

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



**ΑΠΑΛΛΑΚΤΙΚΗ ΕΡΓΑΣΙΑ ΣΤΟ ΜΑΘΗΜΑ
ΕΠΕΞΕΡΓΑΣΙΑ ΣΗΜΑΤΩΝ ΦΩΝΗΣ ΚΑΙ ΗΧΟΥ**
8ο εξάμηνο

ΜΕΛΗ ΟΜΑΔΑΣ:

ΝΙΚΟΛΟΥΤΣΟΠΟΥΛΟΣ ΣΩΤΗΡΙΟΣ Π12108

ΣΤΑΜΑΤΟΥΛΗ ΑΝΑΣΤΑΣΙΑ Π11144

ΠΕΙΡΑΙΑΣ, ΙΟΥΝΙΟΣ 16/6/2016

Για την εργασία μας επιλέξαμε τις ασκήσεις 10.4 και 10.5 από το βιβλίο «Ψηφιακή Επεξεργασία Φωνής, Θεωρία και Εφαρμογές», L.R.Rabiner, R.W.Schafer.
Εργαστήκαμε σε περιβάλλον MATLAB R2015a και R2016a.

Άσκηση 10.4

Εκφώνηση

(Άσκηση MATLAB: Ανιχνευτής Μεμονωμένων Λέξεων βασισμένος σε Κανόνες, Rule-Based Isolated Word Speech Detector).Γράψτε ένα πρόγραμμα σε MATLAB που να ανιχνεύει(απομονώνει) μια λέξη που προφέρεται σε ένα σχετικά ήσυχο και ήπιο ακουστικό περιβάλλον,χρησιμοποιώντας απλούς κανόνες πάνω στις μετρήσεις βραχέος χρόνου,του λογαρίθμου της ενέργειας και του ρυθμού διέλευσης από το μηδέν. Τα αρχεία φωνής που θα χρησιμοποιηθούν στην άσκηση αυτή,μπορούν να μεταφορτωθούν από τον ιστότοπο του βιβλίου και βρίσκονται στο συμπιεσμένο αρχείο ti_isolated_unendpoinited_digits.zip .Μέσα στο αρχείο αυτό,περιέχονται δυο εκφωνήσεις για κάθε μια από τις κυματομορφές των 11 ψηφίων(zero-nine μαζί και oh),ηχογραφημένες σε ένα σχετικά ήσυχο ακουστικό περιβάλλον,με ονομασίες {1-9,O,Z}{A,B}.waV . Έτσι,το αρχείο 3B.waV περιέχει την δεύτερη εκφώνηση του ψηφίου /3/ και το αρχείο ZA.waV περιέχει την πρώτη εκφώνηση του ψηφίου /zero/, κλπ.

Ο στόχος της άσκησης αυτής είναι η υλοποίηση ενός απλού συστήματος αναζήτησης άκρων,βασισμένου σε κανόνες,το οποίο χρησιμοποιεί τις μετρήσεις βραχέος χρόνου του λογαρίθμου της ενέργειας (σε dB) και του ρυθμού διέλευσης από το μηδέν, ανά διαστήματα των 10 msec,για την ανίχνευση των περιοχών του σήματος φωνής και τον διαχωρισμό τους από το σήμα υποβάθρου.

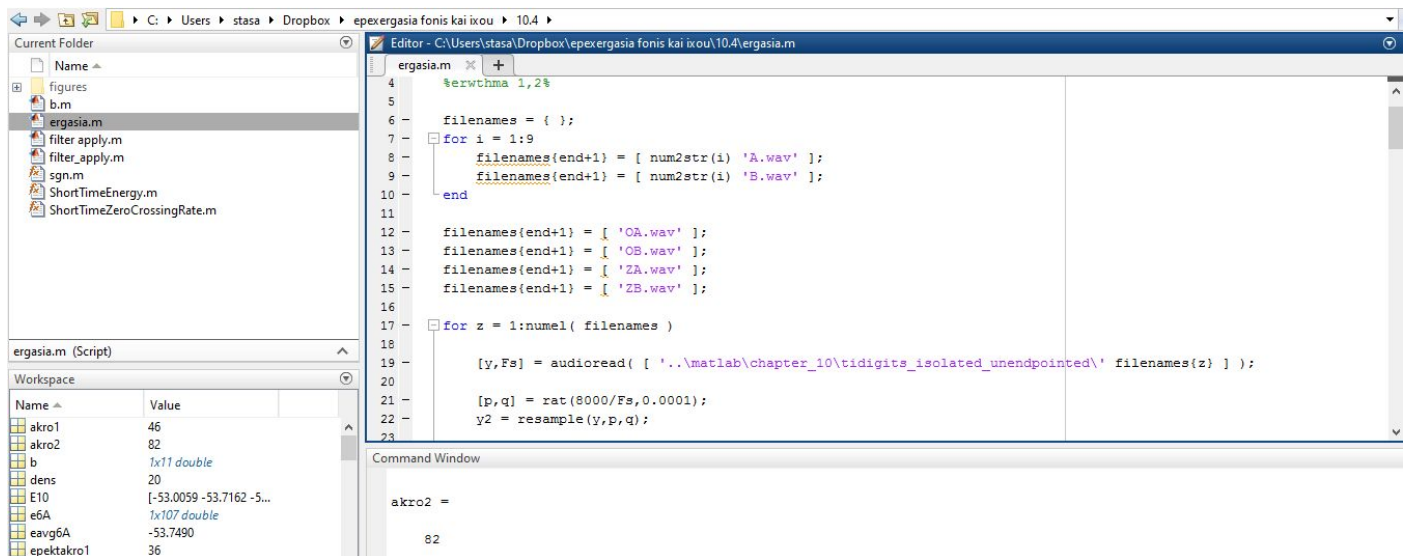
Λύση

Εκτέλεση:

Τα αρχεία βρίσκονται στον φάκελο 10.4 και η λύση είναι στο αρχείο *ergasia.m* .

Βήμα 1ο:Φορτώστε κάθε αρχείο φωνής

Αρχικά φορτώνουμε τα αρχεία φωνής στο περιβάλλον MATLAB όπως φαίνεται στην παρακάτω εικόνα. Στην προκειμένη η συχνότητα δειγματοληψίας των αρχείων είναι $F_s=20000\text{Hz}$.



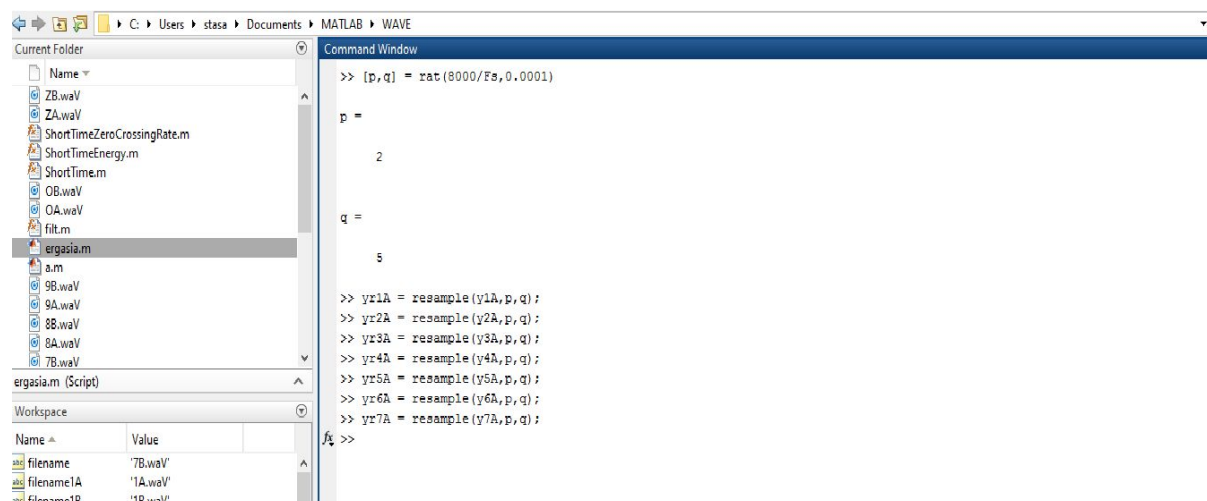
Βήμα 2ο:Μετατρέψτε τον ρυθμό δειγματοληψίας του σήματος εισόδου από $F_s=20000$ δείγματα/sec σε $F_s=8000$ δείγματα/sec

Για την μετατροπή του ρυθμού δειγματοληψίας από $F_s=20000\text{Hz}$ σε $F_s=8000\text{Hz}$ χρησιμοποιήθηκαν οι εντολές $[p,q] = \text{rat}(8000/F_s, 0.0001)$; και $y2 = \text{resample}(y,p,q)$;

.Χρησιμοποιώντας την εντολή rat βρέθηκε η αναλογία των δύο ρυθμών δειγματοληψίας όπου $p=2$ και $q=5$ με σφάλμα 0.0001 και χρησιμοποιώντας την εντολή resample μετατρέψαμε τον ρυθμό δειγματοληψίας σε $F_s=8000\text{Hz}$.

Το στάδιο αλλαγής του ρυθμού δειγματοληψίας περιλαμβάνεται στην επεξεργασία του σήματος φωνής επειδή πολλές κοινές βάσεις δεδομένων δεν χρησιμοποιούν τυπικούς ρυθμούς δειγματοληψίας.

Παρακάτω φαίνεται η διαδικασία της μετατροπής του ρυθμού δειγματοληψίας για τα 22 αρχεία φωνής.



Βήμα 3ο: Σχεδιάστε ένα ζωνοπερατό FIR φίλτρο για την εξάλειψη της DC συνιστώσας, του βόμβου των 60Hz και του θορύβου υψηλής συχνότητας (για συχνότητα δειγματοληψίας, $F_s=8000\text{Hz}$), χρησιμοποιώντας την συνάρτηση σχεδιασμού φίλτρων του MATLAB, `firpm`. Τα χαρακτηριστικά του φίλτρου θα πρέπει να περιλαμβάνουν μια ζώνη αποκοπής που θα αντιστοιχεί στο διάστημα $0 \leq |F| \leq 100\text{ Hz}$, μια ζώνη μετάβασης που θα αντιστοιχεί στο διάστημα $100 \leq |F| \leq 200\text{ Hz}$ και μια ζώνη διέλευσης για το διάστημα $200 \leq |F| \leq 4000\text{ Hz}$. Φιλτράρετε το αρχείο φωνής χρησιμοποιώντας το FIR φίλτρο που προέκυψε.

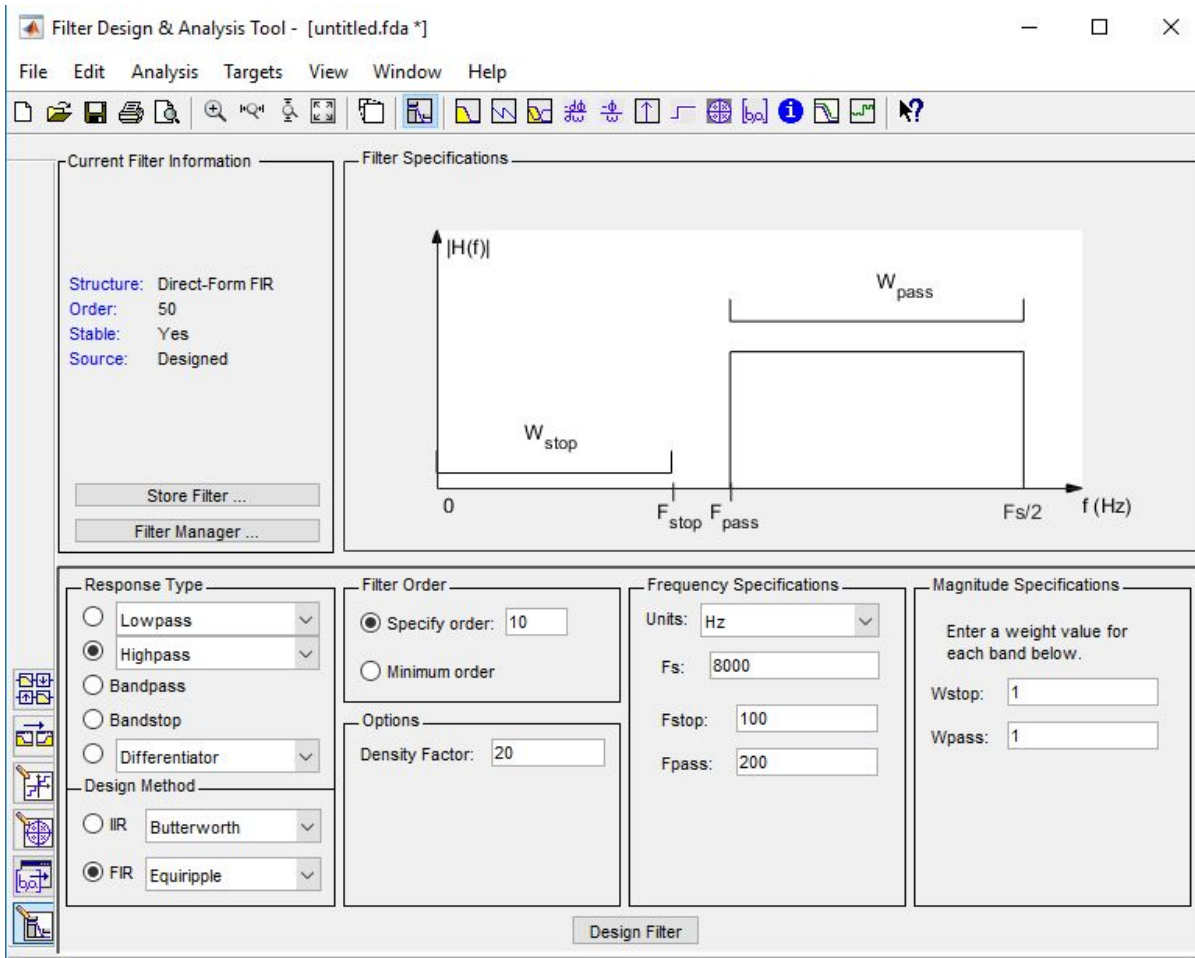
Ένα ψηφιακό φίλτρο είναι ένα γραμμικό χρονικά αμετάβλητο σύστημα διακριτού χρόνου που ενεργεί επιλεκτικά και επιτρέπει ή εμποδίζει την διέλευση ενός σήματος σε μια ορισμένη συχνότητα ή σε μια ορισμένη περιοχή συχνοτήτων. Οι 4 βασικές κατηγορίες είναι οι εξής: χαμηλοπερατά (lowpass), υψιπερατά (highpass), ζωνοπερατά (bandpass) και αποκοπής ζώνης (bandstop).

Για την εξάλειψη της DC συνιστώσας, του βόμβου των 60Hz και του θορύβου υψηλής συχνότητας όπου ζητείται παραπάνω επιλέχθηκε να χρησιμοποιηθεί υψιπερατό φίλτρο που αφήνει αμετάβλητα τα σήματα από μια συχνότητα ως και πάνω ενώ αποκόπτει τα σήματα με συχνότητα μικρότερη της ως.

Συνεπώς, χρησιμοποιώντας το `FDATool` του MATLAB κατασκευάστηκε ανωπερατό FIR φίλτρο με ζώνη αποκοπής $F_{\text{stop}} \leq 100$ και ζώνη διέλευσης $F_{\text{pass}} \geq 400$ έως $F_s/2 = 4000\text{ Hz}$.

Η περιοχή μετάβασης ορίζεται στο διάστημα $100 \leq |F| \leq 200\text{ Hz}$.

Παρακάτω φαίνεται η διαδικασία κατασκευής του φίλτρου χρησιμοποιώντας το `fdatool`, η συνάρτηση του φίλτρου και το φιλτράρισμα των αρχείων φωνής.



Current Folder

Workspace

Name	Value
filename	'7B.wav'
filename1A	'1A.wav'
filename1B	'1B.wav'
filename2A	'2A.wav'
filename2B	'2B.wav'
filename3A	'3A.wav'
filename3B	'3B.wav'
filename4A	'4A.wav'
filename4B	'4B.wav'

```

1  ergasia.m
2  (y,zs) = subaudioresample(filename);
3  [p,q] = rat(8000/Fs,0.0001);
4  y2 = resample(y,p,q);
5
6
7  %filter apply
8  Fnew = 8000; % Sampling Frequency
9
10 N = 10; % Order
11 Fstop = 100; % Stopband Frequency
12 Fpass = 200; % Passband Frequency
13 Wstop = 1; % Stopband Weight
14 Wpass = 1; % Passband Weight
15 dens = 20; % Density Factor
16
17 % Calculate the coefficients using the FIRPM function.
18 b = firpm(N, [0 Fstop Fpass Fs/2]/(Fs/2), [0 0 1 1], [Wstop Wpass], ...
19         (dens));
20 Hd = dfilt.dfir(b);
21
22 yfiltered = filter(Hd,y2);

```

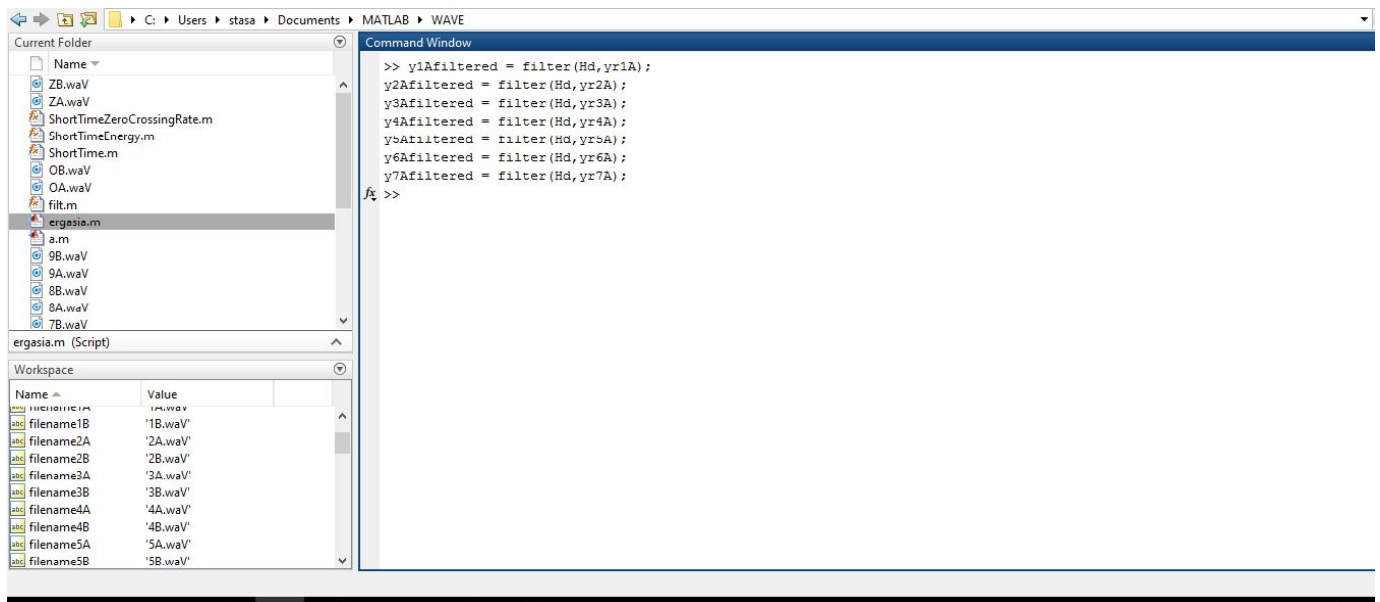
Command Window

```

>> sound(yz1A)
>> sound(y1A)
>> fdatool
>> fdatool
fx >>

```

script Ln 22 Col 2



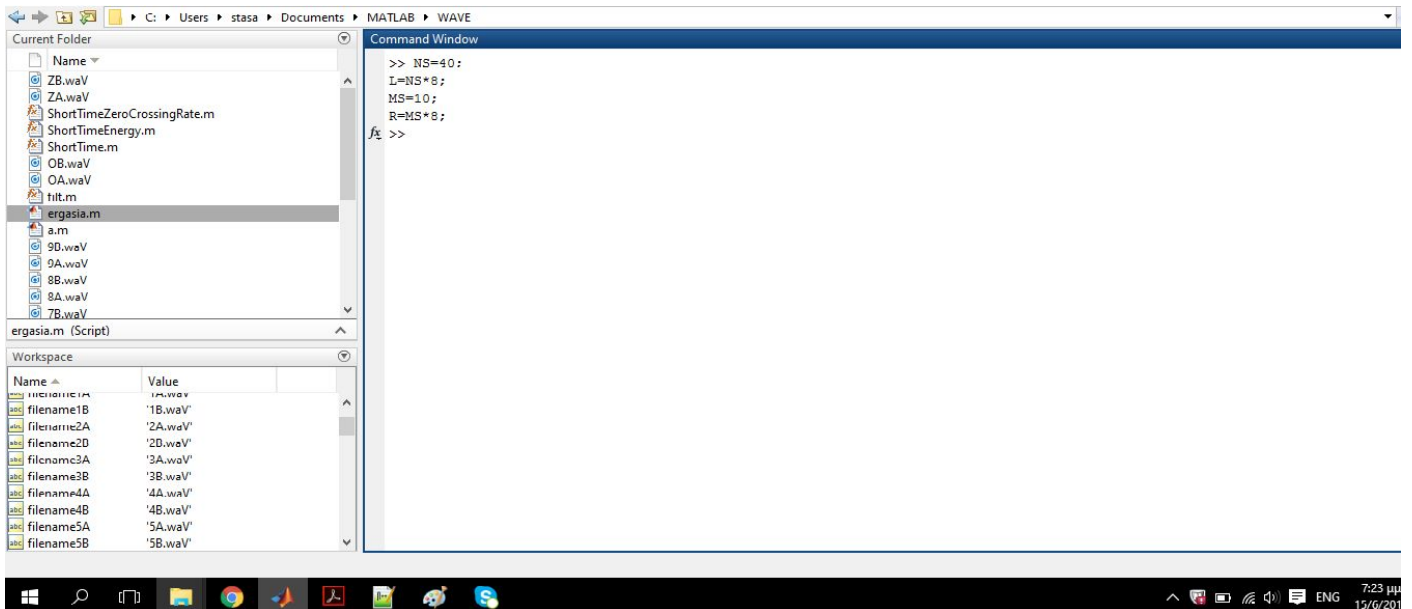
Βήμα 4ο:Καθορίστε τις παραμέτρους επεξεργασίας πλαισίων,δηλαδή:

- **NS= διάρκεια πλαισίου σε msec, $L=NS*8$ είναι η διάρκεια πλαισίου σε δείγματα,για ρυθμό δειγματοληψίας 8000Hz.**
- **MS= ολίσθηση πλαισίου σε msec, $R=MS*8$ είναι η ολίσθηση πλαισίου σε δείγματα,για ρυθμό δειγματοληψίας 8000Hz.**

Στην επεξεργασία βραχέος χρόνου απομονώνονται και υφίστανται επεξεργασία μικρής διάρκειας τμήματα του σήματος φωνής,σαν να ήταν σύντομα τμήματα ενός παρατεινόμενου σε διάρκεια ήχου, με σταθερές χρονικά αμετάβλητες ιδιότητες. Τα τμήματα του σήματος που προκύπτουν,αναφέρονται γενικά ως πλαίσια ανάλυσης(analysis frames).Ένα σημαντικό ζήτημα σε τέτοια συστήματα επεξεργασίας βραχέος χρόνου,είναι η επιλογή της διάρκειας του τμήματος,δηλαδή του μήκους πλαισίου.Όσο συντομότερο είναι ένα τμήμα τόσο λιγότερο πιθανό είναι τα χαρακτηριστικά της φωνής να μεταβάλλονται σημαντικά εντός του τμήματος.Επομένως η ικανότητα αποτελεσματικής παρακολούθησης των απότομων μεταβολών της κυματομορφής γίνεται βέλτιστη για τμήματα μικρής διάρκειας.Ωστόσο,οι εκτιμήσεις παραμέτρων απο μικρά τμήματα παρουσιάζουν σημαντική αβεβαιότητα εξαιτίας της μικρής ποσότητας των δεδομένων που είναι διαθέσιμα προς επεξεργασία.Κατά συνέπεια,κατόπιν συμβιβασμού στα συστήματα επεξεργασίας φωνής χρησιμοποιούνται πλαίσια διάρκειας από 10msec έως 40msec.Παράλληλα,η εκλογή του R δεν είναι ανεξάρτητη του μήκους του παραθύρου.Αν το παράθυρο έχει μήκος L δειγμάτων,τότε θα πρέπει να επιλέξουμε $R < L$ έτσι ώστε κάθε δείγμα φωνής να εμπεριέχεται σε τουλάχιστον ένα τμήμα ανάλυσης.

Για τις ανάγκες του παραπάνω ερωτήματος επιλέξαμε ως διάρκεια πλαισίου NS,το τυπικό μέγεθος πλαισίου $NS=40\text{msec}$ δηλαδή $L=NS*8=320$ δείγματα και ως ολίσθηση πλαισίου επιλέχθηκε σύμφωνα με την εκφώνηση $MS=10\text{msec}$,δηλαδή $R=MS*8=80$ δείγματα.

Παρακάτω φαίνεται ο ορισμός των παραμέτρων όπως περιγράφηκε.



Βήμα 5ο: Υπολογίστε τον λογάριθμο της ενέργειας και τους ρυθμούς διέλευσης (για διαστήματα των 10msec) για όλα τα πλαίσια σε ολόκληρο το αρχείο, δηλαδή κάθε R δείγματα.

Η κυριότερη χρησιμότητα της ενέργειας είναι ότι μας παρέχει την βάση για την διάκριση μεταξύ έμφωνων και μη τμημάτων. Οι τιμές της για τα μη έμφωνα τμήματα είναι σημαντικά μικρότερες από αυτές των έμφωνων περιπτώσεων. Η συνάρτηση της ενέργειας αξιοποιείται επίσης για να διαχωριστεί η φωνή από τα διαστήματα σιγής. Παρ'όλα αυτά υπάρχουν περιπτώσεις όπου η λογαριθμική ενέργεια βραχέος χρόνου δεν μπορεί να παρέχει από μόνη της μια σαφή ένδειξη της αρχής της ομιλίας όπως για παράδειγμα στην περίπτωση της εκφώνησης της λέξης *four* σε περιβάλλον χαμηλού θορύβου όπου η λέξη ξεκινάει με το ασθενές χαμηλής ενέργειας, υψηλής συχνότητας τυρβώδες σύμφωνο F.

Παράλληλα, ο ρυθμός των διελεύσεων από το μηδέν εφαρμόζεται ως εξής στα σήματα φωνής: Το μοντέλο παραγωγής προτείνει ότι η ενέργεια των έμφωνων ήχων συγκεντρώνεται κάτω από τα 3KHz εξαιτίας της εξασθένησης του φάσματος που προκαλείται από το γλωττιδικό κύμα, ενώ για τα φωνήματα που δεν είναι έμφωνα η περισσότερη ενέργεια βρίσκεται σε υψηλότερες συχνότητες. Αφού ένα φασματικό περιεχόμενο υψηλών συχνοτήτων συνεπάγεται υψηλούς ρυθμούς διέλευσης από το μηδέν και ένα φασματικό περιεχόμενο χαμηλών συχνοτήτων συνεπάγεται χαμηλούς ρυθμούς διέλευσης, υπάρχει μεγάλη συσχέτιση μεταξύ του ρυθμού των διελεύσεων από το μηδέν και της κατανομής της ενέργειας του σήματος ανά συχνότητα. Συνεπώς τα έμφωνα φωνήματα χαρακτηρίζονται από υψηλή ενέργεια και χαμηλό ρυθμό διέλευσης από το μηδέν ενώ στην αντίθετη περίπτωση εκδηλώνεται υψηλός αριθμός διέλευσης και χαμηλή ενέργεια.

Σε κάθε περίπτωση είναι προτιμητέο να χρησιμοποιούνται από κοινού οι μετρήσεις του λογαρίθμου της ενέργειας βραχέος χρόνου και του ρυθμού διέλευσης από το μηδέν.

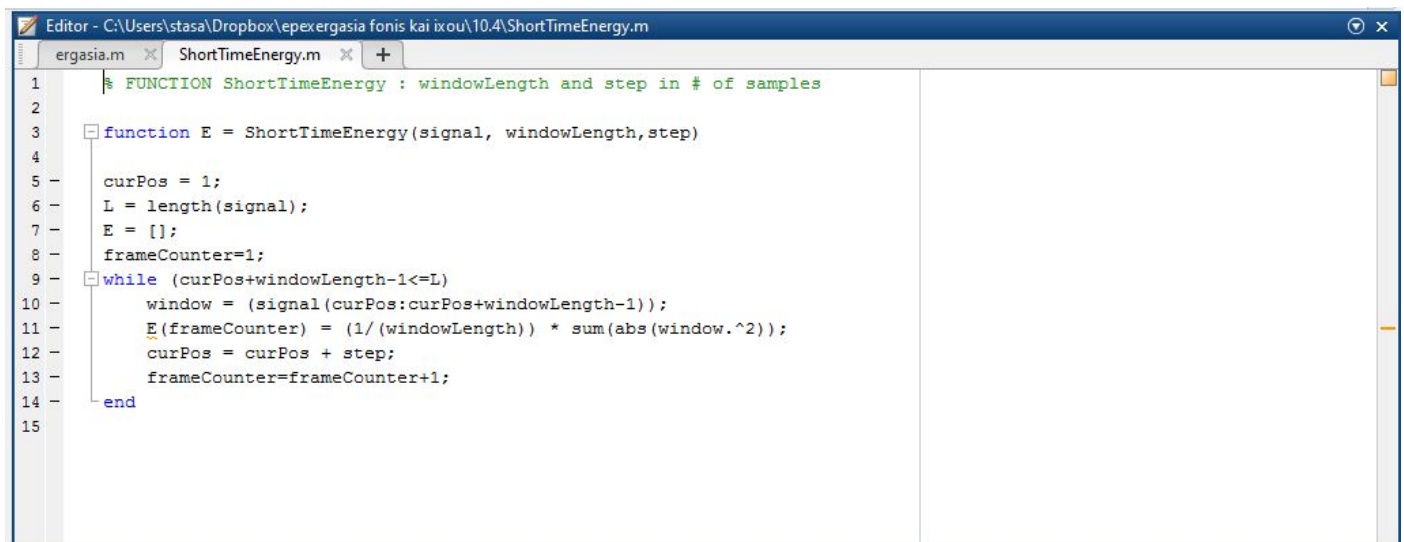
Για τον υπολογισμό του λογάριθμου της ενέργειας βραχέος χρόνου και του ρυθμού διέλευσης από το μηδέν χρησιμοποιήθηκαν οι έτοιμες συναρτήσεις ShortTimeEnergy.m και ShortTimeZeroCrossingRate.m από τα χρήσιμα m-files που παρέχονται στον ιστότοπο του μαθήματος.

Έπειτα από την κλήση της συνάρτησης ShortTimeEnergy εκτελούμε

$E = 10 \cdot \log_{10}(E) - \max(10 \cdot \log_{10}(E));$

όπου κανονικοποιεί τον λογάριθμο της ενέργειας βραχέος χρόνου σε μια μέγιστη τιμή των 0 dB.

Παρακάτω παρουσιάζονται οι συναρτήσεις και ο υπολογισμός των παραμέτρων βραχέος χρόνου για τα αρχεία φωνής.



```
Editor - C:\Users\stasa\Dropbox\epexergasia fonis kai ikou\10.4\ShortTimeEnergy.m
ergasia.m ShortTimeEnergy.m +
1 % FUNCTION ShortTimeEnergy : windowLength and step in # of samples
2
3 function E = ShortTimeEnergy(signal, windowLength, step)
4
5     curPos = 1;
6     L = length(signal);
7     E = [];
8     frameCounter=1;
9     while (curPos+windowLength-1<=L)
10         window = (signal(curPos:curPos+windowLength-1));
11         E(frameCounter) = (1/(windowLength)) * sum(abs(window.^2));
12         curPos = curPos + step;
13         frameCounter=frameCounter+1;
14     end
15
```



```
Editor - C:\Users\stasa\Documents\MATLAB\WAVE\ShortTimeZeroCrossingRate.m*
ShortTimeEnergy.m ShortTimeZeroCrossingRate.m* +
3 function ZCR = ShortTimeZeroCrossingRate(signal, windowLength, step)
4
5     curPos = 1;
6     L = length(signal);
7     ZCR = [];
8     frameCounter=1;
9     while (curPos+windowLength-1<=L)
10         window = (signal(curPos:curPos+windowLength-1));
11         temp=0;
12         for i=2:windowLength
13             temp = temp + abs( sign(window(i))- sign(window(i-1)) );
14         end
15         ZCR(frameCounter) = temp;
16         curPos = curPos + step;
17         frameCounter=frameCounter+1;
18     end
19
20     ZCR = ( ZCR * step ) / ( 2 * windowLength );
21
```



```
Editor - C:\Users\stasa\Dropbox\epexergasia fonis kai ixou\10.4\ergasia.m
ergasia.m
18
19 [y,Fs] = audioread( [ '..\matlab\chapter_10\tidigits_isolated_unendpointed\' filenames{z} ] );
20
21 [p,q] = rat(8000/Fs,0.0001);
22 y2 = resample(y,p,q);
23
24 %erwthma 4%
25 NS=40;
26 L=NS*8;
27 MS=10;
28 R=MS*8;
29
30 filter_apply
31
32 %erwthma 5%
33 e6A=ShortTimeEnergy(yfiltered,L,R);
34 e6A = 10*log10(e6A) -max( 10*log10(e6A) );
35 zcr6A=ShortTimeZeroCrossingRate(yfiltered,L,R);
36
37 %erwthma 6%
```

Βήμα 6ο: Υπολογίστε τη μέση τιμή και την τυπική απόκλιση της λογαριθμικής ενέργειας και του ρυθμού διέλευσης από το μηδέν, για τα πρώτα 10 πλαίσια του αρχείου (υποθέτοντας ότι αντιστοιχούν μόνο στο σήμα υποβάθρου). Ονομάστε τις παραμέτρους αυτές *eavg*, *esig*, για τη μέση τιμή και την τυπική απόκλιση της λογαριθμικής ενέργειας, και *zcavg*, *zcsig*, για τη μέση τιμή και την τυπική απόκλιση του ρυθμού διέλευσης από το μηδέν.

Υποτίθεται ότι τα πρώτα 100ms(10 πλαίσια) του ηχογραφημένου σήματος δεν περιέχουν καθόλου ομιλία. Αυτό δικαιολογείται στις περισσότερες εφαρμογές λόγω του μεγάλου χρόνου αντίδρασης των ομιλητών από την στιγμή που θα τους ζητηθεί να ξεκινήσουν να ομιλούν για προκαθορισμένο διάστημα και αφού έχει δοθεί το σήμα για την έναρξη της ηχογράφησης. Η μέση τιμή και η τυπική απόκλιση του λογαρίθμου ενέργειας και του ρυθμού διέλευσης από το μηδέν, υπολογίζονται σε αυτό το αρχικό διάστημα των 100msec προκειμένου να μας δώσουν μια χονδρική στατιστική εκτίμηση του σήματος υποβάθρου.

Για να υπολογίσουμε την μέση τιμή και την τυπική απόκλιση της λογαριθμικής ενέργειας και του ρυθμού διέλευσης από το μηδέν για τα πρώτα 10 πλαίσια του αρχείου εργαστήκαμε ως εξής:

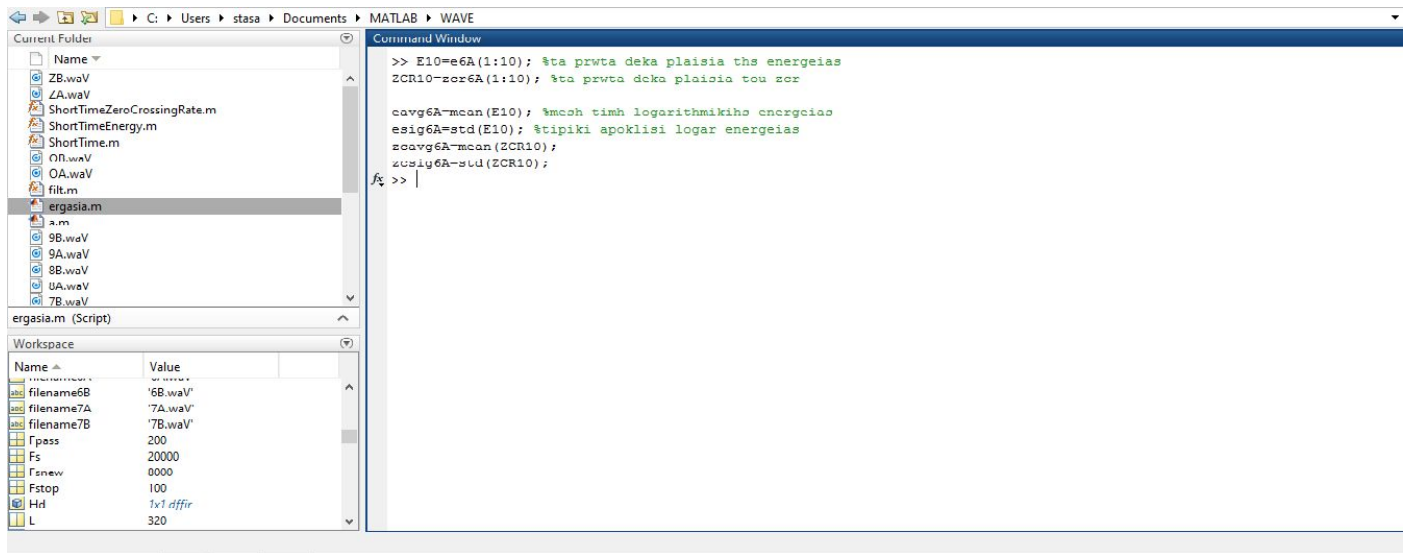
Αρχικά αποθηκεύσαμε τα πρώτα 10 πλαίσια της ενέργειας και του ρυθμού διέλευσης από το μηδέν στις μεταβλητές *E10* και *ZCR10* αντίστοιχα, χρησιμοποιώντας τις εξής εντολές:

```
E10=e6A(1:10);
ZCR10=zcr6A(1:10);
```

Έπειτα υπολογίσαμε την μέση τιμή και τυπική απόκλιση χρησιμοποιώντας τις παρακάτω εντολές:

```
eavg6A=mean(E10);
esig6A=std(E10);
zcavg6A=mean(ZCR10);
zcsig6A=std(ZCR10);
```

Παρακάτω φαίνεται η διαδικασία που ακολουθήσαμε.



The screenshot shows the MATLAB Wave software interface. The 'Current Folder' pane on the left displays a list of files, including 'ergasia.m'. The 'Command Window' on the right shows the execution of the script, with the following commands and results:

```
>> E10=e6A(1:10); %ta prwta deka plaisia ths energeias
ZCR10=zer6A(1:10); %ta prwta deka plaisia tou zer

eavg6A=mean(E10); %mesh timh logarithmikihs energeias
esig6A=std(E10); %tipiki apoklisi logar energeias
zcavg6A=mean(ZCR10);
zcsig6A=std(ZCR10);

%>>
```

The 'Workspace' pane at the bottom shows the variables defined in the script:

Name	Value
filename6B	'6B.waV'
filename7A	'7A.waV'
filename7B	'7B.waV'
Fpass	200
Fs	20000
Fsnew	8000
Fstop	100
Hid	1x1 diff
L	320

Βήμα 7ο:Ορίστε τις παραμέτρους του συστήματος αναζήτησης άκρων:

- **IF = 35** - Σταθερό κατώφλι για τον ρυθμό διέλευσης από το μηδέν.
- **IZCT = max (IF, zcavg+3*zc sig)** – Μεταβλητό κατώφλι για τον ρυθμό διέλευσης από το μηδέν, με βάση τη στατιστική ανάλυση του σήματος υποβάθρου.
- **IMX = max (eng)** – Απόλυτο πλάτος της μέγιστης κορυφής της παραμέτρου της λογαριθμικής ενέργειας.
- **ITU = IMX - 20** – Υψηλό κατώφλι για την παράμετρο της λογαριθμικής ενέργειας.
- **ITL = max (eavg+3*esig, ITU-10)** – Χαμηλό κατώφλι για την παράμετρο της λογαριθμικής ενέργειας.

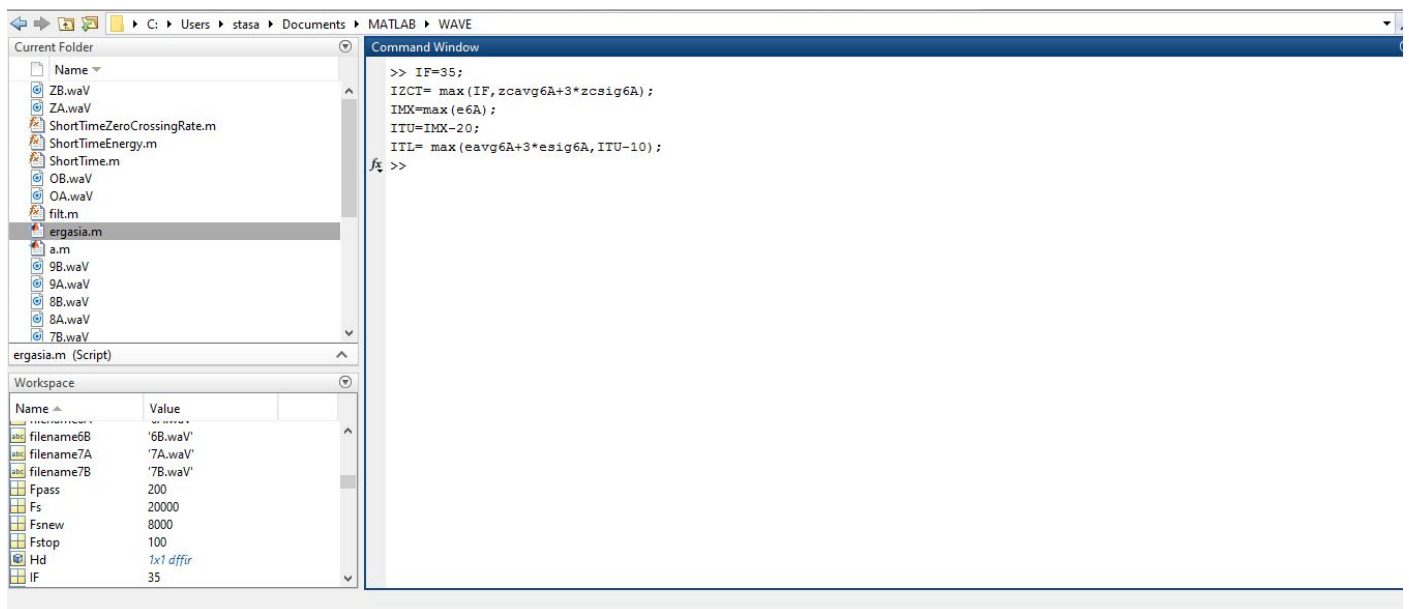
Αξιοποιώντας τις προηγούμενες μετρήσεις υπολογίζεται ένα κατώφλι ρυθμού διέλευσης από το μηδέν, IZCT ως $IZCT = \max (IF, zcavg+3*zc sig)$.

Η ποσότητα IF, η οποία λαμβάνει τιμή 35, αποτελεί ένα γενικό κατώφλι για την ανίχνευση των μη έμφωνων πλαισίων. Η τιμή του κατωφλίου IZCT, αυξάνεται εάν το σήμα υποβάθρου κατά την διάρκεια των πρώτων 100msec της ηχογράφησης εμφανίζει υψηλό ρυθμό διέλευσης από το μηδέν, όπως αυτός εκτιμάται με βάση τις τιμές των zcavg και zcsig.

Ομοίως, για την μέτρηση του λογαρίθμου της ενέργειας, ορίζουμε ένα ζευγάρι κατωφλίων και συγκεκριμένα ως ITU, ένα άνω συντηρητικό κατώφλι, και ως ITR ένα λιγότερο συντηρητικό κατώφλι για την παρουσία της ομιλίας.

Από τα παραπάνω προκύπτει ότι τα κατώφλια του λογαρίθμου της ενέργειας, βασίζονται τόσο σε στατιστικά μακρού χρόνου της λογαριθμικής ενέργειας για ήχους φωνής και μη, όσο και σε συγκεκριμένες τιμές του λογαρίθμου της ενέργειας που προκύπτουν από το πραγματικό σήμα υποβάθρου, στα πρώτα 100 msec της ηχογράφησης.

Παρακάτω βλέπουμε την διαδικασία για τον ορισμό των παραπάνω κατωφλίων.



Βήμα 8ο: Αναζητήστε την περιοχή του σήματος όπου εμφανίζεται η μέγιστη λογαριθμική ενέργεια (αυτή υποτίθεται ότι βρίσκεται γύρω στο κέντρο του διαστήματος φωνής), και κατόπιν πραγματοποιήστε αναζήτηση στο περίγραμμα της λογαριθμικής ενέργειας για να βρείτε την περιοχή της κύριας συγκέντρωσης της ενέργειας, όπου η λογαριθμική ενέργεια πέφτει κάτω από το επίπεδο ITU και στις δύο πλευρές της κορυφής. Κατόπιν, ψάξτε να βρείτε τα σημεία όπου η λογαριθμική ενέργεια πέφτει κάτω από τη στάθμη ITL και αυτό οριοθετεί το επεκταμένο περίγραμμα της κύριας ενέργειας για τη λέξη που έχει εκφωνηθεί. Στη συνέχεια, η περιοχή της λέξης επεκτείνεται με βάση τις τιμές του ρυθμού διέλευσης από το μηδέν, που υπερβαίνουν το κατώφλι IZCT για τουλάχιστον τέσσερα διαδοχικά πλαίσια, μέσα στις περιοχές γειτνίασης των τρεχουσών εκτιμήσεων των άκρων για τη λέξη. Τα άκρα της λέξης διορθώνονται για να συμπεριλάβουν ολόκληρη την περιοχή όπου ο ρυθμός διέλευσης από το μηδέν υπερβαίνει το κατώφλι, από τη στιγμή που έχει διασχιστεί το κατώφλι μικρότερης διάρκειας.

1. Σχεδιάστε τα περιγράμματα του λογαρίθμου της ενέργειας και του ρυθμού διέλευσης από το μηδέν σε δύο ξεχωριστά διαγράμματα πάνω στην ίδια σελίδα, και δείξτε τις αποφάσεις του αλγορίθμου αναζήτησης άκρων, που βασίζονται αποκλειστικά στην αναζήτηση φωνής μέσω της λογαριθμικής ενέργειας (χρησιμοποιήστε διακεκομμένες γραμμές στα διαγράμματα) και αυτές που βασίζονται στη λογαριθμική ενέργεια και στον ρυθμό διέλευσης από το μηδέν μαζί (χρησιμοποιήστε γραμμές από κουκκίδες στα διαγράμματα). (Ίσως θελήσετε επίσης να χρησιμοποιήσετε διακεκομμένες γραμμές στα διαγράμματα, για να δείξετε τα σημεία όπου τα διάφορα κατώφλια τελικά συμπίπτουν). Το σχήμα Π10.4 απεικονίζει ένα τυπικό διάγραμμα της εκφώνησης της λέξης /six/, όπου έχουμε χρησιμοποιήσει διακεκομμένες γραμμές για να δείξουμε τα άκρα που προέκυψαν από τη λογαριθμική ενέργεια, και γραμμές με κουκκίδες για να δείξουμε τα άκρα που προήλθαν από τον ρυθμό διέλευσης από το μηδέν, μαζί με τις διακεκομμένες γραμμές των κατωφλίων της λογαριθμικής ενέργειας και του ρυθμού διέλευσης από το μηδέν.

2. Ακούστε την εντοπισμένη ως φωνή περιοχή του σήματος, για να διαπιστώσετε εάν λείπει κάποιο τμήμα φωνής, ή εάν κάποιο τμήμα του σήματος υποβάθρου άσχετο προς τη φωνή, έχει συμπεριληφθεί στην περιοχή αυτή.

Σημείωση:

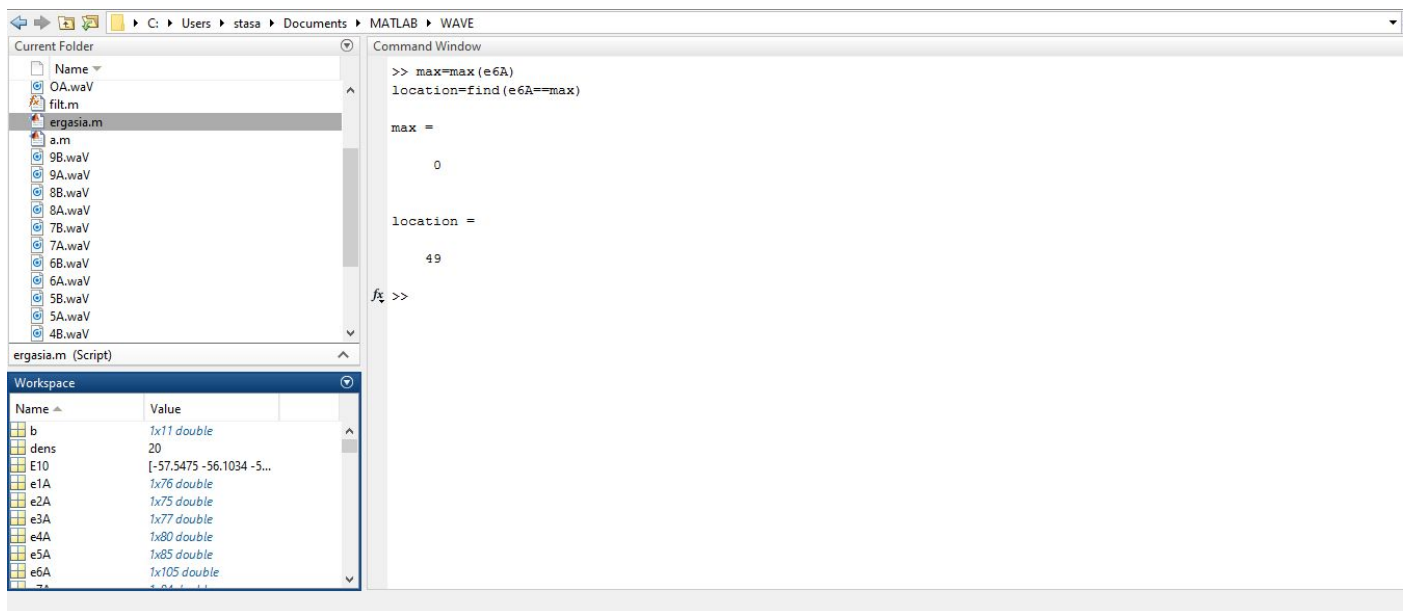
Για το συγκεκριμένο παράδειγμα θα αναφερθούμε στο αρχείο φωνής 6A.waV. Όμοια εκτελείται και για τα υπόλοιπα αρχεία φωνής όπως φαίνεται στα διαγράμματα που προκύπτουν.

Αρχικά αναζητούμε την περιοχή του σήματος όπου εμφανίζεται η μέγιστη λογαριθμική ενέργεια και την τιμή της χρησιμοποιώντας τις εντολές:

```
maxE = max(e6A) ;
```

```
location = find(e6A==maxE);
```

Η μέγιστη λογαριθμική είναι 0 (έχει γίνει κανονικοποίηση των τιμών της ενέργειας στην μέγιστη τιμή του 0) και εμφανίζεται περίπου στην μέση του διαστήματος location=49 αφού η ενέργεια είχε συνολικά 105 τιμές. Αυτό επιβεβαιώνει τον ισχυρισμό ότι η μέγιστη λογαριθμική ενέργεια βρίσκεται περίπου στο κέντρο του σήματος φωνής.



Έπειτα,πραγματοποιείται αναζήτηση στο περίγραμμα της λογαριθμικής ενέργειας για να βρεθεί η περιοχή της κύριας συγκέντρωσης της ενέργειας,όπου η λογαριθμική ενέργεια πέφτει κάτω από το επίπεδο ITU και στις δυο πλευρές τις κορυφής.Δηλαδή πραγματοποιούμε την ακόλουθη σύγκριση: $e6A < ITU$

Η περιοχή όπου βρίσκεται η μέγιστη λογαριθμική ενέργεια θα δώσει 0 στην παραπάνω σύγκριση.Άρα το πρώτο μη μηδενικό στοιχείο απο τα αριστερά της θέσης που βρίσκεται η μέγιστη ενέργεια και το πρώτο μη μηδενικό στοιχείο απο τα δεξιά της θέσης που βρίσκεται η μέγιστη ενέργεια αποτελούν τα άκρα της περιοχής της κύριας συγκέντρωσης της ενέργειας.Τα άκρα της περιοχής της κύριας συγκέντρωσης προσαρμόζονται ως akro1=43 και akro2=55. Παρακάτω φαίνεται η υλοποίηση της παραπάνω ιδέας:

Current Folder: C:\Users\stasa\Documents\MATLAB\WAVE

Command Window

```
>> theseis=[];
for i=1:length(e6A) % ποια στοιχεία είναι μικρότερα της ITU
if e6A(i)<ITU
theseis(i)=i
end
end

theseis =

     1

theseis =

     1     2

theseis =

     1     2     3

theseis =

     1     2     3     4

f1 theseis =
```

Workspace

Name	Value
akro1	43
akro2	55
b	1x11 double
dens	20
E10	[-57.5475 -56.1034 -5...
e1A	1x76 double
e2A	1x75 double
e3A	1x77 double
e4A	1x80 double

Current Folder: C:\Users\stasa\Documents\MATLAB\WAVE

Command Window

```
theseis =

Columns 1 through 20

     1     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17    18    19    20

Columns 21 through 40

    21    22    23    24    25    26    27    28    29    30    31    32    33    34    35    36    37    38    39    40

Columns 41 through 60

    41    42    43     0     0     0     0     0     0     0     0     0     0     0     55    56    57    58    59    60

Columns 61 through 80

    61    62    63    64    65    66    67    68    69    70    71    72    73    74    75    76    77    78    79    80

Columns 81 through 100

    81    82    83    84    85    86    87    88    89    90    91    92    93    94    95    96    97    98    99   100

Columns 101 through 105

   101   102   103   104   105

f1 >>
```

Workspace

Name	Value
akro1	43
akro2	55
b	1x11 double
dens	20
E10	[-57.5475 -56.1034 -5...
e1A	1x76 double
e2A	1x75 double
e3A	1x77 double
e4A	1x80 double

Current Folder: C:\Users\stasa\Documents\MATLAB\WAVE

Workspace:

Name	Value
akro1	43
akro2	55
b	1x11 double
dens	20
E10	[-57.5475 -56.1034 -5...
e1A	1x76 double
e2A	1x75 double
e3A	1x77 double
e4A	1x80 double

Command Window:

```

Columns 81 through 100
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

Columns 101 through 105
101 102 103 104 105

>> k2=find(~theseis)
akro1=k2(1)-1 %akroA perioxis kurias sugkentrwshs ths energeias
akro2=k2(end)+1 %akroB perioxis kurias sugkentrwshs ths energeias

k2 =
44 45 46 47 48 49 50 51 52 53 54

akro1 =
43

akro2 =
55

```

Στην συνέχεια,βρίσκουμε τα σημεία όπου η λογαριθμική ενέργεια πέφτει κάτω από την στάθμη ITL και αυτό οριοθετεί το επεκτεταμένο περίγραμμα της κύριας ενέργειας για την λέξη που έχει εκφωνηθεί. Παρατηρούμε ότι η περιοχή με την μέγιστη συγκέντρωση ενέργειας δεν πέφτει κάτω από το ITL.

Current Folder: C:\Users\stasa\Dropbox\epexergasia fonis kai ixou\10.4

Workspace:

Name	Value
akro1	46
akro2	82
b	1x11 double
dens	20
E10	[-53.0059 -53.7162 -5...
e6A	1x107 double
eavg6A	-53.7490
epektakro1	36
epektakro2	82

Command Window:

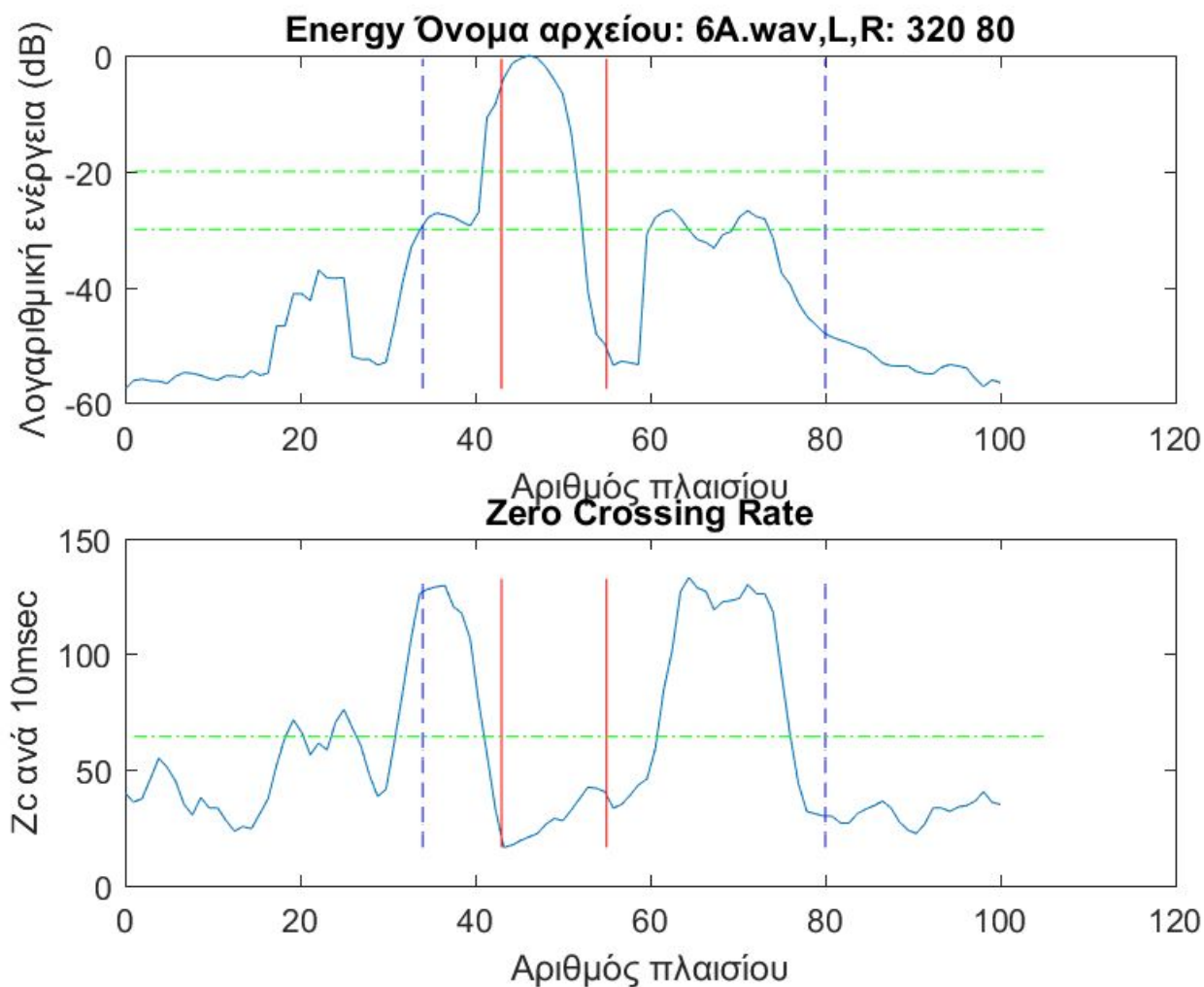
```

akro2 =
82

```

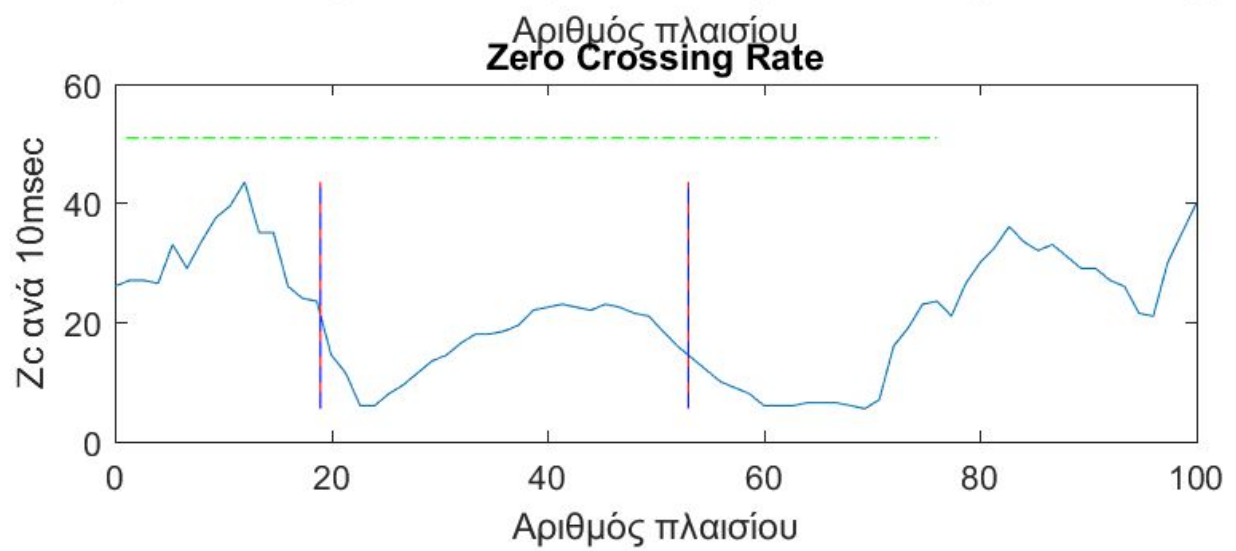
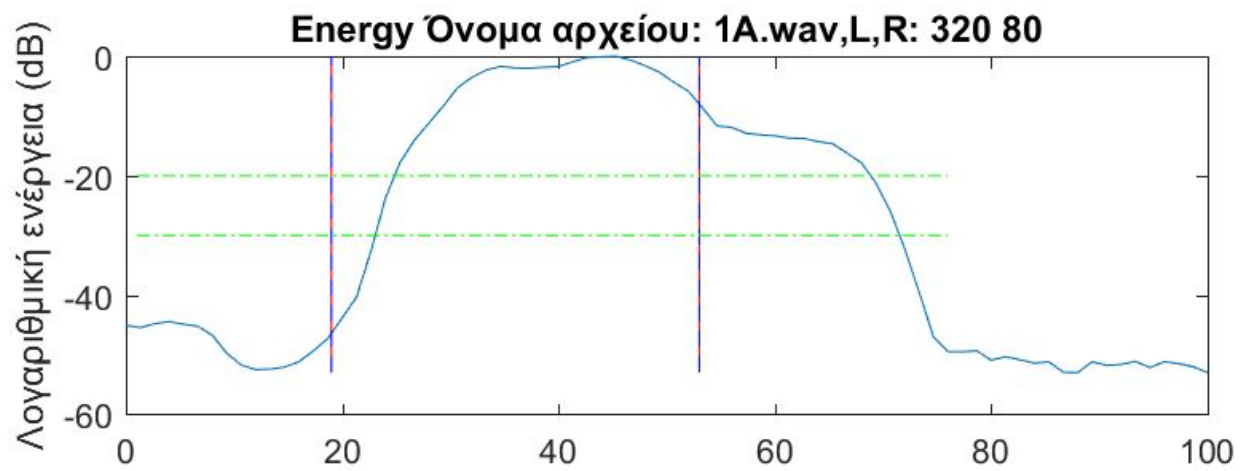
Τέλος,η περιοχή της λέξης επεκτείνεται με βάση τις τιμές του ρυθμού διέλευσης από το μηδέν,που υπερβαίνουν το κατώφλι IZCT για τουλάχιστον 4 διαδοχικά πλαίσια,μέσα στις περιοχές γειτνίασης των τρέχουσων εκτιμήσεων των άκρων για την λέξη.Πραγματοποιείται αναζήτηση σε 25 πλαίσια αριστερά του akro1 και σε 25 πλαίσια δεξιά του akro2 για να βρεθούν τα νέα επεκτεταμένα άκρα. Τα επεκτεταμένα άκρα προσαρμόζονται ως εξής epektakro1=34 και epektakro2=79.

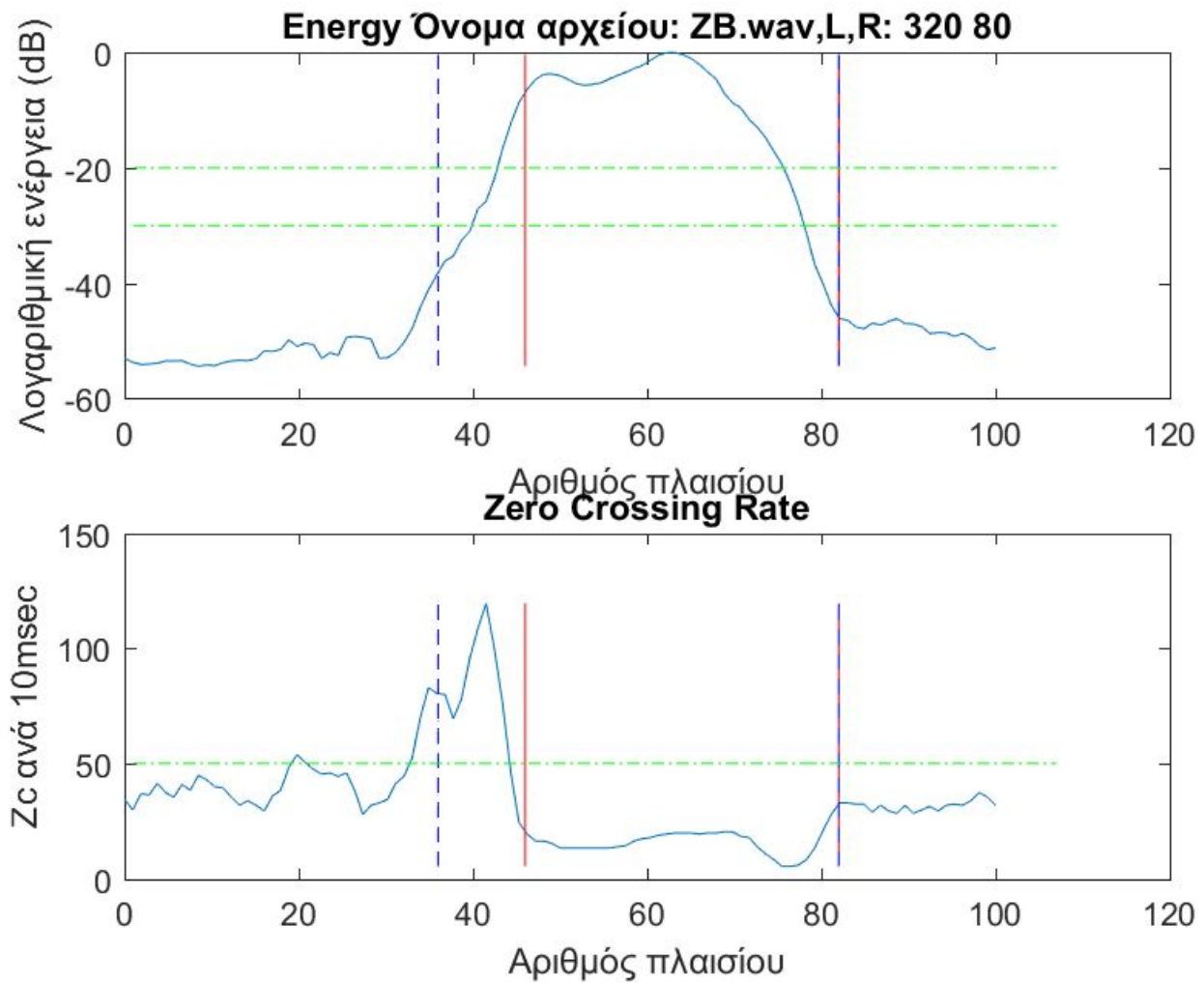
Παρακάτω παρουσιάζεται το διάγραμμα του λογαρίθμου της ενέργειας και του ρυθμού διέλευσης από το μηδέν για το αρχείο 6A.waV



Ενδεικτικά παρουσιάζονται τα διαγράμματα του λογαρίθμου της ενέργειας και του ρυθμού διέλευσης από το μηδέν για τα αρχεία 1A.waV , ZB.waV.

Τα υπόλοιπα διαγράμματα για την άσκηση 10.4 βρίσκονται στον φάκελο 10.4/figures.





Τέλος, ακούμε την εντοπισμένη ως φωνή του σήματος χρησιμοποιώντας τις εντολές

```
voice=[];
voice=yfiltered( epektakro1*R:epektakro2*R);
sound(voice,8000);
```

και παρατηρούμε ότι ο αλγόριθμος αναζήτησης άκρων έδωσε τα σωστά αποτελέσματα αφού δεν λείπει κάποιο τμήμα της φωνής ούτε κάποιο σήμα υποβάθρου άσχετο προς την φωνή έχει συμπεριληφθεί.

Άσκηση 10.5

Εκφώνηση

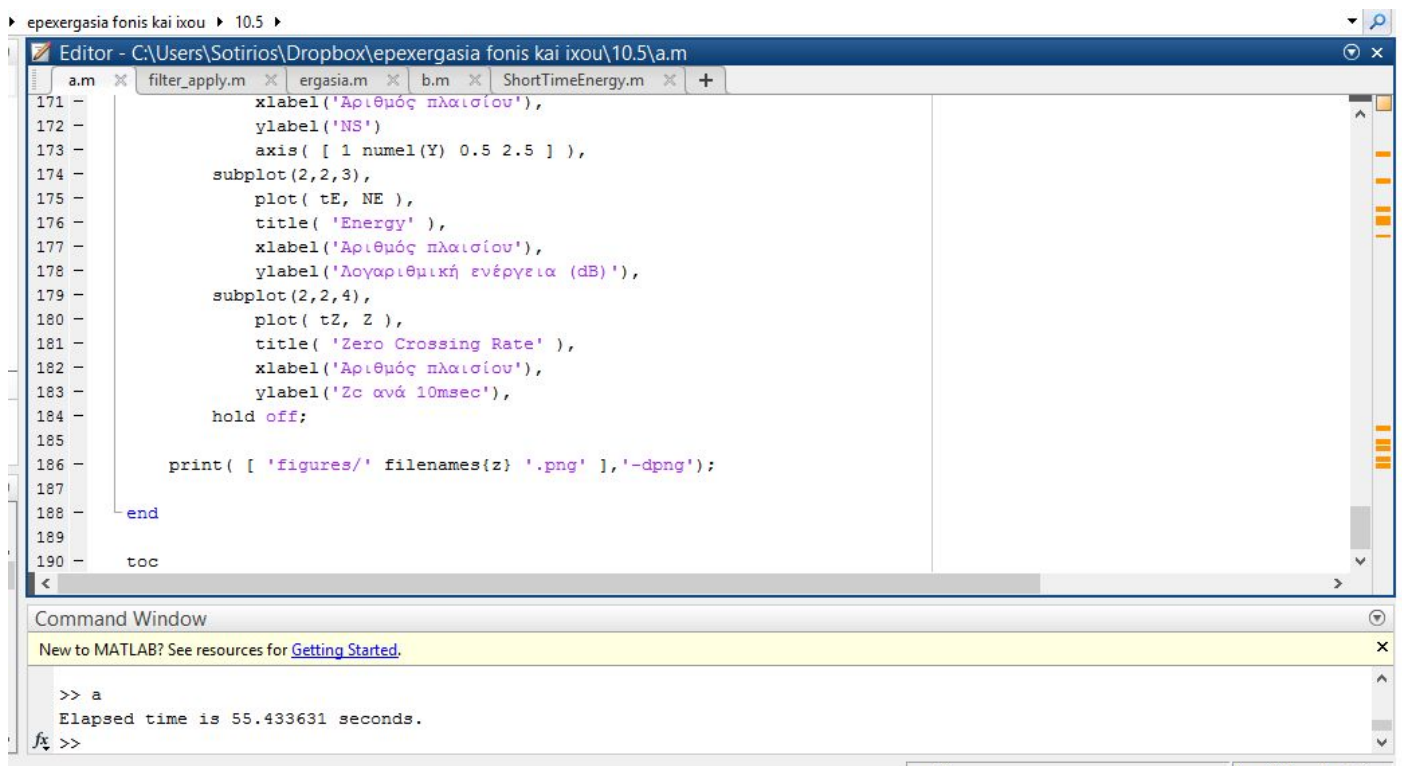
(Άσκηση MATLAB Bayesian Ταξινομητής Μεμονωμένων λέξεων (Bayesian Isolated Word Speech Classifier). Γράψτε ένα πρόγραμμα MATLAB που να ταξινομεί τα πλαίσια ενός σήματος, ως πλαίσια όχι φωνής (Κλάση 1) ή ως πλαίσια φωνής (Κλάση 2), χρησιμοποιώντας ένα Bayesian πλαίσιο στοχαστικής ανάλυσης. Το διάνυσμα χαρακτηριστικών για την κατηγοριοποίηση των πλαισίων αποτελείται από τις μετρήσεις βραχέος χρόνου του λογαρίθμου της ενέργειας και του ρυθμού διέλευσης από το μηδέν (σε πλαίσια των 10msec), και οι δύο συνιστώσες του διανύσματος χαρακτηριστικών μοντελοποιούνται χρησιμοποιώντας μια Gaussian προσέγγιση, θεωρώντας ότι τα χαρακτηριστικά του διανύσματος είναι ασυσχέτιστα.

Τα αρχεία φωνής που θα χρησιμοποιηθούν στην άσκηση αυτή, είναι τα ίδια με αυτά που χρησιμοποιήθηκαν στο Πρόβλημα 10.4 και μπορούν να μεταφορτωθούν από την ιστοσελίδα του βιβλίου.

Λύση

Εκτέλεση:

Τα αρχεία βρίσκονται στον φάκελο 10.5 και η λύση είναι στο αρχείο a.m. Παρακάτω φαίνεται ένα screenshot με την εκτέλεσή του καθώς και ο συνολικός χρόνος που χρειάστηκε για να κάνει την όλη επεξεργασία καθώς και τις εικόνες των διαγραμμάτων.



```
171 - xlabel('Αριθμός πλαισίου'),
172 - ylabel('NS')
173 - axis( [ 1 numel(Y) 0.5 2.5 ] ),
174 - subplot(2,2,3),
175 - plot( tE, NE ),
176 - title( 'Energy' ),
177 - xlabel('Αριθμός πλαισίου'),
178 - ylabel('Λογαριθμική ενέργεια (dB)'),
179 - subplot(2,2,4),
180 - plot( tZ, Z ),
181 - title( 'Zero Crossing Rate' ),
182 - xlabel('Αριθμός πλαισίου'),
183 - ylabel('Zc ανά 10msec'),
184 - hold off;
185
186 - print( [ 'figures/' filenames{z} '.png' ], '-dpng');
187
188 - end
189
190 - toc
```

Command Window

New to MATLAB? See resources for [Getting Started](#).

```
>> a
Elapsed time is 55.433631 seconds.
fx >>
```

Αρχικά φορτώνουμε τα μητρώα ομιλίας και μη ομιλίας. Στη συνέχεια υπολογίζουμε την μέση τιμή και την τυπική απόκλιση των διανυσμάτων κάθε κλάσης.

Για την κλάση μη-ομιλίας προέκυψε:

για την μέση τιμή οι τιμές -52.0866 και 22.9288

για την τυπική απόκλιση οι τιμές 5.1042 και 5.5124

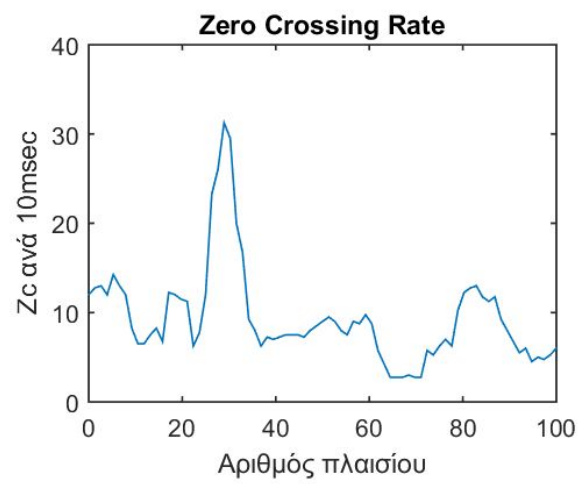
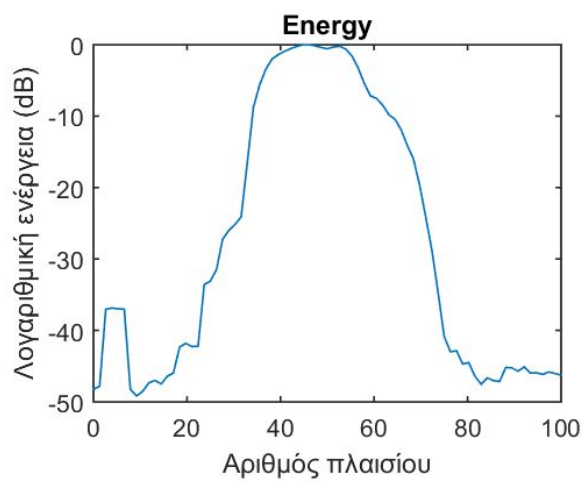
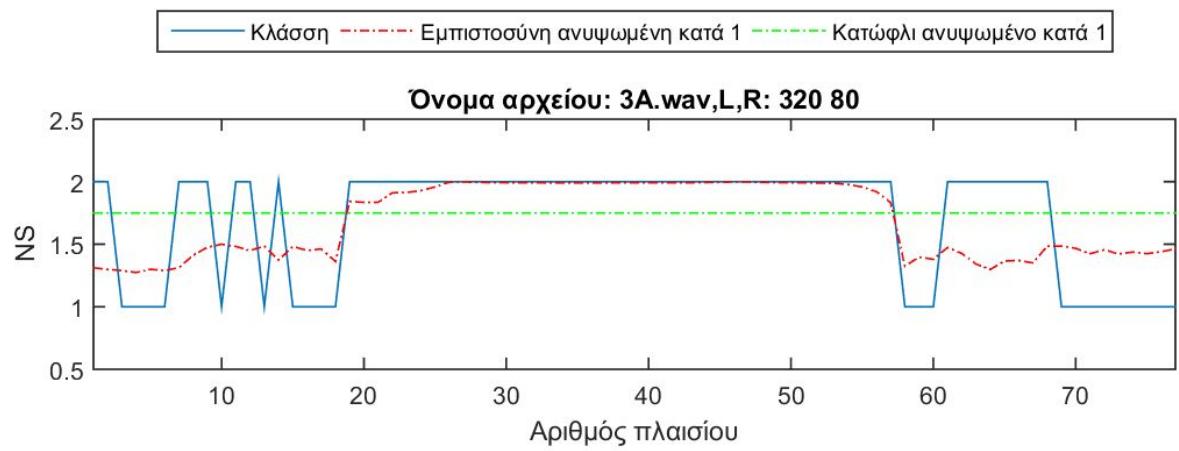
Για την κλάση ομιλίας προέκυψε:

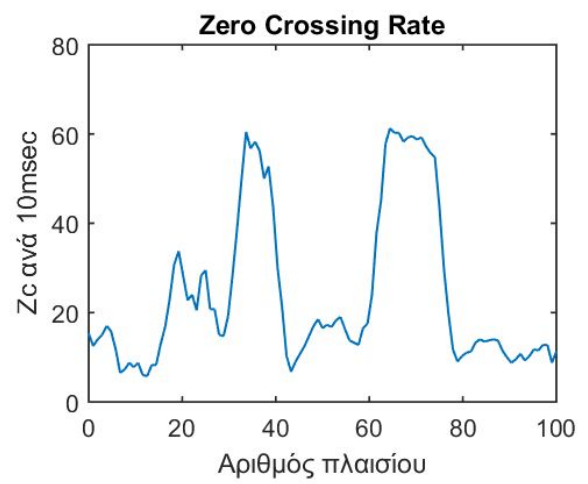
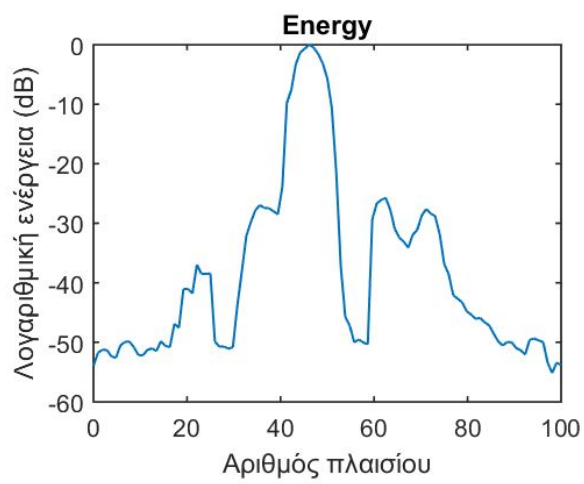
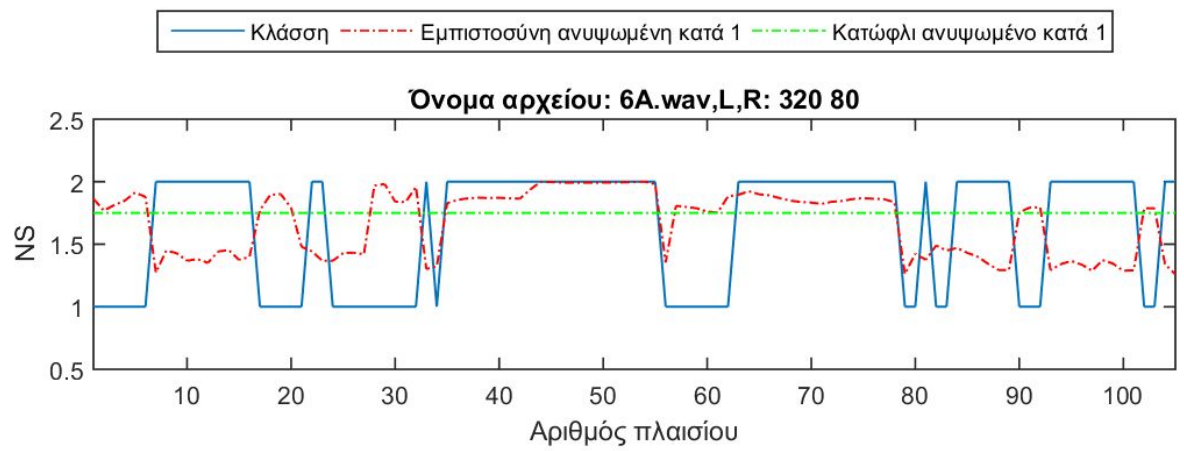
για την μέση τιμή οι τιμές -10.9659 και 15.4631

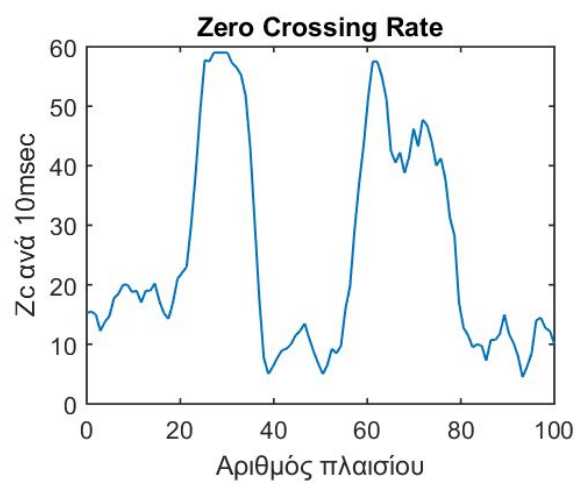
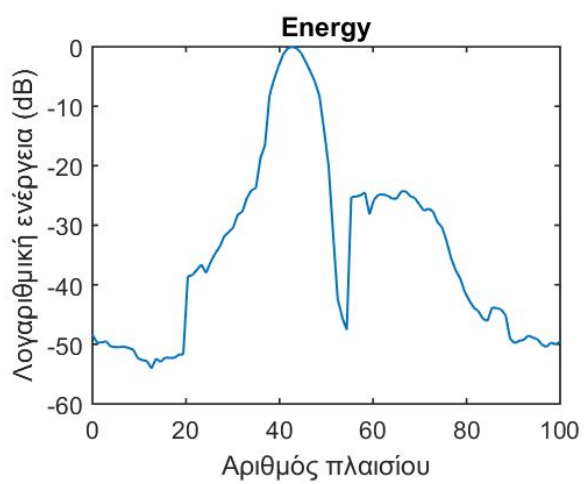
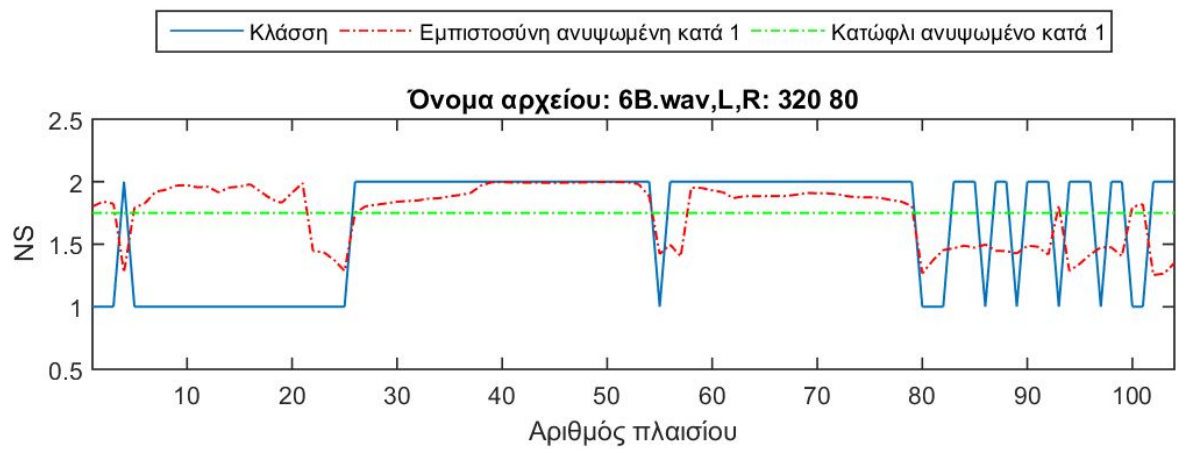
για την τυπική απόκλιση οι τιμές 11.8630 και 15.3330

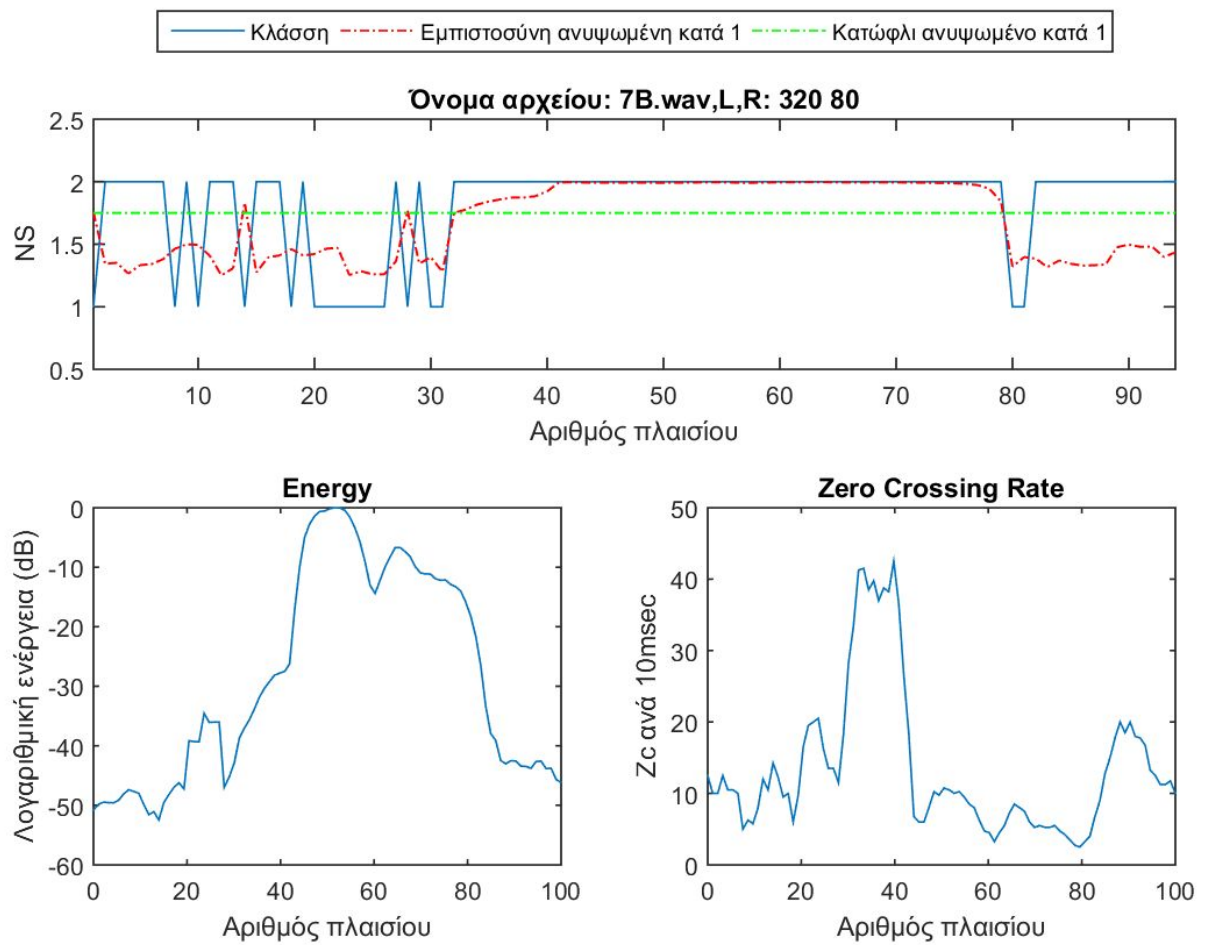
Έπειτα επεξεργαζόμαστε κάθε αρχείο ένα προς ένα. Σε κάθε αρχείο που φορτώνουμε εκτελούμε παρομοίως τα 5 πρώτα βήματα της άσκησης 10.4. Αμέσως μετά για κάθε πλαίσιο υπολογίζουμε τις τιμές των αποστάσεων και εμπιστοσύνης με κατώφλι την τιμή 0.75. Και τέλος απεικονίζουμε την ταξινόμηση των κλάσεων με τις τιμές της εμπιστοσύνης αυξημένες κατά ένα καθώς και το κατώφλι στην τιμή 1.75 ώστε να φαίνεται καλύτερα η αντιστοίχιση, επίσης απεικονίζουμε την ενέργεια και το zero crossing rate.

Παρακάτω φαίνονται 3 ενδεικτικά αποτελέσματα, τα υπόλοιπα βρίσκονται στον φάκελο 10.5/figures.









Τέλος αναφερόμενοι στο παράδειγμα 7B προκύπτουν οι παρακάτω παρατηρήσεις:

παρατηρούμε ότι υπάρχει μεγάλη εμπιστοσύνη σε συνεχόμενα πλαίσια ομιλίας (π.χ. στο κέντρο), αλλά στην αρχή και στο τέλος έχουμε χαμηλή αξιοπιστία σε ορισμένα αρχεία.