.2.3)

Η συνολική συνάρτηση μεταφοράς του συστήματος θα είναι

$$H(z) = \frac{1}{2} [1 + A_2(z)] + \frac{V_0}{2} [1 - A_2(z)] \quad (4.4.2.8)$$

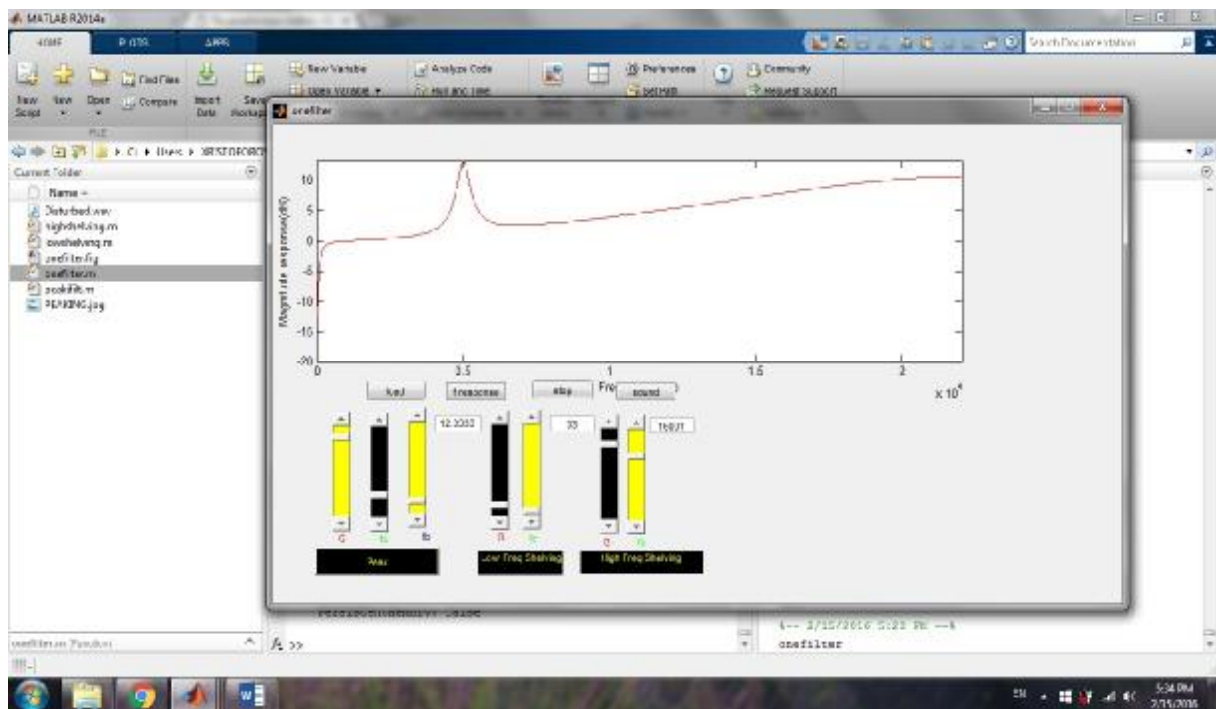
Το notch φίλτρο προκύπτει εάν αντιστρέψουμε το κέρδος και χρησιμοποιήσουμε την εξίσωση 4.4.2.3 για την περίπτωση cut

Το $V_0/2$ ρυθμίζει το κέρδος peak/notch. Έτσι για notch $V_0^{\text{cut}} = 1/V_0^{\text{boost}}$ και χρησιμοποιούμε το a_C για την ρύθμιση του εύρους ζώνης από την 4.4.2.3 και της κεντρικής συχνότητας από την 4.4.2.6 (PHILLIP A. REGALIA & SANJIT K. MITRA 1987)



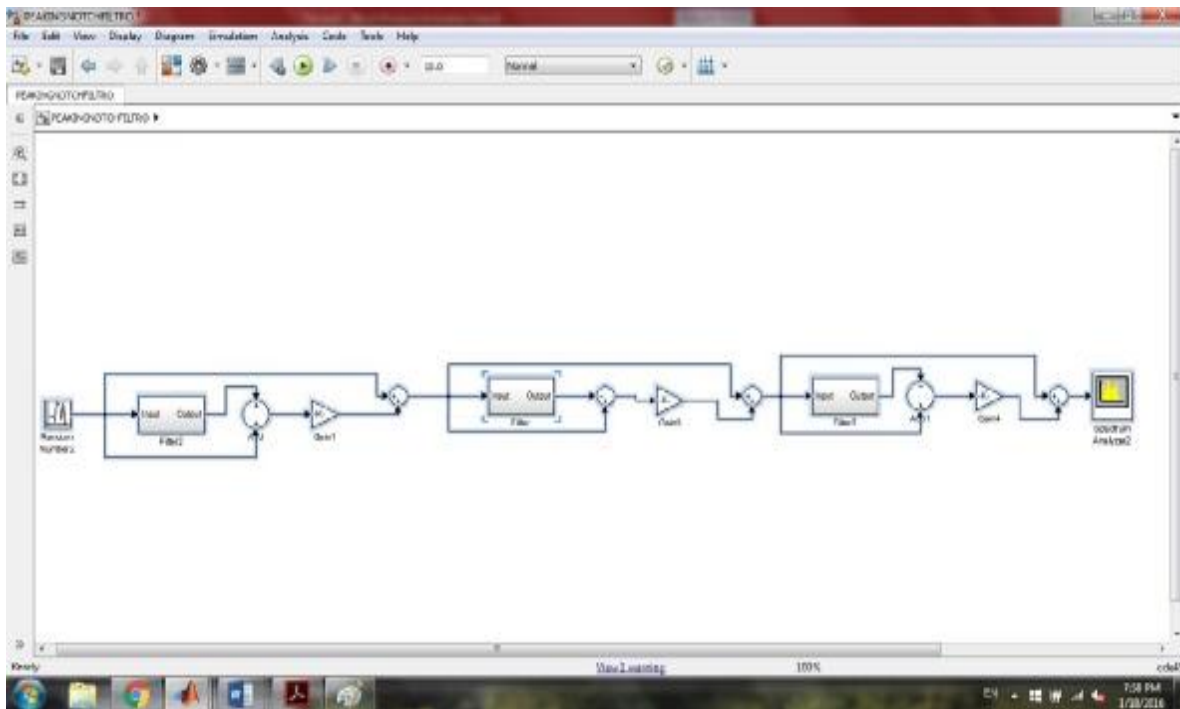
Εικόνα 4.38 Notch φίλτρο

4.4.3 Χρήση τριών μπαντών(3bands) ισοσταθμιστής (full parametric)

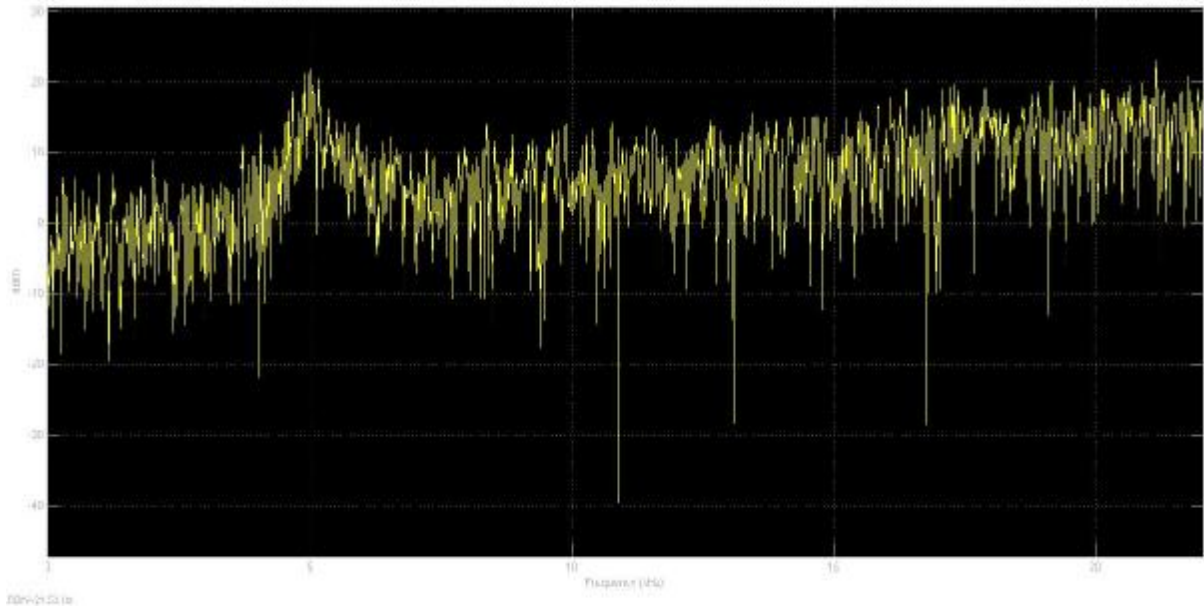


Εικόνα 4.39 Full parametric σχεδιασμός φίλτρων

Στο παραπάνω γραφικό ουσιαστικά βλέπουμε παραμετρικό equalizer(κώδικας 4 παράρτημα) . Τα φίλτρα είναι full parametric Αυτό σημαίνει ότι έχουμε την δυνατότητα να μεταβάλω και τους τρεις παραμέτρους του φίλτρου, εύρος ζώνης, συχνότητα αποκοπής και κέρδος ανεξάρτητα υπολογίζοντας τρεις συντελεστές το cB/C , το d και το $H0=V0-1$,σε αντίθεση με την πρώτη μέθοδο που εκεί εάν αλλάξω μια παράμετρο αλλάζουν όλοι οι συντελεστές του φίλτρου (5 για 2^{ης} τάξης ($b0,b1,b2/a0=1,a1,a2$)). Οι παράμετροι των φίλτρων ορίζονται ως f_c =συχνότητας αποκοπής(Hz), f_b =εύρος ζώνης του φίλτρου(Hz) και G =κέρδος (db) Η μοντελοποίηση του παραπάνω συστήματος γίνεται στο Simulink του Matlab.Η σύνδεση των φίλτρων γίνεται σε σειρά(cascade) .Έτσι έχουμε ένα lowshelving (boost/cut περίπτωση) ένα peak/notch και ένα highshelving(boost/cut)



Εικόνα 4.40 Σύνδεση σε σειρά full parametric eq



Εικόνα 4.41 Απόκριση συχνότητας παραμετρικού ισοσταθμιστή στο Simulink

4.5 Σφάλματα κβαντίσου κατά την υλοποίηση

Για να μπορέσουμε να υλοποιήσουμε τα φίλτρα με έναν ψηφιακό επεξεργαστή θα πρέπει οι συντελεστές του φίλτρου αλλά και η είσοδος του και κατ επέκταση η έξοδος να μετατραπούν σε δυαδική μορφή κάνοντας είτε περικοπή είτε στρογγυλοποίηση κάτι που σημαίνει ότι αυτό οδηγεί στην αλλαγή της θέσης των επιθυμητών-αρχικών συντελεστών(πόλων και μηδενικών για IIR λόγω της στρογγυλοποίησης) κάτι που σημαίνει ότι μπορεί ακόμα και ένας πόλος να οδηγηθεί έξω από τον μοναδιαίο και έτσι το φίλτρο να γίνει ασταθές κυρίως όταν η τάξη του φίλτρου είναι μεγαλύτερη από 2. Επίσης μιας και ένας υπολογιστής αναπαριστά τους αριθμούς σε πεπερασμένες δυαδικές τιμές αυτό μπορεί να σημαίνει όχι ακριβής αναπαράσταση. Αναφερόμαστε στο σφάλμα κβαντίσου που οφείλετε στο ότι δεν κάνουμε αναπαράσταση με ακρίβεια δηλαδή λιγότερα ψηφία(bits) από τα αρχικά επιθυμητά. Έτσι δημιουργούνται προβλήματα στην απόκριση συχνότητάς του φίλτρου(δεν θα είναι το επιθυμητό). Δεν θα δώσουμε ιδιαίτερο βάρος μιας και θα πρέπει να το λάβουμε υπόψιν στην πρακτική υλοποίηση του φίλτρου, αλλά θα κάνουμε μια ενδεικτική αναφορά στο πως μπορούμε να βελτιώσουμε το σφάλμα κβαντίσου στα ψηφιακά IIR φίλτρα.

4.5.1 Δυαδική αναπαράσταση των συντελεστών

Η πεπερασμένη αναπαράσταση δυαδικών τιμών γίνεται με δύο τρόπους

1) Σταθερής υποδιαστολής 2) και κινητής υποδιαστολής.

θα αναφερθούμε σε επεξεργαστές σταθερής υποδιαστολής και στα σφάλματα κβαντίσου που προκύπτουν

Σταθερής υποδιαστολής: ο διαχωρισμός μεταξύ του ακέραιου και του δεκαδικού μέρους, δηλαδή η υποδιαστολή, βρίσκεται πάντα σε μια συγκεκριμένη, σταθερή, θέση

Το πρώτο ψηφίο μας δεινί το πρόσημο του αριθμού 0=θετικός, 1=αρνητικός.

πχ 5μπιτος θα είναι $0.8125=0.11010(1*2^{-1}+1*2^{-2}+1*2^{-4})$ και το $-0.8125=1.11010$. Επεξεργαστές σταθερής υποδιαστολής είναι ταχύτεροι και φθηνότεροι σε σχέση με εκείνους κινητής υποδιαστολής και για αυτό προτιμώνται.

Η πιο συχνή παράσταση αριθμού σταθερής υποδιαστολής είναι η παράσταση συμπληρώματος ως προς 2. Σε αυτήν ο θετικός αριθμός πραγματοποιείται με το προβλεπόμενο τρόπο στο δυαδικό και ο αρνητικός πραγματοποιείται εάν αντικαταστήσουμε όλα τα ψηφία 0 και 1 με το συμπληρωματικό τους

δηλαδή $1 \rightarrow 0$ και $0 \rightarrow 1$ Ξεκινάμε από τα δεξιά προς τα αριστερά και αφήνουμε το LSB(το ελάχιστο σημαντικό bit και το πρώτο 1 όπως είναι και μετά τα αλλά ψηφία τα αντιστρέφουμε.
Π.χ. $0.11010=0.8125$ και $-0.8125=1.00110$

4.5.2 Σφάλματα κβαντίσου IIR φίλτρα.

Το σφάλμα κβαντίσου επηρεάζει περισσότερο ένα IIR φίλτρο μιας και αυτό οι τιμές του παρονομαστή είναι $\neq 0$ οποτε έχει πόλους με οποιαδήποτε τιμή μέσα στον μοναδιαίο. Αυτό σημαίνει ότι οι νέες κβαντισμένες τιμές αυτών μπορούν να κάνουν το φίλτρο ασταθές αν έστω και ένας πόλος βρεθεί έξω από τον μοναδιαίο κυρίως όταν μιλάμε για μια τάξη του φίλτρου μεγαλύτερη του 2 ($N > 2$.) Αρχικά η συνάρτηση μεταφοράς ενός φίλτρου με μη κβαντισμένους συντελεστές είναι

$$H_z = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=0}^N a_k z^{-k}} \quad (4.5.2.1)$$

και με κβαντισμένους συντελεστές είναι

$$H_z(q) = \frac{\sum_{k=0}^M b_k(q) z^{-k}}{1 + \sum_{k=0}^N a_k(q) z^{-k}} \quad (4.5.2.2)$$

όπου οι κβαντισμένοι συντελεστές(μετα) συνδέονται με τους μη κβαντισμένους(πριν) b_k και a_k ως $b_k(q) = b_k + \Delta b_k$ για $k=0, 1, \dots, M$ και $a_k(q) = a_k + \Delta a_k$ για $k=1, 2, \dots, N$ όπου Δa_k και Δb_k είναι το σφάλμα κβαντίσου

Η ευστάθειά του φίλτρου ξέρουμε ότι εξαρτάται από τις θέσεις των πόλων οπότε πρέπει να ξέρουμε πόσο οι κβαντισμένοι συντελεστές του παρονομαστή(a_k) επηρεάζουν τις θέσεις των πόλων οπότε αν ο παρονομαστής της $H(z)$ μπορεί να εκφραστεί

$$D(z) = 1 + \sum_{k=0}^N a_k z^{-k} = \prod_{k=1}^N (1 - p_k z^{-k}) \quad (\text{John Proakis Dimitris G Manolakis}) \quad (4.5.2.3)$$

όπου p_k είναι οι πόλοι του φίλτρου τότε μπορούμε να εκφράσουμε τον παρονομαστή της $H(q)$ $D_z(q) = \prod_{k=1}^N (1 - p_k(q) z^{-k})$ όπου εδώ τώρα $p_k(q) = p_k + \Delta p_k$ με Δp_k είναι το σφάλμα κβαντίσου που προκύπτει από τον κβαντίσιμο των μη κβαντισμένων συντελεστών δηλαδή $p_k(q)$ είναι οι νέες θέσεις μετά την κβαντοποίηση των συντελεστών.

Τώρα το συνολικό σφάλμα κβαντίσου στους πόλους είναι

$$\Delta p_i = \sum_{k=1}^N \frac{\theta p_i}{\theta a_k} \Delta a_k \quad (4.5.2.4)$$

από εδώ καταλαβαίνουμε την αλλαγή των θέσεων των πόλων (θp_i) ενός φίλτρου εξαιτίας των μεταβολών Δa_k των συντελεστών $\theta a(k)$.

όπου $\frac{\theta p_i}{\theta a_k}$ είναι η ευσθεσια των θέσεων των πόλων εξετίας των κβαντισμένων συντελεστών

$$H \quad (4.5.2.4) \text{ μπορεί να γραφεί } \Delta p_i = - \sum_{k=1}^N \frac{p_i^{N-k}}{\prod_{l=1, l \neq i}^N (p_i - p_l)} \quad (4.5.2.5)$$

Οι όροι $(p_i - p_l)$ στον παρονομαστή είναι η απόσταση(το διάνυσμα) μεταξύ του πόλου p_i στον πόλο p_l μέσα στον μοναδιαίο (zplane) . Έτσι αντιλαμβανόμαστε ότι όσο πιο μικρό αυτό το διάνυσμα (η απόσταση μεταξύ των πόλων μικρή αρκετά) οπότε εάν μεταβάλουμε κατά Δa_k τότε το σφάλμα Δp_i μεγαλώνει δηλαδή μεγαλύτερες μεταβολές στις θέσεις των πόλων κυρίως όταν έχουμε ένα φίλτρο υψηλής τάξης. Έτσι οποιοσδήποτε πόλος μπορεί να βγει έξω από τον μοναδιαίο $p_i > 1$ οπότε το φίλτρο να γίνει ασταθές. (John Proakis Dimitris G Manolakis)

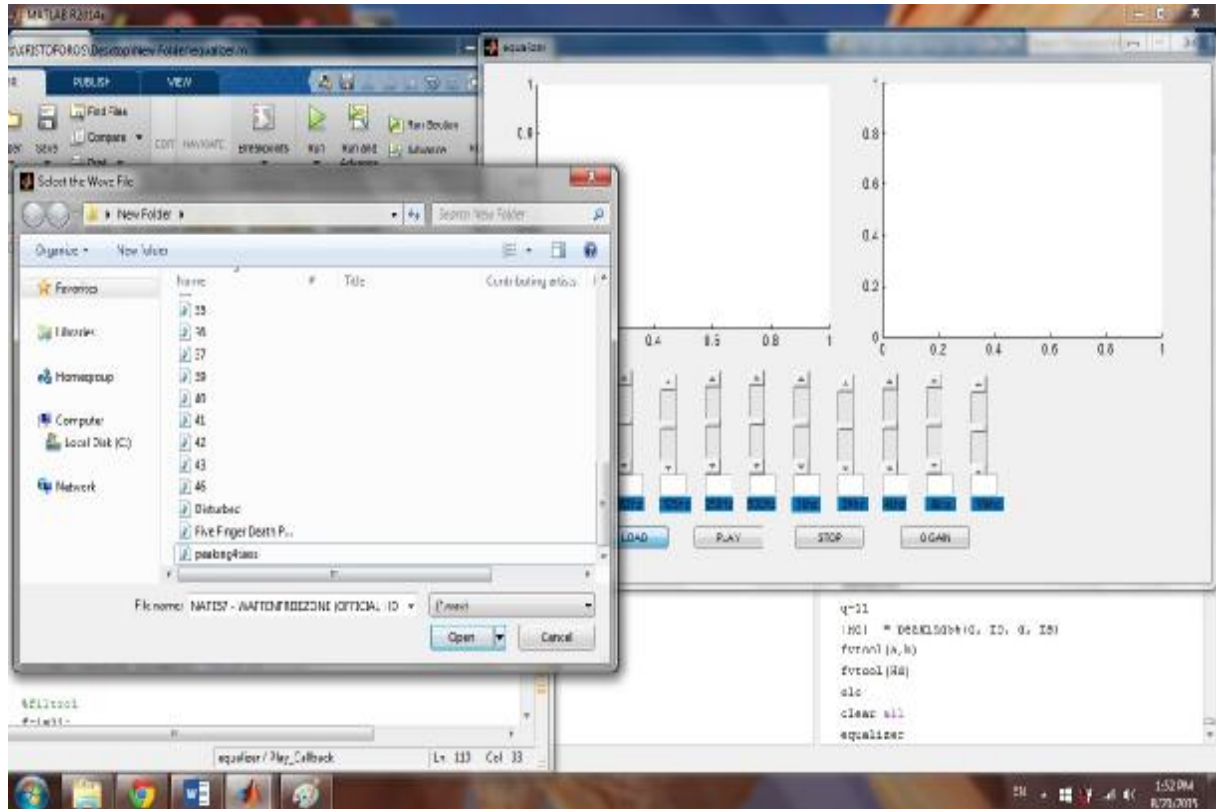
Αυτό που πρέπει να κάνουμε είναι σε ένα φίλτρο υψηλής τάξης, που το πρόβλημα μεγαλώνει καθώς ο κάθε κβαντισμένος συντελεστής επηρεάζει τον άλλον ,αυτό που μπορούμε να κάνουμε είναι να μεγαλώσουμε την απόσταση μεταξύ των δυο τυχαίων πόλων p_i και p_l δηλαδή να δώσουμε χώρο μέσα στον μοναδιαίο κύκλο άρα περισσότερες πιθανές θέσεις για κάθε πόλο μέσα στο μοναδιαίο.

Αυτό μπορούμε να το κάνουμε εάν επιλέξουμε να σχεδιάσουμε ένα φίλτρο κυρίως υψηλής τάξης σπάζοντας το σε τμήματα(section) $2^{ου}$ βαθμού η πρώτης. Έτσι κάθε τμήμα κβαντοποιείται ξεχωριστά οπότε οι πόλοι βρίσκονται σε μεγαλύτερη απόσταση μεταξύ τους κάτι που σημαίνει ότι το φίλτρο-σύστημα θα επηρεάζεται λιγότερο από τα σφάλματα κβαντισμού. Ουσιαστικά μειώνουμε των

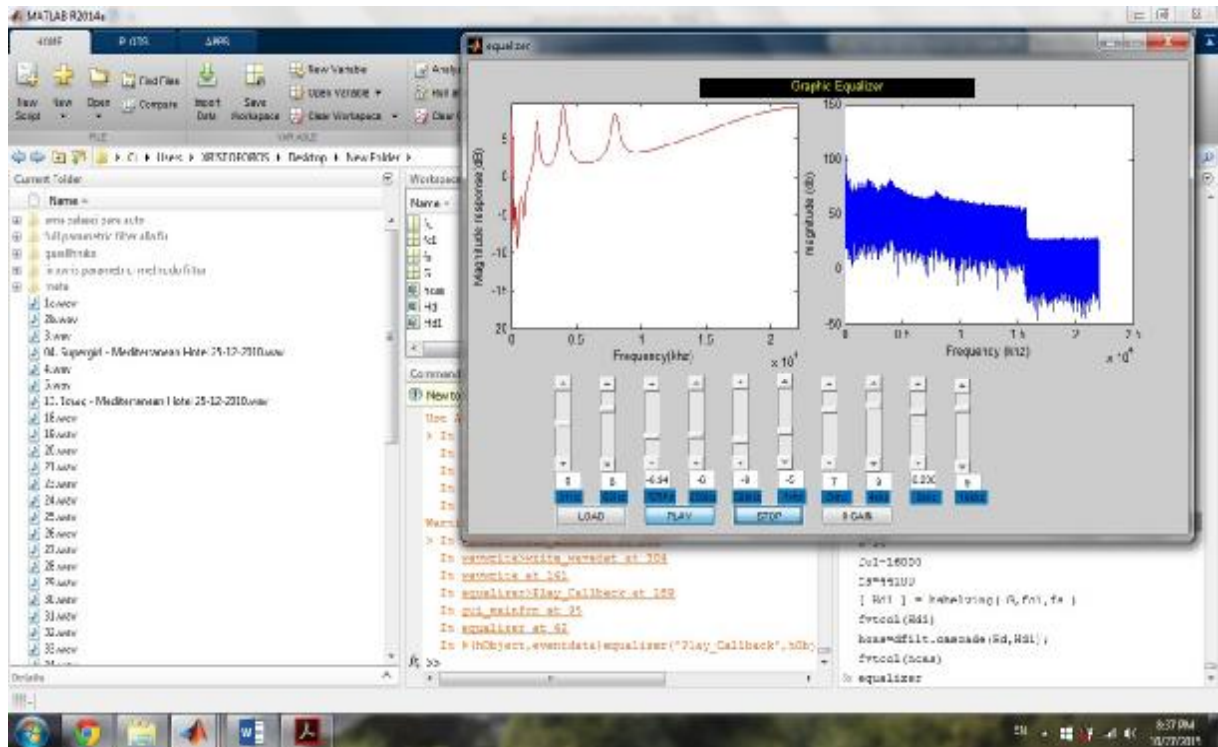
αριθμό των πόλων που επηρεάζονται από κβαντίσμο μέσα στο μοναδιαίο κύκλο και έτσι μειώνουμε το σφάλμα κβαντίσμου συνολικά στο φίλτρο Έτσι το φίλτρο έχει λιγότερες πιθανότητες να είναι ασταθές αφού αυξάνονται οι πιθανές θέσεις του κάθε πόλου μέσα στον μοναδιαίο

Ο τρόπος με τον οποίο θα συνδέονται τα τμήματα ο καταλληλότερος είναι σε σειρά (cascade) γιατί βελτιώνει και την ευαισθησία των μηδενικών γιατί και αυτά μπορούν να σπάσουν σε τμήματα 2^{15} τάξης κάτι που το θέλουμε μιας και η παράλληλη σύνδεση δεν το κάνει αυτό (John Proakis Dimitris G Manolakis 2007)

4.6 Audio Equalizer μεταβλητού κέρδους(θεωρητικός σχεδιασμός εφαρμογής)



Εικόνα 4.42 Equalizer



Εικόνα 4.43 Equalizer

Στην εικόνα 4.42 βλέπουμε έναν γραφικό ισοσταθμιστή .Λέγονται έτσι ουσιαστικά από την μορφή που έχουν δηλαδή έχουν μόνο τα ποτενσιόμετρα(μπάντες) που ρυθμίζουν το πλάτος(κέρδος db) του φίλτρου και ένα figure(γραφικό) που μας δείχνει το κέρδος που θα έχει το φίλτρο σε σχέση με την κεντρική συχνότητα του φίλτρου

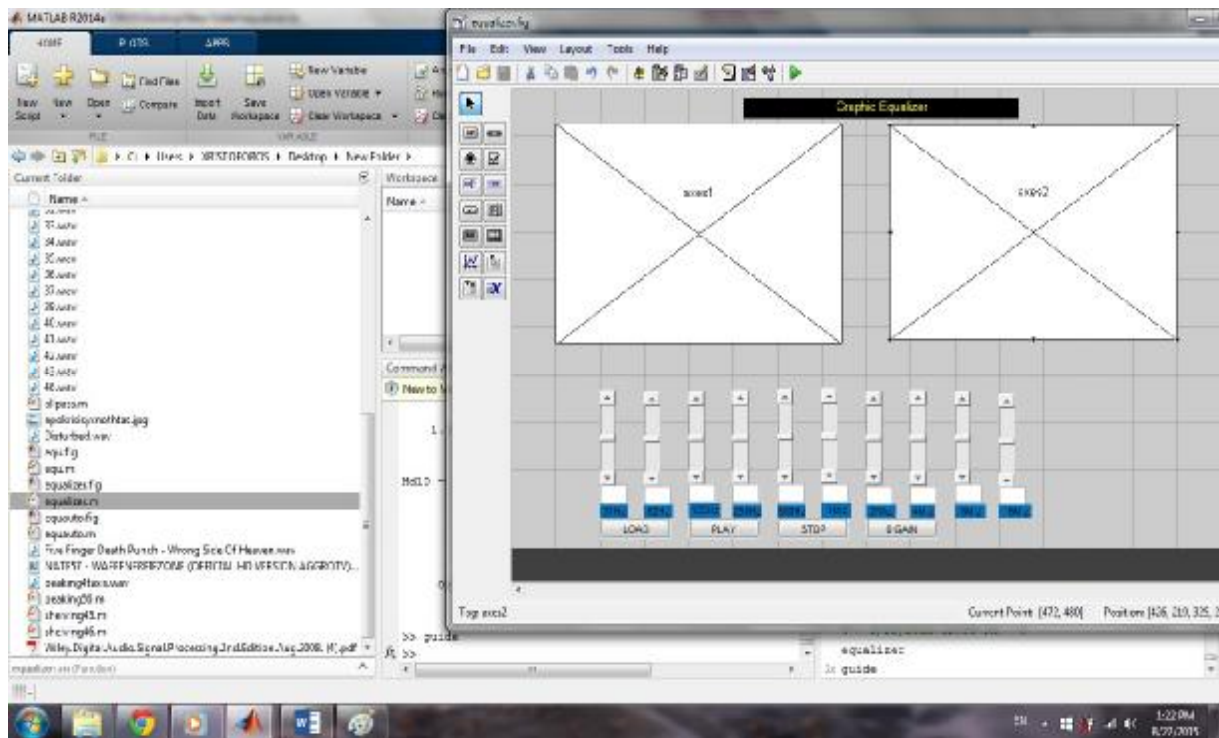
Ο ισοσταθμιστής υλοποιήθηκε στο guide του Matlab. Είναι 10 περιοχών(μπάντες) δηλαδή "σπάμε" το φάσμα συχνοτήτων της εισόδου σε 10 κομμάτια και με τα 10 φίλτρα τα οποία έχουμε επηρεάζουμε το σήμα εισόδου (ηχητικό σήμα) για να πάρουμε στην εξόδου του ένα σήμα ήχου διαφορετικό τονικά η να αλλάξουμε την ακουστότητα κάποιων μουσικών οργάνων Έχει τέσσερα κουμπιά το load που ουσιαστικά με αυτό θα αναζητήσουμε εάν αρχείο τύπου wav και θα είναι το σήμα εισόδου και αφού κάνουμε της αλλαγές στις μπάντες θα κάνει το φιλτράρισμα της εισόδου με τα 10 αντίστοιχα φίλτρα- περιοχές και θα μας δώσει την τελική απόκριση . Το κουμπί play θα διαβάσει την είσοδο, τα φίλτρα συνδεδεμένα σε σειρά θα εξάγουν την απόκριση του ολικού συστήματος Το κουμπί stop θα σταματήσει την απόκριση και έτσι μπορούμε να κάνουμε τις αλλαγές στους ρυθμιστές κέρδους των φίλτρων και να ξανακούσουμε το σήμα από την αρχή με τις αντίστοιχες αλλαγές που έχουμε κάνει και το κουμπί .0 Gain που επαναφέρει το κέρδος των φίλτρων στα 0 db. δηλαδή το σήμα εισόδου δεν έχει δεχθεί καμία αλλαγή. Στο γραφικό(figure)βλέπουμε την απόκριση συχνοτήτάς των 10 φίλτρων 1 low shelving 8 peaking/notch και 1 high shelving με το αντίστοιχο κέρδος της κεντρικής συχνότητάς του κάθε φίλτρου Είναι φίλτρα άπειρης κρουστικής απόκρισης(έχουν πόλους και μηδενικά) και είναι 1^{ης} τάξης τα low/high frequency shelving και 2^{ης} τάξης τα peaking/notch.Για φίλτρα μεγαλύτερης τάξης θα μπορούσαμε να συνδέσουμε δυο φίλτρα σε σειρά με τα ίδια χαρακτηριστικά Για ακουστικές εφαρμογές προτιμάμε τα άπειρης κρουστικής απόκρισης (IIR) γιατί όπως ξέρουμε ένα FIR για να έχει τα ίδια επιθήματα αποτελέσματα(σχεδιαστικά και να καλύπτει τις απαιτήσεις του σχεδιαστή) σε σχέση με ένα IIR θα πρέπει να είναι μεγαλύτερης τάξης Παρόλο που δεν έχουν γραμμική φάση αυτό σημαίνει παραμόρφωση στην μορφή του σήματος(κάτι το οποίο το αντιλαμβανόμαστε μόνο οπτικά) ωστόσο δεν μας ενδιαφέρει γιατί έχουμε να κάνουμε με ήχο οπότε είτε γραμμικής φάσης είτε μη γραμμικής φάσης πάλι το ίδιο ηχητικό αποτέλεσμα θα είχαμε. Είναι πιο οικονομικά σε σχέση με τα FIR(λιγότεροι συντελεστές ,λιγότερες

πράξεις λιγότερες θέσεις μνήμης για την αποθήκευση των συντελεστών εισόδου εξόδου

Ο ισοσταθμιστής αποτελείται από sliders=ρυθμιστικά (ποτενσιόμετρα)τα οποία το καθένα ρυθμίζουν (ανεβάζοντας η κατεβάζοντας) το κέρδος(gain \pm db.) προκαλώντας ενίσχυση η εξασθένηση της αντίστοιχης μοναδικής κεντρικής συχνότητάς στην οποία ανήκει το καθένα δηλαδή πίσω από κάθε slider υπάρχει ένα μοναδικό φίλτρο με μια μοναδική κεντρική συχνότητα για peaking η μια μοναδική σταθερή συχνότητα αποκοπής για low/high shelving φίλτρο το οποίο επηρεάζει(φιλτράρει) μια συγκεκριμένη περιοχή-εύρος συχνοτήτων Στο αριστερό γράφημα δείχνω την απόκριση συχνότητάς και το δεξιά μας δείχνει το φάσμα συχνοτήτων. Στον γραφικό ισοσταθμιστή η συχνότητα αποκοπής και το Q (για peaking notch) παραμένουν σταθερά και το μόνο που μεταβάλατε είναι το κέρδος(\pm db) ενίσχυση η εξασθένηση ενός εύρους συχνοτήτων. Συνήθως ένας γραφικός ισοσταθμιστής μη επαγγελματικός αποτελείται από 10 μπάντες φίλτρων οκτάβας με κεντρικές συχνότητες που θα ρυθμιστούν:31,62.125.250,500,1000,4000,8000,16000Hz.Το εύρος συχνοτήτων είναι από 0 έως 22050 Hz fs/2 (Συχνότητα Nyquist)

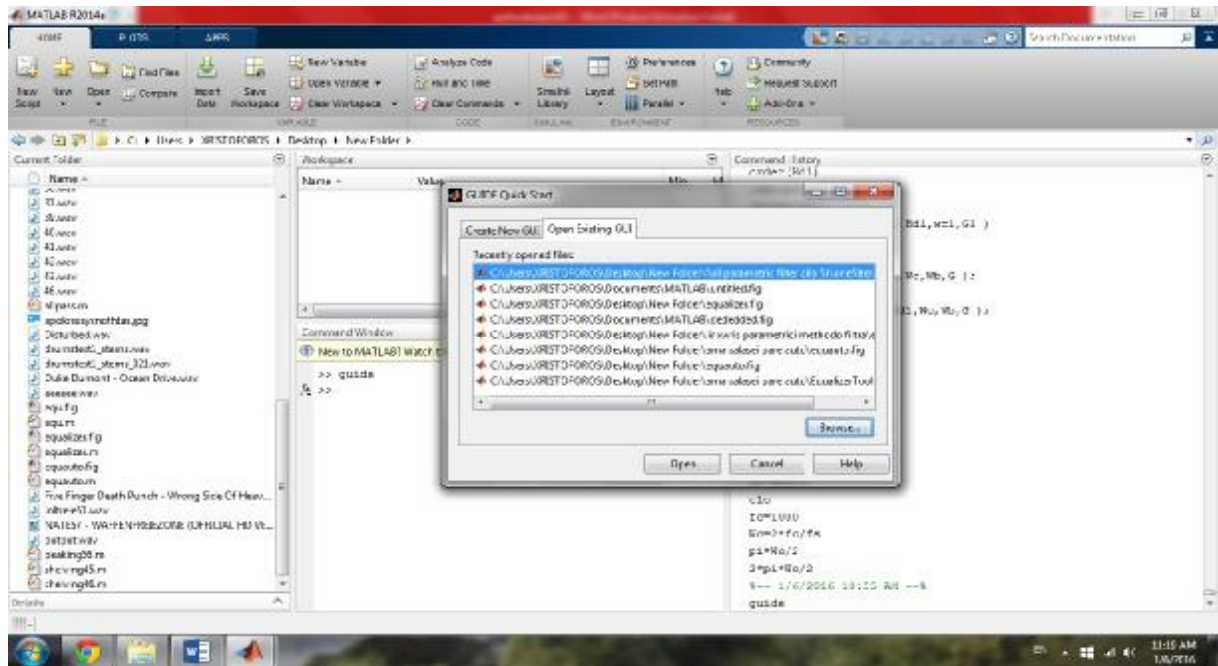
4.7 Ενδεικτική εξήγηση κώδικα (GUI Matlab)

4.7.1 Guide Matlab



Εικόνα 4.44 Βιβλιοθήκη GUI Matlab

Για να ανοίξουμε την βιβλιοθήκη GUI γράφουμε στο command window 'guide'.Εκεί αν θέλουμε να ανοίξουμε ένα αρχείο .fig που υπάρχει είδη τότε κάνουμε browse αλλιώς επιλέγουμε την επιλογή 'create new GUI'



Εικόνα 4.45 παράθυρο αναζήτησης αρχείο. fig

Η βιβλιοθήκη guide είναι μια βιβλιοθήκη του Matlab. Περιέχει διαφορά στοιχεία όπως pushbutton, togglebutton, sliders, text box, edit box, axes κ.α. (εικόνα 4.41) Από τα παραπάνω αντικείμενα όλοι γνωρίζουμε τι κάνουν. Τα παίρνεις από την αριστερή πλευρά και τα σέρνεις μέσα στο γραφικό περιβάλλον.

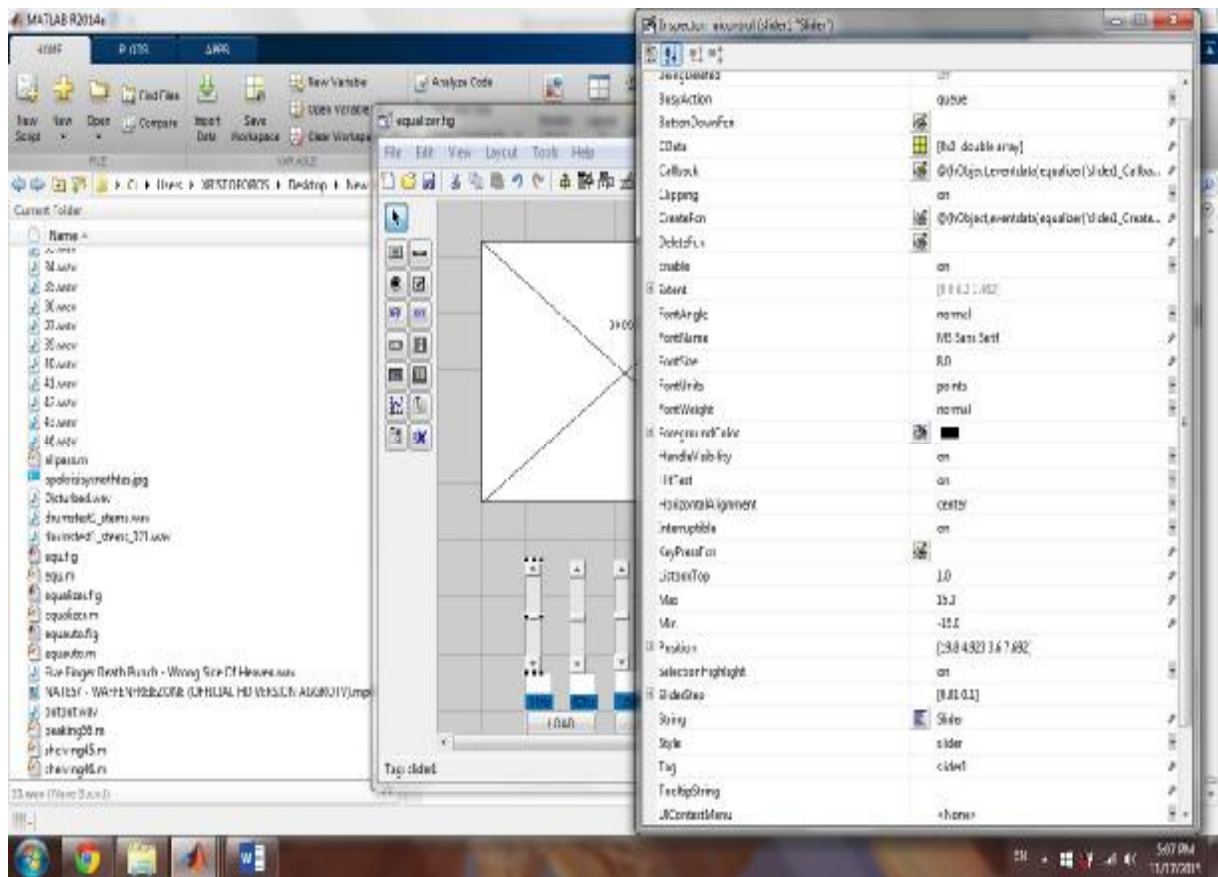
Πατώντας διπλό κλικ πάνω σε κάθε στοιχείο βγαίνει ένα παράθυρο (property inspector = επιθεωρητής ιδιότητας) και από εδώ μπορείς να καθορίσεις τα χαρακτηριστικά του κάθε στοιχείου όνομα, διάμετρος κ.α. Αφού τοποθετήσουμε τα διαφορά στοιχεία κάνουμε save και θα βγει ο editor που εκεί θα προγραμματίσουμε τις λειτουργίες το πώς θα αλληλοεπιδρά το κάθε στοιχείο με το χρήστη



Εικόνα 4.46 Εργαλεία σχεδίασης GUI

Push button	Είναι ένα κουμπί που εκτελεί μια διαδικασία(Callback όταν πατηθεί.
slider	Με το αντικείμενο αυτό μεταβάλλουμε μια τιμή
Radio button	Εδώ επιλέγουμε μια από τις επιλογές από μια σειρά επιλογών. Αν διαλέξουμε την μια αναιρούμε την άλλη
Check box	Εδώ τσεκάρουμε περισσότερες από μια επιλογές που θέλουμε .Μπορούμε να θέσουμε να είναι ενεργό(enable) η όχι κάποιο από αυτά όταν επιλέγουμε να μεσάζονται αυτά από μια επιλογή.
Edit text	Εδώ θα απεικονίζετε μια πληροφορία που θα μεταβάλετε από τον χρήστη όταν αυτός εκτελεί μια άλλη διαδικασία με κάποιο άλλο αντικείμενο
Static text	Στατικό κείμενο μια λέξη η μια φράση
Pop up menu	Εδώ επιλέγουμε μια σειρά από επιλογές. Όταν δεν έχει πατηθεί θα δείχνει την τελευταία μας επιλογή
List box	Μας δίνει την δυνατότητα να επιλέξουμε τις τιμές από μια λίστα αλλά και το αποτέλεσμα που δίνουν αυτές να το αποκοτήσουμε κάπου
Toggle button	Το toggle button μπορεί να έχει δυο τιμές on=1 off=0.Η διαδικασία(Callback εκτελείτε κάθε φορά που πατάμε το κουμπί. Μπορεί να εκτελούνται δυο διαφορετικές ενέργειες από την ίδια διαδικασία πατώντας κάθε φορά το κουμπί
axes	Είναι ένα στοιχείο για την τοποθέτηση γραφικών(figure) εικόνων κ.α. .Δεν βγαίνει κάποια συνάρτηση διαδικασία(Callback) στο m file για αυτό.

Πίνακας 4.1 Βασικά αντικείμενα του GUI



Εικόνα 4.47 Επιθεωρητής ιδιότητας αντικειμένου

Background color	Ορίζω χρώμα φόντου
Enable	Ορίζω αν το στοιχείο είναι ενεργό η όχι
Font name	Όνομα γραμματοσειράς
Font size	Μέγεθος γραμματοσειράς
Font weigh	Το στυλ των γραμμάτων πχ bold,normal κ.α
Horizontal Alignment	Οριζόντια στοίχιση
Position	Ορίζω τις διαστάσεις και την θέση του αντικειμένου

x	y	Οριζόντια απόσταση από την κάτω αριστερή γωνιά	Καθετή απόσταση από την αριστερή γωνιά
width	height	Πλάτος αντικειμένου	Ύψος
String		Ιδιότητα που ορίζει τι θα γραφεί πάνω του το αντικείμενο πχ(όνομα)	
Tag		Η ετικέτα που θα έχει το αντικείμενο μέσα στον mfile του GUI	
Units		Μονάδα μέτρησης του αντικείμενου πχ points centimeters κ.α.	
Visible		Αν είναι ορατό η αόρατο το στοιχείο on off	

Πίνακας 4.2 Ιδιότητες των αιτιώμενων

4.7.2 Κώδικας

Αρχικά όταν ανοίγει ο editor με τον κώδικα βλέπουμε τα εξής:

```
% --- Executes just before equalizer is made visible.
function equalizer_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to equalizer (see VARARGIN)
% Choose default command line output for equalizer
handles.output = hObject;
% Update handles structure
guidata(hObject, handles);
% UIWAIT makes equalizer wait for user response (see UIRESUME)
% uiwait(handles.figure1);
```

Η πρώτη συνάρτηση `function equalizer_OpeningFcn(hObject, eventdata, handles, varargin)`, που είναι η αρχική συνάρτηση, όταν ανοίγει κάθε m-file του GUI. Ουσιαστικά την χρησιμοποιούμε για να εκτελούνται διάφορες διεργασίες όπως το να δημιουργήσει όλη την πληροφορία του αρχείου η διαφορά αρχεία εξωτερικά αυτού του αρχείου, πριν δει τρέξει ο χρήστης αυτό που σχεδιάζει (στην προκειμένη περίπτωση ο equalizer όπως το ονομάσαμε αρχικά). Στην συνέχεια βλέπουμε άλλες δυο συναρτήσεις

1) `handles.output = hObject;`

Αυτή η συνάρτηση είναι μια δομή-αντικείμενο, έχει όλες τις εξόδους όλων των αντικειμένων (`hObject`) που αυτά είναι τα sliders, edit box, graphs, buttons κ.α. και που το κάθε αντικείμενο έχει το δικό του tag όπως το ορίσαμε στην καρτέλα επιθεωρητής ιδιότητας πχ το load button το ορίσαμε ως Load τότε αυτό το αντικείμενο θα είναι στο handles σαν Load. Οπότε κάθε αντικείμενο είναι ένα `hObject`. Κάθε φορά που θα ανοίγει το αρχείο equalizer.m όλα τα αντικείμενα που βάλουμε θα έχουν αποθηκευτεί σε αυτήν την συνάρτηση (handles).

Η εντολή `guidata` είναι για να αποθηκεύονται οι καινούργιες αλλαγές που θα κάνουμε στο αντικείμενο handles. Πριν ακόμα τοποθετήσουμε τον κώδικα που θα καθορίζει τις λειτουργίες του αντικειμένου (`hObject`) στον editor θα εμφανιστεί ο παρακάτω κώδικας

```
% --- Executes on slider movement.
function slider5_Callback(hObject, eventdata, handles)
% hObject    handle to slider5 (see GCBO)
```

```

% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%         get(hObject,'Min') and get(hObject,'Max') to determine range of
slider

```

Στην συνέχεια τοποθετούμε τον αντίστοιχο κώδικα για να εκτελέσει το αντικείμενο την αντίστοιχη λειτουργία

4.7.3 Slider callback

```

% --- Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
% hObject handle to slider1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%         get(hObject,'Min') and get(hObject,'Max') to determine range of
slider

G1=get(handles.slider1,'Value');
set(handles.C1_val,'String',num2str(G1));
% --- Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
% hObject handle to slider1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called
% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

```

Κάθε αντικείμενο το οποίο τοποθετείται μέσα στο περιβάλλον Gui μπορεί να ελέγχετε-καθορίζετε από τον χρήστη. Μέσα στην συγκεκριμένη συνάρτηση `function slider1_Callback` τοποθετούνται τα χαρακτηριστικά του αντικειμένου τα οποία θα πραγματοποιούνται όταν ο χρήστης επιλέξει να αλληλοεπιδράσει με το αντικείμενο και έτσι θα προκληθεί μια διαδικασία -ένα αποτέλεσμα όταν ο χρήστης επιλέξει στην προκειμένη περίπτωση να αλληλοεπιδράσει με το slider. Κάθε φορά που ο χρήστης με το ποντίκι μετακινεί το slider εκτελείτε ο κώδικας για το slider και έτσι μιλάμε για μια επανάκληση=callback

Στο πρώτο κομμάτι “Executes on slider movement” εκεί θα καθορίζεται η λειτουργία του slider . Στην πρώτη σειρά λέω: πάρε την μεταβλητή G1 η οποία θα αποτελέσει τιμή στο slider1(tag=ετικέτα του 1^{ου} slider) η αυτό θα διαβάσει την G1 μέσω της get .Η get παίρνει σαν είσοδο το όνομα του αντικειμένου που θα διαβαστεί η πληροφορία και τον τύπο της πληροφορίας που θα διαβαστεί. Έτσι μπορώ να αλλάζω το κέρδος όταν θα αλλάζω και την τιμή του slider.

Στην δεύτερη σειρά θέσε-βάλε το G1, όταν αυτό θα αλλάζει από το slider , στο αντικείμενο edit text (handles.C1_val) η το edit text που έχω ορίσει θα δείχνει το μεταβαλλόμενο G1.Η set παίρνει είσοδο το αντικείμενο στο οποίο θα αλλάζει η πληροφορία τον τύπο της πληροφορίας και την πληροφορία Έτσι το edit text(C1_val) θα απεικονίζει την τρέχουσα τιμή του slider .Το μεταβλητό G1 αποτελεί ‘String’ τιμή στο c1_val, εκφραζόμενο σε ‘string’ όπου η αρχική ‘String’ τιμή του c2_val ήταν 0.

Το num2str με δίνει την δυνατότητα να ορίζετε το κέρδος και σε δεκαδικά αλλιώς fix (G1)
Αντίστοιχες είναι οι ρυθμίσεις για τις 10 μπάντες (slider1-slider10)

4.7.4 Edit box Callback

```
function C1_val_Callback(hObject, eventdata, handles)
% hObject    handle to C1_val (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of C1_val as text
%        str2double(get(hObject,'String')) returns contents of C1_val as a
double

% --- Executes during object creation, after setting all properties.
function C1_val_CreateFcn(hObject, eventdata, handles)
% hObject    handle to C1_val (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

Εδώ βγαίνει το callback δηλαδή η διαδικασία του κώδικα που εκτελείτε για το edit box (το κουτί που θα δέχεται την μεταβλητή τιμή από το slider).Εδώ δεν ορίζω κάτι μιας και η εντολή για να θέσει την τιμή του slider στο tag C1_val την ορίζω μέσα στο slider1.

4.7.5 Κουμπιά

Play

Η διαδικασία για το κουμπί play περιλαμβάνει όλα τα

```
% --- Executes on button press in Play.
function Play_Callback(hObject, eventdata, handles)
% hObject    handle to Play (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of Play
global file_name;
[x,fs]=wavread(file_name);
```

Και εδώ θέτουμε ως καθολικής εμβέλειας (global)την μεταβλητή “file name” όπως και στο κουμπί load για είναι διαθέσιμη η μεταβλητή που θα πάρω από το κουμπί load και να την διαβάσω με μια wavread.

Έχει τα χαρακτηριστικά των φίλτρων πχ

```
%filtro1
fc1=31;
fs=44100;
G1=get(handles.slider1,'Value');
[ Hd1 ] = lshelving( G1,fc1,fs )

%filtro4
fc4=250;
fs=44100;
G4=get(handles.slider4,'Value');
q=4;
[ Hd4 ] = peakfilter( G4,fc4,q,fs )
```

Η διαδικασία ορισμού για τις άλλες οχτώ περιοχές του σήματος εισόδου είναι ίδια και για τα αλλά οχτώ φίλτρα.

Έχει τα δυο plot το αριστερά(axes1) για απόκριση συχνότητάς και το δεξιά axes(2) για το φάσμα και οι αντίστοιχες συναρτήσεις Matlab.

```
sound(y1,fs);
wavwrite(y1,fs,'out.wav')
```

Να ακούσουμε το φιλτραρισμένο σήμα και να το αποθηκεύσουμε

Load

```
% --- Executes on button press in Load.
function Load_Callback(hObject, eventdata, handles)
% hObject    handle to Load (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of Load
global file_name;
[file_name,pathname]=uigetfile('*.wav');
```

Εδώ βλέπουμε την διαδικασία του κώδικα που θα εκτελεστεί για το κουμπί load

Αρχικά η μεταβλητή filename ορίζετε ως καθολικής εμβέλειας δηλαδή για να μπορέσω να την δηλώσω και στο play(και στην συνάρτηση play που είναι διαφορετική συνάρτηση από την load) έτσι κάθε φορά η play να παίρνει την συγκεκριμένη μεταβλητή(filename) που έχω αναθέσει να πάρει από την συνάρτηση load αν και στην play είναι global δηλαδή για να περαστεί η τιμή της μεταβλητής σε δυο συναρτήσεις -κουμπιά θα πρέπει αυτή η μεταβλητή να οριστεί ως global και στα δυο κουμπιά έτσι ώστε να είναι διαθέσιμη η μεταβλητή και στα δυο κουμπιά.

Με την uigetfile θα εμφανιστεί το παράθυρο από το οποίο θα επιλέξω το filename δηλαδή το αρχείο wav και μετά να το ανοίξω με το open.

Τα αρχεία θα είναι τύπου wav που θα υπάρχουν στον συγκεκριμένο φάκελο που τρέχει και η εφαρμογή Αυτό γίνεται επειδή θέτω με το *.wav που θα με δώσει όλα τα αρχεία τύπου .wav που βρίσκονται στο path του φακέλου.

Zero gain

```
% --- Executes on button press in setzerogain.
function setzerogain_Callback(hObject, eventdata, handles)
```

```

% hObject handle to setzerogain (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

```

```

% Hint: get(hObject,'Value') returns toggle state of setzerogain

```

```

set(handles.C1_val,'string',num2str(0));
set(handles.C2_val,'string',num2str(0));
set(handles.C3_val,'string',num2str(0));
set(handles.C4_val,'string',num2str(0));
set(handles.C5_val,'string',num2str(0));
set(handles.C6_val,'string',num2str(0));
set(handles.C7_val,'string',num2str(0));
set(handles.C8_val,'string',num2str(0));
set(handles.C9_val,'string',num2str(0));
set(handles.C10_val,'string',num2str(0));
set(handles.slider1,'value',0);
set(handles.slider2,'value',0);
set(handles.slider3,'value',0);
set(handles.slider4,'value',0);
set(handles.slider5,'value',0);
set(handles.slider6,'value',0);
set(handles.slider7,'value',0);
set(handles.slider8,'value',0);
set(handles.slider9,'value',0);
set(handles.slider10,'value',0);

```

Και εδώ με την set θα αλλάξουμε την πληροφορία και των sliders και των edit box και ουσιαστικά θα μηδενίσουμε τις τιμές τους. Θέτει το νούμερο 0 στο c1_val(tag για το κουτάκι κάτω από το slider που με δείχνει το τρέχον κέρδος του slider) και μηδενίζει την τιμή του slider1 που έχει εκείνη την στιγμή).

Stop

```

% --- Executes on button press in Stop.
function Stop_Callback(hObject, eventdata, handles)
% hObject handle to Stop (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

```

```

% Hint: get(hObject,'Value') returns toggle state of Stop
clear sound
end

```

Εδώ αυτό που κάνουμε είναι να ‘σκοτώνουμε’ το σήμα εισόδου κάθε φορά που πατάμε το κουμπί stop. Έτσι θα αρχίζει κάθε φορά που αλλάζουν τα ποτενσιόμετρα, από την αρχή. Ωστόσο δεν θα ξανά ακούσουμε την είσοδο όπως ήταν αρχικά. Γιατί η sound δέχεται τις είδη υπάρχουσες τιμές των sliders. Για να ακούσουμε το αρχικό σήμα εισόδου πρέπει να πατήσουμε το κουμπί zero gain.

4.7.6 Απόκριση συχνότητας

```
axes(handles.axes1);
hdcas=dfilt.cascade(Hd1,Hd2,Hd3,Hd4,Hd5,Hd6,Hd7,Hd8,Hd9,Hd10);
d=[zeros(1,0),1,zeros(1,44100)];
y=filter(hdcas,d);
fft1 = fft(y);
f = linspace(0, fs, length(fft1));
h_fft11=20*log10(abs(fft1));
plot(f,h_fft11,'lineWidth',1,'Color',[1 0 0])
axis([0,fs/2,-20,max(abs(h_fft11))]);
ylabel('Magnitude response(dB)');
xlabel('Frequency(kHz)');
```

Ένας τρόπος που μπορώ να πάρω την απόκριση συχνότητάς είναι αρχικά δημιουργώ ένα ‘‘συνολικό φίλτρο’’ (σύνδεση 10 περιοχών φίλτρων σε σειρά) που προκύπτει από την σύνδεση σε σειρά 10 φίλτρων- περιοχών hdcas. Ορίζω μια $\delta(n)$ με ένα εύρος 0 έως 44100. Τόσα θέλω να είναι και τα δείγματα που θα με δώσει ο fft γιατί το σήμα εισόδου(d) θα έχει και αυτό 44100 δείγματα. Κάνω μια filter όπου y το σήμα εξόδου(κρουστική απόκριση) του φίλτρου με είσοδο $\delta(n)$ και μετασχηματισμό Fourier στο y τότε παίρνω την απόκριση συχνότητας.

Εδώ η f καθορίζει τα όρια της συχνότητας 0 έως 44100 και τα δείγματα του $\text{fft1}(\text{length}(\text{fft})=44100)$. Για την απεικόνιση σε db. για να πάρω το μετρό του fft τότε $\text{abs}(\text{fft1})$ και μετατροπή σε db $20*\log_{10}$

```
axis([0,fs/2,-20,max(abs(h_fft11))]);
```

θα εμφανίσει μόνο συχνότητες που βρίσκονται πριν από $fs/2$

4.7.7 Φάσμα

```
axes(handles.axes2);
y1=filter(hdcas,x);
f = linspace(0, fs, length(Y));
idx = 1:length(Y) / 2 + 1;
Y=fft(y1);
plot(f(idx),20*log10(abs(Y(idx))))
xlabel('Frequency (khz)')
ylabel('magnitude (db)')
sound(y1,FS)
```

Εδώ η f καθορίζει τα όρια της συχνότητας 0 έως 44100 και θα έχω δείγματα όσο είναι και το $\text{length}(Y)$. Αλλά θέλουμε τα μισά δείγματα πάλι για 44100 οπότε με το διάνυσμα idx για να χρησιμοποιήσω τα μισά δείγματα αφού πάνω από $\text{fs}/2$ τα δείγματα είναι το ίδιο με τα προηγούμενα οπότε θέλω τα μισά. Γενικά το τελευταίο δείγμα από τα μισά δείγματα του Y θα πρέπει να είναι στο $\text{fs}/2$ για είναι σωστό το φάσμα.

5.8 Συμπεράσματα

Τα φίλτρα τα οποία είναι καταλληλά και χρησιμοποιούνται σε ακουστικές εφαρμογές όπως σε έναν γραφικό ισοσταθμιστή μεταβλητού κέρδους ν περιοχών είναι IIR (infinite impulse response) άπειρης κρουστικής απόκρισης. Αυτά είναι τα shelving (low και high frequency) και τα peaking η notch (bandpass και bandstop). Αυτά έχουν ως βάση τους τα low/high/band pass αλλά ενσωματωμένο και το αντίστοιχο κέρδος (\pm) πλατους πάνω από την συχνότητα αποκοπής (-3db.) και έτσι δίνουν την δυνατότητα του μεταβλητού θετικού και αρνητικού κέρδους επηρεάζοντας μια συγκεκριμένη περιοχή συχνοτήτων. Συνήθως τα shelving είναι 1^{ης} τάξης και τα peaking/notch 2^{ης} τάξης είναι Butterworth φίλτρα. Αυτό το αντιλαμβανόμαστε αφού δεν υπάρχει κυμάτωση στις δυο ζώνες (διέλευσης και αποκοπής). Στην παραπάνω πτυχιακή αναπτύσσονται θεωρητικά τρεις τρόποι σχεδιασμοί για το κάθε φίλτρο και χρησιμοποιούνται προγραμματιστικά δυο. Στη μεν πρώτη δεν θεωρείτε παραμετρική μέθοδος. Οποιαδήποτε αλλαγή στους συντελεστές της συνάρτησης μεταφοράς θα πρέπει να ξανά υπολογίσω όλους τους 5 συντελεστές της συνάρτησης μεταφοράς ($b_0, b_1, b_2/a_1, a_2$) για 2^{ης} τάξης φίλτρο. Ωστόσο στην εφαρμογή με τα ρυθμιστικά (sliders) μπορώ να μεταβάλω το κέρδος κρατώντας τους άλλους συντελεστές σταθερούς (μιας και δεν με νοιάζει να αλλάξω την συχνότητα αποκοπής αφού κάθε φίλτρο (10 περιοχές) έχει την δική του σταθερή συχνότητα αποκοπής). Στην δεύτερη μέθοδο που είναι ημιπαραμετρική (semi-parametric) έχουμε την δυνατότητα να ορίσω το κέρδος που είναι ανεξάρτητο από την ω_c . Έτσι μπορώ να αλλάξω το κέρδος χωρίς να ξαναυπολογίσω όλους τους άλλους 5 συντελεστές. Όπως καταλαβαίνουμε από την εξίσωση εάν αλλάξουμε το ω_c θα αλλάξει και η συνάρτηση μεταφοράς του φίλτρου.

Όμως με την υλοποίηση του φίλτρου με ένα allpass έχουμε την δυνατότητα (μιας και σε αυτό μπορούμε να αλλάξουμε την συχνότητα αποκοπής υπολογίζοντας μόνο ένα παράγοντα το (c) αυτή του η ιδιότητα περνάει και στον σχεδιασμό ενός shelving και peaking/notch οπότε και εδώ έχουμε την δυνατότητα να αλλάξουμε το κέρδος H_0 και την κεντρική συχνότητα αποκοπής (ω_c) υπολογίζοντας μόνο δυο συντελεστές για 2^{ης} τάξης φίλτρο και αυτοί είναι το $H_0 = V_0 - 1$ για το κέρδος και για την συχνότητα αποκοπής ω_c το $C(B/C)$ (για boost/cut) για low/high shelving αντίστοιχα.

Εξίσου το ίδιο συμβαίνει και για peaking φίλτρο μέσω του bandpass που υλοποιείται με ένα allpass (τρίτη μέθοδος) και έτσι μπορούμε να αλλάξουμε δυο συντελεστές (το ω_c και το εύρος ζώνης) χωρίς να υπολογίσουμε ξανά τους άλλους δυο συντελεστές και κάτ. επέκταση ολόκληρη την συνάρτηση μεταφοράς. Προφανώς το ίδιο συμβαίνει με το κέρδος V_0 που ορίζετε είδη ξεχωριστά. Ουσιαστικά με την χρήση ενός allpass μπορώ να σχεδιάσω το φίλτρο μου και να είναι ολοκληρωτικά παραμετρικό (full parametric) γλιτώνοντας καινούργιους υπολογισμούς για τους άλλους συντελεστές σε σχέση με την συνάρτηση μεταφοράς του πρώτου σχεδιασμού.

Η επιλογή του καταλλήλου φίλτρου έχει να κάνει ανάλογα σε ποια εφαρμογή θα χρησιμοποιηθούν. Δηλαδή εάν έχουμε μια εφαρμογή που θέλουμε συχνές αλλαγές των παραμέτρων του φίλτρου (παραμετρικός equalizer) και να αποθηκεύουμε αυτά τα αποτελέσματα των αλλαγών, τότε η τρίτη μέθοδος είναι καλύτερη. Αν μας νοιάζουν οι υπολογισμοί του φίλτρου και θέλουμε μόνο μια παράμετρο να αλλάξουμε πχ το κέρδος (V_0) γραφικός ισοσταθμιστής) τότε επιλέγουμε την πρώτη που από πλευράς σχεδιασμού είναι πιο απλή σε σχέση με τις άλλες δυο και για αυτό επιλέχθηκε για τον σχεδιασμό του παραπάνω ηχητικού συστήματος.

Στην εφαρμογή τα φίλτρα συνδέονται σε σειρά γιατί εφαρμόζοντας την παράλληλη σύνδεση δυο φίλτρων παρατηρούμε ότι δεν παίρνω την επιθυμητή απόκριση συχνότητας των φίλτρων δηλαδή

βλέπω ότι στις δυο περιπτώσεις με δίνει παραπάνω κέρδος για ενίσχυση η εξασθένηση κάτι που δεν είναι ίδιο με το είδη ορισμένο .Η δομή υλοποίησης του κάθε φίλτρου είναι Direct form II (κανονικοποιημένη μορφή) για περισσότερη οικονομία υλικού.

Τα φίλτρα αυτά χαρακτηρίζονται ως φίλτρα ελάχιστης φάσης(πόλοι και μηδενικά βρίσκονται εντός του μοναδιαίου κύκλου).Προφανώς εάν αντιστρέψουμε την συνάρτηση μεταφοράς για να πάρουμε την περίπτωση cut του φίλτρου(δηλαδή αντιστρέψουμε τις θέσεις των πόλων και των μηδενικών) τότε και αυτό με την σειρά του θα είναι ένα ευσταθές φίλτρο γιατί και η αρχική συνάρτηση μεταφοράς του φίλτρου ήταν ελάχιστης φάσης (αν $b_k < 1$ μηδενικά ελάχιστης φάσης οπότε αυτά θα είναι εντός του μοναδιαίου κύκλου που αυτά θα αντιστραφούν με τους πόλους που βρίσκονται επίσης εντός του μοναδιαίου κύκλου και $a_k < 1$ πόλοι) οπότε $H(z) = \frac{B(z)}{A(z)}$ με $z=1$ (άρα ευσταθές) τότε και

$H^{-1} = \frac{A(z)}{B(z)}$ (άρα ευσταθές) οπότε ένα φίλτρο ελαγιστης φασης τοτε το αντίστροφο του θα είναι επίσης ένα ευσταθές φίλτρο .

Η εφαρμογή που υλοποιήθηκε στο GUI του Matlab είναι ένας ισοσταθμιστής 10 περιοχών. Η μεταβολή των ποτενσιόμετρων δηλαδή η μεταβολή του κέρδους δεν είναι real time. Έτσι θα πρέπει να πατήσω stop να και μετά να μεταβάλουμε το κέρδος για να ξανακούσουμε την φιλτραρισμένη απόκριση

Μια ιδέα που θα μπορούσε να χρησιμοποιηθεί για περαιτέρω έρευνα και έχοντας το καταλληλο υποβαθρο είναι να σχεδιάσουμε τα φίλτρα έτσι ώστε οι παράμετροι τους (ο κάθε ένας ξεχωριστά) να μεταβάλετε στο χρόνο από μια αρχική κατάσταση σε μια τελική χ το κέρδος g του φίλτρου να είναι από -15 έως 15 db . Έτσι αναμεσα σε αυτό το εύρος θα υπάρχει μια μεταβατική κατάσταση που εκεί η παράμετρος g θα μεταβάλλεται ανάμεσα στο σταθερό εύρος -15 και 15 db Αυτές τις ενδιάμεσες καταστάσεις τις συμβολίζουμε με μια παράμετρο χ που μας δείχνει πόσες καταστάσεις θα έχει αυτή η παράμετρος $g(x)$.Σίγουρα ο αριθμός αυτών των ενδιάμεσων καταστάσεων παίζει ρόλο γιατι μπορεί να προκαλέσει κάποια προβλήματα στον τελικό φιλτραρισμένο σήμα αλλά ούτε πολύ μικρός γιατί λογικά θα υπάρχει πρόβλημα στην μετάβαση από μια κατάσταση στην άλλη. Έχοντας ως βάση αυτό το σκεπτικό μπορούμε να σχεδιάσουμε ένα real time ισοσταθμιστή

Η εφαρμογή θα μπορούσε να εξελιχθεί περισσότερο δηλαδή ο ισοσταθμιστής να είναι περισσότερων περιοχών- ζωνών μεταβλητού εύρους που μπορεί να είναι τριτοοκταβες και ορίζετε ως $\frac{f_2}{f_1} = 2^{1/3}$ δηλαδή 31 περιοχές φιλτραρίσματος που αυτό είναι πολύ καλύτερο σε σχέση με 10 περιοχές ενός ισοσταθμιστή που θεωρείτε μη επαγγελματικός. Επίσης βελτίωση του θα ήταν να τοποθετηθούν κουμπιά όπως pause και resume έτσι ώστε να είναι δυνατόν η παύση του φιλτραρισμένου σήματος αλλαγή των ποτενσιόμετρων και η συνέχεια του φιλτραρισμένου σήματος με τις νέες αλλαγές .

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Gerard Blanchet & Murice Charbit DSP *Digital signal and image processing using matlab* Wiley-ISTE .2006
- [2] John G.Proakis Dimitris G Manolakis.*DIGITAL SIGNAL PROCESSING. (Principles,Algorithms and Applications)*.PEARSON Practice Hall New Jersey.2007
- [3] Leland B.Jackson Signals Systems Transforms .Addison-Wesley Publishing Company.1991
- [4] MONSON H.HAYES .*ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΣΗΜΑΤΟΣ* .ΤΖΙΟΛΑ .1999
- [5] ΔΗΜΗΤΡΗΣ ΣΚΑΡΛΑΤΟΣ . *ΕΦΑΡΜΟΣΜΕΝΗ ΑΚΟΥΣΤΙΚΗ*. GKOTSIS. 2008
- [6] Udo Zolser .*Digital Audio Signal Proccessing* . A John Wiley & Sons, Ltd, Publication. 2008
- [7] Udo zolser *DAFX DIGITAL Audio Effects*. A John Wiley & Sons, Ltd, Publication .2002

PDF

- [1] Sophocles J. Orfanidis . *INTRODUCTION TO Signal Processing*. Prentice Hall.2010
- [2] Vinay K. Ingle John & G. Proakis Digital *.Signal Processing Using MATLAB®* CL Engineering 3 edition. 2011
- [3] Sen M kuo Bob H Lee Real-Time Digital Signal Processing
- [4] J. N. MOURJOPOULOS, E. D. KYRIAKIS-BITZAROS, AND C. E. GOUTIS *Theory and Real Time Implementation of Time-Varying Digital Audio Filters*
- [5] *.Signal Processing Toolbox™ User's Guide*
- [6]. http://www.music.mcgill.ca/~ich/classes/dafx_book.pdf
- [7] MATLAB® Creating Graphical User Interfaces
- [8] ΑΘΑΝΑΣΙΟΣ ΣΚΟΔΡΑΣ & ΒΑΣΙΛΕΙΟΣ ΑΝΑΣΤΑΣΟΠΟΥΛΟΣ *Ψηφιακή Επεξεργασία Εικόνων και Σημάτων* .ΤΥΡΟΡΑΜΑ.2003
- [9]. Γεώργιος Β Μουστακιδης Βασικές Τεχνικές Ψηφιακής Επεξεργασίας Σημάτων 2003
- [10] Δημήτρης Βαρσάμης. *MATLAB - Graphical User Interfaces* 2010
- [11] PHILLIP A. REGALIA & SANJIT K. MITRA *Tunable Digital Frequency Response Equalization Filters* 1987

Google βιβλία

- [1] Fred taylor Digital Filters: *Principles and Applications with MATLAB*
- [2] Steven Smith Digital Signal Processing. A Practical Guide for Engineers and Scientists.
- [3]. G. Dimopoulos *Analog Electronic Filters: Theory, Design and Synthesis Hercules*
- [4]. Lars Wanhammar *Analog Filters using MATLAB*
- [5.] C. Whitaker *The Electronics Handbook Jerry*
- [6]. Stefan Niewiadoms *Filter Handbook: A Practical Design Guide*
- [7] Maurice Bellanger *Adaptive Digital Filters*
- [8]. Technologies Perez-Meana *Advances in Audio and Speech Signal Processing*
- [9]. Jason corey *.Audio Production and Critical Listening: Technical Ear Training*
- [10]. Ramesh Babu & C. Durai *Digital Signal Processing*
- [11]. J.S.Chitode *Digital Signal Processing*
- [12]. Emmanuel C *Digital Signal Processing: A Practical Approach*
- [13]. Robert Schilling *Fundamentals of Digital Signal Processing Using MATLAB*
- [14]. Dan Hosken *Music Technology and the Project Studio: Synthesis and Sampling*

Url

- [1] <http://www.mathworks.com/help/signal/ref/fdesign.lowpass.html>
- [2] <http://www.mathworks.com/help/signal/ug/opening-fdatool.html>
- [3] <http://www.mathworks.com/help/signal/ref/bilinear.html>
- [4] <http://www.mathworks.com/help/signal/ref/firpmord.html>
- [5] <http://www.mathworks.com/help/signal/ug/iir-filter-design.html>
- [6] <http://www.mathworks.com/help/signal/ug/special-topics-in-iir-filter-design.html>
- [7] <http://www.soundonsound.com/sos/jul01/articles/equalisers1.asp#>
- [8] <http://www.soundonsound.com/sos/jul01/articles/equalisers1.asp>
- [9] <http://ocw.mit.edu/resources/res-6-008-digital-signal-processing-spring-2011/video-lectures/lecture-12-network-structures-for-infinite-impulse-response-iir-systems/>
- [10] <http://ocw.mit.edu/resources/res-6-008-digital-signal-processing-spring-2011/video-lectures/lecture-13-network-structures-for-finite-impulse-response-fir-systems-and-parameter-quantization-effects-in-digital-filter-structures/>
- [11] <http://www.hometracked.com/2008/01/31/eq-cut-narrow-boost-wide/>
- [12] <http://electronics.howstuffworks.com/how-to-understand-sound-equalizer.htm>
- [13] <http://www.presonus.com/community/learn/equalizer-terms-and-tips>
- [14] <http://ethanwiner.com/equalizers.html>
- [15] http://www.reddit.com/r/explainlikeimfive/comments/1ogr7o/eli5_how_to_correctly_use_an_equalizer/
- [16] <http://www.audio-issues.com/music-mixing/eq-bass-harmonics/#>
- [17] <http://www.cirrusresearch.co.uk/blog/2011/11/what-are-octave-and-third-octave-band-filters-on-a-sound-level-meter/>
- [18] <http://www.mathworks.com/matlabcentral/answers/10842-how-to-set-a-value-from-code-or-callback-to-a-edit-text-box>
- [19] <http://www.mathworks.com/matlabcentral/answers/81670-updating-edit-text-box-in-gui>
- [20] <https://www.youtube.com/watch?v=ctC1Q3VBsBM>
- [21] <https://www.youtube.com/watch?v=aj3oDssjYyQ>
- [22] <https://www.youtube.com/watch?v=5RoPKr94-w&spfreload=10>
- [23] https://www.youtube.com/watch?v=JnRy_ULbrpQ
- [24] https://www.youtube.com/watch?v=RFXmu2e8_wc&spfreload=10
- [25] <http://www.mathworks.com/discovery/matlab-gui.html>
- [26] <http://www.soundonsound.com/>
- [27] http://www.hometoys.com/content.php?post_type=1511

Παράρτημα

Κωδικας 1

```
function [ Hd ] = lshelving( G,fc,fs )
%UNTITLED4 Summary of this function goes here
% Detailed explanation goes here
wc=2*pi*fc; % syxnothta apokopis rad/sec analogikou filtrou
V0=10^(G/20); % kerdos se voltage
if G>0
bs=[1, V0*wc];
as=[1, wc]; %arithmitis+paronomastis stou analogiko pedio boost
else G<0
bs=[1, wc];
as=[1, wc/V0]; %arithmitis+paronomastis stou analogiko pedio cut

end;
[b,a] = bilinear(bs, as, fs, fc) % digramikos metasxhmatismos fc einai
[sos,g]=tf2sos(b,a);
Hd=dfilt.df2sos(sos,g);
end
```

Κωδικας 2

```
function [ Hd ] = hshelving( G,fc,fs )
%UNTITLED4 Summary of this function goes here
% Detailed explanation goes here
wc=2*pi*fc; %γωνιακη συχνοτητα % rad/sec
V0=10^(G/20);
if G>0
bs=[V0, wc];
as=[1, wc];
else G<0
bs=[1, wc];
as=[1/V0, wc];
end;
[b,a] = bilinear(bs, as, fs, fc) % digramikos metasxhmatismos
[sos,g]=tf2sos(b,a);
Hd=dfilt.df2sos(sos,g);
End
```

Κωδικας 3

```
function [ Hd ] = peakfilter( G,fc,q,fs )
%UNTITLED4 Summary of this function goes here
% Detailed explanation goes here
wc=2*pi*fc; % rad/sec
V0=10^(G/20); % kerdos periptosi boost
if G>0
bs=[1, wc*V0/q, wc^2];
as=[1, wc/q, wc^2];
else
bs=[1, wc/q, wc^2];
as=[1, wc/V0/q, wc^2];
end; % G<0
[b,a] = bilinear(bs, as, fs, fc); % digramikos metasxhmatismos fc einai h
prewarp frequency pou teriazai kai prin ton digramiko afou wc=2*pi*fc
rad/sec alla kai meta
[sos,g]=tf2sos(b,a);
Hd=dfilt.df2sos(sos,g);
end
```

Κωδικός 4

```
function [ y ] = peakifilt( x,Wc,Wb,G )
%UNTITLED2 Summary of this function goes here
% Detailed explanation goes here
V0=10^(G/20);
H0=V0-1;H01=H0/2
if G>0
    c=[tan(pi*Wb/2)-1]/[tan(pi*Wb/2)+1]; boost bandwidth
else G<0
    c=[tan(pi*Wb/2)-V0]/[tan(pi*Wb/2)+V0]; cut bandwidth
end;
d=-cos(pi*Wc); syxnottha apokopis 2*pi*fc/fs rad/sample
xh=[0,0];
for n=1:length(x);
    xhnew=x(n)-d*(1-c)*xh(1)+c*xh(2);
    apy=-c*xhnew+d*(1-c)*xh(1)+xh(2);
    xh=[xhnew,xh(1)];
    y(n)=0.5*H0*[x(n)-apy]+x(n);
end
b0=1+(1+c)*H01
b1=d*(1-c)
b2=-c-(1+c)*H01
a0=1
a1=b1
a2=-c
b = [ b0,b1,b2]
a = [ a0,a1,a2]
[sos,g]=tf2sos(b,a);
Hd=dfilt.df2sos(sos,g)
```