1 Ψηφιακή επεξεργασία Σήματος Εργασία 2.

Στέφανος Μπακλαβάς 1115201700093

Στο πλαίσιο αυτής της εργασίας υλοποιήθηκαν όλες οι ασκήσεις ενώ υπάρχουν και τα αρχεία της matlab στον σχετικό φάκελο. Για κάθε συνάρτηση που ζητήθηκε υπάρχει μία αντίστοιχη με όνομα call_function_name.m Η οποία καλεί και δείχνει την σωστή λειτουργία την συνάρτηση με όνομα function_name.m

1 Άσκηση 1

Για την άσκηση 1 δημιουργήθηκε η συνάρτηση create_number.m η οποία δέχεται ως είσοδο έναν αριθμό 10 ψηφίων σε μορφή συμβολοσειράς (πχ create_number("5201700093")) και απεικονίζει το χρονικό σήμα που δημιουργείται από τα ψηφία με βάση τις συχνότητες των ψηφίων και την x[n]. Παρακάτω παρατίθενται η x[n] και ο συχνοτικός πίνακας των ψηφίων όπως δόθηκε στην εκφώνηση:

$$x_i[n] = cos(2\pi f1_i) + cos(2\pi f2_i), n = 0, 1, ..., N - 1$$

όπου οι f1,f2 λαμβάνονται από τον παρακάτω πίνακα 1 .

Το σήμα εξόδου αποτελείται από το xi[n] για κάθε ψηφίο πολλαπλασιασμένο με το παράθυρο διάρκειάς του (0,5 sec τόνος + 0.1 sec παύση)

Συχνότητες	1209Hz	1336Hz	1477Hz	1633Hz
697Hz	1	2	3	A
770Hz	4	5	6	B
852Hz	7	8	9	C
941Hz	*	0	#	D

Figure 1

Το χρονικό αυτό παράθυρο έχει την ακόλουθη μορφή:

$$w_i[n] = \begin{cases} 1 & 0.6(i-1) \leq n \leq 0.5i \\ 0 & else \end{cases}$$

Συνεπώς το τελικό σήμα εξόδου θα δίνεται από την ακόλουθη εξίσωση.

$$x[n] = \sum_{i=1}^{10} x_i[n] w_i[n]$$

Ενώ το διάγραμμα του σήματος για τον αριθμό μητρώου 1115201700093 είναι το figure 2:

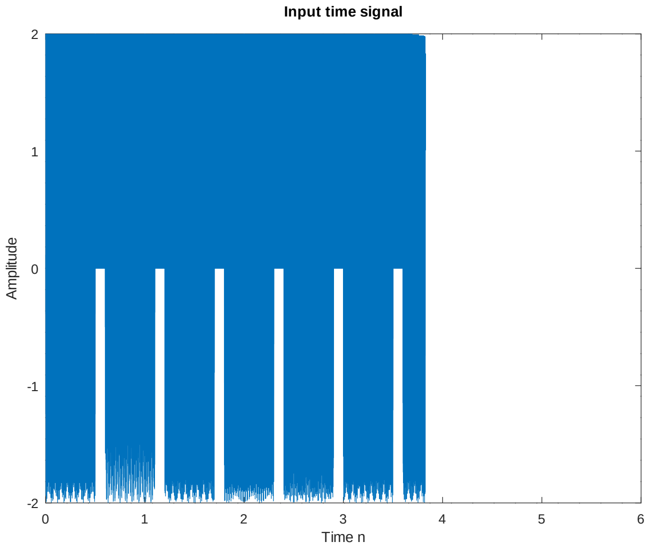

Figure 2

Για να βρούμε το συχνοτικό περιεχόμενο του x[n] θα εφαρμόσουμε σε αυτό μετασχηματισμό Fourier στο x[n] ως εξής:

$$X(w) = \sum_{n=0}^{N-1} x[n] e^{-jwn}$$

Ενώ στην συνέχεια θα πάρουμε το |X(w)| για να δείξουμε το πλάτος του μετασχηματισμού στο figure 3. Από το συγκεκριμένο διάγραμμα μπορούμε να δούμε τις συχνότητες του σήματος εισόδου, οι οποίες είναι αυτές που αντιστιχούν σε κάθε αριθμό από τον πίνακα figure 1. Η διαφορά στο πλάτος των διαφορετικών συχνοτήτων οφείλεται στο ότι κάποιες συχνότητες στο σήμα είσόδου υπάρχουν πιο πολλές φορές από άλλες και έτσι έχουν στην ουσία μεγαλύτερη ισχύ. Στο figure 3 η πιο ισχυρή συχνότητα βλέπουμε ότι είναι η 1336 Hz η οποία περιέχεται στα νούμερα 0,2,5 του αριθμού μητρώου μου ,ενώ το 0 υπάρχει και 3 φορές σε αυτόν. Είναι λογικό λοιπόν η συχνότητα η οποία υπάρχει πιο πολλές φορές να φαίνεται και πιο ισχυρή στο διάγραμμα.

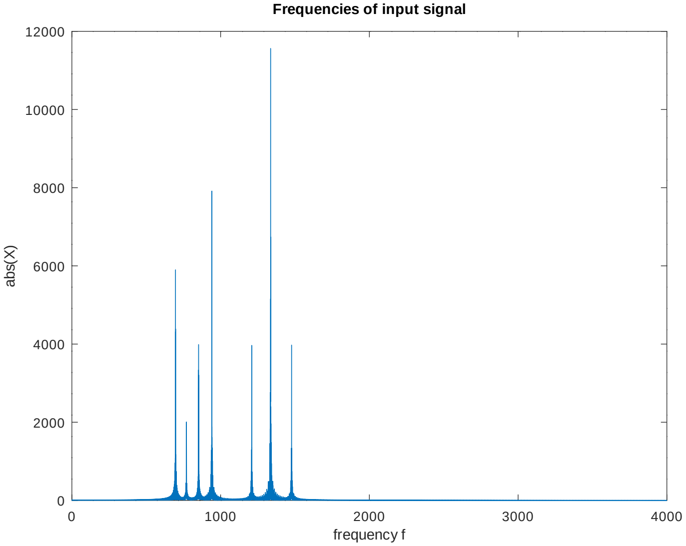
Frequencies of input signal
12000
10000
8000
6000
4000
2000
0
abs(X)
0
1000
2000
3000
4000
frequency f

Figure 3

2 Άσκηση 2

Για την άσκηση 2 δημιουργήθηκε η συνάρτηση filter_number.m η οποία δέχεται ως όρισμα το χρονικό σήμα εξόδου της create_number.m , εφαρμόζει μετασχηματισμό Fourier σε αυτό και κόβει από αυτό με την χρήση υψιπερατού φίλτρου την συχνότητα 697 Hz .Η συχνότητα αυτή επιλέχθηκε καθώς είναι η κοινή συχνότητα μεταξύ των αριθμών 1,2 και 3, ενώ επιλέξαμε υψιπερατό φίλτρο διότι δεν υπάρχει στον πίνακα figure 1 μικρότερη συχνότητα από την 697 Hz και θέλουμε να περάσουν όλες οι μεγαλύτερες.

Πιο συγκεκριμένα δοκιμάστηκαν προγραμματιστικά όλα τα παράθυρα με fs = 697 Hz και fp = 770 Hz καθώς είναι η αμέσως επόμενη συχνότητα που θέλουμε να περάσει. Τα δύο παράθυρα που δούλεψαν ήταν το Blackman και το Hamming ενώ τελικά επιλέχθηκε το Hamming καθώς έχει μικρότερο κόστος υλοποίησης M. Αναλυτικότερα για το συγκεκριμένο προόβλημα έχουμε:

$$fsamp = 8000Hz$$

$$fs = 697Hz => ws = \frac{2\pi fs}{fsamp} = \frac{697\pi}{4000} rad/sec$$

$$fp = 770Hz => wp = \frac{2\pi fp}{fsamp} = \frac{770\pi}{4000} rad/sec$$

$$\Delta\omega = \frac{770\pi - 697\pi}{4000} = \frac{73\pi}{4000} rad/sec$$

$$wc = \frac{ws + wp}{2} = \frac{733.5\pi}{8000} rad/sec$$

Καθώς δεν υπήρχε κάποιο άλλο δεδομένο για το πρόβλημα δοκιμάστηκαν όλα τα παράθυρα με βάσει τις παραπάνω προφιαγραφές μέχρι να δουλέψει κάποιο ώστε να καταλήξουμε σε αυτό με το χαμηλότερο κόστος. Τα δύο παράθυρα που δούλεψαν ήταν το Blackman και το Hamming ενώ για το κόστος τους έχουμε τα εξής.
Με βάση το Εύρος του κεντρικού λοβού έχουμε τον γενικό τύπο:

$$wc = \frac{12\pi}{M_{Blackman}} => M_{Blackman} = \left\lceil \frac{12\pi}{wc} \right\rceil = \left\lceil \frac{12\pi 8000}{733.5\pi} \right\rceil = 131$$

$$wc = \frac{8\pi}{M_{Hamming}} => M_{Hamming} = \left\lceil \frac{8\pi}{wc} \right\rceil = \left\lceil \frac{8\pi 8000}{733.5\pi} \right\rceil = 88$$

Ενώ με βάση το Εύρος Μετάβασης έχουμε:

$$\Delta\omega = \frac{9,19\pi}{M_{Blackman}} => M_{Blackman} = \left\lceil \frac{9,19\pi}{\Delta\omega} \right\rceil = \left\lceil \frac{9,19\pi 4000}{73\pi} \right\rceil = 504$$

$$\Delta\omega = \frac{6,27\pi}{M_{Hamming}} => M_{Hamming} = \left\lceil \frac{6,27\pi}{\Delta\omega} \right\rceil = \left\lceil \frac{6,27\pi 4000}{73\pi} \right\rceil = 344$$

Συνεπώς καθώς και τα δύο παράθυρα δουλέψαν θα επιλέξω το παράθυρο Hamming καθώς έχει μικρότερο κόστος υλοποίησης , ενώ από τα δύο Μ επλέγω το Μ = 344 καθώς με Μ = 88 δεν πέτυχα το ζητούμενο φιλτράρισμα.

Το ιδανικό υψιπερατό φίλτρο έχει απόκριση συχνότητας:

$$|H_{ideal}(w)| = \begin{cases} 1 & -\pi \leq w \leq -wc \cup wc \leq w \leq \pi \\ 0 & w \leq |wc| \end{cases}$$

$$=> H_{ideal}(w) = \begin{cases} e^{-j\alpha w} & -\pi \leq w \leq -wc \cup wc \leq w \leq \pi \\ 0 & w \leq |wc| \end{cases}$$

Η κρουστική απόκριση του ιδανικού υψιπερατού φίλτρου είναι η εξής:

$$h_{ideal}[n] = IDTFT\{H_{ideal}[w]\} = \frac{1}{2\pi}\int_{-\infty}^{\infty} H_{ideal}(w)e^{-jwn}\ dw =$$

$$= \frac{1}{2\pi}\int_{-\pi}^{-wc} e^{-jw(n-\frac{M}{2})}\ dw + \frac{1}{2\pi}\int_{wc}^{\pi} e^{-jw(n-\frac{M}{2})}\ dw =>$$

$$h_{ideal}[n] = sinc(n - \frac{M}{2}) - \frac{sin(wc(n-\frac{M}{2}))}{\pi(n-\frac{M}{2})}, M = 344, wc = \frac{733.5\pi}{8000} rad/sec$$

Η κρουστική απόκριση του παραθύρου Hamming είναι η εξής:

$$w[n] = \begin{cases} 0,54 - 0,64cos(\frac{2\pi n}{M}) & 0 \leq n \leq M \\ 0 & else \end{cases}$$

$$=> w[n] = \begin{cases} 0,54 - 0,64cos(\frac{2\pi n}{344}) & 0 \leq n \leq 344 \\ 0 & else \end{cases}$$

Επομένως το τελικό αποτέλεσμα θα είναι:

$$h_{desired}[n] = h_{ideal}[n]w[n] =$$

$$= \begin{cases} sinc(n - \frac{M}{2}) - \frac{sin(wc(n-\frac{M}{2}))}{\pi(n-\frac{M}{2})}(0,54 - 0,64cos(\frac{2\pi n}{344})) & 0 \leq n \leq 344 \\ 0 & else \end{cases}$$

$$= \begin{cases} sinc(n - 172) - \frac{sin(\frac{733.5\pi}{8000}(n-172))}{\pi(n-172)}(0,54 - 0,64cos(\frac{2\pi n}{344})) & 0 \leq n \leq 344 \\ 0 & else \end{cases}$$

Παρακάτω μπορούμε να δούμε τα αποτελέσματα που έδωσε η matlab για το συγκεκριμένο φίλτρο με M = 344.
Στο figure 4 μπορούμε να δούμε την απόκριση συχνότητας του υψιπερατού φίλτρου με το παράθυρο hamming και τις παραμέτρους που περιγράφηκαν.

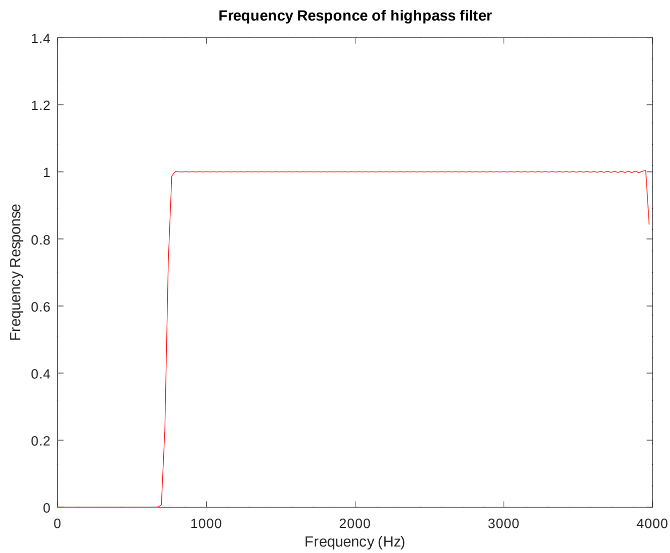

Figure 4

Στο figure 5 βλέπουμε το νέο σήμα που προκύπτει αν κόψουμε τελείως τα ψηφία που περιέχουν την συχνότητα 697. Επειδή ο αριθμός μητρώου μου περιέχει και το 1 και το 2 και το 3 βλέπουμε ότι το μήκος του είναι πλέον 7 αντί για 10.

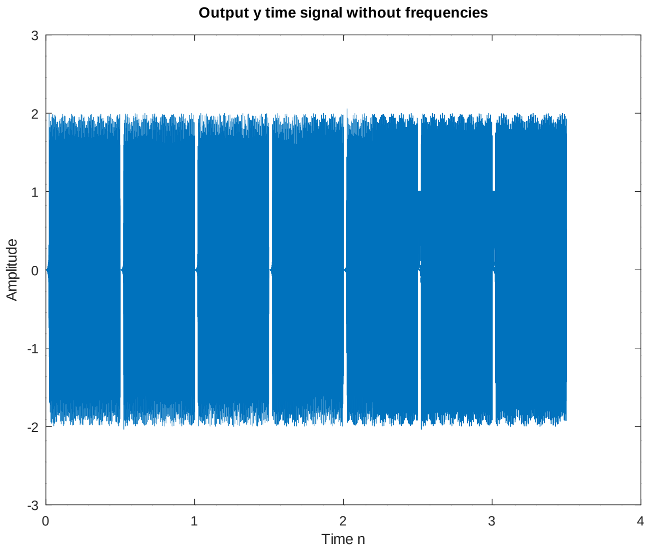

Figure 5

Στο figure 6 βλέπουμε πως το συγκεκριμένο φίλτρο είχε το επιθυμητό αποτέλεσμα, αφού η συχνότητα 697 κόπηκε από το σήμα, ενώ και σε κάποιες άλλες έχει μειωθεί το πλάτος αφού τις περιείχαν οι αριθμοί 1,2 και 3

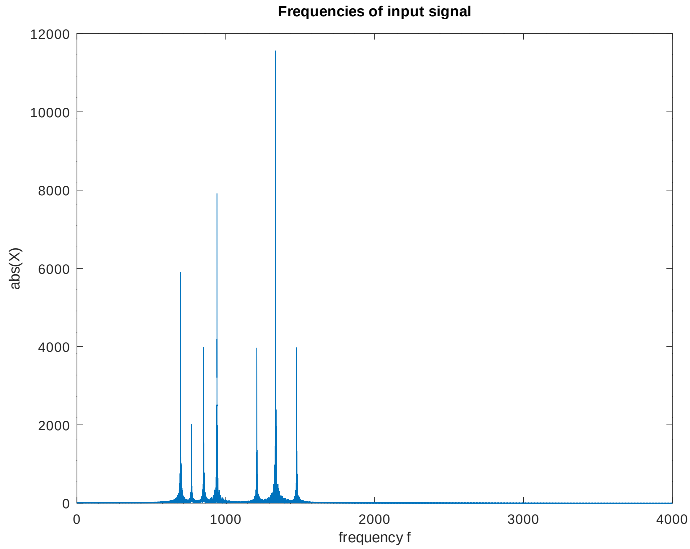

Figure 6

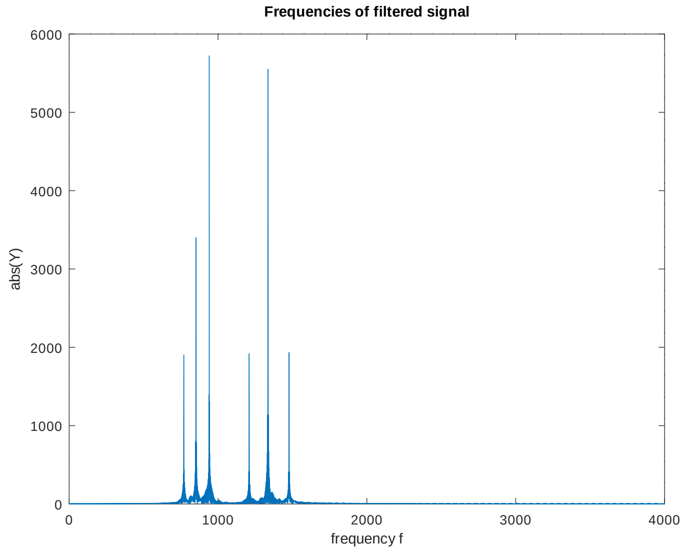

Figure 7

3 Άσκηση 3

Για την άσκηση 3 υλοποιήθηκε η συνάρτηση count_aces.m η οποία δέχεται το χρονικό σήμα εξόδου της create_number (figure 2) και μετράει πόσοι άσσοι υπάρχουν σε αυτό το σήμα. Στην ουσία ένα ψηφίο είναι άσσος αν περιέχει τις συχνότητες 697 και 1209 για τον λόγο αυτό εφαρμόζεται ο εξής αλγόριθμος στο σήμα εισόδου:

- Κόβουμε το σήμα σε στιγμιότυπα έτσι ώστε το κάθε στιγμιότυπο να περιέχει ένα ψηφίο του σήματος.
- Βρίσκουμε την έξοδο του ζωνοπερατού φίλτρου όταν είσοδος είναι το ψηφίο και για κάθε μία από τις 2 συχνότητες
- Αθροίζουμε τις δύο εξόδους για να πάρουμε την συνολική έξοδο
- Εφαρμόζουμε μετασχηματισμό Fourier στην έξοδο για να πάρουμε το συχνοτικό της περιεχόμενο.
- Αν το τελικό σήμα έχει τουλάχιστον 2 σημεία στην γραφική του παράσταση με πλάτος μεγαλύτερο από 1300 τότε είναι άσσος

Η τιμή 1300 επιλέχθηκε πειραματικά καθώς είδα ότι το σήμα του άσσου έχει δύο δείγματα με τιμή πάνω από 1300 στην έξοδο του από το ζωνοπερατό φίλτρο, κάτι το οποίο δεν συμβαίνει με κανένα άλλο σήμα. Θα μπορούσα να κάνω κάποια κανονικοποίηση στα νούμερα και να ελέγχω αν το σήμα εξόδου είναι συχνοτικά ίδιο με αυτό του άσσου αλλά αυτό δεν γινόταν καθώς λόγω του ότι το φίλτρο δεν είναι ιδανικό υπάρχει μία μείωση στο πλάτος των συχνοτήτων που παιρνάνε.

Αναλυτικότερα για το συγκεκριμένο προόβλημα και αν ονομάσουμε fpass την συχνότητα που θέλουμε να περάσει κάθε φορά για το ζωνοπερατό φίλτρο έχουμε:

$$fsamp = 8000Hz$$

$$fs1 = fpass - 40$$

$$fp1 = fpass - 20$$

$$fs2 = fpass + 40$$

$$fp2 = fpass + 20$$

Το εύρος μετάβασης μεταξύ των συχνοτήτων επιλέχθηκε μετά από πειραματισμούς έτσι ώστε να δουλεύει το φίλτρο, αλλά και να έχουμε ένα όσο μικρότερο M γίνεται.Όπως φαίνεται και παραπάνω το εύρος μετάβασης είναι 20 Hz ενώ το παράθυρο που επιλέχθηκε και εδώ είναι το Hamming καθώς είχε τα ίδια αποτελέσματα με το Blackman, αλλά μικρότερο κόστος υλοποίησης .

Επομένως έχουμε με βάση το εύρος ζώνης:

$$\Delta\omega = \frac{6,27\pi}{M_{Hamming}} => \frac{6,27\pi}{M_{Hamming}} = \frac{2\pi 20}{8000}$$

$$=> M_{Hamming} = \left\lceil \frac{6,27\pi 8000}{2\pi 20} \right\rceil = 1255$$

Θέτω M = M+1 για να έχω και γραμμική φάση, άρα M = 1256.

$$w_{Hamming}[n] = \begin{cases} 0,54 - 0,64cos(\frac{2\pi n}{1256}) & 0 \leq n \leq 1256 \\ 0 & else \end{cases}$$

Το ιδανικό ζωνοπερατό φίλτρο έχει απόκριση συχνότητας:

$$|H_{ideal}(w)| = \begin{cases} 1 & -wc2 \ \leq w \leq -wc1 \cup wc1 \leq w \leq wc2 \\ 0 & else \end{cases}$$

$$=> H_{ideal}(w) = \begin{cases} e^{-j\alpha w} & -wc2 \ \leq w \leq -wc1 \cup wc1 \leq w \leq wc2 \\ 0 & else| \end{cases}$$

$$h_{ideal}[n] = IDTFT\{H_{ideal}(w)\} = \frac{1}{2\pi} \int_{-\infty}^{\infty} H_{ideal}(w) e^{-jwn} \ dw =$$

$$= \frac{1}{2\pi} \int_{-wc2}^{-wc1} e^{-jw(n-\frac{M}{2})} \ dw + \frac{1}{2\pi} \int_{wc1}^{wc2} e^{-jw(n-\frac{M}{2})} \ dw =>$$

$$h_{ideal}[n] = \frac{sin(w_{c2}n)}{\pi n} - \frac{sin(w_{c1}n)}{\pi n}$$

Επομένως η τελική απόκρυση συχνότητας του φίλτρου μετά την παραθύρωση θα είναι:

$$h_{desired}[n] = h_{ideal}[n]w[n] =$$

$$= \begin{cases} \frac{sin(w_{c2}n)}{\pi n} - \frac{sin(w_{c1}n)}{\pi n}(0,54 - 0,64cos(\frac{2\pi n}{344})) & 0 \leq n \leq 1256 \\ 0 & else \end{cases}$$

Στην ουσία επειδή ο άσσος έχει δύο συχνότητες αθροίζω την απόκριση συχνότητας δύο ζωνοπεράτών φίλτρων κάθε φορά με τα αποτελέσματα να φαίνονται παρακάτω. Παρακάτω παρουσιάζονται τα διαγράμματα της matlab για τον άσσο:

Αρχικά βλέπουμε την απόκριση συχνότητας ενός άσσου ο οποίος περιέχει τις συχνότητες f1 = 697 Hz και f2 = 1209 Hz

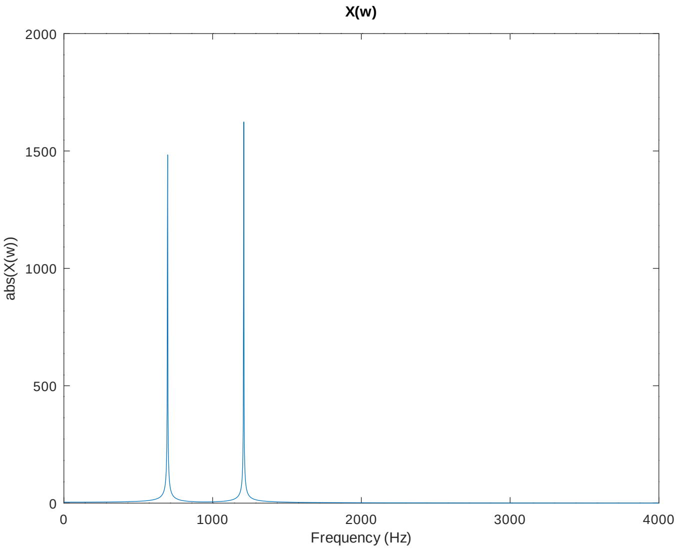

Figure 8

Στο figure 8 βλέπουμε την απόκριση συχνότητας του ζωνοπερατού φίλτρου έτσι ώστε να περνάνε οι συχνότητες του άσσου, ενώ στο figure 9 φαίνεται η έξοδος του συστήματος αυτού όταν η είσοδος είναι άσσος. Παρατηρούμε πως λόγω του ότι το φίλτρο δεν είναι ιδανικό, ενώ περνάνε οι συχνότητες που θέλουμε υπάρχει μία μικρή μείωση στην ισχύ τους .

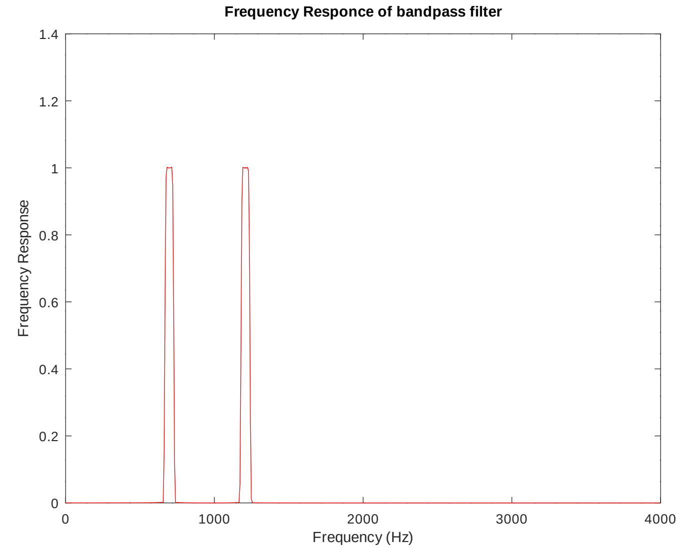

Figure 9

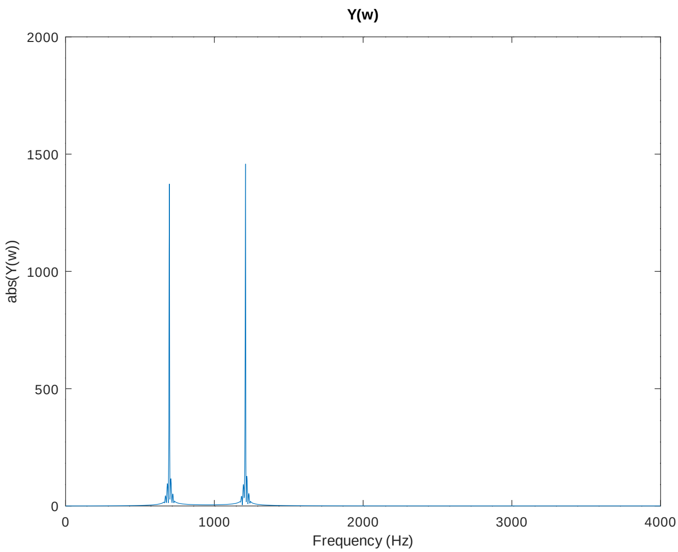

Figure 10

Τέλος παρακάτω παρουσιάζεται η φιλτραρισμένη έξοδος του αριθμού 3 και του αριθμού 5 για να επιβεβαιωθεί πως το φίλτρο δουλεύει σωστά. Αυτό που παρατηρείται είναι πως στον αριθμό 3 περνάει μόνο η συχνότητα 697 που περιέχεται και στον άσσο, ενώ στην ουσία στον αριθμό 5 δεν περνάει καμία καθώς το πλάτος των συχνοτήτων του είναι πολύ μικρό σε σχέση με την αρχική τιμή του που ήταν κοντά στο 1500.

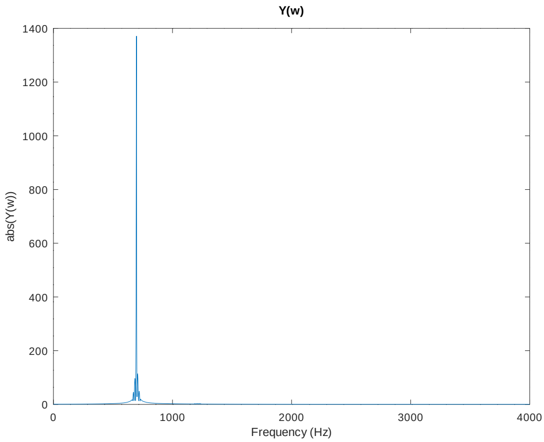

Figure 11: Filtered number 3

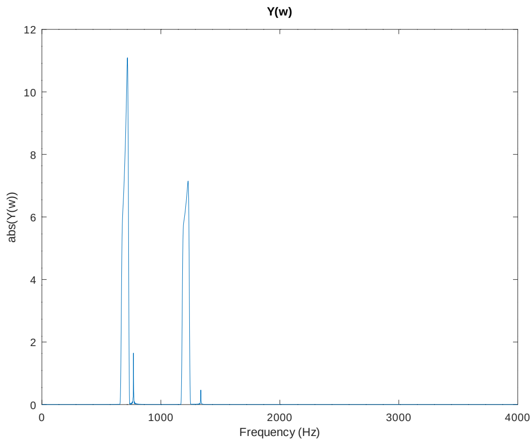

Figure 12: Filtered number 5

4 Άσκηση 4

Στην άσκηση 4 υλοποιήθηκε η συνάρτηση decode_DTMF.m η οποία δέχεται ως όρισμα εισόδου το σήμα του αρχείου 'secret_number.wav' και επιστρέφει τον αριθμό αριθμό που το σήμα αυτό αναπαριστά.

Η υλοποίηση της άσκησης αυτής βασίστηκε στην άσκηση 3 . Χρησιμοποιήθηκαν δηλαδή 10 ζωνοπερατά φίλτρα τα οποία περνάνε πάνω από κάθε ψηφίο του μετασχηματισμένου με Fourier σήματος εισόδου και όποιο σήμα εξόδου έχει δύο δείγματα με πλάτος πάνω από 1300 τότε ο αριθμός του είναι το ψηφίο που αντιστοιχεί στον αριθμό του φίλτρου που χρησιμοποιήθηκε.

Η μόνη διαφορά λοιπόν με την άσκηση 3 είναι ότι αντί να ελέγχουμε αν το ψηφίο είναι άσσος, ελέγχουμε αν είναι οποιοσδήποτε αριθμός . Αν καλέσετε το αρχείο call_decode_DTMF.m που χρησιμοποιεί την decode_DTMF.m θα δείτε το αποτέλεσμα:
6 9 4 5 2 1 4 4 8 0