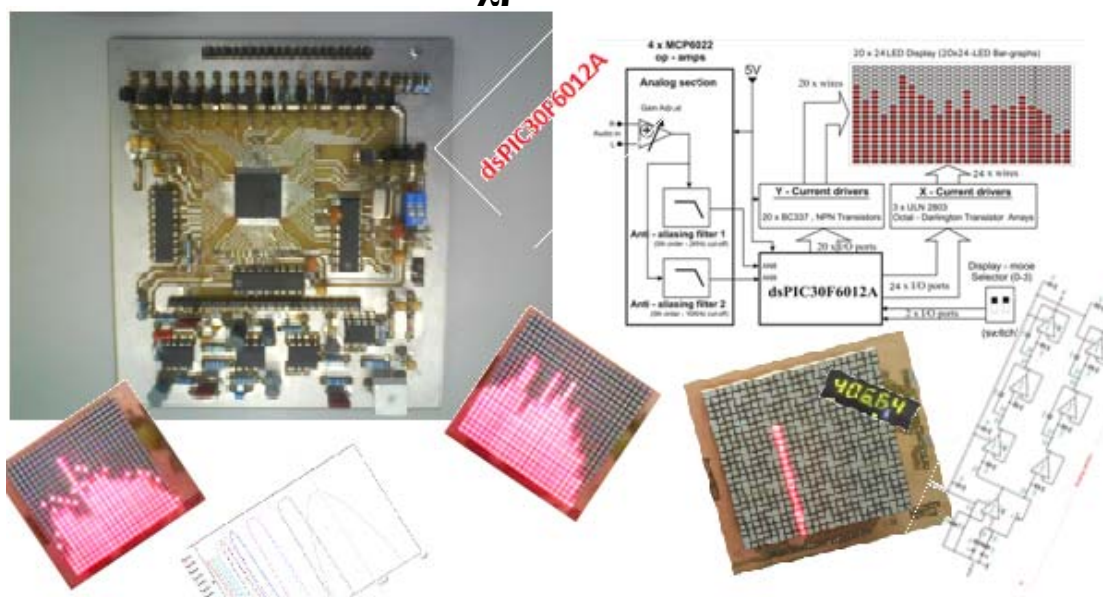


“Ψηφιακός αναλυτής φάσματος ακουστικών συχνοτήτων πραγματικού χρόνου”



Πτυχιακή Εργασία

υπό

*Δαγαλάκη Γ. Ιωάννη &
Αθανάσιου Γ. Κωτσόπουλου*

*Επιβλέπων: Γεώργιος Α. Αδαμίδης, Εργαστηριακός
Συνεργάτης*

Τ.Ε.Ι ΚΡΗΤΗΣ – ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ



ΧΑΝΙΑ 2010

ΤΟΜΕΑΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ – ΤΜΗΜΑ
ΗΛΕΚΤΡΟΝΙΚΗΣ
ΕΡΓΑΣΤΗΡΙΟ ΜΙΚΡΟΚΥΜΑΤΙΚΩΝ
ΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ
ΗΛΕΚΤΡΟΜΑΓΝΗΤΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

**Ψηφιακός αναλυτής φάσματος ακουστικών
συχνοτήτων πραγματικού χρόνου**

*ψηφιακή συσκευή ανάλυσης & απεικόνισης φάσματος
ακουστικού σήματος σε πραγματικό χρόνο
υπό*

*Δαγαλάκη Γ. Ιωάννη &
Αθανάσιου Γ. Κωτσόπουλου*

Τ.Ε.Ι ΚΡΗΤΗΣ – ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ

Επιβλέπων: Γεώργιος Α. Αδαμίδης, Εργαστηριακός Συνεργάτης

*Πτυχιακή Εργασία
για το τμήμα Ηλεκτρονικής Παραρτήματος Χανίων
του Τ.Ε.Ι Κρήτης
Τομέας Τηλεπικοινωνιών*

Χανιά, Απρίλιος 2010

ΤΟΜΕΑΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ – ΤΜΗΜΑ
ΗΛΕΚΤΡΟΝΙΚΗΣ

ΕΡΓΑΣΤΗΡΙΟ ΜΙΚΡΟΚΥΜΑΤΙΚΩΝ
ΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ
ΗΛΕΚΤΡΟΜΑΓΝΗΤΙΚΩΝ ΕΦΑΡΜΟΓΩΝ



ΧΑΝΙΩΝ



ΧΑΝΙΑ
2010

Περιεχόμενα εργασίας

Σελ.

| | |
|---|--------------|
| <i>Περίληψη</i> | 1 |
| 1.Εισαγωγή..... | 1 |
| 2.Η διαδικασία σχεδιασμού | 3 |
| 2.1 Σχεδιαστικό βήμα (2.1) | 3 |
| 2.2 Σχεδιαστικό βήμα (2.2)..... | 4 |
| 2.3 Σχεδιαστικό βήμα (2.3)..... | 5 |
| 2.4 Σχεδιαστικό βήμα (2.4)..... | 6 |
| 2.5 Σχεδιαστικό βήμα (2.5)..... | 11 |
| 3. Το Hardware..... | 12 |
| 3.1 Τα φίλτρα αντιαναδίπλωσης (anti –aliasing filters)..... | 13 |
| 3.2 Σχηματικά διαγράμματα..... | 14 |
| 3.3 Εξαρτήματα..... | 18 |
| 4. Το λογισμικό (Software)..... | 19 |
| 4.1 Η κύρια ρουτίνα..... | 22 |
| 4.2 Η ρουτίνα διακοπής του A/D (ISR)..... | 25 |
| 4.3 Η ρουτίνα ADC_Init..... | 26 |
| 4.4 Η ρουτίνα Filter_Bank..... | 27 |
| 4.5 Η ρουτίνα display..... | 28 |
| 5. Η απόδοση του συστήματος μέτρησης..... | 34 |
| • Κώδικας Common.h..... | Παράρτημα .1 |
| • Κώδικας ADC.c..... | Παράρτημα.2 |
| • Κώδικας Audio Spectrum.c | Παράρτημα.3 |
| • Τυπομένο κατασκευής..... | Παράρτημα.4 |

Περιεχόμενα Ψηφιακού Δίσκου:

Φακέλος:

- 1.Γραπτό μέρος*
- 2.Τυπομένο*
- 3.Video (display τρόποι απεικονισης)*
- 4.Κώδικας*
- 5.Κεντρικές συχνότητες και αρχείο mathcad για τον υπολογισμό τους*
- 6. Εικόνες , σχήματα και φωτογραφίες*

"Ψηφιακός αναλυτής φάσματος ακουστικών συχνοτήτων πραγματικού χρόνου"

“Επιλέξιμα μέρη”: dsPIC30F6012A

“Bonus Parts” : MCP6022

Περίληψη: - Στην τρέχουσα εργασία παρουσιάζεται ένας ψηφιακός αναλυτής φάσματος ακουστικών συχνοτήτων πραγματικού χρόνου. Αυτό το κύκλωμα αφορά τους επαγγελματίες ή hobbyistes που θα επιθυμούσαν να το ενσωματώσουν σε μια συσκευή ήχου ή να το χρησιμοποιήσουν ως αυτόνομη μονάδα. Η ψηφιακή συσκευή ανάλυσης φάσματος πραγματικού χρόνου μπορεί να απεικονίσει τη φασματική κατανομή ενέργειας ενός ακουστικού σήματος, χρησιμοποιώντας την τεχνολογία DSP. Χρησιμοποιεί το γρήγορο μετασχηματισμό Φουριέ (FFT) προκειμένου να αναλύει σε πραγματικό χρόνο ολόκληρο το φάσμα ενός ακουστικού σήματος. Τα αποτελέσματα του μετασχηματισμού Φουριέ, υποβάλλονται σε επεξεργασία που οδηγεί στη δημιουργία 24 παράλληλων φίλτρων συχνότητας. Οι έξοδοι των φίλτρων απεικονίζονται σε μια οθόνη 20 X 24 led. Η μονάδα υποστηρίζει τέσσερις τρόπους απεικόνισης.

Λέξεις κλειδιά: Επεξεργασία ψηφιακού σήματος, γρήγορος μετασχηματισμός κατά Φουριέ, συχνοτικό φάσμα, παράθυρο Hamming, πραγματικά και μιγαδικά διανύσματα, απόκριση συχνότητας, παράλληλα φίλτρα, γραφική παράσταση σε μπάρες, πλαίσιο εικόνας και ρυθμός εναλλαγής πλαισίων.

1. Εισαγωγή

Οι συσκευές ανάλυσης φάσματος υπάρχουν στα περισσότερα από τα σύγχρονα συστήματα ήχου. Ένας ακουστικός αναλυτής φάσματος χρησιμοποιείται για τις μετρήσεις στο φάσμα ακουστικών συχνοτήτων (από 0 έως 20000 Hz) και αποτελεί ένα πολύ ισχυρό εργαλείο για κάθε επαγγελματικό στούντιο ήχου. Πολλές σύγχρονες συσκευές ήχου, όπως οι ενισχυτές, ισοσταθμιστές, μίκτες και οι αναπαραγωγείς ήχου περιέχουν μια συσκευή ανάλυσης φάσματος, η οποία χρησιμοποιείται για να επιδείξει το ακουστικό φασματογράφημα (το φάσμα συχνότητας του ακουστικού σήματος) σε μια CRT ή LCD ή σε led οθόνη. Το φασματογράφημα προσφέρει ένα θεαματικό οπτικό εφέ αλλά επίσης παρέχει έναν γρήγορο τρόπο επιθεώρησης της ακουστικής απόδοσης κατά τη διάρκεια της αναπαραγωγής ήχου, εγγραφής ή κατά τη διάρκεια οποιασδήποτε τονικής ρύθμισης (**μπάσα, πρίμα, μεσαίες συχνότητες ή προσαρμοσμένες ρυθμίσεις ισοστάθμισης**). Σε αυτή την εργασία παρουσιάζεται ένας ψηφιακός αναλυτής φάσματος ακουστικών συχνοτήτων, πραγματικού χρόνου (φωτο 1). Αυτό το κύκλωμα είναι βασισμένο σε έναν επαγγελμαστή dsPIC30F6012A της Microchip, τον οποίο επιλέξαμε για ορισμένα χαρακτηριστικά που ταιριάζουν με τις απαιτήσεις της εφαρμογής αυτής. Διαθέτει μηχανή DSP, μετατροπέα A/D 12-bit, timer, 42 I/O θύρες και μεγάλη εσωτερική μνήμη. Επίσης στην παρούσα εφαρμογή χρησιμοποιείται μια 4άδα ενισχυτών τύπου MCP6022 της Microchip οι οποίοι χρησιμοποιούνται για τα φίλτρα αντι-αναδίπλωσης (anti – aliasing) και για ενίσχυση στο αναλογικό τμήμα του κυκλώματος.

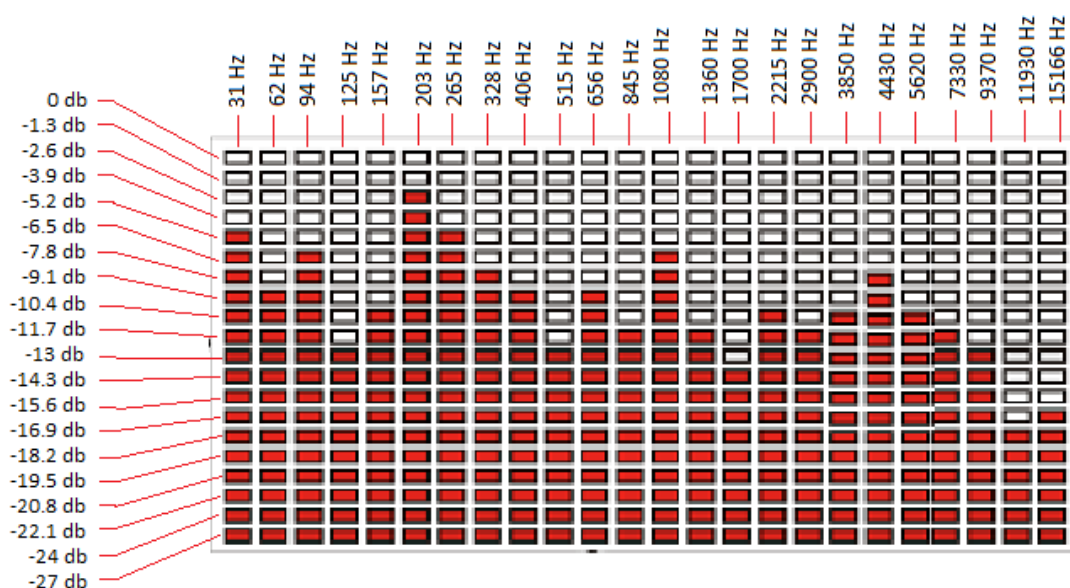
Ο ψηφιακός αναλυτής φάσματος ακουστικών συχνοτήτων, πραγματικού χρόνου μπορεί να συνδεθεί με οποιαδήποτε ακουστική συσκευή. Δέχεται ένα αναλογικό ακουστικό σήμα ως εισοδο. Ψηφιοποιεί και επεξεργάζεται αυτό το ακουστικό σήμα χρησιμοποιώντας ένα DSP. Υπολογίζει τη κατανομή της ακουστικής ενέργειας σημάτων σε 24 συγκεκριμένες ζώνες συχνότητας και την απεικονίζει σε ένα display 20x24 led. Όλα γίνονται σε πραγματικό χρόνο.

Ο ψηφιακός αναλυτής φάσματος ακουστικών συχνοτήτων, πραγματικού χρόνου υποστηρίζει 4 τρόπους απεικόνισης. Το φασματογράφημα (η κατανομή της ακουστικής ενέργειας του σήματος στο πεδίο της συχνότητας) μπορεί να απεικονιστεί με οποιοδήποτε από αυτούς τους 4 τρόπους:

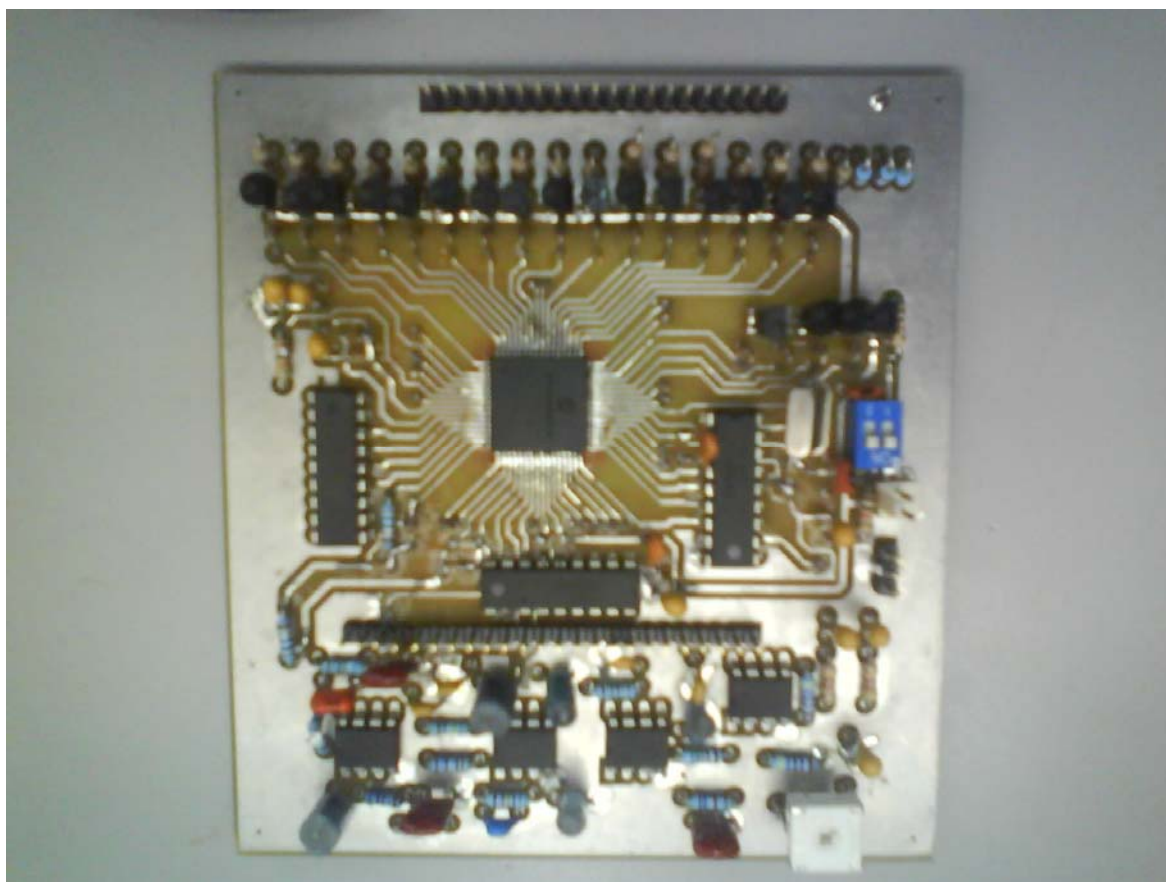
1. "Μπάρες" (είναι μια απεικόνιση πραγματικού χρόνου σε μπάρες)
2. "Μπάρες + κορυφές" (είναι μια απεικόνιση πραγματικού χρόνου με ταυτόχρονη συγκράτηση κορυφής)
3. "Αναρριχώμενες μπάρες" (είναι μια "αργή" απεικόνιση με μπάρες. Οι μπάρες "αναρριχούνται" ως την κορυφή. Μετά από αυτό, πέφτουν με ένα σταθερό ρυθμό έως ότου εμφανίζεται μια ισχυρότερη κορυφή)
4. "Αναρριχώμενες μπάρες + βροχή" (είναι μια "αργή" απεικόνιση με μπάρες που δίνει την αίσθηση "της βροχής". Οι μπάρες "αναρριχούνται" ως την κορυφή και μένουν εκεί. Μετά από αυτό, οι μπάρες πέφτουν με ένα σταθερό ρυθμό A και οι κορυφές πέφτουν με ένα σταθερό ρυθμό B έως ότου εμφανίζεται μια ισχυρότερη κορυφή. Θα πρέπει $A > B$)

Το φασματογράφημα απεικονίζεται σε 24 γραφικές παραστάσεις / μπάρες. Κάθε μπάρα-γραφική παράσταση αποτελείται από 20- leds. Το φασματογράφημα καλύπτει μια δυναμική περιοχή 30db με βήμα 1.3db. Αυτές οι λεπτομέρειες παρουσιάζονται στην εικόνα 1.

Η διαδικασία σχεδιασμού του συστήματος παρουσιάζεται στην παράγραφο 2. Το υλικό και το λογισμικό του συστήματος παρουσιάζονται στην παράγραφο 3 και 4 αντίστοιχα. Μερικές μετρήσεις απόδοσης του συστήματος παρουσιάζονται στην παράγραφο 5.



Εικόνα 1. Το φασματογράφημα απεικονίζεται σε 24 γραφικές παραστάσεις / μπάρες. Υποστηρίζονται 4 τρόποι απεικόνισης.



Φωτογραφία1 . Η κύρια βαθμίδα.

2. Η διαδικασία σχεδιασμού

Για το σχεδιασμό της συσκευής έχουν ληφθεί υπόψη τα παρακάτω:

1. Ο ψηφιακός αναλυτής φάσματος πρέπει να καλύπτει τη περιοχή συχνοτήτων απο 0 -20000Hz.
2. Το φασματογράφημα θα πρέπει να απεικονίζεται σε 24 μπάρες..
3. Κάθε μπάρα θα πρέπει να απεικονίζει το συνολικό επίπεδο ισχύος του σήματος σε μια συγκεκριμένη φασματική περιοχή. Επομένως, υπάρχουν 24 φασματικές ζώνες και χρειαζόμαστε 24 ζωνοπερατά φίλτρα.
4. Κάθε γραφική μπάρα αποτελείται από 20 LEDs
5. Η ανάλυση φάσματος συχνότητας θα πρέπει να εκτελεσθεί και να απεικονιστεί σε πραγματικό χρόνο.
6. Πρέπει να χρησιμοποιήσουμε φίλτρα αντι-αναδίπλωσης (anti-aliasing) το πολύ 5^{ης} τάξης (για τη μείωση του μεγέθους του κυκλώματος).

Σχεδιαστικό βήμα 2.1: Χρειαζόμαστε 24 ζωνοπερατά φίλτρα προκειμένου να διαχωριστεί το ακουστικό σήμα σε 24 ζώνες συχνοτήτων. Ποιά πρέπει να είναι η κεντρική συχνότητα του κάθε ζωνοπερατού φίλτρου;

Απάντηση: Η λύση στο πρόβλημα δεν είναι η μοναδική. Δεν υπάρχει καμία ανάγκη να έχουμε αναλυση καλύτερη από 30Hz στις χαμηλότερες συχνότητες του ακουστικού φάσματος επειδή το ανθρώπινο αυτί δεν έχει μεγάλη διακριτική ικανότητα σε αυτές τις συχνότητες. Κατά συνέπεια, επιλέγουμε 31Hz, 62Hz, 94Hz, 125Hz και 157Hz για κεντρική συχνότητα του 1^{ου}, 2^{ου}, 3^{ου}, 4^{ου} και του 5^{ου} φίλτρου αντίστοιχα:

$$CF(1) = 31\text{Hz} , CF(2) = 62\text{Hz} , CF(3) = 94\text{Hz} , CF(4) = 125\text{Hz} , CF(5) = 157\text{Hz} \quad (2.1)$$

Δεν υπάρχει επίσης καμία ανάγκη να απεικονίσουμε το φασματικό περιεχόμενο με βήμα καλύτερο από 5KHz στον ανώ όριο του ακουστικού φάσματος. Κατά συνέπεια, επιλέξαμε 15KHz για κεντρική συχνότητα του 24^{ου} φίλτρου .

$$CF(24) = 15\text{KHz} \quad (2.2)$$

Λαμβάνοντας υπόψη την τονική αίσθηση του ανθρώπινου αυτιού με βάση την οκτάβα, ο καλύτερος τρόπος να βρεθούν οι κεντρικές συχνότητες για τα 6 ..7 ..8 .. 9 ως και 23 φίλτρα λύνουμε τις εξισώσεις (2.1) - (2.2) και (2,3).

$$CF(i) = CF(5) \cdot 2^{(i-5)*a} \quad \text{Για } i=6,7,8, \dots 24. \text{ Όπου } a, \text{ είναι ένα σταθερό κλάσμα της οκτάβας} \quad (2.3)$$

Από την επίλυση (2,1), (2.2) και (2.3) λαμβάνουμε: $a \approx 0.346$ οκτάβες και

| i | CF(i) (in Hardware) | CF(i) | i | CF(i) (in Hardware) | CF(i) |
|----|---------------------|------------|----|---------------------|----------|
| 1 | 31 Hz | 60.155 Hz | 13 | 1080 Hz | 1069 Hz |
| 2 | 62 Hz | 76.459 Hz | 14 | 1360 Hz | 1359 Hz |
| 3 | 94 Hz | 97.182 Hz | 15 | 1700 Hz | 1728 Hz |
| 4 | 125 Hz | 123.522 Hz | 16 | 2215 Hz | 2196 Hz |
| 5 | 157 Hz | 157 Hz | 17 | 2900 Hz | 2791 Hz |
| 6 | 203 Hz | 199.552 Hz | 18 | 3850 Hz | 3548 Hz |
| 7 | 265 Hz | 253.637 Hz | 19 | 4430 Hz | 4509 Hz |
| 8 | 328 Hz | 322.381 Hz | 20 | 5620 Hz | 5731 Hz |
| 9 | 406 Hz | 409.756 Hz | 21 | 7330Hz | 7285 Hz |
| 10 | 515 Hz | 520.813 Hz | 22 | 9370Hz | 9259 Hz |
| 11 | 656Hz | 661.969 Hz | 23 | 11930Hz | 11770 Hz |
| 12 | 845Hz | 841.384 Hz | 24 | 15166Hz | 14960 Hz |

Πίνακας 1. Η κεντρική συχνότητα κάθε φίλτρου

Σχεδιαστικό βήμα 2.2: Πρέπει να υλοποιήσουμε 24 ζωνοπερατά φίλτρα προκειμένου να διαχωριστεί το ακουστικό σήμα σε ένα σύνολο απο 24 ζώνες συχνοτήτων. Μπορούμε να το πετύχουμε σε πραγματικό χρόνο;

Απάντηση: Εξετάζοντας την τάξη των φίλτρων που απαιτείται (τάξη φίλτρων μεγαλύτερη από 8 στις χαμηλές συχνότητες προκειμένου να επιτευχθεί η ανάλυση 30 Hz) και την ταχύτητα του DSP (μέχρι 30MIPS) είναι αδύνατο να γίνει στον πραγματικό χρόνο με κάποια κλασσική προσέγγιση FIR ή IIR. Ο μόνος τρόπος είναι ο FFT (γρήγορος μετασχηματισμός κατά Φουριέ).

Σχεδιαστικό βήμα 2.3: Ποιο πρέπει να είναι το μήκος του FFT και η συχνότητα δειγματοληψίας προκειμένου να επιτευχθεί η απαραίτητη διακριτική ικανότητα στο πεδίο της συχνότητας;

Απάντηση: Εξετάζοντας τον πίνακα 1, παρατηρούμε ότι χρειαζόμαστε μια διακριτική ικανότητα των 30Hz στις χαμηλές συχνότητες. Η διακριτική ικανότητα εξαρτάται από το μήκος του FFT και τη συχνότητα δειγματοληψίας. Για έναν FFT N-σημείων το βήμα συχνότητας df είναι:

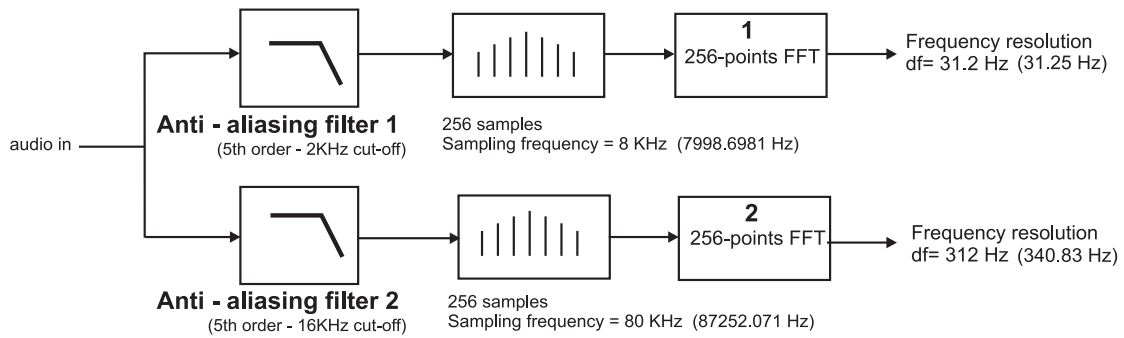
$$df = \frac{\text{Sampling_frequency}}{N} \quad (2.4)$$

Επίσης, η συχνότητα δειγματοληψίας πρέπει να συμμορφωθεί με το θεώρημα δειγματοληψίας Nyquist, το οποίο επιβάλλει ότι ένα συνεχές σήμα μπορεί να ψηφιοποιηθεί σωστά μόνο όταν η μεγαλύτερη συχνοτική συνιστώσα που αυτό περιέχει είναι μικρότερη από το μισό της συχνότητας δειγματοληψίας:

$$\text{Sampling_frequency} > 2 \cdot f_{\max} \quad (2.5)$$

Για ένα ακουστικό σήμα, η μέγιστη συχνότητα f_{\max} περιορίζεται στα 20 KHz με κατάλληλο φιλτράρισμα. Γι αυτό το λόγο το όριο Nyquist είναι στα 40KHz. Προκειμένου να αποφύγουμε ογκώδη anti-aliasing φίλτρα, είναι καλύτερο να τεθεί μια συχνότητα δειγματοληψίας πολύ μεγαλύτερη από το όριο Nyquist. Η χρησιμοποίηση ενός antialiasing φίλτρου τύπου Butterworth 5^{ης} τάξης, καθώς και η χρήση συχνότητας δειγματοληψίας γύρω στα 80KHz είναι ένας ικανοποιητικός ρυθμός. Από την εξίσωση (2.4) καταλήγουμε στο συμπέρασμα ότι το N πρέπει να είναι μεγαλύτερο από 2667 προκειμένου να επιτευχθεί το βήμα 30Hz σε μια συχνότητα δειγματοληψίας 80KHz. Στην πράξη, το N πρέπει να είναι μεγαλύτερο από 4096, επειδή θα πρέπει υποχρεωτικά να είναι δύναμη του 2.

Ένας μετασχηματισμός Φουριέ 4096-σημείων (FFT) δεν μπορεί να υλοποιηθεί χρησιμοποιώντας ένα dsPIC σε πραγματικό χρόνο. Αυτό λόγω περιορισμών που απορρέουν λόγω μνήμης, ταχύτητας και λόγω του ότι το dsPIC είναι τύπου fractional-integer). Αφ' ετέρου, δεν υπάρχει καμία ανάγκη για διακριτική ικανότητα πάνω από 30Hz στις συχνότητες επάνω από 126Hz, όπως μπορεί να φανεί από τον πίνακα 1. Επομένως η λύση στο πρόβλημα είναι η χρήση 2 διαφορετικών συχνοτήτων δειγματοληψίας, 2 ξεχωριστών μετασχηματισμών (2 X 256 points) FFT και 2 antialiasing φίλτρων όπως φαίνεται στο σχήμα 1:



Σχήμα 1. Η απάντηση στο σχεδιαστικό πρόβλημα 2.3 (οι πραγματικές συχνότητες δειγματοληψίας, που χρησιμοποιούνται στο κύκλωμα, είναι 7998.6981 Hz και 87252.071Hz αντίστοιχα και παρέχουν δικριτική ικανότητα της τάξης του 31.25Hz και 340.83 Hz.)

Όπως μπορεί να φανεί από το σχήμα 1 και τον πίνακα 1, ο 1^{ος} FFT μπορεί να παρέχει το βήμα που απαιτείται για τις χαμηλές συχνότητες, ενώ ο 2^{ος} FFT παρέχει ένα πολύ καλό βήμα για τις συχνότητες άνω των 1KHz. Φυσικά, η λύση, που παρουσιάζεται στο σχήμα 1, δεν είναι μοναδική στο πρόβλημα αλλά λειτουργεί τέλεια με το dsPic.

Σχεδιαστικό βήμα 2.4: Πρέπει να προσδιορίσουμε την ισχύ του ακουστικού σήματος σε ένα σύνολο 24 περιοχών συχνοτήτων. Πρέπει να χρησιμοποιήσουμε 24 ζωνοπερατά φίλτρα, που το καθένα έχει κεντρική συχνότητα CF (ι) όπως φαίνεται στον πίνακα 1. Κάθε φίλτρο θα πρέπει να μπορεί να εξάγει τη συνολική ισχύ του σήματος που περιέχεται σε συγκεκριμένη περιοχή συχνοτήτων. Η ισχύς που εξάγεται από τα φίλτρα θα πρέπει να απεικονίζεται σε ένα display 20x24 led, σε 24 μπάρες. Πώς μπορούμε να το κάνουμε αυτό με FFT και το σχήμα 1;

Απάντηση: Χρειαζόμαστε έναν αλγόριθμο υλοποίησης 24 παράλληλων φίλτρων προκειμένου να μπορούμε να διαχωρίσουμε το ακουστικό σήμα σε 24 μη ομοιόμορφες ζώνες συχνοτήτων χρησιμοποιώντας ένα μετασχηματισμό Φουριέ FFT. Πρέπει να χρησιμοποιήσουμε κάποια μαθηματικά για να κάνουμε αυτό:

Η λύση δίνεται με βάση το παρακάτω παράδειγμα.

Εξετάζοντας την περίπτωση όταν εφαρμόζεται ένας καθαρός τόνος S μοναδιαίου πλάτους(0db), εφαρμόζεται στην εισοδο της συσκευής. Αυτός ο τόνος συχνότητας f μπορεί να ψηφιοποιηθεί με μια συχνότητα δειγματοληψίας SR. Αυτό θα οδηγήσει σε μια ψηφιακή ακολουθία S(n, f, SR) όπως περιγράφεται στην εξίσωση (2,6):

$$S(n, f, SR) = \sin\left(2 \cdot \pi \cdot f \cdot \frac{n}{SR}\right), \quad n=0,1,2,3,\dots, \pi=3.14\dots \quad (2.6)$$

Έστω ότι αποθηκεύουμε τα πρώτα 256 δείγματα αυτής της ακολουθίας στη μνήμη και εφαρμόζουμε σε αυτά ένα παράθυρο Hamming κανονικοποιημένο στο 0,5.

Αυτό οδηγεί Y(n, f, SR):

$$w(n) = 0.5 \cdot \left(0.54 - 0.46 \cdot \cos\left(2 \cdot \pi \cdot \frac{n}{255}\right)\right), \quad n=0,1,2 \dots 255 \quad (2.7)$$

$$Y(n, f, SR) = W(n) \cdot S(n, f, SR) \quad (2.8)$$

Κατόπιν, εφαρμόζουμε έναν γρήγορο μετασχηματισμό κατά Φουριέ 256-σημείων $Y(n, f, SR)$, οπότε προκύπτει η ακολουθία $X(n, f, SR)$:

$$X(n, f, SR) = FFT[Y(n, f, SR)] = \frac{1}{256} \cdot \sum_{k=0}^{255} Y(k, f, SR) \cdot e^{i \cdot 2 \cdot \pi \cdot n \cdot \frac{k}{256}} \quad (2.9)$$

Παρατήρηση: $S(n, f, SR)$, $w(n)$ και $Y(n, f, SR)$ είναι πραγματικά σήματα και $X(n, f, SR)$ είναι ένα μιγαδικό σήμα. Έπειτα υπολογίζουμε το τετράγωνο του μέτρου του σήματος $X(n, f, SR)$ και παίρνουμε τη ψηφιακή ακολουθία $P(n, f, SR)$ που είναι η ψηφιακή αναπαράσταση της ισχύος του σήματος εισόδου στο πεδίο της συχνότητας.

$$P(n, f, SR) = (\text{Re}(X(n, f, SR)))^2 + (\text{Im}(X(n, f, SR)))^2 \quad (2.10)$$

Έπειτα ορίζουμε τις συναρτήσεις $\text{Bar0}(f)$, $\text{Bar1}(f)$, $\text{Bar23}(f)$, από την ψηφιακή ακολουθία $P(n, f, SR)$ όπως παρακάτω:

$$\text{Bar0}(f) = P(1, f, SR1) \quad (2.11)$$

$$\text{Bar1}(f) = P(2, f, SR1) \quad (2.12)$$

$$\text{Bar2}(f) = P(3, f, SR1) \quad (2.13)$$

$$\text{Bar3}(f) = P(4, f, SR1) \quad (2.14)$$

$$\text{Bar4}(f) = P(5, f, SR1) \quad (2.15)$$

$$\text{Bar5}(f) = \sum_{k=6}^7 P(k, f, SR1) \quad (2.16)$$

$$\text{Bar6}(f) = \sum_{k=8}^9 P(k, f, SR1) \quad (2.17)$$

$$\text{Bar7}(f) = \sum_{k=10}^{11} P(k, f, SR1) \quad (2.18)$$

$$\text{Bar8}(f) = \sum_{k=12}^{14} P(k, f, SR1) \quad (2.19)$$

$$\text{Bar9}(f) = \sum_{k=15}^{18} P(k, f, SR1) \quad (2.20)$$

$$\text{Bar10}(f) = \sum_{k=19}^{23} P(k, f, SR1) \quad (2.21)$$

$$\text{Bar11}(f) = \sum_{k=24}^{30} P(k, f, SR1) \quad (2.22)$$

$$Bar12(f) = \sum_{k=31}^{36} P(k, f, SR1) \quad (2.23)$$

$$Bar13(f) = P(4, f, SR2) \quad (2.24)$$

$$Bar14(f) = P(5, f, SR2) \quad (2.25)$$

$$Bar15(f) = \sum_{k=6}^7 P(k, f, SR2) \quad (2.26)$$

$$Bar16(f) = \sum_{k=8}^9 P(k, f, SR2) \quad (2.27)$$

$$Bar17(f) = \sum_{k=10}^{11} P(k, f, SR2) \quad (2.28)$$

$$Bar18(f) = \sum_{k=12}^{14} P(k, f, SR2) \quad (2.29)$$

$$Bar19(f) = \sum_{k=15}^{18} P(k, f, SR2) \quad (2.30)$$

$$Bar20(f) = \sum_{k=19}^{24} P(k, f, SR2) \quad (2.31)$$

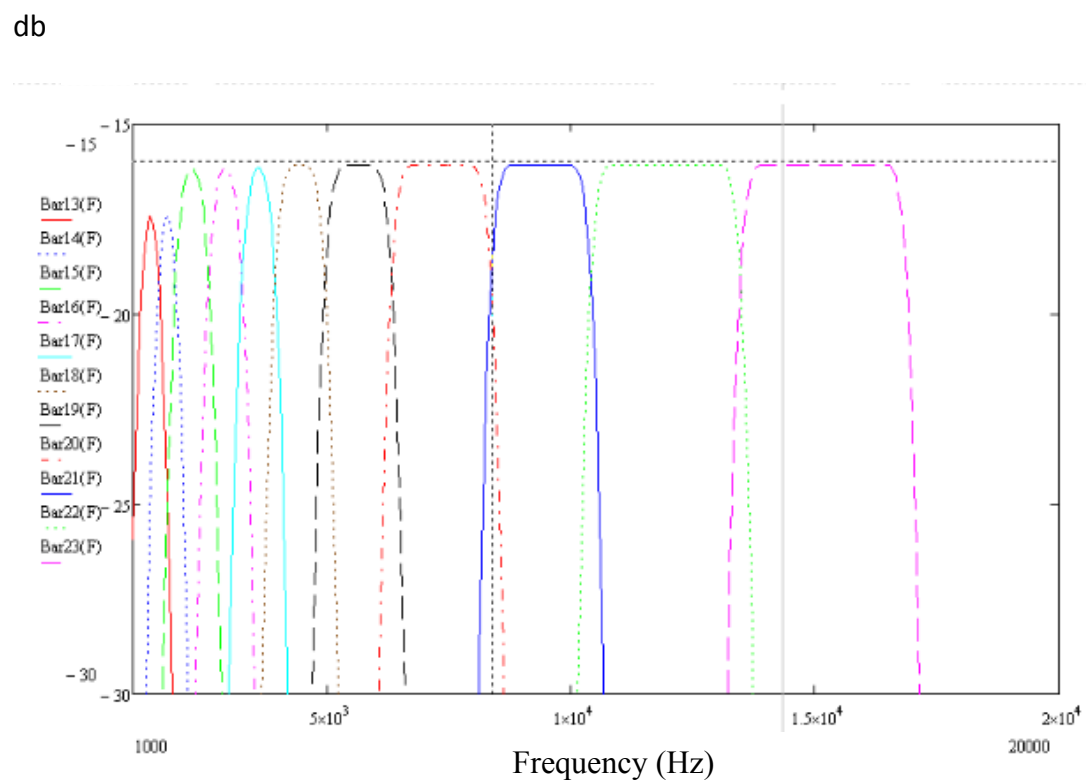
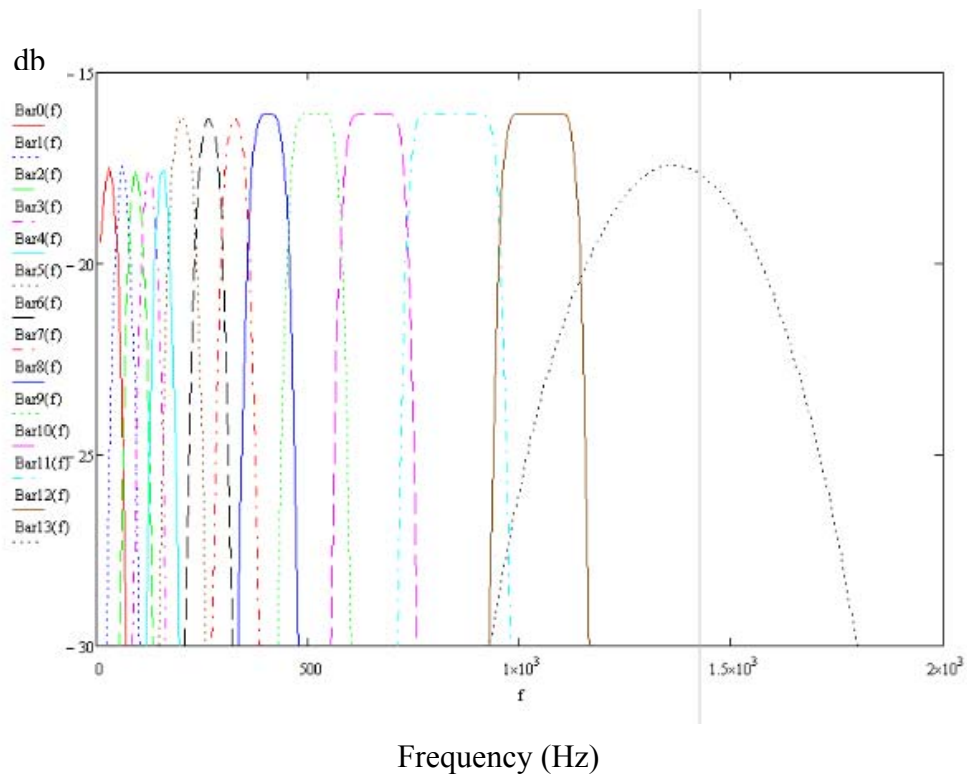
$$Bar21(f) = \sum_{k=25}^{30} P(k, f, SR2) \quad (2.32)$$

$$Bar22(f) = \sum_{k=31}^{39} P(k, f, SR2) \quad (2.33)$$

$$Bar23(f) = \sum_{k=40}^{49} P(k, f, SR2) \quad (2.34)$$

Όπου SR1, SR2 είναι οι συχνότητες δειγματοληψίας 7998.6981 Hz και 87252.071 Hz, αντίστοιχα.

Έπειτα, σχεδιάζουμε τις γραφικές παραστάσεις Bar0(f), Bar1(f).....Bar23(f):



Γραφικές παραστάσεις 1,2. Τα 24 φίλτρα, καθένα που έχει μια κεντρική συχνότητα CF (i) σύμφωνα με τον πίνακα 1

Όπως μπορεί να φανεί από τα διαγράμματα 1 και 2, Bar0(f), Bar1(f).....Bar23(f) είναι 24 ζωνοπερατά φίλτρα, που το καθένα έχει μια κεντρική συχνότητα CF(i) σύμφωνα με τον πίνακα 1. Κατά συνέπεια, έχουμε καταφέρει να υλοποιήσουμε 24 μη ομοιόμορφα ζωνοπερατά με τη βοήθεια ενός μετασχηματισμού Φουριέ FFT. Τα 24 φίλτρα αλληλεπικαλύπτονται κοντά στα σημεία μισής ισχύος (-3db κάτω από την κορυφή). Τα φίλτρα 5-12 και φίλτρα 15-23 είναι Flat-top στα -16db. Στα φίλτρα 0.1.2.3.4.13 και 14 έχουν 1db περισσότερη απώλεια από τα άλλα.

Τελικά, προκειμένου να υλοποιηθεί το επιθυμητό σύστημα των 24 παράλληλων φίλτρων που τη βοήθεια του μετασχηματισμού FFT θα πρέπει να ακολουθήσουμε τα βήματα:

1. Λαμβάνουμε 256 δείγματα του ακουστικού σήματος $SR1=7998.6981$ Hz (ρυθμός δειγματοληψίας.)
2. Εφαρμόζουμε ένα παράθυρο Hamming 256-σημείων σε αυτά.
3. Υπολογίζουμε το μετασχηματισμό Φουριέ του προκύπτοντος διανύσματος.
4. Υπολογίζουμε το διανυσματικό μέγεθος P, του προκύπτοντος διανύσματος X.
5. Από το διάνυσμα P, υπολογίζουμε τους συντελεστές Bar0, Bar1, Bar2, Bar3, Bar4, Bar5, Bar6, Bar7, Bar8, Bar9, Bar10 Bar11 και Bar12
6. Λαμβάνουμε 256 δείγματα του ακουστικού σήματος $SR2=87252.071$ Hz. (ρυθμός δειγματοληψίας.)
7. Εφαρμόζουμε ένα παράθυρο Hamming 256-σημείων σε αυτά.
8. Υπολογίζουμε τα 256 σημεία του μετασχηματισμού Φουριέ FFT του προκύπτοντος διανύσματος.
9. Υπολογίζουμε το διανυσματικό μέγεθος P, του προκύπτοντος διανύσματος X.
10. Από το διανυσματικό μέγεθος P, υπολογίζουμε τις τιμές Bar13, Bar14, Bar15, Bar16, Bar17, Bar18 Bar19, Bar20, Bar21, Bar22 και Bar23 δείγματα.

Μετά από αυτά τα βήματα, μπορούμε απλά να απεικονίσουμε τα Bar0 – Bar23 σε μια μπάρα led το καθένα, και έτσι να έχουμε μια ακουστική συσκευή ανάλυσης φάσματος 24 περιοχών. Το dsPIC30F6012A έχει αρκετή μνήμη και ταχύτητα προκειμένου να εκτελεστούν τα βήματα 1 έως 10 σε πραγματικό χρόνο.

ΣΗΜΕΙΩΣΗ 1: Το παράθυρο Hamming πρέπει να είναι κανονικοποιημένο στο 0,5. Αυτό γιατί χρησιμοποιούμε ένα fractional (κλασματικό) DSP και οι τιμές της ακολουθίας $Y(n,f,SR)$ πρέπει να είναι μικρότερες ή το πολύ ίσες με 0,5 (λόγω των κλασματικών περιορισμών του DSP)

ΣΗΜΕΙΩΣΗ 2: Το ακουστικό σήμα είναι ένας συνδυασμός πολλών απλών τόνων, οι οποίοι έχουν τυχαία φάση (η φάση δεν μας αφορά). Γι αυτό δεν υπάρχει κανένας όρος φάσης στην εξίσωση (2,6). Θέλουμε ακριβώς να υπολογίσουμε τη συνολική ισχύ που περιλαμβάνεται σε κάθε μία από τις 24, φασματικές περιοχές.

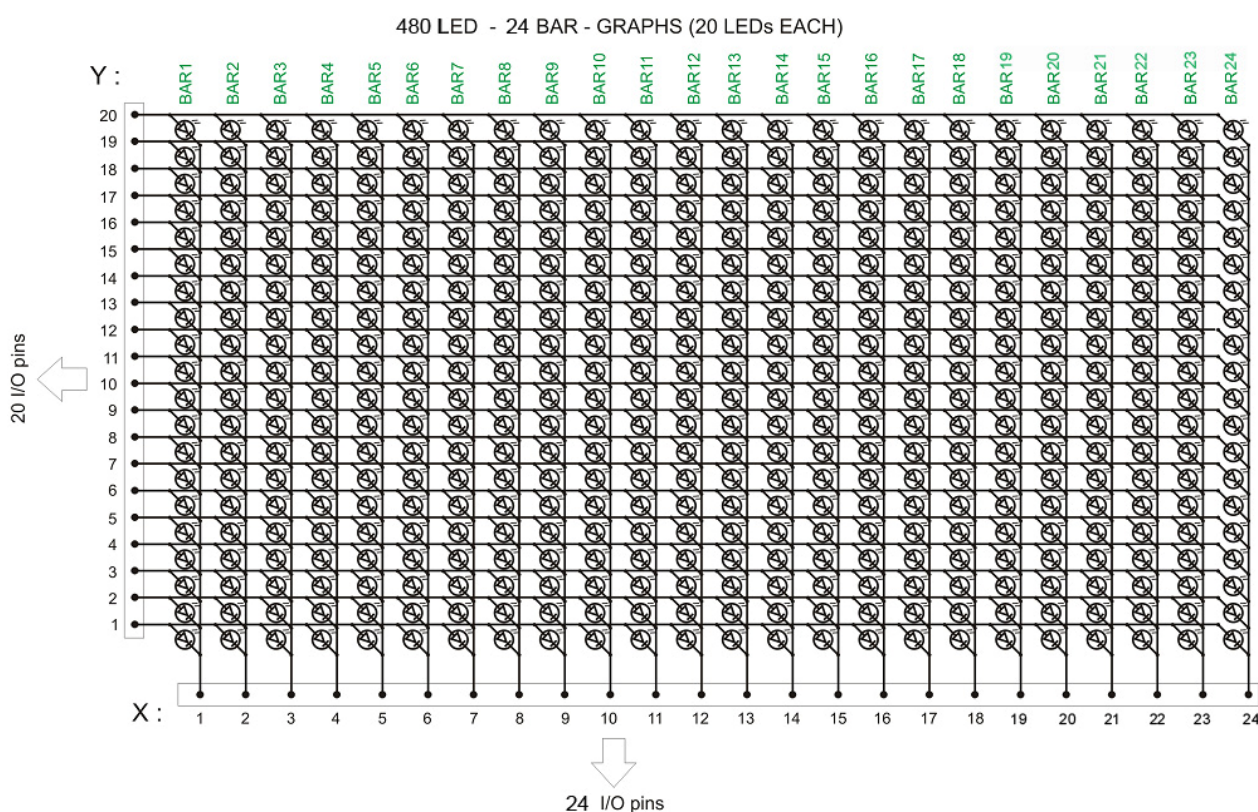
ΣΗΜΕΙΩΣΗ 3: Οι συντελεστές Bar i αντιπροσωπεύουν τη συνολική ισχύ που περιλαμβάνεται μέσα στη ζώνη i. Χωρίς τη χρήση ενός παραθύρου δεν θα μπορούσαμε να επιτύχουμε την flat-top συμπεριφορά των φίλτρων και θα είχαμε επίσης πάρα πολύ

θόρυβο (φίλτρα με σημαντική πλευρική απόκριση). Οποιοδήποτε παράθυρο μπορεί να χρησιμοποιηθεί αντί για το Hamming, αλλά αυτό θα αλλάξει την απώλεια του φίλτρου και την οξύτητά του. Το παράθυρο Hamming προσφέρει τη μικρότερη δυνατή απώλεια.

ΣΗΜΕΙΩΣΗ 4: Ο FFT οδηγεί στη σημαντική μείωση της δυναμικής περιοχής, σε ένα κλασματικό DSP. Αυτό οφείλεται στους κλασματικούς αριθμητικούς περιορισμούς κατά τη διάρκεια του υπολογισμού FFT. Για τα 256-σημεία FFT, η δυναμική περιοχή των δειγμάτων «μπάρες» είναι σχεδόν 30db. Γι αυτό η δυναμική περιοχή του ψηφιακού αναλυτή φάσματος ακουστικών συχνοτήτων, πραγματικού χρόνου περιορίζεται σε 30db.

Σχεδιαστικό βήμα 2.5: Πρέπει να απεικονίσουμε κάθε συντελεστή Bar i σε μια μπάρα 20 led. Έχουμε 24 δείγματα, κατά συνέπεια 24 συντελεστές Bar i οπότε χρειαζόμαστε 24 μπάρες των 20 led. Πώς μπορεί να γίνει αυτή η απεικόνιση;

Απάντηση: Καταρχήν πρέπει να αποφασίσουμε πόσα LEDs θέλουμε να έχουμε σε κάθε μπάρα. Σε αυτή την εργασία θα χρησιμοποιούμε 20 LEDs σε κάθε μπάρα. Χρειαζόμαστε $24 \times 20 = 480$ LEDs προκειμένου να απεικονίσουμε το φασματογράφημα των 24 ζωνών. Εάν αποφασίσουμε να οδηγήσουμε αυτά τα LEDs ταυτόχρονα θα χρειαστούμε 480 I/O pins και ένα συνολικό ρεύμα περίπου 9,6A που φυσικά, δεν είναι μια πολύ καλή λύση. Αντί να οδηγήσουμε όλα τα LEDs ταυτόχρονα, μπορούμε να χρησιμοποιήσουμε 44 I/O pins προκειμένου να οδηγηθούν αυτά τα LEDs με σειριακό τρόπο. Θα χρησιμοποιήσουμε 24 I/O pins για τη X συντεταγμένη, και 20 I/O pins για τη Y συντεταγμένη των leds όπως φαίνεται στο σχήμα 2. Κατά συνέπεια, οδηγώντας κάθε χρονική στιγμή μόνο μία μπάρα το συνολικό ρεύμα δε θα είναι μεγαλύτερο από 400mA. Φυσικά, ο ρυθμός αλλαγής πλαισίων(**frame rate**) πρέπει να είναι μεγαλύτερος από 50 πλαίσια/SEC προκειμένου να μη αναβοσβήνει η εικόνα



Σχήμα 2. To display

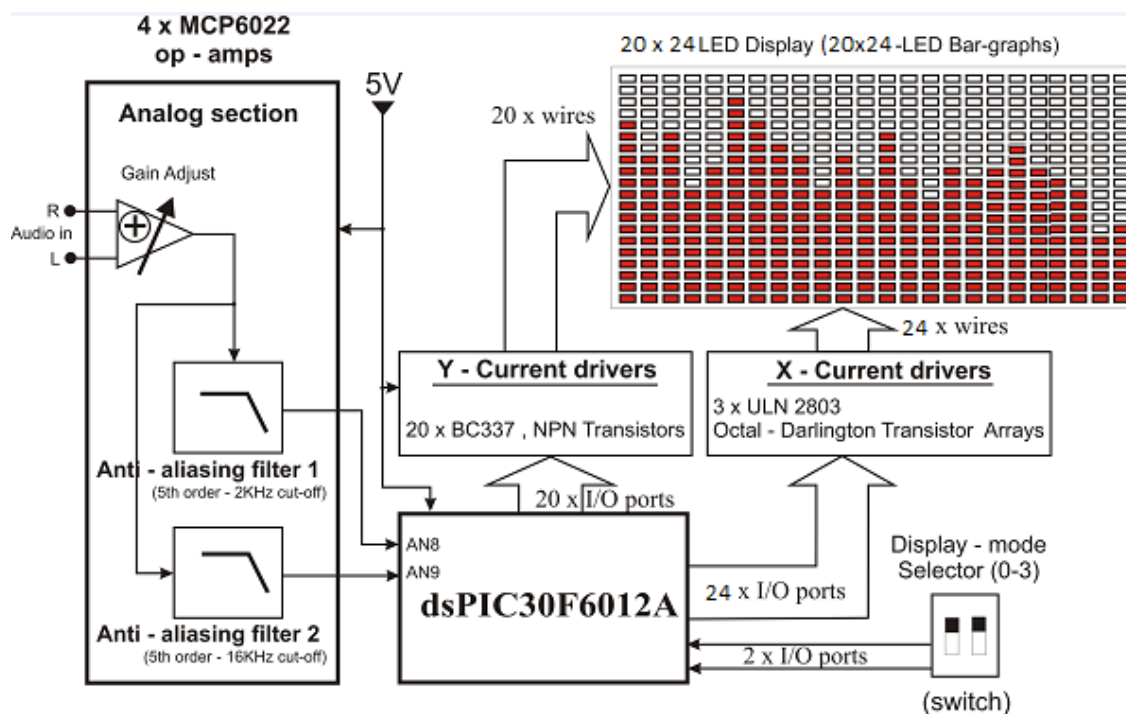
Ο dsPIC30F6012A μπορεί να κάνει ταυτόχρονα τόσο την ψηφιακή επεξεργασία σήματος (DSP) όσο και την απεικόνιση σε πραγματικό χρόνο, αλλά δεν μπορεί να οδηγήσει τα LEDs με το ρεύμα που απαιτείται. Γι αυτό χρειάζονται μερικά κυκλώματα οδήγησης.

3. Το Hardware

Λαμβάνοντας υπόψη τα ζητήματα που έχουν αναφερθεί στην παράγραφο 2, μπορούμε να σχεδιάσουμε ένα διάγραμμα βαθμίδων του ψηφιακού αναλυτή φάσματος ακουστικών συχνοτήτων πραγματικού χρόνου. Αυτό το διάγραμμα παρουσιάζεται στο σχήμα 3.

Σε αυτό το διάγραμμα παρουσιάζεται ένας διπλός διακόπτης (display mode selector). Χρησιμοποιώντας το διακόπτη, ο χρήστης είναι σε θέση να επιλέξει μεταξύ των διαφορετικών τρόπων απεικόνισης. Τέσσερις (4) διαφορετικοί τρόποι απεικόνισης υποστηρίζονται σε αυτή τη συσκευή. Αυτοί οι τρόποι απεικόνισης είναι:

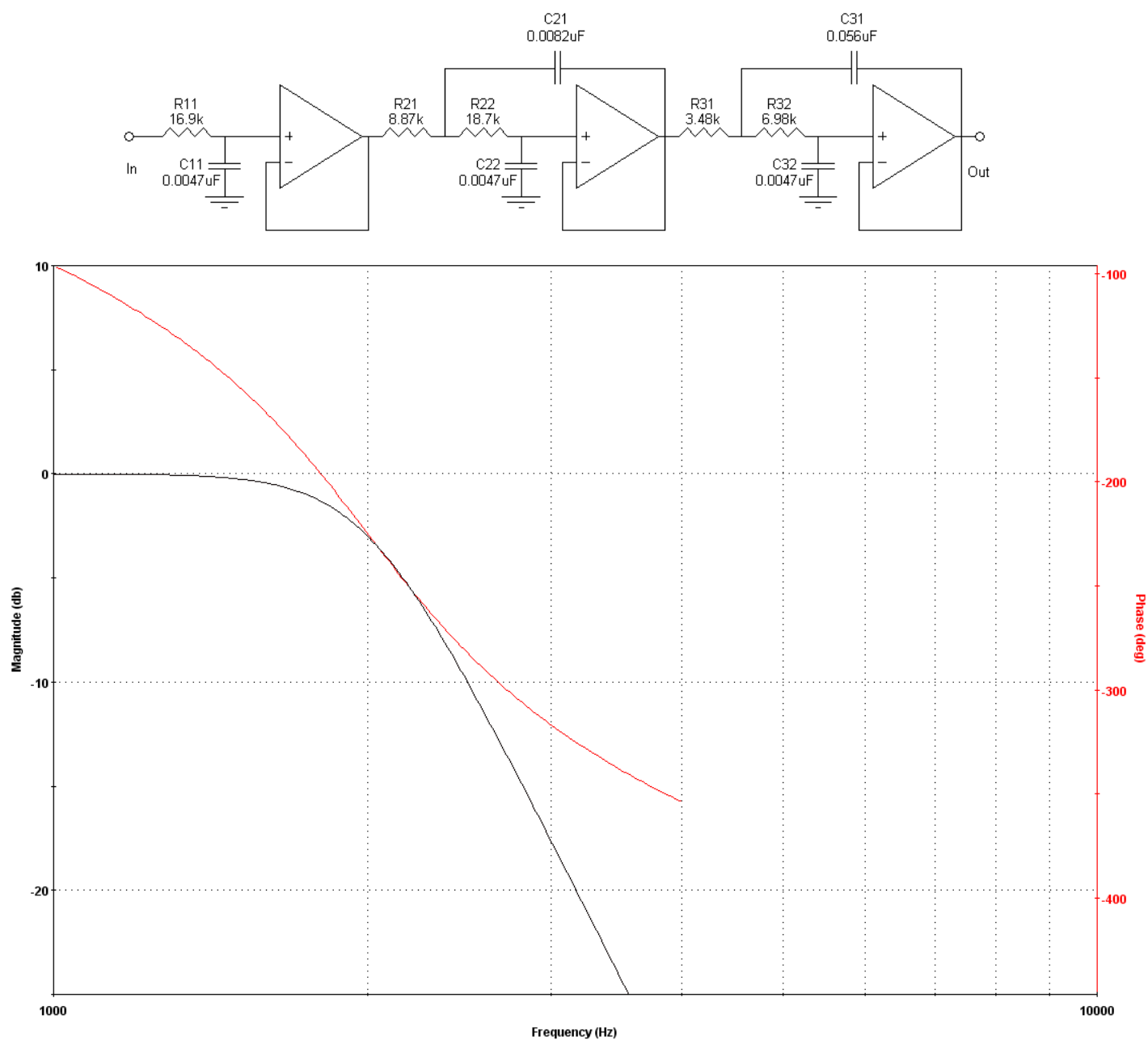
1. "Μπάρες" (είναι μια απεικόνιση πραγματικού χρόνου σε μπάρες)
2. "Μπάρες + κορυφές" (είναι μια απεικόνιση πραγματικού χρόνου με ταυτόχρονη συγκράτηση κορυφής)
3. "Αναρριχώμενες μπάρες" (είναι μια "αργή" απεικόνιση με μπάρες. Οι μπάρες "αναρριχούνται" ως την κορυφή. Μετά από αυτό, πέφτουν με ένα σταθερό ρυθμό έως ότου εμφανίζεται μια ισχυρότερη κορυφή)
4. "Αναρριχώμενες μπάρες + βροχή" (είναι μια "αργή" απεικόνιση με μπάρες που δίνει την αίσθηση "της βροχής". Οι μπάρες "αναρριχούνται" ως την κορυφή και μένουν εκεί. Μετά από αυτό, οι μπάρες πέφτουν με ένα σταθερό ρυθμό A και οι κορυφές πέφτουν με ένα σταθερό ρυθμό B έως ότου εμφανίζεται μια ισχυρότερη κορυφή. Θα πρέπει $A > B$)



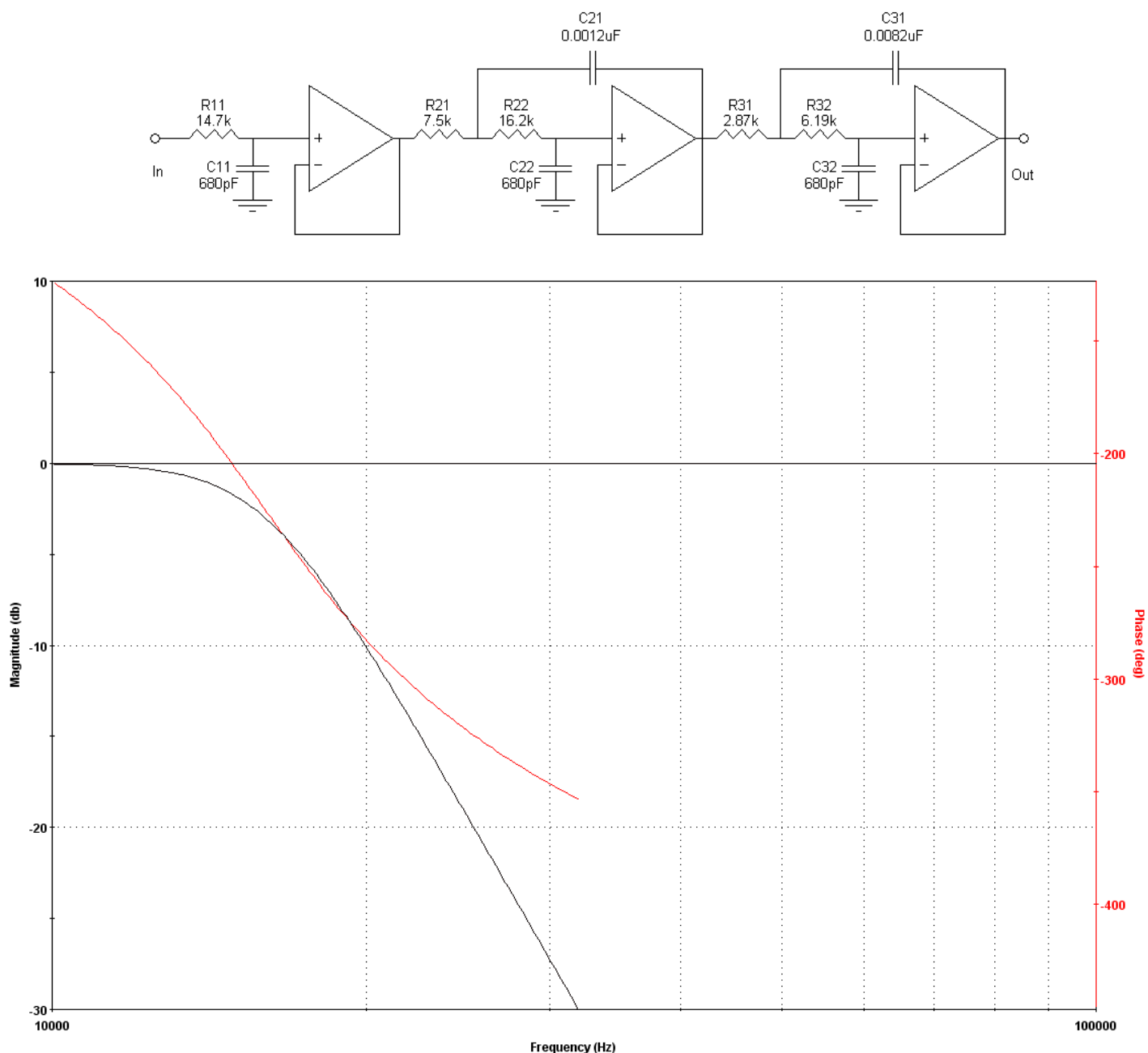
Σχήμα 3. Γενικό διάγραμμα βαθμίδων του συστήματος.

3.1 Τα φίλτρα αντιαναδίπλωσης (anti-aliasing filters)

Τα φίλτρα αντιαναδίπλωσης έχουν σχεδιαστεί χρησιμοποιώντας το λογισμικό Microchip FilterLab. Έχουν σχεδιαστεί για τη συχνότητα αποκοπής 2KHz και 16KHz αντίστοιχα, και για λόγο σήματος/θόρυβο καλύτερη από 30db. Λόγω της περιορισμένης δυναμικής περιοχής του αναλυτή φάσματος (που είναι περίπου 30db) οποιοσδήποτε θόρυβος κάτω από αυτό το κατώτατο όριο (-30db) μπορεί να αγνοηθεί. Με βάση αυτό και λόγω της τεχνικής της υπερ-δειγματοληψίας (oversampling) που χρησιμοποιούμε, η τάξη των απαιτούμενων anti-aliasing φίλτρων μπορεί να περιοριστεί στο 5.



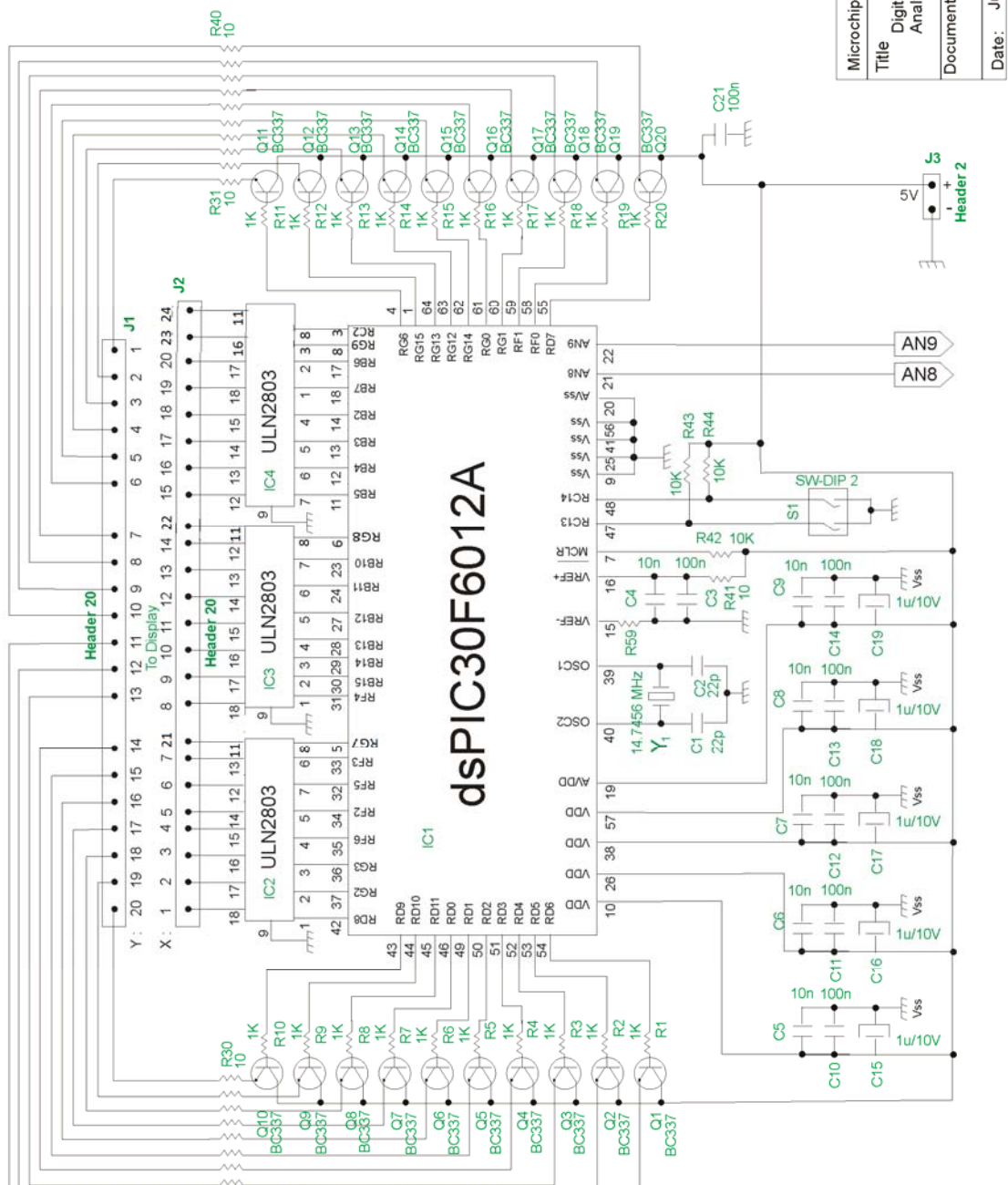
Σχήμα 4. Το 1^ο anti-aliasing filter (στιγμιότυπα από Microchip's FilterLab)



Σχήμα 5. Το 2ο anti-aliasing filter (στιγμιότυπα από Microchip’s FilterLab)

3.2 Σχηματικά διαγράμματα

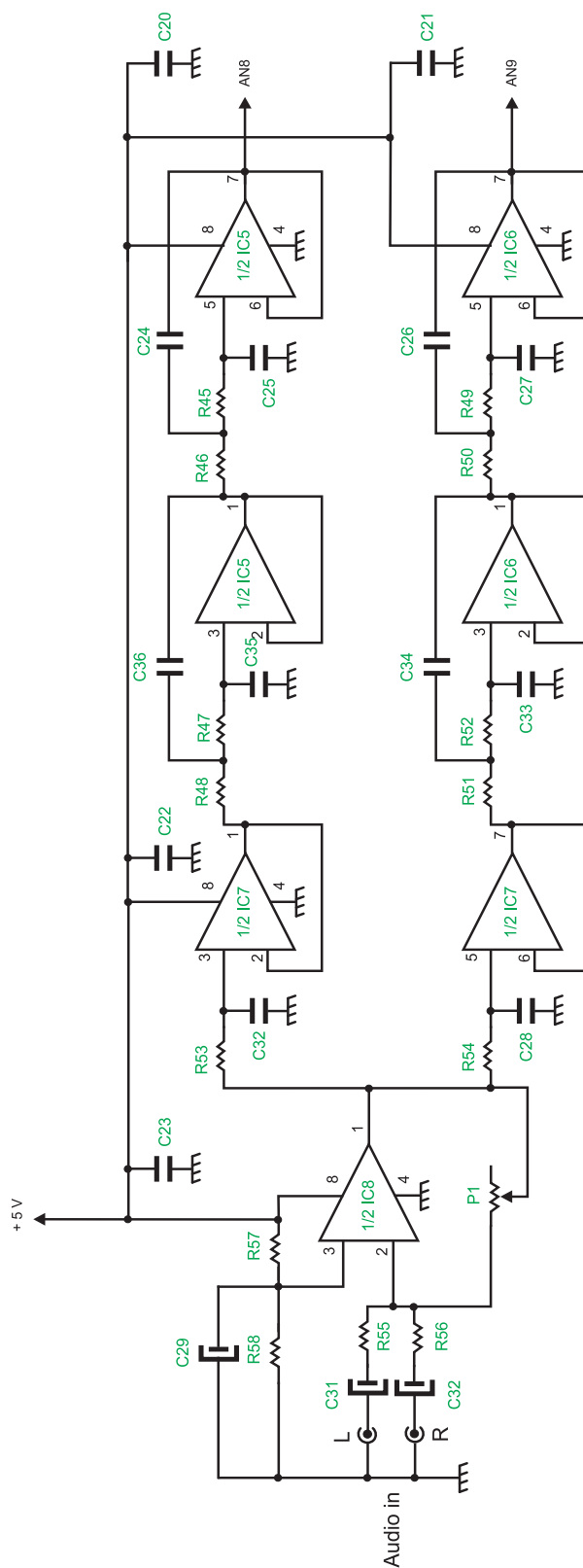
Το hardware αποτελείται από 2 πλακέτες, την κύρια πλακέτα και την πλακέτα του display. Όλες οι βαθμίδες, εκτός του display, βρίσκονται στην κύρια πλακέτα. Τα πλήρη σχηματικά διαγράμματα του ψηφιακού αναλυτή φάσματος πραγματικού χρόνου, παρουσιάζονται παρακάτω:



| | | |
|--|--|--------------|
| Microchip 16-bit Embedded Control Design Contest | | |
| Title | Digital, real time Audio Frequency Spectrum Analyzer - Digital Section | |
| Document Number | Entry # MT1750 | REV 2 |
| Date: | June 11, 2007 | Sheet 1 of 3 |

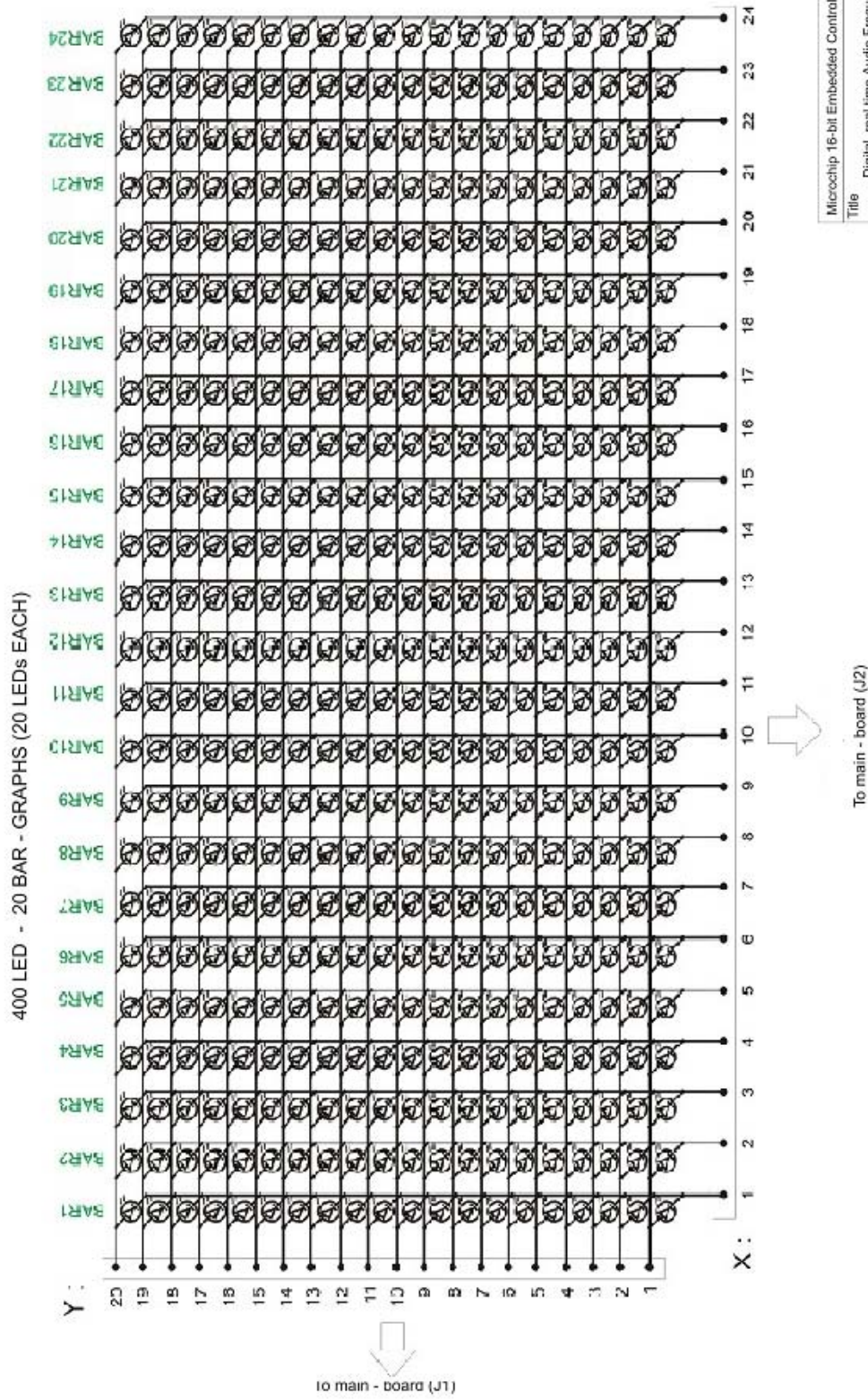
Σχήμα 6. Η κύρια πλακέτα (main board)

| | | | |
|--|----------------|--|---|
| Microchip 16-bit Embedded Control Design Contest | | | |
| Title | | Digital, real time Audio Frequency Spectrum Analyzer for audio devices - Analog Section | |
| Document Number | Entry # MT1750 | REV | 1 |
| Date: June 11, 2007 | Sheet | 2 of | 3 |



Analog section

Σχήμα 7. Οι αναλογικές βαθμίδες.



Σχήμα 8. To display

| | | | |
|--|--|-------|--------|
| Microchip 16-bit Embedded Control Design Contest | | | |
| Title | Digital, real time Audio Frequency Spectrum Analyzer for audio devices - Display | | |
| Document Number | Entry # MT1750 | REV | 1 |
| Date: | June 11, 2007 | Sheet | 3 of 3 |

3.3 Εξαρτήματα.

Microchip 16-bit Embedded Control Design Contest Entry # **MT1750**

Ψηφιακός αναλυτής φάσματος ακουστικών συχνοτήτων πραγματικού χρόνου.

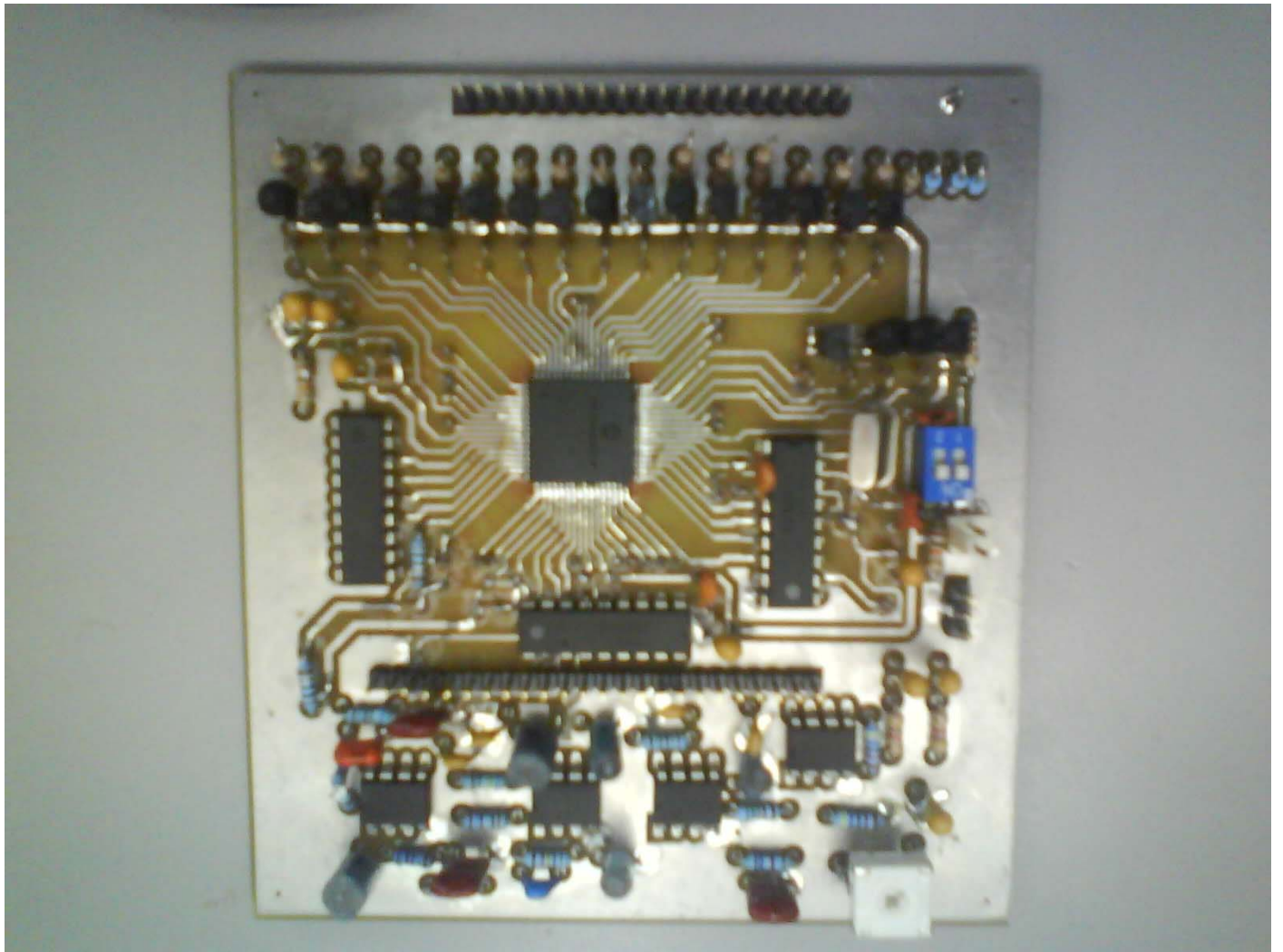
Κατάλογος υλικών

| Qty | Description | Ref Des |
|-----|---|------------------------|
| 1 | Microchip dsPIC30F6012A | IC1 |
| 3 | ULN2803 – διάταξης Darlington transistor | IC2, IC3, IC4 |
| 4 | Microchip MCP6022 διπλός τελεστικός ενισχυτής | IC5, IC6, IC7, IC8 |
| 24 | BC337 NPN transistor | Q1, Q2, Q3, ... Q20 |
| 24 | 20 – LED Bar – graph * | BAR1, BAR2, ...BAR24 |
| 1 | 14.74560 MHz Crystal | Y1 |
| 24 | 1K αντίσταση επιφανειακής στήριξης | R1, R2, R3 ... R20 |
| 24 | 0 – 180 Ohm ** 1/2W αντίσταση | R21, R22 ... R40 |
| 3 | 10K 5%, 1/4W αντίσταση | R42, R43, R44 |
| 1 | 6.98K 1%, 1/4W αντίσταση | R45 |
| 1 | 3.48K 1%, 1/4W αντίσταση | R46 |
| 1 | 18.7K 1%, 1/4W αντίσταση | R47 |
| 1 | 8.87K 1%, 1/4W αντίσταση | R48 |
| 1 | 6.49K 1%, 1/4W αντίσταση | R49 |
| 1 | 3.09K 1%, 1/4W αντίσταση | R50 |
| 1 | 7.87K 1%, 1/4W αντίσταση | R51 |
| 1 | 17.4K 1%, 1/4W αντίσταση | R52 |
| 1 | 16.9K 1%, 1/4W αντίσταση | R53 |
| 1 | 15.8K 1%, 1/4W αντίσταση | R54 |
| 2 | 4.7K 5%, 1/4W αντίσταση | R55, R56 |
| 1 | 47K 5%, 1/4W αντίσταση | R57, R58 |
| 2 | 10 Ohm 5%, 1/4W αντίσταση | R41, R59 |
| 1 | 100K trimmer | P1 |
| 2 | 22pF κεραμικός πυκνωτής | C1, C2 |
| 6 | 10nF πυκνωτής επιφανειακής στήριξης | C4, C5, C6, C7, C8, C9 |
| 6 | 100nF πυκνωτής επιφανειακής στήριξης | C3, C10, C11 ... C14 |
| 4 | 100nF multi-layer πυκνωτής | C20, C21, C22, C23 |
| 1 | 56nF πυκνωτής μίκας | C24 |
| 3 | 4.7nF πυκνωτής μίκας | C25, C32, C35 |
| 2 | 8.2nF πυκνωτής πολυστρωματικός | C26, C36 |
| 3 | 680pF κεραμικός πυκνωτής | C27, C28, C33 |
| 6 | 1uF / 10V ηλεκτρολυτικός πυκνωτής | C29, C15, C16 ... C19 |
| 2 | 10uF / 10V ηλεκτρολυτικός πυκνωτής | C30, C31 |
| 1 | 1.2nF πυκνωτής πολλαπλών στρωμάτων | C34 |
| 1 | Διπλός διακόπτης | S1 |
| 4 | 20-pin σύνδεσμος | J1, J2, J4, J5 |
| 1 | 2-pin σύνδεσμος | J3 |

Σημείωση: οι πυκνωτές είναι ανοχής 5%, εκτός αν διευκρινίζονται.

* οποιοσδήποτε τύπος LED μπορεί να χρησιμοποιηθεί

** οι τιμές των αντιστάσεων R21-40 εξαρτώνται από τον τύπο των LED που χρησιμοποιούνται.(χρησιμοποιούμε 0 ohm για το συγκεκριμένο τύπο led που επιλέξαμε. Για άλλους τύπους led οι τυπικές τιμές είναι από 22-240 Ohm για φωτεινότητα ίδιας εντάσης 18mA LEDs)

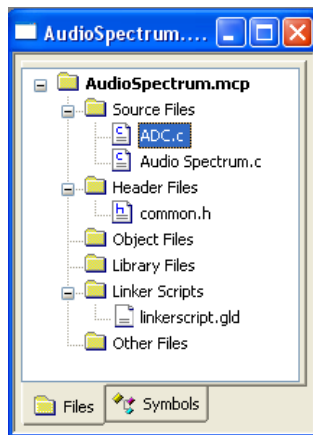


Φωτογραφία 1 . Η κύρια βαθμίδα.

Για τη δημιουργία του τυπομένου της πλακέτας χρησιμοποιήσαμε το πρόγραμμα :
Altium Designer. (παράρτημα4)

4. Το λογισμικό (Software)

Το software έχει αναπτυχθεί σε γλώσσα C, χρησιμοποιώντας Microchip C30 Compiler και κάποιες έτοιμες ρουτίνες που βρίσκονται στη βιβλιοθήκη DSP. Ο κώδικας βρίσκεται σε 3 αρχεία:



1. common.h → Το αρχείο περιέχει κάποιους γενικούς ορισμούς

Παράρτημα 1.

2. ADC.c → Το αρχείο περιέχει 2 συναρτήσεις, την - ADC_Init() και _ADCInterrupt()

Παράρτημα2.

```
void ADC_Init(char mode);
```

```
void __attribute__((__interrupt__)) _ADCInterrupt(void);
```

ADC_Init() χρησιμοποιείται για να ρυθμίσει τον A/D converter έτσι ώστε για κάθε 16 δείγματα του σήματος ήχου να προκαλεί μια διακοπή. Ο A/D converter ρυθμίζεται για δειγματοληψία 7998.6981 Hz (mode=1) και στα 87252.071 Hz (mode=2). Ο Timer3 παρέχει χρονισμό στον A/D converter. Το σήμα ήχου λαμβάνεται από τα pins AN8 (για mode=1) και AN9 (για mode=2).

_ADCInterrupt() ρουτίνα διακοπών στον A/D (ISR).

3. Audio Spectrum.c → Αυτό το αρχείο περιέχει 15 συναρτήσεις:

Παράρτημα3.

```
int main(void);
```

Αυτή είναι η κύρια ρουτίνα.

```
void ConvertToComplex(void);
```

Αυτή η συνάρτηση μετατρέπει τα πραγματικά ψηφιακά δεδομένα της εισόδου σε μιγαδικά. Σε ένα μιγαδικό πίνακα, στο πραγματικό μέρος τοποθετούνται τα πραγματικά δεδομένα από το σήμα εισόδου και το φανταστικό μέρος μηδενίζεται.

```
void Filter_Bank(unsigned int Samplesflag);
```

Αυτή η συνάρτηση υπολογίζει τους συντελεστές Bar0, Bar1, ...Bar23 σύμφωνα με τις εξισώσεις (2.11) και (2.34). Όταν η μεταβλητή SamplesFlag=1 ή 3 τότε η συχνότητα δειγματοληψίας είναι 7998.6981Hz ή 87252.071 Hz αντίστοιχα.

```
void FindPeakHold(void);
```

Αυτή η συνάρτηση χρησιμοποιείται κατά τη διάρκεια του τρόπου απεικόνισης «Μπάρες και Κορυφές»

`void FindSnapToPeak(void);`

Αυτή η συνάρτηση χρησιμοποιείται κατά τη διάρκεια του τρόπου απεικόνισης «Αναρριχόμενες μπάρες»

`void initializeIO(void);`

Αυτή η συνάρτηση είναι η ρουτίνα αρχικοποίησης των IO pins.

`void initialize_rest(void);`

Αυτή η ρουτίνα αρχικοποιεί τις μεταβλητές του προγράμματος.

`void FindClimbingBars_Rain(void);`

Αυτή η συνάρτηση χρησιμοποιείται κατά τη διάρκεια του τρόπου απεικόνισης «Αναρριχόμενες μπάρες + Βροχή»

`unsigned int display_mode(void);`

Αυτή η ρουτίνα επιστρέφει τον τρόπο απεικόνισης(0-3) σύμφωνα με το διακόπτη s1. Υπάρχουν 4 τρόποι απεικόνισης: 0-Μπάρες, 1- Αναρριχόμενες μπάρες + Βροχή, 2- Μπάρες και Κορυφές, 3- Αναρριχόμενες μπάρες.

`void display (unsigned int dis_mode);`

Αυτό είναι η ρουτίνα απεικόνισης. Απεικονίζει σε πραγματικό χρόνο σε ένα 20x24 LED display τους συντελεστές Bar. Υποστηρίζει 4 τρόπους απεικόνισης.

`void Set_Y (unsigned int x);`

Η ρουτίνα Sets Y (figure 2) ρυθμίζει το πόσα led θα ανάβουν σε κάθε μπάρα. Χρησιμοποιείται κατά τη διάρκεια του τρόπου απεικόνισης «Μπάρες»

`void Set_Y_Peak (unsigned int z);`

Η ρουτίνα Sets Y (figure 2) ρυθμίζει το πόσα led θα ανάβουν σε κάθε μπάρα. Χρησιμοποιείται κατά τη διάρκεια του τρόπου απεικόνισης «Μπάρες + Κορυφές»

`void Set_Y_SnapTP (unsigned int n);`

Η ρουτίνα Sets Y (figure 2) ρυθμίζει το πόσα led θα ανάβουν στη n-ωστή μπάρα. Χρησιμοποιείται κατά τη διάρκεια του τρόπου απεικόνισης «Αναρριχόμενες μπάρες»

`void Set_Y_Rain(unsigned int j);`

Η ρουτίνα Sets Y (figure 2) ρυθμίζει το πόσα led θα ανάβουν στη j - μπάρα. Χρησιμοποιείται κατά τη διάρκεια του τρόπου απεικόνισης «Αναρριχόμενες μπάρες + Βροχή»

`fractional add_Elements(int num, fractional* point);`

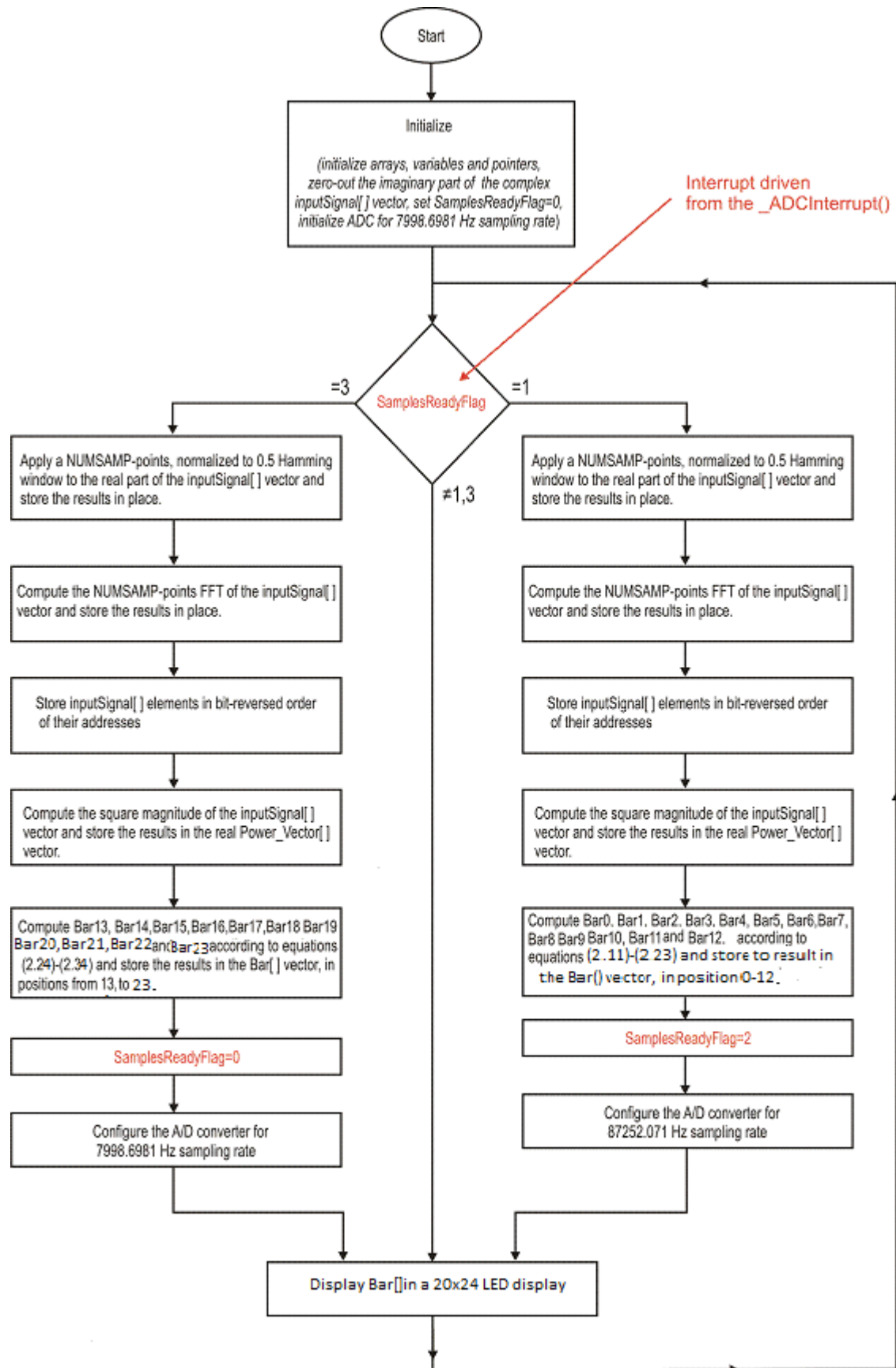
Επιστρέφει το άθροισμα num διαδοχικών στοιχείων ενός fractional διανύσματος (point : δείκτης. Δείχνει το πρώτο στοιχείο του fractional διανύσματος).

Οι σημαντικότερες ρουτίνες αυτού του προγράμματος παρουσιάζονται λεπτομερώς, παρακάτω:

4.1 Η κύρια ρουτίνα

Η κύρια ρουτίνα είναι μια μηχανή καταστάσεων οδηγούμενη από μια ρουτίνα διακοπής. Η κύρια ρουτίνα οδηγείται από τη ρουτίνα διακοπής του μετατροπέα A/D. Ο 12 bit A/D μετατροπέας λαμβάνει 256 δείγματα από το pin AN8 ή από το pin AN9 με ρυθμό δειγματοληψίας 7998.6981 Hz και 87252.071 Hz αντίστοιχα και τοποθετεί τα αποτελέσματα της ψηφιοποίησης στο πραγματικό μέρος του πίνακα `inputSignal[]`. Το μιγαδικό διάνυσμα `inputSignal[]` υποβάλλεται σε επεξεργασία που περιλαμβάνει: πολλαπλασιασμό με παράθυρο Hamming, γρήγορο μετασχηματισμό Φουριέ και υπολογισμό ισχύος. Μετά τον υπολογισμό της φασματικής ισχύος του σήματος εισόδου, το ψηφιακό σήμα που την αντιπροσωπεύει, αποθηκεύεται στο διάνυσμα `Power_Vector[]`. Από το διάνυσμα `Power_Vector[]` υπολογίζονται οι συντελεστές `Bar[]`, σύμφωνα με τις εξισώσεις (2.11) ως (2.34). Ο υπολογισμός αυτός γίνεται κατά τη διάρκεια της εκτέλεσης της ρουτίνας `Filter_Bank`. Κάθε συντελεστής `Bar[]` απεικονίζεται σε μια μπάρα led στο display.

Το διάγραμμα ροής της κύριας ρουτίνας παρουσιάζεται παρακάτω:



Διάγραμμα ροής 1. Η κύρια ρουτίνα

```

int main (void)
{
    // Initialization
    initializeIO();
    initialize_rest();

    while (1)          //Loop Endlessly
    {                  //from this point on. (interrupt driven state-machine)

        if (SamplesReadyFlag==1) // NUMSAMP samples have been acquired (sampling rate:7998.6981 Hz)
        {

            /*Apply HammingFactors to the input sample array and store the resulting windowed vector in place*/
            VectorMultiply (NUMSAMP*2,&inputSignal[0].real,&inputSignal[0].real,&HammingFactors[0]);

            /* Perform FFT operation */
            FFTComplexIP (LOG2_BLOCK_LENGTH, &inputSignal[0], &twiddleFactors[0], COEFFS_IN_DATA);

            /* Store output samples in bit-reversed order of their addresses */
            BitReverseComplex (LOG2_BLOCK_LENGTH, &inputSignal[0]);

            /* Compute the square magnitude of the complex FFT output array so we have a Real output vector */
            SquareMagnitudeCplx(NUMSAMP, &inputSignal[0], &Power_Vector[0]);

            Filter_Bank(SamplesReadyFlag);    // Find the Bar[0-9] samples which will be displayed in a 20-bars led display

            p_inputSignal = &inputSignal[0]; //Initialize pointer for the A/D conversion
            SamplesReadyFlag = 2;             //Initialize A/D converter's Samples-Ready flag for 87252.071Hz sampling rate
            ADC_Init(2);                      //Initialize the A/D converter for 87252.071 Hz sampling rate

        }

        if (SamplesReadyFlag==3) // NUMSAMP samples have been acquired (sampling rate:87252.071 Hz)
        {

            /*Apply HammingFactors to the input sample array and store the resulting windowed vector in place*/
            VectorMultiply (NUMSAMP*2,&inputSignal[0].real,&inputSignal[0].real,&HammingFactors[0]);

            /* Perform FFT operation */
            FFTComplexIP (LOG2_BLOCK_LENGTH, &inputSignal[0], &twiddleFactors[0], COEFFS_IN_DATA);

            /* Store output samples in bit-reversed order of their addresses */
            BitReverseComplex (LOG2_BLOCK_LENGTH, &inputSignal[0]);

            /* Compute the square magnitude of the complex FFT output array so we have a Real output vector */
            SquareMagnitudeCplx(NUMSAMP, &inputSignal[0], &Power_Vector[0]);

            Filter_Bank(SamplesReadyFlag);    // Find the Bar[10-19] samples which will be displayed in a 20-bars led display

            p_inputSignal = &inputSignal[0]; //Initialize pointer for the A/D conversion
            SamplesReadyFlag = 0;             //Initialize A/D converter's Samples-Ready flag for 7998.6981 Hz sampling rate
            ADC_Init(1);                      //Initialize the A/D converter for 7998.6981 Hz sampling rate

        }

        display (display_mode());             // Display routine (for display_mode 0, 1, 2, 3)

    }
    return 0;
}

```

Δείγμα κώδικα 1. Η κύρια ρουτίνα

ΣΗΜΕΙΩΣΗ 1: Η μεταβλητή `inputSignal []` είναι ένας πίνακας που αποτελείται από `NUMSAMP*2` στοιχεία. Ο πίνακας αυτός αντιπροσωπεύει ένα μιγαδικό fractional διάνυσμα (`fractcomplex`). Το πραγματικό μέρος αυτού του διανύσματος περιέχει `NUMSAMP` δείγματα του ακουστικού σήματος εισόδου. Το φανταστικό μέρος αυτού του διανύσματος αρχικά τίθεται στο μηδέν (when `initialize_rest()` runs).

Κατά τη διάρκεια της δειγματοληψίας το πρόγραμμα λαμβάνει τα 256 δείγματα εισόδου και τα τοποθετεί τόσο στο πραγματικό όσο και στο φανταστικό μέρος του διανύσματος `inputSignal []`. Αυτό γίνεται πολλαπλασιάζοντας την `inputSignal []` με ένα παράθυρο

Hamming. Όμως επειδή δε μπορούμε να πολλαπλασιάσουμε ένα πραγματικό διάνυσμα με ένα μιγαδικό διάνυσμα άμεσα, χρησιμοποιούμε ένα τροποποιημένο παράθυρο Hamming με μήκος NUMPSAMP*2 (512 θέσεις). Έτσι πετυχαίνουμε τον πολλαπλασιασμό αυτό μέσω του software.

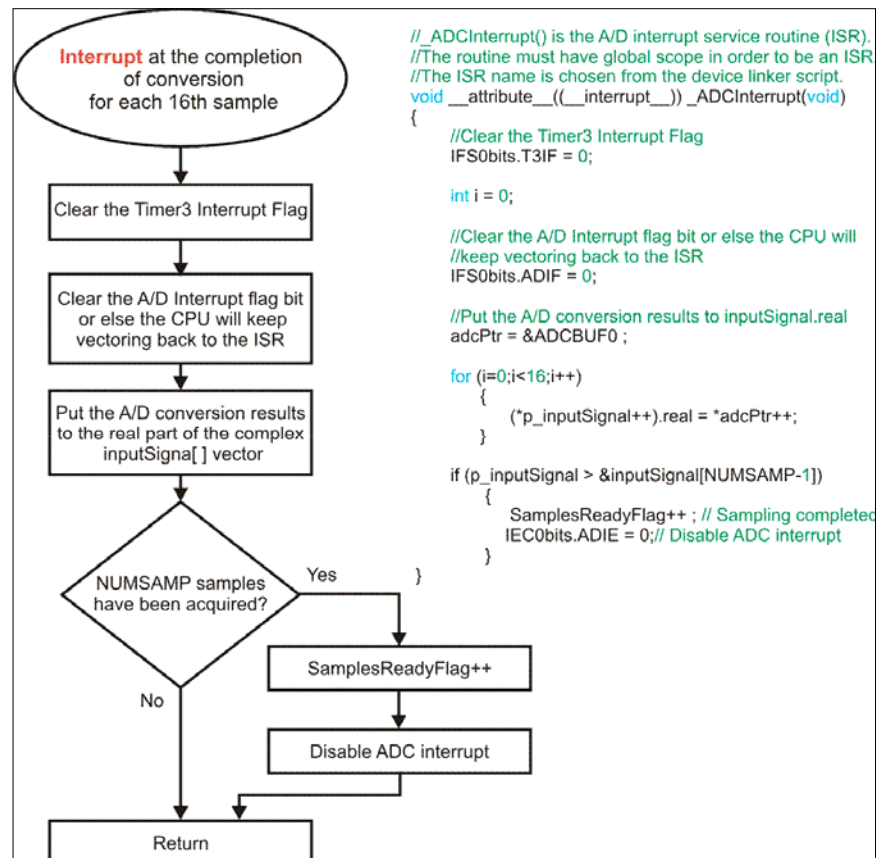
ΣΗΜΕΙΩΣΗ 2: Η ρουτίνα FFTComplexIP απαιτεί η κάθε τιμή που βρίσκεται στα ορίσματά της, να είναι ανάμεσα στο [-0.5, 0.5]. Γι αυτό οι συντελεστές του παραθύρου Hamming, που βρίσκονται στον πίνακα HammingFactors[], είναι κανονικοποιημένοι στο 0,5 κατά την έναρξη της εκτέλεσης του κώδικα.

ΣΗΜΕΙΩΣΗ 3: Τα αποτελέσματα από την FFTComplexIP αποθηκεύονται σε αντίστροφη διάταξη bit. Λόγω αυτού, εκτελούμε τη ρουτίνα BitReverseComplex για να επαναφέρουμε την κανονική διάταξη bit .

ΣΗΜΕΙΩΣΗ 4: Οι μεταβλητές SamplesReadyFlag όπως και η μεταβλητή/ δείκτης p_inputSignal χρησιμοποιούνται ταυτόχρονα και από την κύρια ρουτίνα και από τη ρουτίνα διακοπής.

4.2 Η ρουτίνα διακοπής του A/D (ISR)

Ο Timer3 χρησιμοποιείται για να παρέχει χρονισμό για το 12bit ADC. Ο 12-bit ADC προκαλεί μια διακοπή μετά τη λήψη 16 δειγμάτων στην είσοδο. Ο ρυθμός δειγματοληψίας αλλάζει κάθε φορά που τρέχει η ρουτίνα ADC_Init(char mode).



Διάγραμμα ροής 2. Η ρουτίνα διακοπής του A/DC(ISR)

4.3 Η ρουτίνα ADC_Init

Η ρουτίνα `ADC_Init(char mode)` καλείται από την κύρια ρουτίνα του κώδικα. Η ρουτίνα αυτή αρχικοποιεί τον A/D μετατροπέα έτσι ώστε να παραλαμβάνει 16 δείγματα από το σήμα εισόδου και στη συνέχεια να προκαλεί μια διακοπή. Ο ADC δειγματοληπτεί το ακουστικό σήμα με συχνότητα δειγματοληψίας 7998.6981 Hz (mode=1) ή 87252.071 Hz (mode=2). Ο Timer3 χρησιμοποιείται για να παρέχει χρονισμό. Ο ADC δειγματοληπτεί τα pins AN8 (for mode=1) και AN9 (for mode=2).

```

void ADC_Init(char mode)
{
    //ADCON1 Register
    //Set up A/D for Automatic Sampling
    //Use Timer3 to provide sampling time
    //Set up A/D conversion results to be read in 1.15 fractional
    //number format.
    //All other bits to their default state
    ADCON1bits.FORM = 3;
    ADCON1bits.SSRC = 2;
    ADCON1bits.ASAM = 1;

    //ADCON2 Register
    ADCON2bits.VCFG = 3; //External Vref+, External Vref-
    ADCON2bits.SMPI = 15; //Set up A/D for interrupting after 16 samples get filled in the buffer

    //ADCON3 Register
    //We would like to set up a sampling rate of 7998.6981Hz or 87252.071 Hz
    //Total Conversion Time= 1/Sampling Rate = 125.02 microseconds or 11.461 microseconds
    //For fosc=117.968MHz, Tcy = 33.91 ns = Instruction Cycle Time
    //We will set up Sampling Time using Timer3 & Tad using ADCS<5:0> bits
    //All other bits to their default state
    //Let's set up ADCS arbitrarily to 38
    //So Tad = Tcy*(ADCS+1)/2 = 661.2 nanoseconds
    //So, the A/D converter will take 14*Tad periods to convert each sample
    ADCON3bits.ADCS = 40;

    //Next, we will to set up Timer 3 to time-out every X=125.02 microseconds (for mode 1) or X=11.461 microseconds (for mode 2)
    //As a result, the module will stop sampling and trigger a conversion
    //on every Timer3 time-out, i.e., 125.02 microseconds or 11.461 microseconds. At that time,
    //the conversion process starts and completes 14*Tad periods later.
    TMR3 = 0x0000;
    if (mode==1) PR3 = SAMPCOUNT1;
    if (mode==2) PR3 = SAMPCOUNT2;
    IFS0bits.T3IF = 0;
    IEC0bits.T3IE = 0;
    T3CONbits.TCS = 0;
    T3CONbits.TCKPS=0;

    //ADCHS Register
    //Set up A/D Channel Select Register to convert AN8 (mode=1) or AN9 (mode=2) on Mux A input
    if (mode==1) ADCHS = 0x0008;
    if (mode==2) ADCHS = 0x0009;

    //ADCSSL Register
    //Channel Scanning is disabled. All bits left to their default state
    ADCSSL = 0x0000;

    //ADPCFG Register
    //Set up channels AN8 and AN9 as analog inputs and configure rest as digital
    //Recall that we configured all A/D pins as digital when code execution
    //entered main() out of reset
    ADPCFG = 0xFFFF;
    ADPCFGbits.PCFG8 = 0;
    ADPCFGbits.PCFG9 = 0;

    //Clear the A/D interrupt flag bit
    IFS0bits.ADIF = 0;

    //Set the A/D interrupt enable bit
    IEC0bits.ADIE = 1;

    //Turn on the A/D converter
    //This is typically done after configuring other registers
    ADCON1bits.ADON = 1;

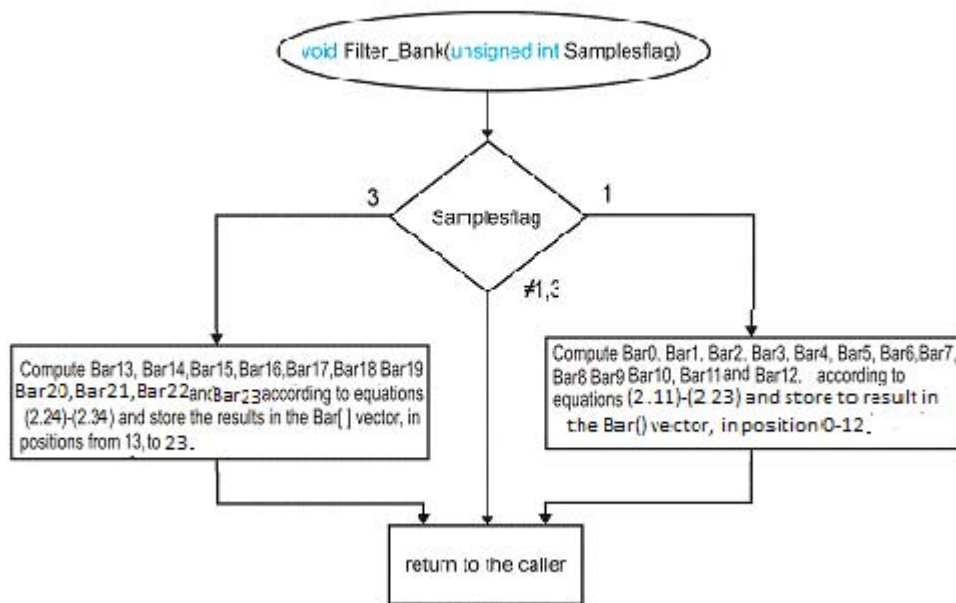
    //Start Timer 3
    T3CONbits.TON = 1;
}

```

Δείγμα κώδικα 2. Ο κώδικας της ρουτίνας ADC_Init

4.4 Η ρουτίνα Filter_Bank

Η void Filter_Bank(unsigned int Samplesflag) ρουτίνα καλείται από την κύρια ρουτίνα. Αυτή η ρουτίνα υπολογίζει τους συντελεστές Bar0, Bar1, ...Bar23 σύμφωνα με τις εξισώσεις (2.11) ως (2.30). Όταν το όρισμα SamplesFlag=1 ή 3 η συχνότητα δειγματοληψίας είναι 7998.6981Hz ή 87252.071 Hz αντίστοιχα.



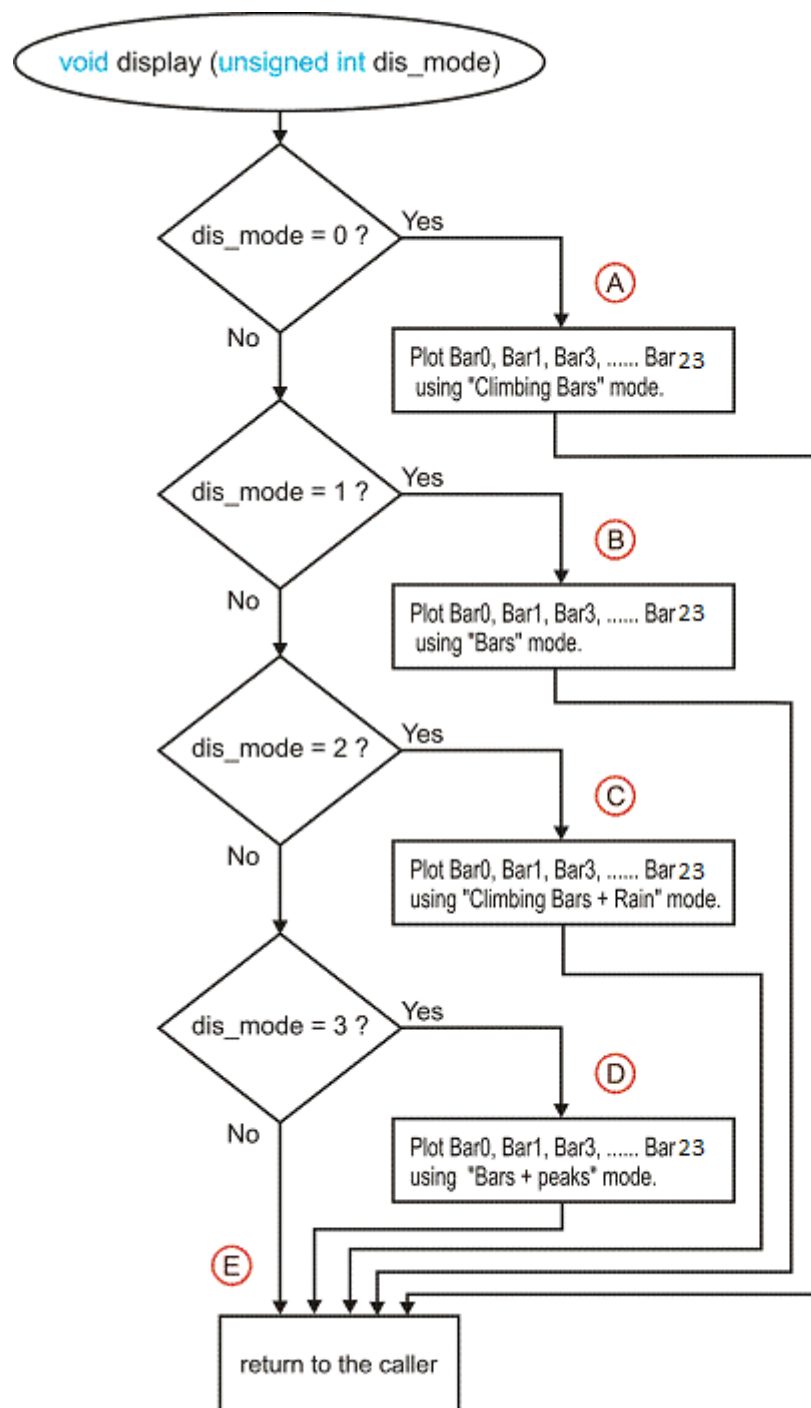
Διάγραμμα ροής 3. η ρουτίνα Filter_Bank

ΣΗΜΕΙΩΣΗ: Η ρουτίνα Filter_Bank υπολογίζει τους συντελεστές Bar0, Bar1, ...Bar23 σύμφωνα με τα δεδομένα που περιέχονται στο διάνυσμα Power_Vector, και επίσης χρησιμοποιείται και η ρουτίνα add_Elements.

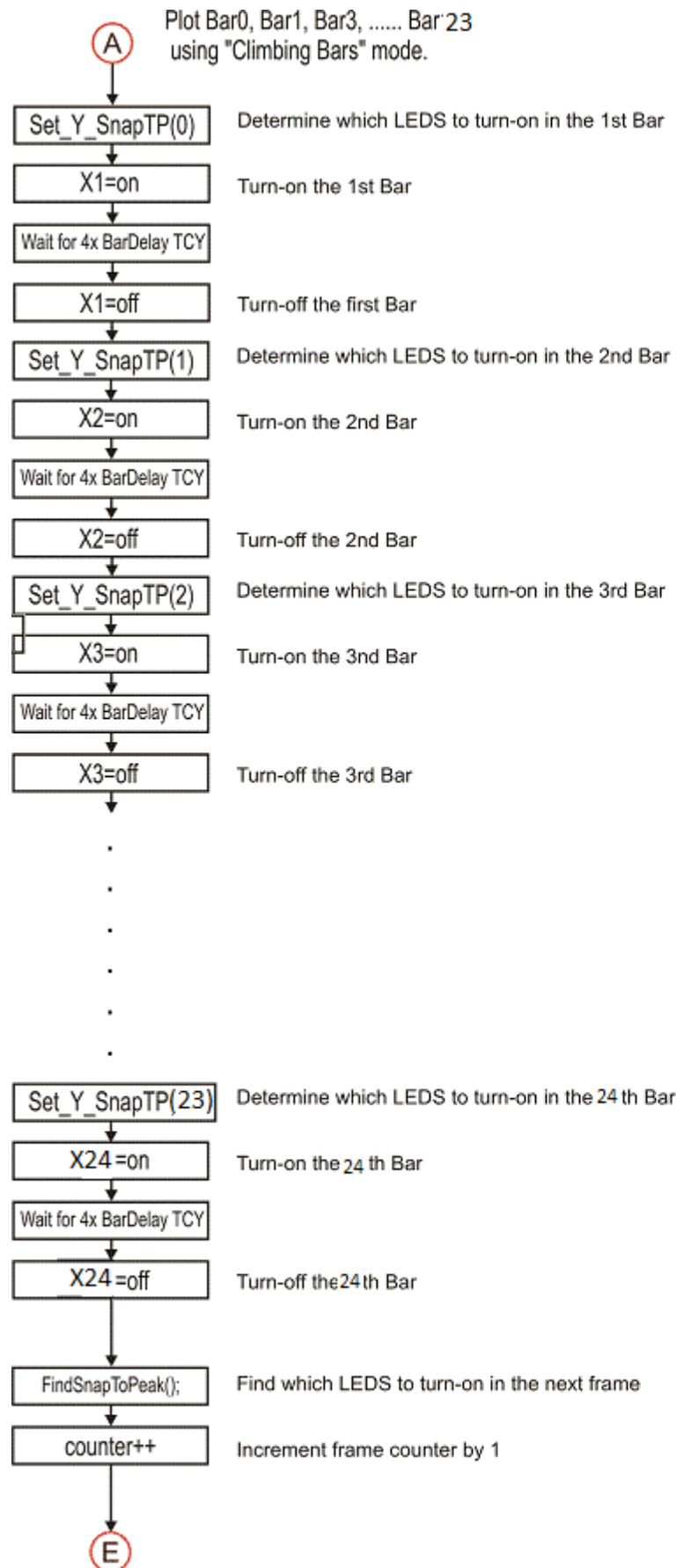
4.5 Η ρουτίνα display:

1. "Μπάρες" (είναι μια απεικόνιση πραγματικού χρόνου σε μπάρες)
2. "Μπάρες + κορυφές" (είναι μια απεικόνιση πραγματικού χρόνου με ταυτόχρονη συγκράτηση κορυφής)
3. "Αναρριχώμενες μπάρες" (είναι μια "αργή" απεικόνιση με μπάρες. Οι μπάρες "αναρριχούνται" ως την κορυφή. Μετά από αυτό, πέφτουν με ένα σταθερό ρυθμό έως ότου εμφανίζεται μια ισχυρότερη κορυφή)
4. "Αναρριχώμενες μπάρες + βροχή" (είναι μια "αργή" απεικόνιση με μπάρες που δίνει την αίσθηση "της βροχής". Οι μπάρες "αναρριχούνται" ως την κορυφή και μένουν εκεί. Μετά από αυτό, οι μπάρες πέφτουν με ένα σταθερό ρυθμό A και οι κορυφές πέφτουν με ένα σταθερό ρυθμό B έως ότου εμφανίζεται μια ισχυρότερη κορυφή. Θα πρέπει $A > B$)

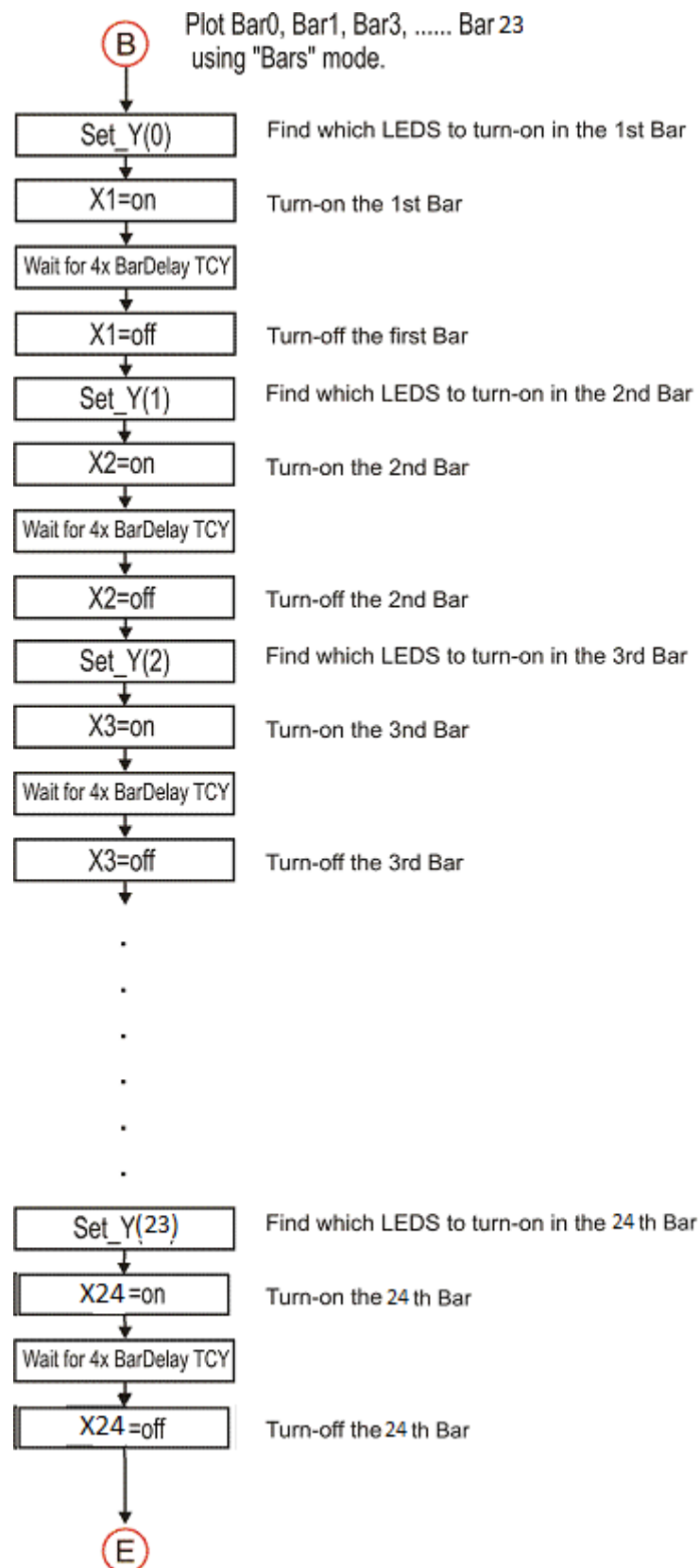
Η void display (unsigned int dis_mode) απεικονίζει τους συντελεστές Bar0....Bar23 σε ένα 20x24 LED display. Καλείται από την κύρια ρουτίνα. Το LED display αποτελείται από 24 LED bar-graphs. Κάθε ένα LED bar-graph αποτελείται από 20 LEDs. Πρώτα απεικονίζεται ο συντελεστής Bar0, έπειτα απεικονίζεται ο συντελεστής Bar1, ... και τελικά απεικονίζεται ο συντελεστής Bar23). Ένα πλαίσιο ολοκληρώνεται κάθε φορά που η display routine ολοκληρώνεται. Προκειμένου ο χρήστης να μην αντιλαμβάνεται ότι η απεικόνιση γίνεται σειριακά και όχι ταυτόχρονα, θα πρέπει να υπάρχουν περισσότερα από 50 πλαίσια εικόνας ανά δευτερόλεπτο (δηλ. η display routine καλείται περισσότερες από 50 φορές ανά δευτερόλεπτο). Η display routine υποστηρίζει τέσσερις τρόπους απεικόνισης:



Διάγραμμα ροής 4. Display routine

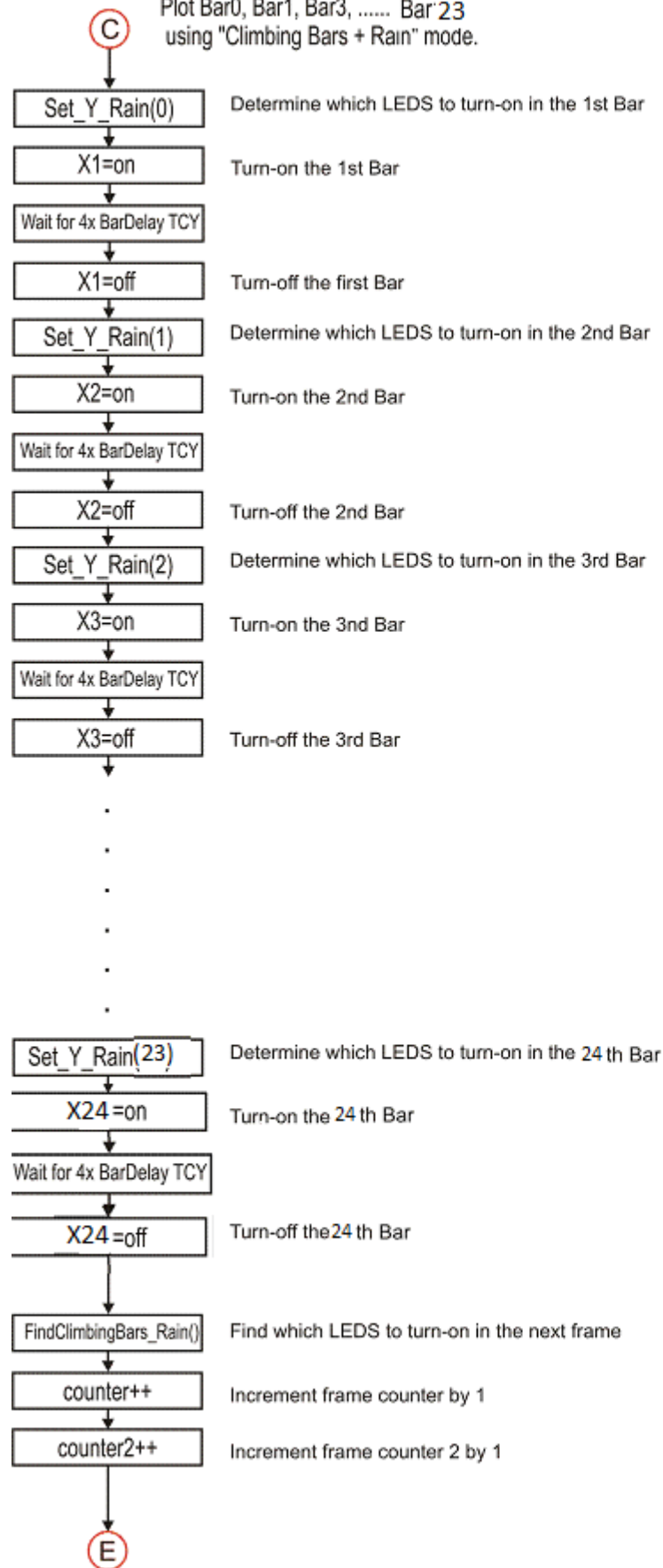


Διάγραμμα ροής 5. Ο "Climbing Bars" τρόπος επίδειξης (display routine)

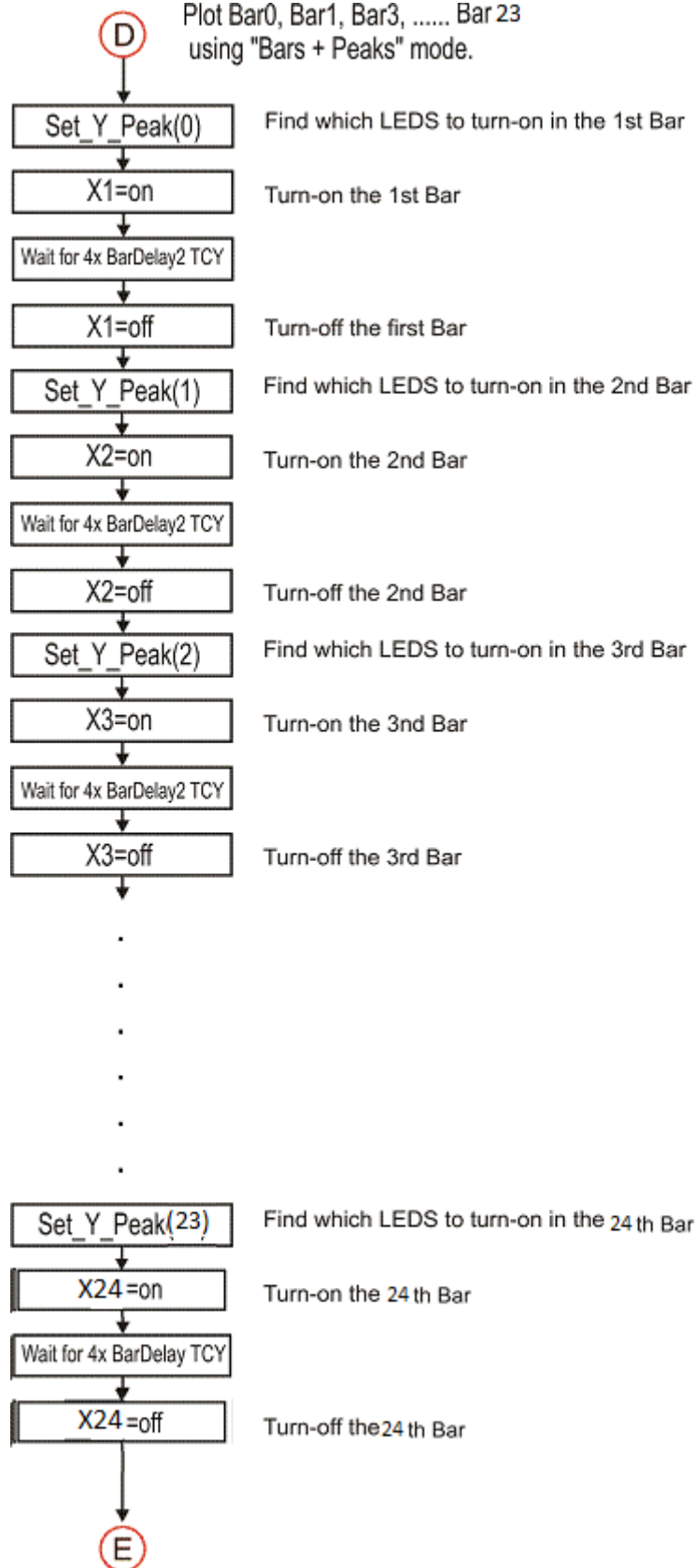


Διάγραμμα ροής 6. Ο “Bars” τρόπος επίδειξης (display routine)

Plot Bar0, Bar1, Bar3, Bar23
using “Climbing Bars + Rain” mode.



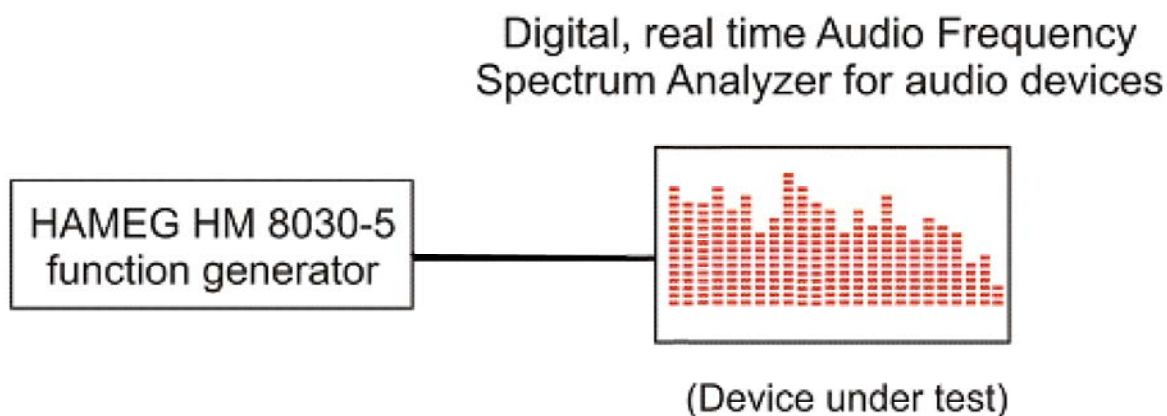
Διάγραμμα ροής 7. Ο “Climbing Bars + Rain” τρόπος επίδειξης (display routine)



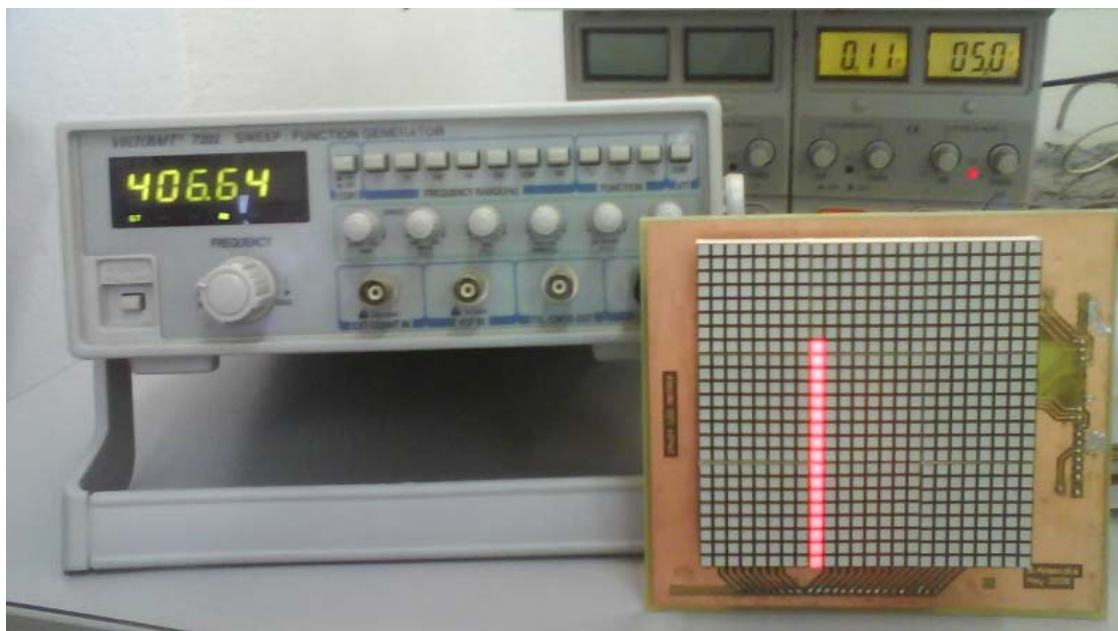
Διάγραμμα ροής 8. Ο “Bars + Peaks” τρόπος επίδειξης (display routine)

5. Η απόδοση του συστήματος μέτρησης.

Η απόδοση του συστήματος έχει μετρηθεί χρησιμοποιώντας τη γεννήτρια συναρτήσεων HAMEG HM 8030-5. Το display που χρησιμοποιήσαμε είναι 24x24 αντίθετα με την έξοδο της συσκευής μας που είναι 20x24, και έτσι στο display που χρησιμοποιήσαμε για τη δοκιμή της κύριας βαθμύδας τα LED στις 4 πάνω σειρές δεν ανάβουν ποτέ. Μερικές από τις μετρήσεις παρουσιάζονται παρακάτω:



Σχήμα 9. Διάταξη δοκιμών



Φωτογραφία 2. Στιγμιότυπο από τη διαδικασία μέτρησης

Η φωτογραφία 2 λήφθηκε κατά τη διάρκεια της δοκιμής της συσκευής. Ένας καθαρός ακουστικός τόνος 406.64 Hz. Μπορούμε να δούμε ότι η υπό δοκιμή συσκευή εμφανίζει ένδειξη στην 9^η bar graph (από αριστερά προς τα δεξιά), όπως περιμέναμε.



Φωτογραφία 3. Στιγμιότυπο από τη διαδικασία μέτρησης

Η φωτογραφία 3 λήφθηκε κατά τη διάρκεια της δοκιμής της συσκευής. Ένας καθαρός ακουστικός τόνος 513.78 Hz εφαρμόζεται στην είσοδο της υπό δοκιμή συσκευής. Μπορούμε να δούμε ότι η συσκευή στη δοκιμή εμφανίζει ένδειξη στη 10^1 μπάρα (από αριστερά προς τα δεξιά), όπως περιμέναμε.



Φωτογραφία 4. Στιγμιότυπο από τη διαδικασία μέτρησης

Στη φωτογραφία 4 ένας καθαρός ακουστικός τόνος 44082 Hz εφαρμόζεται στην είσοδο της υπό δοκιμή συσκευής. Η συσκευή υπό δοκιμή εμφανίζει ένδειξη στην 19^η μπάρα (από αριστερά προς τα δεξιά), όπως περιμέναμε.



Φωτογραφία 5. Στιγμιότυπο από τη διαδικασία μέτρησης

Στη φωτογραφία 5, ένας καθαρός ακουστικός τόνος 11876 Hz εφαρμόζεται στην είσοδο της συσκευής υπό δοκιμή. Η συσκευή υπό δοκιμή εμφανίζει ένδειξη στην 23^η μπάρα (από αριστερά προς τα δεξιά), όπως περιμέναμε.



Φωτογραφία 6. Στιγμιότυπο από τη διαδικασία μέτρησης

Στη φωτογραφία 6, ένας καθαρός ακουστικός τόνος 201.87 Hz εφαρμόζεται στην είσοδο της συσκευής υπό δοκιμή. Η συσκευή υπό δοκιμή δείχνει 0db στην 6^η μπάρα και -16.9db στην 5^η και 7^η μπάρα (από αριστερά προς τα δεξιά). Αυτό οφείλεται στο γεγονός ότι υπάρχει μια φτωχή διακριτική ικανότητα σε συχνότητες από 0 ως 200Hz λόγω της αλληλεπικάλυψης 1^{ου}, 2^{ου}, 3^{ου}, 4^{ου} και 5^{ου} φίλτρου.



Φωτογραφία 7. Στιγμιότυπο από τη διαδικασία μέτρησης

Στη φωτογραφία 7, ένας καθαρός ακουστικός τόνος 201.87 Hz εφαρμόζεται στην είσοδο της συσκευής υπό δοκιμή. Η συσκευή υπό δοκιμή δείχνει -3.9db στην 12^η μπάρα και την αρμονική της -19.5db στην 14^η μπάρα (από αριστερά προς τα δεξιά)

Παράρτημα .1

ΚΩΔΙΚΑΣ: common.h

Παράρτημα .2

ΚΩΔΙΚΑΣ: ADC.c

Παράρτημα .3

ΚΩΔΙΚΑΣ: Spectrum.c

Παράρτημα .4

Τυπομένο