

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ**  
**ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ**  
**ΤΜΗΜΑ ΜΗΧΑΝΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ**



**Ανάπτυξη Αλγορίθμου για τον Πειραματικό Έλεγχο  
της Δομικής Ακεραιότητας Κατασκευών**

**ΤΟΥ: ΚΩΝΣΤΑΝΤΙΝΟΥ ΝΙΚΟΛΟΠΟΥΛΟΥ**

**ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΚΩΝΣΤΑΝΤΙΝΟΣ  
ΠΑΠΑΔΗΜΗΤΡΙΟΥ**

**ΒΟΛΟΣ 2022**

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ**

**ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ**

**ΤΜΗΜΑ ΜΗΧΑΝΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ**



**Ανάπτυξη Αλγορίθμου για τον Πειραματικό Έλεγχο  
της Δομικής Ακεραιότητας Κατασκευών**

**ΤΟΥ: ΚΩΝΣΤΑΝΤΙΝΟΥ ΝΙΚΟΛΟΠΟΥΛΟΥ**

**ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΚΩΝΣΤΑΝΤΙΝΟΣ  
ΠΑΠΑΔΗΜΗΤΡΙΟΥ**

**ΒΟΛΟΣ 2022**

**Διπλωματική εργασία**

Υπεβλήθη για την εκπλήρωση μέρους των απαιτήσεων για την απόκτηση Διπλώματος  
Μηχανολόγου Μηχανικού

© 2022 Νικολόπουλος Κωνσταντίνος

Η έγκριση της διπλωματικής εργασίας από το τμήμα Μηχανολόγων Μηχανικών της  
Πολυτεχνικής Σχολής του Πανεπιστημίου Θεσσαλίας δεν Υποδηλώνει αποδοχή των  
απόψεων του συγγραφέα (Ν. 5343/32 αρ. 202 παρ. 2)

## Εγκρίθηκε από τα μέλη της Τριμελούς εξεταστικής επιτροπής

Πρώτος Εξεταστής Παπαδημητρίου Κωνσταντίνος  
(Επιβλέπων) Καθηγητής, Τμήμα Μηχανολόγων  
Μηχανικών, Πανεπιστήμιο Θεσσαλίας

Δεύτερος Εξεταστής Γεώργιος Χαραλάμπους  
Επίκ. Καθηγητής, Τμήμα Μηχανολόγων Μηχανικών,  
Πανεπιστήμιο Θεσσαλίας

Τρίτος Εξεταστής Τάσος Σταματέλλος  
Καθηγητής, Τμήμα Μηχανολόγων Μηχανικών, Πανεπιστήμιο  
Θεσσαλίας

## **Ευχαριστίες**

Θα ήθελα να ευχαριστήσω τον καθηγητή και επιβλέποντα μου κύριο Παπαδημητρίου Κωνσταντίνο για την ευκαιρία που μου έδωσε για να αναλάβω αυτό το project και να δουλέψω δίπλα του, καθώς και τον καθηγητή κύριο Χαραλάμπους Γεώργιο για την βοήθεια και την καθοδήγησή του καθ' όλη την διάρκεια της δουλείας μου.

Επίσης θα ήθελα να ευχαριστήσω τους γονείς μου για όλη την στήριξη που μου πρόσφεραν όσες φορές τους χρειάστηκα.

# Ανάπτυξη Αλγορίθμου για τον Πειραματικό Έλεγχο της Δομικής Ακεραιότητας Κατασκευών

ΝΙΚΟΛΟΠΟΥΛΟΣ ΚΩΝΣΤΑΝΤΙΝΟΣ

Τμήμα Μηχανολόγων Μηχανικών, Πανεπιστήμιο Θεσσαλίας, 2022

Επιβλέπων Καθηγητής: Δρ. Κωνσταντίνος Παπαδημητρίου,

Καθηγητής Δυναμικής Κατασκευών

## Περίληψη

Σκοπός της διπλωματικής είναι το στήσιμο ενός συστήματος με το οποίο ένας υπολογιστής θα επικοινωνεί με ένα τραπέζι διέγερσης, το οποίο θα προκαλεί δονήσεις σε μία κατασκευή. Ταυτόχρονα ο υπολογιστής θα λαμβάνει μετρήσεις από αισθητήρες, οι οποίοι θα είναι προσκολλημένοι πάνω στην κατασκευή. Στη συνέχεια αυτές τις μετρήσεις τις οποίες θα λάβει θα μπορεί να τις αποθηκεύσει ο υπολογιστής έτσι ώστε να μπορεί να τις επεξεργαστεί και να προσδιορίζει με βάση την ανάλυση των μετρήσεων τα χαρακτηριστικά της κατασκευής.

Για να λάβει χώρα το προαναφερθέν πείραμα χρησιμοποιείται ένα σύνολο από εργαλεία κατάλληλα για αυτό το σκοπό, καθώς και το κατάλληλο λογισμικό. Επιπλέον αναπτύχθηκε ένας κώδικας ο οποίος δίνει την εντολή ώστε να εκτελέσει το πείραμα, καθώς και να λάβει τις απαραίτητες μετρήσεις και να τις αποθηκεύσει. Τέλος γίνεται ανάλυση Fourier πάνω στα σήματα που λαμβάνονται καθ' όλη τη διαδικασία.

Στο παρόν έγγραφο λοιπόν παρουσιάζονται ο εξοπλισμός που χρειάστηκε για να εκτελεστεί η παραπάνω διαδικασία, καθώς και η συνδεσμολογία που χρειάστηκε. Επιπλέον θα γίνει η επεξήγηση αυτού του κώδικα που αναπτύχθηκε για αυτό το σκοπό, ενώ θα παρουσιαστεί ένα παράδειγμα εκτέλεσης του πειράματος για την εύρεση της συνάρτησης μεταφοράς ενός προβόλου και των ιδιοσυχνοτήτων της κατασκευής.

Λέξεις κλειδιά: **MATLAB, Κώδικας, Αισθητήρες, Ταλαντωτής, Fast Fourier Transform**

# **Algorithm Development for experimental identification of structural condition**

NIKOLOPOULOS KONSTANTINOS

Department of Mechanical Engineering, University of Thessaly, 2022

Supervising Professor: Dr. Konstantinos Papadimitriou,

Professor of Structural Dynamics

## **Summary**

The purpose of the diploma is to set up a system with which a computer will communicate with a shaking table, which will cause vibrations in a structure. At the same time the computer will receive measurements from sensors attached to the structure. Afterwards, these measurements that will be taken will be able to be saved so that it can be processed and to determine based on the analysis of the measurements the characteristics of the construction

To carry out the aforementioned experiment, a set of tools suitable for this purpose is used, as well as the appropriate software. In addition, a code was developed which gives the command to run the experiment, as well as to take the necessary measurements and save them. Finally, a Fourier analysis is performed on the signals received throughout the process.

This document therefore presents the equipment that was needed to perform the above procedure, as well as the wiring that was needed. In addition, this code developed for this purpose will be explained, while an example of running the experiment to find the transfer function of a cantilever and the idiosyncrasies of the structure will be presented.

**Keywords: MATLAB, Code, Sensors, Oscillator, Fast Fourier Transform**



## Περιεχόμενα

<b>1. Κατάλογος.....</b>	<b>11</b>
1.α . Κατάλογος Εικόνων.....	11
1.β Κατάλογος Σχεδίων.....	11
1.γ Κατάλογος Γραφημάτων.....	12
<b>2. Εισαγωγή.....</b>	<b>13</b>
<b>3. Λογισμικό.....</b>	<b>14</b>
3.α NI-DAQmx.....	14
3.β Measurement & Automation Explorer.....	15
3.γ MATLAB.....	16
<b>4. Εργαλεία.....</b>	<b>18</b>
4.α. NI BNC 2110.....	18
4.β Kistler Power supply/coupler.....	19
4.γ Kistler accelerometer.....	19
4.δ APS Amplifer model 124.....	20
4.ε APS Electronic-Seis .....	20
<b>5. Συνδεσμολογία.....</b>	<b>23</b>
<b>6. Θεωρία Fourier Transform.....</b>	<b>28</b>
6.α Σειρές Fourier.....	28
6.β Φόρμουλα Euler.....	29
6.γ Μετασχηματισμός Fourier.....	30
6.δ Discrete Fourier Transform.....	31
6. ε Fast Fourier Transform.....	32

<b>7. Κώδικας.....</b>	<b>33</b>
7.α Δημιουργία καναλιών.....	33
7.β Δήλωση στοιχείων πειράματος.....	36
7.γ Δημιουργία διέγερσης.....	37
7.δ Εισαγωγή και εξαγωγή δεδομένων.....	40
7.β Fast Fourier Transform.....	43
<b>8. Πειραματική διαδικασία .....</b>	<b>45</b>
8.α Διάταξη Πειράματος.....	45
8.β Εκτέλεση πειράματος.....	46
<b>9. Συμπέρασμα.....</b>	<b>50</b>
<b>10. Βιβλιογραφία.....</b>	<b>51</b>
<b>11. Παράρτημα.....</b>	<b>52</b>

## **1.α Κατάλογος Εικόνων**

Εικόνα 3.1: NI BNC 2110

Εικόνα 3.2: Kistler Power supply/coupler

Εικόνα 3.3: Kistler accelerometer

Εικόνα 3.4: APS Amplifer model 124

Εικόνα 3.5: APS Electronic-Seis

Εικόνα 3.6: APS ταλαντωτής

Εικόνα 4.1 : Σύνδεση του NI BNC

Εικόνα 4.2 : Θύρες εισαγωγής και εξαγωγής δεδομένων

Εικόνα 4.3 : Θύρες εισόδου/ εξόδου του Kistler power supply

Εικόνα 4.4 : Καλώδια σύνδεσης Kistler power supply με αισθητήρες

Εικόνα 4.5 : Καλώδια σύνδεσης Kistler power supply με αισθητήρες

Εικόνα 4.6: Διακόπτης ρύθμισης λειτουργίας σε τάση ή ένταση

Εικόνα 4.7 : Πίσω μέρος APS ενισχυτή με θύρες εισόδου και εξόδου 1β.

Εικόνα 5.1: γεωμετρικός τόπος φόρμουλας Euler

Εικόνα 7.1: Ενεργοποίηση καναλιών.

Εικόνα 7.2: Τοποθέτηση αισθητήρων

## **1.β Κατάλογος Σχεδίων**

Σχέδιο 4.1: Σχέδιο ένδειξης συνδεσμολογίας για δύο αισθητήρες.

Σχέδιο 4.2: Σχέδιο ένδειξης συνδεσμολογίας του ταλαντωτή

## 1.γ Κατάλογος Γραφημάτων

Γράφημα 5.1 : Παράδειγμα διέγερσης ημιτονοειδούς μορφής

Γράφημα 5.2 : Παράδειγμα διέγερσης τριγωνικής μορφής

Γράφημα 5.3 : Παράδειγμα διέγερσης white noise μορφής

Γράφημα 5.4: Αποτέλεσμα από αισθητήρα

Γράφημα 5.5: Αποτέλεσμα Fast Fourier Transform

Γράφημα 8.1 : Αποτελέσματα αισθητήρα βάσης

Γράφημα 8.2 : Αποτελέσματα αισθητήρα κεφαλής

Γράφημα 8.3 : Αποτελέσματα μετασχηματισμού Fourier αισθητήρα βάσης

Γράφημα 8.4: Αποτελέσματα μετασχηματισμού Fourier αισθητήρα κεφαλής

Γράφημα 8.5: Συνάρτηση μεταφοράς

## 2. Εισαγωγή

Στην καθημερινή ζωή, είναι ζωτικής σημασίας να μπορεί να εκλεχθεί η κατάσταση των κατασκευών, χωρίς να προκαλείτε μόνιμη ζημιά σε αυτήν. Για αυτό το σκοπό αναπτύχθηκαν διάφοροι μέθοδοι μη καταστρεπτικού ελέγχου, όπως ραδιογραφία, υπερήχους ή με χρήση διεισδυτικών υγρών. Η μέθοδος η οποία μελετάται σε αυτή την πτυχιακή είναι με τη δημιουργία ταλαντώσεων. Ο τρόπος με τον οποίο εκτελείτε είναι να προκαλείται ταλάντωση μέσω μίας συσκευής και στη συνέχεια να μετριέται η απόκριση της σε διάφορα σημεία αυτής. Είναι ένας τρόπος για να ελεγχθεί η κατάσταση της υγείας της κατασκευής που μελετάται. Καθιστά δυνατή την εύρεση ανωμαλίας στο υλικό σε πρώιμο στάδιο, ώστε να μπορέσει να αποφευχθεί η πλήρης αστοχία του. Είναι μια πολυχρησιμοποιημένη μέθοδος στο χώρο των μηχανολόγων και πολιτικών μηχανικών για τον έλεγχο κατασκευών όπως βιομηχανικών κατασκευών, γεφυρών, κτηρίων, ενεργειακών κατασκευών (π.χ. ανεμογεννήτριες, offshore κατασκευές).

Το πείραμα που περιγράφεται παρακάτω πραγματοποιήθηκε σε εργαστηριακό επίπεδο και ο κώδικας που αναπτύχθηκε δοκιμάστηκε σε μηχανήματα εντός αυτού. Παρόλα αυτά, με τη χρήση των κατάλληλων συσκευών, η χρήση του μπορεί να επεκταθεί και εκτός εργαστηρίου. Επίσης η Fast Fourier Transform μπορεί να εξελιχθεί και για πιο αναλυτική ανάλυση. Παρακάτω όπως περιγράφεται, η διαδικασία αυτή χρησιμοποιήθηκε μόνο για την εύρεση των ιδιοτιμών και της συνάρτησης μεταφοράς κατασκευής.

### 3.Λογισμικό

Το λογισμικό που χρειάζεται για να τεθεί σε λειτουργία το σύστημα είναι απαραίτητο για να μπορέσουν να επικοινωνήσουν οι συσκευές μεταξύ τους, καθώς και για αναπτυχθεί ο κώδικας. Για αυτό το σκοπό χρειάζονται 2 προγράμματα. Τα drivers, και συγκεκριμένα το NI-DAQmx, και η MATLAB στην οποία θα γραφεί ο κώδικας

#### 3 α. NI-DAQmx

Το NI-DAQmx είναι ένα λογισμικό το οποίο επιτρέπει την επικοινωνία με άλλα μηχανήματα για την λήψη και αποστολή δεδομένων. Έχει την δυνατότητα να αξιοποιηθεί από πολλά προγράμματα όπως LabVIEW, LabWindows/CVI και MATLAB, ώστε να μπορέσει να πραγματοποιηθεί ο προγραμματισμός των συσκευών προς χρήση.

Υπάρχουν πολλές εκδόσεις για το συγκεκριμένο πρόγραμμα και έπρεπε να γίνει η σωστή επιλογή, ανάλογα με διάφορους παράγοντες. Αυτοί είναι:

- Το λειτουργικό σύστημα το οποίο χρησιμοποιεί ο υπολογιστής για να τρέξει (Windows XP)
- Την έκδοση του προγράμματος που χρησιμοποιείται για να δημιουργηθεί ο κώδικας.  
(MATLAB 2013)
- Το μηχανήμα που συνδέονται με τον υπολογιστή  
(NI BNC-2110)

### 3 β. Measurement & Automation Explorer (MAX)

Η Measurement & Automation Explorer είναι μία εφαρμογή η οποία ενημερώνει τα υπόλοιπα προγράμματα ποιες συσκευές είναι αποθηκευμένες στο σύστημα και πώς είναι ρυθμισμένες. Συχνά μαζί με αυτό το πρόγραμμα εγκαθίσταται και το NI-DAQmx. Το συγκεκριμένο πρόγραμμα προσφέρει ένα γραφιστικό περιβάλλον, μέσα από το οποίο καθιστάτε δυνατή η:

- Διαμόρφωση του υλικού και το λογισμικού της National Instruments
- Δημιουργία και επεξεργασία καναλιών, εργασιών, διεπαφών, και κλιμάκων
- Εκτέλεση διαγνωστικών συστημάτων
- Προβολή συσκευών και οργάνων που είναι συνδεδεμένα στο σύστημά σας
- Ενημέρωση του λογισμικού

Επιπλέον μέσα από το MAX μπορεί να γίνει πρόσβαση στο DAQ Assistant. Αυτή η εφαρμογή δίνει την δυνατότητα:

- Δημιουργίας και επεξεργασίας εργασιών και εικονικών καναλιών
- Προσθήκη εικονικών καναλιών στις εργασίες
- Δημιουργία και επεξεργασία κλιμάκων
- Έλεγχος και αποθήκευση ρυθμίσεων
- Δημιουργία κώδικα στο λογισμικό εφαρμογής NI για χρήση σε διαφορετική εφαρμογή
- Προβολή διαγραμμάτων σύνδεσης για τους αισθητήρες

### 3 γ. MATLAB 2013

Η MATLAB είναι μια γλώσσα τέταρτης γενιάς. Χρησιμοποιείται για τον υπολογισμό, προγραμματισμό και την οπτικοποίηση δεδομένων σε ένα περιβάλλον φιλικό για τον χρήστη. Πιο συγκεκριμένα είναι κατάλληλη για:

- Μαθηματικούς υπολογισμούς
- Ανάπτυξη αλγορίθμων
- Απόκτηση δεδομένων
- Μοντελοποίηση, προσομοίωση και δημιουργία πρωτοτύπων
- Ανάλυση, εξερεύνηση και οπτικοποίηση δεδομένων
- Ανάπτυξη εφαρμογών, καθώς και την δημιουργία του οπτικού περιβάλλοντος

Η MATLAB είναι ένα σύστημα του οποίου το βασικό στοιχείο δεδομένων είναι πίνακας που δεν απαιτεί διαστασιολόγηση. Αυτό επιτρέπει τη λύση πολλών υπολογιστικών προβλημάτων, ειδικά αυτών των οποίων τα στοιχεία είναι αποθηκευμένα σε μορφή μητρώων ή διανύσματος, σε πολύ γρήγορο χρονικό διάστημα. Αυτό επιτυγχάνεται με τη χρήση βιβλιοθηκών LAPACK και BLAS τις οποίες έχει ενσωματωμένες. Η MATLAB διαθέτει πληθώρα πρόσθετων λύσεων για συγκεκριμένες εφαρμογές που ονομάζονται εργαλειοθήκες. Οι εργαλειοθήκες είναι ολοκληρωμένες συλλογές συναρτήσεων MATLAB που επεκτείνουν το περιβάλλον της MATLAB για την επίλυση συγκεκριμένων κατηγοριών προβλημάτων.

Το σύστημα MATLAB αποτελείται από πέντε κύρια μέρη:

- Αναπτυξιακό Περιβάλλον: Αυτό είναι το σύνολο εργαλείων και εγκαταστάσεων που βοηθάει στη χρήση λειτουργιών και αρχείων MATLAB. Περιλαμβάνεται η επιφάνεια εργασίας της MATLAB, το παράθυρο εντολών, ένα ιστορικό εντολών, ένα πρόγραμμα επεξεργασίας, πρόγραμμα εντοπισμού σφαλμάτων και προγράμματα περιήγησης για την προβολή βοήθειας, του χώρου εργασίας, τον χώρο των αρχείων, και τη διαδρομή αναζήτησης.



- Η βιβλιοθήκη μαθηματικών συναρτήσεων: Είναι μια τεράστια συλλογή υπολογιστικών αλγορίθμων. Κυμαίνονται από στοιχειώδεις συναρτήσεις, όπως πρόσθεση, έως πιο περίπλοκες συναρτήσεις όπως αντιστροφή μητρώων, μητρώα ιδιοτιμών, συναρτήσεις Bessel και γρήγοροι μετασχηματισμοί Fourier.
- Γλώσσα: Είναι μια γλώσσα μητρώων/πινάκων υψηλού επιπέδου με, συναρτήσεις, δομής δεδομένων, είσοδος/εξόδου και αντικειμενοστραφή προγραμματισμού, που επιτρέπουν στην Matlab από τη δημιουργία από σύντομων προγραμμάτων και απλών υπολογισμών, μέχρι και τον προγραμματισμό πιο σύνθετων προγραμμάτων και τη δημιουργία εφαρμογών.
- Γραφικά: Διατίθεται μια πληθώρα δυνατοτήτων για την παρουσίαση διανυσμάτων και πινάκων ως γραφήματα, καθώς και την επεξεργασία και εκτύπωση αυτών. Περιλαμβάνει λειτουργίες για δισδιάστατα και τρισδιάστατα δεδομένα οπτικοποίησης, επεξεργασία εικόνας, κινούμενα σχέδια και γραφικά παρουσίασης. Επίσης περιλαμβάνονται λειτουργίες που επιτρέπουν την πλήρη προσαρμογή και εμφάνιση γραφημάτων, καθώς και τη δημιουργία και προσαρμογή του περιβάλλοντος των εφαρμογών.
- Οι εξωτερικές διεπαφές/API του MATLAB: Είναι μια βιβλιοθήκη που επιτρέπει τον προγραμματισμό σε γλώσσα C και Fortran, οι οποίες έχουν την δυνατότητα αλληλεπίδρασης με την MATLAB. Έχει την δυνατότητα για κλήση ρουτίνων από την MATLAB (δυναμική σύνδεση), καλώντας την ως υπολογιστική μηχανή και για ανάγνωση και εγγραφή αρχείων MAT.

## 4.Εργαλεία

### 4 α. NI BNC 2110

Ο NI BNC 2110 αποτελείται από 2 κομμάτια. Το πρώτο είναι μία Multifunction DAQ κάρτα η οποία συνδέεται στην κεντρική μονάδα του υπολογιστή, και το δεύτερο είναι ένα σύστημα ακροδεκτών, εξωτερικά από την κεντρική μονάδα. Αυτή η συσκευή διαμεσολαβεί ανάμεσα στην κεντρική μονάδα του υπολογιστή και στις διάφορες συσκευές που χρειάζονται να χρησιμοποιηθούν. Η κάρτα είναι υπεύθυνη για την λειτουργία της συσκευής, ενώ η χρήση των ακροδεκτών είναι απλά οι θύρες για την σύνδεσή της συσκευής με τα υπόλοιπα μηχανήματα. Έχει σχεδιαστεί για την ταυτόχρονη σύνδεση και χρήση πολλαπλών συσκευών. Συγκεκριμένα προσφέρει 10 θύρες για την σύνδεση με άλλες συσκευές.



Εικόνα 3.1: Θύρες ακροδεκτών NI BNC 2110

#### 4 β. Kistler Power supply/coupler

Αυτή η συσκευή λειτουργεί ως διαμεσολαβητής για μηχανήματα τα οποία μπορούν να μετατρέψουν μηχανικό έργο σε ηλεκτρικό ρεύμα. Χρησιμοποιείται κυρίως για λήψη δεδομένων από επιταχυνσιόμετρα μίας κατεύθυνσης ή και τριών κατευθύνσεων. Τέλος η δημιουργία θορύβου μικρού μεγέθους το κάνει ιδανικό για μελέτη μικρών ταλαντώσεων.



Εικόνα 3.2: Kistler power supply/coupler

#### 4 γ. Kistler accelerometer

Το κυβικό σχήμα του συγκεκριμένου επιταχυνσιόμετρου επιτρέπει την εύκολη εγκατάσταση σε κατασκευές, ενώ το μικρό βάρος το κάνει ικανό να μην επηρεάζει ιδιαίτερα τα χαρακτηριστικά της κατασκευής. Μπορεί να χρησιμοποιηθεί με μεγάλη ποικιλία καλωδίων και έχει χαμηλή ευαισθησία.



Εικόνα 3.3: Επιταχυνσιόμετρο Kistler

#### 4 δ. APS Amplifer model 124

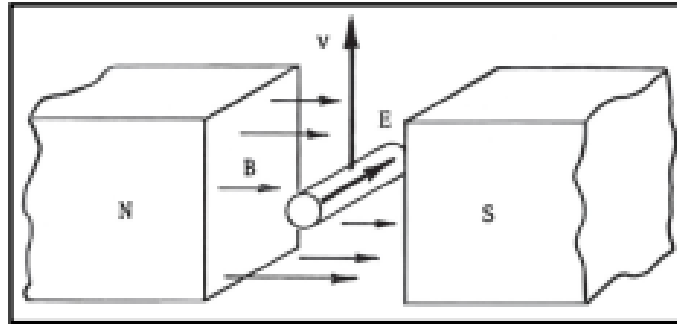
Είναι ένας ενισχυτής, ο οποίος συμβάλει στην μεταφορά σήματος για την δημιουργία διέγερσης. Έχει δύο μενού λειτουργίας. Συγκεκριμένα μπορεί δουλέψει ως ενισχυτής έντασης ρεύματος ή τάσης.



Εικόνα 3.4: Ενισχυτής APS

#### 4 ε. APS Electronic-Seis

Το APS 113 ELECTRO-SEIS είναι βασικά μια συσκευή παραγωγής δύναμης. Η βασική αρχή λειτουργίας του είναι αυτή της παραγωγής δύναμης σε έναν αγωγό που φέρει ρεύμα, που βρίσκεται σε ένα μαγνητικό πεδίο συνεχούς ρεύματος, που είναι κάθετα προς την κατεύθυνση του ρεύματος. Η δύναμη δρα στον αγωγό που μεταφέρει το ρεύμα. Το μέγεθος της παραγόμενης δύναμης είναι ανάλογο με τη στιγμιαία τιμή του ρεύματος που ενεργεί σε κατεύθυνση κάθετη προς τις διευθύνσεις του ρεύματος και του μαγνητικού πεδίου. Είναι επομένως ικανός να παράγει οποιαδήποτε συνάρτηση δύναμης που ενεργεί μεταξύ του σπλισμού και του σώματός του, ανάλογα με την συνάρτηση ρεύματος που του παρέχεται. Σε αυτή τη συσκευή λαμβάνει χώρα το πείραμα. Έχει μέγιστη διαδρομή 158 mm, μέγιστο διάνυσμα δύναμης 133 N και εύρος συχνοτήτων μεταξύ 0 και 200 Hz.



Εικόνα 3.5: Τρόπος λειτουργίας APS Electronic-Seis

Το APS Electronic-Seis μπορεί να λειτουργήσει σε συνδυασμό με διάφορα επιπρόσθετα εξαρτήματα, ανάλογα με την χρήση που χρειάζεται να γίνει. Το συγκεκριμένο μηχάνημα έχει σχεδιαστεί ειδικά για να λειτουργεί σε τρεις κύριες λειτουργίες. Αυτές είναι:

- **ΛΕΙΤΟΥΡΓΙΑ ΣΤΑΘΕΡΟΥ ΣΩΜΑΤΟΣ:** Στη λειτουργία σταθερού σώματος, το σώμα του μηχανήματος στερεώνεται πάνω στο αντικείμενο προς δοκιμή. Σε αυτόν τον τρόπο λειτουργίας, η δύναμη που παρέχεται στη σώμα προς δοκιμή μπορεί να μετρηθεί με την σύνδεση ενός force transducer στη σύνδεση ώσης μεταξύ του οπλισμού και του σώματος προς δοκιμή. Εναλλακτικά, με την αφαίρεση των λωρίδων ανάρτησης οπλισμού/σώματος, το ρεύμα ανακίνησης μπορεί να χρησιμοποιηθεί ως άμεσο μέτρο της παραγόμενης δύναμης. Η παραγόμενη δύναμη είναι περίπου ίση με τη δύναμη που παρέχεται στο σώμα προς δοκιμή κατά τον συντονισμό, επειδή η μάζα του οπλισμού είναι τυπικά μικρή σε σύγκριση με τη μάζα των περισσότερων κατασκευών προς δοκιμή.
- **ΛΕΙΤΟΥΡΓΙΑ ΕΛΕΥΘΕΡΟΥ ΣΩΜΑΤΟΣ:** Τα επιθυμητά σημεία στα οποία χρειάζεται να ασκηθεί δύναμη σε πολλές κατασκευές βρίσκονται σε σημαντική απόσταση πάνω από το επίπεδο του εδάφους. Για δοκιμές σε τέτοιες περιπτώσεις, καθίσταται δύσκολος και δαπανηρός ο σχεδιασμός, η κατασκευή και η χρήση δομών υποστήριξης λειτουργία σταθερού σώματος που παραμένουν μη συντονισμένες στο εύρος συχνοτήτων ενδιαφέροντος. Για τέτοιες εφαρμογές, η λειτουργία ελευθέρου σώματος προτείνεται προς χρήση. Σε αυτό τον τρόπο λειτουργίας, συσκευή λειτουργεί ως μια μάζα αντίδρασης,

στηριζόμενη από στήριγμα από πάνω της. Στη λειτουργία ελεύθερου σώματος, η δύναμη που παρέχεται στη κατασκευή προς δοκιμή μπορεί να μετρηθεί με την εισαγωγή ενός force transducer μεταξύ του οπλισμού και της δομής. Επίσης μπορεί να χρησιμοποιηθεί ρεύμα οπλισμού, αλλά μόνο σε συχνότητες πολύ πάνω από τη συχνότητα συντονισμού ανάρτησης. Δεν παρέχει ακριβή μέτρηση της δύναμης που παρέχεται στον οπλισμό και το φορτίο σε πολύ χαμηλές συχνότητες. Ωστόσο, ένα πολύ βολικό μέτρο της δύναμης φορτίου είναι η αξονική επιτάχυνση του σώματος της συσκευής. Η δύναμη που επενεργεί στη δομή συντονισμού είναι ίση με την δύναμη που ασκείται στο σώμα του μηχανήματος, όπως μετριέται από τη μάζα του σώματος και την επιτάχυνση, ακόμη και κοντά και κάτω από τον συντονισμό ανάρτησης οπλισμού αμαξώματος. Αυτό μπορεί να επιτρέψει σημαντική απλοποίηση των οργάνων του συστήματος δοκιμής, καθώς η δύναμη μπορεί να μετρηθεί με ένα σύστημα απλό επιταχυνσιόμετρο. Στη λειτουργία ελεύθερου σώματος δεν υπάρχει πρόσθετος εξωτερικός περιορισμός στο σώμα για στατική στήριξη βάρους. Ως εκ τούτου, ένα αξονικά προσανατολισμένο επιταχυνσιόμετρο στο σώμα μπορεί να παράγει ένα ακριβές σήμα δύναμης.

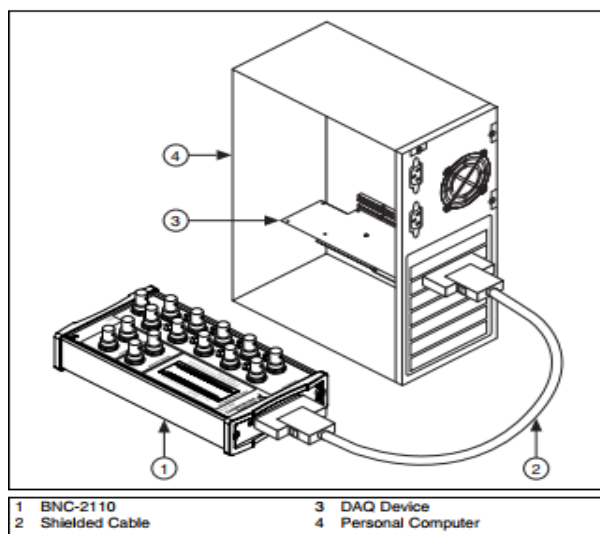
- **ΛΕΙΤΟΥΡΓΙΑ ΕΛΕΥΘΕΡΟΥ ΤΡΑΠΕΖΙΟΥ:** Ο συγκεκριμένος τρόπος είναι αυτός ο οποίος χρησιμοποιήθηκε για την διεξαγωγή του πειράματος. Το APS 113 μπορεί να χρησιμοποιηθεί σε μια κατακόρυφη λειτουργία ελεύθερου οπλισμού ακουμπώντας το πίσω ακραίο πλαίσιο του μηχανήματος στην οριζόντια επιφάνεια της δομής δοκιμής. Σε αυτόν τον τρόπο λειτουργίας, το μηχάνημα κάνει μόνο ένα το κομμάτι του τραπέζιου να ταλαντωθεί με ένα συγκεκριμένο ρυθμό και διαδρομή. Πάνω σε αυτό το τραπέζι είναι βιδωμένο το σώμα προς μελέτη.



Εικόνα 3.6: APS ταλαντωτής

## 5. Συνδεσμολογία

Το πρώτο και πιο σημαντικό εξάρτημα είναι το σύστημα ακροδεκτών NI BNC 2110. Για να συμβεί αυτό πρέπει να είναι συνδεδεμένο στην κεντρική μονάδα η Multifunction DAQ κάρτα. Σε αυτή την κάρτα γίνεται η εισαγωγή και εξαγωγή των δεδομένων. Η σύνδεση γίνεται με ένα καλώδιο SHC68-68-EPM από την πίσω πλευρά της κεντρικής μονάδας (Εικόνα 5.1).



Εικόνα 4.1: Σύνδεση του NI BNC

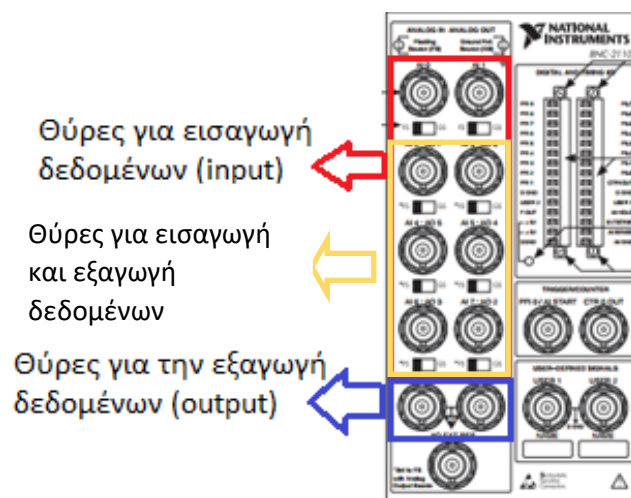
Πάνω σε αυτή τη συσκευή συνδέονται όλες οι υπόλοιπες συσκευές, έμμεσα ή άμεσα. Για αυτό το σκοπό χρησιμοποιούνται τα κατάλληλα καλώδια. Πιο συγκεκριμένα αυτά τα καλώδια είναι τα BNC αρσενικά (Εικόνα 4.2),



Εικόνα 4.2: καλώδια BNC αρσενικά

Η θέση στην οποία συνδέονται τα καλώδια εξαρτάται από τον σκοπό τον οποίο πρόκειται να εκτελέσουν. Υπάρχουν τρεις ειδών θύρες:

- Οι θύρες για την εισαγωγή δεδομένων (input). Η αρίθμηση ξεκινάει από το μηδέν μέχρι το μηδέν μέχρι το ένα (από το AI0 μέχρι το AI1).
- Οι θύρες για την εξαγωγή δεδομένων (output). Η αρίθμηση ξεκινάει από το μηδέν μέχρι το μηδέν μέχρι το ένα (από το AO0 μέχρι το AO1).
- Οι θύρες που εξυπηρετούν και τους δύο σκοπούς (από AI2 ή AO7 μέχρι AI7 μέχρι AO2)



Εικόνα 4.3: Θύρες εισαγωγής και εξαγωγής δεδομένων



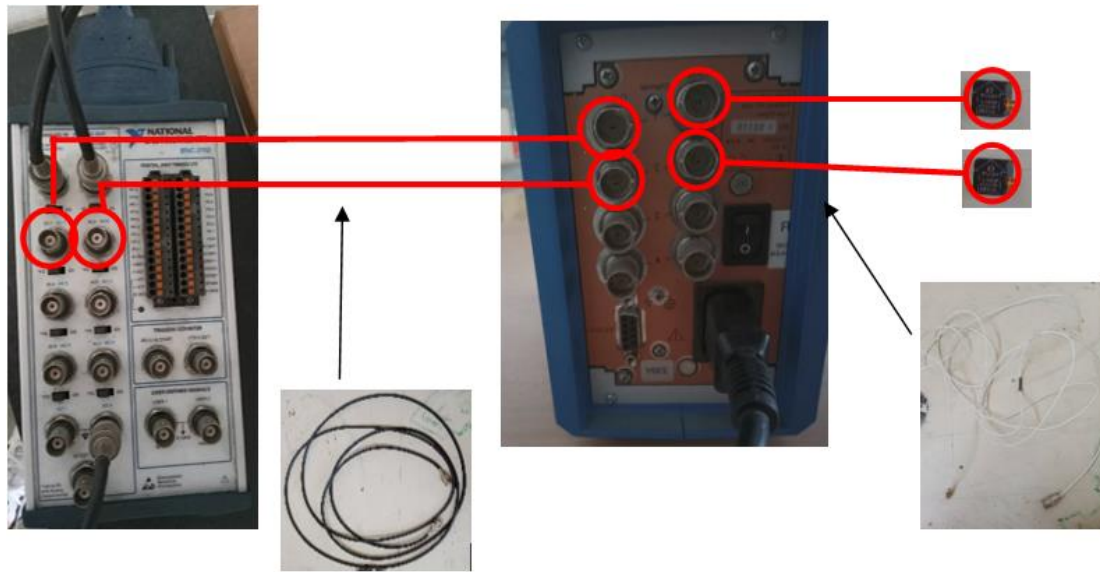
Στις θύρες εισαγωγής συνδέονται όσα καλώδια BNC όσοι και οι αισθητήρες οι οποίοι χρειάζονται, και καταλήγουν στο πίσω μέρος του Kistler power supply. Εκεί το καθένα συνδέεται σε μία από τις τέσσερις θύρες εξαγωγής με την ένδειξη output. Στην συνέχεια χρειάζεται ακόμη ένα καλώδιο ώστε να συνδεθεί με τους αισθητήρες. Αυτό το καλώδιο συνδέεται στις θύρες με την ένδειξη input και βιδώνουν πάνω στα επιταχυνσιόμετρα. Όπως και με τα καλώδια BNC κάθε ένα καλώδιο αντιστοιχεί σε έναν αισθητήρα.



*Εικόνα 4.4: Θύρες εισόδου/ εξόδου του Kistler power supply*



*Εικόνα 4.5: Καλώδια σύνδεσης Kistler power supply με αισθητήρες*



Σχέδιο 4.1: Σχέδιο ένδειξης συνδεσμολογίας για δύο αισθητήρες.

Παρόλο που υπάρχουν οχτώ θύρες στην κάρτα NI-BNC 2110, στον ενισχυτή Kistler υπάρχουν μόνο τέσσερις. Για αυτό τον λόγο μπορούν να συνδεθούν το πολύ τέσσερις αισθητήρες.

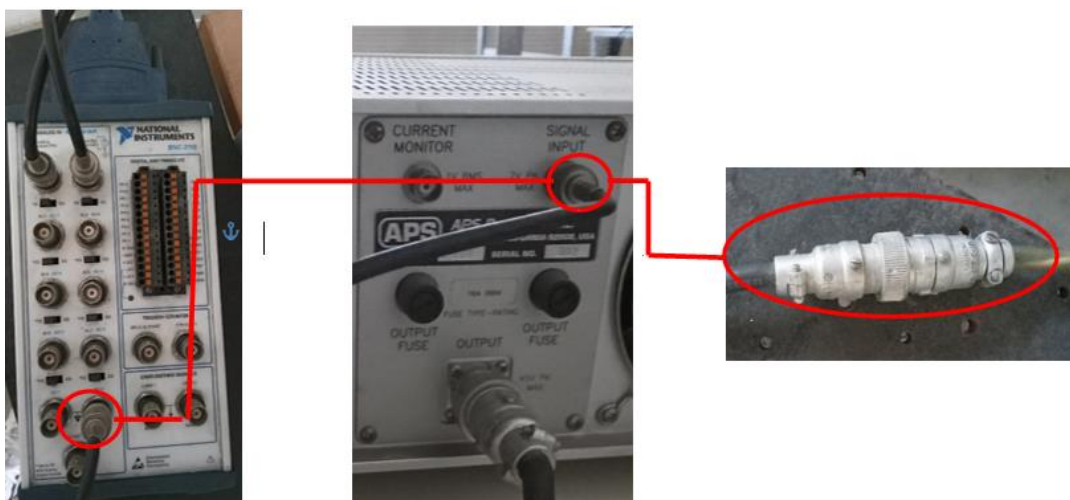
Για την έξοδο δεδομένων η συνδεσμολογία γίνεται με αντίστοιχο τρόπο. Η μία άκρη ενός καλωδίου BNC συνδέεται σε μία από τις θύρες εξόδου του BNC 2110, και η άλλη με τον APS amplifier στη θύρα signal input, στο πίσω μέρος. Ρυθμίζουμε τον ενισχυτή να λειτουργεί σε volt, καθώς θα χρησιμοποιήσουμε τάση για την μετάδοση του σήματος. Αυτό στη συνέχεια συνδέεται με τον ταλαντωτή για να εκτελεστεί το πείραμα. Πάνω στη βάση του ταλαντωτή είναι τοποθετείται το αντικείμενο προς μελέτη. Στο συγκεκριμένο παράδειγμα είναι ένας πρόβολος. Πάνω σε αυτό το πρόβολο θα γίνει η εναπόθεση των αισθητήρων με κατάλληλο κερί.



Εικόνα 4.6: Διακόπτης ρύθμισης λειτουργίας σε τάση ή ένταση ρεύματος



Εικόνα 4.7: Πίσω μέρος APS ενισχυτή με θύρες εισόδου και εξόδου



Σχέδιο 4.2: Σχέδιο ένδειξης συνδεσμολογίας του ταλαντωτή

## 6. Θεωρία Fourier transform

### 6.a Σειρές Fourier

Ο μετασχηματισμός Fourier βασίζεται σε μεγάλο βαθμό στις σειρές Fourier. Οι σειρές Fourier είναι ένας τρόπος προσέγγισης μιας συνάρτησης ως άθροισμα συναρτήσεων ημιτόνου και συνημίτονου. Πρακτικά είναι ένας τρόπος αντικατάστασης συναρτήσεων από μία σειρά τριγωνομετρικών συναρτήσεων

$$f(t) = \sum_{k=1}^n A_k \sin(\omega_k t) + B_k \cos(\omega_k t)$$

Χρησιμοποιείται το άθροισμα και ημιτόνων και συνημίτονων και όχι μόνο ημιτόνων, έτσι ώστε να είναι δυνατή η έκφραση συναρτήσεων για τις οποίες η συνάρτηση  $f$  δεν ισούται με 0 για την τιμή  $t=0$ . Ένα επιπλέον πλεονέκτημα της χρήσης και των δύο όρων είναι ότι δεν χρειάζεται η χρήση αρχικής φάσης σε κανέναν από τους τριγωνομετρικούς όρους.

Στη περίπτωση όπου χρειάζεται να προσεγγιστεί μία μιγαδική συνάρτηση οι σειρές Fourier παίρνουν την μορφή:

$$f(t) = \sum_{k=1}^n (A_k \sin(\omega_k t) + iB_k \cos(\omega_k t))$$

και χρησιμοποιώντας τον τύπο του Euler, ο τύπος γίνεται:

$$f(t) = \sum_{k=1}^n C_k e^{i\omega_k t}$$

Για να είναι δυνατόν μία συνάρτηση να αναπαρασταθεί από μία σειρά Fourier που συγκλίνει πρέπει να είναι περιοδική, και η περίοδος είναι ίση με αυτή των ημιτόνων και συνημίτονων που χρησιμοποιούνται.

Για να μπορέσει να εκφραστεί μία συνάρτηση με αυτό πρέπει οι συναρτήσεις που χρησιμοποιούνται να είναι ορθογωνικές. Αυτό ισχύει για στην συγκεκριμένη περίπτωση (δηλαδή το ημίτονο με το συνημίτονο είναι ορθογωνικές μεταξύ τους.). Για να αποδειχθεί αυτό πρέπει:

$$\int_{-\pi}^{\pi} e^{i\omega_{\kappa}t} * e^{i\omega_j t} = 0 \text{ για κάθε } j \neq \kappa$$

Άρα

$$\frac{1}{i(\omega_j - \omega_{\kappa})} [e^{i(\omega_j - \omega_{\kappa})t}]_{-\pi}^{\pi} = 0 \Rightarrow$$

$$\frac{1}{i(\omega_j - \omega_{\kappa})} (e^{i\omega_{\kappa}\pi} - e^{i\omega_j(-\pi)}) = 0$$

Είναι γνωστό όμως ότι η συνάρτηση  $e^{i2\pi\omega_{\kappa}t}$  είναι περιοδική από στο διάστημα  $-\pi$  έως  $\pi$ , οπότε η διαφορά της αφαίρεσης είναι μηδέν. Για την περίπτωση όπου  $j=\kappa$  δημιουργείται ένα υπόλοιπο  $0/0$ , του οποίου το όριο είναι ίσο με  $2\pi$ .

## 6.β. Η Φόρμουλα του Euler

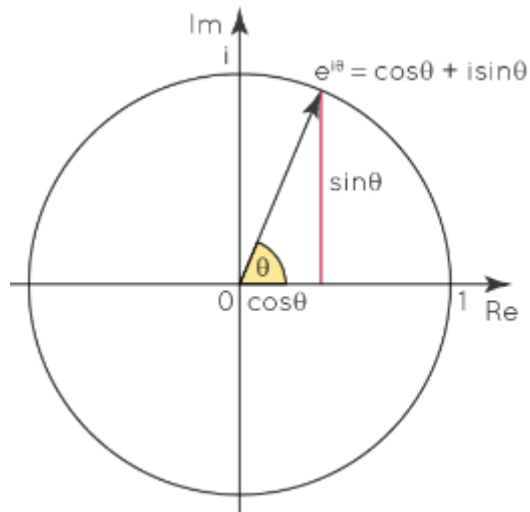
Είναι γνωστό από την τριγωνομετρία ότι ένα άθροισμα ημιτόνου με συνημίτονο μπορεί να εκφραστεί ως τον νεπέριο αριθμό υψωμένο σε έναν εκθέτη. Η σχέση η οποία συνδέει αυτούς τους όρους είναι γνωστή ως φόρμουλα Euler.

$$e^{ix} = \cos(x) + i\sin(x)$$

Είναι ένας τύπος χρήσιμος για την επίλυση μιγαδικών εκθετικών συναρτήσεων. Η φόρμουλα του Euler, για τιμές την ειδική περίπτωση όπου  $x = \pi$ , παίρνει την ειδική μορφή:

$$e^{i\pi} + 1 = 0$$

Χρησιμοποιείται για τον καθορισμό της σχέσης μεταξύ τριγωνομετρικών συναρτήσεων και μιγαδικών εκθετικών συναρτήσεων. Αυτός ο τύπος μπορεί να παρασταθεί στο μιγαδικό επίπεδο ως συνάρτηση  $f = e^{i\theta}$ , της οποίας ο γεωμετρικός τόπος είναι ένας κύκλος ακτίνας μία μονάδα. Η συνάρτηση αυτή έχει ως εξαρτημένη μεταβλητή την τιμή  $\theta$ , η οποία μετριέται σε ακτίνια.



Εικόνα 5.1: γεωμετρικός τόπος φόρμουλας Euler

## 6.γ Μετασχηματισμός Fourier

Οι συμβατικές σειρές Fourier μπορούν να χρησιμοποιηθούν μόνο για συναρτήσεις που είναι περιοδικές. Για τις μη περιοδικές συναρτήσεις έχει αναπτυχθεί η διαδικασία που ονομάζεται μετασχηματισμός Fourier. Αντιμετωπίζει αυτές τις συναρτήσεις ως περιοδικές με άπειρη περίοδο. Χρησιμοποιείται κυρίως για την μετατροπή συναρτήσεων και από το πεδίο του χρόνου στο πεδίο της συχνότητας. Εμφανίζεται σε μεγάλος εύρος της επιστημών, όπως στην ηλεκτρολογία, φυσική και μηχανολογία.

Ο τύπος για τον μετασχηματισμό Fourier είναι:

$$f(t) = \sum_{-\infty}^{\infty} C_k e^{-i\omega_k t}$$

Ο συγκεκριμένος τύπος στην πραγματικότητα είναι ό ίδιος με αυτόν της σειράς Fourier. Το  $\omega$  και στις δύο περιπτώσεις ισούται με  $2\pi/L$ , όπου  $L$  είναι το μήκος της περιόδου. Στην περίπτωση όπου δεν είναι περιοδική η συνάρτηση, θεωρώντας άπειρη περίοδο, συνεπάγεται ότι  $L \rightarrow \infty$ . Αυτό έχει ως αποτέλεσμα το  $\omega \rightarrow 0$ . Αυτό μετατρέπει το άπειρο άθροισμα σε ολοκλήρωμα.

$$f(t) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} d\omega$$

## 6.δ Θεωρία Discrete Fourier Transform (DFT)

Ο Discrete Fourier Transform (DFT) είναι το ισοδύναμο του συνεχούς μετασχηματισμού Fourier για σήματα για δεδομένο αριθμό δειγμάτων. Μετατρέπει ένα πεπερασμένο αριθμό δειγμάτων μιας συνάρτησης, τα οποία έχουν ίση απόστασης μεταξύ τους. Έστω ότι  $f(k)$  είναι η  $k$  στην σειρά τιμή η οποία λαμβάνεται από τους αισθητήρες. Τότε η συνάρτηση Fourier μπορεί να μετασχηματιστεί σε:

$$f_n = \sum_{k=0}^{N-1} F e^{-2\pi i n k / N}$$

Όταν  $\omega_j = 2\pi i / N$

Η συγκεκριμένη έκφραση μπορεί να παρουσιαστεί με την μορφή μητρώων ως:

$$\begin{pmatrix} f_0 \\ f_1 \\ \cdot \\ \cdot \\ \cdot \\ f_n \end{pmatrix} = \begin{pmatrix} 1 & 1 & \cdot & \cdot & \cdot & 1 \\ 1 & W_N & \cdot & \cdot & \cdot & W_N^n \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ W_N^{n^2} & \cdot & \cdot & \cdot & \cdot & \cdot \end{pmatrix} \begin{pmatrix} F_0 \\ F_1 \\ \cdot \\ \cdot \\ \cdot \\ F_n \end{pmatrix}$$

Με  $W_N = e^{-j2\pi n k / N}$

## 6.ε Θεωρία Fast Fourier Transform (FFT)

Η μέθοδος DFT είναι πρακτικά ασύμφορη όσον αφορά την χρήση της σε κώδικα, διότι είναι πολύ απαιτητική σε υπολογιστική δύναμη και χρονοβόρα. Συγκεκριμένα για  $N$  αριθμό δεδομένων χρειάζονται να γίνουν  $N^2$  πράξεις ( $O(N^2)$ ). Για να αντιμετωπιστεί αυτό το πρόβλημα λοιπόν αναπτύχθηκε ένας αλγόριθμος ο οποίος ονομάζεται Fast Fourier Transform. Ο αριθμός των πράξεων σε αυτή την περίπτωση είναι της τάξης μεγέθους του  $N \log(N)$  ( $O(N \log(N))$ ).

Η συγκεκριμένη μέθοδος εκμεταλλεύεται το γεγονός ότι στο μητρώο των συντελεστών εμφανίζονται πολλοί επαναλαμβανόμενοι συντελεστές, οι οποίοι χρειάζεται να υπολογιστούν πολλές φορές. Χωρίζουμε λοιπόν το άθροισμα σε δύο άλλα αθροίσματα, το ένα για ζυγούς στη σειρά αριθμούς και ένα για μονούς. Το κάθε άθροισμα σε έχει περίοδο ίση με το μισό της αρχικής περιόδου. Οπότε η σχέση γίνεται

$$f_n = \sum_{k=0}^{\frac{N}{2}-1} F(2k)W^{n2k} + \sum_{k=0}^{\frac{N}{2}-1} F(2k+1)W^{n(2k+1)}$$

Η παράμετρος  $k$  εκφράζεται με αυτό τον τρόπο για να ορίζονται ζυγές και ακέραιες τιμές αντίστοιχα. Επίσης ισχύει ότι

$$W_N^{2kn} = e^{-i2\pi(2kn)/N} = e^{\frac{-i2\pi kn}{N/2}} = W_{N/2}^{kn}$$

Οπότε η σχέση γίνεται

$$f_n = \sum_{k=0}^{N/2-1} F(2k)W_{N/2}^{nk} + W_{N/2}^k \sum_{k=0}^{N/2-1} F(2k+1)W^{nk}$$

Με αυτό τον τρόπο μπορούν να χρησιμοποιηθούν μόνο τα μισά δεδομένα για τον μετασχηματισμό, ρίχνοντας την ταχύτητα στο μισό. Η ίδια διαδικασία όμως μπορεί να επαναληφθεί, ρίχνοντας τον απαιτούμενο χρόνο για τον υπολογισμό ακόμη πιο κάτω.



## 7. Κώδικας

### 7 α. Δημιουργία καναλιών

Ο κώδικας δημιουργήθηκε για να εκτελέσει τρεις συγκεκριμένες λειτουργίες. Η πρώτη είναι να μπορεί να στέλνει την κατάλληλη διέγερση με τα χαρακτηριστικά τα οποία θέλει ο χρήστης. Η δεύτερη είναι η ταυτόχρονη λήψη δεδομένων από τους αισθητήρες. Τέλος η τρίτη λειτουργία είναι να μπορεί να επεξεργαστεί τα δεδομένα. Πιο συγκεκριμένα να μπορεί να βρίσκει ιδιοσυχνότητες του σώματος που μελετάται.

```
% clear workspace
```

```
clear
```

```
close all
```

Οι πρώτες δύο εντολές είναι οι clear και η close all. Η clear διαγράφει όλες τις αποθηκευμένες μεταβλητές του που είναι αποθηκευμένες, ενώ η close all οποιοδήποτε παράθυρο της MATLAB είναι ανοιχτό, όπως γραφικές παραστάσεις ή εικόνες. Η εντολή clear είναι πολύ σημαντική διότι σε περίπτωση που μείνουν αποθηκευμένες τιμές από προηγούμενο πείραμα, όταν θα πάει να εκτελεστεί το επόμενο, μπορεί να γίνει ανάμειξη των δύο τιμών, ή ακόμη και να βγάλει πρόβλημα ο κώδικας και να μην τρέξει.

```
% Discover Devices
```

```
devices = daq.getDevices;
```

```
devices(1)
```

Η εντολή daq.getDevices εμφανίζει στη γραμμή εντολών της MATLAB μία λίστα με όλες τις συσκευές οι οποίες είναι συνδεδεμένες στον υπολογιστή. Στην συγκεκριμένη περίπτωση εμφανίζει την κάρτα NI BNC 2110 και ότι είναι της εταιρίας National Instrument. Επίσης η εντολή devices(1) παρουσιάζει δεδομένα για την πρώτη συσκευή, όπως τι είδους μετρήσεις μπορεί να λάβει και να στείλει.

```
%% Create a session
```

```
s = daq.createSession('ni');
```

Με αυτή την εντολή δημιουργείτε ένα session το οποίο θα επικοινωνεί με συσκευές τις National Instrument. Το session είναι μια δομή στην οποία καθορίζονται όλα τα χαρακτηριστικά του πειράματος, όπως η συχνότητα με την οποία θα λαμβάνονται και θα στέλνονται δεδομένα. Στο session είναι απαραίτητο να δηλωθεί ένα όνομα. Ο λόγος είναι ότι σε παρακάτω εντολές θα χρειαστεί να καθοριστεί το session με το οποίο σχετίζονται.

```
%% Ask how many chanel does the user need
```

```
prompt = 'How many sensors do you want?: ';
```

```
i = input(prompt);
```

```
while (i<1 || i>4)
```

```
prompt = 'Invalid value. You can only use from 1 to 4 sensors.How many sensors do you want?: ';
```

```
i = input(prompt);
```

```
end
```

```
%% create chanel
```

```
%input
```

```
for j=1:i;
```

```
ch_i(j)= addAnalogInputChannel(s,'Dev1', (j-1), 'Voltage');
```

```
end
```

```
%output
```

```
ch_o=addAnalogInputChannel(s, 'Dev1', 1, 'Voltage');
```

Το επόμενο βήμα είναι η δημιουργία καναλιών. Τα κανάλια είναι αυτά τα οποία είναι υπεύθυνα για την επικοινωνία με τα διάφορα εξαρτήματα τα οποία χρησιμοποιούνται. Κάθε εξάρτημα αντιστοιχεί σε ένα κανάλι, είτε είναι για εισαγωγή δεδομένων είτε για εξαγωγή. Για κανάλι εισαγωγής δεδομένων χρησιμοποιείτε η εντολή `addAnalogInputChannel`, ενώ για την εξαγωγή δεδομένων η `addAnalogOutputChannel`. Σε οποιαδήποτε από τις δύο περιπτώσεις πρέπει να θέτονται οι εξής παράμετροι.

- Το session στο οποίο ανήκει το κάθε κανάλι:

Σε αυτή την περίπτωση για όλες τις συσκευές είναι το όνομα του session, δηλαδή `s`. Δεν χρειάζεται να τεθεί άλλο session σε κάποιο κανάλι, καθώς οι παραμέτρους για όλα τα κανάλια είναι οι ίδιες και θα οριστούν σε ένα session.

- Η συσκευή η οποία αφορά το κανάλι:

Το πρόγραμμα αναγνωρίζει μόνο μία συσκευή και αυτή είναι η κάρτα NI-BNC, καθώς αυτή μεσολαβεί για την επικοινωνία όλων των υπολοίπων μηχανημάτων. Το όνομα με το οποίο δηλώνεται είναι `Dev1`. Οπότε αυτό δηλώνεται για όλα τα κανάλια.

- Η θύρα στην οποία αντιστοιχεί:

Η θύρα μπορεί να δηλωθεί είτε με έναν αριθμό, είτε με το όνομα της. Στη συγκεκριμένη περίπτωση συμφέρει να δηλωθεί με έναν αριθμό. Συγκεκριμένα για την είσοδο χρησιμοποιείται η έκφραση  $(j-1)$ . Αυτή παίρνει την κατάλληλη τιμή για τον κατάλληλο αισθητήρα, επειδή η αρίθμηση των θυρών ξεκινάει από το μηδέν. Συγκεκριμένα η αντιστοίχιση είναι:

Αισθητήρας 1 -> Θύρα 0

Αισθητήρας 2 -> Θύρα 1

Αισθητήρας 3 -> Θύρα 2

Αισθητήρας 4 -> Θύρα 3

- Τον τύπο μέτρησης που θέλουμε να πραγματοποιήσει:  
Η κάρτα BNC 2110 λειτουργεί μόνο με τάση, οπότε αυτή θα χρησιμοποιηθεί για το στήσιμο του πειράματος.

## 7.β Δήλωση στοιχείων Πειράματος

Αρχικά το πρόγραμμα ζητάει πόσους αισθητήρες θέλει ο χρήστης να χρησιμοποιήσει. Αν ο χρήστης ζητήσει περισσότερους από τέσσερις, το πρόγραμμα του ζητάει να ξαναβάλει μία τιμή από έναν μέχρι τέσσερις. Στην συνέχεια δημιουργείτε ένα μητρώο με διαστάσεις ένα επί όσους αισθητήρες ζητάει ο χρήστης. Σε αυτό αποθηκεύονται τα κανάλια που ορίστηκαν. Για την είσοδο το μητρώο έχει όνομα  $ch_i$  και για την έξοδο  $ch_o$ . Ο αριθμός τέσσερα επιλέχτηκε ως ανώτατο όριο διότι μέχρι τόσοι αισθητήρες μπορούν να συνδεθούν

```
%% Set Session Rate and Duration
prompt=' set sample rate: ';
s.Rate = input(prompt);

prompt=' set Duration: ';
Duration = input(prompt);
```

Στην συνέχεια ο χρήστης δηλώνει τις παραμέτρους του πειράματος. Συγκεκριμένα δηλώνει τον αριθμό των δειγμάτων που θα στέλνονται από τον κώδικα, καθώς και διάρκεια εκτέλεσης. Το sample rate ορίζεται ως παράμετρος του session, ενώ η διάρκεια σαν απλή μεταβλητή.

## 7.γ Δημιουργία διέγερσης

```
%% create output signal
prompt = 'What kind of stimulation do you want? (1=sinusoidal, 2=triangular, 3=white noise): ';
k= input(prompt);
output = linspace(0,Duration,s.Rate*Duration);

prompt = 'set amplitude in volts: ';
amplitude=input(prompt);

prompt = 'set frequency: ';
f=input(prompt);

if k==1;

outputSignal(:,1) = amplitude*sin(2*pi*f*output);

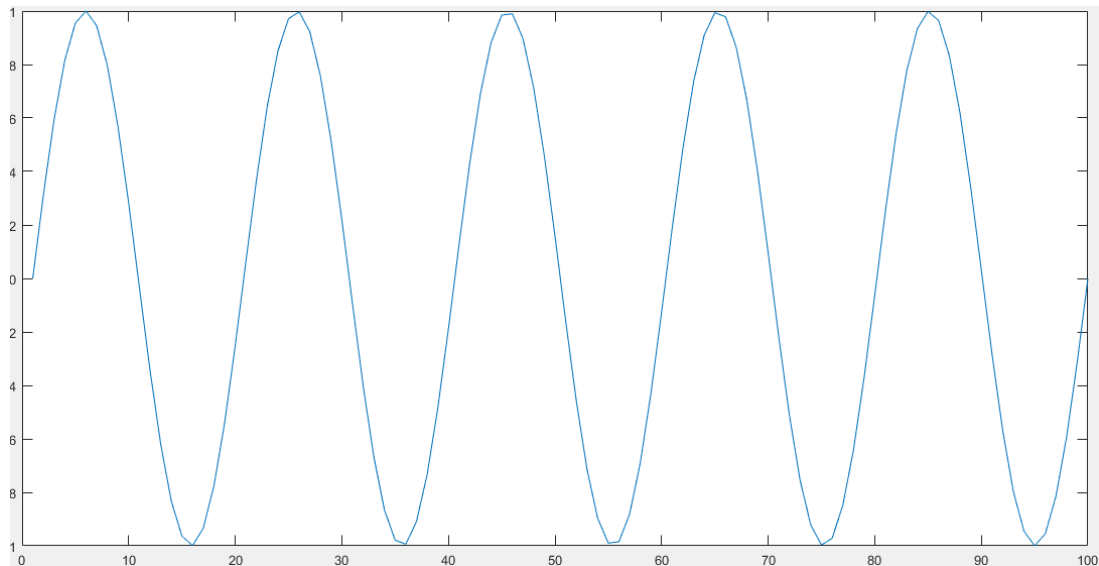
elseif k==2;
outputSignal(:,1)= amplitude*2/pi*asin(sin(2*pi*f*output));

elseif k==3;
outputSignal(:,1)=amplitude*rand([Duration*dq.Rate 1]);
outputSignal(:,1)=2*(outputSignal(:,1)-max_volt/2);

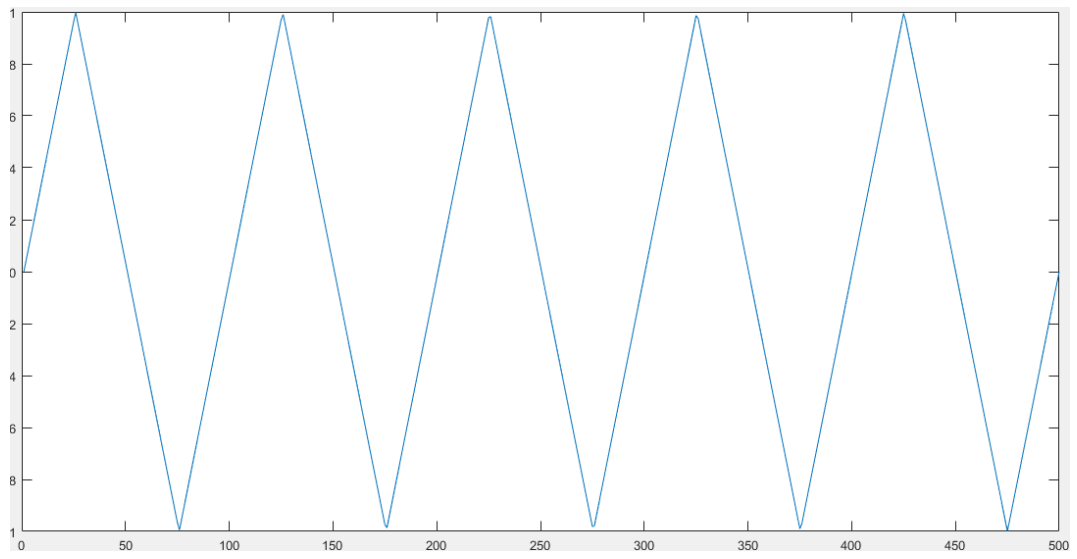
end
```

Ύστερα ο χρήστης δημιουργεί το σήμα που χρειάζεται να σταλθεί από τον κώδικα. Η εντολή `linspace` δημιουργεί ένα πλήθος τιμών ισόποσα χωρισμένες, ανάμεσα σε δύο τιμές. Το πλήθος καθώς και τις ακραίες τιμές ορίζονται από τον χρήστη. Στην συγκεκριμένη περίπτωση θέλουμε το πλήθος να είναι όσα και τα σήματα που θα στέλνουμε, ενώ οι ακραίες τιμές να είναι το 0 και το πλήθος των σημάτων. Αυτό που δημιουργείται είναι οι χρονικές στιγμές για τις οποίες θα στέλνεται το σήμα. Στην συνέχεια δηλώνεται το είδος της διέγερσης, μαζί με τη μέγιστη τιμή της τάσης καθώς και την συχνότητα του σήματος της διέγερσης. Ο χρήστης μπορεί να επιλέξει ανάμεσα σε τρεις διεγέρσεις :

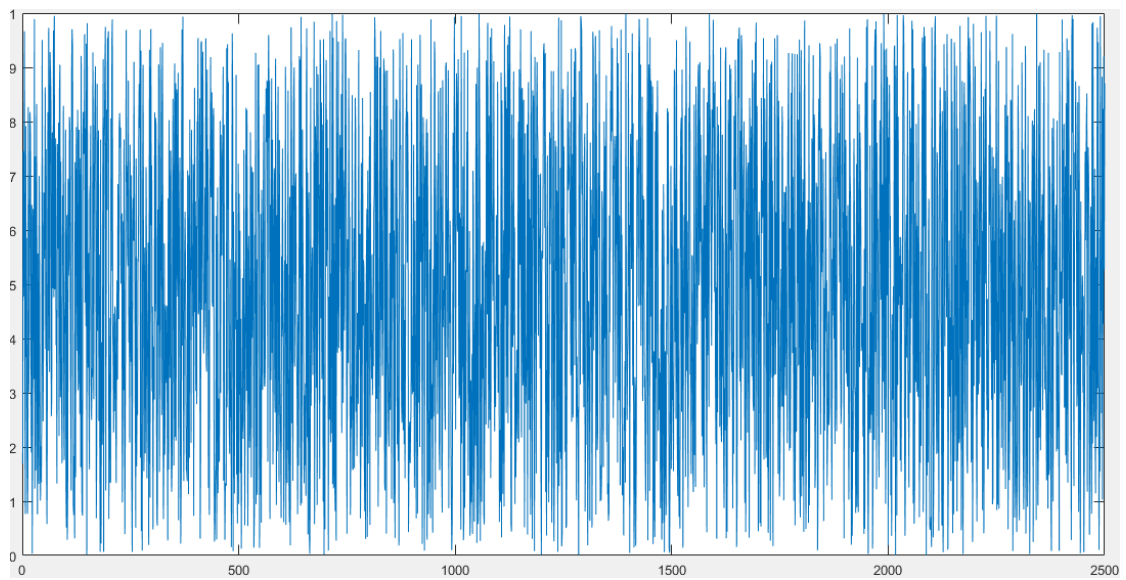
- Ημιτονοειδής μορφή (επιλέγοντας την τιμή 1)  
Χρησιμοποιώντας της παραμέτρους τις οποίες όρισε ο χρήστης παραπάνω, ορίζεται η συνάρτηση, η οποία είναι το σήμα που θα στείλουμε.
- Τριγωνική μορφή (επιλέγοντας την τιμή 2)  
Ορίστηκε η συνάρτηση η οποία αντιστοιχεί στην τριγωνική μορφή.
- White noise (επιλέγοντας την τιμή 3)  
Για την παραγωγή του white noise χρησιμοποιήθηκε η εντολή rand. Αυτή η εντολή όμως δίνει τιμές από μηδέν μέχρι ένα. Για αυτό το λόγο πολλαπλασιάζεται με την μέγιστη τιμή η οποία ορίστηκε από τον χρήστη και ξανά με δύο. Στην συνέχεια αφαιρείται η μέγιστη τιμή για να παραχθούν τιμές ανάμεσα στη μέγιστη τιμή και μείον μέγιστη τιμή.



Γράφημα 5.1 : Παράδειγμα διέγερσης ημιτονοειδούς μορφής



*Γράφημα 5.2 : Παράδειγμα διέγερσης τριγωνικής μορφής*



*Γράφημα 5.3 : Παράδειγμα διέγερσης μορφής white noise*

## 7.δ Εισαγωγή και εξαγωγή δεδομένων

```
% Queue the output data

queueOutputData(s,outputSignal);

fid1=fopen('log.bin','w');

lrh=addlistener(s,'DataRequired',@(src,event)... src.queueOutputData(outputData));

lah=addlistener(s,'DataAvailable',@(src,event)Matrix(src,event,fid1))
```

Η εντολή `queueOutputData` είναι αυτή η οποία προετοιμάζει την εκτέλεση του πειράματος. Αυτή στέλνει τις τιμές για την διέγερση στην κάρτα NI BNC 2110, για να σταλθούν στον ταλαντωτή όταν δοθεί η εντολή. Παράμετροι για αυτή την εντολή είναι το `session` το οποίο εκτελείτε και οι τιμές που θέλουμε να στείλουμε. Έπειτα, με την εντολή `fopen`, δημιουργείτε ένα αρχείο στο οποίο θα γράφονται τα δεδομένα ταυτόχρονα ενώ λαμβάνονται. Παράμετροι είναι του αρχείου και το 'w', το οποίο δηλώνει ότι στο αρχείο αυτό πρόκειται να γραφτούν δεδομένα, καθώς και το όνομα του αρχείου.

Στην συνέχεια χρησιμοποιούνται `listeners`. Τα `listeners` βεβαιώνουν την σωστή λειτουργία για την έξοδο και για την είσοδο. Συγκεκριμένα το πρώτο `listener` σιγουρεύει πως όλες οι τιμές έχουν σταλθεί στην κάρτα, ενώ το δεύτερο διατηρεί τους αισθητήρες ενεργούς μέχρι να τελειώσει το πείραμα. Επιπλέον αποθηκεύει τις τιμές σε ένα αρχείο, ώστε να τις διαβάσει από αυτό μετά. Τα `listeners` είναι απαραίτητα διότι ένα `session` είναι προγραμματισμένο από την MATLAB να λειτουργεί για μόνο ένα δευτερόλεπτο. Με την χρήση αυτών κάνουμε τα μηχανήματα να αλληλεπικοινωνούν για όσο το `session` παραμένει ανοιχτό. Ο τρόπος για να μείνει ανοιχτό για όσο θέλουμε περιγράφεται παρακάτω.

Για το `listener` εξόδου δίνεται η ονομασία `lrh`. Η πρώτη παράμετρος είναι το `session` το οποίο σχετίζεται με το `listener`. Η δεύτερη είναι η 'DataRequired'. Στόχος της είναι



να διατηρήσει ενεργή την εντολή `queueOutputData` όσο χρειάζεται ώστε να σταλθούν όλα τα δεδομένα. Τέλος ορίζεται η μεταβλητή η οποία περιέχει τις τιμές οι οποίες επιθυμούμε να σταλθούν.

Όσο αφορά το listener σχετικό με την είσοδο των τιμών του έχει δοθεί η ονομασία `lah`. Όπως και στο `lrh`, πρέπει πρώτα να οριστεί το `session` το οποίο αφορά το συγκεκριμένο listener. Στη συνέχεια ορίζεται η εντολή παράμετρος `'DataAcquired'`, ώστε να λαμβάνει τιμές καθ' όλη τη διάρκεια που εκτελείτε το πείραμα, και στο τέλος δηλώνεται η συνάρτηση `Matrix`.

```
function Matrix(src, evt, fid)

data=evt.Data;
data=data';
fwrite(fid, data, 'double');

end
```

Οι αποθήκευση των τιμών γίνεται μέσω της συνάρτησης `Matrix`. Η δημιουργία της έγινε σε διαφορετικό αρχείο, διότι η έκδοση της `MATLAB` που χρησιμοποιείται (2013b) δεν υποστηρίζει την δημιουργία συναρτήσεων στο ίδιο αρχείο. Στόχος αυτής της συνάρτησης είναι να αποθηκεύει τις τιμές που λαμβάνει από τους αισθητήρες. Αυτό το καταφέρνει καταγράφοντάς τις σε ένα αρχείο, και πιο συγκεκριμένα στο αρχείο που άνοιξε με την εντολή  `fopen`. Ο λόγος που δεν τις αποθηκεύει απευθείας σε ένα μητρώο είναι ότι δεν είναι δυνατή η απευθείας αποθήκευση τιμών από μία callback function όπως αυτή. Στην συνάρτηση αυτή θα χρειαστεί να δηλωθεί οι παραμέτρους `src`, `evt` και το αρχείο στο οποίο θα γραφτούν οι τιμές. Η καταγραφή των τιμών γίνεται με την εντολή  `fwrite`. Ακόμη έχει προγραμματιστεί οι εντολές να καταγράφονται με διπλή ακρίβεια.

Μία παρατήρηση όσο αφορά και στις δύο περιπτώσεις των listeners, όσο και στην συνάρτηση, είναι ότι και στις δύο περιπτώσεις εμφανίζεται η σύνταξη `@(src,evt)`. Η ένδειξη `src` περνάει σαν πληροφορία το αρχείο από το καλείται, ενώ το `evt` το γεγονός για το οποίο καλείται. Δηλαδή στο συγκεκριμένο πείραμα είναι η είσοδος και έξοδος των δεδομένων.

```
s.IsContinuous = true;

startBackground(s);

pause(Duration);

stop(s);
delete(lah);
delete(s);

fclose(fid1);
fid2=fopen('log.bin','r');
data=fread(fid2,'double');
fclose(fid2);
```

Ο χρόνος για τον οποίο θα εκτελείται το πείραμα καθορίζεται από τις δύο επόμενες εντολές. `s.IsContinuous=true` διατηρεί το session ενεργό, μέχρι να δοθεί εντολή από τον χρήστη. Η εντολή `pause(Duration)` σταματά τον κώδικα να τρέχει καινούριες εντολές. Παρόλα αυτά συνεχίζει να λαμβάνει τιμές και να προκαλεί τη διέγερση. Ο χρόνος που γίνεται αυτή η παύση είναι αυτός που έχει οριστεί από τον χρήστη για την διάρκεια εκτέλεσης του πειράματος. Μόλις τελειώσει αυτός ο χρόνος εκτελείτε η εντολή `stop(s)`, η οποία σταματάει το session να τρέχει, οπότε σταματάει και το πείραμα, ενώ διαγράφονται τα listeners, καθώς δεν χρειάζονται άλλο.

Οι τιμές είναι αποθηκευμένες σε ένα αρχείο το οποίο όμως δεν επιτρέπει την επεξεργασία τους. Οπότε πρέπει να διαβαστούν από το πρόγραμμα και να αποθηκευτούν σε μια μορφή η οποία να επιτρέπει την επεξεργασία τους. Το αρχείο, σε αυτό το σημείο του κώδικα είναι ανοιχτό, αλλά μπορούν να γραφτούν τιμές σε αυτό, και όχι το αντίθετο, καθώς ανοίχτηκε με την ένδειξη 'w'. Για αυτό τον λόγο πρέπει να κλείσει και να ξανανοίξει με την ένδειξη 'r'. Στην συνέχεια διαβάζονται οι τιμές μέσω της εντολής `fread` και αποθηκεύονται σε μία μεταβλητή `data`, με διπλή ακρίβεια. Τέλος το αρχείο κλείνει διότι δεν το χρειάζεται άλλο.

```

%% Sensitivity

prompt = 'Set sensor sensitivity (mV/g): ';
sens=input(prompt);
g= 9,81; %m/s^2
data=data*g/sens; %convert to acceleration (SI)

```

Όμως, οι τιμές οι οποίες έχουν ληφθεί δεν είναι σωστές. Για την ακρίβεια το πρόγραμμα έχει λάβει τα δεδομένα σε τάση. Για την μετατροπή στην επιθυμητή μορφή, η οποία είναι η επιτάχυνση, χρειάζεται να γνωρίζουμε την ευαισθησία του αισθητήρα. Η μετατροπή σε επιτάχυνση πρέπει να γίνει χειροκίνητα, καθώς το μηχάνημα Kistler power supply δεν μπορεί να αντιληφθεί την ευαισθησία και να κάνει την μετατροπή από μόνο του. Η πληροφορία που μας προσφέρει η συγκεκριμένη τιμή είναι το sensitivity του επιταχυνσιόμετρου. Η ευαισθησία για όλα τα επιταχυνσιόμετρα που χρησιμοποιήθηκαν είναι η ίδια και ίση με 492 mV/g. Για την περίπτωση όμως που τοποθετηθούν άλλοι αισθητήρες, το πρόγραμμα έχει σχεδιαστεί ώστε να ζητάει από τον χρήστη πληκτρολογήσει την ευαισθησία των επιταχυνσιομέτρων.

## 7.δ Fast Fourier Transform

Η MATLAB κάνει εύκολη την χρήση της με την εντολή fft(). Παρακάτω εξηγούνται ο ορισμός των παραμέτρων καθώς και η επεξεργασία που χρειάζεται να κάνουμε για να πάρουμε το κατάλληλο αποτέλεσμα από την συνάρτηση.

```

%% Fast Fourier Transform

Fs = s.Rate;      % Sampling frequency
T = 1/Fs;        % Sampling period
L = s.Rate*Duration; % Length of signal
t = (0:L-1)*T;   % Time vector

for j=1:i

```

```

Y = fft(data(:,1));

P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);

f = Fs*(0:(L/2))/L;
plot(f,P1)
title('Single-Sided Amplitude Spectrum of X(t)')
xlabel('f (Hz)')
ylabel('|P1(f)|')

end

```

Στην αρχή πρέπει να οριστούν κάποιες παράμετροι. Αυτές είναι:

Fs: Ρυθμός με τον οποίο λαμβάνονται οι τιμές

T: Χρονική περίοδος με την οποία λαμβάνονται οι τιμές

L: Συνολικός αριθμός τιμών

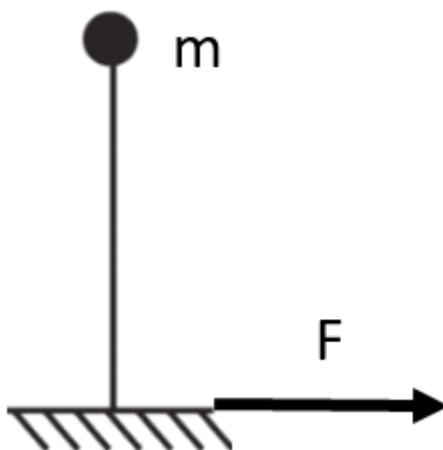
t: Σύνολο χρονικών στιγμών που λαμβάνονται οι τιμές

Στη συνέχεια γίνεται η διαδικασία για την Fast Fourier transform, με την εντολή `fft()`. Το αποτέλεσμα όμως που σημειώνεται δεν είναι το σωστό. Συγκεκριμένα το μητρώο περιέχει τις τιμές για θετικές συχνότητες, αλλά και για αρνητικές. Μία από τις δύο είναι περιττές, οπότε χρειάζεται να κρατήσουμε μόνο ένα από τα δύο πεδία. Επιπλέον οι τιμές είναι οι μισές από ότι είναι στην πραγματικότητα, για αυτό πρέπει να τις διπλασιάσουμε (εκτός από τις ακραίες τιμές). Τέλος ορίζεται τις τιμές τις συχνότητας που αντιστοιχούν σε κάθε επιτάχυνση και δημιουργείτε μια γραφική παράσταση όπου προβάλλονται τα αποτελέσματα.

## 8. Πειραματική διαδικασία

### 8.α. Διάταξη Πειράματος

Σε αυτό το κεφάλαιο περιγράφεται η διαδικασία η οποία πραγματοποιήθηκε για την εκτέλεση του πειράματος. Επιπλέον παρουσιάζονται οι τιμές οι οποίες δόθηκαν στο σύστημα καθώς και τα αποτελέσματα αυτού.



Η κατασκευή πάνω στην οποία πραγματοποιείται το πείραμα είναι όμοια με την παραπάνω εικόνα. Είναι ένας πρόβολος ο οποίος πάνω του έχει τοποθετημένο μία μάζα  $m$ . Η διέγερση προκαλείται από την μετατόπιση της βάσης του προβόλου με έναν συγκεκριμένο ρυθμό και ένταση.

Η ιδιοσυχνότητα, είναι χαρακτηριστικό του συστήματος. Στο παράδειγμα της μάζας και της δέσμης, η φυσική συχνότητα καθορίζεται από δύο παράγοντες: την μάζας και την ακαμψία του προβόλου, το οποίο λειτουργεί ως ελατήριο. Για παράδειγμα η μείωση της μάζας και μια πιο άκαμπτη δέσμη αυξάνουν την ιδιοσυχνότητα

Το πρώτο βήμα, αφότου έχει πραγματοποιηθεί η συνδεσμολογία, είναι να ενεργοποιηθούν οι αισθητήρες. Αυτό γίνεται από τον ενισχυτή των αισθητήρων Kistler Power supply/coupler. Πηγαίνοντας στο μενού bias μέσω του κουμπιού menu, γίνεται φανερό το ποια κανάλια που συνδέονται με τα επιταχυνσιόμετρα είναι ενεργά. Στα ενεργά κανάλια εμφανίζεται ένας αριθμός ο οποίος ρυθμίζεται αυτόματα από τον ενισχυτή, ενώ στα ανενεργά εμφανίζεται η τιμή off. Με τα κουμπιά τα οποία έχουν την ένδειξη με τα βέλη μπορούν να ενεργοποιηθούν τα κανάλια που χρειάζονται.

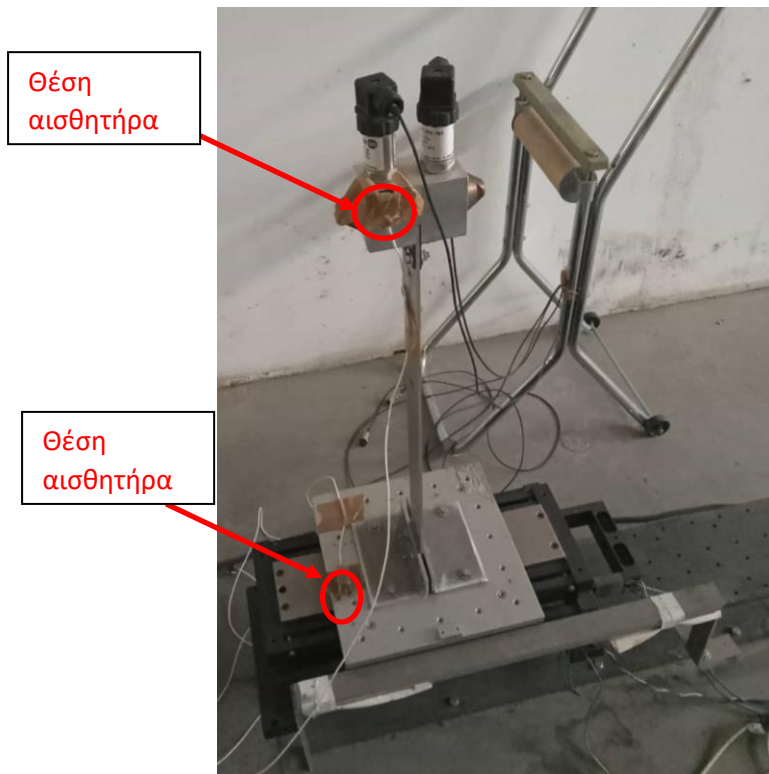


Εικόνα 7.1: Ενεργοποίηση καναλιών.

## 8.β Εκτέλεση πειράματος

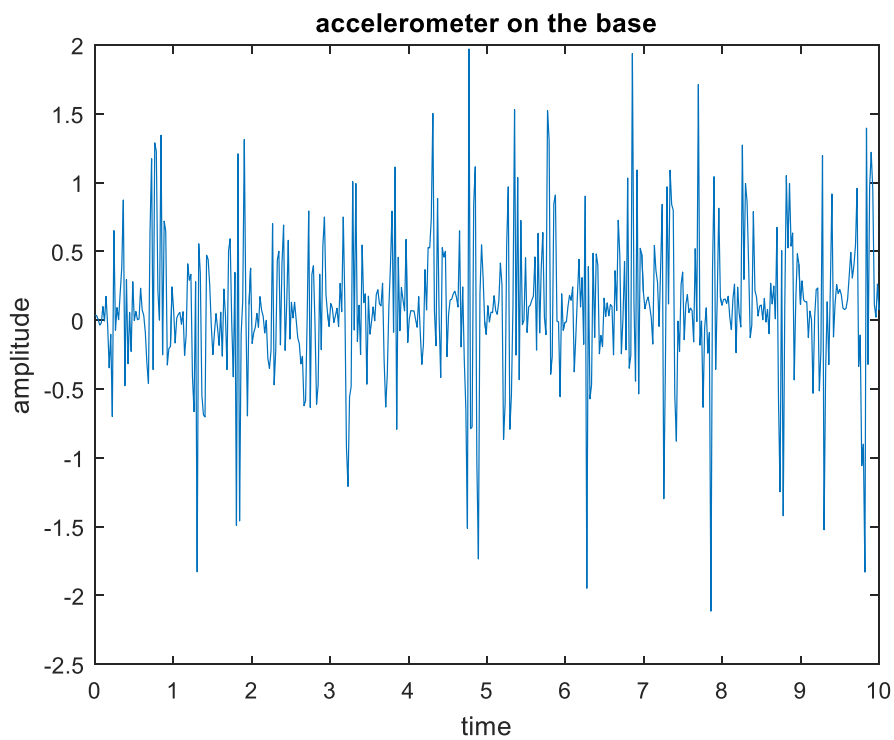
Για την εκτέλεση του πειράματος χρησιμοποιήθηκαν 2 επιταχυνσιόμετρα. Το ένα ήταν δεμένο στην βάση του προβόλου, ενώ το δεύτερο ήταν στην τοποθετημένο στην κορυφή. Ο αισθητήρας την βάση χρησιμεύει για να μεταφραστεί η διέγερση, ενώ ο άλλος για να βρεθεί η συνάρτηση μεταφοράς. Τα δεδομένα για την διέγερση που προκαλείται, αλλά και για την λήψη των δεδομένων είναι.

Δείγματα/ Δευτερόλεπτο	100
Τάση	1 Volt
Συχνότητα	1 Hz
Τύπος Διέγερσης	Τριγωνική
Διάρκεια	10 sec

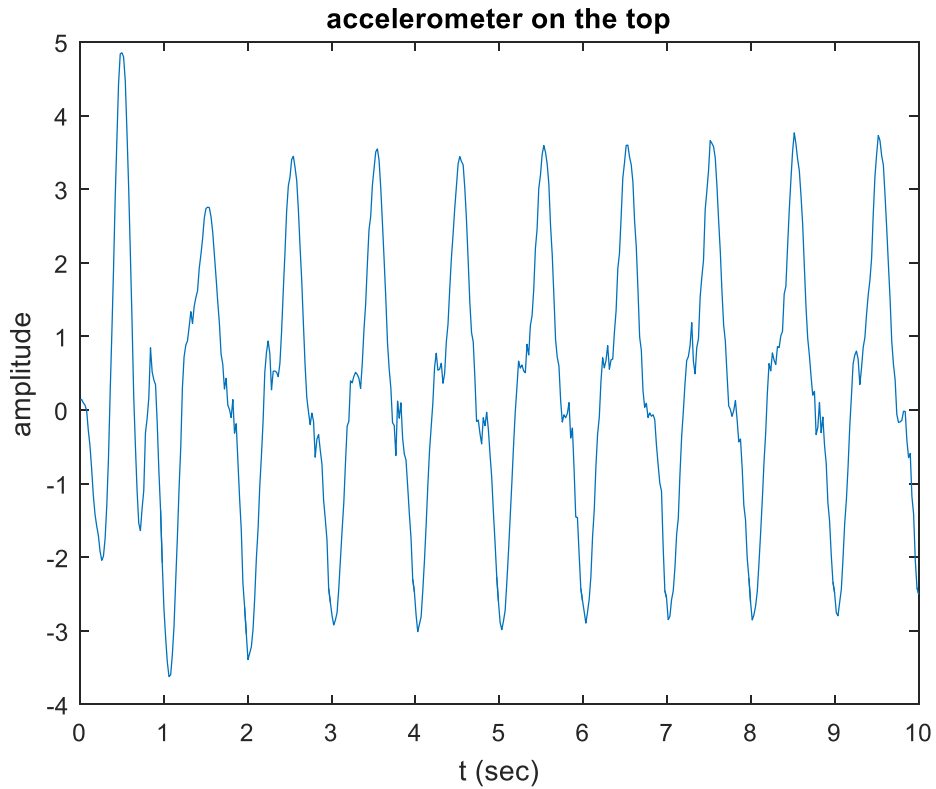


Εικόνα 8.1: Τοποθέτηση αισθητήρων

Μετά από την εκτέλεση του πειράματος τα αποτελέσματα που λαμβάνονται από τους αισθητήρες παρουσιάζονται στις παρακάτω γραφικές παραστάσεις.

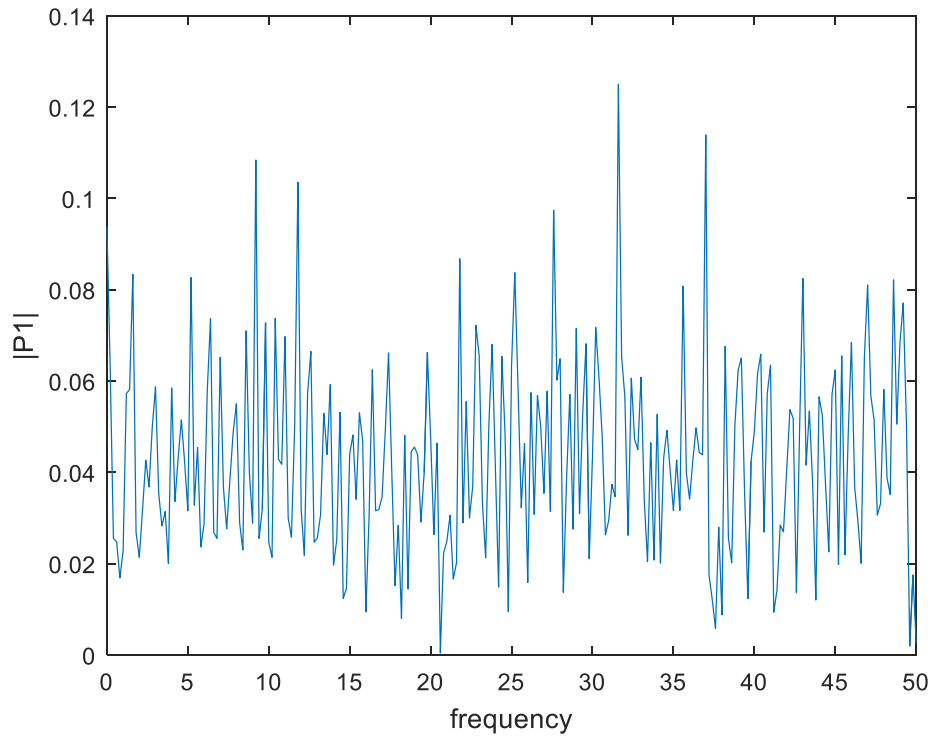


Γράφημα 8.1: Αποτελέσματα αισθητήρα βάσης



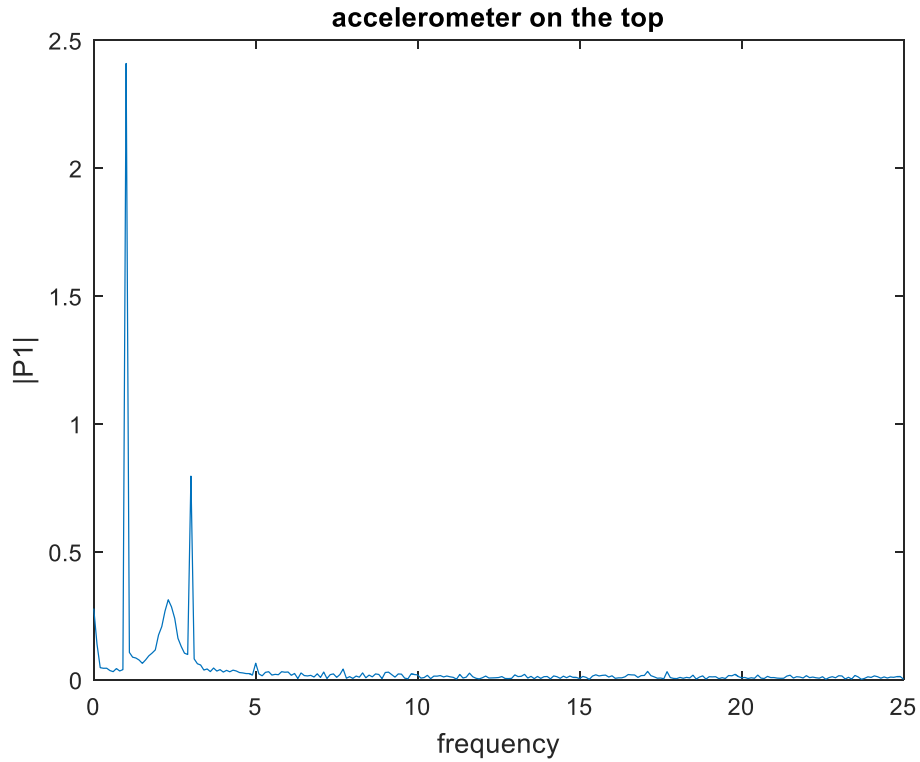
Γράφημα 8.2: Αποτελέσματα αισθητήρα κεφαλής

και μετά από τον μετασχηματισμό Fourier, είναι:



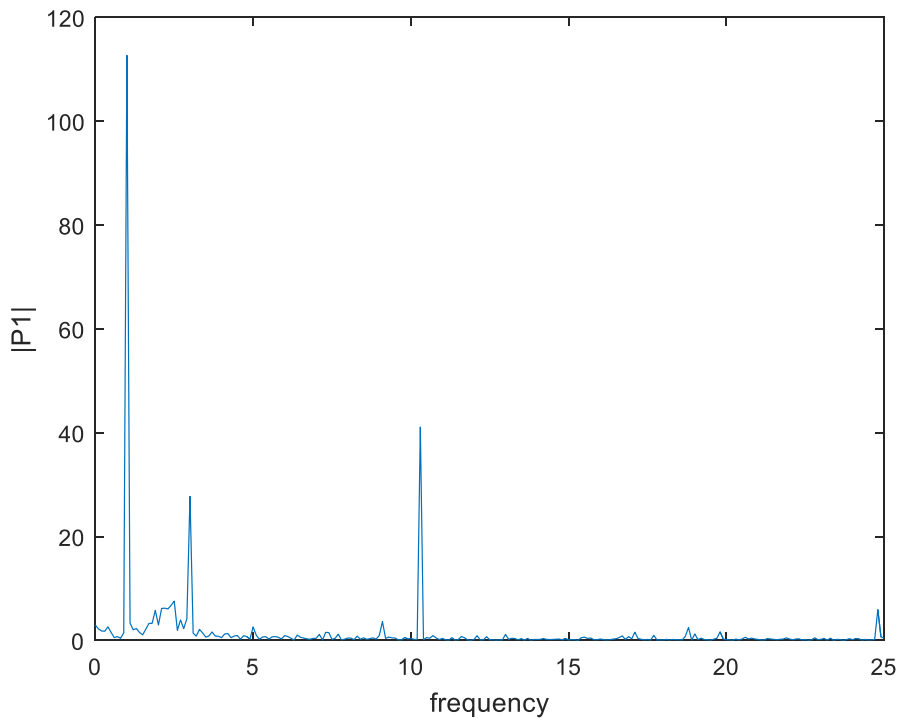
Γράφημα 8.3: Αποτελέσματα μετασχηματισμού Fourier αισθητήρα βάσης





Γράφημα 8.4: Αποτελέσματα μετασχηματισμού Fourier αισθητήρα κεφαλής

Και με την διαίρεση των δύο συναρτήσεων από τον μετασχηματισμό Fourier βγαίνει η συνάρτηση κίνησης. Αυτή οποία παρουσιάζεται στην παρακάτω γραφική παράσταση.



Γράφημα 8.5: συνάρτηση μεταφοράς

Από την συγκεκριμένη γραφική παράσταση εμφανίζονται οι ιδιοσυχνότητες. Αυτές μπορούν να βρεθούν κοιτώντας τον άξονα των συχνοτήτων. Στις τιμές συχνότητας στις οποίες εμφανίζονται τοπικά μέγιστα αντιστοιχούν στην ιδιοσυχνότητα του συστήματος. Στην συγκεκριμένη περίπτωση αυτές οι τιμές είναι οι συχνότητες 1.1, 3.1 και 10.4 Hz.

## 9. Σύνοψη

Ο συγκεκριμένος αλγόριθμος που αναπτύχθηκε μπορεί να χρησιμοποιηθεί σε πολλές και διάφορες εφαρμογές. Έχει την δυνατότητα εκτελέσει πειράματα σε ποικίλα κατασκευών, μεγάλων και μικρών, καθώς μπορεί να επικοινωνήσει με πολλές συσκευές, ανάλογα με την χρήση που χρειάζεται να γίνει.

Το συγκεκριμένο πρόγραμμα επιτρέπει όχι μόνο την εκτέλεση πειραμάτων, αλλά και την ανάλυση των αποτελεσμάτων που λαμβάνονται από τον υπολογιστή, δίνοντας την δυνατότητα στον χρήστη να καταλήξει σε σημαντικά συμπεράσματα για τις ιδιότητες της κατασκευής σύντομα και χωρίς κόπο. Επιπλέον, ο κώδικας μπορεί να αναπτυχθεί ακόμη περισσότερο ώστε να μπορεί να γίνει πιο λεπτομερής ανάλυση της κατασκευής.

Στη παρούσα μελέτη δοκιμαστικέ ο κώδικας σε έναν πρόβολο, ο οποίος ήταν συγκρατημένος σε μια συσκευή που μπορούσε να προκαλέσει διέγερση. Πάνω του ήταν τοποθετημένοι δύο αισθητήρες από τους οποίους έγινε λήψη της επιτάχυνσης του προβόλου σε δύο σημεία. Συγκεκριμένα στη βάση και στην κορυφή. Με αυτές τις τιμές, και μετά από μία ανάλυση Fourier που ακολούθησε, βρέθηκε η συνάρτηση μεταφοράς του προβόλου, καθώς και οι ιδιοτιμές του.

## 10. Βιβλιογραφία

1. Michael Cerna and Audrey F. Harve, The Fundamentals of FFT-Based Signal analysis and measurement
2. MATLAB the language of technical Computing Version 7
3. Mechanical vibration and Shock- Mechanical mounting of accelerometers ISO 5348 (2021)
4. NI-DAQmx (2022)
5. Simple and Practical Algorithm for Sparse Fourier Transform by Haitham Hassanieh, Piotr Indy, Dina Katabi and Eric Price
6. Data translation DAQ Adaptor for MATLAB (2016)
7. Fourier Series and Fourier Transforms, EECS (6.082), MIT Fall 2006 Lectures 2 and 3
8. Daq in MATLAB Hans-Petter Halvorsen (2012)
9. J. W. Cooley and J. W. Tukey, An algorithm for the machine calculation of complex Fourier series. *Moth. of Comput.*(1965)
10. S. C. Liu and L.W. Fagel, A Fast Fourier transform approach to earthquake soil-structure interaction problems (1971)
11. James w. Cooley, Peter a. W. Lewis, and Peter D. Welch, The Fast Fourier Transform and Its Applications
12. S. Bilawchuk, and K.R Fyfe, Multichannel data acquisition and signal processing using matlab (2003)

## 11. Παράρτημα

### Κύριος Κώδικας

```
% clear workspace

clear
close all
% Discover Devices

devices = daq.getDevices;
devices(1)
%% Create a session

s = daq.createSession('ni');
%% Ask how many chanelns does the user need

prompt = 'How many sensors do you want?: ';
i = input(prompt);

while (i<1 || i>4)
prompt = 'Invalid value. You can only use from 1 to 4 sensors.How many sensors do you want?: ';
i = input(prompt);
end

%% create chanelns

%input
for j=1:i;
ch_i(j)= addAnalogInputChannel(s,'Dev1', (j-1), 'Voltage');
end

%output
ch_o=addAnalogInputChannel(s, 'Dev1', 1, 'Voltage');
%% Set Session Rate and Duration
prompt=' set sample rate: ';
s.Rate = input(prompt);
```

```

prompt=' set Duration: ';
Duration = input(prompt);
%% create output signal
prompt = 'What kind of stimulation do you want? (1=sinusoidal, 2=triangular, 3=white noise): ';
k= input(prompt);
output = linspace(0,Duration,s.Rate*Duration);

prompt = 'set amplitude in volts: ';
amplitude=input(prompt);

prompt = 'set frequency: ';
f=input(prompt);

if k==1;

outputSignal(:,1) = amplitude*sin(2*pi*f*output);

elseif k==2;
outputSignal(:,1)= amplitude*2/pi*asin(sin(2*pi*f*output));

elseif k==3;
outputSignal(:,1)=amplitude*rand([Duration*dq.Rate 1]);
outputSignal(:,1)=2*(outputSignal(:,1)-max_volt/2);

end
% Queue the output data

queueOutputData(s,outputSignal);

fid1=fopen('log.bin','w');

lrh=addlistener(s,'DataRequired',@(src,event)... src.queueOutputData(outputData));

lah=addlistener(s,'DataAvailable',@(src,event)Matrix(src,event,fid1))
s.IsContinuous = true;

startBackground(s);

```

```

pause(Duration);

stop(s);
delete(lah);
delete(s);

fclose(fid1);
fid2=fopen('log.bin','r');
data=fread(fid2,'double');
fclose(fid2);
%% Sensitivity

prompt = 'Set sensor sensitivity (mV/g): ';
sens=input(prompt);
g= 9,81; %m/s^2
data=data*g/sens; %convert to acceleration (SI)
%% Fast Fourier Transform

Fs = s.Rate;      % Sampling frequency
T = 1/Fs;        % Sampling period
L = s.Rate*Duration; % Length of signal
t = (0:L-1)*T;   % Time vector

for j=1:i

Y = fft(data(:,1));

P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);

f = Fs*(0:(L/2))/L;
plot(f,P1)
title('Single-Sided Amplitude Spectrum of X(t)')
xlabel('f (Hz)')
ylabel('|P1(f)|')

end

```

## Συνάρτηση

```
function Matrix(src, evt, fid)

data=evt.Data;
data=data';
fwrite(fid, data, 'double');

end
```