

AN ABSTRACT OF THE THESIS OF

Thanat Jitpraphai for the degree of Master of Science in Mechanical Engineering presented on June 11, 1997. Title: Model Based Visualization of Vibrations in Mechanical Systems.

Abstract approved: _____
Redacted for Privacy
||| Swavik A. Spiewak

To visualize vibrations in mechanical systems, *e.g.*, machine tools, their movements are measured by means of suitable sensors. The signals from these sensors are processed and displayed as animated pictures on a computer screen.

Accelerometers have been chosen as the most suitable sensors for this purpose. Their main advantages include small size, wide sensitivity range and frequency bandwidth. In addition, accelerometers measure signals with reference to the Earth, so they do not require stable fixtures such as used with cameras or lasers.

The visualization methodology involves nine accelerometers attached to a mechanical component, *e.g.*, a dynamometer's platform. Vibration signals were acquired using a data acquisition (DAQ) system which is controlled by a LabVIEW[®]-based program. These signals are processed to suppress errors and convert acceleration into generalized coordinate that describes motion of the visualized component as a rigid plate's movement in 3-D space.

The animation is accomplished by displaying a time series of pictures representing instantaneous position of the plate. The animation program employs homogenous coordinate transformation to draw 3-D 'wireframe' pictures. Since various errors distort the measured signals, the animated movement may be inaccurate. The knowledge of a

mathematical model of the system whose vibrations are animated allows detection and suppression of distortions. For this purpose, the signals measured from the actual dynamic system are compared with the signals simulated by the system's model subjected to the same excitation as the actual system. Discrepancies between the actual and simulated signals are detected. They are analyzed to identify possible sources and forms of distorting signals. As the next step, the measured (actual) signals are corrected by removing estimated distortions.

A methodology and software package capable of performing all functions necessary to implement the visualization of vibration have been developed in this research using LabVIEW[®] programming environment. As compared with commercial software for experimental modal analysis, the most distinctive feature of the developed package is improved accuracy achieved by applying concepts utilized in control theory, such as modeling of multi-input-multi-output (MIMO) systems and on-line system identification for the model development and correction of signals.

© Copyright by Thanat Jitpraphai

June 11, 1997

All Rights Reserved

MODEL BASED VISUALIZATION OF VIBRATIONS IN MECHANICAL SYSTEMS

by

Thanat Jitpraphai

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented June 11, 1997
Commencement June 1998

Master of Science thesis of Thanat Jitpraphai presented on June 11, 1997

APPROVED:

Redacted for Privacy

Major Professor, representing Mechanical Engineering

Redacted for Privacy

Head of Department of Mechanical Engineering

Redacted for Privacy

Dean of Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Redacted for Privacy

Thanat Jitpraphai, Author

ACKNOWLEDGMENT

I would like to express my appreciation to my advisor, Dr. Swavik A. Spiewak. I first met Professor Spiewak in his Smart Product and Design class where he showed me the power of computer programs (that have led to the LabVIEW® programs in this thesis). Without the inspiration as well as his advice, encouragement and great effort, this thesis would not have been accomplished. I would also like to thank the students in my research group, Thomas Nickel, Brian Brisbane, Ben Chen and all of my friends in helping me out of many difficulties.

I would like to thank sincerely to my parents, Dr. Phaibul and Dr. Chattaya Jitpraphai as well as my brothers, Peera and Siros, for their encouragement and support for my study.

I would like to thank to my special person, Waranush Sorasuchart, for her care and patience during this work. Thanks also due to all of my Thai-OSU friends who help me in many things.

TABLE OF CONTENTS

| | <u>Page</u> |
|------------------------------------------------------------------------------|-------------|
| CHAPTER 1 | |
| INTRODUCTION | 1 |
| 1.1 Graphical Representation of Vibration | 1 |
| 1.2 Scope of Work | 2 |
| 1.3 Chapter Overview..... | 3 |
| CHAPTER 2 | |
| LITERATURE REVIEW..... | 4 |
| 2.1 Visualization in Vibration Analysis..... | 4 |
| 2.2 System Identification in Vibration Analysis..... | 7 |
| 2.3 Conventional Visualization Software..... | 10 |
| 2.4 Measurement of Signals for Visualization | 12 |
| 2.4.1 Piezoelectric Accelerometer Measurement | 13 |
| 2.4.2 Conventional Low Frequency Accelerometers | 20 |
| 2.4.3 Multi Directional Accelerometer | 24 |
| 2.5 Closure | 25 |
| CHAPTER 3 | |
| MODEL BASED VISUALIZATION OF VIBRATIONS..... | 26 |
| 3.1 An Overview and Definition of the Proposed Visualization Technique | 26 |
| 3.2 Signal Based Vibration Visualization | 28 |
| 3.3 Model Based Vibration Visualization | 29 |
| 3.4 Feasibility Study of the Visualization Enhancement | 35 |

TABLE OF CONTENTS (Continued)

| | <u>Page</u> |
|----------------------------------------------------------------------------------|-------------|
| 3.5 Modeling of the Dynamometer | 37 |
| 3.5.1 Mechanistic Model | 39 |
| 3.5.2 State Space Model | 41 |
| 3.5.3 Transfer Function Model | 43 |
| 3.6 Closure | 44 |
| | |
| CHAPTER 4 | |
| THE SIGNAL BASED VIBRATION VISUALIZATION..... | 45 |
| 4.1 Visualization of Machine Vibrations | 45 |
| 4.1.1 Overview of the Methodology..... | 46 |
| 4.1.2 Introduction to LabVIEW® Programming Environment | 48 |
| 4.2 Data Acquisition System Used in Vibration Visualization | 52 |
| 4.2.1 Anti-aliasing Filtering | 53 |
| 4.2.2 Attenuation of Noise in a Data Acquisition | 53 |
| 4.2.3 Data Acquisition Program | 55 |
| 4.3 Signal Processing in Vibration Visualization | 55 |
| 4.3.1 Conversion to Physical Units and Subtraction of the Average Value | 56 |
| 4.3.2 High-pass Filtering and Double Integration Procedure | 57 |
| 4.4 Coordinate Systems | 59 |
| 4.5 Calculation of the Generalized Coordinates from Experimental Data | 61 |
| 4.6 Animation of the Rigid Body Motion | 64 |
| 4.6.1 Finding Absolute Position of the Reference Corner Point | 65 |
| 4.6.2 Calculating Coordinates of the Plate Corners | 66 |
| 4.6.3 Projection of 3-D Object to a Planer (2-D Screen) | 71 |
| 4.6.4 Changing the Viewpoint | 73 |
| 4.6.5 Drawing a Single 3-D Picture | 75 |

TABLE OF CONTENTS (Continued)

| | <u>Page</u> |
|----------------------------------------------------------------------|-------------|
| 4.6.6 Animation of Generated 3-D Pictures | 76 |
| 4.7 Transformation of Generalized Coordinates in a Rigid Plate | 78 |
| 4.8 Closure | 79 |
| | |
| CHAPTER 5 | |
| EXPERIMENTAL IMPLEMENTATION AND RESULTS | 80 |
| 5.1 Experimental Set Up | 80 |
| 5.2 Data Acquisition and Vibration Visualization Software | 84 |
| 5.3 Verification of the Developed Software Modules | 87 |
| 5.3.1 Experimental Procedure | 87 |
| 5.3.2 Results and Discussion | 88 |
| 5.4 Comparison of Signal Based and Model Based Responses | 91 |
| 5.4.1 Experimental Procedure | 91 |
| 5.4.2 Results and Discussion | 92 |
| 5.5 Closure | 99 |
| | |
| CHAPTER 6 | |
| CONCLUSIONS AND RECOMMENDATIONS | 100 |
| 6.1 Conclusions | 100 |
| 6.2 Recommendations | 101 |
| | |
| BIBLIOGRAPHY | 103 |
| | |
| APPENDICES | 107 |

LIST OF FIGURES

| <u>Figure</u> | <u>Page</u> |
|---------------|----------------------------------------------------------------------------------------------------------------------|
| 2.1 | Equivalent forms of a mechanical model used in vibration analysis5 |
| 2.2 | A 2-DOF model of an automobile6 |
| 2.3 | Vibrating lumped system in two different mode shapes.....7 |
| 2.4 | Obtaining the system's model by means of an identification procedure8 |
| 2.5 | Classification of the system identification techniques.....9 |
| 2.6 | Parametric identification.....10 |
| 2.7 | An animation of vibration on the display of ME Scope™.....11 |
| 2.8 | Six coordinates describing motion of a rigid plate12 |
| 2.9 | Configurations of piezoelectric accelerometers including (a) compression type and (b) shear type.....14 |
| 2.10 | Comparison diagrams of instrument setup between the high and low impedance system.....15 |
| 2.11 | Cross-section diagram of the ICP accelerometer16 |
| 2.12 | Magnitude plot of the FRF of a piezoelectric accelerometer17 |
| 2.13 | Illustration of the impact of a low frequency drift on the displacement obtained from acceleration signal20 |
| 2.14 | Signals obtained by means of a variable capacitance accelerometer and its results from double-integration22 |
| 2.15 | Schematic illustration of a single-element electron tunneling accelerometer.....23 |
| 2.16 | A cross section of a piezoresistive accelerometer24 |
| 2.17 | A cross section of the PiezoBEAM® accelerometer24 |
| 3.1 | Flowchart of model based visualization of vibrations27 |
| 3.2 | A flowchart representation of the methodology developed for the signal based vibration visualization28 |
| 3.3 | Classification of dynamic systems29 |
| 3.4 | The MIMO system30 |
| 3.5 | Block diagram of the model based vibration visualization32 |
| 3.6 | Block diagram of the Comparison and Correction33 |
| 3.7 | An alternative flowchart of the model based vibration visualization35 |

LIST OF FIGURES (Continued)

| <u>Figure</u> | <u>Page</u> |
|---------------|--------------------------------------------------------------------------------------------------------------------------|
| 3.8 | Major components of the dynamometer36 |
| 3.9 | Main research subjects pertaining to model based visualization37 |
| 3.10 | A high speed machine tool38 |
| 3.11 | A simplified model of the machine tool from Fig. 3.10 with the dynamometer installed38 |
| 3.12 | Simplified mechanical model of the dynamometer under consideration40 |
| 4.1 | Components of the generalized coordinate list, \mathbf{d}_G , describing the 'rigid-body' motion of a plate46 |
| 4.2 | Flowchart of the methodology used for the visualization of machine vibrations47 |
| 4.3 | An example program (virtual instrument) in LabVIEW®49 |
| 4.4 | Front panels of LabVIEW® programs developed for the vibration visualization 50 |
| 4.5 | A block diagram of the basic data acquisition system used in this research52 |
| 4.6 | A flowchart of signal processing in vibration visualization55 |
| 4.7 | Diagram of the signal processing procedure59 |
| 4.8 | Coordinate systems used in describing the plate motion60 |
| 4.9 | Locations of nine accelerometers required for the calculation of the generalized coordinates63 |
| 4.10 | A flowchart of the procedure calculating the list of absolute generalized coordinates66 |
| 4.11 | Abbreviations used to designate the plate's corner67 |
| 4.12 | Definition of coordinate transformation matrices69 |
| 4.13 | Application of the homogeneous coordinate transformation for finding coordinates of point A70 |
| 4.14 | Flowchart shows procedure of calculating coordinates of all corners72 |
| 4.15 | Illustration of the projection of a 3-D object on a 2-D planer73 |
| 4.16 | Illustration of the effect from changing the viewpoint on the 2-D picture74 |
| 4.17 | Diagram of coordinate transformation procedure for changing the viewpoint ...75 |
| 4.18 | A flowchart of the entire animation procedures77 |

LIST OF FIGURES (Continued)

| <u>Figure</u> | <u>Page</u> |
|---------------------------------------------------------------------------------------------------------------|-------------|
| 5.1 Schematic diagram of the experimental setup | 81 |
| 5.2 Impact hammer (PCB® type 208B03) used for exciting the dynamometer | 81 |
| 5.3 Locations of nine accelerometers (Kistler® type 8702B25M1) mounted on the dynamometer | 83 |
| 5.4 The data acquisition system | 83 |
| 5.5 Icons and wiring terminals of the major LabVIEW® modules employed in the vibration visualization | 84 |
| 5.6 A simplified diagram of the LabVIEW® visualization programs using modules shown in Fig. 5.5..... | 86 |
| 5.7 Definitions of characteristic time instances referred to Table 5.1 and 5.2 | 89 |
| 5.8 Graphs show results from the test number 5 | 93 |
| 5.9 Graphs show results from the test number 6 | 94 |
| 5.10 Graphs show results from the test number 7 | 95 |
| 5.11 Graphs show results from the test number 8 | 96 |
| 5.12 Illustration of the flexible mode of platform's vibration from the test number 5 | 98 |

LIST OF TABLES

| <u>Table</u> | <u>Page</u> |
|---------------------------------------------------------------------------------------------|-------------|
| 2.1 Effect of time constant on the error in measuring various transient responses .. | 18 |
| 4.1 Coordinates of the corners and equations used for calculations | 67 |
| 4.2 Order of corner plotting for creating a complete 3-D rectangular plate | 76 |
| 5.1 Descriptions of the test procedures used in the experiment | 88 |
| 5.2 Characteristic locations of the dynamometer's platform obtained experimentally | 90 |
| 5.3 Description of procedures in the experiment | 92 |

LIST OF APPENDICES

| | <u>Page</u> |
|-------------------------------------------------------------------------|-------------|
| Appendix A Experiment Specifications..... | 108 |
| Appendix B Parameters of the Dynamometer's Model..... | 110 |
| Appendix C MATLAB® Program Used in the Experiment..... | 112 |
| Appendix D Descriptions of the LabVIEW® Visualization Programs | 118 |
| Appendix E Data Management in the LabVIEW® Visualization Programs | 127 |
| Appendix F Block Diagrams of the LabVIEW® Visualization Programs | 132 |

LIST OF APPENDIX FIGURES

| <u>Figure</u> | <u>Page</u> |
|-----------------------------------------------------------------------------------------------------------------------|-------------|
| A.1 Power spectrum density of signal measured from an accelerometer | 108 |
| A.2 Dimensions of the dynamometer used in the experiment in units of mm (the sensing elements are not shown) | 109 |
| A.3 Digital high-pass filter's coefficients used in Eq. 4.4 and 4.6 | 109 |
| D.1 Front panel of Data Acquisition Controller program (DAC) | 118 |
| D.2 Front panel of the signal processor program (SP) | 120 |
| D.3 Front panel of the generalized coordinate calculator program (GCC) | 122 |
| D.4 Front panel of the 3-D animation generator program (AG) | 123 |
| D.5 Dimensions used in the 'center reference' drawing option | 125 |
| D.6 Dimensions used in the 'corner reference' drawing option | 126 |
| E.1 Data system | 127 |

LIST OF APPENDIX TABLES

| | <u>Page</u> |
|-----------------------------------------------------------------|-------------|
| A.1 Descriptions of sensors used in the experiment | 108 |
| E.1 Formats of data used in the visualization programs | 128 |
| E.2 Systeminfo assignment | 129 |
| E.3 Example of the Systeminfo file used in the experiment | 131 |

LIST OF ABBREVIATIONS

| | |
|---------------|------------------------------------|
| C.S. | = Coordinate System |
| CFR | = Characteristic Forced Response |
| DAQ | = Data Acquisition |
| EMA | = Experimental Modal Analysis |
| FLL | = Front Left Lower Corner |
| FLU | = Front Left Upper Corner |
| FRF | = Frequency Response Function |
| FRL | = Front Right Lower Corner |
| FRU | = Front Right Upper Corner |
| G | = Graphical Programming Language |
| ICP | = Integrated Circuit Piezoelectric |
| IIR | = Infinite Impulse Response |
| MBR | = Model Based Response |
| MDOF | = Multi Degree of Freedom |
| MIMO | = Multi Input Multi Output |
| <i>N</i> -DOF | = <i>N</i> Degrees of Freedom |
| ODS | = Operational Deflection Shape |
| RLL | = Rear Left Lower Corner |
| RLU | = Rear Left Upper Corner |
| RRL | = Rear Right Lower Corner |
| RRU | = Rear Right Upper Corner |
| SBR | = Signal Based Response |
| SISO | = Single Input Single Output |
| VI | = Virtual Instrument |

NOMENCLATURE

(XYZ) = global coordinate system comprises of X , Y and Z axes

$(XYZ)_G$ = coordinate system at the plate's center of mass comprises of X_G , Y_G and Z_G axes

$(XYZ)_I$ = instantaneous coordinate system of the plate comprises of X_I , Y_I and Z_I axes

$(XYZ)_R$ = reference coordinate system of the plate comprises of X_R , Y_R and Z_R axes

$(XYZ)_V$ = viewpoint coordinate system of the plate comprises of X_V , Y_V and Z_V axes

A, B, C, D = coefficient matrices of state space equations

AG = "3-D Animation Generator.VI" LabVIEW® program

A_o = amplitude of acceleration applied to an accelerometer

a_c, a_p = accelerations of point C and P , respectively, in $(XYZ)_R$

$a_{i,j}$ = acceleration in i direction at corner j ; $i = x, y$ and z ; $j = C, 1, 2$, and 3

a_x, a_y, a_z = distances between the force application point and G in the X_G, Y_G and Z_G directions, respectively

c = damping coefficient

\mathbf{c} = damping matrix of the spatial model

C = origin of $(XYZ)_I$

C_R = origin of $(XYZ)_R$

Cu = constant matrix converting actual input \mathbf{u} into \mathbf{F}_e

\mathbf{D}_1^2 = 1×4 matrix describes a vector from point 1 to point 2

DAC = "Data Acquisition Controller.VI" LabVIEW® program

$\mathbf{d}_c, \mathbf{d}_p$ = generalized coordinates of point C and P , respectively, with respect to (XYZ)

\mathbf{d}_I = generalized coordinate of point C with respect to $(XYZ)_R$

\mathbf{d}_o = generalized coordinate of point C_R with respect to (XYZ)

\mathbf{D}_T = vector of translational motion of the dynamometer's base

F = vector of the forces acting on the dynamometer's platform

F_e = vector of the forcing function acting on the platform's center of mass

F_i = components of the force vector, $i = x, y$ and z

Fr = vector of the measured force signal

f^* = Nyquist frequency

- f_m = measurement frequency bandwidth of the piezoelectric accelerometer
 f_s = sampling frequency
 G = center of mass
GCC = "Generalized Coordinate Calculator.VI" LabVIEW® program
 $\mathbf{G}_E(s)$ = transfer function matrix of the equivalent system
 G_f = gain of the amplifier in the anti-aliasing filter
 $G_{i,j}(s)$ = element of the transfer function matrix $\mathbf{G}_s(s)$
 $\mathbf{G}_s(s)$ = transfer function matrix of the system
 H_L = FRF of the low impedance piezoelectric accelerometer system
 H_r = rotational magnification factor
 H_t = translational magnification factor
 $\mathbf{I}_{n \times n}$ = n by n identity matrix
 i = index denoting the direction
 $\mathbf{i}_G, \mathbf{j}_G, \mathbf{k}_G$ = unit vectors in X_G, Y_G and Z_G axes, respectively
 $\mathbf{i}_I, \mathbf{j}_I, \mathbf{k}_I$ = unit vectors in X_I, Y_I and Z_I axes, respectively
 J = mass moment of inertia
 j = index denoting the corners of the plate
 k, k_1, k_2 = stiffness coefficient
 \mathbf{k} = stiffness matrix of the spatial model
 k_A = conversion factor
 l, w, h = dimensions of the plate in X_I, Y_I and Z_I directions
 l_1, l_2 = length of the automobile model
 l_G, w_G, h_G = distances from point C to point G in X_I, Y_I and Z_I directions
 l_O, w_O, h_O = distances from point O to point C_R in X, Y and Z directions
 m = mass
 \mathbf{m} = mass matrix of the spatial model
 \mathbf{M} = vector of torques acting on the dynamometer's platform
 M_i = moment acting on the dynamometer's platform; $i = x, y$ and z
 M_{iF} = moment caused by the force \mathbf{F} acting away from G ; $i = x, y$ and z
 N_F, N_R = number of the forward and reverse coefficients, respectively

O = origin of the (XYZ)
 $\mathbf{O}_{n \times n}$ = n by n zero matrix
 P = arbitrary point on a rigid plate
 $\mathbf{q}(t)$ = vector of state variable
 $\mathbf{Q}(s)$ = Laplace transform of $\mathbf{q}(t)$
 \mathbf{r} = position vector of point P from point C
 r_x, r_y, r_z = distances between accelerometers
 R_k = k^{th} reverse coefficient of the digital filter
 \mathbf{R}_r = rotational motion of the dynamometer's base
 s = Laplace variable
 \mathbf{SP} = "Signal Processor.VI" LabVIEW[®] program
 SPF = signal processing function
 s_v = accelerometer sensitivity
 \mathbf{T}_1^2 = homogeneous transformation matrix from C.S.1 to C.S.2
 t_t = measuring time of the transient response of the piezoelectric accelerometer
 TC = time constant of the miniature amplifier in the transducer
 TCI = time constant of the power supply
 $\mathbf{U}(s)$ = Laplace transform of the input $\mathbf{u}(t)$
 $\mathbf{u}(t)$ = input vector
 $\mathbf{U}_m(s)$ = Laplace transform of the measured input $\mathbf{u}_m(t)$
 $\mathbf{u}_m(t)$ = measured input signal
 V_o = amplifier output voltage
 V_r = voltage signal recorded by the data acquisition program
 x', y', z' = coordinates in a different coordinate system
 $x_1[n]$ = n^{th} element of a sequence of the discrete acceleration signal
 $x_2[n]$ = n^{th} element of a sequence of the discrete filtered acceleration signal
 $x_3[n]$ = n^{th} element of a sequence of the discrete velocity signal
 $x_4[n]$ = n^{th} element of a sequence of the discrete filtered velocity signal
 $x_5[n]$ = n^{th} element of a sequence of the discrete displacement signal
 x, y, z = translations of point G relatively to point O , parallel to X, Y and Z axes

x_C, y_C, z_C = translations of point C relatively to point O , parallel to X, Y and Z axes

x_G, y_G, z_G = translations of point G relatively to point O , parallel to X, Y and Z axes

x_P, y_P, z_P = translations of point C relatively to point C_R , parallel to X_R, Y_R and Z_R axes

X_T, Y_T, Z_T = translations of the dynamometer's base in X, Y and Z directions

$Y(s)$ = Laplace transform of the output $y(t)$

$y(t)$ = output vector

$y_E(t)$ = estimated output vector from $G_E(s)$ and $U(s)$

Z = amplitude of the automobile's translation

α = angular acceleration of point P in $(XYZ)_I$

α_i = angular accelerometer component of the vector α ; $i = x, y$ and z

Δt = sampling period

Θ = amplitude of the automobile's rotation

θ, ϕ, ψ = rotations of C.S. $(XYZ)_G$ around X, Y and Z axes, respectively

θ_C, ϕ_C, ψ_C = rotations of C.S. $(XYZ)_I$ around X, Y and Z axes, respectively

θ_P, ϕ_P, ψ_P = rotations of C.S. $(XYZ)_I$ around X_R, Y_R and Z_R axes, respectively

θ_O, ϕ_O, ψ_O = rotations of C.S. $(XYZ)_R$ around X, Y and Z axes, respectively

θ_T, ϕ_T, ψ_T = rotations of the dynamometer's base around X, Y and Z axes, respectively

θ_V, ϕ_V, ψ_V = rotations of C.S. $(XYZ)_V$ around X, Y and Z axes, respectively

$\bar{\omega}$ = angular velocity of point P in C.S. $(XYZ)_I$

$\bar{\omega}_i$ = angular velocity component of the vector $\bar{\omega}$; $i = x, y$ and z

ω = frequency variable

ω_1, ω_2 = natural frequency of an example automobile

ω_n = natural frequency

ω_L, ω_H = low and high limits of the measuring frequency of the accelerometer

Ψ = mode shape

ζ = damping coefficient

MODEL BASED VISUALIZATION OF VIBRATIONS IN MECHANICAL SYSTEMS

CHAPTER 1

INTRODUCTION

1.1 Graphical Representation of Vibration

Graphical representation of vibration aids engineers in analysis of dynamic behavior of mechanical systems. Analysts have better understanding of vibration problems by looking at actual movement of components under consideration. Unlike the vibration analysis based on the finite element method and modal analysis software, the visualization of the actual movement can provide information that is often beyond the estimated display generated by the analytical methods.

The visualization of actual vibrations in mechanical systems is accomplished by measuring the movement of these systems with suitable sensors. Signals from these sensors are processed and displayed as graphical representation of the vibration. To measure signals required for the visualization, many different methods have been developed, such as laser beam scanning schemes, fiber optics based sensors, vision systems, or magnetic sensors. The application of these systems is costly and therefore limited. Accelerometers, on the other hand, are effective with reasonable cost. They also provide advantages including small size, wide sensitivity range and frequency bandwidth. In addition, accelerometers measure the signals with reference to the Earth, so they do not require stable fixtures such as used with cameras or lasers. Therefore, the accelerometers have been chosen as the most suitable sensors in this research.

Algorithms have been developed and implemented as programs for the visualization purpose. In the course of this project numerous problems have emerged

associated with the use of accelerometers. These problems centered around errors in measured signals. These errors were attributed either to the environment or characteristics of the sensors themselves. A suitable method for suppressing the errors has been proposed in this research.

Despite the precautions taken to suppress errors in the signals used for visualization there is no guarantee that the displayed motion accurately represents the actual behavior of tested system. To improve the reliability of visualization a methodology have been proposed that allows the detection and correction of errors. This methodology involves concepts developed in control theory as well as analytical models of systems whose vibrations are visualized.

1.2 Scope of Work

The research discussed in this thesis addresses two major areas and has two objectives. The first objective is the development of a visualization program capable of acquiring acceleration signals that represent vibrations. These signals are processed to obtain three-dimensional movements of mechanical systems and displayed as graphical animation representing the vibrations. This objective is accomplished and presented in this thesis.

The second objective is an enhancement of the above visualization program by utilizing information encapsulated in analytical models of investigated systems to detect and eliminate errors of visualization. The work in this area is only outlined here and recommended as the future development. However, a preliminary experiment was conducted in this thesis to study the feasibility of using analytical models for the purpose of visualization enhancement.

The visualization technique and software developed in the thesis apply to a rigid body motion. In particular, the motion of a plate suspended on four three-dimensional springs is investigated. This plate is one component of a multi-component force sensor (*dynamometer*). A model of this dynamometer is known from previous research (Chung, 1993; Chen, 1996) and is ready to be used in a comparative experiment for the feasibility study of the model based enhancement of visualization.

1.3 Chapter Overview

Visualization by means of vibration analysis is introduced in Chapter 2. System identification technique is also described since it will be used as part of the model development necessary for enhancing the visualization program. Basics of piezoelectric accelerometers dealt within this research are reviewed with respect to their applications and possible errors resulted from their characteristics. Alternative accelerometers are discussed as a suggestion for the research improvement.

In Chapter 3, a concept of the model based enhancement of visualization is introduced. A terminology used in the research is defined at the beginning of the chapter followed by a brief explanation of a methodology of the visualization. Possible extension of work is also delineated. Finally, the mathematical model of a dynamometer employed from previous researches is discussed as this model underlines the proposed extension of work.

In Chapter 4 theory and algorithms used in the visualization program developed in this thesis is explained. Experiments conducted for the program verification are described with results and discussion in Chapter 5. The experiments also involve a comparison between results from the visualization program and those from the mathematical model. Finally, conclusions and a future direction of the research are stated in Chapter 6.

CHAPTER 2

LITERATURE REVIEW

The objectives and application of vibration analysis are discussed and a motivation for the visualization of vibration is presented. System identification techniques as an enhancement of the vibration analysis are also reviewed. Representative visualization techniques and commercial programs are briefly characterized followed by an introductory discussion of the proposed visualization technique. Since piezoelectric accelerometers have been used to measure vibrations of system in this research, the design and performance of these accelerometers is described. Finally, the latest advances in relevant sensor technology, *i.e.*, low frequency accelerometers and multi directional sensors, are reviewed.

2.1 Visualization in Vibration Analysis

Vibration analysis is a study of dynamic behavior of a system. As a rule, the analysis involves an attempt to define mechanical systems by means of mathematical models. The systems are usually described by one of the following model forms: a spatial model, a modal model, and a response model (Ewins, 1984). These three forms are related to each other and one form can be transformed into another as shown in Fig. 2.1.

The spatial model describes the system based on its physical parameters, including masses (m), stiffness (k) and damping coefficients (c). These properties are usually formed in the equation of motion. *The modal model* is obtained by applying 'free vibration analysis' to the spatial model which results in the information given by a set of natural frequencies (ω_n), damping ratios (ζ), and corresponding mode shapes (ψ).

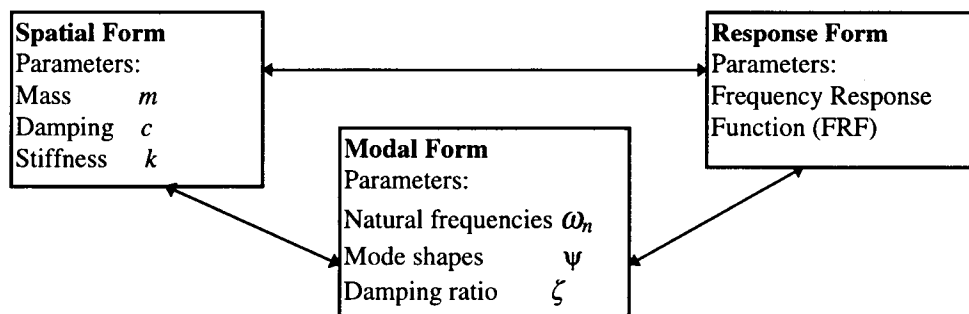


Figure 2.1 Equivalent forms of a mathematical model used in vibration analysis.

By applying ‘forced vibration analysis’ to either the modal or the spatial model, a *response model* is obtained. This model describes the response of the system when exposed to an external excitation. The response is usually expressed in a standard form as the system’s response to a unit-amplitude sinusoidal force applied to each point on the system individually and at every frequency within a specified range. Therefore, the response model consists of a set of frequency response functions, FRFs (Ewins, 1984).

In the visualization aspect, the vibration analysis is applied, obtained and interpreted the movements of an excited system. The motion of a system usually does not occur in one direction, but involves various translations, rotations, as well as deflection. Such motion is referred to as having more than one degree of freedom.

A *multi degree of freedom system* (MDOF) requires more than one coordinate to describe its dynamic motion. If a system requires N coordinates to characterize its motion such the system is termed N degrees of freedom system, or briefly an N -DOF system.

Mode shape (ψ) is a set of relative amplitudes of the coordinates at certain frequency of vibration. It describes how one coordinate behaves relatively to the others.

An N -DOF system has N natural frequencies with N corresponding mode shapes (Thomson, 1993).

An example of the mode shape representation applied to an automobile is shown in Fig. 2.2.

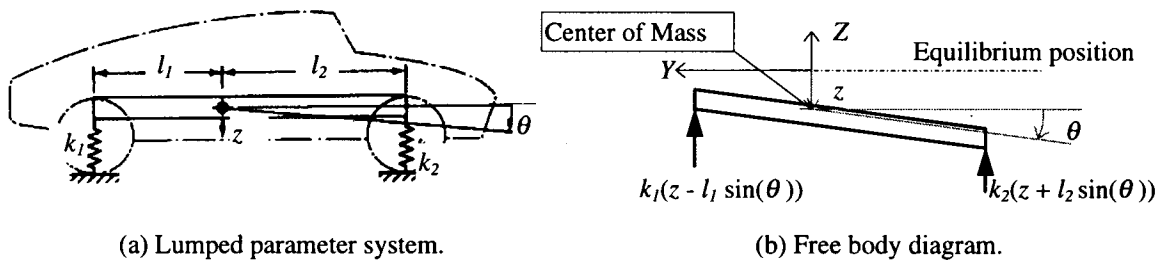


Figure 2.2 A 2-DOF model of an automobile (Thomson, 1993).

The automobile's dynamics is simplified such that it can be described by a lumped mass that moves only in a vertical direction and rotates around its center of mass. Therefore, this simplified automobile requires two coordinates, z and θ , to describe its motion (or it has two degrees of freedom, 2-DOF).

To define z and θ coordinates completely, reference axis of position and orientation is required. As shown in Fig. 2.2b, the system is reduced to a free body diagram (FBD) in which only external forces are concerned. The reference point is chosen at the center of mass. Any vertical translation from the reference position is described by z and a rotation around the center of mass is defined by θ as shown in the FBD.

To illustrate the mode shape, constant parameters are chosen to represent the automobile's properties. These parameters include $m = 3220$ lb., $k_1 = 2400$ lb./ft, $k_2 =$

2600 lb./ft, $l_1 = 4.5$ ft, $l_2 = 5.5$ ft, and $J = 51520$ lb. \cdot ft². By assuming that the responses of the model have the forms: $z(t) = Z \cdot e^{j\omega t}$ and $\theta(t) = \Theta \cdot e^{j\omega t}$, the natural frequencies are calculated as $\omega_1 = 6.90$ rad/sec and $\omega_2 = 9.06$ rad/sec. The corresponding mode shapes are calculated (Thomson, 1993).

$$\psi_1 = \begin{Bmatrix} Z/\Theta \\ 1 \end{Bmatrix}_1 = \begin{Bmatrix} -14.6 \\ 1 \end{Bmatrix} \text{ and } \psi_2 = \begin{Bmatrix} Z/\Theta \\ 1 \end{Bmatrix}_2 = \begin{Bmatrix} -1.09 \\ 1 \end{Bmatrix} \quad (2.1)$$

The mode shapes can be symbolically represented as shown in Fig. 2.3.

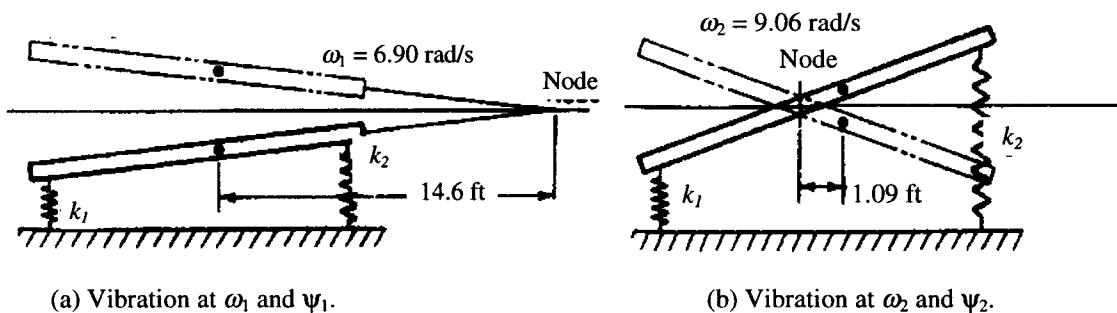


Figure 2.3 Vibrating lumped system in two different mode shapes (Thomson, 1993).

2.2 System Identification in Vibration Analysis

Theoretically estimated parameters (m , c , k) in the model are usually not adequate to describe an actual system. Some parameters are difficult to identify such as the damping coefficients and stiffness of the system. An efficient method is required to estimate the model parameters which are as close to the actual system as possible. System identification provides means to obtain such parameters for the purpose of developing mathematical models which describe the static and dynamic behavior of systems in a sufficiently accurate manner (Unbehauen, 1982). In system identification a model equivalent of the system of interest is determined based upon an input and an output of the system as shown in Fig. 2.4 (Natke, 1982).

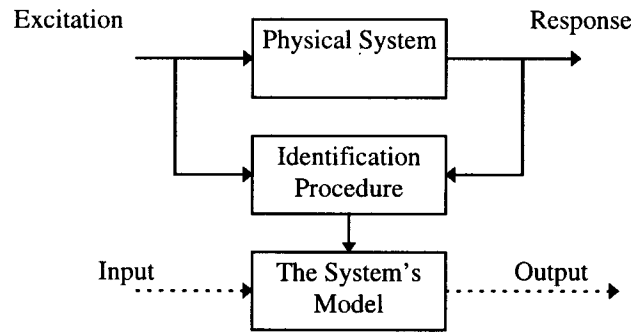


Figure 2.4 Obtaining the system's model by means of an identification procedure (Natke, 1982).

Algorithms used in the system identification can be classified as shown in Fig. 2.5. Mainly, determination of the model can be performed either in the frequency domain or in the time domain. In the frequency domain, the estimation of model parameters is based on fitting the model frequency response to the measured frequency response of the system. In the time domain approach the estimate of system parameters are based directly on the measured system's excitations and responses (Collins et. al., 1972).

The methods are also classified based on the spatial model or the modal model. Both approaches govern either one or both techniques: direct or iterative schemes. The "direct" or *non-parametric identification* approach is based on developing a single step solution procedure that estimates all required parameters. The "iterative parameter optimization" or *parametric identification* is based on the use of algorithms that relate change in parameter values to change in model responses in which a priori knowledge of the system is required to derive a mathematical model (Collins et al., 1972).

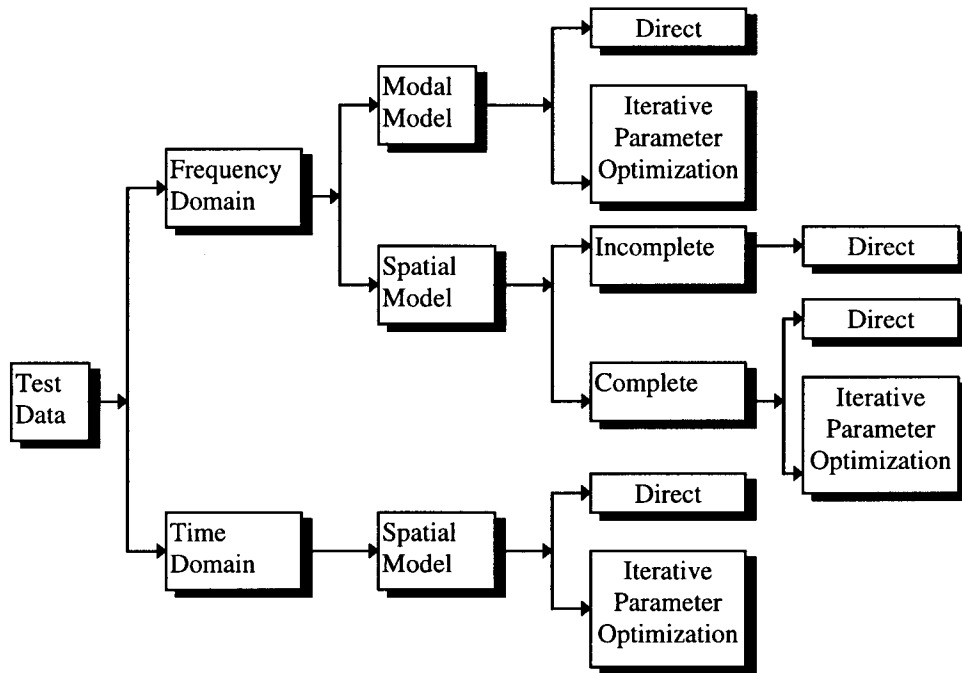


Figure 2.5 Classification of the system identification techniques (Collins et. al., 1972).

A distinction between the “complete” and “incomplete” methods are determined by comparing a number of the model’s degrees of freedom to the number of normal modes possessed by the mathematical model. If the number of the model degrees of freedom and measured modes are equal, a unique equivalent structure model can be identified in a straight forward mathematical manner (Collins et al., 1972).

A block diagram representation idea of the parametric identification methods is shown in Fig. 2.6. A critical examination of the quality of the model is obtained by a comparison of the system’s output with the model’s output where the system and the model are both excited by the same input signal. The measurable system output consists of an non-measurable output signal and the noise signal. Iterative procedures, *e.g.*, least squares method (Isermann, 1981), are performed to find such parameter values which

yield the error between the model output and the system output as small as possible (Unbehauen, 1982).

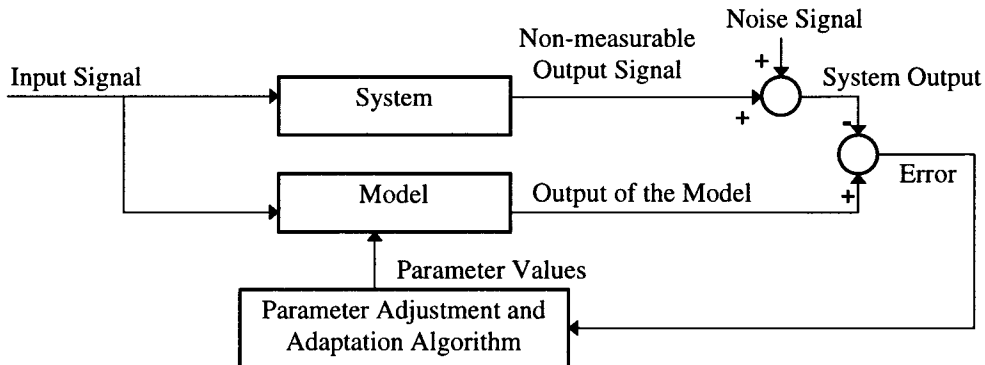


Figure 2.6 Parametric Identification (Unbehauen, 1982).

2.3 Commercial Visualization Software

The majority of commercial programs in vibration analysis visualize vibrations in mechanical systems by presenting the mode shapes of investigated system. These mode shapes are calculated either from an analytical approach or an experiment. In practical *experimental modal analysis* (EMA) software generates the mode shapes from a transfer function calculated using data from an experiment. Once the transfer function of the system of interest is obtained (based on the input and output signals), the natural frequencies and corresponding mode shapes are determined (Powell, 1992).

The vibration is presented as a relative movement of one part of the system with respect to the others. For example, if a user decomposes the visualized system into a sufficient number of elements, each point or *node* is assigned to represent the vibration of that particular element. In the animation of vibration, one reference node is displayed as oscillating sinusoidal with corresponds to a forced harmonic FRF responses at the selected resonance frequency. Positions of the other nodes are then calculated by using the relative

displacement information provided in the identified mode shapes. Therefore, the graphical display shows vibrating shape or deflected shape of the system at this selected resonance frequency (Vibrant Technology, 1996). An example of visualization display presenting mode shape of a plate at a certain frequency is illustrated in Fig. 2.8. Available software in the EMA area are LMS CADA-X™, STAR system™, EMODAL™, I-DEAS™, PC MODAL™, and ME Scope™ (Lang, 1990; Spectral Dynamics, 1995; Structural Dynamics Research Corporation, 1996; Vibration Engineering Consultants, 1997).

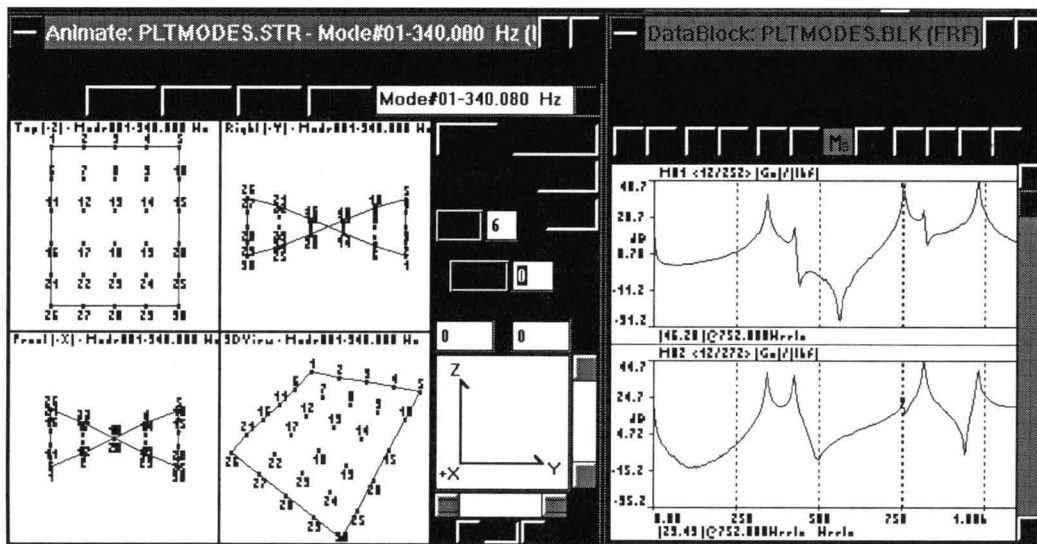


Figure 2.7 An animation of vibration on the display of ME Scope™ (Vibrant Technology, 1996).

Another approach to visualize vibrations in commercial software is to display *the operational deflection shape* (ODS) in which the forced dynamic deflection is determined at the operating frequency (Powell, 1992). This technique is utilized to find the mode shape of the system directly from an experiment. By mounting sensors (usually accelerometers) at specified nodes on the system's surfaces and then exciting the system with a suitable excitation, the relative deflection of each node is obtained by capturing all

signals from all nodes simultaneously when the system is subjected to excitation (Powell, 1992).

2.4 Measurement of Signals for Visualization

An approach of visualization technique in this research is similar to the Operational deflection Shape technique discussed above. Vibration of the system under consideration are measured in actual operating conditions using suitable sensors and then displayed directly. Therefore, this visualization method presents the actual movement of the system.

The technique proposed in this research is explained by way of example. A rigid plate fixed with springs and dampers represents a mechanical system. To visualize the complete motion of the plate six coordinates are required, which include three for rotations and three for translations. Discussion of these coordinates follows.

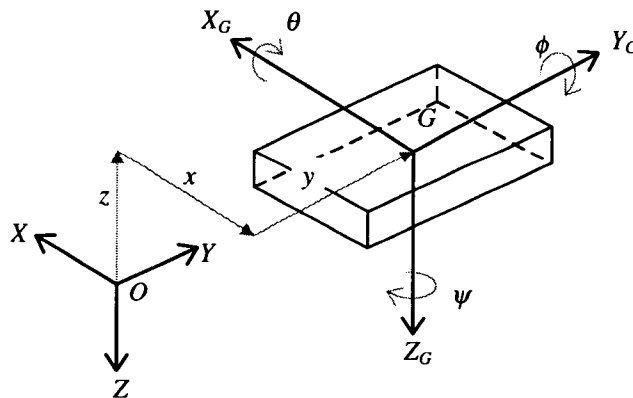


Figure 2.8 Six coordinates describing motion of a rigid plate.

The plate under investigation is shown in Fig. 2.8 in the Cartesian coordinate system comprising three orthogonal axes X_G , Y_G , and Z_G , defining a *plate coordinate system* $(XYZ)_G$. The origin of $(XYZ)_G$ is the center of mass of the plate, G . Another coordinate system (XYZ) , composed of three perpendicular axes, X , Y , Z , is introduced as a *global reference coordinate system*. When the (XYZ) is fixed with reference to the Earth, a motion of the plate can be described as a relative movement between the $(XYZ)_G$ and the (XYZ) . Variables required for describing this motion are:

- x - the translation of point G relatively to point O , parallel to the axis X ,
- y - the translation of point G relatively to point O , parallel to the axis Y ,
- z - the translation of point G relatively to point O , parallel to the axis Z ,
- θ - the rotation of the plate around the X axis (roll angle),
- ϕ - the rotation of the plate around the Y axis (pitch angle), and
- ψ - the rotation of the plate around the Z axis (yaw angle).

Linear accelerometers are employed in this research to detect linear vibration of the specific points of the plate in the X , Y and Z directions. The rotations are indirectly calculated from these linear accelerations using a technique proposed by Padgaonkar et al. (1975). This technique is further described in Section 4.5.

Piezoelectric accelerometers have been chosen because of their reasonable cost and excellent performance. However, the piezoelectric sensors are charge generators that require special amplifiers. Without proper amplifiers, these sensors can significantly distort the motion measurement. Details are discussed in the following section.

2.4.1 Piezoelectric Accelerometer Measurement

Piezoelectric transducers utilize piezoelectric effects occurring in certain crystals. The deformations of these crystals, due to an applied pressure, produce electrical charges on the external crystal surfaces. These effects occur in materials such as single-crystal

quartz, polycrystalline barium titanate, or lead zirconate (Kial and Mahr, 1984). A piezoelectric accelerometer consists of a piezoelectric sensing element sandwiched between the transducer's body and a seismic mass, as shown in Fig. 2.9. By mounting the transducer on a surface of a vibrating object, internal forces from the seismic mass cause deformations in the sensing element which, in term, produce an electrical charge (Dally et al., 1993).

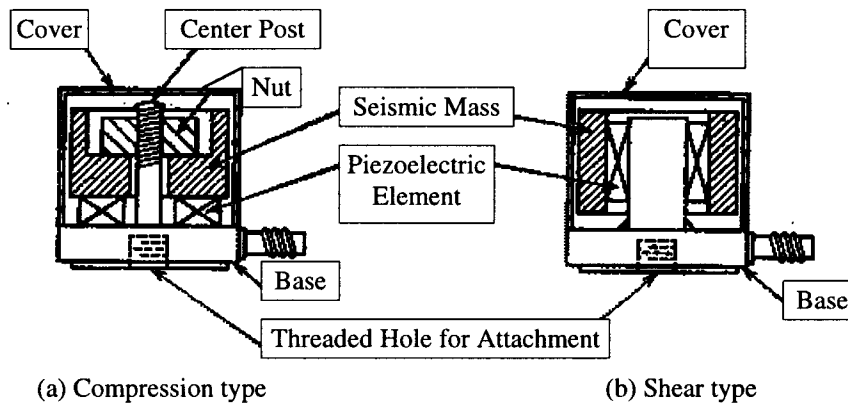


Figure 2.9 Configurations of piezoelectric accelerometers including (a) compression type and (b) shear type (Dally et al.,1993).

Since the output of the piezoelectric accelerometers is a high impedance charge, suitable amplifier circuits are required to obtain low impedance output. Charge amplifiers are the most popular instruments used for this purpose because of their advantages including high linearity, good frequency response, low influence of the cable capacitance, and high stability (Kistler, 1995). A typical measurement system that involves a high impedance piezoelectric accelerometer and a charge amplifier is shown in Fig. 2.10a (Kistler, 1995).

Integrated circuit piezoelectric (ICP) sensors have been developed with an advanced MOSFET¹ technology. The ICP transducers consist of high impedance piezoelectric accelerometers combined with miniature, built-in charge-to-voltage converter circuits, as shown in Fig. 2.11 (PCB, 1996). As a result, the charges generated by piezoelectric sensing elements are converted into voltage inside of the transducers providing low impedance that can be measured by any typical readout instrument (McConnell, 1995).

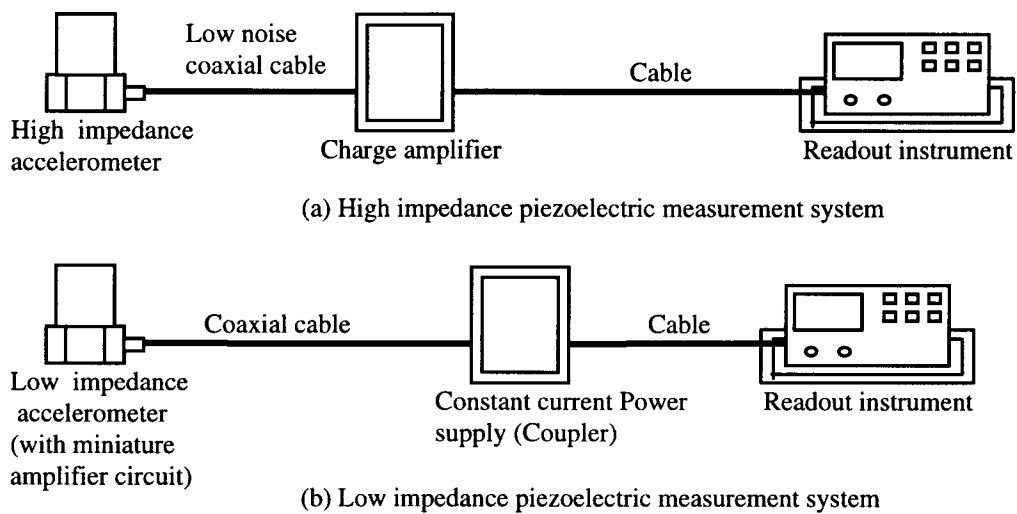


Fig. 2.10 Comparison diagrams of instrument setup between the high and low impedance system (Kistler, 1995).

Therefore, the low impedance piezoelectric systems requires only constant current power supply generated from an external low impedance power supply, or a coupler, to supply the transducer as shown in Fig. 2.10b (Kistler, 1995).

¹ MOSFET is an abbreviation of Metal-Oxide-Semiconductor-Field-Effect Transistor.

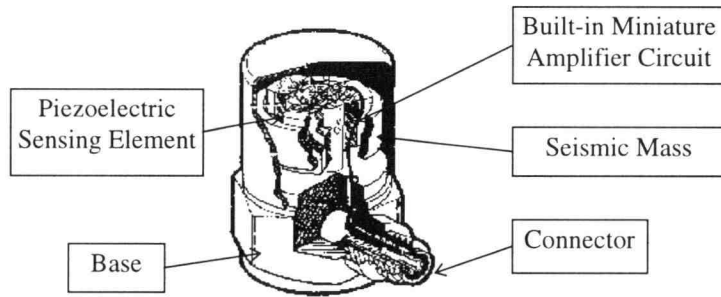


Figure 2.11 Cross-section diagram of the ICP accelerometer (PCB, 1996).

The low impedance systems offer potentially lower cost, and more convenient implementation than the sensors described earlier (Kial and Mahr, 1984; Kistler, 1995). However, the high impedance sensors used together with charge amplifiers offer significantly wider range of operation, since the time constant and gains can be controlled at the charge amplifier. In addition, the high impedance transducers contain no built-in electronics, therefore, they work in wider temperature range than the low impedance system (Kistler, 1995).

The overall FRF of a measurement system of the low impedance (ICP) accelerometer, $H_L(\omega)$ is (Dally et al., 1993)

$$H_L(\omega) = \frac{V_o}{s_v \cdot A_o} = \frac{1}{1 - \left(\frac{\omega}{\omega_n}\right)^2 + j \cdot 2\zeta \left(\frac{\omega}{\omega_n}\right)} \cdot \frac{j \cdot (TC1) \cdot \omega}{1 + j \cdot (TC1) \cdot \omega} \cdot \frac{j \cdot (TC) \cdot \omega}{1 + j \cdot (TC) \cdot \omega} \quad (2.2)$$

where

- s_v - the system's sensitivity (volts/g).
- A_o - amplitude of the acceleration applied to the transducer (g),
- V_o - the amplifier output voltage (volts),
- ω_n - the natural frequency of the accelerometer,
- ζ - the damping ratio of the accelerometer,
- $TC1$ - the time constant of the power supply,

- TC - the time constant of the miniature amplifier in the transducer, and
 j - the imaginary unit, $j = \sqrt{-1}$.

A magnitude plot of the above FRF is given in Fig. 2.12. It is clearly seen that the high end of the frequency bandwidth is limited by the natural frequency, ω_n , of the transducer. An estimated values of this limit are in the range of $0.1\omega_n$ to $0.3\omega_n$ (Dally et al., 1993; Brüel & Kjær, 1982). At low frequency the drop of gain is mainly due to the amplifier characteristics which define the discharge rate of the piezoelectric sensor. Concisely expressed by the discharge time constant of the amplifier which has properties of a high pass filter. This time constant is used to calculate lower limit of the bandwidth. For practical applications (5% error), the minimal measuring frequency is $\omega_L = 3 \cdot TC^{-1}$. (Kail and Mahr, 1984; Kistler, 1995). Therefore, the overall measurement bandwidth, f_m , is

$$3 \cdot TC^{-1} < f_m < \frac{0.3}{2\pi} \omega_n \quad [\text{Hz}] \quad (2.3)$$

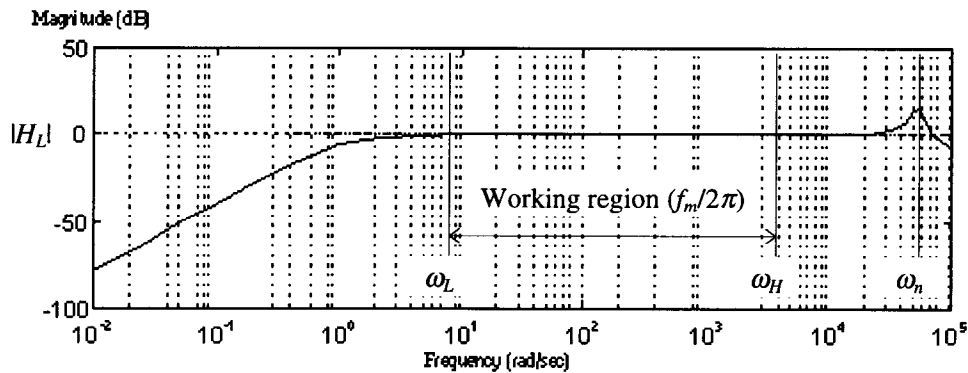


Figure 2.12 Magnitude plot of the FRF of a piezoelectric accelerometer (McConnell, 1995).

Application of piezoelectric accelerometers to measure transient vibrations requires additional consideration. Transient signals may contain DC signal components that are strongly attenuated as shown in Fig. 2.12. The result can be a drift in the response signal. The amplifier's time constant must be considered in the error analysis in such cases. Table 2.1 presents the time constant necessary to limit the error in measuring various transient signals (Dally et al., 1993).

To visualize motion of the 6-DOF plate considered in Section 2.4, acceleration signals have to be converted to displacements. Although performing double-integration on acceleration signals for this purpose seems possible in theory, its implementation is difficult. Combined distortion of signals measured by piezoelectric accelerometers usually results in a gradual change of the output voltage (drift). This change is as a relearn unknown function of time since drift may result, in addition to the distortions caused by the dynamics of the amplifier, from changes of temperature, line voltage or amplifier characteristics.

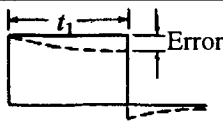
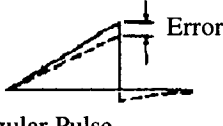
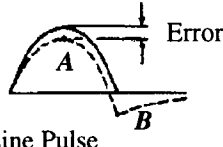
| Pulse Shape | Time Constant Required for Limiting Error | | |
|----------------------------------------------------------------------------------------------------------|-------------------------------------------|-------------------|------------------|
| | 2% Error | 5% Error | 10% Error |
|  Rectangular Pulse | $50t_1$ | $20t_1$ | $10t_1$ |
|  Triangular Pulse | $25t_1$ | $10t_1$ | $5t_1$ |
|  Half-sine Pulse | A $16t_1$ B $31t_1$ | $6t_1$ $12t_1$ | $3t_1$ $6t_1$ |

Table 2.1 Effect of time constant on the error in measuring various transient responses (Dally et al., 1993).

Low frequency measurement errors discussed above are strongly amplified when integration is applied in the process of converting accelerations into displacements. These errors are illustrated by the following example involving monitoring displacements of one point of the plate (see Section 2.4) in response to an impulse force acting on it. Kistler® model 8702B25M1 piezoelectric accelerometer (Kistler, 1995) is employed to measure the acceleration of this point. A signal from the accelerometer after subtracting the DC offset is shown in Fig. 2.13a. It is known that, given enough time after the force impulse, vibrations die down and the investigated point of the plate returns to its initial position. Although no drift is noticeable in the acceleration signal, its presence is evident after its integration, *i.e.*, in the velocity signals which rise gradually as shown in Fig. 2.13b. The displacement signal shown in Fig. 2.13c, which is obtained by applying another integration, has clearly visible drift, strongly exceeding the level of vibrations caused by the force impulse.

The primary solution to the drift problem in this research is the use of high-pass filters to suppress the low frequency components (drift) in the acceleration signals. Suitable high cut-off frequency has been selected by determining experimentally characteristics of the transducer. For this purpose, a power spectrum of the acceleration measured from the stationary (*i.e.* at rest) sensor was examined. The cut-off frequency was selected to be above the frequency of the dominating components appearing in the spectrum. The results obtained by applying high-pass filters (between each integration) to the signals measured from the excited (vibrating) system are shown as dashed lines in Fig. 2.13b and 2.13c.

A side effect of using high pass filters are distortion of the signals manifesting themselves as a phase shift at lower frequencies. These distortions can be clearly visible in signals having broad frequency contents, such as a squarewaves or impulses. It goes without saying that high-pass filters are acceptable only for visualizing vibrations of objects with a frequency higher than the selected cut-off frequency.

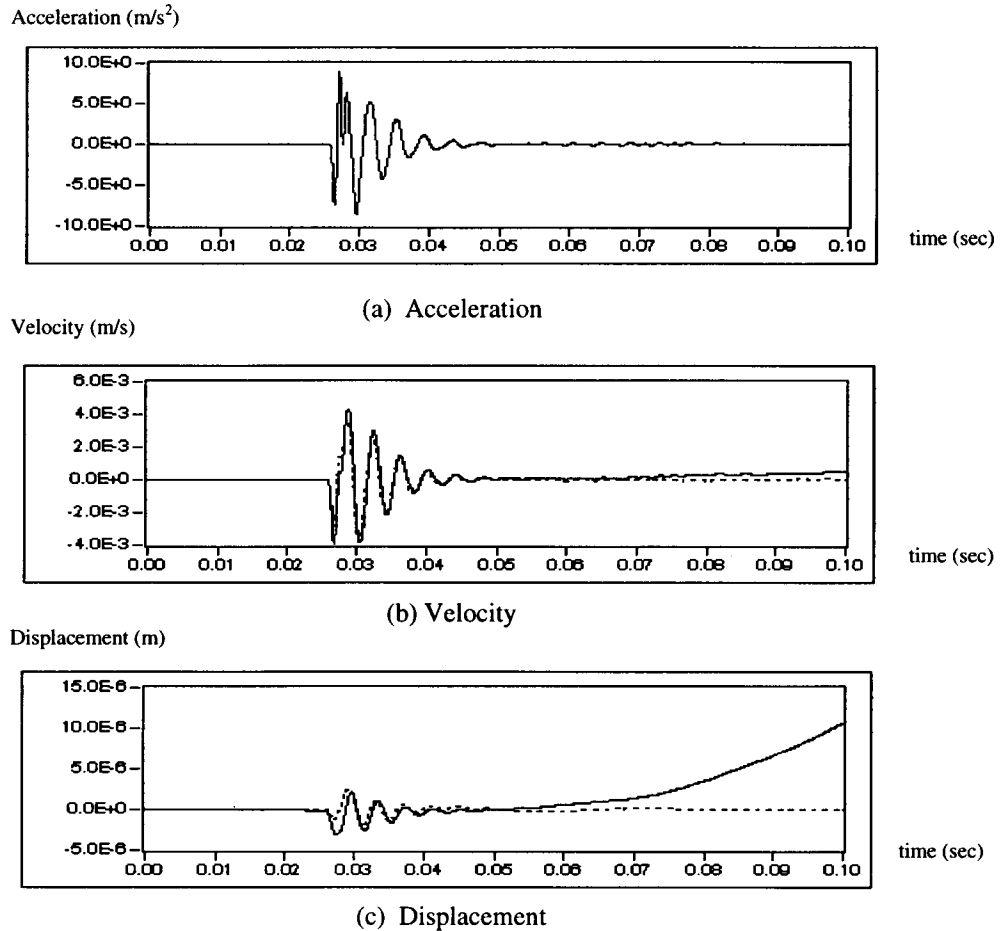


Figure 2.13 Illustration of the impact of a low frequency drift on the displacement obtained from acceleration signal. Solid lines in figures (b) and (c) represent integrated and double-integrated signals. Dashed lines represent the same signals after using digital high-pass filters (see Section 4.3.2).

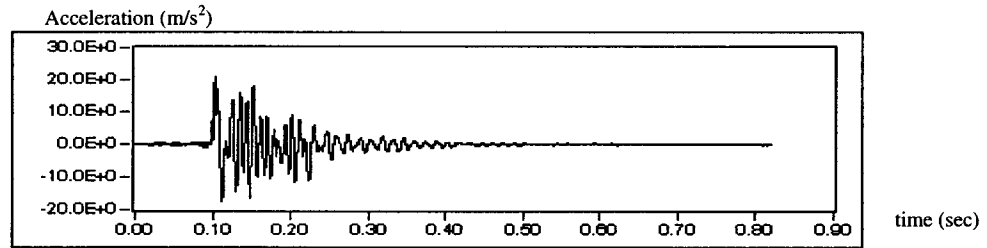
2.4.2 Conventional Low Frequency Accelerometers

The application of piezoelectric accelerometers can cause serious difficulties when measuring low frequency vibrations. Possible errors include drift and noise as discussed above. A significant improvement can be achieved by using high performance accelerometers. In this section, various types of accelerometers, which have recently been developed with a purpose of improving the low-frequency performance, are

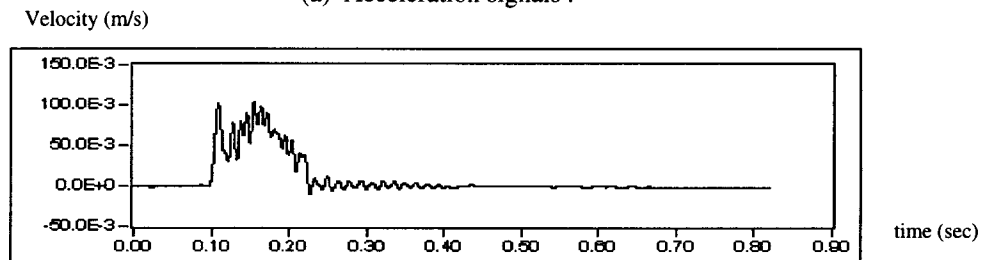
reviewed. These new accelerometers are recommended for application in the visualization technique presented in this thesis.

Variable capacitance microsensors are designed for a measurement of low frequency acceleration, inclination, and shock vibration (Endevco, 1996). They employ a capacitance sensing element that is composed of a very small seismic mass chemically etched from a single piece of silicon and positioned between two electrode plates. As the seismic mass deflects under accelerations, the capacitance between these plates changes. With a signal conditioning circuit, position of the sensing mass is detected and converted to an output voltage. To prevent excessive shock motion, the sensing mass is damped by a gas damper resulting in an increase in the measurement bandwidth from DC to 2000 Hz, depending on the sensitivity (Kistler, 1995).

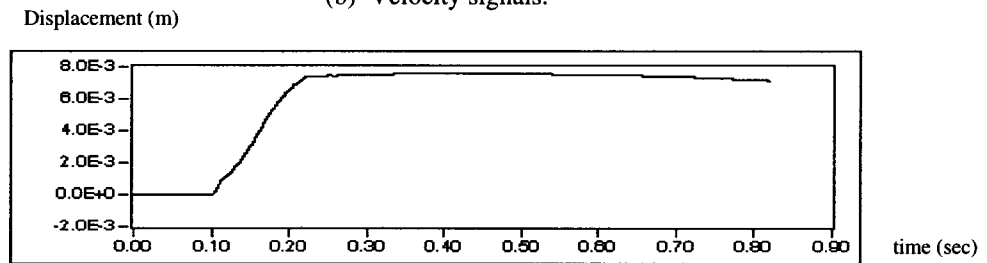
Fig. 2.14a shows example acceleration signal acquired by a capacitive accelerometer, Kistler[®] 8302B2S1 (Kistler, 1995) mounted on a rigid object. This object is slowly moved from one position to another, 7.8 mm away. The initial portion of the signal (0 - 0.09 sec) representing an output signal of the stationary accelerometer is used to calculate the value of its DC offset. This value is next subtracted from the signal, and the resultant is integrated twice to obtain the velocity and displacement measured by the sensor. These latter signals are shown in Fig. 2.14b. An inspection of the measured and actual displacement over the period shown in the figure indicates a suitability of the sensors under consideration for visualization of vibrations with frequencies down to as low as 2 - 5 cycles per second.



(a) Acceleration signals .



(b) Velocity signals.



(c) Displacement signals.

Figure 2.14 Signals obtained by means a variable capacitance accelerometer and its results from double-integration (Kistler, 1995).

Still better sensors can be developed by improving the measuring technique for detecting position of the seismic mass since the acceleration can be calculated from the relative displacement between the seismic mass and the sensor's body. For example, the electron tunneling effect can be used in the measurement of this displacement (Rockstad et al., 1992). The transducer consists of a miniature silicon cantilever that acts as a sensing mass as shown in Fig. 2.15.

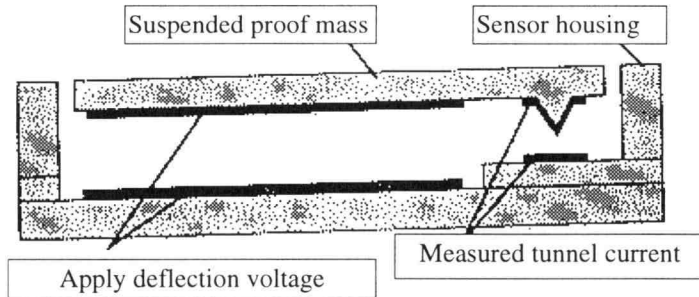


Figure 2.15 Schematic illustration of a single-element electron tunneling accelerometer (Rockstad et al., 1992).

The cantilever is a gold-coated electrode and supported over the other electrode by a small spacing that is maintained constant by a feedback circuit applying voltage bias to the electrodes. When subjected to an acceleration, the cantilever tip deflects, thus, changes 'tunnel' current between the electrodes. The feedback circuit then controls the electrostatic voltage such that it returns to its initial position. The variations in voltage required to maintain constant distance between electrodes indicates the sensor's acceleration. This voltage is free from hysteresis and drift, and insensitive to temperature. The FRF cover the range from DC up to 5000 Hz (Rockstad et al., 1992).

Another approach of the low frequency measurement is the use of a piezoresistive element to detect the motion of the seismic mass. These elements are mounted on a beam that suspends the micromachined silicon seismic mass as shown in Fig. 2.16. Bending of the beam caused by the accelerated mass yields a strain that is detected by the piezoresistor. With a bridge circuit, the changes in resistance are converted to an output voltage.

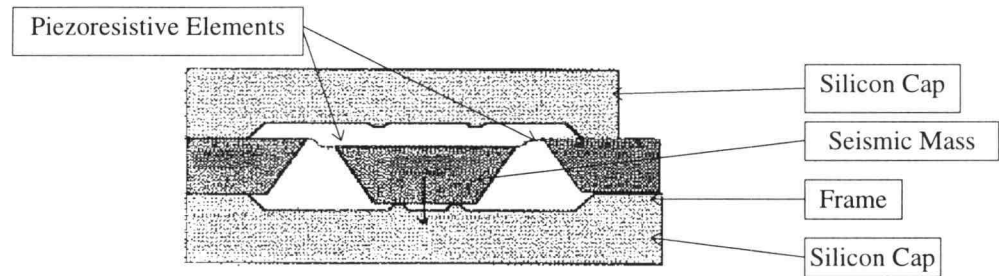


Figure 2.16 A cross section of the piezoresistive accelerometer (IC Sensors, 1997).

2.4.3 Multi Directional Accelerometer

To describe a rigid body motion in 6-DOF, at least six linear accelerometers are required (see Section 2.4). The number of transducers can be minimized by using multi-axis accelerometers, resulting in a reduction of equipment cost. PiezoBEAM[®] transducer from Kistler[®] involves a pair of cantilever beams made of a piezoelectric ceramic material as shown in Fig. 2.17. The beams are sensing elements that are constructed in a T-shape. By comparing the phase difference of the electrical charge signals from both beams, the linear and angular accelerations can be calculated (Kistler, 1996).

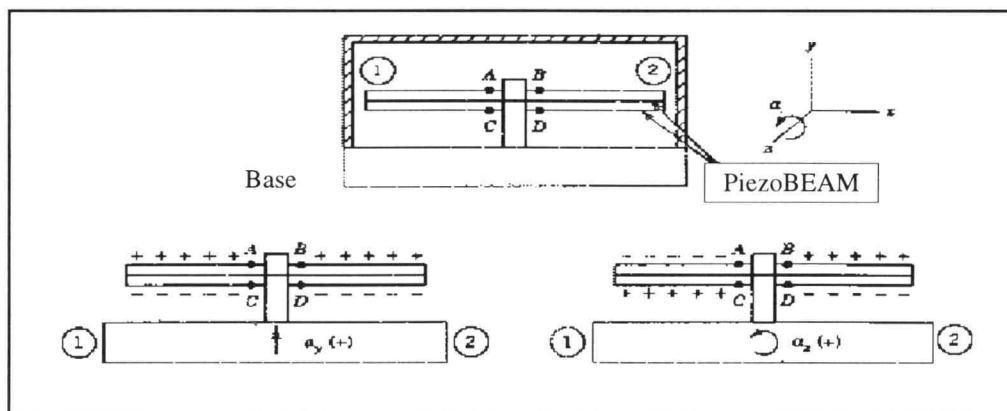


Figure 2.17 A cross section of the PiezoBEAM[®] accelerometer (Dally et al., 1993).

2.5 Closure

Visualization of vibration in theoretical analysis is introduced. The mode shapes from the theoretical vibration analysis provide graphical representation of coordinates' relative motion in the system at certain frequencies. The visualization technique is proposed in which actual movement is detected experimentally using accelerometers. By studying characteristics of the piezoelectric accelerometer measurement system, problems in the visualization method are presented as limitation of measurement at low frequency caused by discharge of output voltage. Higher performance accelerometers are reviewed and suggested for improvement direction. In the next chapter, concepts of the proposed visualization technique are described. These concepts include the use of sensors in the signal based vibration visualization and the use of a mathematical model to improve the visualization in the model based vibration visualization.

CHAPTER 3

MODEL BASED VISUALIZATION OF VIBRATIONS

The background and essential characteristics of the model based vibration visualization are presented in this Chapter. Algorithms implemented in this thesis and an outline of future work are discussed. The model based vibration visualization comprises two stages. The first stage, a *signal based vibration visualization*, is developed in this thesis. The second stage, proposed as the future work, is an improvement of the above visualization by using the system's mathematical model. An experimental model of commercial dynamometer is selected as a mechanical system to be visualized. Since a mathematical model of this dynamometer is employed for generating the estimated responses to applied excitation signals, a technique used to obtain this latter model is reviewed at the end of this chapter. The same model is also used to interpret experimental results in Chapter 5.

3.1 An Overview and Definition of the Proposed Visualization Technique

The proposed visualization program has been developed to use real signals from sensors attached to vibrating rigid bodies for generating animated motion of these bodies. Together with the acquired signals, a mathematical model of the dynamic system comprising the bodies is also used to detect and suppress errors in the acquired signals.

The entire visualization scheme is implemented in two stages. The first stage, referred to as '*signal based vibration visualization*', is the focal point of this research. A suitable methodology discussed in detail in chapter 4 facilitates visualization of vibrations of rigid bodies by using signals from acceleration sensors attached at precisely defined locations of these bodies. Set of data collected from these sensors are referred to as

'*signal based responses*' (SBR). Typically, conducting one experiment generates one SBR.

In the second stage of the visualization each specific SBR is analyzed to detect possible errors in recorded signals by using a mathematical model of the system under consideration. This is accomplished by comparing the SBR with a hypothetical response which is generated by stimulating the model with the actual input signals that acted on the physical system. These generated hypothetical output signals are referred to as '*model based responses*' (MBR). In addition to detecting errors (*e.g.* drift) in signals from the actual sensors the knowledge of mathematical model facilitates significant suppression of these. Thus, the obtained enhanced visualization is referred to as the '*model based vibration visualization*'. Fig. 3.1 shows in the form of a flowchart the essential units of the developed visualization and software.

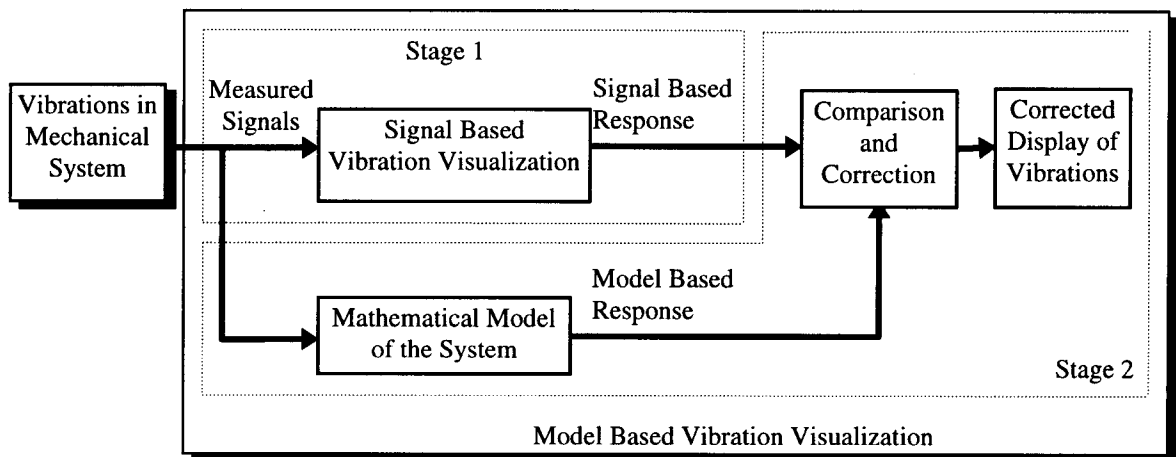


Figure 3.1 Flowchart of model based visualization of vibrations.

3.2 Signal Based Vibration Visualization

A flowchart of the methodology used in the signal based vibration visualization is illustrated in Fig. 3.2. Vibrations of a mechanical system are detected by means of accelerometers attached to the system. The acceleration signals acquired using a data acquisition (DAQ) system are processed in two steps to obtain generalized coordinates². These steps are designated in the figure as the Signal Processing and Generalized Coordinate Calculation. Finally, the processed data are transformed into three dimensional (3-D) animated pictures representing the vibrations of the mechanical system.

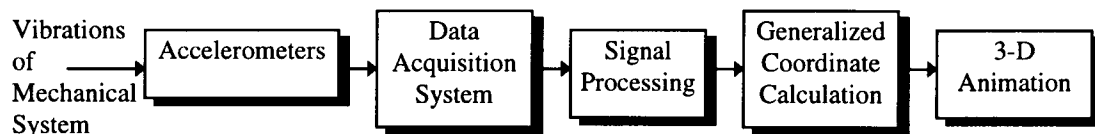


Figure 3.2 A flowchart representation of the methodology developed for the signal based vibration visualization.

The major purpose of the signal processing stage is the calculation of variables necessary for describing spatial motion of the system from measured data. This is accomplished by double integration of the acceleration signals to obtain translational and rotational displacement variables required for the visualization. An implementation of the integration is difficult since various errors distort the measured signals, as discussed in Section 2.4.1. As a result, the animated vibration of the system may be quite different from the actual ones. A software package developed in this thesis comprises all functions represented in Fig. 3.2. It also suppress a large portion of errors in the measured signals by using rudimentary techniques, namely subtraction of the average value of the signal and high-pass filtering. These techniques are well documented in the professional

² Definitions of the generalized coordinates is introduced in Section 3.5.

literature and applied in commercial systems (see Section 2.3). To achieve further attenuation of errors it is recommended to use a constitutive mathematical model of the system whose vibrations are visualized. A full implementation of this new approach is a complex task, beyond the scope of this thesis. The remaining part of this chapter contains a discussion of the ideas involved in implementing constitutive mathematical models for the purpose of error suppression.

3.3 Model Based Vibration Visualization

The model based vibration visualization is based on an assumption that vibrations of a system can be estimated if its mathematical model is known. To model the system, a *priori* knowledge of the system is required.

There are various classifications of dynamic systems depending upon the features of these systems that are interest. A classification proposed by Chung (1991) is applicable in this research. It divides dynamic systems based on their features relevant to parameter identification as shown in Fig. 3.3. Shaded boxes indicate the path applicable in this research .

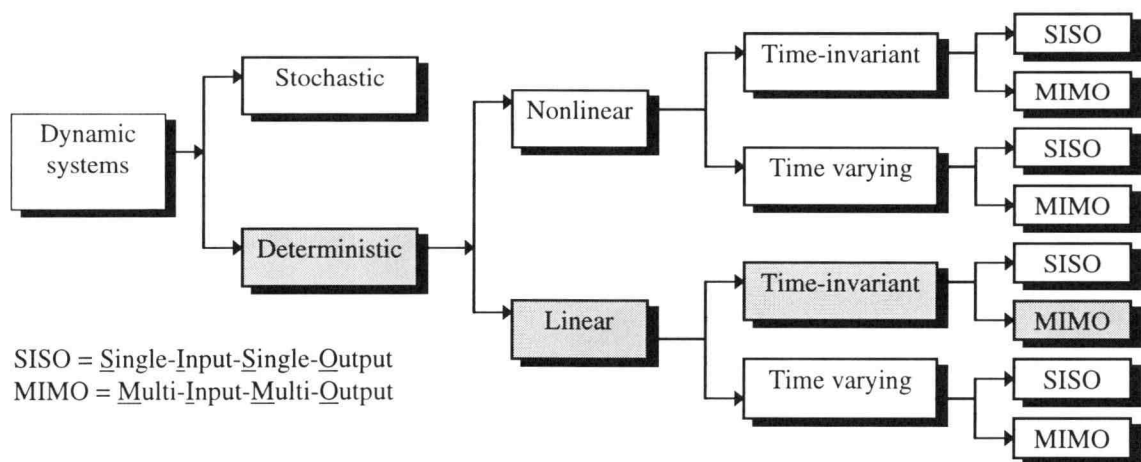


Figure 3.3 Classification of dynamic systems (Chung ,1991).

A system that has several input signals that stimulate responses on several outputs as shown diagrammatically in Fig. 3.4, is termed The multi-input multi-output (MIMO) system.

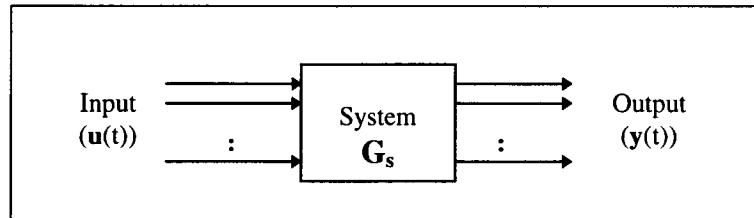


Figure 3.4 The MIMO system.

Dynamic mechanical systems under consideration are MDOF systems and also MIMO systems which can be described by the following vector-metric equation. (Chung, 1993)

$$\mathbf{Y}(s) = \mathbf{G}_s(s) \cdot \mathbf{U}(s) \quad (3.1)$$

where

$\mathbf{Y}(s)$ - Laplace transform of the output vector $\mathbf{y}(t)$, *i.e.*, a vector whose elements are all output signals,

$\mathbf{U}(s)$ - Laplace transform of the input vector $\mathbf{u}(t)$, and

$\mathbf{G}_s(s)$ - the transfer function matrix of the system.

The matrix $\mathbf{G}_s(s)$ has $n \times m$ elements, $G_{ij}(s)$, where $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, m$. Each of these elements is a rational polynomial representing a transfer function between input j^{th} and output i^{th} for a system with m inputs and n outputs.

It is usually difficult to estimate the exact transfer function, $\mathbf{G}_s(s)$ of an actual physical system. Therefore, an Equivalent System is often introduced, which retains the major properties of the original system but at the same time has a transfer function model that can be experimentally estimated. If this latter transfer function, $\mathbf{G}_E(s)$, is obtained by

means of a modeling technique, the estimated response, $\mathbf{y}_E(t)$, can be calculated from the input, $\mathbf{u}(t)$, acting on the original physical system and the transfer function of the equivalent system, \mathbf{G}_E , using the following relationship

$$\mathbf{y}_E(t) = \mathbf{L}^{-1} [\mathbf{G}_E(s) \cdot \mathbf{U}(s)] \quad (3.2)$$

where

\mathbf{L}^{-1} denotes the inverse Laplace transform operator.

Since the model based vibration visualization employs the estimated response, \mathbf{y}_E , to compare with the signal based response (see Section 3.2), the accuracy of \mathbf{y}_E is of utmost importance. Thus, high accuracy of the system's modeling and estimation are necessary. This can be achieved by applying accurate system identification techniques introduced in Section 2.2. A structure of the model, *i.e.*, form of equations, is derived such that it captures the essential physical laws governing the system's behavior. Parameters in this model structure are estimated from the input and the output signals. The estimation procedure can be performed in real time, which means that the identification takes place at the same time as the system's vibrations are measured.

The model based vibration visualization is an extension of the signal based vibration visualization presented in Section 3.2, in which the system identification and model response analysis technique are included. A flow diagram of the enhanced visualization is shown in Fig. 3.5. The visualization begins with the measurement of vibrations of a mechanical system by means of sensors (accelerometers) which is performed by the *Data Acquisition* (DAQ) system. The measured signals are processed in the *Signal Processing* unit, and the *Generalized Coordinate Calculation* unit yielding the 'signal based' response. In a parallel thread, a suitable mathematical model of the system is derived analytically in the *Model Derivation* unit. The modeling technique that involves symbolic computations can be applied to generate a structure of the model, while the essential unknown parameters are represented as symbolic coefficients in model's equations (Rehsteiner et al., 1997). To complete the model, a system identification is applied to estimate the unknown parameters. As already explained in

Section 2.2, this technique employs an iterative procedure that adjusts values of the model's parameters in order to minimize a difference between the signals recorded from the sensors and calculated responses of the model.

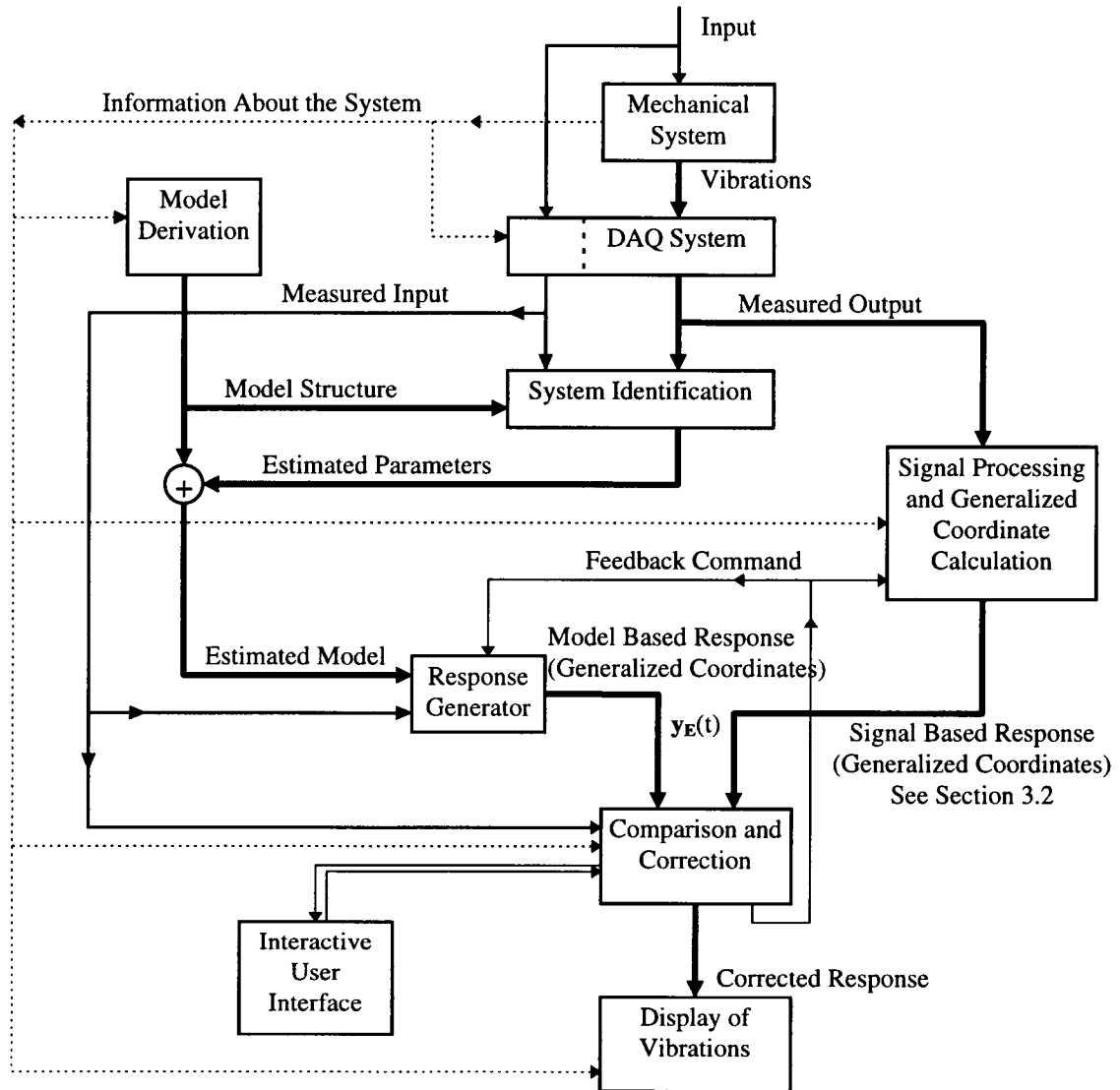


Figure 3.5 Block diagram of the model based vibration visualization.

The complete analytical model is represented by the analytical generated structure and the estimated parameters. This model is employed for generating the model based

response using the stimulation acting on the actual tested system³. Information from the estimated model is also used in the *Comparison and Correction* unit to generate suitable feedback commands for tuning of the Signal Processing unit and the Response Generator. The *comparison and correction* unit is also responsible for generating messages and warning to the user.

A flowchart of the *Comparison and Correction* unit is shown in Fig. 3.6. This unit comprises *Data Analysis and Filter Tuning* sub-unit in which the actual and estimated responses of the system are compared. The comparison can be based upon various indicators, such as, the coherence function (Bendit and Piersol, 1980). Results from the comparison are used for the validation of the acquired data and diagnosis of likely errors.

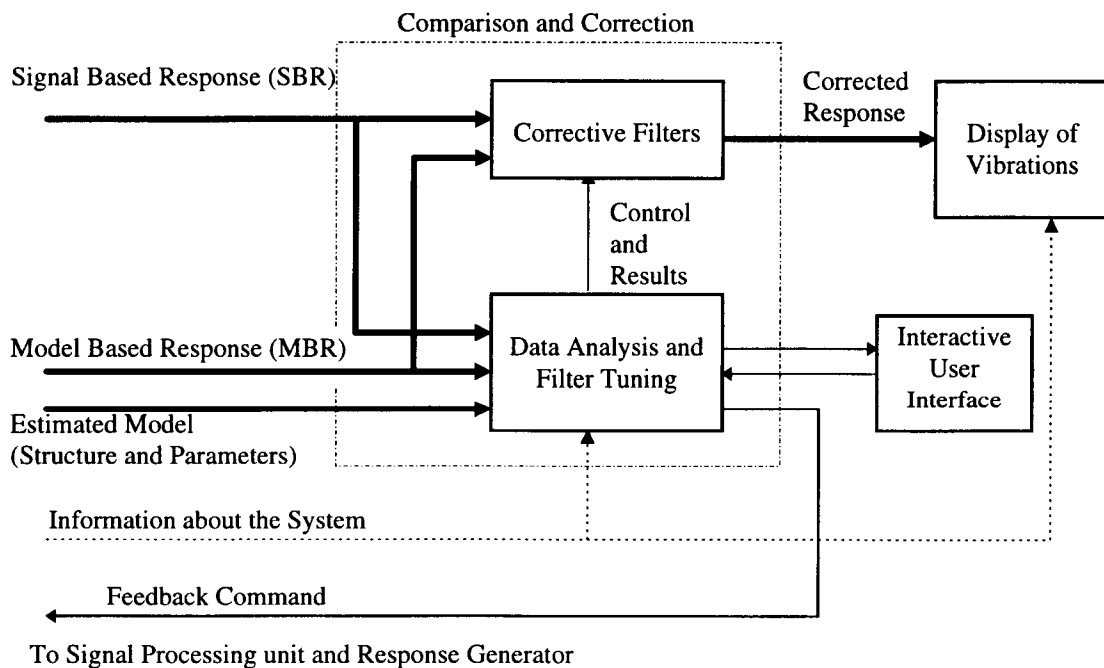


Figure 3.6 Block diagram of the Comparison and Correction unit.

³ For example, a force impulse from an impact hammer.

For example, if the coherence results show discrepancy between both responses above a certain pre-set level, the control unit informs the user about the incompatibility of the data via the *Interactive User Interface*. The user may insist to display the response by sending a command to the *Data Analysis and Filter Tuning unit*. Relevant correction methods may be employed, for example, subtracting mean values of the estimated errors from the signal based responses or re-processing the signals using new parameters set in the *Feedback Command*.

An alternative method for the visualization improvement is to compare directly the acceleration signals acquired from the sensors with acceleration responses calculated from the model. A block diagram corresponding to this better method is shown in Fig. 3.7. The drift embedded in the signals can be detected and eliminated by comparing quasi-static values of acceleration signals obtained by these two essentially different approaches. One significant advantage of this method is the elimination of high-pass filters (see Section 4.3.2) which, as an effect, introduce distortion into the signals recorded from the sensors.

The above discussion of the algorithms facilitating enhanced visualization of vibrations in mechanical systems addressed only the main aspects of this complex problem. There are numerous ways to implement these algorithms and broad spectrum of additional signal processing methods that can be applicable. More research is needed to find the most reliable and efficient methods.

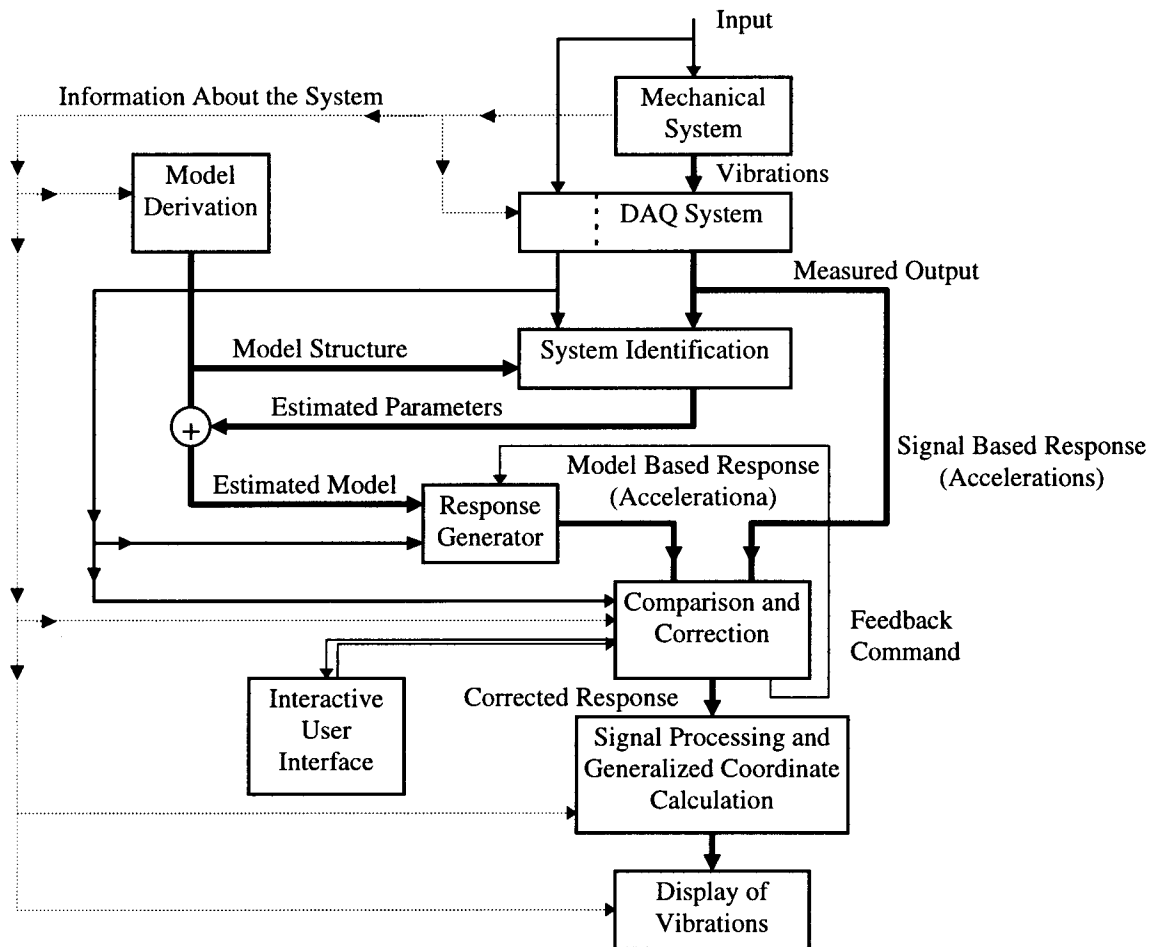


Figure 3.7 An alternative flowchart of the model based vibration visualization.

3.4 Feasibility Study of the Visualization Enhancement

An application of the mathematical model is investigated to generate responses of a mechanical system stimulated by known input signals. This system is a physical model of a high performance dynamometer (Kistler[®] type 9257A; Kistler, 1995). The *dynamometer* is a multi-component force sensor comprising the major parts shown in Fig. 3.8. A platform is supported by four piezoelectric-based sensing elements. When a force is applied to the workpiece attached to the platform, the platform is moved from its equilibrium position, thus causing the piezoelectric sensing elements to deform. Electric

charges generated from the deformations are measured and combined to calculate the applied force. If the platform is considered as a rigid body, its motion is described by six degrees of freedom: three translations and three rotations as previously described in Section 2.4.

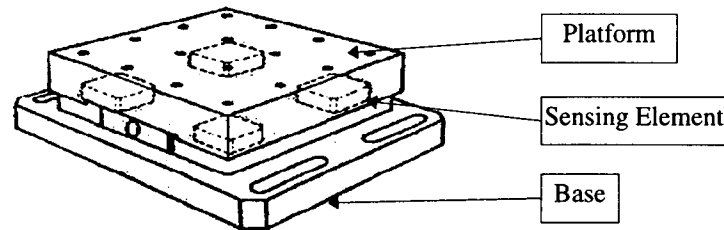


Figure 3.8 Major components of the dynamometer (Kistler, 1995).

The signal based vibration visualization technique proposed in this Chapter is applied⁴ to visualize vibrations of the dynamometer's platform. A linear model of the dynamometer (Chung 1993 and Chen 1996) is used for calculating theoretical vibration responses. Calculated and measured generalized coordinates are then compared. Results and errors of the visualization are analyzed. The main research subjects dealt with in this thesis and pertain to model based visualization are schematically shown in Fig. 3.9. These subjects are laid out in the figure such that they resemble the flow chart of signal based visualization algorithm shown in Fig. 3.2.

⁴ Detailed discussion of the experiments is presented in Chapter 5.

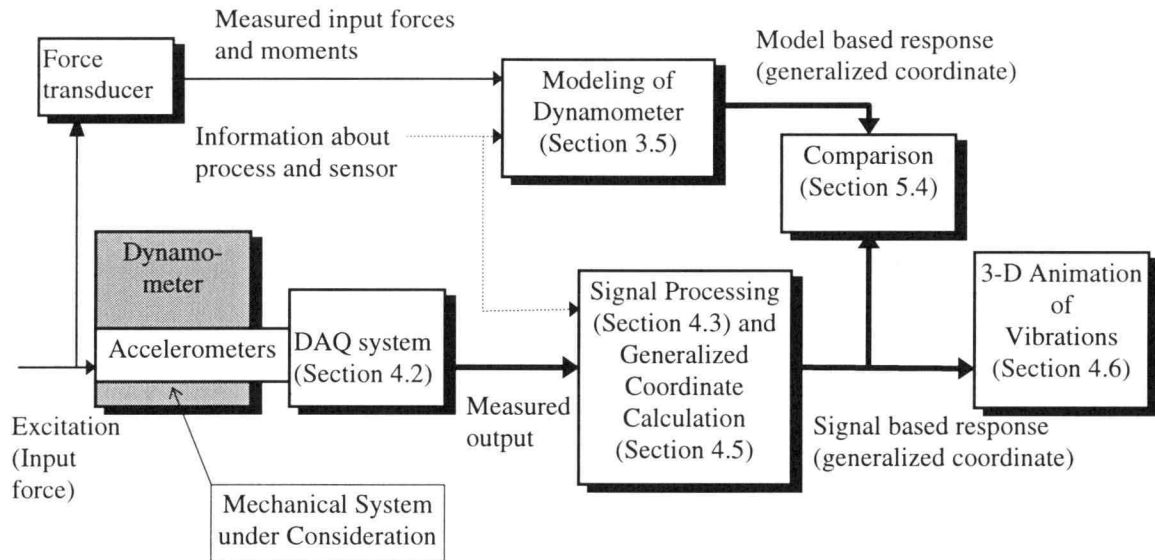


Figure 3.9 Main research subjects pertaining to model based visualization.

3.5 Modeling of the Dynamometer

Practical application of the model based visualization hinges upon the capability to derive analytical models of the systems under consideration efficiently and rapidly. A modeling technique of the dynamometer proposed by Chung (1993) has been used in this thesis. This technique using Lagrange's equation has been combined with computer-aided symbolic computation proposed by Chen (1996). The modeling involves the following four stages: (1) decomposition level in which the physical system is converted into components where motion attributes are assigned to each part, (2) mapping and identification level where the energy attributes are assigned to each decomposed component, (3) synthesis level in which generalized coordinates are assigned to the components and the mathematical model is built, and (4) modification level where the generated model is checked the analytical model specifications or experiment results (Chen, 1996).

This modeling methodology is applied for modeling the dynamometer under consideration installed on a machine tool (Fig. 3.10). The machine tool is treated as a linear, rigid-body system and it is decomposed into four major units shown in Fig. 3.11. Spring-dashpot elements represent bolts, joints, and guideways that link all components (Chen, 1996).

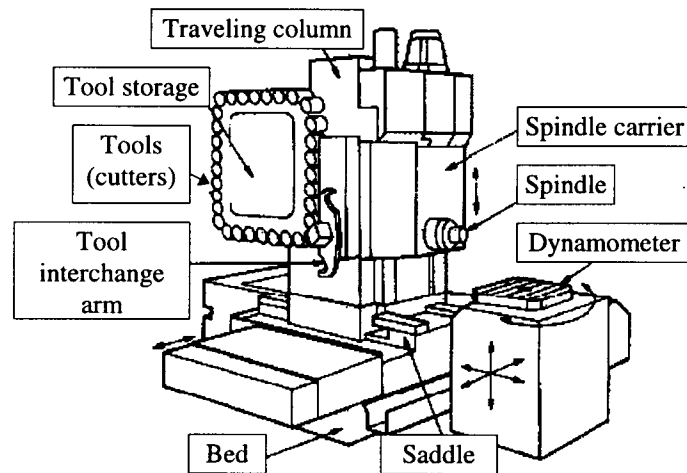


Figure 3.10 A high speed machine tool (Kalpakjian, 1992).

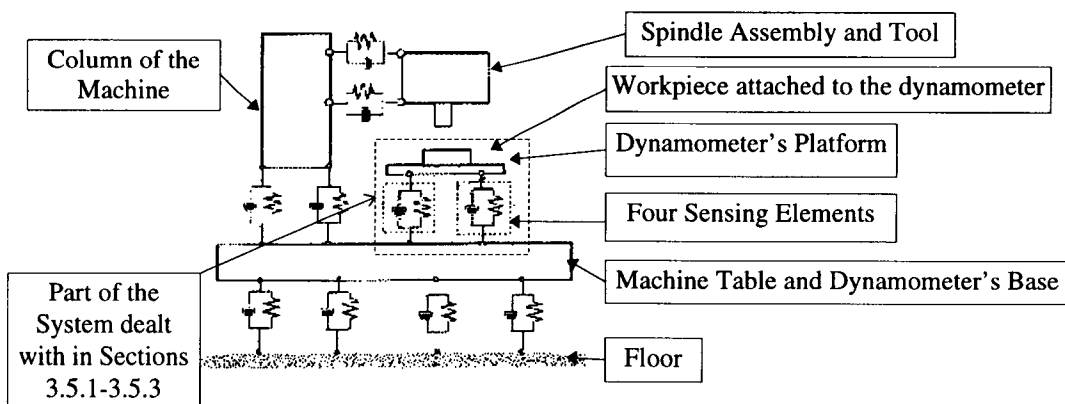


Figure 3.11 A simplified model of the machine tool from Fig. 3.10 with the dynamometer installed (Chen, 1996).

3.5.1 Mechanistic Model

The dynamometer described in the preceding section, whose simplified mechanical diagram is shown in Fig. 3.12, is considered as a MIMO system with an input vector, $\mathbf{u}(t)$, and output vector, \mathbf{d}_G . The input vector has in a general case twelve components (Chung, 1993)

$$\mathbf{u}(t) = [\mathbf{F}(t) \quad \mathbf{M}(t) \quad \mathbf{D}_T(t) \quad \mathbf{R}_T(t)]^T \quad (3.3)$$

where

$\mathbf{F}(t) = [F_x \quad F_y \quad F_z]$, forces acting on the platform,

$\mathbf{M}(t) = [M_x \quad M_y \quad M_z]$, torques applied on the platform around X_G , Y_G and Z_G ⁵, respectively,

$\mathbf{D}_T(t) = [X_T \quad Y_T \quad Z_T]$, translational motions of the dynamometer's base, and

$\mathbf{R}_T(t) = [\theta_T \quad \phi_T \quad \psi_T]$, rotational motions of the dynamometer's base.

The output vector, \mathbf{d}_G , is a set of variables describing position and orientation of the rigid platform. This set is called a *generalized coordinate list*, \mathbf{d}_G , that is defined as follows

$$\mathbf{d}_G = [x \quad y \quad z \quad \theta \quad \phi \quad \psi]^T \quad (3.4)$$

where

the $x, y, z, \theta, \phi, \psi$ are defined in Section 2.4.

Under simplifying assumption that the dynamometer's base is fixed and \mathbf{D}_T and \mathbf{R}_T are equal to zero, the input vector in Eq. 3.3 reduces to a 6x1 vector

$$\mathbf{u}(t) = [\mathbf{F}(t) \quad \mathbf{M}(t)]^T \quad (3.5)$$

⁵ Definitions of axes are introduced in Section 2.4.

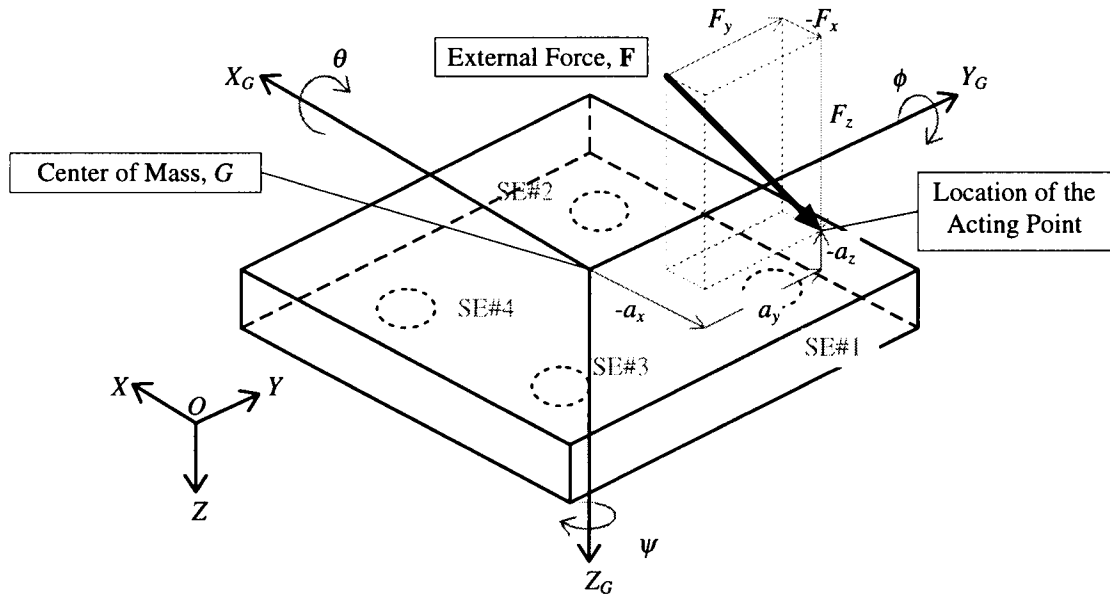


Figure 3.12 Simplified mechanical model of the dynamometer under consideration (Chung, 1993).

When the workpiece is being machined, an external force, $\mathbf{F}(t)$, is acting on the platform as shown in Fig. 3.12. The acting point of the force is located at $a_x \mathbf{i}_G + a_y \mathbf{j}_G + a_z \mathbf{k}_G$, where \mathbf{i}_G , \mathbf{j}_G and \mathbf{k}_G are unit vectors of the X_G , Y_G and Z_G , respectively. The quantities a_x , a_y and a_z are signed numbers measured from point G along the X_G , Y_G and Z_G directions. The external force contains three components, F_x , F_y and F_z which also generate three associated moments (M_{xF} , M_{yF} , M_{zF}) with respect to the center of gravity

$$M_{xF} = a_y F_z - a_z F_y \quad (3.6)$$

$$M_{yF} = -a_x F_z + a_z F_x \quad (3.7)$$

$$M_{zF} = a_x F_y - a_y F_x \quad (3.8)$$

The three components of the measured force and three resultant moments acting on the platform represent a complete set of independent variables in the equations of motion. This set is henceforth referred to as a forcing function vector, \mathbf{F}_e . It is obtained

by combining six elements of the simplified input vector \mathbf{u} given by Eq. 3.5 with the moments M_{xF}, M_{yF}, M_{zF} ,

$$\mathbf{F}\mathbf{e} = [F_x \quad F_y \quad F_z \quad (M_{xF} + M_x) \quad (M_{yF} + M_y) \quad (M_{zF} + M_z)]^T \quad (3.9)$$

F_x, F_y and F_z are signed numbers that are positive if they are in the positive directions of the respective X_G, Y_G and Z_G axes. The above equation can be rewritten in terms of the input vector \mathbf{u} and a matrix $\mathbf{C}_u(a_x, a_y, a_z)$ whose exact form is given in Appendix B.

$$\mathbf{F}\mathbf{e} = \mathbf{C}_u \cdot \mathbf{u} \quad (3.10)$$

By applying Lagrange's method, the equation of motion can be derived in a matrix-vector form (Chung, 1993)

$$\mathbf{m} \cdot \ddot{\mathbf{d}}_G + \mathbf{c} \cdot \dot{\mathbf{d}}_G + \mathbf{k} \cdot \mathbf{d}_G = \mathbf{F}\mathbf{e} \quad (3.11)$$

where matrices \mathbf{m} , \mathbf{c} and \mathbf{k} are listed in Appendix B.

3.5.2 State Space Model

Since the dynamometer is a MIMO system, its analysis can be greatly simplified by introducing the state space model (Franklin et al., 1994). One of the main reason of using the state space model in this research is the availability of commercial software packages, such as Matlab[®] or Mathematica[®], that allow easy implementation of complicated algorithms developed in control theory for the investigation of MIMO system.

To transform the equation of motion into the state space form, Eq. 3.11 is first rearranged as follows

$$\ddot{\mathbf{d}}_G = -\mathbf{m}^{-1}\mathbf{k} \cdot \mathbf{d}_G - \mathbf{m}^{-1}\mathbf{c} \cdot \dot{\mathbf{d}}_G + \mathbf{m}^{-1} \cdot \mathbf{F}\mathbf{e} \quad (3.12)$$

Next, by introducing an auxiliary vector \mathbf{d}_{GD} such that $\mathbf{d}_{GD} = \dot{\mathbf{d}}_G$, can be written as (Peters and Mergeay, 1976)

$$\dot{\mathbf{q}} = \mathbf{A} \cdot \mathbf{q} + \mathbf{B} \cdot \mathbf{u} \quad (3.13)$$

$$\mathbf{d}_G = \mathbf{C} \cdot \mathbf{q} + \mathbf{D} \cdot \mathbf{u} \quad (3.14)$$

where

\mathbf{d}_G - the vector containing 6 generalized coordinates,

\mathbf{q} - the vector of state variables defined as

$$\mathbf{q} = [\mathbf{d}_G \quad \mathbf{d}_{GD}]^T \quad (3.15)$$

The dynamic matrix \mathbf{A} and control matrix \mathbf{B} contain elements of the matrices \mathbf{m} , \mathbf{c} and \mathbf{k} from the equation of motion model, Eq. 3.11. Their form is

$$\mathbf{A} = \begin{bmatrix} \mathbf{O}_{6 \times 6} & \mathbf{I}_{6 \times 6} \\ -\mathbf{m}^{-1} \cdot \mathbf{k} & -\mathbf{m}^{-1} \cdot \mathbf{c} \end{bmatrix} \quad (3.16a)$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{O}_{6 \times 6} \\ \mathbf{m}^{-1} \end{bmatrix} \cdot \mathbf{C} \mathbf{u} \quad (3.16b)$$

where $\mathbf{I}_{6 \times 6}$ is a 6×6 identity matrix and $\mathbf{O}_{6 \times 6}$ is a 6×6 zero matrix.

The observation matrix \mathbf{C} and the direct transmission matrix \mathbf{D} represent relationships between \mathbf{d}_G and \mathbf{q} , as well as, \mathbf{d}_G and \mathbf{u} , respectively. If the output of the state space equation is desired to be the generalized coordinates vector, \mathbf{d}_G , the form of matrix \mathbf{C} is given by Eq. 3.16c. Since no direct relationship exists between the generalized coordinates and the input vector \mathbf{u} , the matrix \mathbf{D} is a zero matrix.

$$\mathbf{C} = [\mathbf{I}_{6 \times 6} \quad \mathbf{O}_{6 \times 6}]^T \quad (3.16c)$$

$$\mathbf{D} = \mathbf{O} \quad (3.16d)$$

3.5.3 Transfer Function Model

The state space model can be transformed into the transfer function model by taking Laplace transformation of Eq. 3.13 and 3.14. Under the assumption of the linearity and time-invariance, the matrices **A**, **B**, **C**, and **D** are considered as constant matrices.

For zero initial conditions a new set of equations is obtained

$$s \cdot \mathbf{Q}(s) = \mathbf{A} \cdot \mathbf{Q}(s) + \mathbf{B} \cdot \mathbf{U}(s) \quad (3.17a)$$

$$\mathbf{D}_G(s) = \mathbf{C} \cdot \mathbf{Q}(s) \quad (3.17b)$$

where

$\mathbf{Q}(s)$ is the Laplace transform of the state vector, $\mathbf{q}(t)$,

$\mathbf{D}_G(s)$ is the Laplace transform of the generalized coordinates vector, $\mathbf{d}_G(t)$.

By applying rules of linear algebra, the above equations can be converted into the following input-output form.

$$\mathbf{G}_E(s) = \mathbf{D}_G(s)/\mathbf{U}(s) = \mathbf{C} \cdot (s \cdot \mathbf{I}_{12 \times 12} - \mathbf{A})^{-1} \cdot \mathbf{B} \quad (3.18)$$

The above equation encapsulates the relationship between the input vector (external forces and moments) and the output vector (generalized coordinates) in the transfer function matrix \mathbf{G}_E . There are several methods to generate responses to actual input signals by using the transfer function model of the dynamometer. One approach that is conceptually simple involves the following equation

$$\mathbf{d}_m(t) = \mathbf{L}^{-1} [\mathbf{G}_E(s) \cdot \mathbf{U}_m(s)] \quad (3.19)$$

where

$\mathbf{U}(s)$ - Laplace transform of the actual input signal $u(t)$,

$\mathbf{d}_m(t)$ - generated model based response,

\mathbf{L}^{-1} - inverse Laplace transform operator.

Commercial software packages such as Matlab[®] or Mathematica[®] allow quick and easy calculation of these responses.

The state space model of the dynamometer presented above is used in Section 5.4 to generate the response vector \mathbf{d}_m . The input \mathbf{u}_m is obtained using an actual force signal from an impact hammer in the experiment. The calculated and recorded responses are compared and their differences are discussed. A Matlab[®] program used to generate response of the dynamometer's model to the actual excitation is listed in Appendix C.

3.6 Closure

The model based vibration visualization proposed in this research is performed in two stages. The first stage is the signal based visualization in which vibrations of a mechanical system are measured and recorded by means of accelerometers and presented in 3-D graphical animation. In the second stage, the mathematical model of the visualized system is used to detect errors that can occur in the first stage. A MIMO model of the actual system is applied to generate estimated responses to the actual excitation signals. The system identification technique is used to assure high fidelity of the model.

A dynamometer is used as an example dynamic system. Its model developed in previous researches (Chung, 1993; Chen, 1996) is employed. This model obtained by means of Lagrange's method treats the actual dynamometer as a linear time invariant, MIMO system. Details of a methodology developed for the signal based vibration visualization applied to a rigid plate are explained in the next chapter. The application of this methodology and its experimental evaluation are presented in Chapter 5.

CHAPTER 4

THE SIGNAL BASED VIBRATION VISUALIZATION

The underlying theory and algorithms employed in the signal based vibration visualization are presented in this chapter. Convenient coordinate systems and several crucial procedures are discussed in detail. The entire visualization process is partitioned into four major steps, namely: 1) data acquisition procedure, 2) signal processing, 3) generalized coordinate calculation, and 4) 3-D animation procedure.

4.1 Visualization of Machine Vibrations

The visualization of machine vibrations presented below assumes that each specific part of the machine is a rigid plate with six degrees of freedom (6-DOF) motion. This means that six variables, three for translation and three for rotation, are required to describe three dimensional (3-D) motion of the plate. These variables are formed into the generalized coordinate list, \mathbf{d}_G , in Eq. 4.1. Graphical representation of this list is shown in Fig. 4.1.

$$\mathbf{d}_G = [x \ y \ z \ \theta \ \phi \ \psi]^T \quad (4.1)$$

One generalized coordinate list represents one specific location of the plate in space and is used to generate one picture of the plate in the 3-D space. Hence, a time series of coordinate lists is needed to create a sequence of pictures that represent motion of the vibrating plate. A technique to calculate the generalized coordinate lists from experimental data is discussed below. A method employed to present the results as 3-D graphical animation is also described.

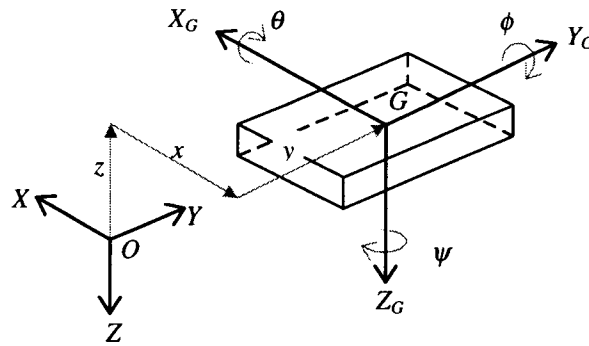


Figure 4.1 Components of the generalized coordinate list, \mathbf{d}_G , describing the 'rigid-body' motion of a plate.

4.1.1 Overview of the Methodology

A flowchart of the algorithm used in the visualization is shown in Fig. 4.2. First, a simplified mechanical model is developed by dividing the machine into a suitable number of rigid solids, (e.g., plates). For example, a simplified machine shown in Fig. 3.11 comprises four prismatic solids representing the column, spindle assembly, table, and the dynamometer's platform with workpiece. Multiple sensors are installed on the machine to measure vibrations of each prismatic solid.

The solution employed in this research requires nine accelerometers per each solid to measure its list of generalized coordinates (Padgaonkar, 1975). Acceleration signals from these transducers are recorded by a data acquisition (DAQ) system which is controlled by a LabVIEW[®]-based *Data Acquisition Controller* program (see Fig. 4.4a).

In the second step the voltage signals from accelerometers are: (1) amplified, (2) low-pass filtered (anti-aliasing), and (3) converted to acceleration using respective transducer's sensitivities. Average quasi-static offset voltage having its source in thermal drift in each sensor is eliminated by subtracting the average (or moving average) value calculated for each signal. Adjusted signals are numerically integrated twice to obtain displacement signals. Since signals from piezoelectric accelerometers are affected by

drift and noise, direct integration of these signals yields strongly distorted results. Therefore, digital high-pass filters are employed to remove quasi-static errors.

Angular components of the generalized coordinate list, \mathbf{d}_G , are then calculated from the processed displacement signals using equations proposed by Padgaonkar et al. (1975), as further explained in Section 4.5. Finally, a sequence of generalized coordinate lists representing consecutive ‘snap-shots’ is generated for the animation of motion.

The animation procedure uses the homogeneous coordinate transformation (Wolovich, 1987) to draw 3-D wireframe pictures that present the consecutive positions of visualized solid. Rapid displaying of pictures generated from each data point (6 coordinates) creates the effect of animated motion. Detailed discussion of animation is presented in Section 4.6.

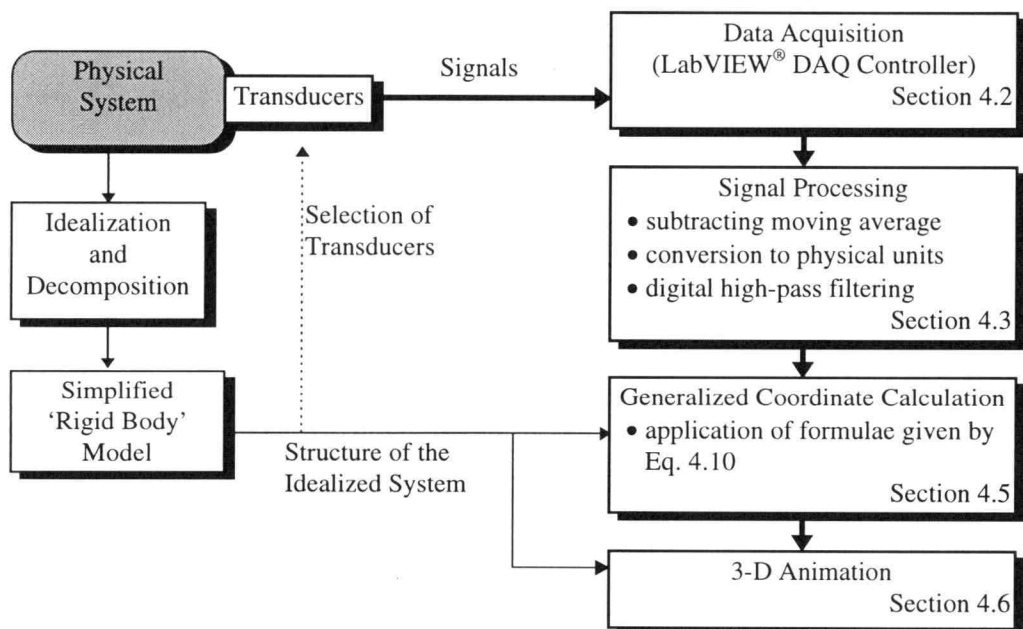


Figure 4.2 Flowchart of the methodology used for the visualization of machine vibrations.

4.1.2 Introduction to LabVIEW® Programming Environment

LabVIEW® is a program development application that uses graphical programming language, G, to create programs in a flow-chart-like form, called a block diagram (National Instrument, 1994). Unlike other programming languages such as C, FORTRAN or BASIC, LabVIEW® relies on graphic symbols rather than textual statements to describe programming actions (National Instruments, 1994). These symbols (icons) such as, switches, knobs, and displays represent and resemble elements found in physical instruments. Because of its appearance, the LabVIEW® programs are called *virtual instruments* (VI).

LabVIEW® offers extensive libraries of functions and subroutines, for example, application-specific libraries for data acquisition, data analysis, data presentation, and data storage. In data acquisition aspect, LabVIEW® can command DAQ boards to read analog input signals (with A/D conversion), generate analog output signals (D/A conversion), read voltage data for processing or other manipulation, and write digital signals into storage devices (National Instruments, 1994).

A typical virtual instrument (VI) consists of two parts, a *front panel* and a *block diagram*, displayed in two separate windows. In the panel window a user interfaces with the VI via *Controls* and *Indicators*. The Controls simulate instrument input devices and supply data to the block diagram of the VI. The Indicators simulate instrument output devices that display data acquired or generated by the block diagram of the VI. In the block diagram window the user constructs a graphical source code of the VI by wiring together objects (represented by icons) that send or receive data, perform specific functions, and control the flow of execution. Fig. 4.3 shows an example of the VI (variable amplitude sinewave generator) which displays a knob and graph in the front panel (a) while the calculations take place in the block diagram (b).

All visualization programs in this research were written in LabVIEW[®]. The developed set of programs consists of the following major VIs:

1. “Data Acquisition Controller.VI”. This program facilitates acquisition of up to 16 vibration signals from transducers attached to a machine. It is further discussed in Section 4.2.
2. “Signal Processor.VI”. This program performs the signal conversion, subtraction of the moving average, filtering, and numerical integration on the acquired data. Calculations used in this VI are further discussed in Section 4.3.
3. “Generalized Coordinate Calculator.VI”. This program calculates a sequence of the generalized coordinate lists, $\mathbf{d}_G(t)$, introduced in Section 4.1. Each list is calculated from nine linear acceleration signals recorded at any given time instant for each prismatic solid representing an element of the simplified machine. Equations employed in the calculations are further explained in Section 4.5.
4. “3-D Animation Generator.VI”. This program generates 3-D pictures using a time series of coordinate lists $\mathbf{d}_G(k)$, $k=1,2,\dots,m$; m is the number of data. The animation is achieved by rapid display of these pictures.

The above programs can perform their functions individually. Alternatively, each of them can work as a subprogram that can be run and interfaced from other programs. Figure 4.4 shows the front panels these VIs. Program descriptions and usage are explained in Chapter 5 and in Appendix D.

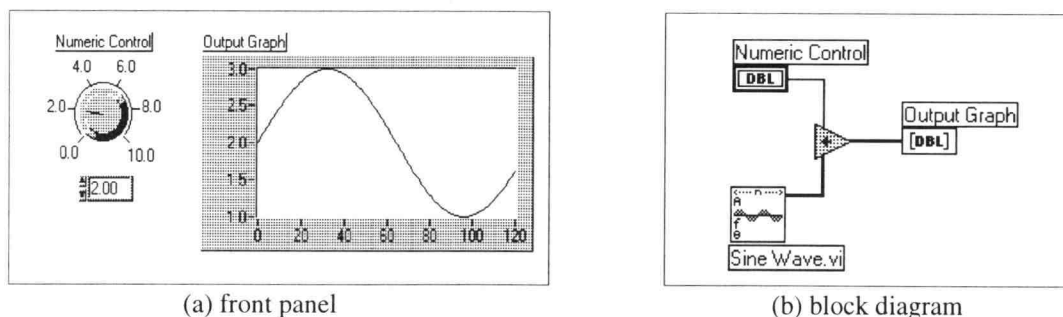
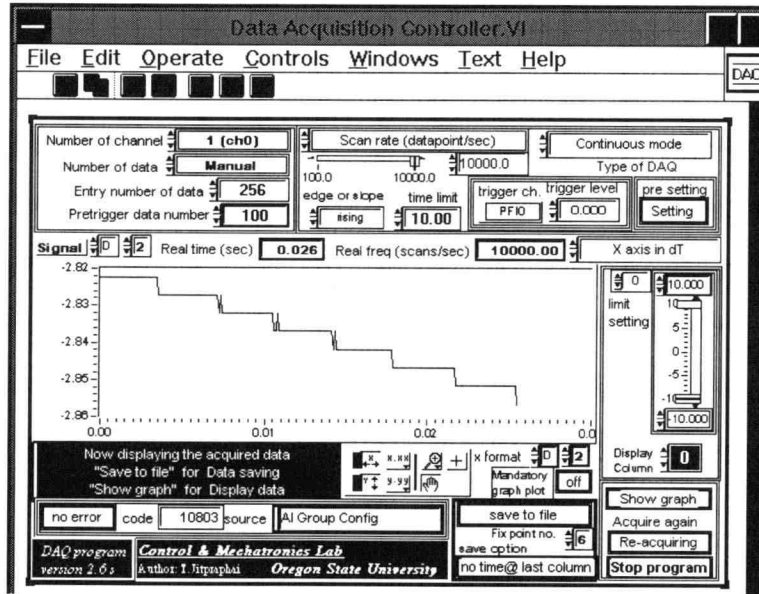
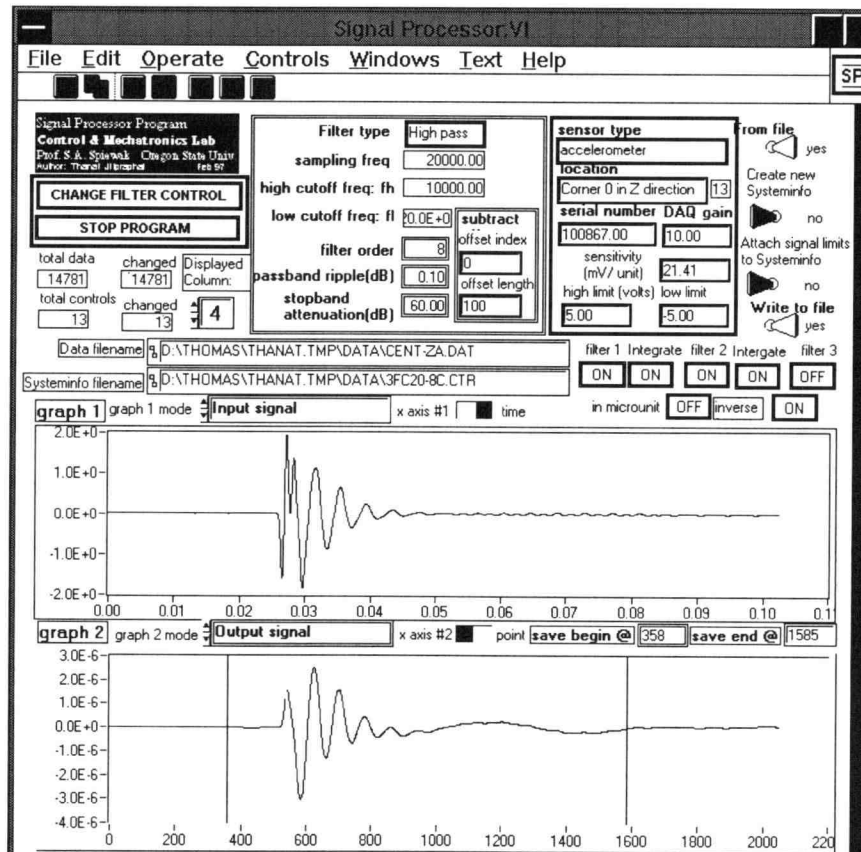


Figure 4.3 An example program (virtual instrument) in LabVIEW[®].

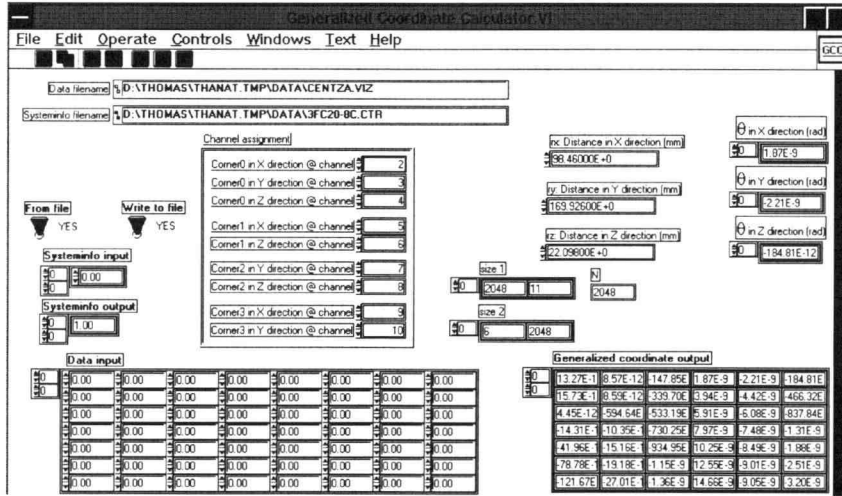


(a) Data Acquisition Controller.VI

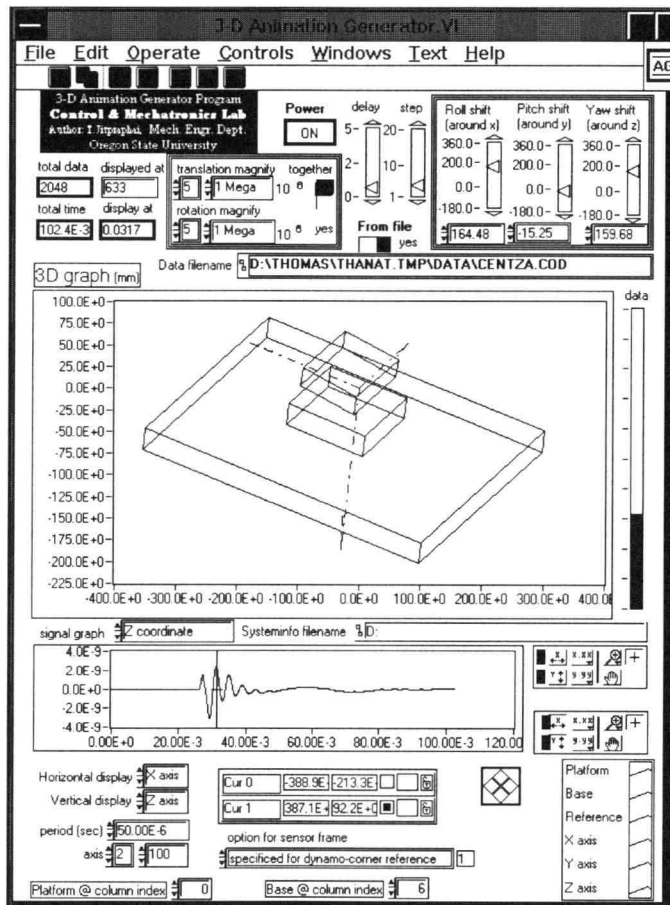


(b) Signal Processing (double integral) ii.vi

Figure 4.4 Front panels of LabVIEW[®] programs developed for the vibration visualization.



(c) Generalized Coordinate Calculator.VI



(d) 3-D Animation Generator.VI

Figure 4.4 (continued) Front panels of LabVIEW® programs developed for the vibration visualization.

4.2 Data Acquisition System Used in Vibration Visualization

A standard data acquisition (DAQ) system comprises the following basic components: (1) a controller, (2) a signal conditioner, (3) a multiplexer and amplifier, (4) an analog-to-digital converter (ADC), (5) a storage unit or a memory unit, and (6) a readout device (Dally et al.,1993). The DAQ system employed in this research is a computer-based instrument. The control part is implemented in LabVIEW[®] program. The memory part and the readout device are added as a desktop computer. The ADC, multiplexer and amplifier are provided in a plug-in DAQ printed circuit board type AT-MIO16E2 from National Instruments (1994). Low-pass filters serve as signal conditioners to prevent signal aliasing. A schematic diagram of the employed DAQ system is shown in Fig. 4.5. Further details are given in Section 5.1.

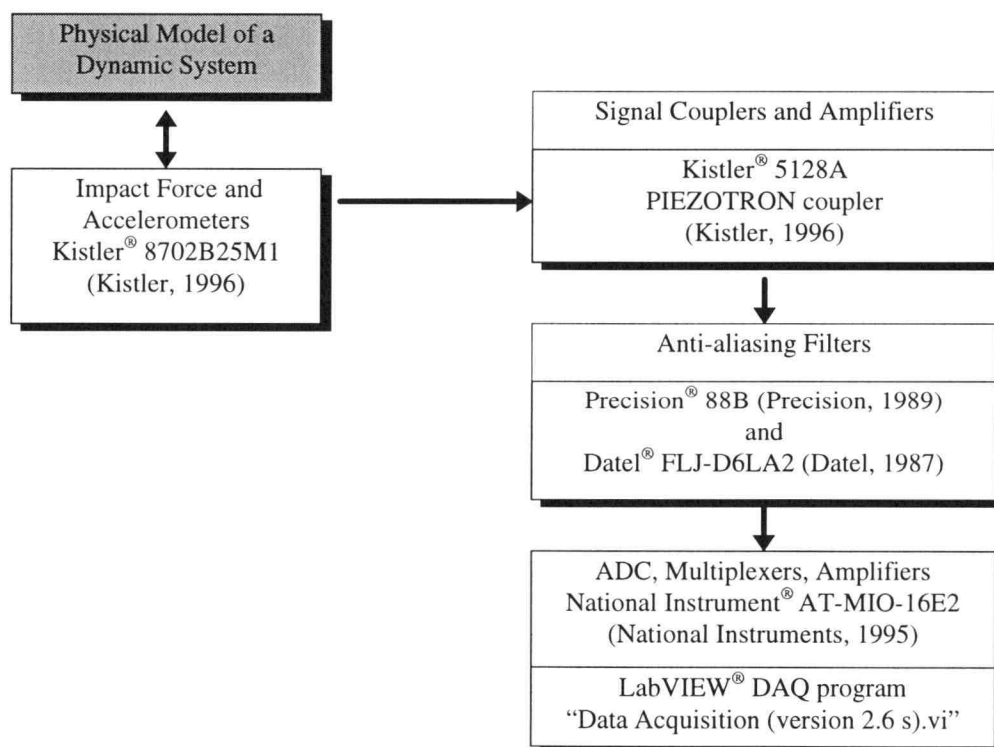


Figure 4.5 A block diagram of the basic data acquisition system used in this research.

4.2.1 Anti-aliasing Filtering

The '*aliasing*' phenomenon can occur when converting an analog signal to a digital data with sampling frequency lower than the minimum required rate. To prevent this phenomenon, a minimum sampling rate should be set at least twice of the highest frequency component of the measured signal, or the Nyquist frequency (f^*). Practically, the sampling frequency (f_s) is set to be $2.56 \cdot f^*$ (Doebelin, 1990).

Anti-aliasing filters are employed to assure the aliasing prevention regardless the actual spectral contents of recorded signals. They are analog low-pass filters with the high cut-off frequency equal to f^* , which reduce signal components that may produce aliasing to an insignificant level. Anti-aliasing filters are useful for primary signal conditioning. However, these devices can severely distort transient signals with high frequency components. The sampling rate should be carefully selected for a specific high-frequency measurement to prevent distortion on one hand and aliasing of the signal on the other (Doebelin, 1990).

4.2.2 Attenuation of Noise in a Data Acquisition

The noise introduced into measured signals by imperfect sensors causes distortion on the data used for visualization of motion. A suitable hardware configuration of DAQ can decrease the level of the noise. This section provides information about sources of signal disturbances and some techniques to eliminate them.

Drift is defined as gradual changes in the sensor's output or in the amplifier offset voltage and bias current resulting from temperature, time and line voltage that can be minimized by keeping temperature and/or line voltage nearly constant. The effect of drift can also be reduced using an ac coupling acting as a high-pass filter or compensating the drift with automatic zero balancing techniques or subtracting average in digital systems

(Doebelin, 1990). Cable shielding and grounding design within the instruments should also be taken into account since inductive pickup, electrostatic pickup and ground loops can cause large error voltage. Any conductor carrying alternative current generates magnetic field that can produce interfering voltages to signal wires by the inductive pickup. These effects can be reduced by removing equipment such as power lines, motors, and transformers from the neighborhood of sensitive signal circuits (Doebelin, 1990). The use of an enclosing conductive shield can further decrease electrostatic pickup. The shield captures unintentional charges and drains them off to a satisfactory ground. However, a shield required for a magnetically induced noise is quite thick and is not practically used. The inductive noise can be minimized by twisting the two signal conductors so that the loop area available for inducing error voltages is balanced providing a canceling effect. Commercially available cables include twisted conductors, wrapped foil shields and grounding drain wire (Deobelin, 1990).

A ground loop is created by connecting a signal circuit to more than one ground. Since a conductor that serves as ground generally carries currents and also has some resistance, potentials between two points (grounds) are not identical. Any potential difference produces current that flows through the shield or signal circuit yielding large noise voltage. The ground loop is simply broken by grounding a shield at only one point and using a floating-input (isolated) amplifier to break the other loop which results in great reduction of noise pickup (Doebelin, 1990; Potter, 1992).

Tribo-electric noise is often induced into the accelerometer cable by mechanical motion of the cable itself. It originates from local capacity and charge changes due to dynamic bending, compression and tension of the layers making up the cable. This problem is avoided by using a proper graphite accelerometer cable and taping or gluing the cable down to a surface as close to the accelerometer as possible (Brüel & Kjær, 1982).

4.2.3 Data Acquisition Program

A controller program for data acquisition (DAQ) is required to read data from transducers for further use. “DAQ Controller.VI” written in the LabVIEW®’s G language is an interface program between the user and a DAQ board AT-MIO 16E2 (National Instrument, 1995) employed for the digitization of signals. The user is able to command the board to acquire analog voltage signals with desired parameters. The user can easily inspect the acquired signals, select suitable sampling parameters for these signals and acquire data again. A front panel of this program is shown in Fig. 4.4a.

4.3 Signal Processing in Vibration Visualization

Signal processing procedure transforms signals acquired from nine accelerometers (for each rigid plate referred to in Section 4.1) into the generalized coordinate list, \mathbf{d}_G , in Eq. 4.1. The signal processing steps are summarized in Fig. 4.6.

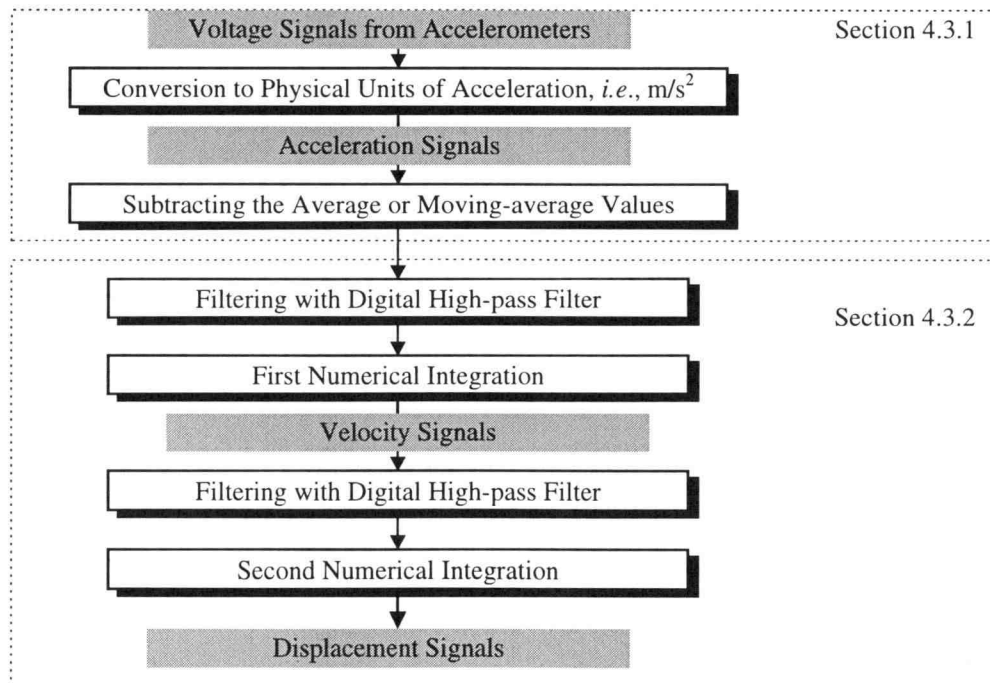


Figure 4.6 A flowchart of signal processing in vibration visualization.

4.3.1 Conversion to Physical Units and Subtraction of the Average Value

Voltage signals obtained from accelerometers require adjustment to represent the actual acceleration values. A factor (number) needed to obtain these actual values is determined from the frequency response function (FRF) of the system. For the piezoelectric transducer, the amplitude of an acceleration acting on the sensor, A_o , is related to the measured output signal of the sensor, V_o , as follows (see Eq. 2.2).

$$A_o = \frac{1}{s_v} \cdot \frac{1}{H_L(\omega)} \cdot V_o \quad (4.2)$$

A calculation of these values requires a knowledge of the sensitivity, s_v , of each employed accelerometer. The sensitivity, s_v , is provided by the sensor's manufacturer in units of mV/g. Practically, the signal available directly on the output from the transducer is weak. An amplifier with gain G_f is therefore included in the data acquisition system to amplify the signal. Within measuring frequency range, *i.e.*, the working region in Fig. 2.12, the FRF can be assumed to be unity, *i.e.*, $H_L(\omega) = 1$. Therefore, Eq. (4.2) can be rewritten as

$$A_o = \frac{1000 \cdot k_A}{s_v} \cdot \frac{1}{G_f} \cdot V_r \quad (4.3)$$

where

- A_o - the amplitude of the actual acceleration (m/s^2),
- V_r - the voltage recorded by the data acquisition (DAQ) program (Volts),
- s_v - the accelerometer sensitivity provided by the manufacturer (mV/g),
- G_f - the gain of the amplifier in the anti-aliasing filter⁶, and
- k_A - the conversion factor, $k_A = 9.806 \text{ m/s}^2\text{g}$.

The voltage recorded by the data acquisition (DAQ) system is not necessarily equal to zero when no acceleration is applied to the transducer. This is a result of noise in the DAQ system. To eliminate DC component of this error, which is strongly amplified

⁶ The gain of anti-aliasing filters is adjustable (see Chapter 5).

by the signal integration, an average value, \bar{V}_r , of each signal, V_r , is subtracted from it. The obtained 'zero centered' signal is henceforth denoted x_l (see Eq. 4.4).

4.3.2 High-pass Filtering and Double Integration Procedure

The measurement of position (required to visualize vibrations) by means of accelerometers has one serious drawback, namely high sensitivity to low frequency noise superimposed on the measured accelerations. To overcome this drawback a suitable digital signal processing is proposed in this research in addition to the 'zero centering' procedure described in the preceding Section.

A priori knowledge of the system's model allows to predict likely characteristics of the measured signals at low frequency for any specific excitation. By selecting excitations which do not have low frequency components, the system of interest should not have low frequency components in responses to these excitations. Such components in the measured signals are treated as errors and eliminated. The correction of signal is performed using digital high-pass filters. The cutoff frequency of these filters is determined by analyzing signals that are acquired when the system is not excited. Power spectra of these signals clearly indicate the range of low frequency errors and aid in choosing suitable cut-off frequencies. Recursive digital filters (type infinite impulse response-IIR) are employed. Because of sharp drop of the frequency response function (FRF) at the cut-off frequency, Elliptic filters are selected (National Instruments, 1994). They are described by the following formula.

$$x_2[i] = \frac{1}{R_0} \cdot \left[\sum_{j=0}^{N_R-1} F_j \cdot x_1[i-j] - \sum_{k=1}^{N_F-1} R_k \cdot x_2[i-k] \right], \quad i = 0, 1, 2, \dots, m-1 \quad (4.4)$$

where

- $x_1[n]$ - the n^{th} element of a sequence of the discrete input signal (acceleration),
- $x_2[n]$ - the n^{th} element of a sequence of the filtered output signal (acceleration),
- R_k - the k^{th} reverse coefficient,

- F_j - the j^{th} forward coefficient,
 N_R - the number of the reverse coefficients,
 N_F - the number of the forward coefficients, and
 m - the number of data, *i.e.*, number of elements in x_1 and x_2 .

The number and values of the reverse and forward coefficients depend on types and orders of filters.

After removing low frequency noise from acceleration signals, a numerical integration (National Instruments, 1994) is performed to calculate velocities. The integral, $x_3[i]$, of an input time series signal, $x_2[i]$, is calculated as (National Instruments, 1994)

$$x_3[i] = \frac{1}{6} \sum_{j=0}^i (x_2[j-1] + 4x_2[j] + x_2[j+1]) \cdot \Delta t, \quad i = 0, 1, 2, \dots, m-1 \quad (4.5)$$

where

- $x_2[n]$ - the n^{th} element of a sequence of the discrete acceleration signal (input),
 $x_3[n]$ - the n^{th} element of a sequence of the velocity signal,
 Δt - the sampling period, calculated from inverse of the sampling frequency.

Before performing the second integration of the velocity signal, the Elliptic filter is applied to this signal again. A position signal is then obtained by integrating the filtered velocity signal using the above integration method.

$$x_4[i] = \frac{1}{R_0} \cdot \left[\sum_{j=0}^{N_R-1} F_j \cdot x_3[i-j] - \sum_{k=1}^{N_F-1} R_k \cdot x_4[i-k] \right] \quad (4.6)$$

$$x_5[i] = \frac{1}{6} \sum_{j=0}^i (x_4[j-1] + 4x_4[j] + x_4[j+1]) \cdot \Delta t, \quad i = 0, 1, 2, \dots, m-1 \quad (4.7)$$

where

- $x_4[n]$ - the n^{th} element of a sequence of the filtered velocity signal,
 $x_5[n]$ - the n^{th} element of a sequence of the displacement signal.

Figure 4.7 graphically illustrates the signal processing described above. A signal processing function, SPF , is henceforth introduced to represent all operations performed on the signals available as outputs of the accelerometers.

$$x_5 \equiv SPF\{V_r\}, \quad (4.8)$$

where

$SPF\{\bullet\}$ - the processing steps that transform the readout voltage, V_r , into the displacement values, x_5 .

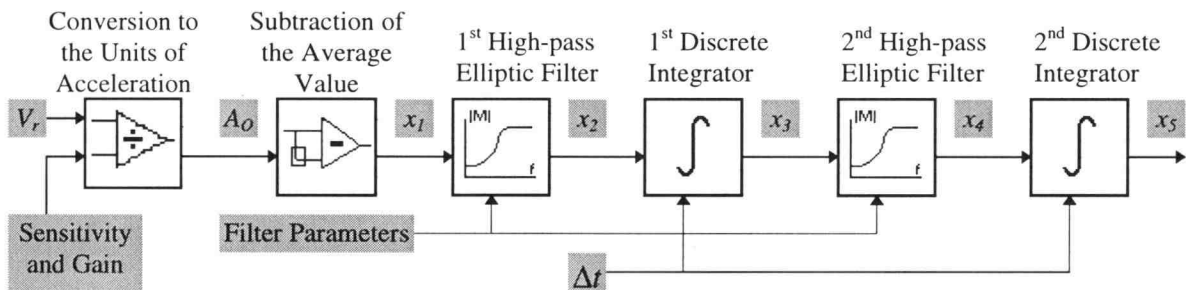


Figure 4.7 Diagram of the signal processing procedure.

4.4 Coordinate Systems

Several variables, vectors and coordinate systems relevant to the vibration visualization are defined in this Section. To avoid confusion, a brief explanation of the terminology used henceforth is warranted. The explanation begins with the Coordinate Systems (C.S.) shown in Fig. 4.8 (Spiewak, 1994).

1. Reference Coordinate System of the Plate $(XYZ)_R$

This C.S. is associated with the initial position of the plate. The origin of $(XYZ)_R$ is at the corner C_R of the plate and the three orthogonal axes, X_R , Y_R and Z_R , are parallel to the plate's edges.

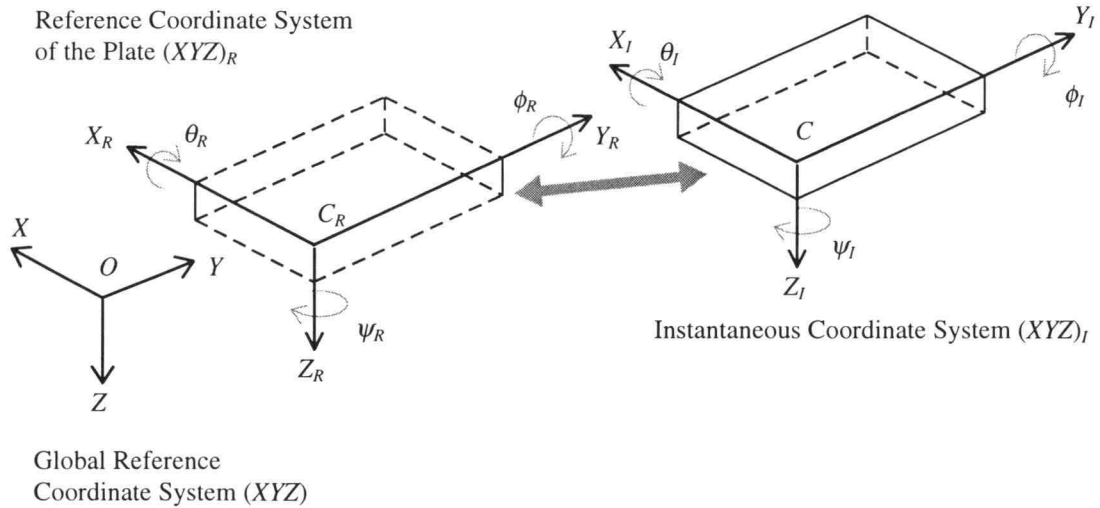


Figure 4.8 Coordinate systems used in describing the plate motion.

2. Instantaneous Coordinate System $(XYZ)_I$

Vibrating plate assumes varying translations and rotations. The C.S. $(XYZ)_I$ moves together with vibrating plate. Therefore, the coordinates of characteristic points of the plate (its corners) in the $(XYZ)_I$ system remain unchanged. Consequently, converting these coordinates into the reference coordinate $(XYZ)_R$ system is easy.

3. Global Reference Coordinate System (XYZ)

The above coordinate systems suffice to visualize the motion of one only plate. However, if multiple plates are involved, it is advantageous to introduce one global coordinate system. With this better system, each plate has its unique Reference and Instantaneous coordinate systems.

Each individual Reference C.S. describes the initial location of the plate associated with it. This location is defined by a six component list, \mathbf{d}_O , whose first three components are coordinates of the origin C_R of the Reference C.S. given in the Global Reference C.S., the three remaining components are angles that define the spatial rotation

of the Reference C.S. with respect to the Global Reference C.S. The entire list \mathbf{d}_O is defined as

$$\mathbf{d}_O = [l_O \ w_O \ h_O \ \theta_O \ \phi_O \ \psi_O]^T \quad (4.9)$$

4.5 Calculation of the Generalized Coordinates from Experimental Data

The shape of objects whose motion is to be visualized imposes constraints upon the mounting locations of the accelerometers. In the case under consideration, *i.e.*, a plate, accelerometers can not be easily mounted at the center of mass. Since they can be easily mounted at the corners of the plate (see Fig. 5.3), it is advantageous to redefine the generalized coordinates describing the instantaneous spatial location of the plate.

The generalized coordinates introduced in Chapter 3, *i.e.*, \mathbf{d}_G , that are defined at the center of mass of the plate are now transformed to new coordinates that are defined at one specific corner point, C , of the plate⁷. The first three components of the new list of generalized coordinates, \mathbf{d}_I , define the position of point C in the reference C.S. $(XYZ)_R$. Three positions are designated x_I , y_I and z_I . The remaining three components define orientation of the plate with respect to $(XYZ)_R$. These later components are designated θ_I , ϕ_I , and ψ_I . Thus the entire new list of generalized coordinates is

$$\mathbf{d}_I = [x_I \ y_I \ z_I \ \theta_I \ \phi_I \ \psi_I]^T \quad (4.10)$$

The values of x_I , y_I , and z_I can be obtained by integrating twice the filtered acceleration signals, as discussed in Section 4.3.2. The rotations θ_I , ϕ_I , and ψ_I , are calculated according to a method proposed by Padgaonkar et al. (1957) briefly summarized below.

⁷ This corner point coincides with the origin of the coordinate system $(XYZ)_I$ in Fig. 4.8.

For a rigid body, the relative acceleration of a point P is given by the formula (Hibbeler, 1995).

$$\mathbf{a}_P = \mathbf{a}_C + \boldsymbol{\alpha} \times \mathbf{r} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}) \quad (4.11)$$

where

P - the arbitrary point on a rigid plate, in this case one of the corners; $P = 1, 2$ and 3 , shown in Fig. 4.9,

\mathbf{a}_P - the acceleration of point P in $(XYZ)_R$,

\mathbf{a}_C - the acceleration of point C in $(XYZ)_R$,

$\boldsymbol{\omega}$ - the angular velocity of point P in $(XYZ)_I$,

$\boldsymbol{\alpha}$ - the angular acceleration of point P in $(XYZ)_I$, and

\mathbf{r} - the position vector of point P from the origin, C , of $(XYZ)_I$.

By applying Eq. 4.7 to coordinates of the plate corresponding to points 1, 2, and 3 in Fig. 4.9, the following equations are obtained (Padgaonkar et al., 1975).

$$\alpha_x = (a_{z1} - a_{zC})/r_y - \omega_y \cdot \omega_z \quad (4.12a)$$

$$\alpha_y = -(a_{z2} - a_{zC})/r_x + \omega_x \cdot \omega_z \quad (4.12b)$$

$$\alpha_z = (a_{y2} - a_{yC})/r_x - \omega_x \cdot \omega_y \quad (4.12c)$$

$$\alpha_x = -(a_{y3} - a_{yC})/r_z + \omega_y \cdot \omega_z \quad (4.12d)$$

$$\alpha_y = (a_{x3} - a_{xC})/r_z - \omega_x \cdot \omega_z \quad (4.12e)$$

$$\alpha_z = -(a_{x1} - a_{xC})/r_y + \omega_x \cdot \omega_y \quad (4.12f)$$

where

α_i - the angular acceleration component of the vector $\boldsymbol{\alpha}$ around the i axis; $i = x, y,$

$z,$

ω_i - the angular velocity component of the vector $\boldsymbol{\omega}$ around the i axis; $i = x, y, z.$

r_x, r_y, r_z - the distances between accelerometers shown in Fig. 4.9.

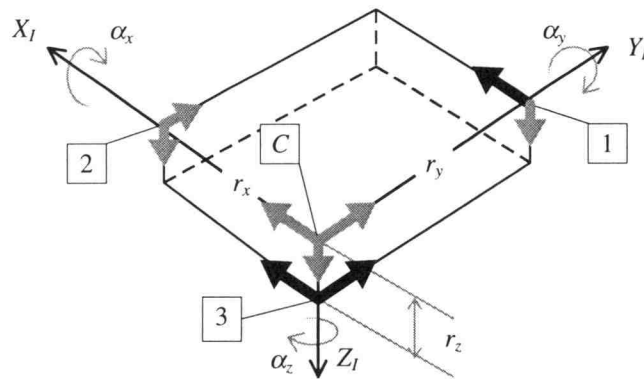


Figure 4.9 Locations of nine accelerometers required for the calculation of the generalized coordinates (Padgaonkar et al., 1975).

A computation of the linear and angular accelerations \mathbf{a} and $\boldsymbol{\alpha}$, according to equations 4.9a, 4.9b, and 4.9c, requires, at minimum, six linear accelerometers. These accelerometers are represented in Fig. 4.9 by gray thick arrows. Positive sense of each accelerometer is indicated by the arrow's heads. However, if only six accelerometers are used, it is necessary to solve nonlinear (due to the products of the angular velocities, $\boldsymbol{\omega}_i$) differential equations, which is time consuming. Worse, the solution is non-unique (Padgaonkar, 1975; Liu, 1976). Padgaonkar (1975) showed that the angular accelerations, $\boldsymbol{\alpha}$, can alternatively be calculated by adding three more accelerometers, mounted on the plate as shown by the solid thick arrows in Fig. 4.9. The addition of new equations (Eq. 4.12c, 4.12d and 4.12e) leads to the following linear relationships for the angular accelerations due to changes in angles, θ_i , ϕ_i , and ψ_i , in Eq. 4.9.

$$\alpha_x = (a_{z1} - a_{zC})/2 \cdot r_y - (a_{y3} - a_{yC})/2 \cdot r_z \quad (4.13a)$$

$$\alpha_y = (a_{x3} - a_{xC})/2 \cdot r_z - (a_{z2} - a_{zC})/2 \cdot r_x \quad (4.13b)$$

$$\alpha_z = (a_{y2} - a_{yC})/2 \cdot r_x - (a_{x1} - a_{xC})/2 \cdot r_y \quad (4.13c)$$

where

$$\alpha_x = \frac{d}{dt^2} [\theta_1(t)]^2; \quad \alpha_y = \frac{d}{dt^2} [\phi_1(t)]^2; \quad \alpha_z = \frac{d}{dt^2} [\psi_1(t)]^2.$$

The signal processing procedure used in this research generates linear displacements by double integration of the signals from accelerometers as described in Section 4.3. Therefore, the instantaneous translations, x_I , y_I and z_I are obtained by applying the signal processing function, $SPF\{\bullet\}$ (Eq. 4.8) to the accelerations, a_{ij} ; $i = x, y$ and z ; $j = C, 1, 2$, and 3 . It is symbolically written as

$$x_I = x_{xC} = SPF\{a_{xC}\} \quad (4.14a)$$

$$y_I = x_{yC} = SPF\{a_{yC}\} \quad (4.14b)$$

$$z_I = x_{zC} = SPF\{a_{zC}\} \quad (4.14c)$$

The roll (θ_I), pitch (ϕ_I), and yaw (ψ_I) angles are also calculated using displacements obtained by processing signals from the all nine accelerometers on the plate.

$$\theta_I = (x_{z1} - x_{zC})/2 \cdot r_y - (x_{y3} - x_{yC})/2 \cdot r_z \quad (4.14d)$$

$$\phi_I = (x_{x3} - x_{xC})/2 \cdot r_z - (x_{z2} - x_{zC})/2 \cdot r_x \quad (4.14e)$$

$$\psi_I = (x_{y2} - x_{yC})/2 \cdot r_x - (x_{x1} - x_{xC})/2 \cdot r_y \quad (4.14f)$$

where

$$x_{ij} = SPF\{a_{ij}\}$$

- displacement obtained by double integration of the acceleration a_{ij} ,

i, j - the notation for axis, $i = x, y$ and z ; $j = C, 1, 2$, and 3 (corners of the plate where the accelerometers are located).

4.6 Animation of the Rigid Body Motion

A knowledge of the list of generalized coordinates, \mathbf{d}_I , given by Eq. 4.10 associated with the instantaneous position of the selected point (corner C) of the plate allows calculation of the remaining corners in the C.S. $(XYZ)_R$. on the condition that the plate's dimensions are defined in the C.S. $(XYZ)_I$.

The animation procedure consists of five steps: (1) finding the absolute position of the reference corner point, C , (2) calculating coordinates of the plate's corners in the C.S. (XYZ) using homogeneous coordinate transformation, (3) modification of the corner coordinates according to the chosen viewing point, (4) drawing a single 3-D picture representing instantaneous position of the plate, and (5) animation of the 3-D pictures. These steps are discussed in the following subsections.

4.6.1 Finding Absolute Position of the Reference Corner Point

Data obtained from the *Generalized Coordinate Calculator* (see Section 4.1.2) defines only the relative position and orientation of the C.S. $(XYZ)_I$ with respect to C.S. $(XYZ)_R$ in Fig. 4.8. Furthermore, the position and orientation of the reference C.S. $(XYZ)_R$ in the global C.S. (XYZ) is defined by additional six variables contained in the list \mathbf{d}_O (Eq. 4.9). *Absolute generalized coordinates* of the point C in Fig. 4.8 in the global C.S. (XYZ) can be calculated by combining the list of instantaneous position, \mathbf{d}_I (Eq. 4.10), and the list of reference position, \mathbf{d}_O (Eq. 4.9). As the plate's movements can be very small compared to its dimensions, a translational magnification coefficient, H_t , and a rotational magnification coefficient, H_r , are introduced to multiply the vibration signals (instantaneous position of the plate) before adding them to the reference position.

Therefore, the list of *adjusted absolute generalized coordinates*, \mathbf{d}_C , is calculated as

$$\mathbf{d}_C = [x_C \ y_C \ z_C \ \theta_C \ \phi_C \ \psi_C]^T \quad (4.15)$$

where

$$\begin{aligned} x_C &= w_O + H_t \cdot x_I, & \theta_C &= \theta_O + H_r \cdot \theta_I, \\ y_C &= l_O + H_t \cdot y_I, & \phi_C &= \phi_O + H_r \cdot \phi_I, \\ z_C &= h_O + H_t \cdot z_I, \text{ and} & \psi_C &= \psi_O + H_r \cdot \psi_I. \end{aligned}$$

A flowchart of the procedure used to calculate the list of absolute generalized coordinates is shown in Fig. 4.10.

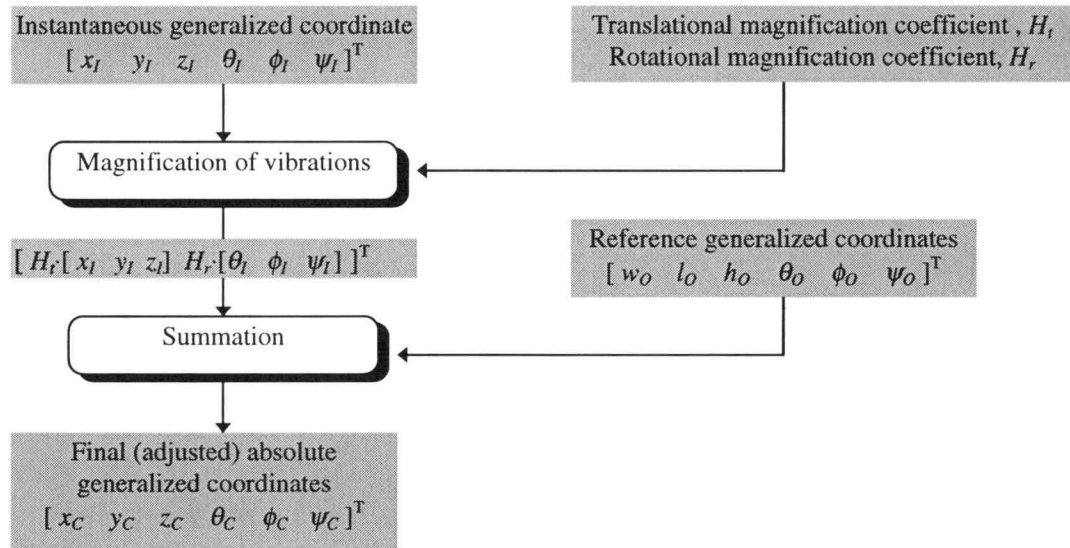


Figure 4.10 A flowchart of the procedure calculating the list of absolute generalized coordinates.

4.6.2 Calculating Coordinates of the Plate Corners

To create a 'wireframe' image of the plate for visualization, the coordinates of its corners are required. Since one of the corners designated C in Fig. 4.8 coincides with the origin of the C.S. $(XYZ)_I$, only 7 remaining corners need to be calculated. For the sake of clarity, each corner of the plate is designated by a symbol shown in Fig. 4.11 and further described in Table 4.1. The homogeneous coordinate transformation (Wolovich, 1987) is employed to calculate coordinates of these corner points. The transformation and its application are described in this section.

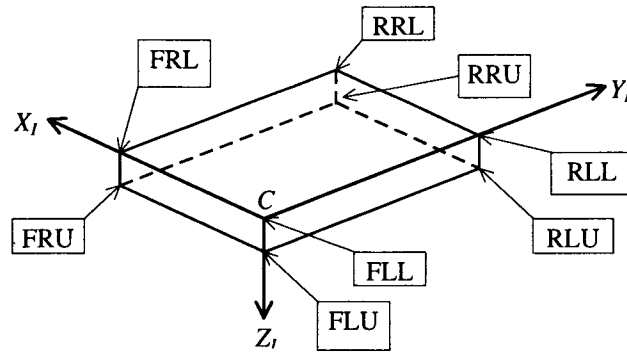


Figure 4.11 Abbreviations used to designate the plate's corners.

| Corner | Coordinates of the Corner ⁸ | Transformed Coordinates |
|-------------------------|-------------------------------------------------|------------------------------------------------------------------------------|
| Front-Right-Upper (FRU) | $\mathbf{D}_C^{\text{FRU}} = [l \ 0 \ h \ 1]^T$ | $\mathbf{D}_O^{\text{FRU}} = \mathbf{T}_0^4 \cdot \mathbf{D}_C^{\text{FRU}}$ |
| Front-Right-Lower (FRL) | $\mathbf{D}_C^{\text{FRL}} = [l \ 0 \ 0 \ 1]^T$ | $\mathbf{D}_O^{\text{FRL}} = \mathbf{T}_0^4 \cdot \mathbf{D}_C^{\text{FRL}}$ |
| Front-Left-Upper (FLU) | $\mathbf{D}_C^{\text{FLU}} = [0 \ 0 \ h \ 1]^T$ | $\mathbf{D}_O^{\text{FLU}} = \mathbf{T}_0^4 \cdot \mathbf{D}_C^{\text{FLU}}$ |
| Front-Left-Lower (FLL) | $\mathbf{D}_C^{\text{FLL}} = [0 \ 0 \ 0 \ 1]^T$ | $\mathbf{D}_O^{\text{FLL}} = \mathbf{T}_0^4 \cdot \mathbf{D}_C^{\text{FLL}}$ |
| Rear-Right-Upper (RRU) | $\mathbf{D}_C^{\text{RRU}} = [l \ w \ h \ 1]^T$ | $\mathbf{D}_O^{\text{RRU}} = \mathbf{T}_0^4 \cdot \mathbf{D}_C^{\text{RRU}}$ |
| Rear-Right-Lower (RRL) | $\mathbf{D}_C^{\text{RRL}} = [l \ w \ 0 \ 1]^T$ | $\mathbf{D}_O^{\text{RRL}} = \mathbf{T}_0^4 \cdot \mathbf{D}_C^{\text{RRL}}$ |
| Rear-Left-Upper (RLU) | $\mathbf{D}_C^{\text{RLU}} = [0 \ w \ h \ 1]^T$ | $\mathbf{D}_O^{\text{RLU}} = \mathbf{T}_0^4 \cdot \mathbf{D}_C^{\text{RLU}}$ |
| Rear-Left-Lower (RLL) | $\mathbf{D}_C^{\text{RLL}} = [0 \ w \ 0 \ 1]^T$ | $\mathbf{D}_O^{\text{RLL}} = \mathbf{T}_0^4 \cdot \mathbf{D}_C^{\text{RLL}}$ |

\mathbf{T}_0^4 denotes the homogeneous transformation matrix (Eq. 4.20).

Table 4.1 Coordinates of the corners and equations used for calculations.

Homogeneous transformation matrices (Wovolich, 1987) are utilized in coordinate transformation. Each matrix represents a relationship between arbitrary x , y , and z coordinates of a point in one coordinate system and corresponding x' , y' , and z' coordinates in another coordinate system. For a 6-DOF system, four transformation matrices are required to change one coordinate system to another. The matrices are defined as follow.

⁸ The form is compatible with the homogeneous transformation.

1. 'Yaw' transformation matrix, $\mathbf{T}_0^1(\psi)$.

This matrix accounts for a rotation around the Z_0 axis by an angle ψ shown in Fig. 4.12a. This matrix is defined as follows

$$\mathbf{T}_0^1(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 & 0 \\ \sin(\psi) & \cos(\psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.16)$$

2. 'Pitch' transformation matrix, $\mathbf{T}_1^2(\phi)$.

This matrix accounts for a rotation around the Y_1 axis by an angle ϕ shown in Fig. 4.12b. Its form is

$$\mathbf{T}_1^2(\phi) = \begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\phi) & 0 & \cos(\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.17)$$

3. 'Roll' transformation matrix, $\mathbf{T}_2^3(\theta)$.

This matrix accounts for a rotation around the X_2 axis by an angle θ shown in Fig. 4.12c. This matrix is defined as follows

$$\mathbf{T}_2^3(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.18)$$

4. Translational transformation matrix, $\mathbf{T}_3^4(x, y, z)$.

This matrix accounts for x , y , and z translations along the X_3 , Y_3 , and Z_3 axes, respectively, as shown in Fig. 4.12d.

$$\mathbf{T}_3^4(x, y, z) = \begin{bmatrix} 0 & 0 & 0 & x \\ 0 & 0 & 0 & y \\ 0 & 0 & 0 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.19)$$

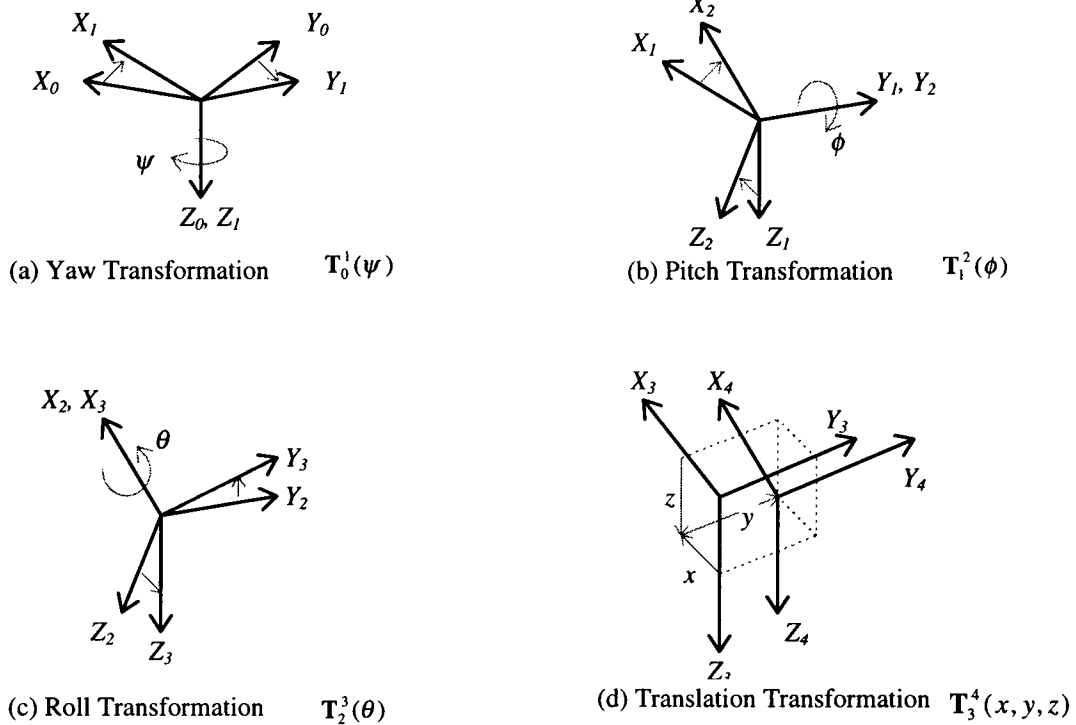


Figure 4.12 Definition of coordinate transformation matrices (Wolovich, 1987).

The definitions of operations accounted for by these matrices are graphically shown in Fig. 4.12. A succession of coordinate transformations, each represented by a particular transformation matrix, can be represented by a matrix product of individual transformation matrices (Wolovich 1987). Therefore, the overall transformation from the first coordinate system $(XYZ)_0$ to the final coordinate system $(XYZ)_4$ is obtained using the following matrix

$$\mathbf{T}_0^4(x, y, z, \theta, \phi, \psi) = \mathbf{T}_3^4(x, y, z) \cdot \mathbf{T}_2^3(\theta) \cdot \mathbf{T}_1^2(\phi) \cdot \mathbf{T}_0^1(\psi) \quad (4.20)$$

As already mentioned, a position of the point C in the C.S. $(XYZ)_R$ is obtained directly from the measured signals. This position is described by the vector

$\mathbf{V}_O^C = [x_C, y_C, z_C]^T$, that links the point C and the point O in the global C.S. (XYZ) shown in Fig. 4.13. Hence, these values in \mathbf{V}_O^C are given by the list of adjusted absolute generalized coordinates, \mathbf{d}_C , described by Eq. 4.15.

Homogeneous coordinate transformation technique is applied to find coordinates of the remaining corners. For example, to calculate the coordinates of corner A , whose coordinate vector is $\mathbf{V}_O^A = [x_A, y_A, z_A]^T$, as shown in Fig. 4.13 the following steps are performed. First, vectors \mathbf{V}_O^A and \mathbf{V}_O^C are rearranged to the column form 1×4 shown below to be compatible with the homogeneous transformation.

$$\mathbf{D}_O^C = \begin{bmatrix} \mathbf{V}_O^C \\ 1 \end{bmatrix} = \begin{bmatrix} x_C \\ y_C \\ z_C \\ 1 \end{bmatrix} \quad (4.21) \quad \text{and} \quad \mathbf{D}_O^A = \begin{bmatrix} \mathbf{V}_O^A \\ 1 \end{bmatrix} = \begin{bmatrix} x_A \\ y_A \\ z_A \\ 1 \end{bmatrix} \quad (4.22)$$

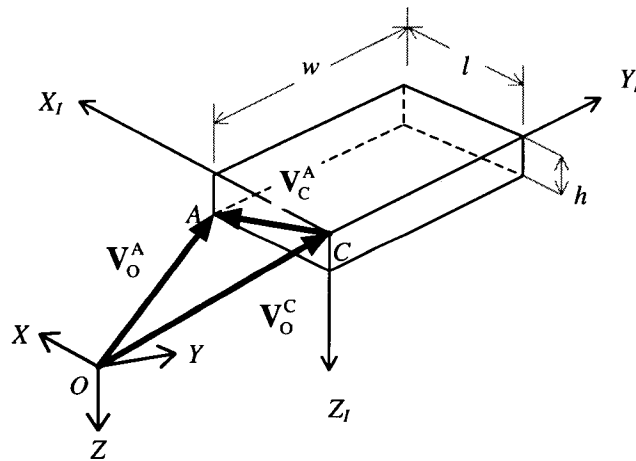


Figure 4.13 Application of the homogeneous coordinate transformation for finding coordinates of point A .

It should be noted that the coordinates of point A defined in C.S. $(XYZ)_I$ are converted to its coordinates in the global C.S. (XYZ) . A *location vector*, \mathbf{V}_C^A , is defined as a vector that contains local coordinates of the corner under consideration in the C.S. $(XYZ)_I$, *i.e.*, the coordinates with reference to the corner, C. In this example, dimensions of the plate are l , w and h , where

- l - the length of the plate measured parallel to the X_I axis,
- w - the of width the plate measured parallel to the Y_I axis, and
- h - the height of the plate measured parallel to the Z_I axis.

The location vector, \mathbf{V}_C^A , is therefore

$$\mathbf{V}_C^A = l \cdot \mathbf{i}_I + 0 \cdot \mathbf{j}_I + h \cdot \mathbf{k}_I \quad (4.23)$$

where

$\mathbf{i}_I, \mathbf{j}_I, \mathbf{k}_I$ are the unit vectors of the X_I, Y_I and Z_I axes, respectively.

The column vector, $\mathbf{D}_C^A = [l \ 0 \ h \ 1]^T$, arranged from the vector \mathbf{V}_C^A is then used in the claculation of the coordinates of the point A in the global C.S. (XYZ) as

$$\mathbf{D}_O^A = \mathbf{T}_O^4(x_C, y_C, z_C, \theta_C, \phi_C, \psi_C) \cdot \mathbf{D}_C^A \quad (4.24)$$

The same technique can be used to obtain the coordinates of the remaining corners. The location vectors of the remaining corners are listed in the second column of Table 4.1. A procedure for calculation of the coordinates of all corner is illustrated in Fig. 4.14.

4.6.3 Projection of 3-D Object to a Planer (2-D Screen)

The calculated coordinates of each corner of the plate comprises of three variables, *i.e.*, x_C, y_C, z_C , representing the plate as a 3-D object. To display this 3-D picture of the plate on a planer (2-D computer screen) a projection technique is performed

to 'project' the 3-D object on a 2-D plane. This technique is accomplished by assigning a viewpoint to be located on one of the axes of the global C.S. (XYZ) shown as an eye symbol in Fig. 4.15. The eye, *e.g.*, on the X axis, looks toward the origin, O , of the C.S. (XYZ) yielding a screen or a plane defined by the Y and Z axes, shown as a gray plane in Fig. 4.15.

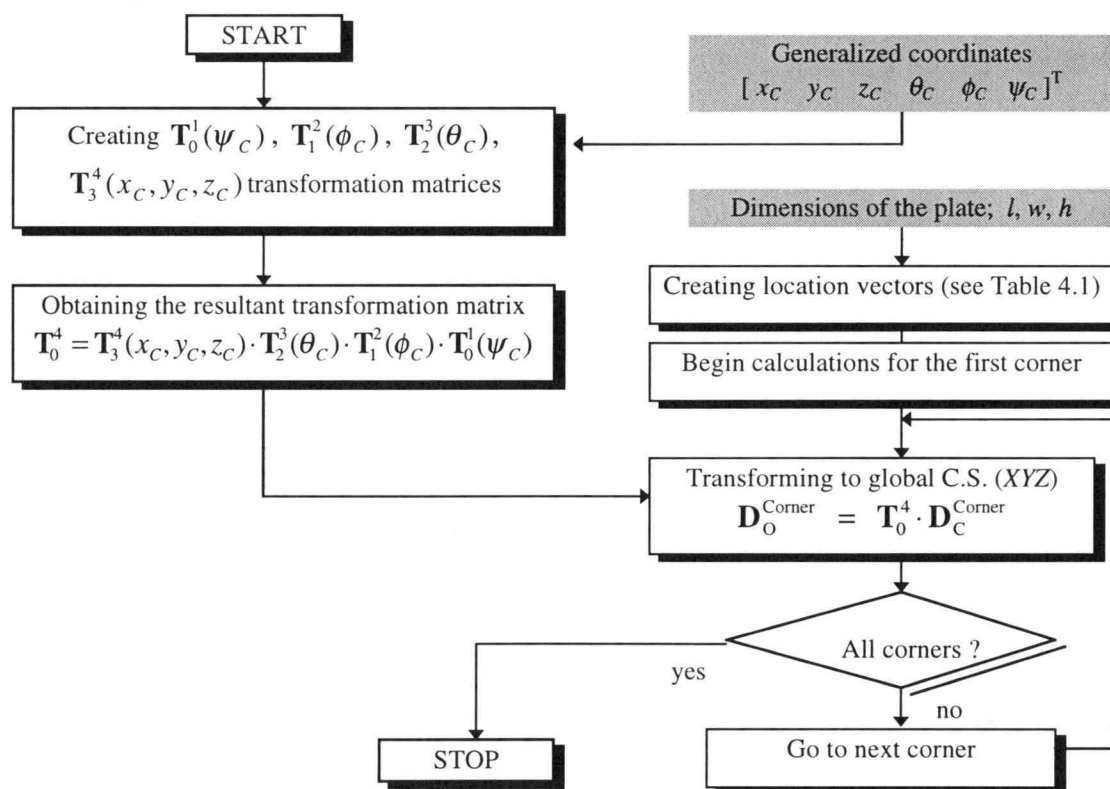


Figure 4.14 Flowchart shows procedure of calculating coordinates of all corners.

The line of sight from the eye projects the 3-D object on this 2-D screen resulting in an image of 2-D picture representing this object. This 'projection procedure' is simply accomplished in the visualization program by discarding the component of the coordinates that is on the same axis as the viewpoint, *e.g.*, the component x_c is neglected as the eye is assigned on the X axis in Fig. 4.15 (Rybaczyk, 1989).

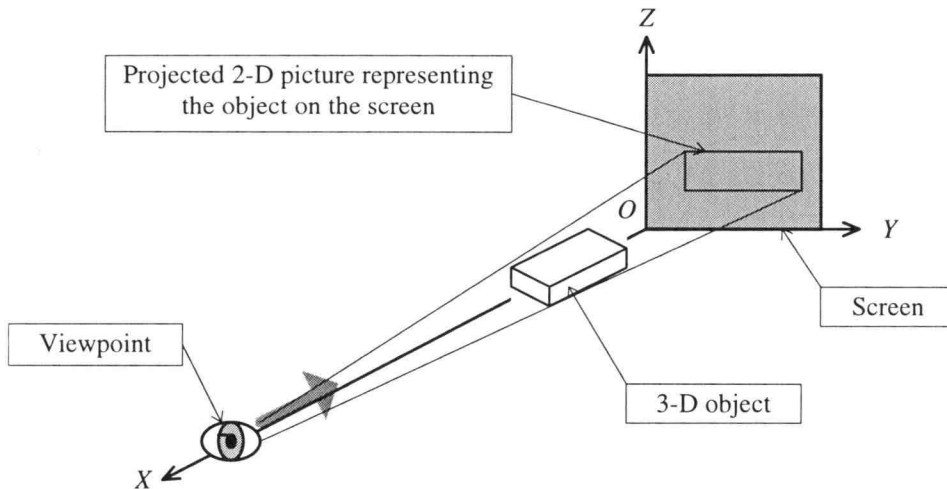


Figure 4.15 Illustration of the projection of a 3-D object on a 2-D planer.

4.6.4 Changing the Viewpoint

Vibration visualization can be enhanced by employing viewpoint changing feature. The method is accomplished using an additional coordinate transformation technique. Discussed in this section are techniques used in rotating a viewpoint around the object and creating an effect of a perspective that distorts the dimensions such that the object size is reduced with increasing distance from the viewpoint. Thus, when a viewer looks at the picture on the screen, he feels like looking at an actual object in 3-D environment (Rogers and Adams, 1990).

Mathematically, a rotation of the object in one direction relative to the eye results in the same way on the computer screen as the rotation of the eye in a negative direction relative to the object. Since the viewpoint, *i.e.*, the eye and the projection plane are fixed together with the computer screen. If a user wants to rotate the plate by certain angles, θ_v , ϕ_v , and ψ_v around the X, Y and Z axes, respectively, the calculation of new coordinates on the screen is performed by transforming the coordinate system of the eye, $(XYZ)_v$, to be rotated around the C.S. (XYZ) with the same angles shown in Fig. 4.16.

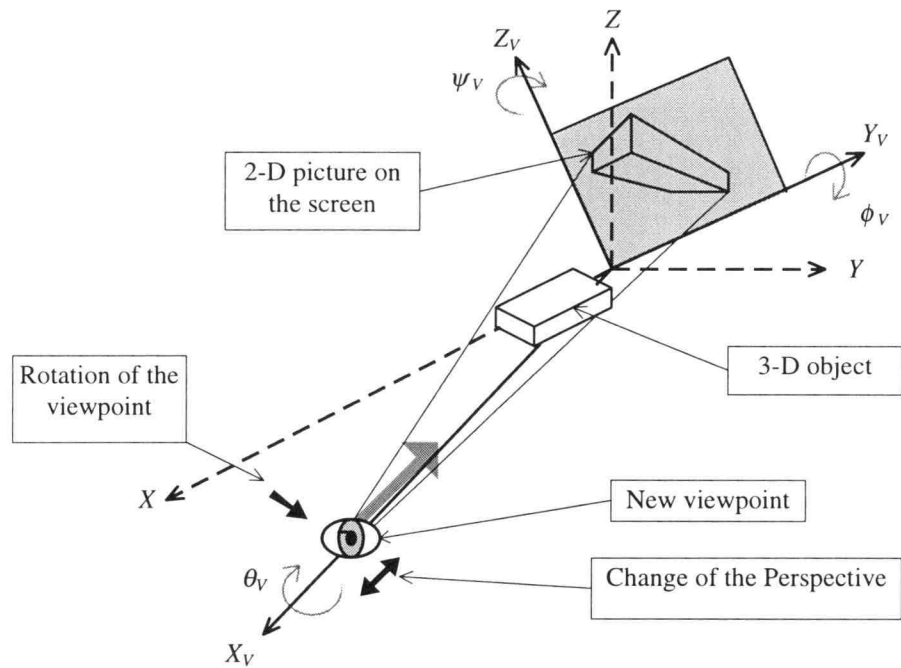


Figure 4.16 Illustration of the effect from changing the viewpoint on the 2-D picture.

An effect of a perspective on the picture is accomplished by changing mathematically the viewpoint's location on the axis whose it is located on (Rybaczyk, 1989). This procedure also governs a homogeneous coordinate transformation matrix that its first three components of the fourth column are not zero (Rogers and Adams, 1990). The calculation of new coordinates from the rotation and translation of the viewpoint, therefore, governs two transformation matrices, shown in Fig. 4.17 (Rybaczyk, 1989).

The visualization program written in this research performs the rotation of the viewpoint where the 'rotation matrix' in Fig. 4.17 is modified from the homogeneous transformation matrix, \mathbf{T}_0^4 (see Eq. 4.20), with elements as the rotation angles, θ_v , ϕ_v , and ψ_v . Therefore, new coordinates of each characteristic point of the plate in the viewpoint C.S., $(XYZ)_v$, can be calculated using the homogeneous transformation matrix modified from Eq. 4.24 as follows.

$$\mathbf{D}_V^A = \mathbf{T}_0^4(0, 0, 0, \theta_V, \phi_V, \psi_V) \cdot \mathbf{D}_O^A \quad (4.25)$$

where

\mathbf{D}_O^A - the coordinate vector of point A in C.S. (XYZ),

\mathbf{D}_V^A - the coordinate vector of the same point A in C.S. (XYZ)_V.

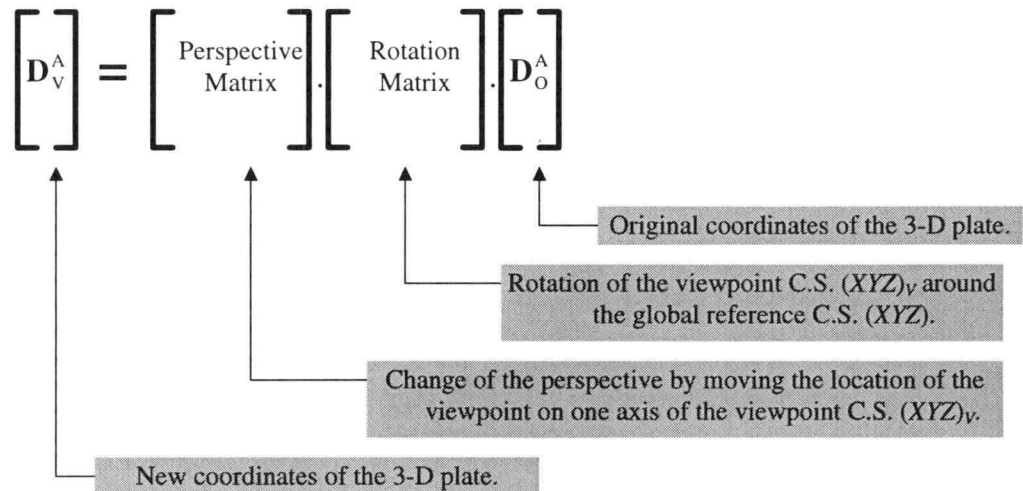


Figure 4.17 Diagram of coordinate transformation procedure for changing the viewpoint.

4.6.5 Drawing a Single 3-D Picture

To create a complete wireframe picture of the plate, its edges are plotted on *X-Y graph* display object provided by LabVIEW[®]. These edges are the lines connecting the corners defined in the rotated C.S. (XYZ)_V. To accomplish this, the set of experimental data is arranged as a 2-D array that consists of three columns (representing x_C , y_C , and z_C) with 8 rows (representing 8 corners) for each column. After projecting these corner's coordinates on the 2-D screen (see Section 4.6.3), the projected coordinates are arranged in the order such that consecutive plot of these corners with lines connecting between two successive points creating a structure of a rectangular plate without any line passing through the plate's volume. Table 4.2 shows the order of the plotted corners designed in

the visualization program that results as the wireframe diagram of the plate shown in Fig. 4.11.

| Plotting order | Corner | Plotting order | Corner | Plotting order | Corner |
|----------------|--------|----------------|--------|----------------|--------|
| 1 | FRL | 7 | RLL | 13 | RRU |
| 2 | RRL | 8 | FLL | 14 | RLU |
| 3 | RRU | 9 | FLU | 15 | FLU |
| 4 | RRL | 10 | FLL | 16 | FRU |
| 5 | RLL | 11 | FRL | | |
| 6 | RLU | 12 | FRU | | |

Table 4.2 Order of corner plotting for creating a complete 3-D rectangular plate

4.6.6 Animation of Generated 3-D Pictures

To animate vibrations of the plate, one generalized coordinate list are selected from a series of the generalized coordinates formatted in 2-D array (see Section 4.5). This list is transformed and rearranged such that a single 3-D picture is created and plotted on the *X-Y Graph* display object using procedures described in Sections 4.6.1-4.6.5. After the complete picture is displayed, next generalized coordinate list is selected from the initial 2-D array and processed to create another 3-D picture. The new picture is then plotted replacing the previous one. Consequent plotting yields an animation of the plate movement according to the actual vibration. All steps of the animation process described in this section are summarized in Fig. 4.18.

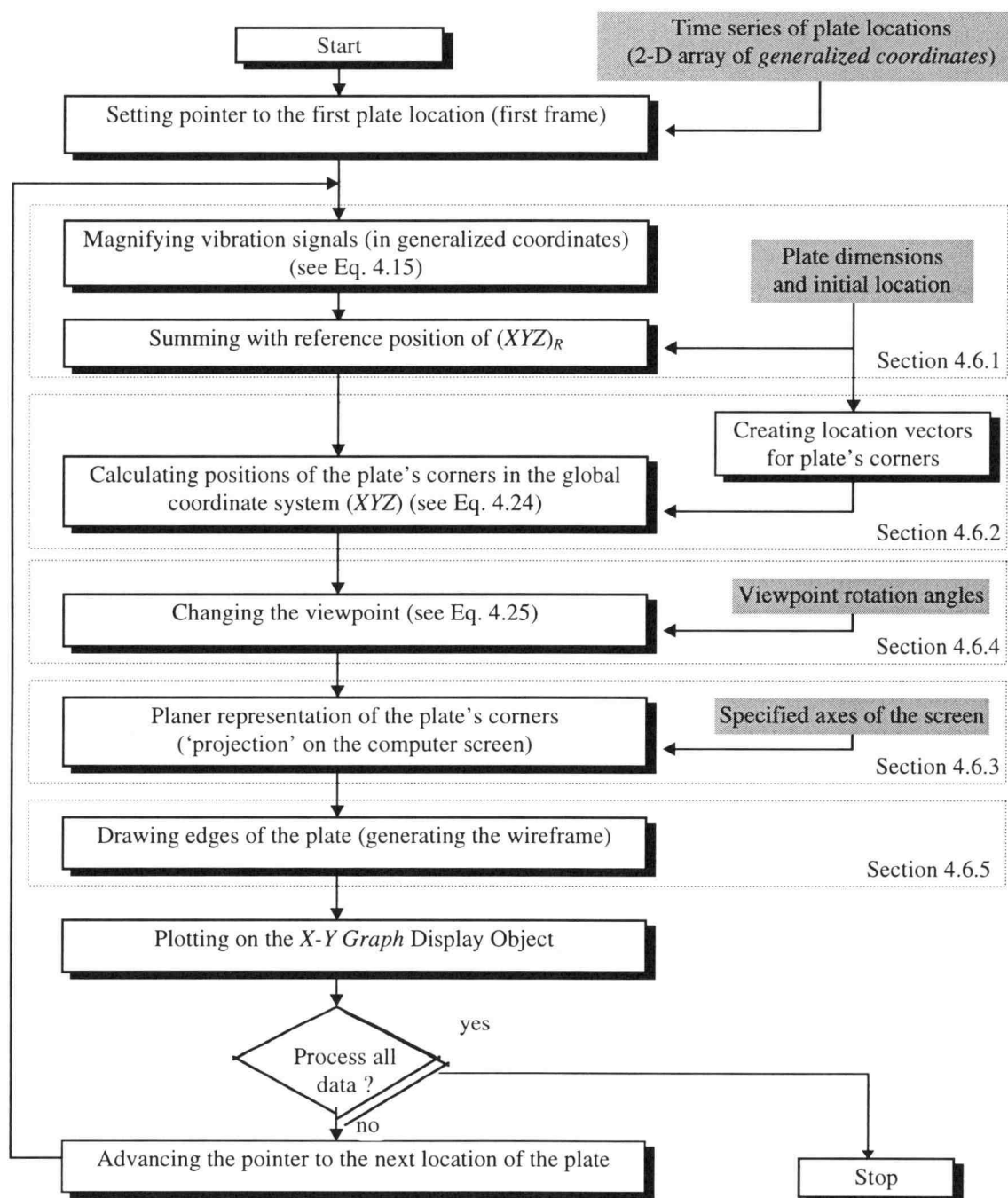


Figure 4.18 A flowchart of the entire animation procedures.

4.7 Transformation of Generalized Coordinates in a Rigid Plate

The generalized coordinates of point C , \mathbf{d}_C (Eq. 4.15), calculated from the measured signals represent location of the plate's corner. To obtain the generalized coordinates of the center of mass, G , in Fig. 4.1 the translation variables of C are transformed using the following homogeneous transformation matrix.

$$\mathbf{D}_0^G = \mathbf{T}_0^4(x_C, y_C, z_C, \theta_C, \phi_C, \psi_C) \cdot \mathbf{D}_C^G = [x_G \ y_G \ z_G \ 1]^T \quad (4.26)$$

where

\mathbf{D}_C^G - a column vector⁹ (see Eq. 4.27) defining a vector from the point C to the point G where this vector is $\mathbf{V}_C^G = l_G \cdot \mathbf{i}_l + w_G \cdot \mathbf{j}_l + h_G \cdot \mathbf{k}_l$.

$$\mathbf{D}_C^G = [l_G \ w_G \ h_G \ 1]^T \quad (4.27)$$

In a case of a rigid plate with uniform mass distribution the center of mass coincides with the plate's geometric center. Therefore \mathbf{D}_C^G can be obtained from the plate dimensions as $l_G = l/2$, $w_G = w/2$ and $h_G = h/2$.

The rotation variables of the generalized coordinate list, \mathbf{d}_G , at point G , are equal to those of \mathbf{d}_l (the plate is a rigid body). Therefore \mathbf{d}_G becomes

$$\mathbf{d}_G = [x_G \ y_G \ z_G \ \theta_C \ \phi_C \ \psi_C]^T \quad (4.28)$$

On the other hand, the generalized coordinates of point C can be calculated from the generalized coordinates of the point G as follows

$$\mathbf{D}_0^C = \mathbf{T}_0^4(x_G, y_G, z_G, \theta_C, \phi_C, \psi_C) \cdot \mathbf{D}_G^C \quad (4.29)$$

where

$$\mathbf{D}_G^C = [-w_G \ -l_G \ -h_G \ 1]^T \quad (4.30)$$

⁹ Compatible with homogeneous transformation.

4.8 Closure

To generate an animation of 3-D pictures from actual signals measured from sensors, four procedures are employed. The first procedure controls both hardware and software during data acquisition. Anti-aliasing filters are employed to prepare signals for digital acquisition. In the second step acceleration signals are processed to obtain displacements. Digital high-pass filters are applied to the signals to suppress the drift. The third stage is the calculation of the generalized coordinates of the plate. Linear equations for determining angular accelerations (Padgaonkar et al., 1975) are employed in these calculations. The final step is drawing 3-D pictures using the homogeneous coordinate transformation (Wovolich, 1987) and presenting the animation in which a series of the calculated pictures is rapidly displayed.

CHAPTER 5

EXPERIMENTAL IMPLEMENTATION AND RESULTS

The signal based visualization programs introduced in previous chapters are complicated and prone to conceptual as well as programming errors. It is therefore necessary to verify these programs experimentally before further development and application. Several experiments discussed in this chapter have been designed and conducted with the following objectives.

- (1) Verification of the visualization programs applied to a test object, namely a rigid plate suspended by four three-dimensional springs.
- (2) A comparison of signals obtained from the test system with the signals generated by a mathematical model of the system to study the feasibility of the model usage for the visualization enhancement.

5.1 Experimental Set Up

Transient excitation was applied to the experimental model of a Kistler[®] dynamometer type 9257A (Kistler, 1996) using an impact hammer. Several acceleration signals (the system's responses that represented vibrations¹⁰) were recorded and processed to obtain animated motion of the model. Instruments were set up according to the schematic diagram shown in Fig. 5.1. The *Impact Hammer* shown in Fig. 5.2 was employed as an exciter to stimulate the dynamometer. The hammer comprised of a *Piezoelectric Load Cell* (PCB[®] model 208B03; PCB, 1996) attached at the hammer's head and a *Signal Conditioner* (PCB[®] type 408D06; PCB, 1996). The signal from the load cell was also used as a trigger to initiate the data acquisition.

¹⁰ Vibration (oscillatory motion) is intuitively understood as a repeating change of position. However it can also be described in terms of the first and second time derivatives of this position, *i.e.*, velocity and acceleration, respectively.

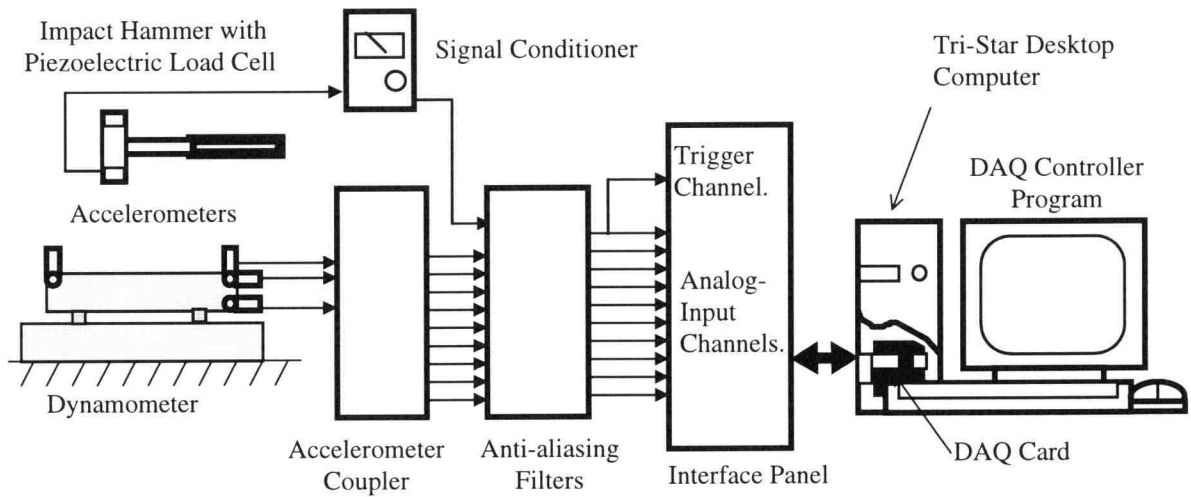


Figure 5.1 Schematic diagram of the experimental setup.

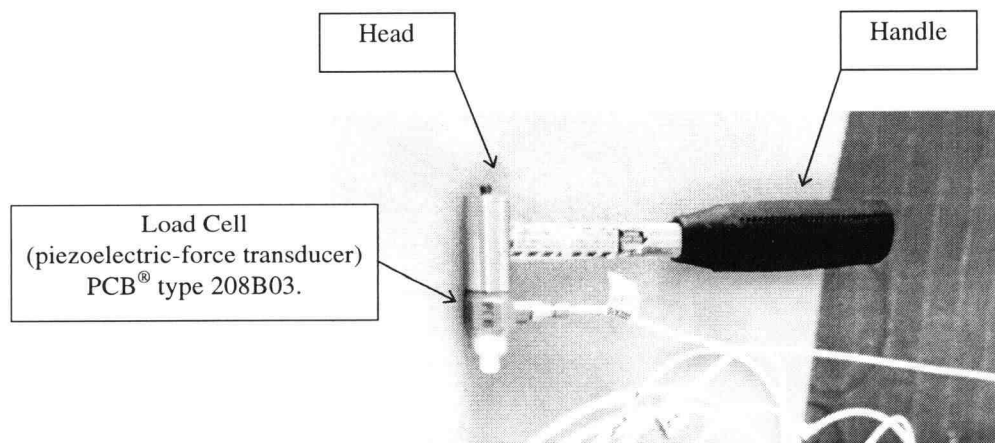


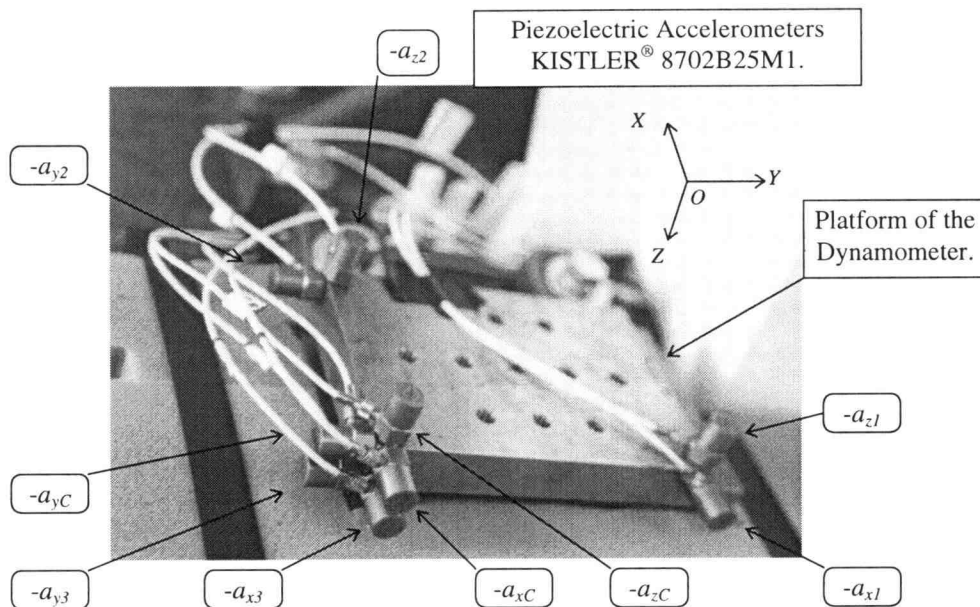
Figure 5.2 Impact hammer (PCB® type 208B03) used for exciting the dynamometer.

Nine shear-type, low impedance piezoelectric *Accelerometers* (Kistler® type 8702B25M1; Kistler, 1996) were used to measure vibrations of the dynamometer's platform. They were attached by beeswax (Doebelin, 1990; Brüel & Kjær, 1982) at the specific locations discussed in Section 4.5, as shown in Fig. 5.3. Technical specifications of the accelerometers are provided in Appendix A. Signals from the accelerometers were conditioned by an *Accelerometer Coupler* (Kistler® model 5128A; Kistler, 1996) and passed through a low-pass *Anti-aliasing Filters* (Precision® model 88-B and Datel® model FLJ-D6LA2 programmable filters; Precision, 1989; Datel, 1987). The cut-off frequency was set to 1000 Hz and the filters provided signal gains of 10.

Filtered and amplified signals were passed through an *Interface Panel* to the *DAQ Card* (Data Acquisition Card; National Instrument® AT-MIO 16E2) installed inside of a *Desktop Computer* (Tri-Star). The DAQ card used a 12 bit Analog-To-Digital Converter (ADC), a multiplexer, and additional amplifiers. The gains of these amplifiers were set by a program written in LabVIEW®. Figure 5.4 shows the entire data acquisition system.

The system allowed selection of the following parameters during visualization.

- Sampling frequency: 0 - 40,000 Hz.
- Number of recorded signals: 10 (impact force and nine accelerations). Further details are provided in Appendix A.
- Number of data: 512 - 16,384 per signal with 0 - 1,024 pre-trigger data points.
- Signal acquisition mode: triggered by a signal from the impact hammer.



a_{ij} = the acceleration at the corner j in the i direction, where $i = C, 1, 2,$ and $3, j = x, y$ and z (see Section 4.4). Minus sign indicates an inverted signal as compared with the values in equation 4.10.

Figure 5.3 Locations of nine accelerometers (Kistler® type 8702B25M1) mounted on the dynamometer. The symbols indicate accelerations assigned according to Eq. 4.10.

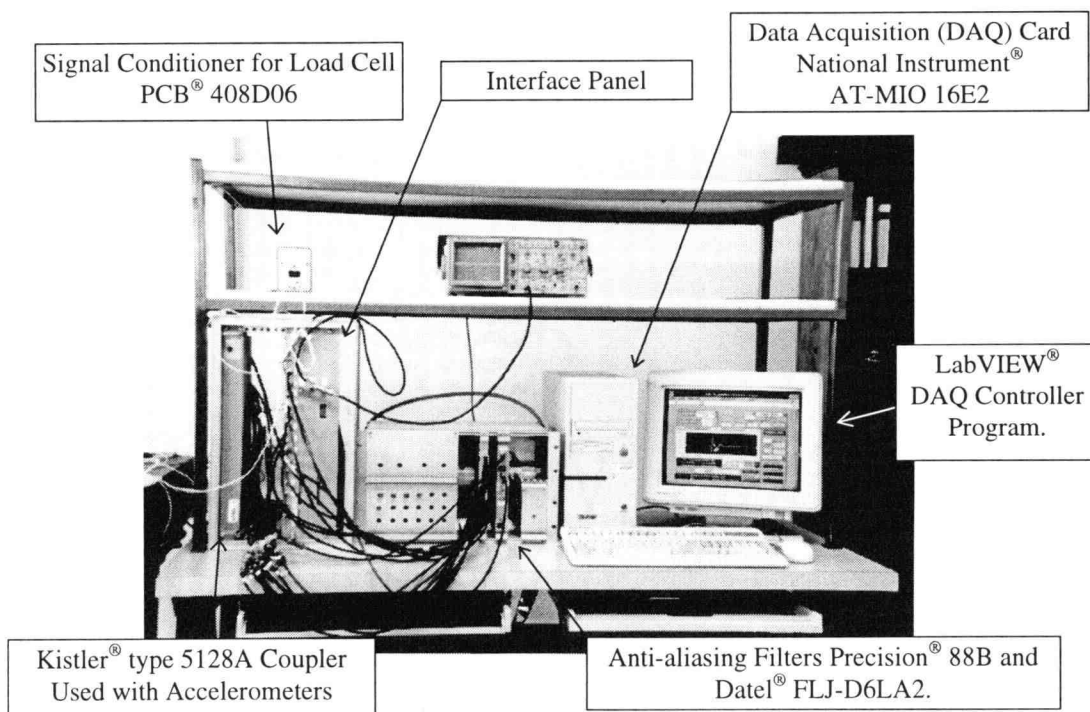


Figure 5.4 The data acquisition system.

5.2 Data Acquisition and Vibration Visualization Software

Four major program modules written in LabVIEW[®] have been developed, as stated in Section 4.1.2. A brief discussion of these modules follows.

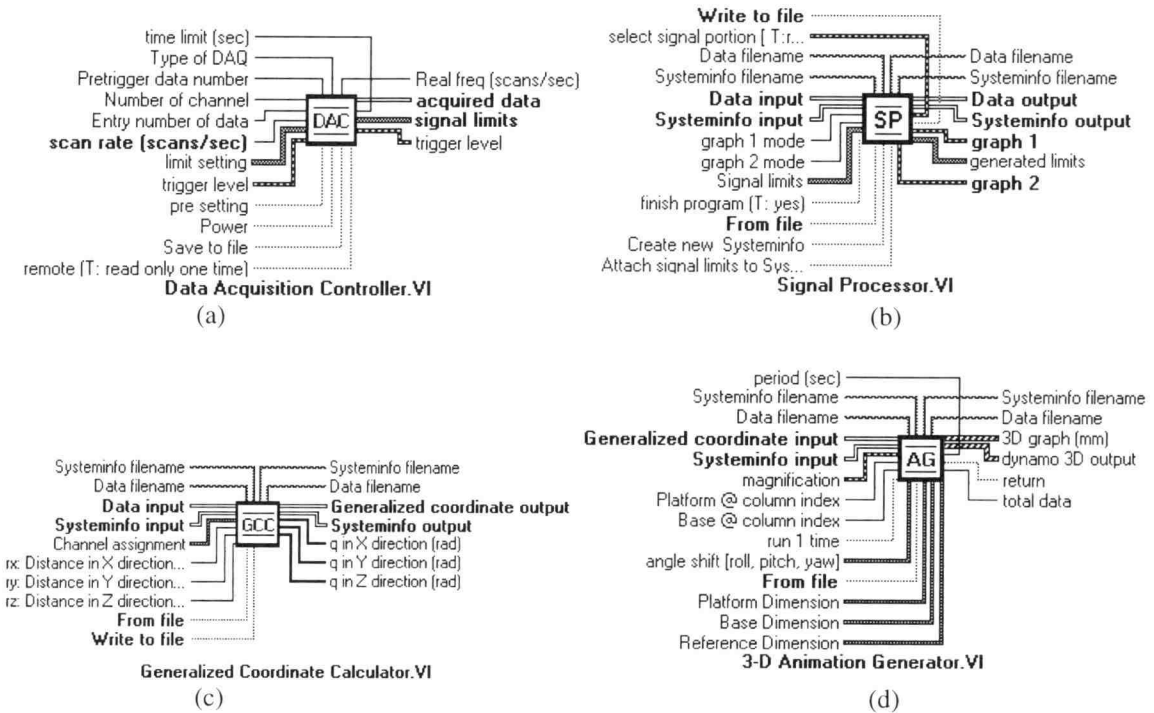


Figure 5.5 Icons and wiring terminals of the major LabVIEW[®] modules employed in the vibration visualization.

(1) “Data Acquisition Controller.VI”, henceforth denoted as **DAC**.

This module controls the DAQ card to acquire analog signals with desired parameters and also serves as a readout instrument. The parameters that can be set are listed at the end of Section 5.1. A front panel of the module is shown in Fig. 4.4a and the corresponding icon used in the ‘Block Diagram’ in LabVIEW[®] is shown in Fig. 5.5a.

(2) “Signal Processor.VI”, henceforth denoted as **SP**.

This program performs filtering and double integration of the acquired signals using the procedures described in Section 4.3.2. A front panel of the program is shown in Fig. 4.4b and its corresponding icon is shown in Fig. 5.5b. The program also converts voltage signals to the units of acceleration and force, and subtracts the average values of these signals using the calculation method described in Section 4.3.1. Digital filters are applied twice, before and after the first signal integration, to attenuate drift in the acceleration signals. Parameters of these filters are adjustable. All results shown in this thesis were obtained with the following setting.

- High-pass, 8th order elliptic filters.
- Cut-off frequency $f_c = 20$ Hz.

(3) “Generalized Coordinate Calculator.VI”, henceforth denoted as **GCC**.

The program calculates generalized coordinates of the corner point, C , using the equations described in Section 4.5. Front panel of the program is shown in Fig. 4.4c and its corresponding icon is shown in Fig. 5.5c.

(4) “3-D Animation Generator.VI”, henceforth denoted as **AG**.

The animation program creates and displays a ‘wire-frame’ picture of the moving plate in the 3-D space. Front panel of the program is shown in Fig. 4.4d and its corresponding icon is shown in Fig. 5.5d. Methodology used in the program is described in Section 4.6. Dimensions and positions of the dynamometer’s platform, base and reference plate used in the animation are listed in Appendix A.

The four software modules discussed above can be used as major building blocks in various specific programs needed for the visualization of vibrations. As shown in Fig. 5.5, each module is represented by an icon with wiring terminals. The input and output variables of these modules are represented by the terminals and signals ‘wired’ to the icons. Different appearance of wires represents different data types (according to the standard LabVIEW® designations). For example, a single line represents a simple

numeric data type while a pair of parallel lines represents a 2D-array data type. Further discussions of these modules are briefly discussed in Appendix D.

A simplified diagram of the example visualization program comprising the modules discussed above is shown in Fig. 5.6. When the program is executed, the sequence of actions can be considered as moving from left to the right. A *false* dotted line signal shown at the bottom of the diagram is 'wired' to the *from file* terminals of **SP**, **GCC** and **AG**. In response to this signal each module gets data from its 'predecessor' rather than from a disk file. Thus, a new sample of data is collected, processed and visualized per every run of the program. When a *true* signal is applied to the *from file* line, the program requests data files (in a spreadsheet compatible format) that were partially processed at some earlier time. The default value, *i.e.*, no external value applied, to the *from file* line is *true*, *i.e.*, reading data from disk files. A more detailed diagram of the program is shown in Appendix F as the "Simple **DAC**, **SP**, **GCC**, **AG**.VI".

The 'Systeminfo' file is wired to a *systeminfo* input terminal of the **SP** program. The Systeminfo file contains information about transducers, DAQ parameters, and signal processing procedures. In addition, the user can design any specific signal processing algorithm and code it (*i.e.*, write a script) for automatic execution in the Systeminfo file. The data management and communication between the modules shown in Fig. 5.6 are described in Appendix E.

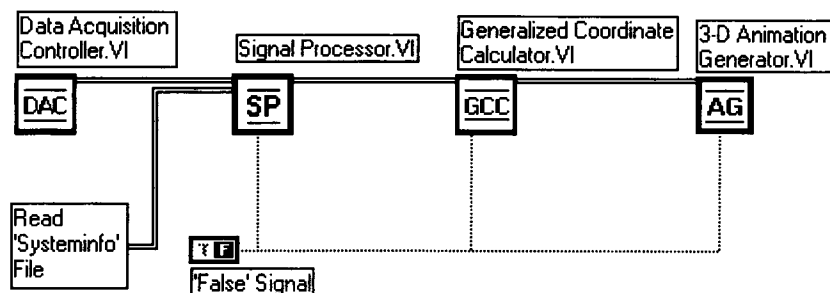


Figure 5.6 A simplified diagram of the LabVIEW[®] visualization programs using modules shown in Fig. 5.5.

5.3 Verification of the Developed Software Modules

Experiments are conducted with both simple and complex dynamic systems to verify the calculations in the programs **SP** and **GCC** as well as the graphical animation generated by the program **AG**. One representative experiment is discussed in this thesis. This experiment involved a mechanical system whose structure was similar to a high performance dynamometer (Kistler® type 9257A; Kistler, 1996). This system is henceforth referred to as the dynamometer.

5.3.1 Experimental Procedure

Several tests were conducted by exciting the dynamometer with an impact hammer. Signals from nine accelerometers and the impact force were collected and saved in 'spreadsheet compatible' files¹¹. Table 5.1 shows the locations where the impact occurred and the directions of the force (white arrows). Expected response to each impact is briefly characterized in the same table. *The Characteristic Forced Response (CFR)* of the dynamometer is the response which is dominant during the short period of time when the force from the impact hammer is applied, were presented in 3-D graphical display on the front panel of program **AG**. The orientation of the picture showed immediate response direction which was then compared with the expected response.

¹¹ ASCII, 'tab delimited' file.

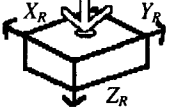
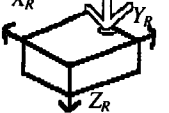
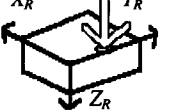
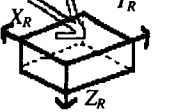
| Test No. | Location and direction of the impact | Comment |
|----------|------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 |  | <ul style="list-style-type: none"> • The impact hammer excites the dynamometer in the positive Z_R direction at a center of the dynamometer's platform. • Translation parallel to the Z_R axis should dominate the <i>CFR</i>. |
| 2 |  | <ul style="list-style-type: none"> • The impact hammer excites the dynamometer at a mid-width of the platform near the edge. • Rotation around X_R axis and translation in the Z_R axis should dominate the <i>CFR</i>. |
| 3 |  | <ul style="list-style-type: none"> • The impact hammer excites the dynamometer at mid-length of the platform, near the edge coinciding with the Y_R axis. • Rotation around Y_R axis and translation in the Z_R axis should dominate the <i>CFR</i>. |
| 4 |  | <ul style="list-style-type: none"> • The impact hammer excites the dynamometer at a mid-height of the side of the platform near to the edge. • Rotation around Z_R axis and translation in the X_R axis should dominate the <i>CFR</i>. |

Table 5.1 Descriptions of the test procedures used in the experiment.

5.3.2 Results and Discussion

The results of experiments are summarized in Table 5.2. Representative positions of the dynamometer's platform shown in the table are obtained as 'snap-shots' of the screen generated by the **AG** program module. The dash-dotted wireframe box represents a location of the platform before the impact (initial position). The wireframe drawn with the solid line represents the *Characteristic Forced Response* (CFR), *i.e.*, the location of the platform shortly after the force impact. The criteria for selecting presented pictures from the displacement response signals are illustrated in Fig. 5.7. The initial position of the platform corresponds typically to the time instance 25 ms (50 data samples) before the

impact. The CFR corresponds typically to the time when the response signal reaches the first peak after the initial contact of the impact hammer with the platform. In general, the characteristic force responses generated by the tested program agree with the expected behavior of the tested dynamic system (see Table 5.1).

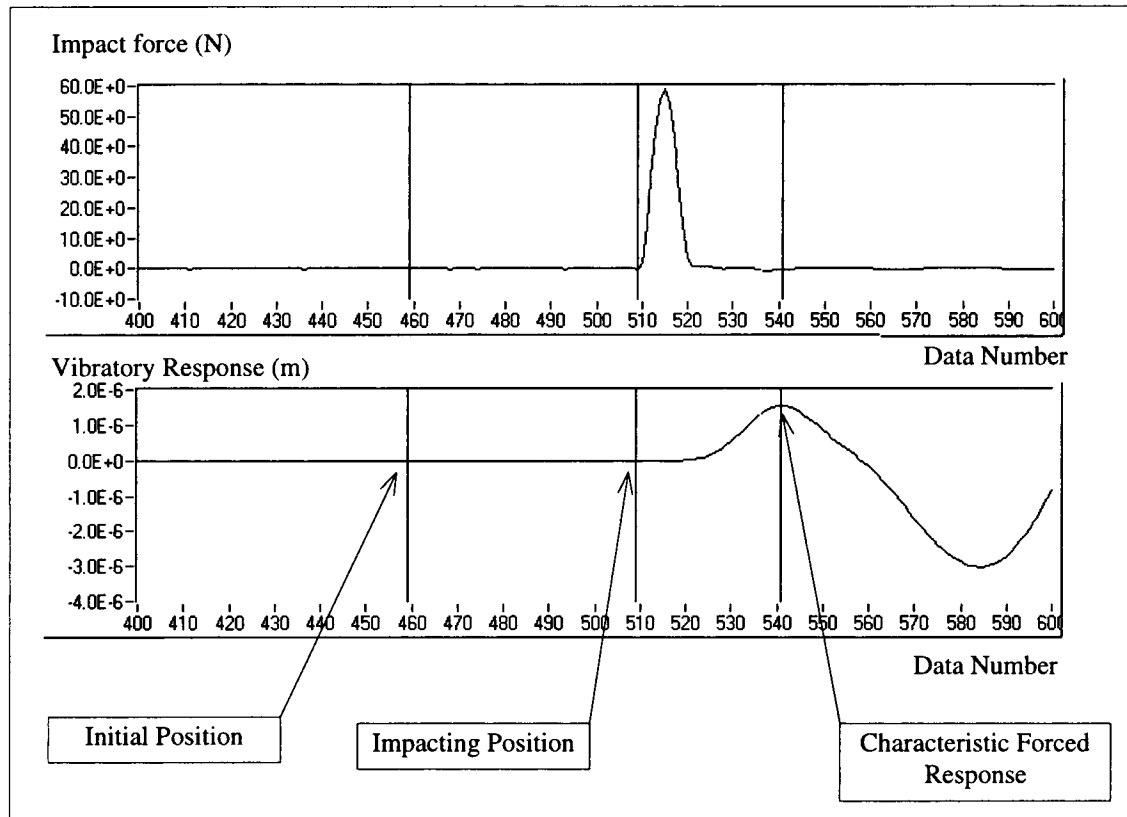


Figure 5.7 Definitions of characteristic time instances referred to in Table 5.1 and 5.2.

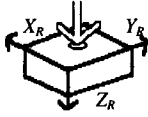
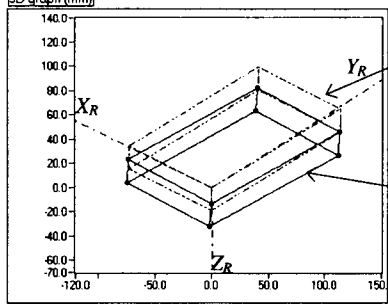
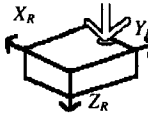
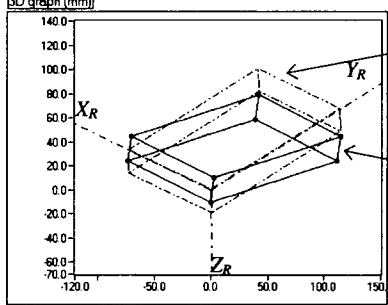
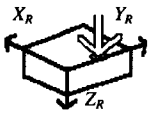
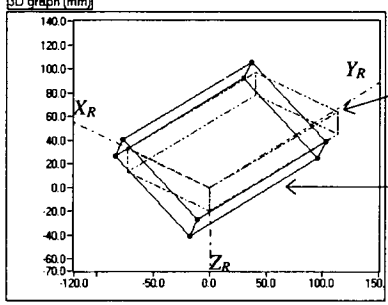
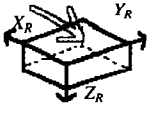
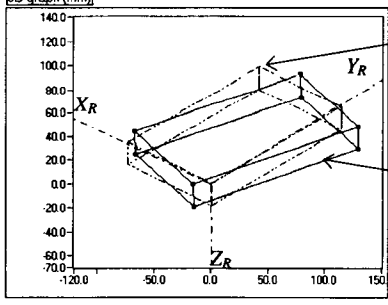
| Test number | Characteristic locations of the dynamometer's platform | |
|----------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| <p>1</p>  |  | <p>Initial Position.</p> <p>Characteristic Forced Response.</p> <p>Magnification factors $H_t = H_r = 10^7$</p> |
| <p>2</p>  |  | <p>Initial Position.</p> <p>Characteristic Forced Response.</p> <p>Magnification factors $H_t = H_r = 10^7$</p> |
| <p>3</p>  |  | <p>Initial Position.</p> <p>Characteristic Forced Response.</p> <p>Magnification factors $H_t = H_r = 10^7$</p> |
| <p>4</p>  |  | <p>Initial Position.</p> <p>Characteristic Forced Response.</p> <p>Magnification factors $H_t = H_r = 10^7$</p> |

Table 5.2 Characteristic locations of the dynamometer's platform obtained experimentally. H_t and H_r are magnification factors applied to translatory and rotational motions, respectively.

5.4 Comparison of Signal Based and Model Based Responses

5.4.1 Experimental Procedure

A linear time-invariant model of the dynamometer developed by Chung (1993) has been adopted to investigate the feasibility and potential advantages of the model based vibration visualization. A computer program written in Matlab[®] generates responses of this model to the actual force recorded during the experiment. A list of this program is provided in Appendix C.

Four representative data sets discussed in this Section are characterized in Table 5.3. Each data set contains the following variables:

1. Force signal from the impact hammer.
2. Nine signals from accelerometers mounted on the plate. These signals are converted to the measured generalized coordinates of the plate (signal based responses) according to Sections 4.4 and 4.5.
3. The predicted generalized coordinates calculated from the model (model based responses) subjected to the actual measured force impact.

Force signals from the impact hammer are read from the data files. These signals are used to stimulate the mathematical model of the dynamometer (in the previously described state space form given in Chapter 3). The computer generated responses of the model are plotted in together with the recorded and processed acceleration signals generated by the developed signal based vibration visualization software.

Although the model's responses are calculated using the actual input force data, it is necessary to provide program information regarding locations and directions of the forces acting on the platform. Therefore, in each set of experimental data, matrices \mathbf{C}_u and \mathbf{u}_m in Eq. 3.10 and Eq. 3.19 are formed differently from the others if the acting points

of the force (a_x , a_y and a_z) and the force directions are not identical. The values of a_x , a_y and a_z in the \mathbf{u}_m and \mathbf{C}_u are shown in Table 5.3. The \mathbf{F}_r is an $m \times 1$ matrix obtained from the measured force signals containing m data point.

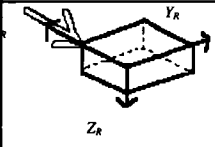
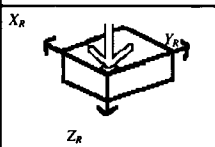
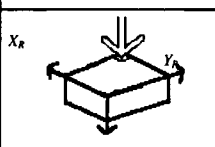
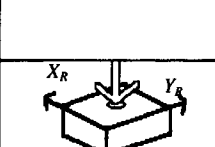
| Test No. | Locations and Directions of the Impact | Comment | Parameters of \mathbf{C}_u (mm) | \mathbf{u}_m Matrix |
|----------|-------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------|-----------------------------------------------------------------|
| 5 |  | The impact hammer hit the dynamometer in a negative X_R direction on the side plane. The hitting location was close to corner 2 in Fig. 4.9. | $a_x = 50.025$ $a_y = -75.761$ $a_z = -6.320$ | $[-\mathbf{F}_r \mathbf{0} \mathbf{0} \mathbf{0} \mathbf{0}]^T$ |
| 6 |  | The impact hammer hit the dynamometer in a Z_R direction on the top plane. The hitting location was near the origin of $(XYZ)_R$. | $a_x = -37.325$ $a_y = -68.141$ $a_z = -11.400$ | $[\mathbf{0} \mathbf{0} \mathbf{F}_r \mathbf{0} \mathbf{0}]^T$ |
| 7 |  | The impact hammer hit the dynamometer several times in a Z_R direction on the top plane. The hitting location was near the farthest corner from the origin of $(XYZ)_R$. | $a_x = -37.325$ $a_y = -68.141$ $a_z = -11.400$ | $[\mathbf{0} \mathbf{0} \mathbf{F}_r \mathbf{0} \mathbf{0}]^T$ |
| 8 |  | The impact hammer hit the dynamometer several times in a Z_R direction at a center of the top plane. | $a_x = 0$ $a_y = 4.122$ $a_z = -11.399$ | $[\mathbf{0} \mathbf{0} \mathbf{F}_r \mathbf{0} \mathbf{0}]^T$ |

Table 5.3 Description of procedures in the experiment.

5.4.2 Results and Discussion

The comparison of graphs representing *signal based* and *model based* responses are shown in Fig. 5.8 to Fig. 5.11. The solid lines represent the model based responses calculated using the mathematical model of the dynamometer (Chung, 1993). The dashed

lines represent signal based responses calculated using the signal based vibration visualization technique.

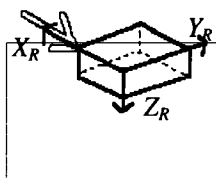
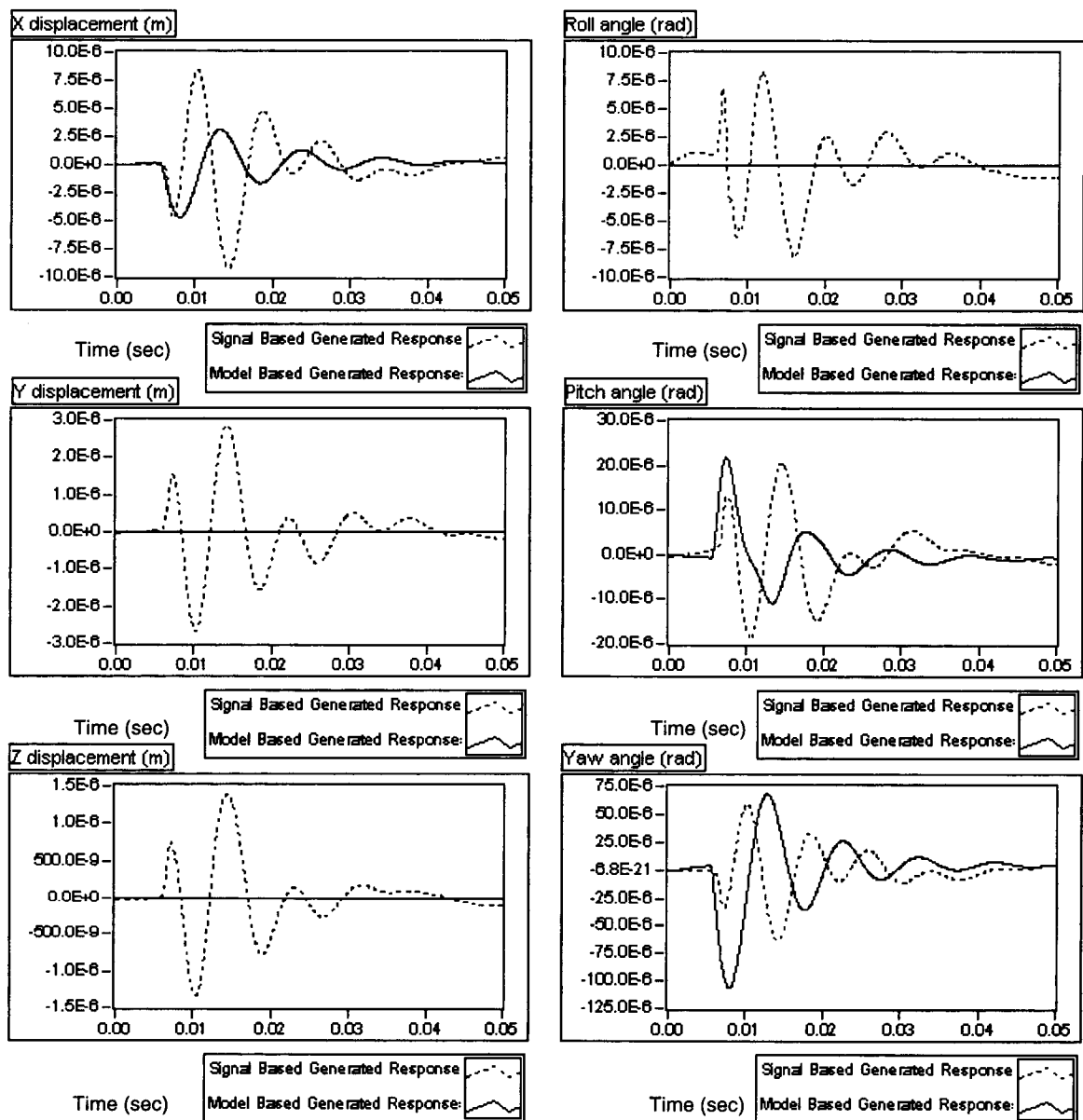


Figure 5.8 Graphs show results from the test number 5.

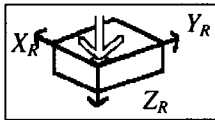
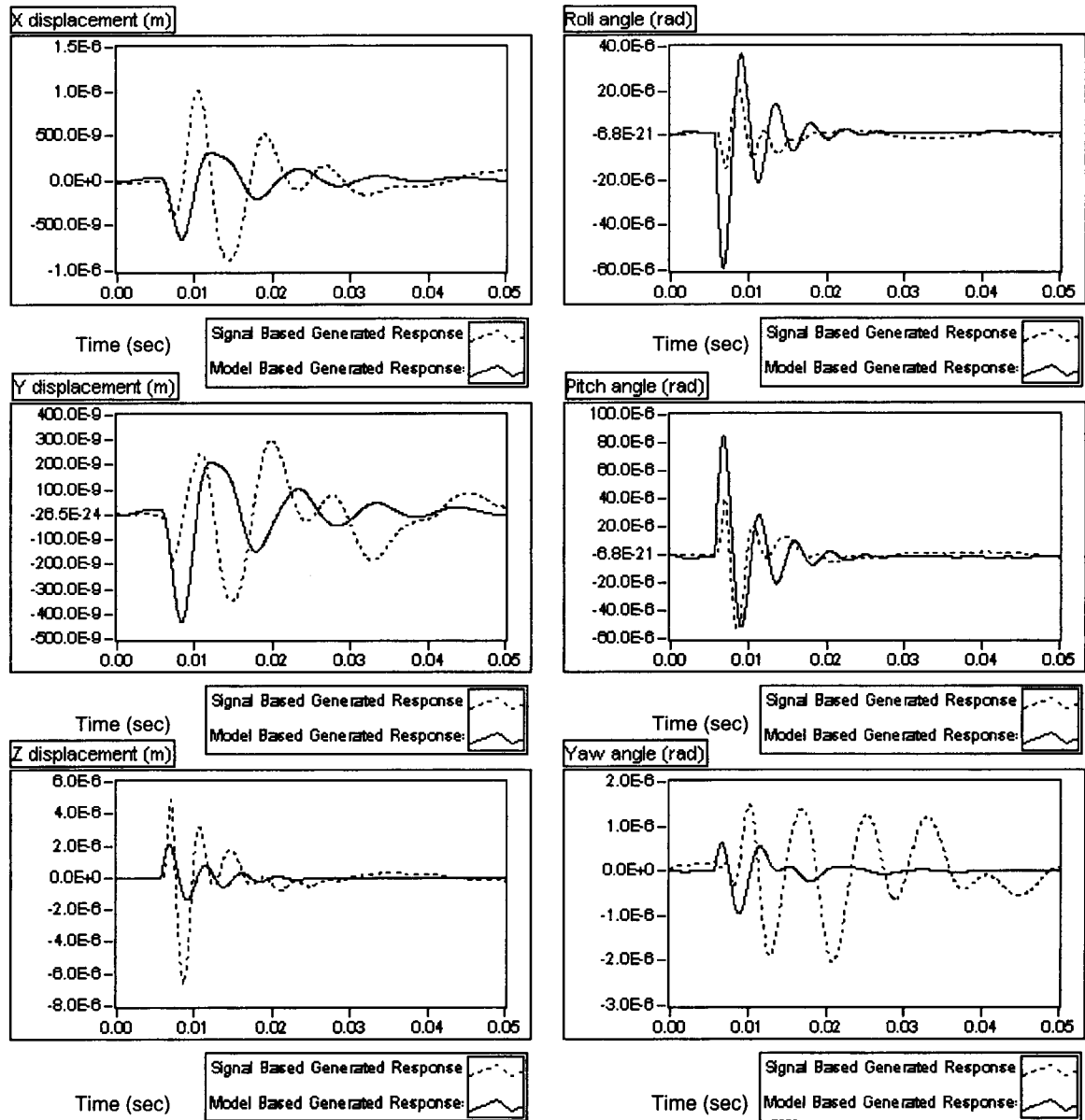


Figure 5.9 Graphs show results from the test number 6.

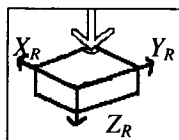
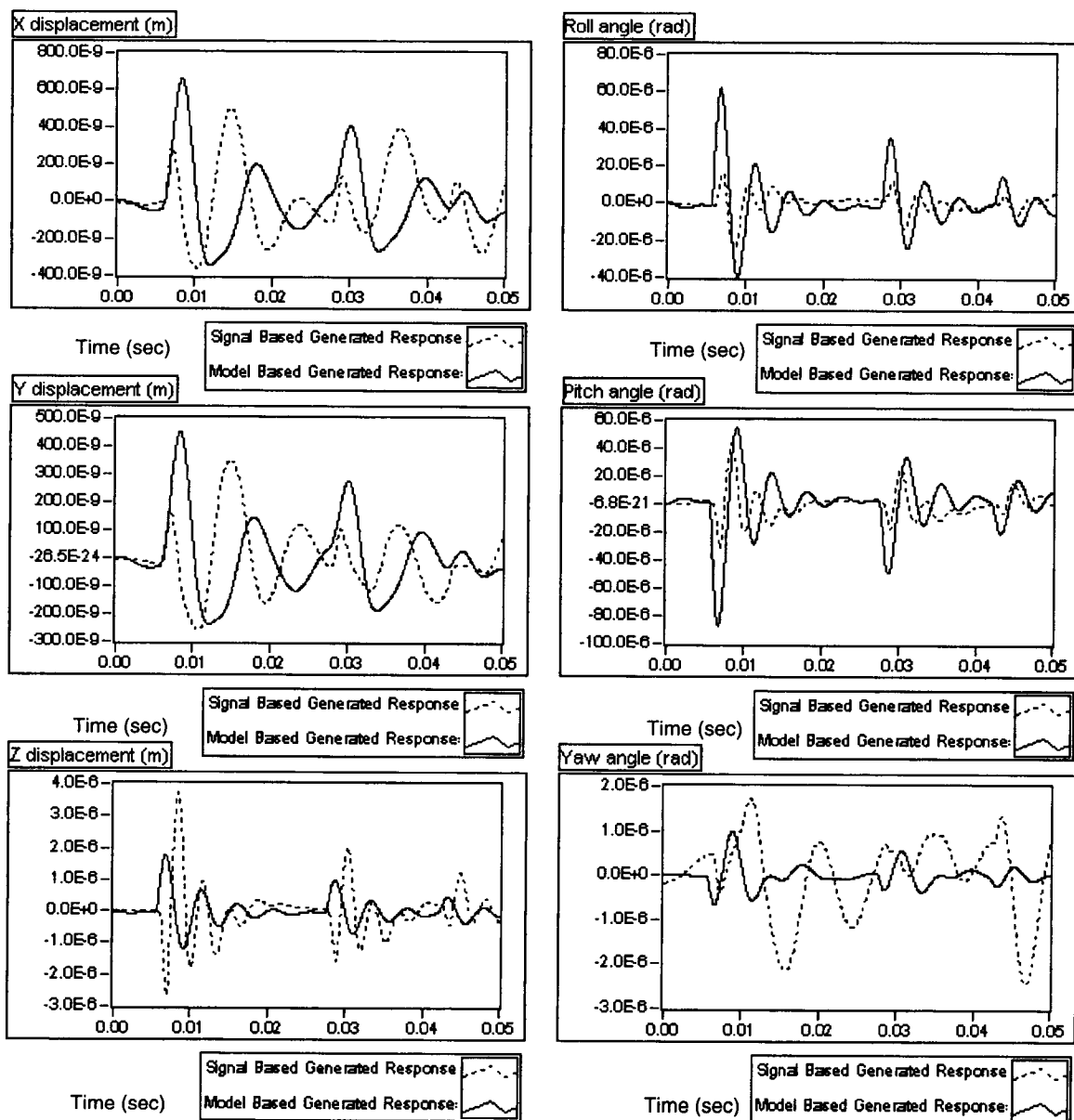


Figure 5.10 Graphs show results from the test number 7.

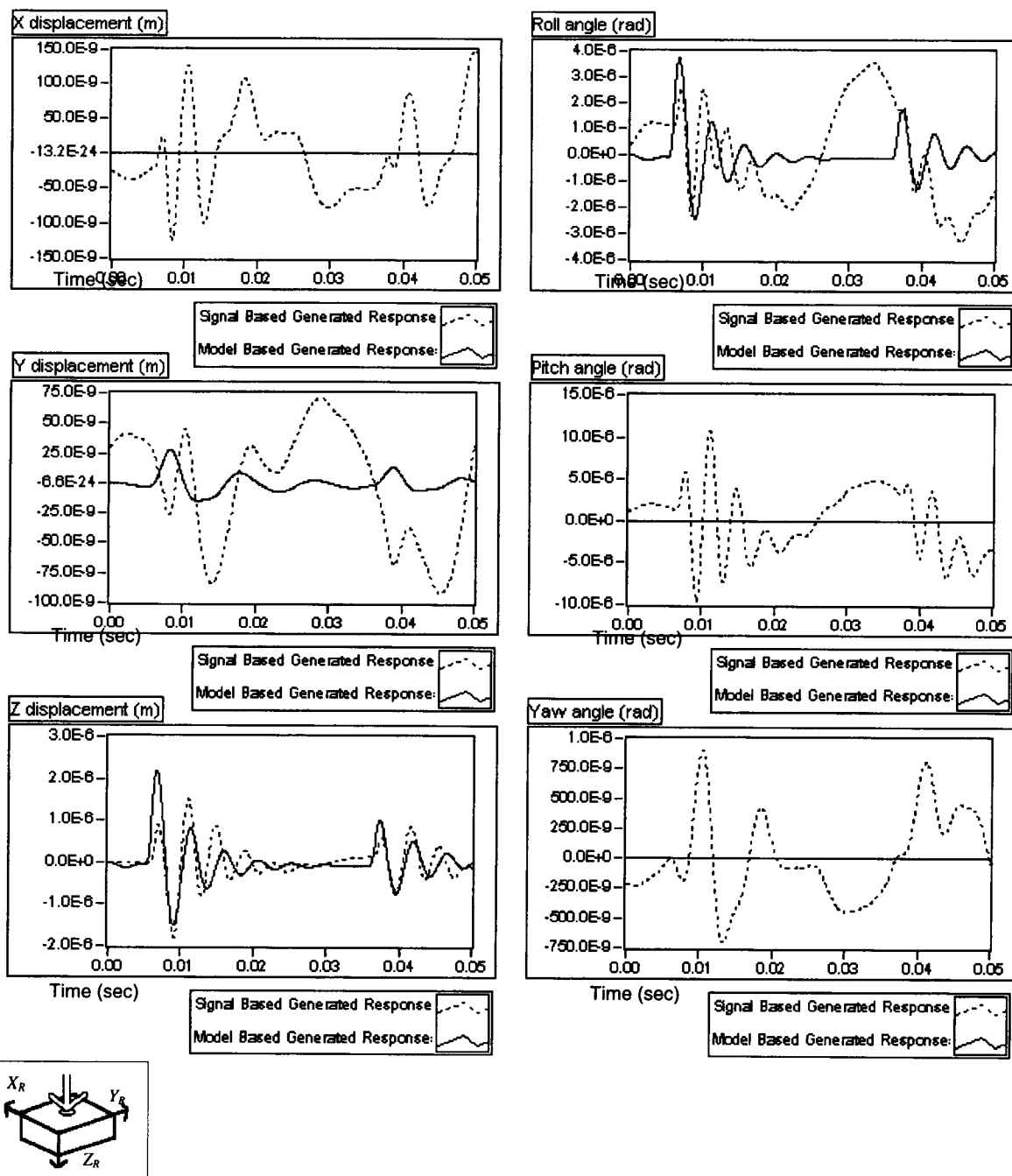


Figure 5.11 Graphs show results from the test number 8.

From the test no. 5 and 6, The response signals from both methods represented movement of the platform in the same direction. Except in the Y, Z and Row directions of

the test no. 5 (see Fig. 5.8), no signal is generated from the model (solid lines). Since, the model of the dynamometer is calculated under an assumption of perfect geometric structure, *e.g.*, locations of the sensing elements are symmetry and the mass distribution is uniform. Thus, the center of mass of the platform is almost the same as the center of the configuration. Therefore, when the input force is assigned to hit the platform in the negative X direction (by arranging $\mathbf{C}\mathbf{u}$ and \mathbf{u}_m matrices as shown in Table 5.3), the calculation of the model based response then provide a pure rotation around Z axis without any translation in the Y and Z directions.

By comparing the characteristic of the signals in the test no. 5 and 6, obviously the signal based responses (dashed lines) contained higher frequency component than the model based responses (solid lines) from the analysis of periods of the signals. This behavior leads to an assumption that the stiffness coefficients of the actual system are higher than estimated stiffness coefficients of the model. In addition, the decay of the model based response is faster than the signal based response, thus, the damping coefficients of the model are also assumed to be higher than those of the actual system. Therefore, the experimental result showed incompatibility of the parameters in the governed model.

In the test no. 7 and 8, the signal based responses (dashed lines) showed fluctuation around zero level even at the portion of signals before the impact, especially in X and Y direction of the test no. 8 (see Fig. 5.11) and in *Yaw* of the test no. 7 (see Fig. 5.10). This oscillation of the based line of the signal based response is suspected to be caused by the digital filters used in the signal processing procedure (see Section 4.3.2). The fluctuation errors in the displacement signals also have an influence on the rotations (from the calculation described in Section 4.5). The results are also the fluctuation of rotation signals as shown in the *Roll*, *Pitch* and *Yaw* signals in the test no. 8. However, the same type of error does not have much influence on such strong signals found in the test no. 5 and 6. The recorded signals from the sensors are stronger in these tests because the impact locations are closer to the accelerometer's mounting locations than those in the

test no. 7 and 8. Therefore, the signal processing technique in the visualization program (the program **SP**) still have problems when this technique is applied to weak signals.

As analyzing the result from the test no. 8 in which the impact hammer hit the platform at the plate's center. Even though the major movement occurs in the Z direction as expected, the signal based responses also show unexpected rotations of the platform in the *Pitch* and *Yaw* directions (see Fig. 5.11). The signal based response in the *Roll* direction also contains an additional rotation besides the oscillatory similar to those of the model based response. These errors are assumed to be caused by the flexible vibrations of the plate. A diagram of the dynamometer shown in Fig. 5.12 illustrates a possible mode shape of the response when a flexible mode is taken into account.

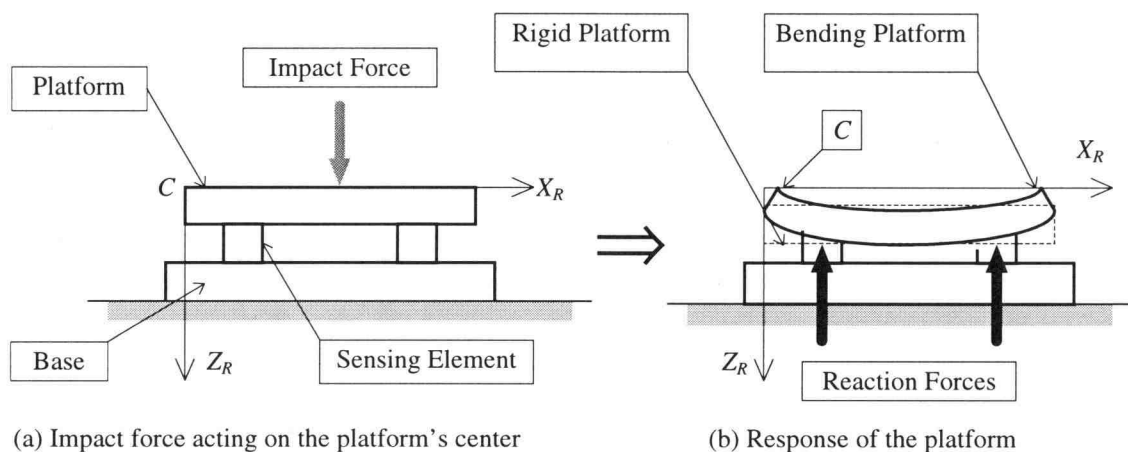


Figure 5.12 Illustration of the flexible mode of platform's vibration from the test number 5.

When the impact hammer hits the platform (gray arrow in Fig. 5.12a), the platform should move purely downward (shown as the dashed plate in Fig. 5.12b) since it is assumed to be rigid. However, the movement of the platform against the sensing elements yielding reaction forces acting on the platform shown as solid arrows in Fig. 5.12b. These reaction forces cause the platform to bend. Thus, besides the movement in

the Z direction, the corner C also moves toward the platform's center in the X and Y directions. Since the visualization program assumes the component to be a rigid plate, the vibrations in the X and Y directions from the actual bending platform result as the plate's rotations shown in the *Roll*, *Pitch* and *Yaw* directions in Fig. 5.8.

However, these errors can be indicated by observing the predicted responses from the model. From the test no. 8, the model based responses (solid lines) show the indication of no significant rotation in the response according to a calculation from the location and direction of the impact force. Hence, the discrepancy between the signal based (dashed lines) and the model based responses (solid lines) in the X and Y directions in the test no. 8 indicates the possible bending of the platform. Thus, the model based responses can be used to detect the error in the signal based visualization.

5.5 Closure

Programs designed for visualizing vibrations in mechanical systems have been tested in the experiment. The program provided sufficient results in presentation of vibration. The results from a comparison between the signal based responses and the model based responses showed similarity in movement directions, however, some characteristics of the responses (frequencies and decay rates) were different due to the primary estimation of the parameters. The discrepancy between the signal based and the model based responses also occurred due to distortion of signals, especially in data that were calculated from weak signals. Furthermore, the governed model was developed under the assumptions of perfect geometric structure and mass distribution which is not what the real system is. Further development is required to improve the model to be close to the actual system.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

The algorithm of visualization of vibrations in mechanical systems using sensors with model based enhancement technique is proposed and partially developed in this thesis. The first stage designated the signal based vibration visualization has been implemented as computer programs written in LabVIEW[®]. Within these programs, signals from sensors attached to the mechanical systems are acquired, processed and presented in 3-D graphical animation representing the actual vibrations.

As a contribution part, the second stage designated the model based vibration visualization is emphasized on the improvement of the proposed visualization programs. The underlined concepts involved in this stage are outlined in the thesis that can be summarized as the use of mathematical models of the mechanical systems to detect and correct errors in the signal based vibration visualization scheme.

The developed visualization programs were verified in the experiments governing an experimental model of a dynamometer. As a result, the animated pictures generated from the programs represented actual vibrations of the dynamometer in logical manners. In addition, the vibrations obtained experimentally from the programs responded accordingly to the vibrations generated analytically from a dynamometer's model, thus, confirmed reliability of the programs. However, some errors were presented in the display of excessive vibrations that can be a result of the sensor's characteristic, the signal processing procedure used in the visualization programs or noises in the data acquisition system. Even though this misrepresentation is difficult to be distinguished from the

actual vibrations, the errors are able to be detected using a comparison with the model based generated vibrations as studied in the experiments. These errors may also be reduced by applying the second stage to the visualization procedure.

6.2 Recommendations

The model based vibration visualization technique may be accomplished based upon the following options of developments.

1. Application of the system identification technique will be implemented by developing programs that perform the identification of system's parameters on real time data.
2. A methodology of the comparison and correction unit (see Fig. 3.6) will be investigated. A study for the development will be emphasized on the signal analysis, for example, random data analysis, digital signal processing, and fuzzy knowledge.
3. The graphical display of the animation program (AG) should be enhanced. The perspective effect as well as the presentation of the plate as a solid block should be applied to the 3-D display. In addition to the LabVIEW[®], a number of software packages with excellent graphical display, *e.g.*, Mathematica[®] and Matlab[®], can be used for this purpose.
4. The model of the system can be improved by considering flexible mode of vibrations. Since the model employed in the first stage is designed under the assumption of a rigid body, the generated responses using this model may not be precisely close to the actual vibration.
5. In similar to the model development, the flexible mode of vibrations should be considered in the visualization program. The animation program can be modified

such that it presents a plate comprising of many elements. The movement of each element is calculated from individual data. The signal processing and the generalized coordinate calculation procedures can also be extended to be able to calculate more data without significant time delay. More sensors can be employed and a modal analysis technique can also be applied to detect the flexible mode of vibrations.

6. Low frequency accelerometers and the multidirection accelerometers may be used to improve the visualization. Inclometers can also be employed where the angular movement is directly measured; thus, a number of sensors is reduced.
7. Suitable software should be selected as the final application program for implementing the visualization with consideration of efficient time, memory management and cost are considered. The software programs under the consideration include OS/2[®] GPF based on C program and Visual Basic[®].

BIBLIOGRAPHY

- Bendat, J. S. and Piersol, A. G. Random Data, Analysis and Measurement Procedures. New York: John Wiley & Sons, Inc., 1985.
- Brüel & Kjær. Measuring Vibration. Available from Brüel & Kjær Instruments, Inc., 1982.
- Chen, Ben. "Computer-aided Derivation of State Space Models for Linear Dynamic Systems." Diss. Oregon State U, 1996.
- Chung, Y. L. "Adaptive Compensation of Dynamic Characteristics of In-process Sensors." Diss. U of Wisconsin-Madison, 1991.
- . "Model Based Adaptive Compensation of Dynamic Characteristics of In-process Sensors." Diss. U of Wisconsin-Madison, 1993.
- Collins, R., et al. "Methods and Applications of System Identification in Shock and Vibration." System Identification of Vibration Structures, Mathematical Models from Test Data. Ed. Plikey, W. D. and Cohen, R. New York: ASME., 1972.
- Dally, J. W., Riley, W. F. and McConnell, K. G. Instrumentation for Engineering Measurement. 2nd ed. New York: John Wiley & Sons, Inc., 1993.
- Datel. FLJ-D5, D6 Digital-programmable High-order Low-pass Filter Product Data Sheet. Available from Datel, 1987.
- Doebelin, E. O. Measurement Systems: Application and Design. 4th ed. New York: McGraw-Hill, Inc., 1990.
- Endevco. Endevco® Piezoelectric and Variable Capacitance Accelerometers. Available from Endevco Corp., 1996.
- Ewins, D. J. Modal Testing: Theory and Practice. New York: Research Studies Press Ltd., 1984.

- Franklin, G. F., Powell, J. D. and Abbas, E. N. Feedback Control of Dynamic Systems. 3rd ed. New York: Addison-Wesley Publishing Company, Inc., 1994.
- Hibbeler, R. C. Engineering mechanics-Statics and Dynamics. 7th ed. New Jersey: Prentice-Hall, Inc., 1995.
- IC Sensors. Understanding Accelerometer Technology. Available from IC Sensors, Inc., 1991.
- Kalpakjian, S. Manufacturing Engineering and Technology. 2nd ed. New York: Addison-Wesley Publishing Company, Inc., 1992.
- Kial, R. and Mahr, W. Piezoelectric measuring Instruments and their Applications. Trans. Kial, R. and Mahr, W. Kistler Instrument Corp., 1984.
- Kistler. Advanced Instrumentation for a World of Applications. Available from Kistler Instrument Corp., 1995.
- Kistler. K-BEAM® Accelerometer. Available from Kistler Instrument Corp., 1996.
- Lang, G. F. "PC Based Modal analysis Comes of Age." Sound and Vibration Jan. 1990: pp. 20-30.
- McConnell, K. G. Vibration Testing; Theory and Practice. New York: John-Wiley & Sons, Inc., 1995.
- National Instruments. AT-MIO E Series User Manual. Available from National Instruments Corp., 1995.
- National Instruments. LabVIEW® for Windows User Manual. Available from National Instruments Corp., 1994
- Natke, H. G. "Identification of Vibrating Structures: an Introduction." Identification of Vibrating Structures. Italy: Springer-Verlag Wien., 1982.
- Padgaonkar, A. J., Krieger, K. W. and King, A. I., "Measurement of Angular Acceleration of a Rigid Body Using Linear Accelerometers." Journal of Applied Mechanics Transactions of the ASME, Vol. 42, pp. 552-556, 1975.

- PCB. Piezoelectric Accelerometers. Available from PCB Piezotronics., 1996.
- Peters, J. and Mergeay, M. "Dynamic Analysis of Machine Tools Using Complex Modal Method." Annals of the CIRP, Vol. 25/1, 1976, pp. 257-261.
- Potter, D. "Keys to Success in Data Acquisition System Design." In Tech. Vol. 39. Aug. 1992: pp. 24 -28.
- Powell, C. P. "Machinery Troubleshooting Using Vibration Analysis Techniques." Sound and vibration. Jan. 1992: pp. 42-54.
- Precision. 88 Series 8 Pole, 8 Zero Programmable Filter Modules. Available from Precision Filters, Inc., 1989.
- Rehsteiner, F., et al. "Modeling and Assessment of Machine Tool Dynamics and Accuracy." (To be published).
- Rockstad, H.K., et al. "A Miniature High-Resolution Accelerometer Utilizing Electron Tunneling." Mocromachinical Systems, ASME. DSC. Vol. 40., 1992: pp. 41.
- Rogers, D. F. and Adams, J. A. Mathematical Elements for Computer Graphics. 2nd ed. New York: McGraw-Hill Publishing Company., 1990.
- Rybarczyk, P. R. An Integrated System for the Computer Aided Monitoring and Diagnosis of Machine Tools. Diss. U of Wisconsin-Madison., 1989.
- Spectral Dynamics. STAR System™. Available from Spectral Dynamics, Inc., 1995.
- Spectral Dynamics Research Corporation. I-DEAS Model™. Available from Structural Dynamics Research Corp., 1996.
- Spiewak, S. A. "Modeling of Cutting Point Trajectories in Milling.", ASME Journal of Engineering for Industry. Vol. 116. No. 4. pp. 440-448, 1994.
- Thomson, W. T. Theory of Vibration with Applications. New Jersey: Prentice-Hall, Inc., 1993.

Unbehauen, H. "Some Modern Developments in System Identification Using Parameter Estimation Methods." Identification of Vibrating Structures. NY: Springer-Verlag Wien, 1982.

Vibrant Technology. ME'scope™ Demo Guide. Available from Vibrant Technology, Inc., 1996.

Vibration Engineering Consultants. PCMODAL-Modal Analysis Software for the PC. Available from Vibration Engineering Consultants, Inc., 1997.

Wolovich, W. A. Robotics: Basic Analysis and Design. New York: CBS Collage Publishing, 1987.

APPENDICES

Appendix A Experiment Specifications

A.1 Information of Sensors Used in the Experiment

| Comment | Location (Fig. 5.3) | Serial no. | Sensitivity | | Channel no. | Amplifier Gain (G_f) |
|----------------------------|------------------------|------------|----------------|--------------------------|----------------|-----------------------------|
| | | | (s_y) mV/g | mVs ² /metres | | |
| Accelerometer ¹ | a_{xC} | C100868 | 199 | 20.292 | 2 | 10 |
| Accelerometer ¹ | a_{yC} | C100859 | 205 | 20.903 | 3 | 10 |
| Accelerometer ¹ | a_{zC} | C100867 | 210 | 21.413 | 4 | 10 |
| Accelerometer ¹ | a_{xI} | C100852 | 197 | 20.088 | 5 | 10 |
| Accelerometer ¹ | a_{zI} | C101813 | 201 | 20.496 | 6 | 10 |
| Accelerometer ¹ | a_{y2} | C100858 | 198 | 20.190 | 7 | 10 |
| Accelerometer ¹ | a_{z2} | C101804 | 203 | 20.699 | 8 | 10 |
| Accelerometer ¹ | a_{x3} | C101812 | 200 | 20.394 | 9 | 10 |
| Accelerometer ¹ | a_{y3} | C100862 | 196 | 19.986 | 10 | 10 |
| Load cell ² | Force | 1519 | 2.248 mV/N | - | 0 | 100 |

Table A.1 Descriptions of sensors used in the experiment.

A.2 Power Spectrum of Signal from Accelerometer

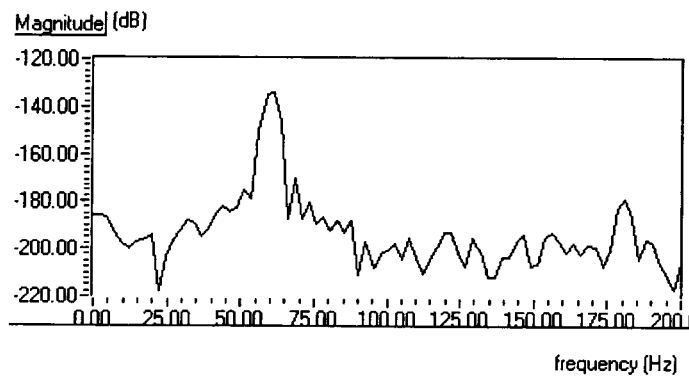


Figure A.1 Power spectrum density of signal measured from an accelerometer that is attached to the dynamometer when no vibration applied.

¹ Piezoelectric accelerometer (shear type) Kistler® model 8702B25M1 (Kistler, 1996).

² Load cell (piezoelectric-force transducer) PCB® model 208B03 (PCB, 1996).

A.3 Dimensions of the Dynamometer

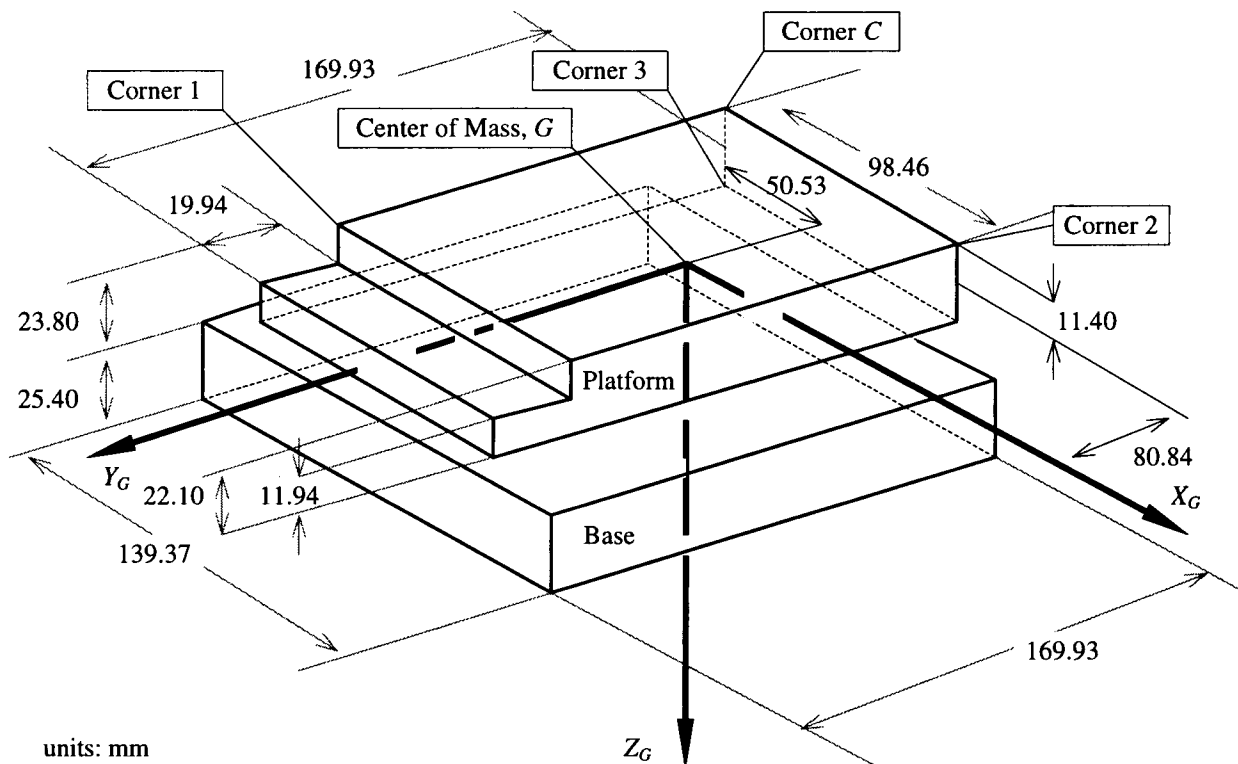


Figure A.2 Dimensions of the dynamometer used in the experiment in units of mm (the sensing elements are not shown).

A.4 Digital High-pass Filter's Coefficients

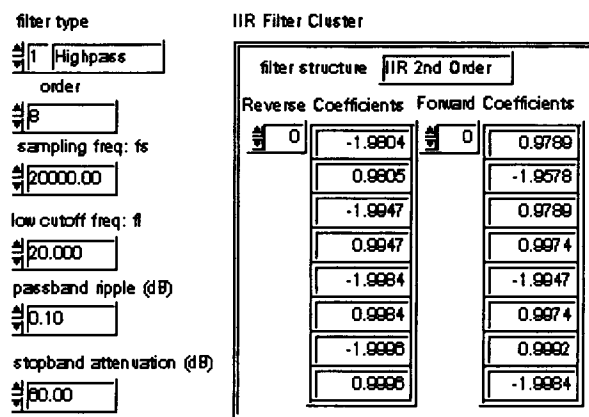


Figure A.3 Digital high-pass filter's coefficients used in Eq. 4.4 and Eq. 4.6.

Appendix B Parameters of the Dynamometer's Model

B.1 Spatial Matrices of the Dynamometer

These matrices are employed in the spatial model of the dynamometer (Eq. 3.10 and Eq. 3.11) (Chen, 1996).

$$\mathbf{m} = \begin{bmatrix} [m & 0 & 0 & 0 & 0 & 0], & [0 & m & 0 & 0 & 0 & 0], & [0 & 0 & m & 0 & 0 & 0], \\ [0 & 0 & 0 & I_{xx} & -I_{xy} & -I_{xz}], & [0 & 0 & 0 & -I_{xy} & I_{yy} & -I_{yz}], & [0 & 0 & 0 & -I_{xz} & -I_{yz} & I_{zz}] \end{bmatrix}.$$

$$\mathbf{c} = \begin{bmatrix} [cx_4 + cx_1 + cx_2 + cx_3, 0, 0, 0, (cx_4 + cx_1 + cx_2 + cx_3) h, (cx_3 - cx_1 + cx_4 - cx_2) b], \\ [0, cy_4 + cy_1 + cy_2 + cy_3, 0, (-cy_4 - cy_3 - cy_2 - cy_1) h, 0, (cy_4 + cy_2 - cy_1 - cy_3) a], \\ [0, 0, cz_4 + cz_1 + cz_2 + cz_3, (-cz_4 - cz_3 + cz_2 + cz_1) b, (-cz_4 - cz_2 + cz_1 + cz_3) a, 0], \\ [0, (-cy_4 - cy_3 - cy_2 - cy_1) h, (-cz_4 - cz_3 + cz_2 - cz_1) b, \\ (cz_4 + cz_1 + cz_2 + cz_3) b^2 + (cy_4 + cy_1 + cy_2 + cy_3) h^2, (-cz_2 - cz_3 + cz_4 + cz_1) b \cdot a, \\ (cy_3 + cy_1 - cy_2 - cy_4) h \cdot a], \\ [(cx_4 + cx_1 + cx_2 + cx_3) h, 0, (-cz_4 - cz_2 + cz_1 + cz_3) a, (-cz_2 - cz_3 + cz_4 + cz_1) b \cdot a, \\ (cz_4 + cz_1 + cz_2 + cz_3) a^2 + (cx_4 + cx_1 + cx_2 + cx_3) h^2, (cx_3 - cx_1 + cx_4 - cx_2) h \cdot b], \\ [(cx_3 - cx_1 + cx_4 - cx_2) b, (cy_4 + cy_2 - cy_1 - cy_3) a, 0, (cy_3 + cy_1 - cy_2 - cy_4) h \cdot a, \\ (cx_3 - cx_1 + cx_4 - cx_2) h \cdot b, (cy_4 + cy_1 + cy_2 + cy_3) a^2 + (cx_4 + cx_1 + cx_2 + cx_3) b^2] \end{bmatrix}.$$

$$\mathbf{k} = \begin{bmatrix} [[kx_1 + kx_2 + kx_3 + kx_4, 0, 0, 0, (kx_1 + kx_2 + kx_3 + kx_4) h, (kx_4 + kx_3 - kx_2 - kx_1) b], \\ [0, ky_1 + ky_2 + ky_3 + ky_4, 0, (-ky_1 - ky_2 - ky_3 - ky_4) h, 0, (-ky_1 + ky_2 - ky_3 + ky_4) a], \\ [0, 0, kz_1 + kz_2 + kz_3 + kz_4, (kz_1 + kz_2 - kz_3 - kz_4) b, (kz_1 - kz_2 + kz_3 - kz_4) a, 0], \\ [0, (-ky_1 - ky_2 - ky_3 - ky_4) h, (kz_1 + kz_2 - kz_3 - kz_4) b, \\ (kz_1 + kz_2 + kz_3 + kz_4) b^2 + (ky_1 + ky_2 + ky_3 + ky_4) h^2, (kz_4 - kz_2 + kz_1 - kz_3) b \cdot a, \\ (-ky_4 + ky_3 + ky_1 - ky_2) h \cdot a], \\ [(kx_1 + kx_2 + kx_3 + kx_4) h, 0, (kz_1 - kz_2 + kz_3 - kz_4) a, (kz_4 - kz_2 + kz_1 - kz_3) b \cdot a, \\ (kz_1 + kz_2 + kz_3 + kz_4) a^2 + (kx_1 + kx_2 + kx_3 + kx_4) h^2, (kx_4 + kx_3 - kx_2 - kx_1) h \cdot b], \\ [(kx_4 + kx_3 - kx_2 - kx_1) b, (-ky_1 + ky_2 - ky_3 + ky_4) a, 0, (-ky_4 + ky_3 + ky_1 - ky_2) h \cdot a, \\ (kx_4 + kx_3 - kx_2 - kx_1) h \cdot b, (ky_1 + ky_2 + ky_3 + ky_4) a^2 + (kx_1 + kx_2 + kx_3 + kx_4) b^2] \end{bmatrix}.$$

$$\mathbf{C}_u = \begin{bmatrix} [[1, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0], [0, 0, 1, 0, 0, 0], \\ [0, -a_z, a_y, 1, 0, 0], [a_z, 0, -a_x, 0, 1, 0], [-a_y, a_x, 0, 0, 0, 1]] \end{bmatrix}.$$

B.2 Constant Parameters in the Spatial Matrices

Mass of the platform (Chen, 1996).

$$m = 2.714 \text{ kg.}$$

Mass moment of inertia of the platform around the center of mass, G (Chen, 1996).

$$I_{xx} = .006187217392 \text{ kg}\cdot\text{m}^2, I_{yy} = .002398407627 \text{ kg}\cdot\text{m}^2, I_{zz} = .008366544210 \text{ kg}\cdot\text{m}^2,$$

$$I_{xy} = 0 \text{ kg}\cdot\text{m}^2, I_{yz} = .00007500541682 \text{ kg}\cdot\text{m}^2, I_{xz} = 0 \text{ kg}\cdot\text{m}^2.$$

Stiffness of the sensing elements (Chen, 1996).

$$k_{01} = 262000 \text{ N/m}, k_{02} = 1260000 \text{ N/m.}$$

$$k_{01} = k_{x1} = k_{x2} = k_{x3} = k_{x4} = k_{y1} = k_{y2} = k_{y3} = k_{y4}.$$

$$k_{02} = k_{z1} = k_{z2} = k_{z3} = k_{z4}.$$

Damping coefficients of the sensing elements (Chen, 1996).

$$c_{0x} = 140 \text{ N}\cdot\text{s/m}, c_{0y} = 125 \text{ N}\cdot\text{s/m}, c_{0z} = 275 \text{ N}\cdot\text{s/m.}$$

$$c_{0x} = c_{x1} = c_{x2} = c_{x3} = c_{x4}.$$

$$c_{0y} = c_{y1} = c_{y2} = c_{y3} = c_{y4}.$$

$$c_{0z} = c_{z1} = c_{z2} = c_{z3} = c_{z4}.$$

Distances of the sensing elements from the center of mass (Chen, 1996).

$$a = 0.02999 \text{ m (in } X_G \text{ direction),}$$

$$b = 0.04979 \text{ m (in } Y_G \text{ direction),}$$

$$h = 0.01070 \text{ m (in } Z_G \text{ direction).}$$

Appendix C

MATLAB® Program Used in the Experiment

This program is applied to the experiment number 10 described in Section 5.3.1.

```

clear; format short e;
%=====
%                               Read Data from File
%=====

%-----Signal Information and DAQ Parameters-----

InputColm=1;           % Assign column number of force signal;
Xcolm=1;               % Assign column number of Response signal;
Ycolm=2;
Zcolm=3;
THETAc olm=4;
PHIcolm=5;
PSIcolm=6;

smp lFreq = 20000;     % sampling frequency, Hz

%-----Loading Force Data -----

load NEARZA.VIZ       % enter specific name of your force data file
force1=NEARZA;
u1 = force1(:,InputColm) ; % select the first column as input 'u'

dataSize = size(u1);

force1=0;

%-----Input Filename (appearing on graph)-----
name='NEARZA.COD'

%-----Loading Response Data -----

load NEARZA.COD       % enter specific name of your Response data file
response1=NEARZA;
y1X = response1(:,Xcolm) ; % select the second column as 'y'
y1Y = response1(:,Ycolm) ;
y1Z = response1(:,Zcolm) ;
y1TH = response1(:,THETAc olm) ;
y1PH = response1(:,PHIcolm) ;
y1PS = response1(:,PSIcolm) ;
response1=0;

```

```

PDSZ = 1000;          % Plot Data Size

% text((dataSize(1)-150)*deltaT,0,name)

%===== CONSTANT PARAMETERS =====

%----- Plate Dimension -----
Lall=3.939*.0254; Wall=6.69*.0254; Hall=0.87*.0254;

%----- Center of Gravity Location Vector ----
ICG=1.9695*.0254; wCG=3.1827*.0254; hCG=0.4488*.0254;

%----- Plate Moment of Inertia -----
cont1=(.0254^5)*7.841*10^3; m=2.714;
Ixx=74.6372*cont1; Iyy=28.9323*cont1; Izz=100.9267*cont1;
%Ixy=0; Iyz=.9048*cont1; Ixz=0;
Ixy=-0.05*cont1; Iyz=.9048*cont1; Ixz=-0.9*cont1;

%----- Location of Impacting Point (From Center of Mass) ----
ax=-37.325;
ay=-68.141;
az=-11.40;

%----- Inputs are "kx1,ky1,...,kz4"-----
k01=262*10^3; k02=(1260/1)*10^3;
kx1=k01; kx2=k01; kx3=k01; kx4=k01;
ky1=k01; ky2=k01; ky3=k01; ky4=k01;
kz1=k02; kz2=k02; kz3=k02; kz4=k02;

%----- Location of Sensing Elements-----
a=2.999*10^(-2); b=4.979*10^(-2); h=1.07*10^(-2);

%----- Damping Coefficients -----
cox=140; coy=125; coz=275;
cx1=cox; cx2=cox; cx3=cox; cx4=cox;
cy1=coy; cy2=coy; cy3=coy; cy4=coy;
cz1=coz; cz2=coz; cz3=coz; cz4=coz;

%----- SPATIAL MATRICES -----

M=[m,0, 0, 0, 0, 0;0,m,0,0,0,0;0,0,m,0,0,0;
  0,0,0,Ixx,-Ixy,-Ixz;0,0,0,-Ixy,Iyy,-Iyz;0,0,0,-Ixz,-Iyz,Izz];

%----- C matrix -----
C=[cx4+cx1+cx2+cx3,0, 0, 0, (cx4+cx1+cx2+cx3)*h, (cx3-cx1+cx4-cx2)*b;
  0,cy4+cy1+cy2+cy3,0,(-cy4-cy3-cy2-cy1)*h,0,(cy4+cy2-cy1-cy3)*a;
  0,0,cz4+cz1+cz2+cz3,(-cz4-cz3+cz2+cz1)*b,(-cz4-cz2+cz1+cz3)*a,0;

```

```

0,(-cy4-cy3-cy2-cy1)*h,(-cz4-
cz3+cz2+cz1)*b,((cz4+cz1+cz2+cz3)*b^2)+((cy4+cy1+cy2+cy3)*h^2),(-cz2-
cz3+cz4+cz1)*b*a,(cy3+cy1-cy2-cy4)*h*a;
(cx4+cx1+cx2+cx3)*h,0,(-cz4-cz2+cz1+cz3)*a,(-cz2-
cz3+cz4+cz1)*b*a,((cz4+cz1+cz2+cz3)*a^2)+((cx4+cx1+cx2+cx3)*h^2),(cx3-cx1+cx4-
cx2)*h*b;
(cx3-cx1+cx4-cx2)*b,(cy4+cy2-cy1-cy3)*a,0,(cy3+cy1-cy2-cy4)*h*a,(cx3-cx1+cx4-
cx2)*h*b,((cy4+cy1+cy2+cy3)*a^2)+((cx4+cx1+cx2+cx3)*b^2)];

%----- K matrix -----
K=[kx1+kx2+kx3+kx4,0,0,0,(kx1+kx2+kx3+kx4)*h,(kx4+kx3-kx2-kx1)*b;
0,ky1+ky2+ky3+ky4,0,(-ky1-ky2-ky3-ky4)*h,0,(-ky1+ky2-ky3+ky4)*a;
0,0,kz1+kz2+kz3+kz4,(kz1+kz2-kz3-kz4)*b,(kz1-kz2+kz3-kz4)*a,0;
0,(-ky1-ky2-ky3-ky4)*h,(kz1+kz2-kz3-
kz4)*b,((kz1+kz2+kz3+kz4)*b^2)+((ky1+ky2+ky3+ky4)*h^2),(kz4-kz2+kz1-kz3)*b*a,(-
ky4+ky3+ky1-ky2)*h*a;
(kx1+kx2+kx3+kx4)*h,0,(kz1-kz2+kz3-kz4)*a,(kz4-kz2+kz1-
kz3)*b*a,((kz1+kz2+kz3+kz4)*a^2)+((kx1+kx2+kx3+kx4)*h^2),(kx4+kx3-kx2-kx1)*h*b;
(kx4+kx3-kx2-kx1)*b,(-ky1+ky2-ky3+ky4)*a,0,(-ky4+ky3+ky1-ky2)*h*a,(kx4+kx3-kx2-
kx1)*h*b,((ky1+ky2+ky3+ky4)*a^2)+((kx1+kx2+kx3+kx4)*b^2)];

%----- Identity and Zero Matrices -----
I6=eye(6,6); I3=eye(3,3); O6=zeros(6); O3=zeros(3);

%----- Cu Matrix -----
Cu=[I3,O3;[0,-1*az,ay;az,0,-1*ax;-1*ay,ax,0],I3];

%----- STATE SPACE MATRICES -----
ssA=[O6,I6;(-1)*inv(M)*K,(-1)*inv(M)*C];
ssB=[O6;inv(M)]*Cu;
ssC=[I6,O6];
ssD=O6;

%===== SIGNAL CONDITIONING =====

%----- Remove Constant Offset -----
u = dtrend(u1);
yX = dtrend(y1X); yY = dtrend(y1Y); yZ = dtrend(y1Z);
yTH = dtrend(y1TH); yPH = dtrend(y1PH); yPS = dtrend(y1PS);

%----- Time Scale -----
deltaT = 1/smplFreq; % sampling interval, sec
sampleTime = dataSize*deltaT; % total time of sampling, sec

% the following vector "time" defines sampling instances
time = 0:deltaT:sampleTime-deltaT;

%-----Plotting Input & Output Signals -----

```

```

plot(time(1:PDSZ),(10^(-1))*u(1:PDSZ),'y--',time(1:PDSZ),(10^6)*yX(1:PDSZ),'r-
',time(1:PDSZ),(10^6)*yY(1:PDSZ),'c.',time(1:PDSZ),(10^6)*yZ(1:PDSZ),'g-.');
xlabel('Time, [sec]');
title('Input(y), [10N] and Output: X(r), Y(c), Z(g), [micro-m] Signals');
disp('Sample input and output in X,Y,Z direction data plotted. Hit any key to plot angles.');
```

```

text((dataSize(1)-150)*deltaT,0,name)
pause; disp("");

plot(time(1:PDSZ),(10^(-1))*u(1:PDSZ),'y--',time(1:PDSZ),(10^6)*yTH(1:PDSZ),'r-
',time(1:PDSZ),(10^6)*yPH(1:PDSZ),'c.',time(1:PDSZ),(10^6)*yPS(1:PDSZ),'g-.');
xlabel('Time, [sec]');
title('Input(y), [10N] and Output: Theta(r), Phi(c), Psi(g), [micro-rad] Signals');
text((dataSize(1)-150)*deltaT,0,name)
disp('Sample input and output in Theta, Phi, Psi rotation data plotted. Hit any key to plot model-
based responses');
pause; disp("");
```

```
%----- Simulate Impulse Signal -----
```

```
t=0:dataSize;
for i=1:dataSize T0(i)=0; T(50)=1; end;
```

```
%===== SIMULATE MODEL BASED RESPONSE USING REAL INPUT SIGNAL
```

```
=====
```

```
z5=zeros(size(u));
RealUz=[z5,z5,u,z5,z5,z5];
[time2 Zresp]=lsim(ssA,ssB,ssC,ssD,RealUz,time);
```

```
%===== FIND GENERALIZED COORDINATE OF CORNER C
```

```
=====
```

```

for i = 1:dataSize
    T01=[cos(Zresp(i,6)),-sin(Zresp(i,6)),0,0;
        sin(Zresp(i,6)),cos(Zresp(i,6)),0,0;
        0,0,1,0;0,0,0,1];

    T12=[cos(Zresp(i,5)),0,sin(Zresp(i,5)),0;0,1,0,0;
        -sin(Zresp(i,5)),0,cos(Zresp(i,5)),0;0,0,0,1];

    T23=[1,0,0,0;0,cos(Zresp(i,4)),-sin(Zresp(i,4)),0;
        0,sin(Zresp(i,4)),cos(Zresp(i,4)),0;0,0,0,1];

    T34=[0,0,0,Zresp(i,1);0,0,0,Zresp(i,2);
        0,0,0,Zresp(i,3);0,0,0,1];

    DGC=[-lCG;-wCG;-hCG;1];

    DOC=T01*T12*T23*T34*DGC;

    for j=1:3 ZofC(i,j)=DOC(j); end;
```



```

        for j=4:6 ZofC(i,j)=Zresp(i,j); end;
end;

%----- Spread Output into Separate Array -----

%----- initial array -----
%genX=array(dataSize,1);

%----- Before Transform -----
for i=1:dataSize
    genGX(i,1)=Zresp(i,1);
    genGY(i,1)=Zresp(i,2);
    genGZ(i,1)=Zresp(i,3);
    genGTH(i,1)=Zresp(i,4);
    genGPH(i,1)=Zresp(i,5);
    genGPS(i,1)=Zresp(i,6);
end;
%----- After Transform -----
for i=1:dataSize
    genX(i,1)=ZofC(i,1);
    genY(i,1)=ZofC(i,2);
    genZ(i,1)=ZofC(i,3);
    genTH(i,1)=ZofC(i,4);
    genPH(i,1)=ZofC(i,5);
    genPS(i,1)=ZofC(i,6);
end;

%===== PLOT SIGNAL BASED AND MODEL BASED RESPONSES IN COMPARISON
=====

TranGain=1
RotGain=1
ForceGain=0

plot(time(1:PDSZ),ForceGain*u(1:PDSZ),'y.',time(1:PDSZ),genX(1:PDSZ),'m-
',time(1:PDSZ),TranGain*yX(1:PDSZ),'g--');
xlabel('Time, [sec]');
title('Simulated (solid) and Actual (dash) in X direction');
disp('Simulated (solid) and Actual (dash) Response in X direction plotted. Hit any key to
continue.');
```

text((dataSize(1)-150)*deltaT,0,name)

```

    pause;
plot(time(1:PDSZ),ForceGain*u(1:PDSZ),'y.',time(1:PDSZ),genY(1:PDSZ),'m-
',time(1:PDSZ),TranGain*yY(1:PDSZ),'g--');
xlabel('Time, [sec]');
title('Simulated (solid) and Actual (dash) in Y direction');
disp('Simulated (solid) and Actual (dash) Response in Y direction plotted. Hit any key to
continue.');
```

text((dataSize(1)-150)*deltaT,0,name)

```

    pause;
    plot(time(1:PDSZ),ForceGain*u(1:PDSZ),'y.',time(1:PDSZ),genZ(1:PDSZ),'m-
',time(1:PDSZ),TranGain*yZ(1:PDSZ),'g--');
    xlabel('Time, [sec]');
    title('Simulated (solid) and Actual (dash) in Z direction');
    disp('Simulated (solid) and Actual (dash) Response in Z direction plotted. Hit any key to
continue.');
```

text((dataSize(1)-150)*deltaT,0,name)

```

    pause;
    plot(time(1:PDSZ),ForceGain*u(1:PDSZ),'y.',time(1:PDSZ),genTH(1:PDSZ),'m-
',time(1:PDSZ),RotGain*yTH(1:PDSZ),'g--');
    xlabel('Time, [sec]');
    title('Simulated (solid) and Actual (dash) in THETA rotation');
    disp('Simulated (solid) and Actual (dash) Response in THETA rotation plotted. Hit any key to
continue.');
```

text((dataSize(1)-150)*deltaT,0,name)

```

    pause;
    plot(time(1:PDSZ),ForceGain*u(1:PDSZ),'y.',time(1:PDSZ),genPH(1:PDSZ),'m-
',time(1:PDSZ),RotGain*yPH(1:PDSZ),'g--');
    xlabel('Time, [sec]');
    title('Simulated (solid) and Actual (dash) in PHI rotation');
    disp('Simulated (solid) and Actual (dash) Response in PHI rotation plotted. Hit any key to
continue.');
```

text((dataSize(1)-150)*deltaT,0,name)

```

    pause;
    plot(time(1:PDSZ),ForceGain*u(1:PDSZ),'y.',time(1:PDSZ),genPS(1:PDSZ),'m-
',time(1:PDSZ),RotGain*yPS(1:PDSZ),'g--');
    xlabel('Time, [sec]');
    title('Simulated (solid) and Actual (dash) in PSI rotation');
    disp('Simulated (solid) and Actual (dash) Response in PSI rotation plotted. Hit any key to
continue.');
```

text((dataSize(1)-150)*deltaT,0,name)

```

    pause;

%----- Bode Plot -----
%Bode(ssA,ssB,ssC,ssD);
%title('Bode Plot of Dynamometer model (Xz/Fz)');
%text((dataSize(1)-150)*deltaT,0,'name')

%===== ARRANGE ARRAY AND SAVE TO DATA FILE =====

tosave=[yX,yY,yZ,genX,genY,genZ,u];
save MDNERZA.COD tosave -ascii;
```

Appendix D Descriptions of the LabVIEW® Visualization Programs

D.1 Data Acquisition Controller.VI

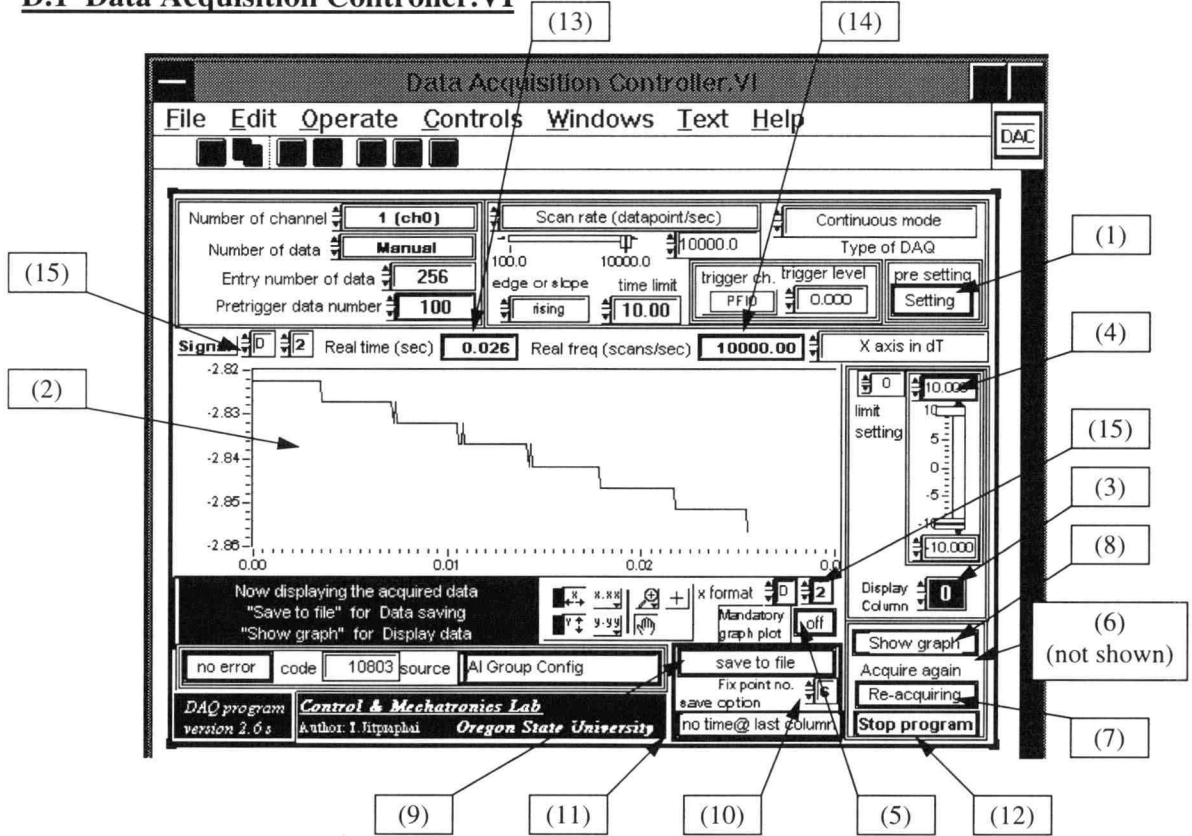


Figure D.1 Front panel of Data Acquisition Controller program (DAC).

| | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Pre-acquisition Mode: A user can set all DAQ parameters, <i>e.g.</i> , sampling frequency and number of data, before applying these parameters to the program DAC. | |
| (1) | Press 'Setting' button (1) if wanting to activate a presetting mode. Program does not acquire data but display acquisition time (13) and sampling frequency (14) calculated from specified DAQ parameters. |
| | Determine suitable DAQ parameters (number of data, sampling frequency, DAQ type, and trigger conditions) and release 'APPLY' button to begin the acquisition. |

| | |
|------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Acquisition Mode: The program DAC reads data. | |
| (2) | When the program is reading data, the acquired data are displayed on the graph (2) with a light green color on a background. However, in a continuous DAQ mode, these data are not recorded. The user can record data at any time by pressing 'capture' button (6). If the acquisition is complete, captured data are then displayed on the graph (2) and the background color is changed into dark blue. |
| (3), (4) | Scan through all channels using the channel selection control (3) and set suitable gain for each channel using limit setting control (4). Precision of the displayed data can be set using the precision selection (15). |
| (5) | If data size is bigger than 60,000 points, program will suppress the data plotting on the graph to prevent a memory failure. However, the user can force the program to plot these data on the graph by switching 'mandatory plot' (5) switch to 'ON'. |

| | |
|-------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Post- acquisition Mode: the user can look at the acquired data, save data to a disk, and capture new data. | |
| (7) | Press 'Re-acquiring' button (7) to capture a new set of data. Program will immediately begin new data acquisition using previously set DAQ parameters unless the 'Presetting' button (1) is pressed. |
| (8) | Press 'Show graph' button (8) to activate a data monitor program that plots acquired data on a big graph. |
| (9) | Press 'Save data' button (9) to save data in a 'spreadsheet compatible' file that contains m rows and n column; m = number of data; n = number of channels. A user can set a precision of data by selecting number of digits after point (10). The user can also attach time information by pressing button (11) to ON in which the program DAC creates an additional column containing time data and attaches this column to the last column of the data set. |
| (12) | Press 'Stop program' button to finish the program. |

D.2 Signal Processor.VI

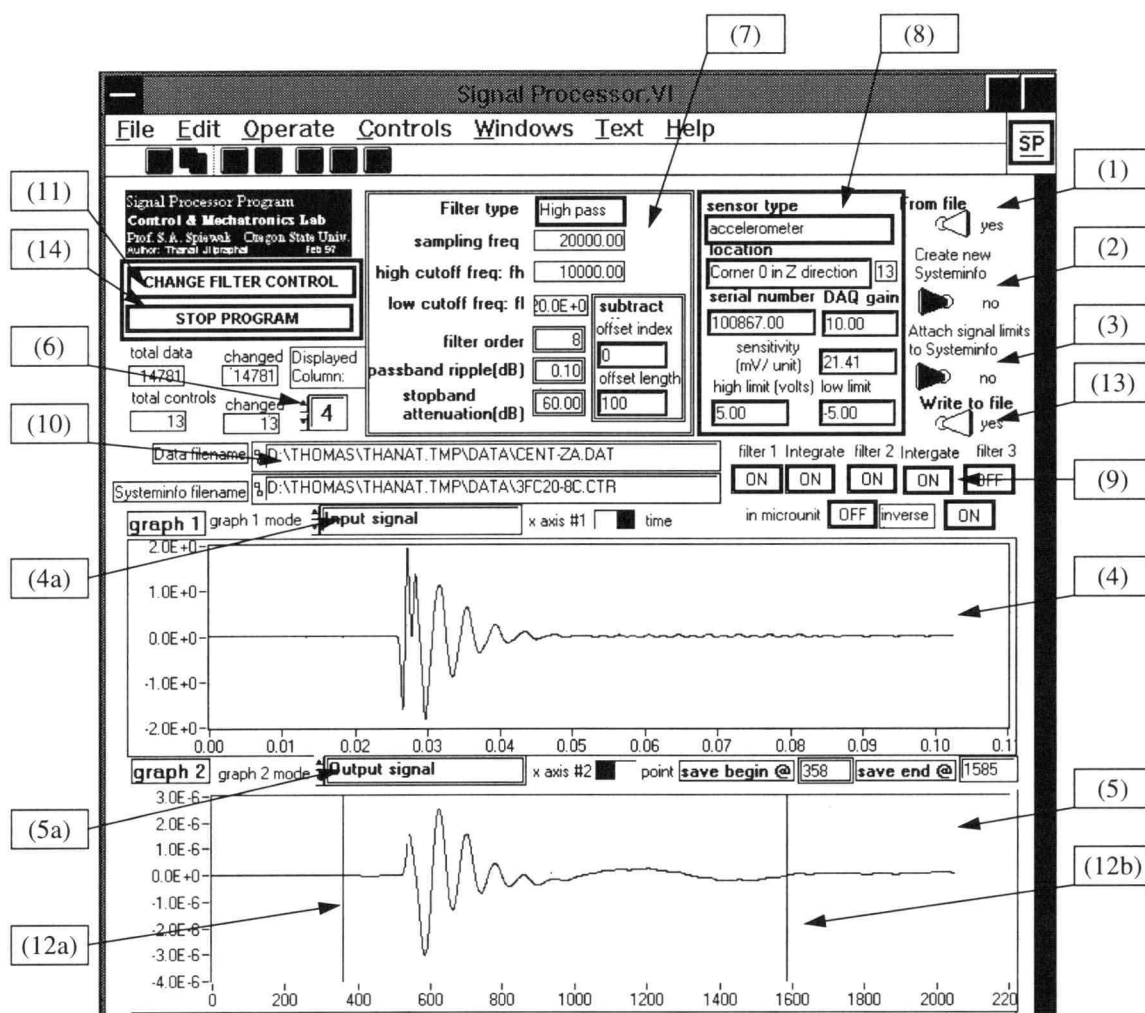


Figure D.2 Front panel of the signal processor program (SP).

| Before running the program SP | |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | Select 'from file' switch to "yes" to read data and Systeminfo from spreadsheet files. |
| (2) | Select 'create new Systeminfo' switch to "yes" to build an empty Systeminfo file. Program will ignore the Systeminfo file that is read from step (1). |
| (3) | Select "yes" to replace information of signal limits in the Systeminfo with DAQ limit values that are 'wired' from the program DAC to the <i>DAQ limit</i> terminal. |

| Run the program SP: Program processes data in all channels using the specified Systeminfo and displays data on the graph (4) and (5). | |
|---------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|
| (4) | Use the graph mode selection (4a) to select data type to be plotted on graph (4). |
| (5) | Use the graph mode selection(5a) to select data type to be plotted on graph (5). |

| | |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (6) | Use the column selection (6) to select a data column to be displayed. |
| (7) | Display of filter parameters and subtracting-average parameters that are used in the signal processing of data in the selected and displayed channel. |
| (8) | Display of sensor information and DAQ parameters that are associated with data in the selected and displayed channel. |
| (9) | Display signal processing procedure that is applied to data in the selected and displayed channel. |
| (10) | Display file names and paths of the data and the Systeminfo files |
| (11) | Press 'Change filter control' button to change any parameters in signal processing parameters (in the Systeminfo). A <i>Systeminfo changing window</i> will 'pop up' for the user to set suitable processing parameters. After the 'APPLY' button is pressed, the program SP will process the data in the selected channel according to new parameters. |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| Before stopping the program | |
| (12) | Select portion of data to be saved (or used in next programs) by moving the beginning bar (12a) and the ending bar (12b) to specify the range of data. |
| (13) | Select 'save data' switch to "yes" to save data and Systeminfo. Program will do a save procedure after the program is stopped. |
| (14) | Press 'Stop program' to finish the program. |

The user can also 'wire' true signal to the finish program terminal for controlling the program SP to run only one time. The program SP will process all data according to the information in the Systeminfo and then stop.

D.3 Generalized Coordinate Calculator.VI

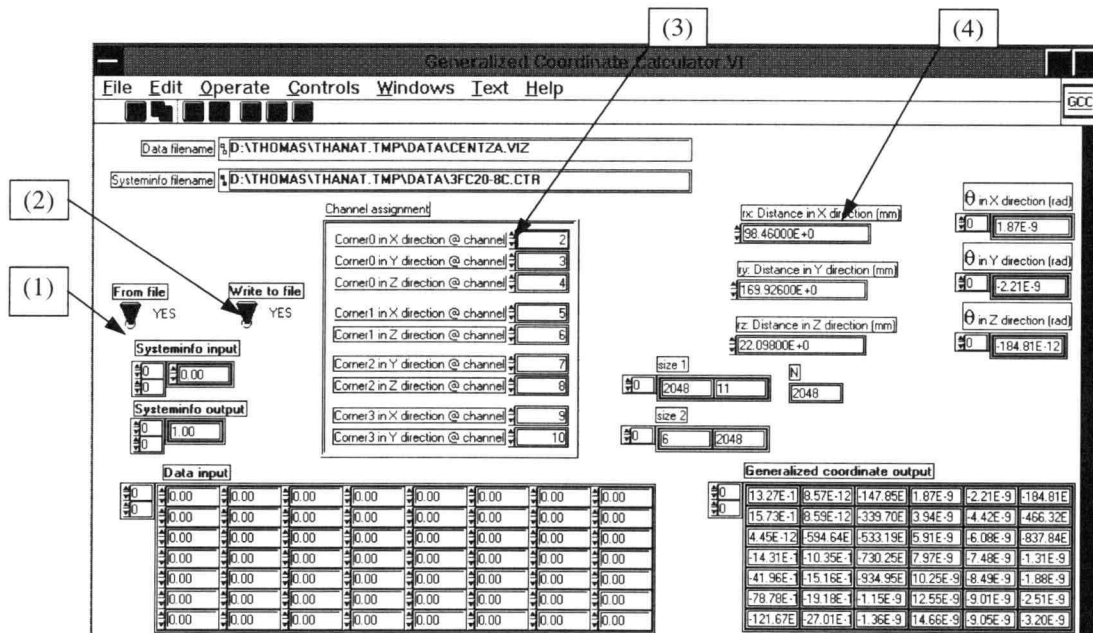


Figure D.3 Front panel of the generalized coordinate calculator program (GCC).

| Before running the program GCC. | |
|---------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | Select 'from file' switch to 'yes' to read data from a spreadsheet file. A user can select data sources weather to read data from a spreadsheet compatible file or to use data wired from another program at the 'data input' terminal. |
| (2) | Select 'write to file' switch to 'yes' to save data. Program will do a save procedure after the program is stopped. |
| (3) | Assign channel numbers for specified variables used in Padgaonkar's equations (Eq. 4.13 or Eq. 4.14). |
| (4) | Input distances between the accelerometers used in calculation (Eq. 4.13 or Eq. 4.14). |

D.4 3-D Animation Generator.VI

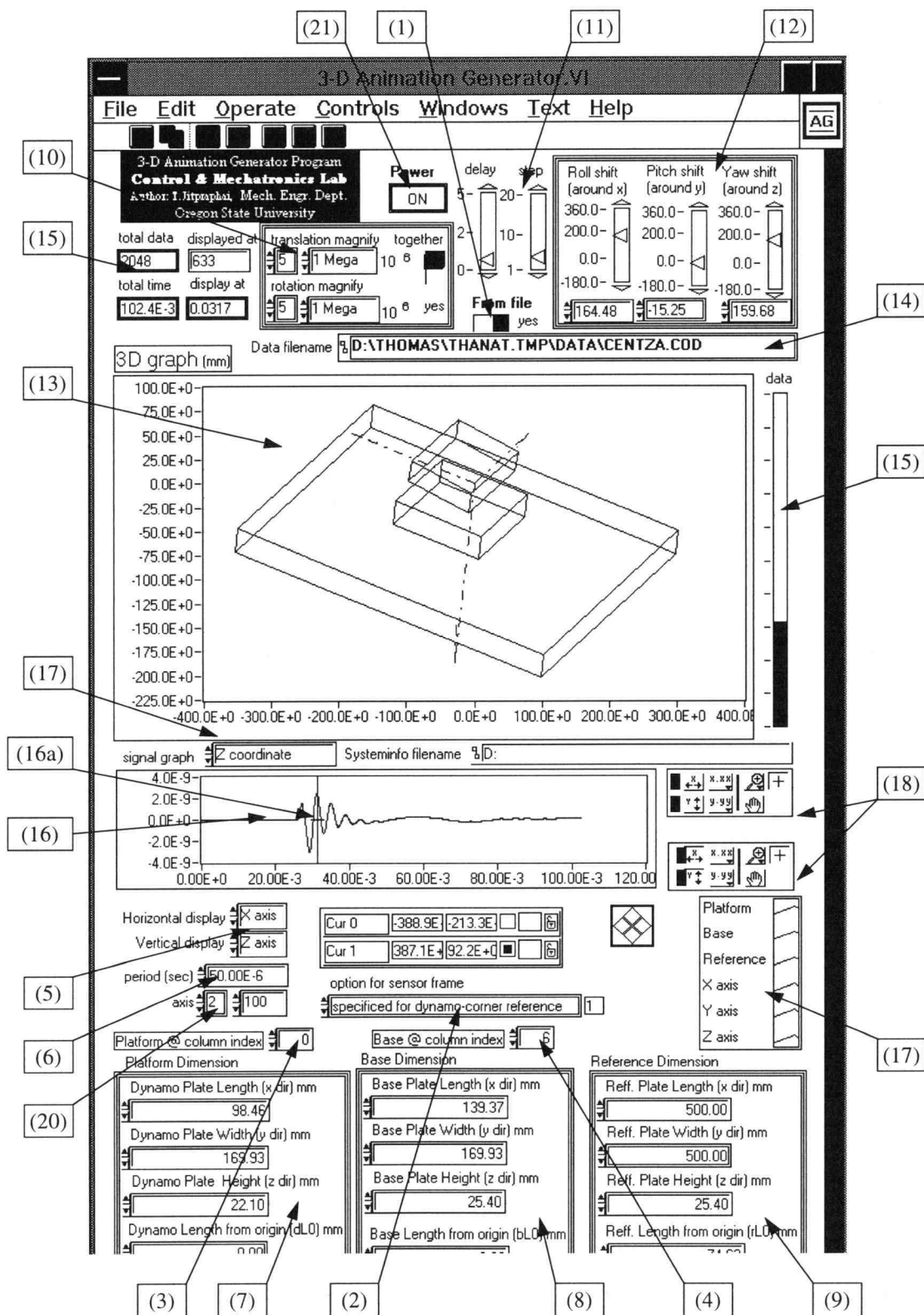


Figure D.4 Front panel of the 3-D animation generator program (AG).

| Before running the program AG. | |
|---------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (1) | Select 'from file' switch to "yes" to read data from a spreadsheet file. |
| (2) | Specify drawing option . Two types of drawing are available. 1. <u>Center reference option.</u> Program animates plate's vibrations using generalized coordinate data of the center of mass G that refers vibrations of C.S. $(XYZ)_G$. to the global C.S. (XYZ) . Origin of reference axes (XYZ) is at a center of a top surface of the reference plate with axis of the (XYZ) are drawn (Fig. D.5). 2. <u>Specify for dynamometer-corner reference.</u> Program animates plate's vibrations using the generalized coordinate data of the corner point, C , that refers vibrations C.S. $(XYZ)_C$. to the global reference C.S. (XYZ) . Axes of the $(XYZ)_C$. are drawn with its origin at the corner. |
| (3) | Specify beginning column index of platform's generalized coordinate from data file. |
| (4) | Specify beginning column index of base plate's generalized coordinate from data file. |
| (5) | Specify horizontal axis and vertical axis of the viewing screen whose the 3-D object is projected on (when no rotation of a viewpoint is applied). |
| (6) | Specify period of data for calculating time display in (15). |
| (7) | Specify platform's dimension and its center of mass (or corner) position referring to reference coordinate system (XYZ) . |
| (8) | Specify base plate's dimension and its center of mass (or corner) position referring to reference coordinate system (XYZ) . |
| (9) | Specify reference plate's dimension and its center of mass (or corner) position referring to reference coordinate system (XYZ) . |

| Run the program AG. | |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| (10) | Set magnification factors of translations (H_t) and rotations (H_r) to be displayed on the 3-D graph. |
| (11) | Specify animation speed. <i>Delay</i> control is used to set a delay time between two consequent plots in the animation. <i>Step</i> control is used to specify number of pictures to be skipped in the animation. |
| (12) | Set rotating angles of the viewpoint coordinate system $(XYZ)_V$. |
| (13) | 3-D picture of plates is animated on the X-Y graph. |
| (14) | Displaying data's filename and path. |
| (15) | Total data number and current data number are displayed. Percentage of the displayed data is also shown graphically in the vertical bar (15). |
| (16) | Data in the generalized coordinate are displayed on the graph (16). The displayed column can be selected using the column selection (17). The moving bar (16a) shows the currently displayed data. |
| (18) | A user can utilize a palette to zoom, scale and move picture. |
| (19) | Color of the plates and the axes on the 3-D graph can be selected. |
| (20) | Control length of three reference axes (X , Y and Z). |
| (21) | Press 'Power' button to "off" to finish the program. |

Figures D.5 and D.6 show definitions of dimensions used for generating 'wireframe' plates that are displayed on the program's 3-D graph. The pictures shown in Fig. D.5 are generated under drawing option of 'center reference option' whereas the 'corner reference' drawing option is defined in Fig. D.6.

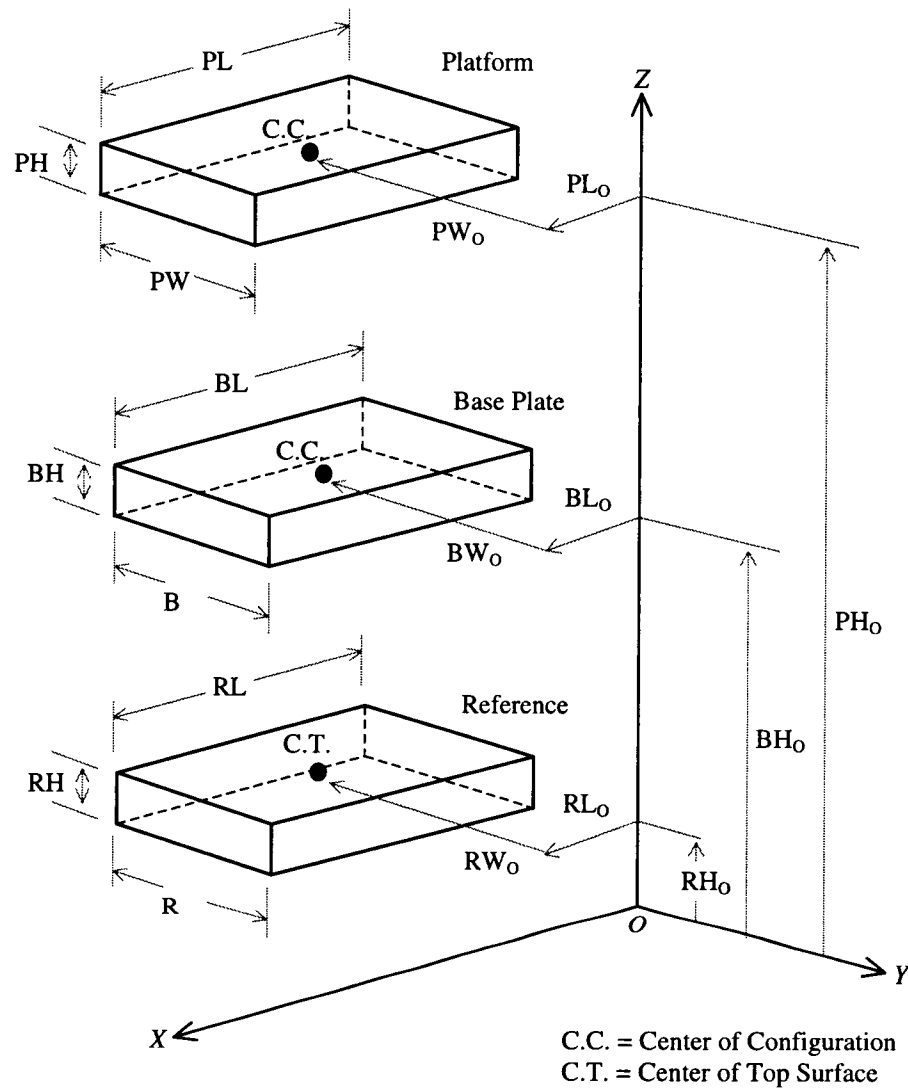


Figure D.5 Dimensions used in the 'center reference' drawing option.

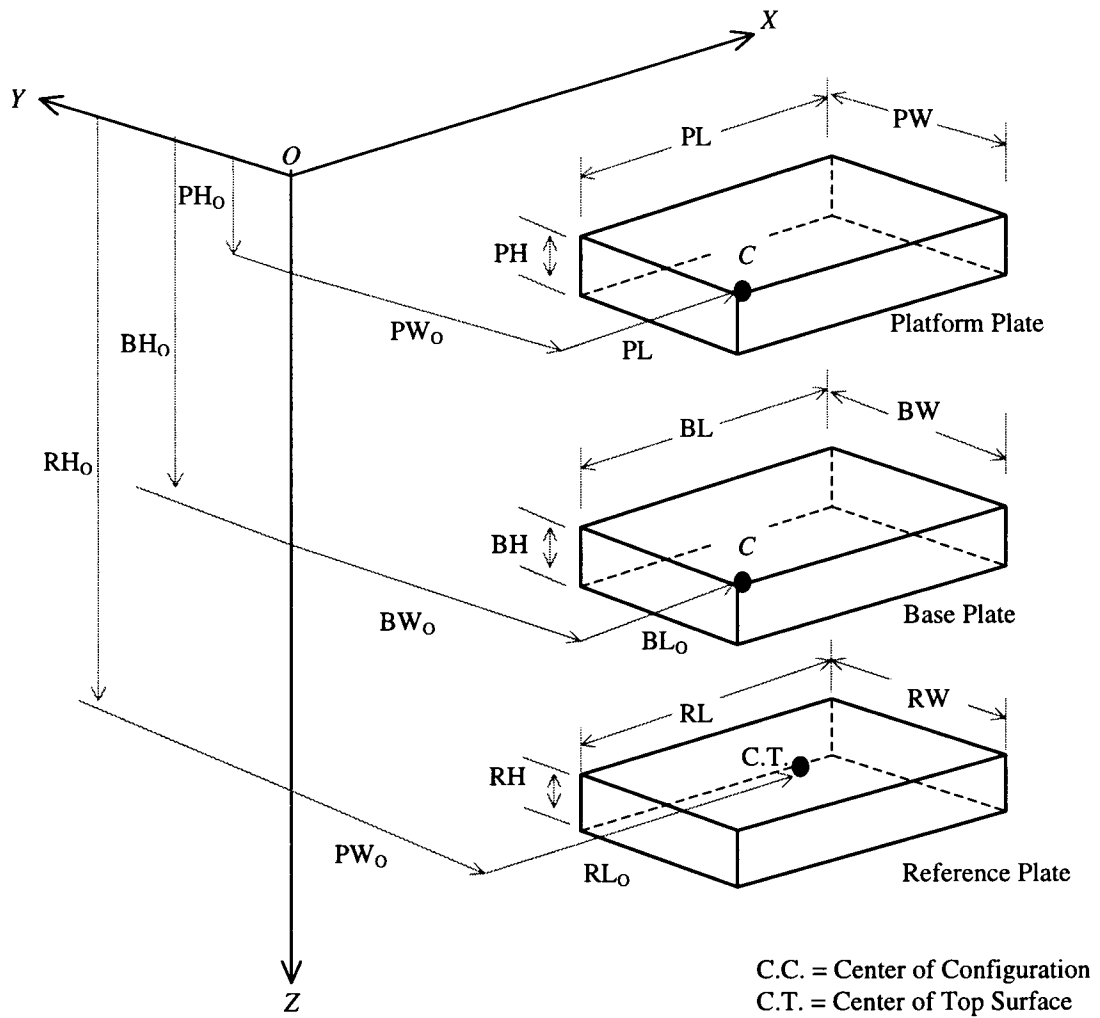


Figure D.6 Dimensions used in the 'corner reference' drawing option.

Appendix E Data Management in the LabVIEW® Visualization Programs

E.1 Types of Data

Since many processes have been performed to transform the acceleration data into the generalized coordinates, explanations of the data types and their meaning are provided in this Section. Figure E.1 shows transformations of data in the signal processing and the generalized coordinate calculation programs. Data types are displayed in round boxes with shadow whereas the processing programs are shown in rectangular boxes. Formats of these data are shown in Table E.1.

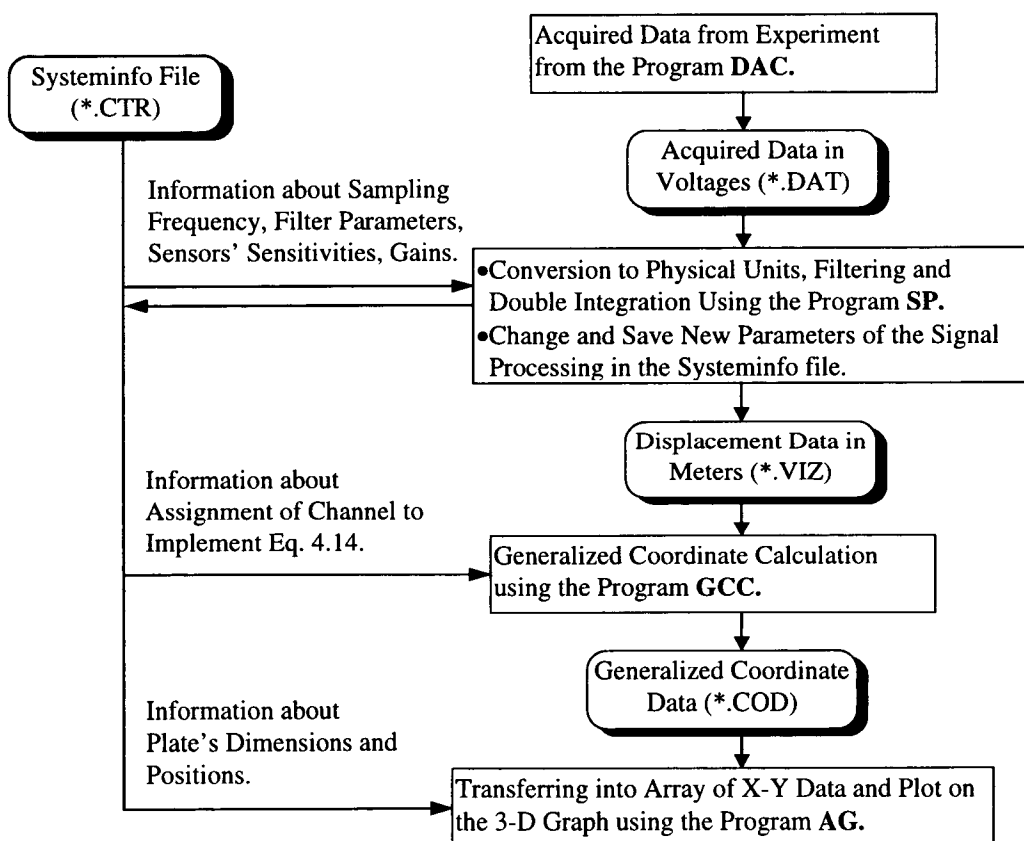


Figure E.1 Data system.

| Data type | Format |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| *.DAT | 2-D array of m rows and n columns; m = number of data; n = number of channels used in the DAQ system. |
| *.VIZ | 2-D array of m rows and n columns; m = number of data; n = number of channels used in the DAQ system. |
| *.COD | 2-D array of m rows and 6 columns for one plate (or 12 columns for two plates); m = number of data. Each row contains data in a format of $[\mathbf{d}_P^T \quad \mathbf{d}_B^T]$, where \mathbf{d}_P = the generalized coordinate of the platform and \mathbf{d}_B = the generalized coordinate of the base plate (option). |
| *.CTR | 2-D array of 34 rows and n columns; n = number of channels (see Table E.2). |

Table E.1 Formats of data used in the visualization programs.

E.2 Systeminfo File

A *Systeminfo* file is a data file created for describing the information of signal data obtained from an experiment. All signal processing procedures can also be performed automatically using information recorded in the Systeminfo. A Systeminfo or a control file contains all information of data in a spreadsheet compatible format in which number of columns of the Systeminfo equals to number of channels used in the DAQ³. One Systeminfo file should be created for each experiment since the parameters used in an experiment setup are unique. These parameters include sampling frequency, DAQ limits, transducer type, serial number and sensitivities, and sensor's locations, *etc.* A user can create or edit a Systeminfo using the program 'Signal Processor.VI'. An assignment of a Systeminfo that is used as a standard format in this research is shown in Table E.2.

³ One column is assigned for one channel.

| Row | Description | Values in Systeminfo (XXX = values) |
|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| 0 | Sensor type | 0: No information (default). 1: Loadcell. 2: Proximity sensor. 3: Voltage signal. 4: Accelerometer. |
| 1 | Locations of sensors | XXX |
| 2 | Serial numbers of sensors | XXX |
| 3 | Sensor's sensitivity | XXX |
| 4 | Integration selection | 0: No integration (default). 1: Double integration is applied. 2: Single integration is applied. |
| 5 | Digital filter type selection | 0: Low-pass filter (default). 1: High-pass filter. 2: Band-pass filter. 3: Band-stop filter. |
| 6 | Filter order | XXX |
| 7 | Sampling frequency | XXX |
| 8 | High cut-off frequency of the filters | XXX |
| 9 | Low cut-off frequency of the filters | XXX |
| 10 | Pass-band ripple of the filters | XXX |
| 11 | Stop-band attenuation of the filters | XXX |
| 12 | <u>Filter ON/OFF selection</u> Set number based on binary number basis for controlling 3 filters, e.g., (1) ₁₀ = filter1 ON, filter2 OFF, filter3 = OFF (001) ₂ , (4) ₁₀ = filter1 OFF, filter2 ON, filter3 = OFF (010) ₂ , (16) ₁₀ = All filter is ON = (111) ₂ . | 0 .. 16 (0 is the default) |
| 13 | Converse data to microunit If 'YES' is selected, program will multiply data in the associated column by 1 million. | 0: NO (default) 1: YES |
| 14 | Inverse direction of signal If 'YES' is selected, program will multiply data in the associated column by (-1). | 0: NO (default) 1: YES |
| 15 | Beginning index number of data to calculate averaged value used in the subtracting average procedure. | XXX |
| 16 | Length of data in the subtracting average procedure. | XXX |
| 17 | Gain number used in the DAQ's amplifier (G_p). | XXX |
| 18 | Filter type selection | 0: Elliptic digital filter (default). 1: Butterworth digital filter. |
| 19 | High limit setting of DAQ for this signal (column). | XXX |
| 20 | Low limit setting of DAQ for this signal (column). | XXX |
| 21 | <u>Column 0</u> : selection to use information from Systeminfo file in the program GCC. <u>Column 1</u> : selection to use information from Systeminfo file in the program AG. | 0: NO (default) 1: YES use Systeminfo |
| 22 | <u>Column 0</u> : Platform's length in X direction, PL. <u>Column 1</u> : Base's length in X direction, BL. <u>Column 2</u> : Reference's length in X direction, RL. | XXX (mm) |

Table E.2 Systeminfo assignment.

| | | |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|
| 23 | <u>Column 0</u> : Platform's width in Y direction, PW . <u>Column 1</u> : Base's width in Y direction, BW . <u>Column 2</u> : Reference's width in Y direction, RW . | XXX (mm) |
| 24 | <u>Column 0</u> : Platform's height in Z direction, PH . <u>Column 1</u> : Base's height in Z direction, BH . <u>Column 2</u> : Reference's height in Z direction, RH . | XXX (mm) |
| 25 | <u>Column 0</u> : Platform's length from origin, PL_O . <u>Column 1</u> : Base's length from origin, BL_O . <u>Column 2</u> : Reference's length from origin, RL_O . | XXX (mm) |
| 26 | <u>Column 0</u> : Platform's width from origin, PW_O . <u>Column 1</u> : Base's width from origin, BW_O . <u>Column 2</u> : Reference's width from origin, RW_O . | XXX (mm) |
| 27 | <u>Column 0</u> : Platform's height from origin, PH_O . <u>Column 1</u> : Base's height from origin, BH_O . <u>Column 2</u> : Reference's height from origin, RH_O . | XXX (mm) |
| 28 | Option for plotting picture | 0: Reference frame at plate's origin, O . 1: Reference frame at plate corner, C_R . |
| 29 | <u>Column 0</u> : Platform generalized coordinate's column index. <u>Column 1</u> : Base generalized coordinate's column index. | XXX |
| 30 | r_x : Distance between accelerometers in the X_I direction (Eq. 4.12). | XXX (mm) |
| 31 | r_y : Distance between accelerometers in the Y_I direction (Eq. 4.12). | XXX (mm) |
| 32 | r_z : Distance between accelerometers in the Z_I direction (Eq. 4.12). | XXX (mm) |
| 33 | Assignment of data column for calculation in Eq. 4.13 or Eq. 4.14. <u>Column 0</u> : x_{xC} , corner C in the X_I direction. <u>Column 1</u> : x_{yC} , corner C in the Y_I direction. <u>Column 2</u> : x_{zC} , corner C in the Z_I direction. <u>Column 3</u> : x_{x1} , corner 1 in the X_I direction. <u>Column 4</u> : x_{z1} , corner 1 in the Z_I direction. <u>Column 5</u> : x_{y2} , corner 2 in the Y_I direction. <u>Column 6</u> : x_{z2} , corner 2 in the Z_I direction. <u>Column 7</u> : x_{x3} , corner 3 in the X_I direction. <u>Column 0</u> : x_{y3} , corner 3 in the Y_I direction. | XXX |

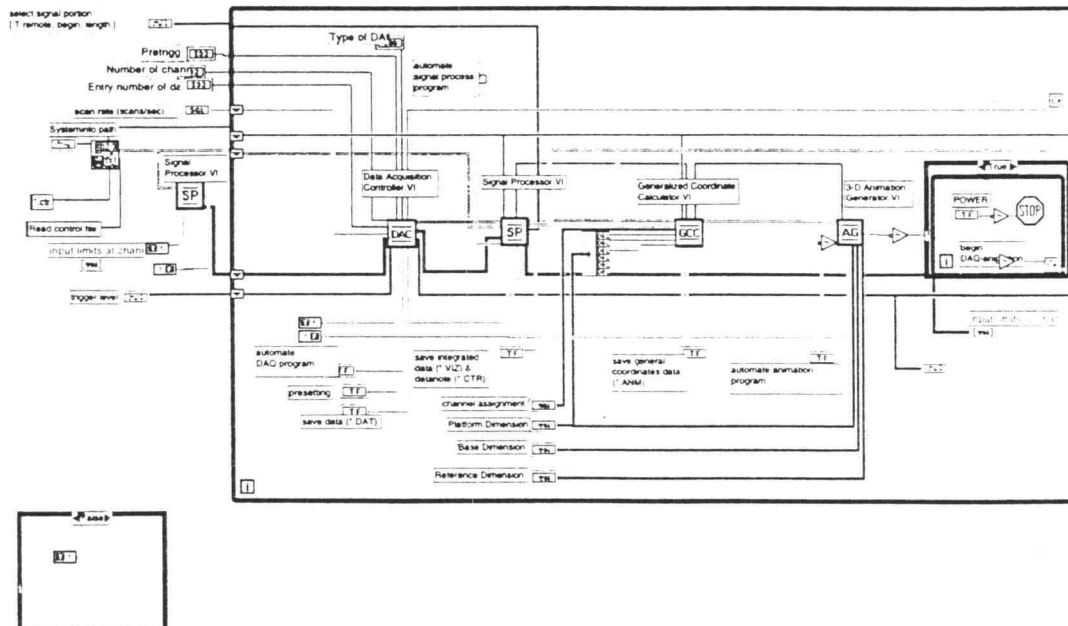
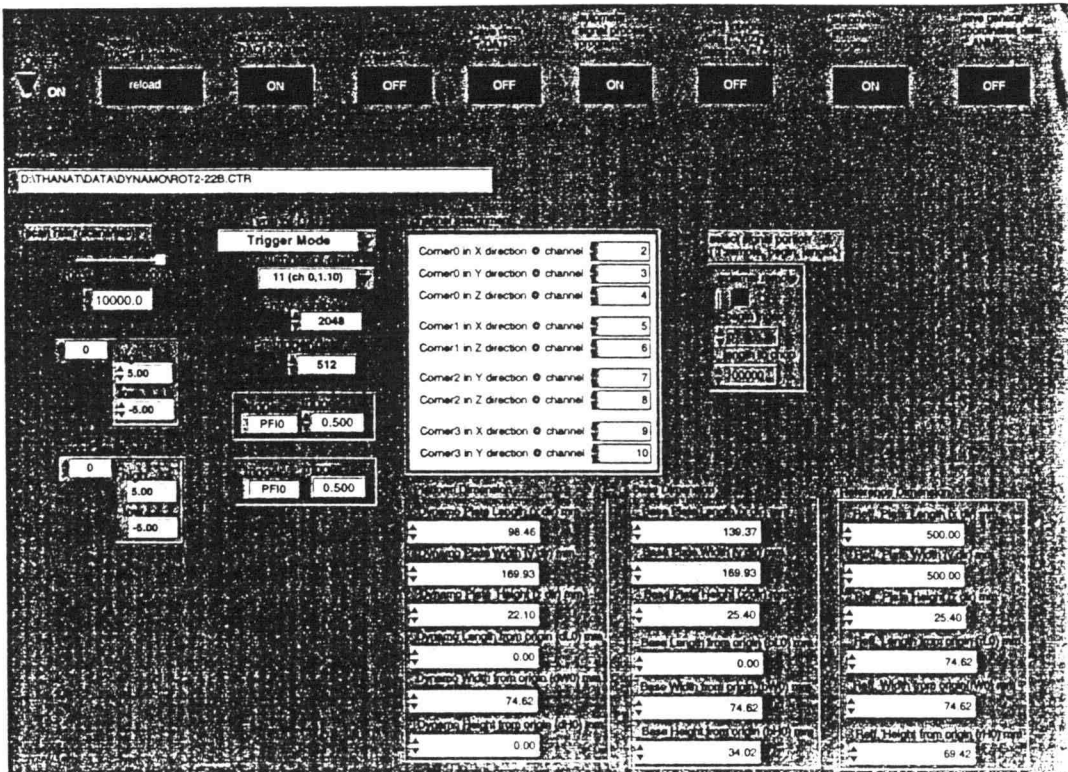
Table E.2 (continued) Systeminfo assignment.

Table E.3 shows an example of the Systeminfo file that is designed for the experiment in Chapter 5.

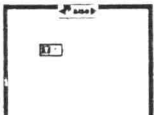
| | Column | | | | | | | | | | |
|-----|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Row | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 0 | 1 | 6 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 1 | 10 | 0 | 11 | 12 | 13 | 14 | 16 | 18 | 19 | 20 | 21 |
| 2 | 1519 | 0 | 100868 | 100859 | 100867 | 100852 | 01813 | 100858 | 100804 | 101812 | 100862 |
| 3 | 2.248 | 1000 | 20.292 | 20.903 | 21.413 | 20.088 | 20.496 | 20.19 | 20.699 | 20.394 | 19.986 |
| 4 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 5 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 6 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| 7 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 | 20000 |
| 8 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 | 10000 |
| 9 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 |
| 10 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 |
| 11 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 | 60 |
| 12 | 0 | 0 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 16 | 400 | 400 | 400 | 100 | 100 | 400 | 100 | 100 | 100 | 100 | 100 |
| 17 | 10 | 1000 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 |
| 18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 19 | 5 | 3 | 0.5 | 0.5 | 5 | 0.5 | 5 | 0.5 | 5 | 0.5 | 0.5 |
| 20 | -5 | -2 | -0.5 | -0.5 | -5 | -0.5 | -0.5 | -0.5 | -5 | -0.5 | -0.5 |
| 21 | 1 | 1 | - | - | - | - | - | - | - | - | - |
| 22 | 98.46 | 139.37 | 500 | - | - | - | - | - | - | - | - |
| 23 | 169.93 | 169.93 | 500 | - | - | - | - | - | - | - | - |
| 24 | 22.10 | 25.40 | 25.40 | - | - | - | - | - | - | - | - |
| 25 | 0 | 0 | 74.82 | - | - | - | - | - | - | - | - |
| 26 | 74.82 | 74.82 | 74.82 | - | - | - | - | - | - | - | - |
| 27 | 0 | 23.00 | 38.10 | - | - | - | - | - | - | - | - |
| 28 | 1 | - | - | - | - | - | - | - | - | - | - |
| 29 | 0 | 6 | - | - | - | - | - | - | - | - | - |
| 30 | 98.46 | - | - | - | - | - | - | - | - | - | - |
| 31 | 169.93 | - | - | - | - | - | - | - | - | - | - |
| 32 | 22.10 | - | - | - | - | - | - | - | - | - | - |
| 33 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | - | - |

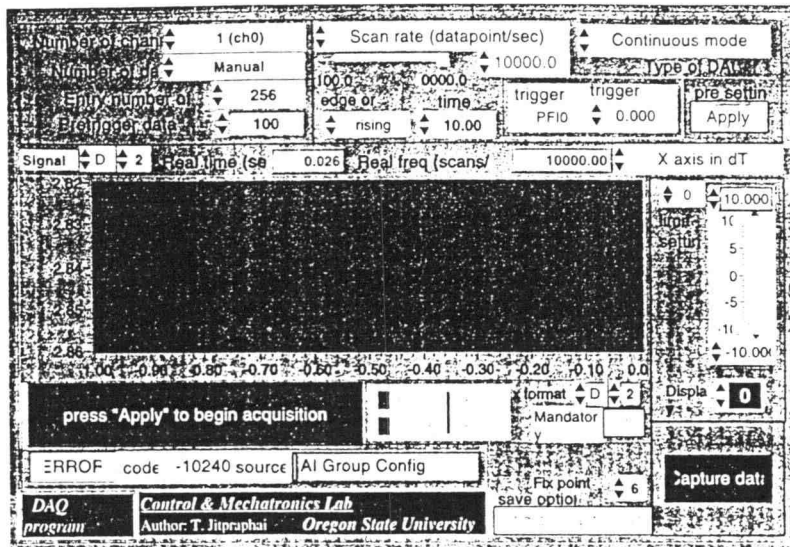
Table E.3 Example of the Systeminfo file used in the experiment.

Appendix F
Block Diagrams of the LabVIEW® Visualization Programs



VIBRATION VISUALIZATION 1





[*]** coupling & input config (no change:0)
 ((cluster) coupling & input config: an array of clusters, of which each array element specifies the coupling and input configuration for the channel(s) in the corresponding element of the channels array. If there are fewer elements in this array than the number of channels, the VI uses the last element for the remaining channels. The default input is an empty array, which means the parameters do not change from their default settings. Each cluster contains the following parameters:

- (u16) coupling:
 - 0: Do not change the coupling setting
 - 1: DC
 - 2: AC
 - 3: Ground
 - 4: Internal reference.


- (u16) input config:
 - 0: Do not change the input config setting
 - 1: Differential
 - 2: Referenced single-ended
 - 3: Nonreferenced single-ended

[*]** coupling & input config is an array of clusters. Each array element assigns the configuration for the channels specified by the corresponding element of the channels array. If there are fewer elements in this array than in channels, the VI uses the last element of coupling & input config for the rest of the channels. The default input for coupling & input config is an empty array, which means the parameters keep their default settings. Each cluster contains the following parameters:


- [U16]** coupling (no change:0)
 coupling. Refer to Chapter 2, Hardware Overview, for information about coupling on your specific type of DAQ board.
 - 0: Do not change the coupling setting
 - 1: DC
 - 2: AC
 - 3: Ground
 - 4: Internal reference.

- [U16]** input config (no change:0)
 input config.
 - 0: Do not change the input config setting
 - 1: Differential (default)
 - 2: Referenced single-ended
 - 3: Nonreferenced single-ended

Data Acquisition Controller.VI
05/30/97 01:20 AM

Page 3 

- [I16]** **device (1)**
The device number is the ID number of the plug-in data acquisition card to be used for the acquisition.
- You can check the device numbers using the configuration utility. The configuration utility displays a board number or slot number next to each National Instruments board; you use that number as the device number for the board.
- [DBL]** **zero array**
- [DBL]**
- [TF]** **Acquire again**
Press this button to begin acquisition again.
- Users can select pre-setting mode to adjust DAQ parameters again.
- [TF]** **Display monitor**
Show acquired data on larger graph (Data monitor ver 2.4s).
- [TF]** **Power**
Valid after program has acquired data successfully (graph color is dark blue and this button blinks).
- [I32]** **Entry number of data**
A scan is defined as a reading taken from each channel in the channel list, taken in the order specified.
- The number of scans to acquire is the number of times to scan through all channels in the list; which also is the number of samples to acquire from each channel in the list.
- [TF]** **Capture data**
Valid only in Continuous mode.
- Program just read data and plots on graph. Until "Capture data" button is pressed, the acquisition then begins.
- [SCL]** **scan rate (scans/sec)**
The scan rate determines how many scans per second to acquire. Because all channels in the channel list are sampled during each scan, each channel sampled at the chosen scan rate.
- [I32]** **Pretrigger data number**
This number is valid in "Trigger mode" DAQ.
- Pretrigger data number specify number of data to be acquired before a trigger-signal occurs (at time = 0).
- [TF]** **Save to file**
Program writes acquired data to a spreadsheet file (ASCII format). To save data, press "save to file" button.
Precision of saved data is specified by setting "fix. point no." (number of digits to the right of the decimal point).
Users can select to save data with time information (a data set shows gradual time number) to last column using "time @ first column".

Page 4 

Data Acquisition Controller.VI
05/30/97 01:20 AM

- [TF]** **pre setting**
In pre-setting ("Apply" is shown), program will not read data but calculates DAQ-parameters. User can justify feasible parameters before applying to real measurement.
- After "Apply" button is pressed, data acquisition then begins.
- [TF]** **Mandatory graph plot**
In case of large data numbers (more than 60,000 scans), program deny to plot data on graph to prevent memory failure.
- Users can force program to plot graph by press "Mandatory graph plot" button.
- [U16]** **Number of data**
Set number of Scans.
In "Manual", user specify number by setting number in "Entry number of data" control (below).
- [U16]** **Type of DAQ**
- (1) Trigger mode: acquisition begins after getting a trigger signal (at channel PFI0 or Trng 0).
 - (2) Continuous mode: program just read signals until "Capture data" button is pressed. Data is then acquired.
 - (3) Read only one time: program acquires data immediately after "Apply" button is pressed.
- [U16]** **Show**
- X axis display option:
- (1) Show time (sec) on X axis.
 - (2) Show point (data number) on X axis.
- Ground display option:
- (3) Show signal and ground (0 volt) in time-axis.
 - (4) Show signal and ground (0 volt) in point-axis.
- Power spectrum option:
- (5) Show auto power spectrum of signal in dB.
- [U16]** **set freq.**
2 options for setting DAQ parameters:
- (1) Set "Scan rate (datapoint/sec)" is to specify a sampling frequency. With data number, this program calculate acquired time.
 - (2) Set "Time to acquire (sec)", with data number, this program calculate sampling frequency.

Data Acquisition Controller.VI
05/30/97 01:20 AM

[I32] Number of channel
Set total number of channels that are employed in DAQ system.

[I32] Display
Column
Select data at specified channel to be displayed.

[U16] edge or slope
edge or slope.
0: Do not change the default setting (default input).
1: Leading edge for digital trigger; positive slope for analog trigger.
1: Trailing edge for digital trigger; negative slope for analog trigger.

[SGL] time limit (sec)
time limit restricts the amount of time LabVIEW waits for the trigger to occur. The default input is -1.0, which tells LabVIEW not to change the time limit set default setting is 0 seconds, no waiting.

[S2] trigger level
analog chan and level is only used when trigger type is analog trigger.

[abc] trigger ch.
trigger channel is valid only when trigger type is 1. trigger channel describes the analog channel that is the source of the trigger following the syn described in the Channel Addressing section of Chapter 2, Getting Started with the Data Acquisition Vis, except for the following cases.
An empty string tells LabVIEW not to change the trigger source setting. An empty string is the default input.
EXTx denotes an external analog input trigger, where x is the channel number. Currently, x must be 0. You can substitute the string EXTERNAL
*****More detail*****
When Trigger type is 1 (analog trigger), 5, or 6, trigger source defaults to 0 for analog input channel 0. The following are valid trigger source values: "n" where n is an analog input channel number
"PFIO" (E series only)
I set trigger channel to PFIO (trig1) in this program.

[SGL] trigger level
level is the voltage value the analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or trailing edge with the edge or slope input. The default input for level is 0.0 V.

[SGL] Time to acquire (sec)
Enter time to be acquire.
Program calculates sampling frequency by
Sampling freq = Number of data (includes pre-trig) / time to acquire

Data Acquisition Controller.VI
05/30/97 01:20 AM

[TF] save option
Users can select to save data with time information attached to last column using "time @ first column".
If this option is selected, program creates a column with equal size as acquired data in which this additional column contains time values gradually increasing according to real period in measurement. This column is attached after last signal column.
If pre-trig number is set, time will begin with negative values (before time = 0 sec).

[I32] Fix point no.
Number of digits to the right of the decimal point of saved data.

[TF] remote (T: read only one time)

[I32] y precision
Number of digits to the right of the decimal point of saved data.

[U16] y format

[I32] x precision
Number of digits to the right of the decimal point of saved data.

[U16] x format

[vsi] limit setting

[vsa] current limit

[vsa] acquired data

[vsa]

[abc] Warning

[vsa] signal limits

input limits is an array of clusters. Each array element assigns the limits for the channels specified by the corresponding element of channels. If there are more elements in this array than in channels, the VI uses the last array element for the rest of the channels. The input limits array defaults an empty array, which means the input limits keep their default settings. Each cluster contains the following parameters.

[vsa] ((cluster)) input limits: an array of clusters, of which each array element specifies the range limits for the channel(s) in the corresponding element of channels array. If there are fewer elements in this array than the number of channels, the VI uses the last element for the rest of the channels. If default input is an empty array, which means the input limits do not change from their default settings. Each cluster contains the following parameters:

(sgl) high limit: specifies the maximum voltage the board measures at a particular channel.

(sgl) low limit: specifies the minimum voltage the board measures at a particular channel.

VIBRATION VISUALIZATION 4

Data Acquisition Controller.VI
05/30/97 01:20 AM

SG1 high limit (10V)
high limit specifies the maximum voltage the board measures at a particular channel.

SG1 low limit (-10V)
low limit specifies the minimum voltage the board measures at a particular channel.

PG1 trigger level
analog chan and level is only used when trigger type is analog trigger.

abc trigger channel
trigger channel is valid only when trigger type is 1. trigger channel describes the analog channel that is the source of the trigger following the syntax described in the Channel Addressing section of Chapter 2, Getting Started with the Data Acquisition VIs, except for the following cases. An empty string tells LabVIEW not to change the trigger source setting. An empty string is the default input. EXTx denotes an external analog input trigger, where x is the channel number. Currently, x must be 0. You can substitute the string EXTERNALMore detail!.....
When Trigger type is 1 (analog trigger), 5, or 6, trigger source defaults to 0 for analog input channel 0. The following are valid trigger source values: "n" where n is an analog input channel number
"PFIO" (E series only)
I set trigger channel to PFIO (trig1) in this program.

SG1 trigger level (0 V)
level is the voltage value the analog source must cross for a trigger to occur. You must also specify whether level must be crossed on a leading or falling slope with the edge or slope input. The default input for level is 0.0 V.

PG1 DAQ error
error out contains error information. If the error in cluster indicated an error, the error out cluster contains the same information. Otherwise, error out describes the status of this VI.

TF status
status is TRUE if an error occurred. If status is TRUE, this VI does not perform any operations.

IS2 code
code is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. See Appendix A, Error Codes, for a code description.

Data Acquisition Controller.VI
05/30/97 01:20 AM

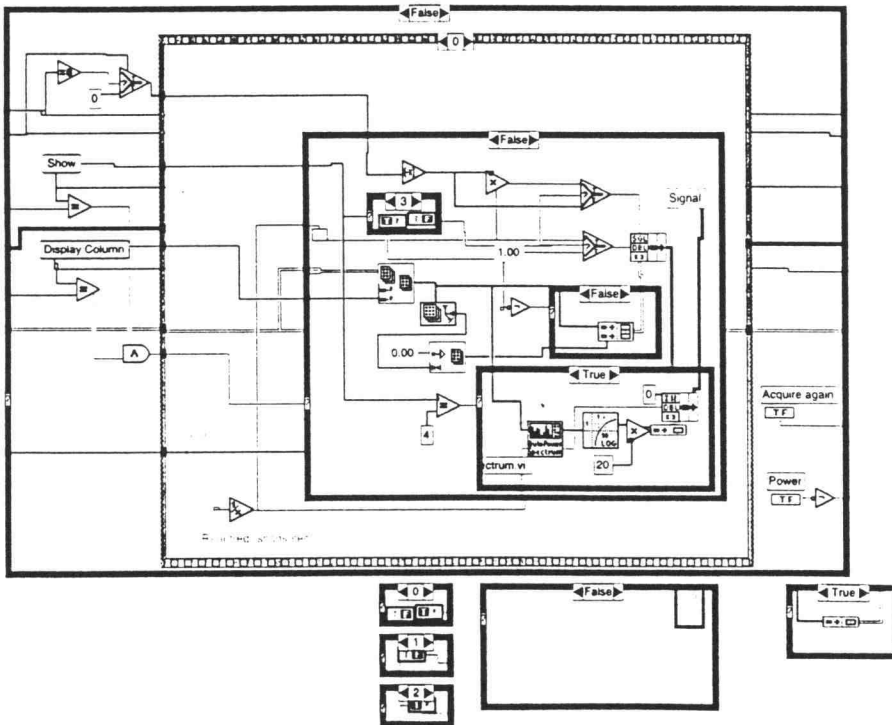
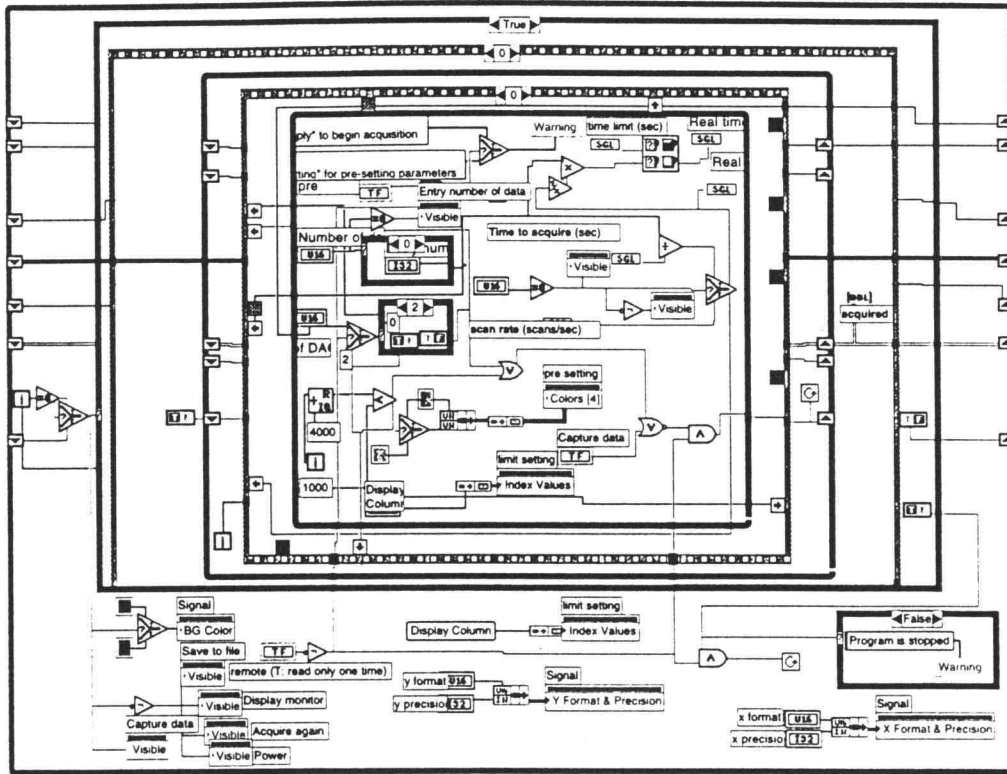
abc source
source shows where an error occurred. The source string is usually the name of the VI that produced the error.

PG1 Signal
Graph Color Code:
(1) Light green color: program is reading data.
(2) Dark blue color: acquired data is plotted on graph.

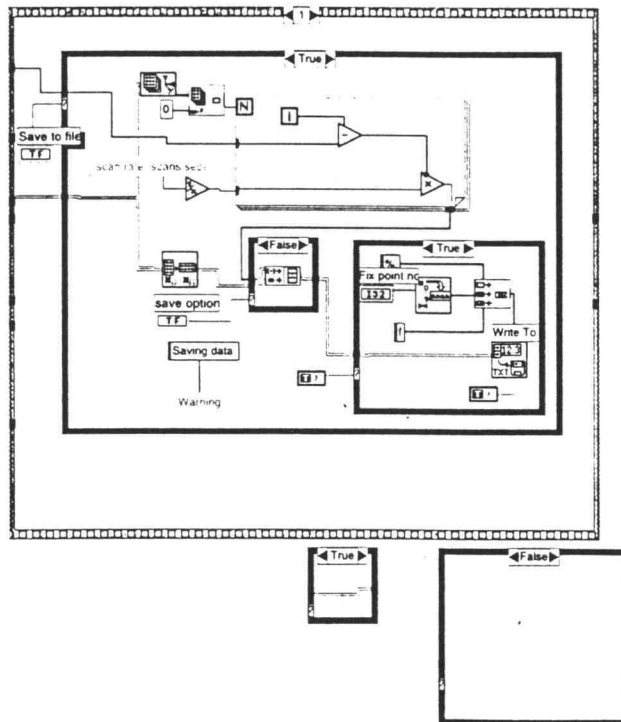
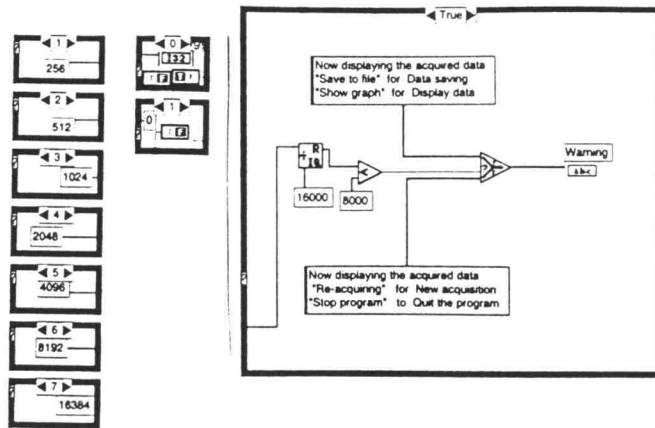
Users can zoom and re-scale graph using palette below.

SG1 Real time (sec)
Total time used in acquisition.

SG1 Real freq (scans/sec)
Sampling frequency used in acquisition.



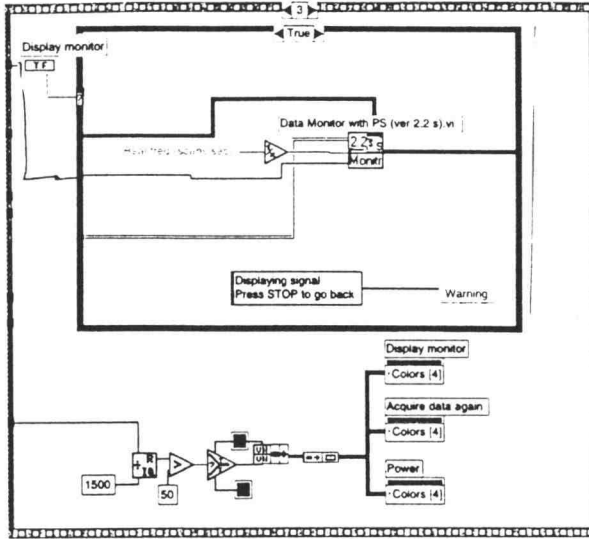
VIBRATION VISUALIZATION 6




VIBRATION VISUALIZATION 7

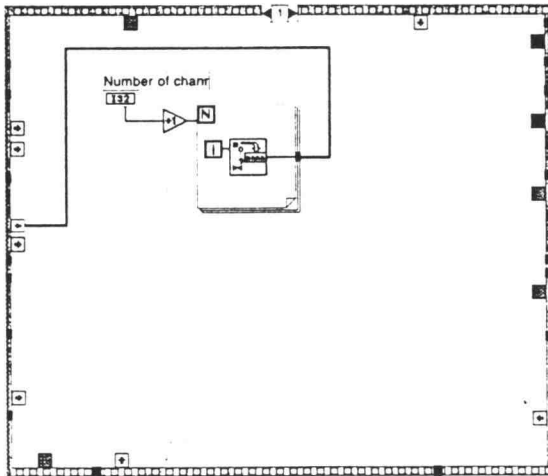
Data Acquisition Controller.VI
05/30/97 01:20 AM

Page 16 



Data Acquisition Controller.VI
05/30/97 01:20 AM

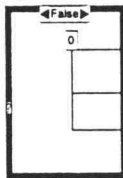
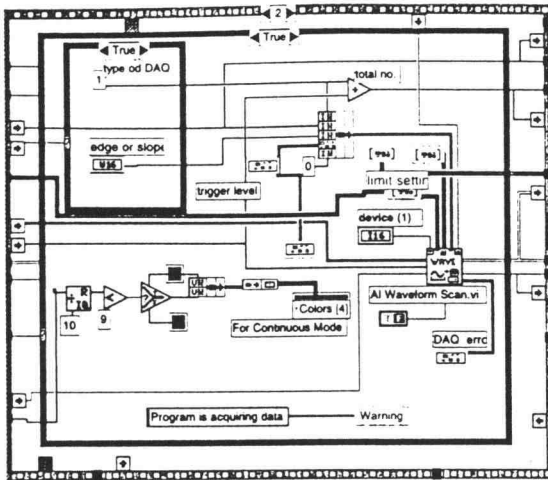
Page 18 



VIBRATION VISUALIZATION 8

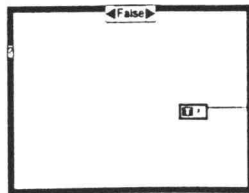
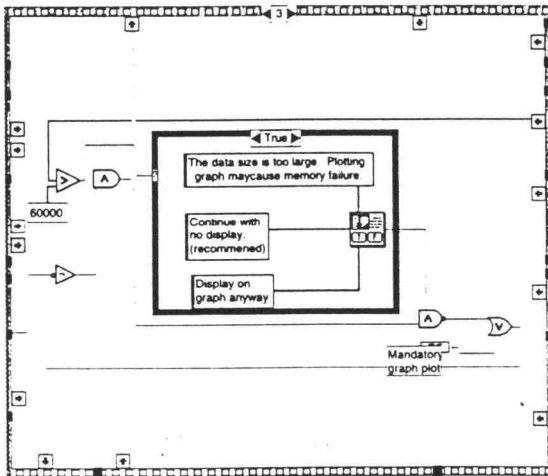
Data Acquisition Controller.VI
05/30/97 01:20 AM

Page 10 



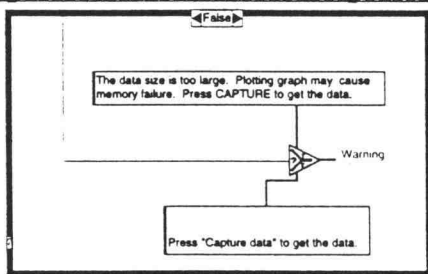
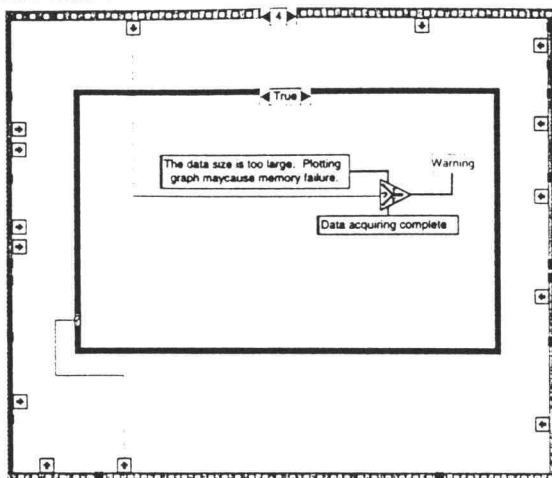
Page 21 

Data Acquisition Controller.VI
05/30/97 01:20 AM

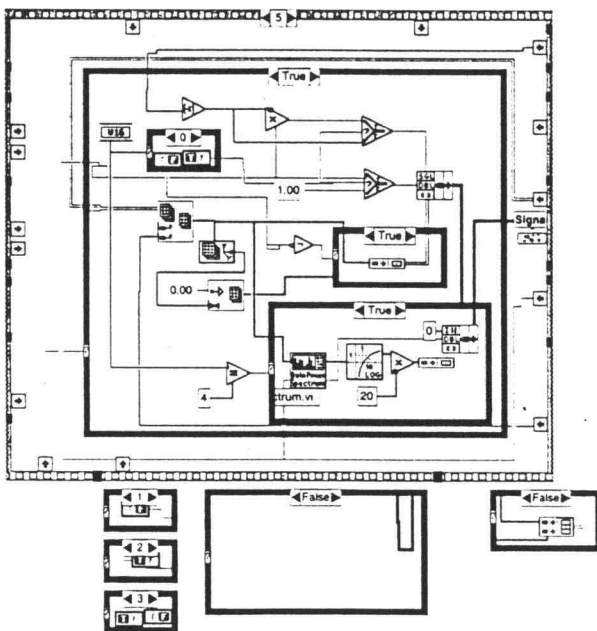


VIBRATION VISUALIZATION

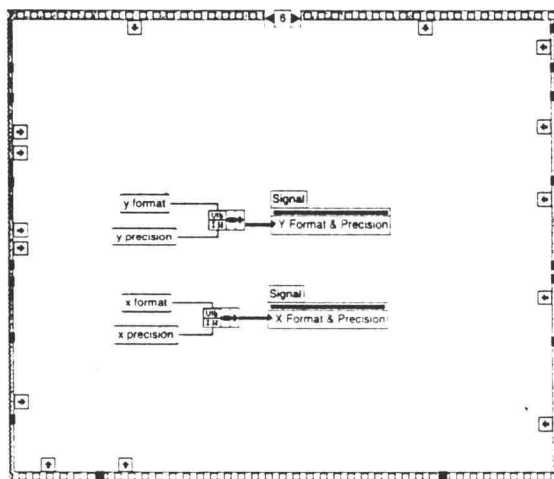
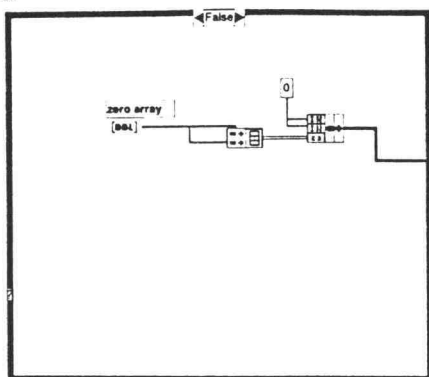
Data Acquisition Controller.VI
05/30/97 01:20 AM



Data Acquisition Controller.VI
05/30/97 01:20 AM



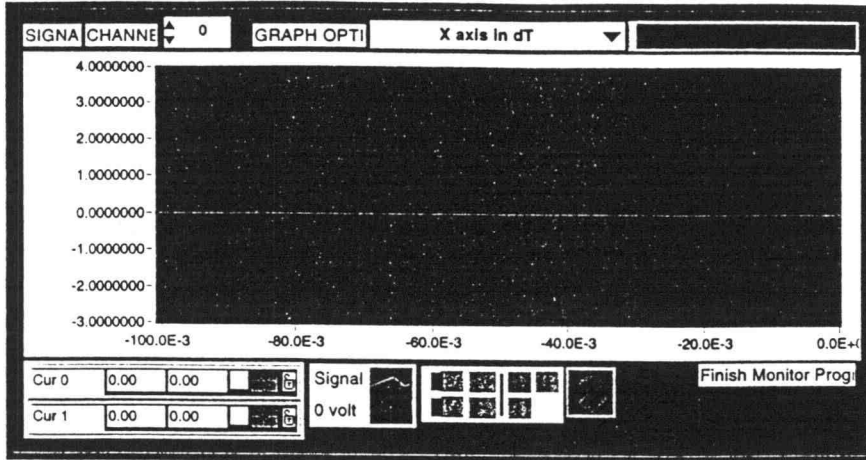
VIBRATION VISUALIZATION IO




VIBRATION VISUALIZATION 11

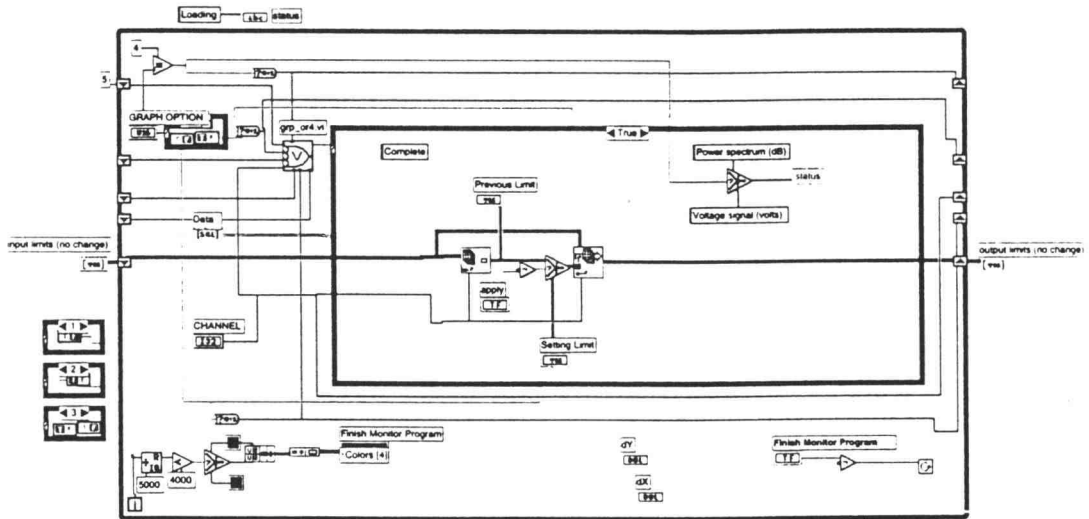
Data Monitor with PS (ver 2.2 s).vi
05/30/97 02:26 AM

Page 1 



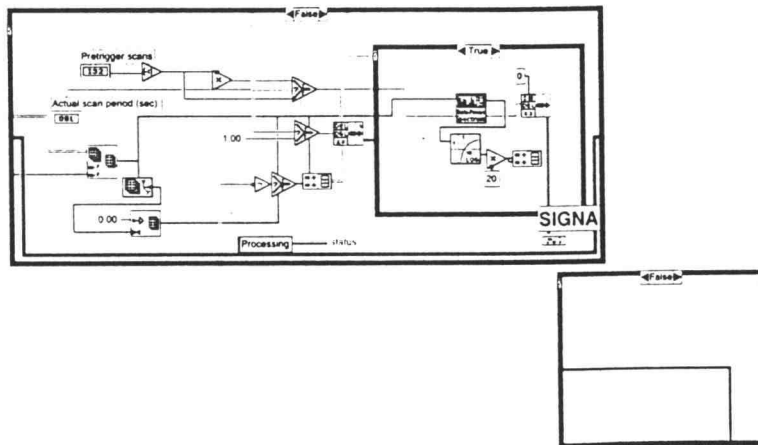
Page 2 

Data Monitor with PS (ver 2.2 s).vi
05/30/97 02:26 AM



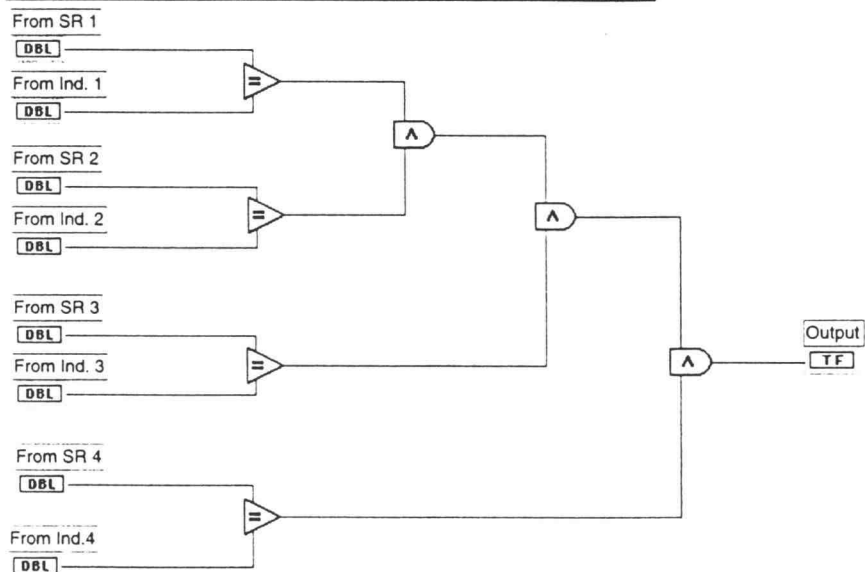
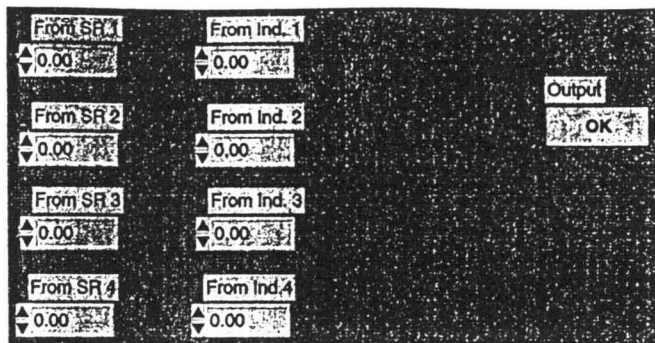
Page 3 

Data Monitor with PS (ver 2.2 s).vi
05/30/97 02:26 AM



VIBRATION VISUALIZATION 12

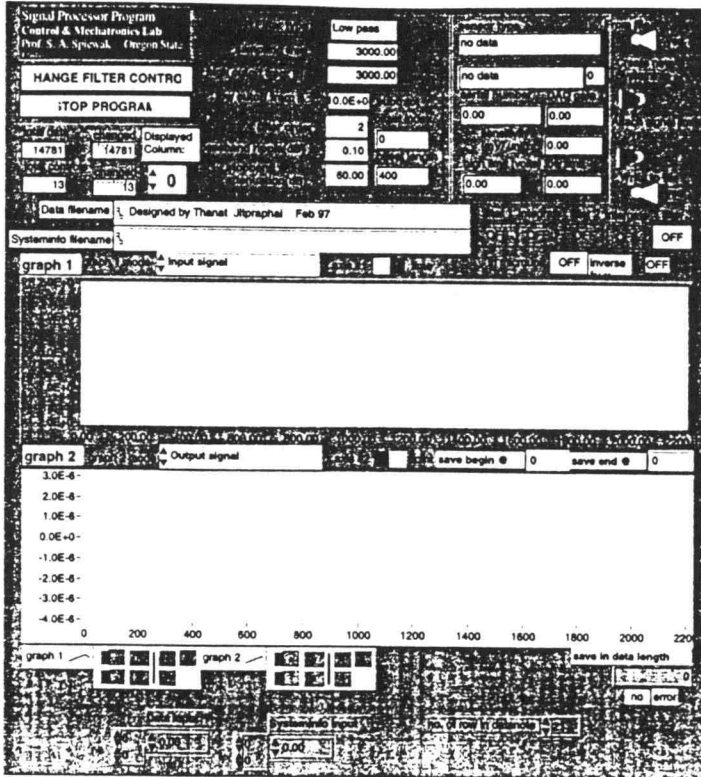
grp_or4.vi
05/30/97 01:24 AM



VIBRATION VISUALIZATION 13

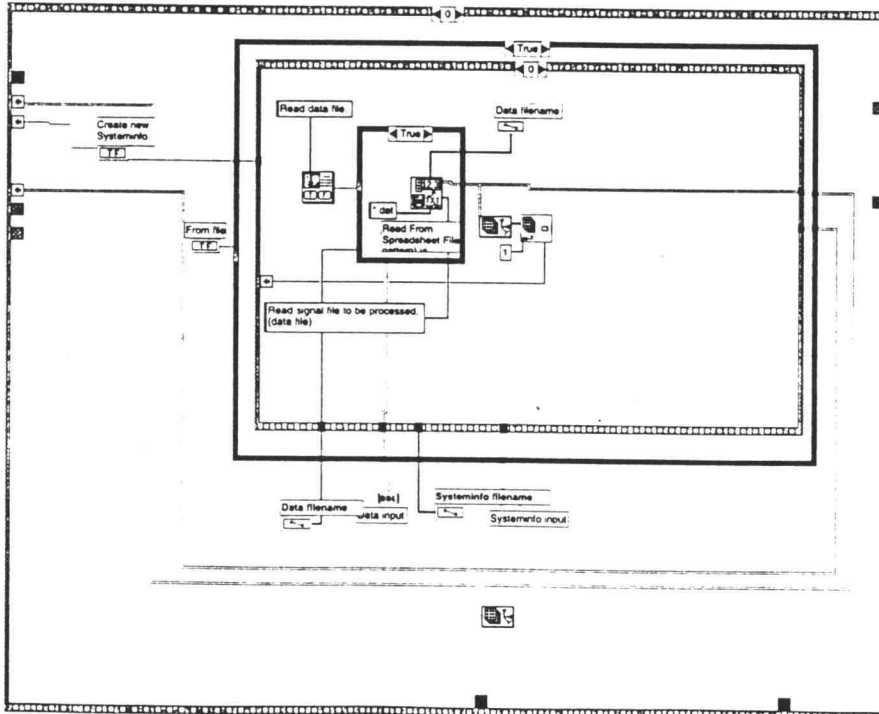
Signal Processor.VI
05/30/97 01:38 AM

Page 1 SP



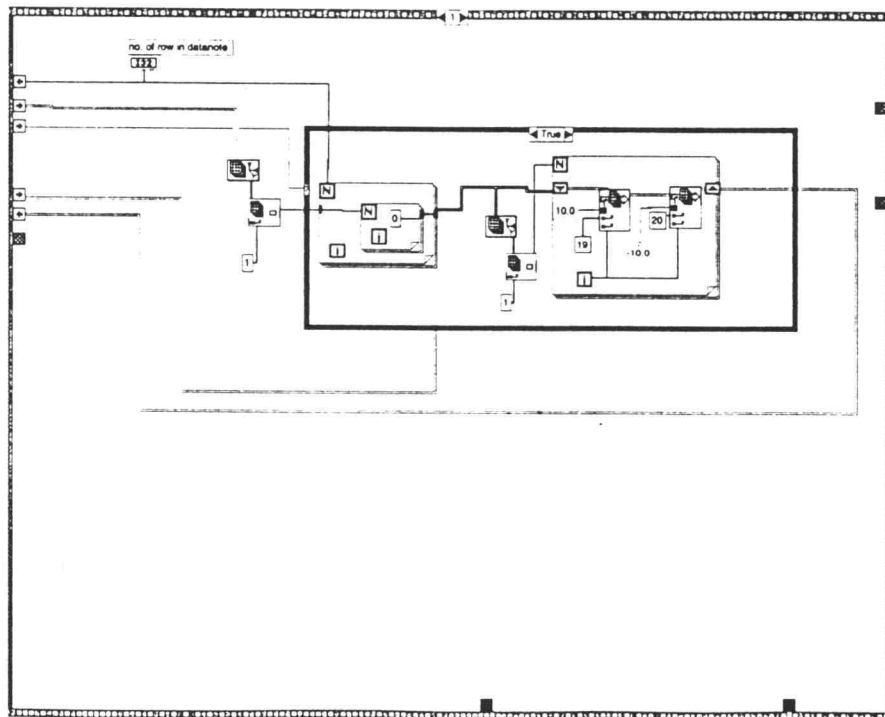
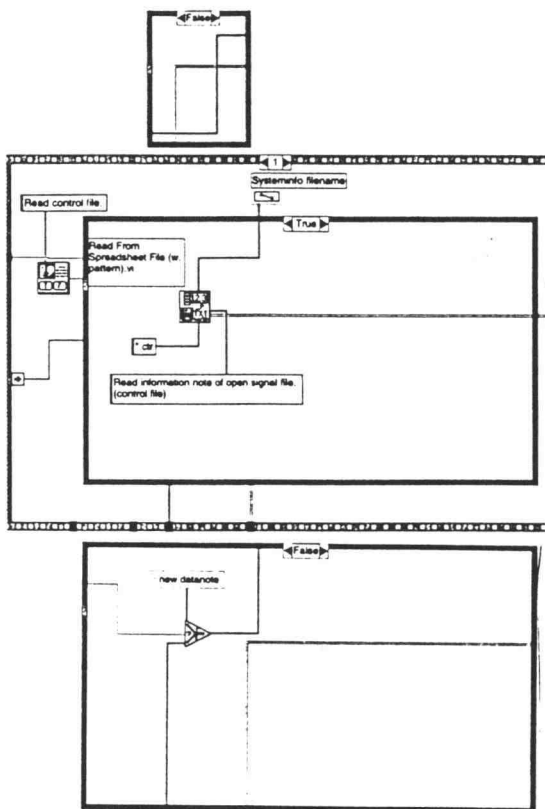
Signal Processor.VI
05/30/97 01:29 AM

Page 2 SP



- DATANOTE Description
[* or file]
- 0: sensor
 - 0: no data
 - 1: load cell
 - 2: proximity sensor
 - 3: voltage signal
 - 4: accelerometer
- 1: location
- 2: serial number
- 3: transducer sensitivity (mV/V/Hz)
- 4: double integral?
- 0: no integral
 - 1: double integral
 - 2: single integral
- 5: filter type
- 0: lowpass
 - 1: highpass
 - 2: bandpass
 - 3: band stop
- 6: filter order
- 7: sampling frequency
- 8: high cutoff frequency
- 9: low cutoff frequency
- 10: passband ripple (dB)
- 11: stopband attenuation (dB)
- 12: filter ON/OFF
(according to base2 number)
- 13: represent data in microvolt?
- 0: no convert
 - 1: convert to microvolt
- 14: inverse signal direction?
- 0: no inverse
 - 1: inverse
- 15: offset index
- 16: offset length
- 17: DAQ gain
- 18: Filter selection
- 0: Elastic filter
 - 1: Butterworth filter
- 19: DAQ high limits (volts)
- 20: DAQ low limits (volts)

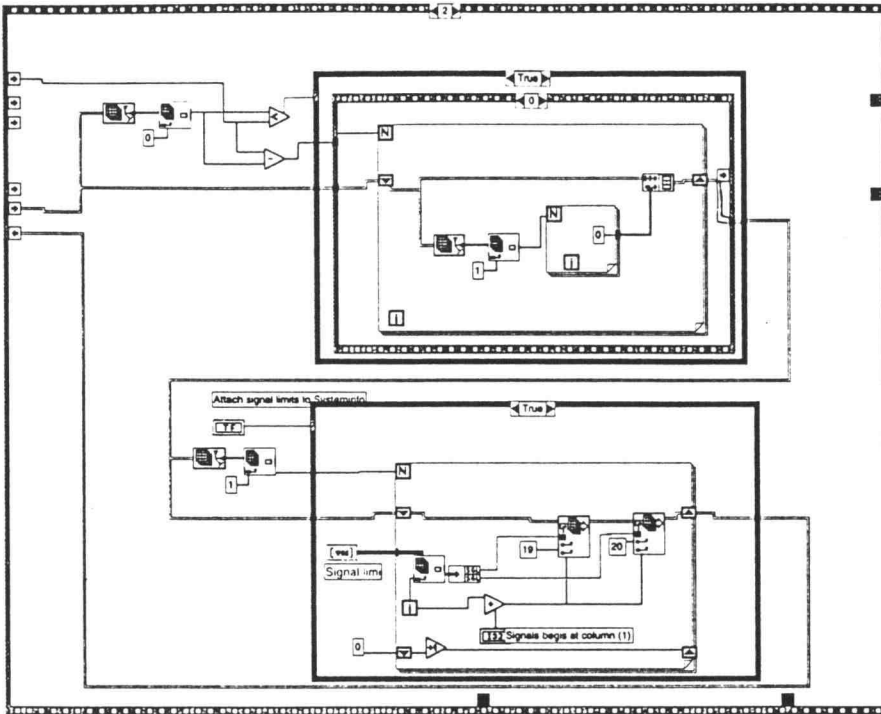
VIBRATION VISUALIZATION 14



VIBRATION VISUALIZATION 15

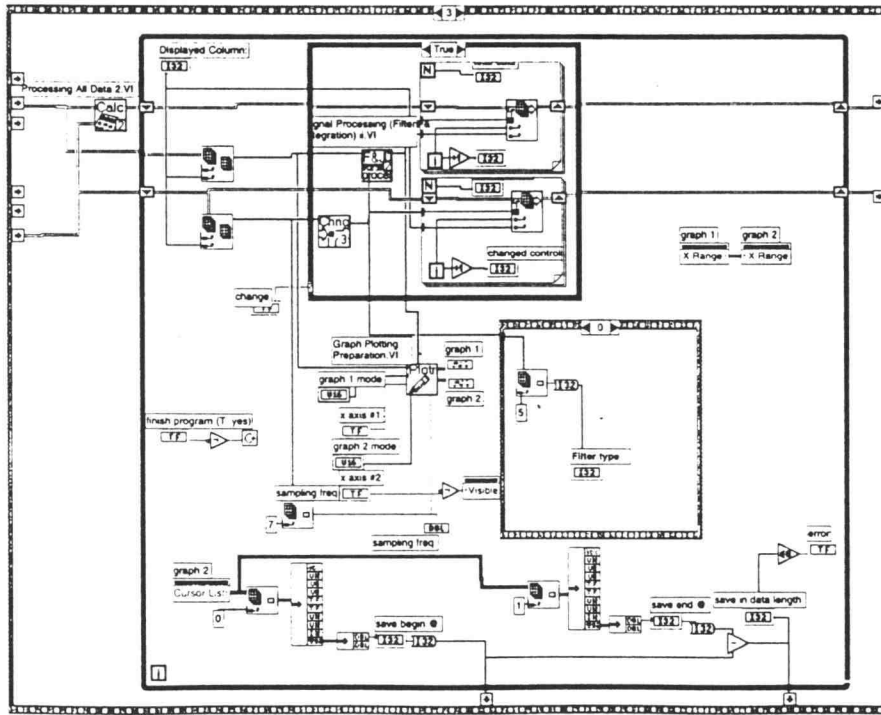
Signal Processor.VI
05/30/97 01:29 AM

Page 1 SP



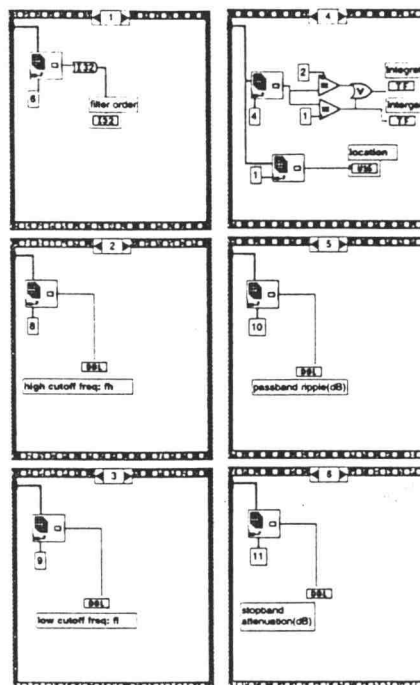
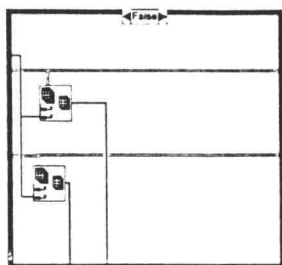
Signal Processor.VI
05/30/97 01:29 AM

Page 10 SP



VIBRATION VISUALIZATION 16

Signal Processor.VI
05/30/97 01:29 AM



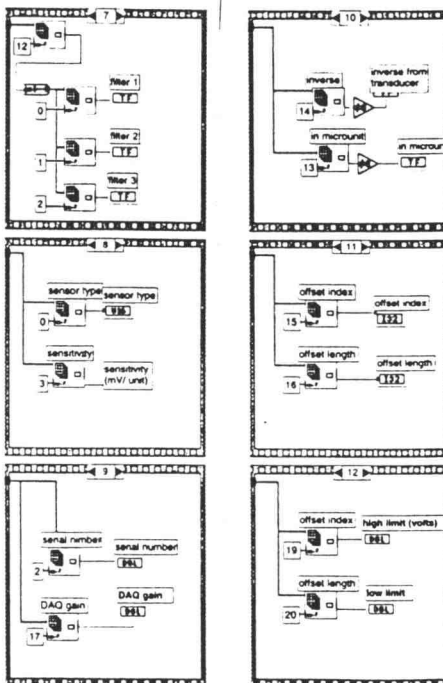
SP

Signal Processor.VI
05/30/97 01:29 AM

Page 14

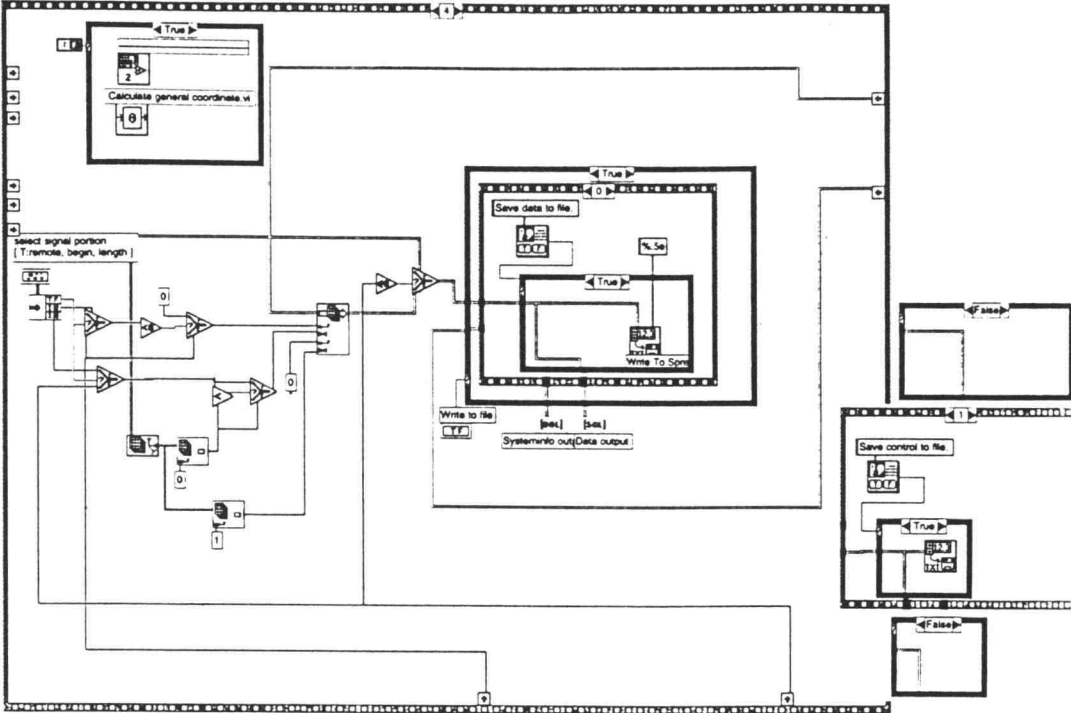
SP

VIBRATION VISUALIZATION 17

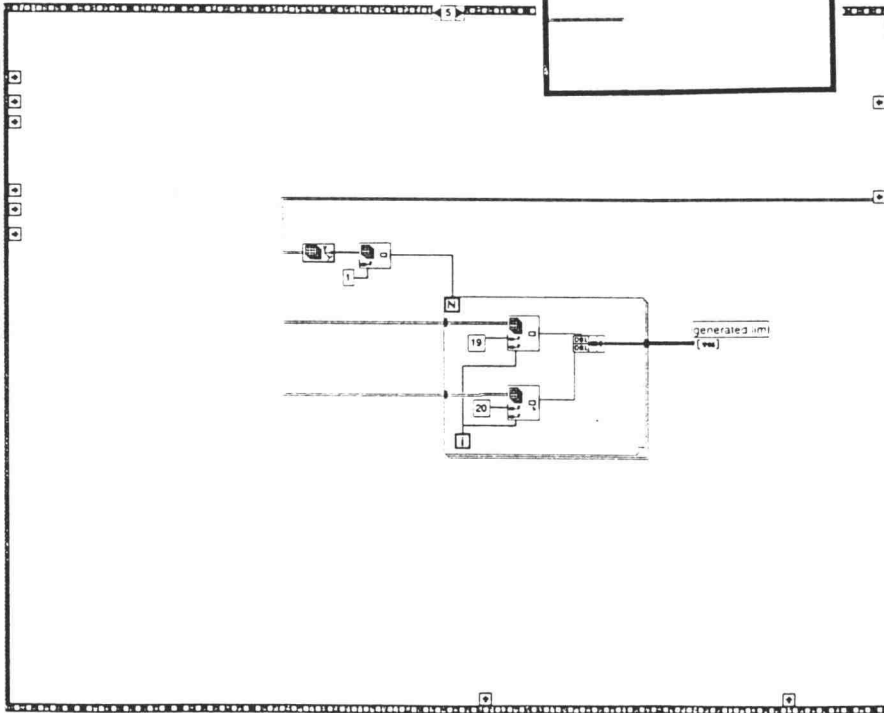


SP

Signal Processor.VI
05/30/97 01:29 AM



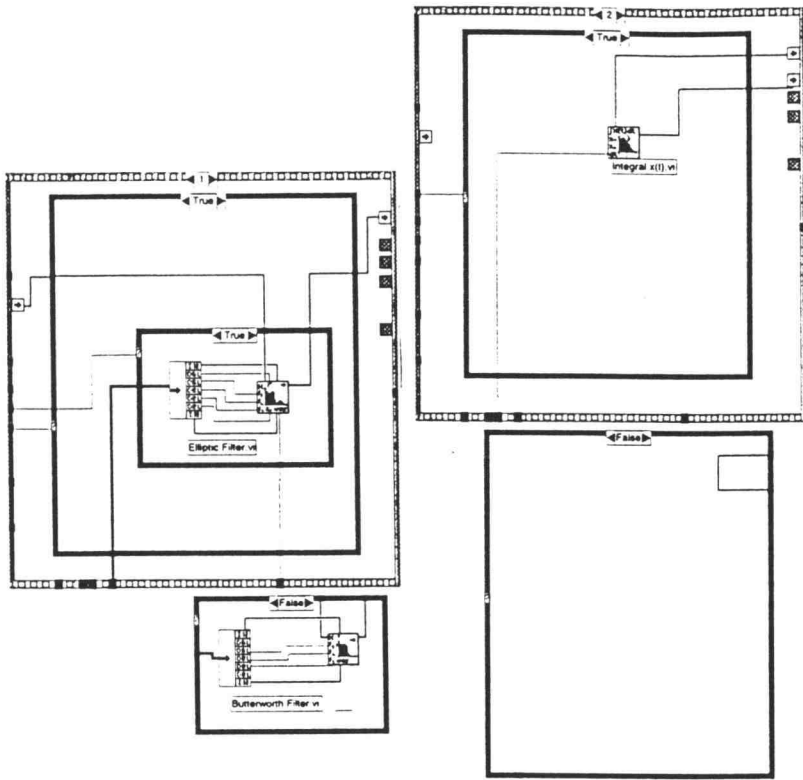
Signal Processor.VI
05/30/97 01:29 AM



VIBRATION VISUALIZATION IS

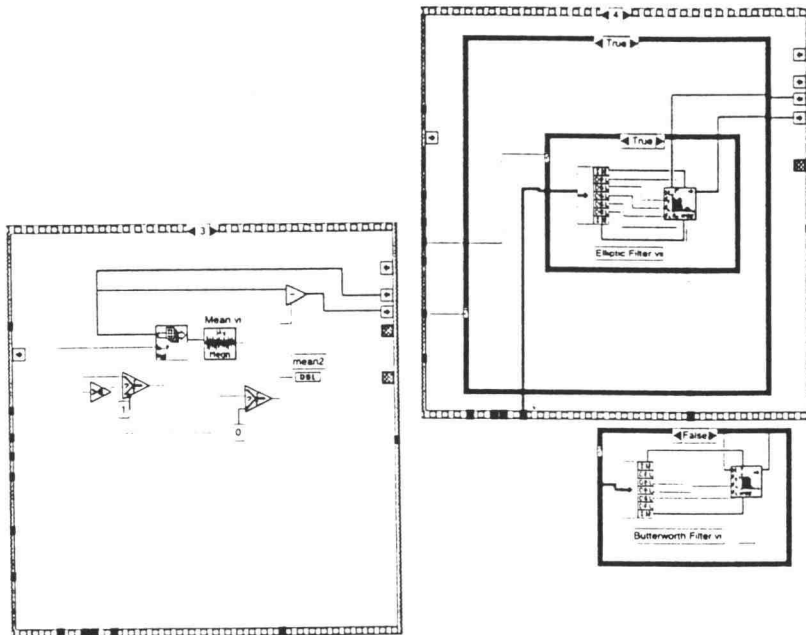
Signal Processing (Filters & Integration) ii.VI
05/30/97 01:44 AM

Page 3 



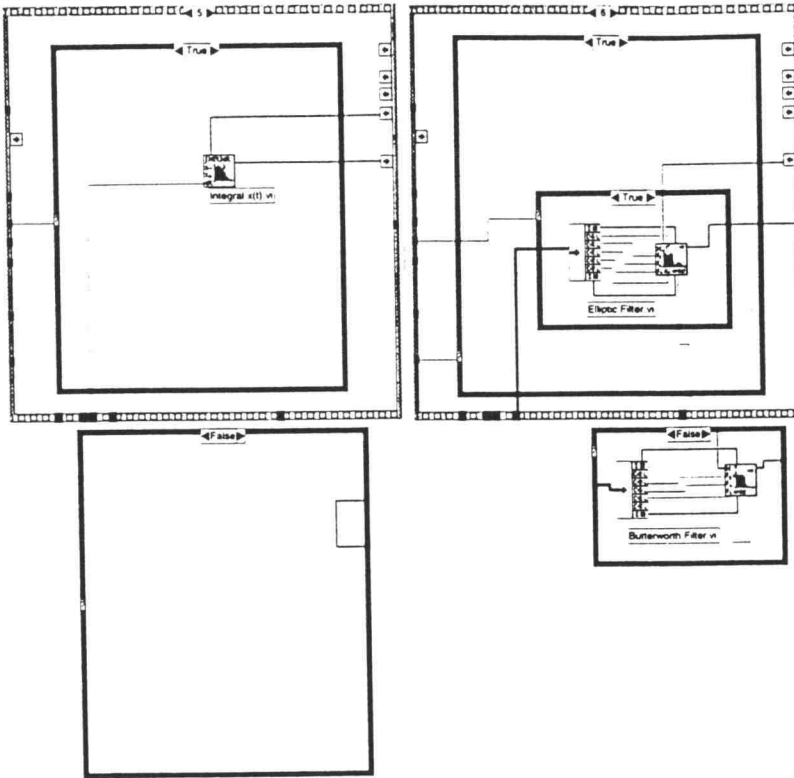
Signal Processing (Filters & Integration) ii.VI
05/30/97 01:44 AM

Page 7 



VIBRATION VISUALIZATION 20

Signal Processing (Filters & Integration) ii.VI
05/30/97 01:44 AM



Filter Controls

high cutoff freq: fh
3000.00
original high
0.00

low cutoff freq: fl
10.0000
original low
0.00

passband ripple(dB)
order: 2
original: 0

stopband attenuation(dB)
60.00
original: 0.00

selection
Elptic filter
original selection
Elptic filter
Filter type
High pass
original filter type
High pass

pressed1
changeable1
CHANGE

DAQ Parameters

high limit: 10.00
original high: 0.00
(volts)

low limit: -10.00
original low: 0.00
(volts)

pressed6
changeable6
OFF CHANGE

Subtract offset
offset index: 0
original: 0
offset length: 400
original: 400

pressed2
changeable2
OFF CHANGE

Transducer Parameters

sensor type: accelerometer
original sensor type

location: no data

serial no.: 0.00
original serial no.: 0.00

pressed4
changeable4
OFF CHANGE

sensitivity (mV/unit): 1000.00
original: 0.00

APPLY

Signal Processing Controls

1st filter: integral ON
2nd filter: integral ON
filter 3: ON

convert to micro unit: ON
inverse signal: ON

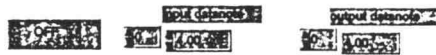
pressed5
changeable5
OFF CHANGE

Signal Information

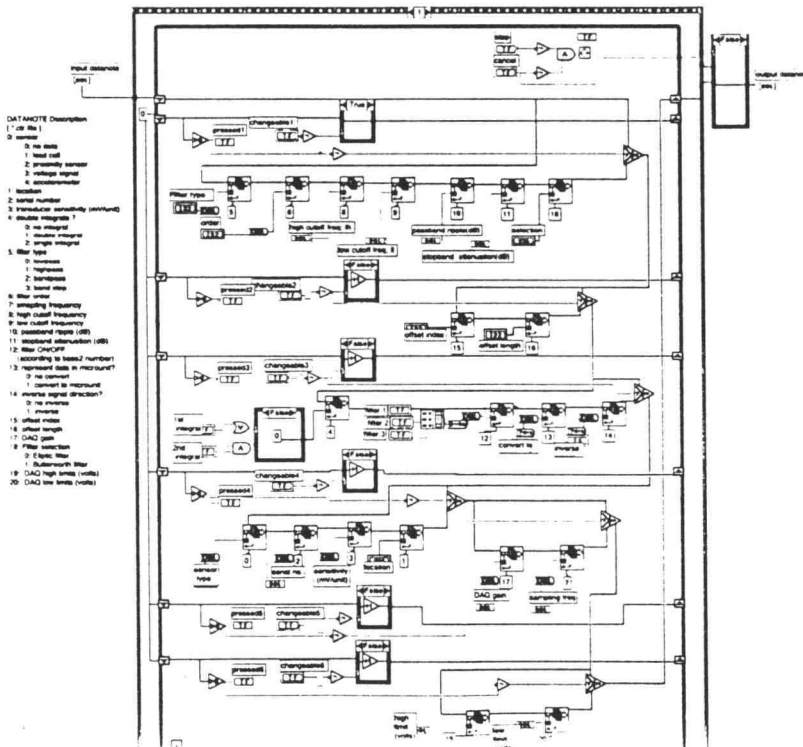
sampling freq: 10000.00
original sampling: 0.00

DAQ gain: 1.00
original gain: 0.00

- DATANOTE Description
[*.ctr file]
- 0: sensor
 - 1: location
 - 2: serial number
 - 3: transducer sensitivity (mV/unit)
 - 4: double integrate ?
 - 5: filter type
 - 6: filter order
 - 7: sampling frequency
 - 8: high cutoff frequency
 - 9: low cutoff frequency
 - 10: passband ripple (dB)
 - 11: stopband attenuation (dB)
 - 12: filter ON/OFF (according to base2 number)
 - 13: represent data in microcount?
 - 14: inverse signal direction?
 - 15: offset index
 - 16: offset length
 - 17: DAQ gain
 - 18: Filter selection
 - 19: DAQ high limits (volts)
 - 20: DAQ low limits (volts)

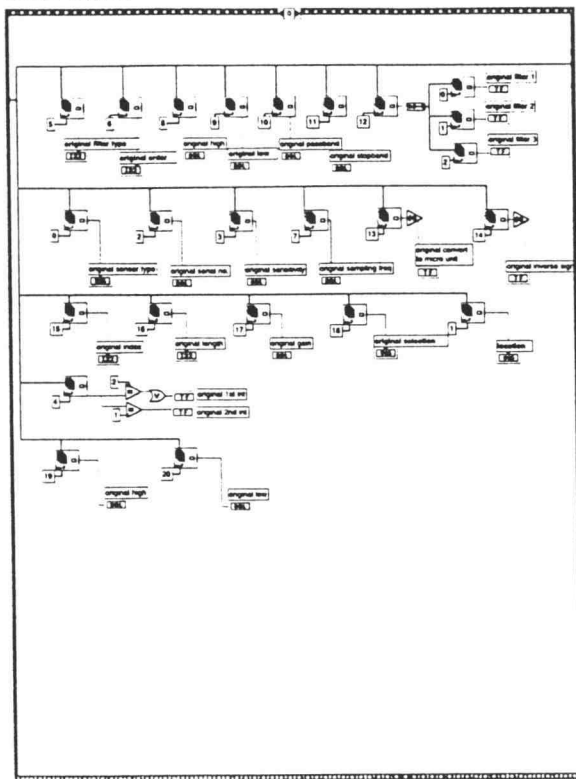


VIBRATION VISUALIZATION 2.2



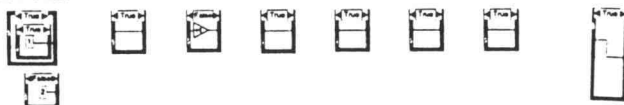
Datanote Changing 3.vi
05/30/97 01:32 AM

Page 3 Chng
3

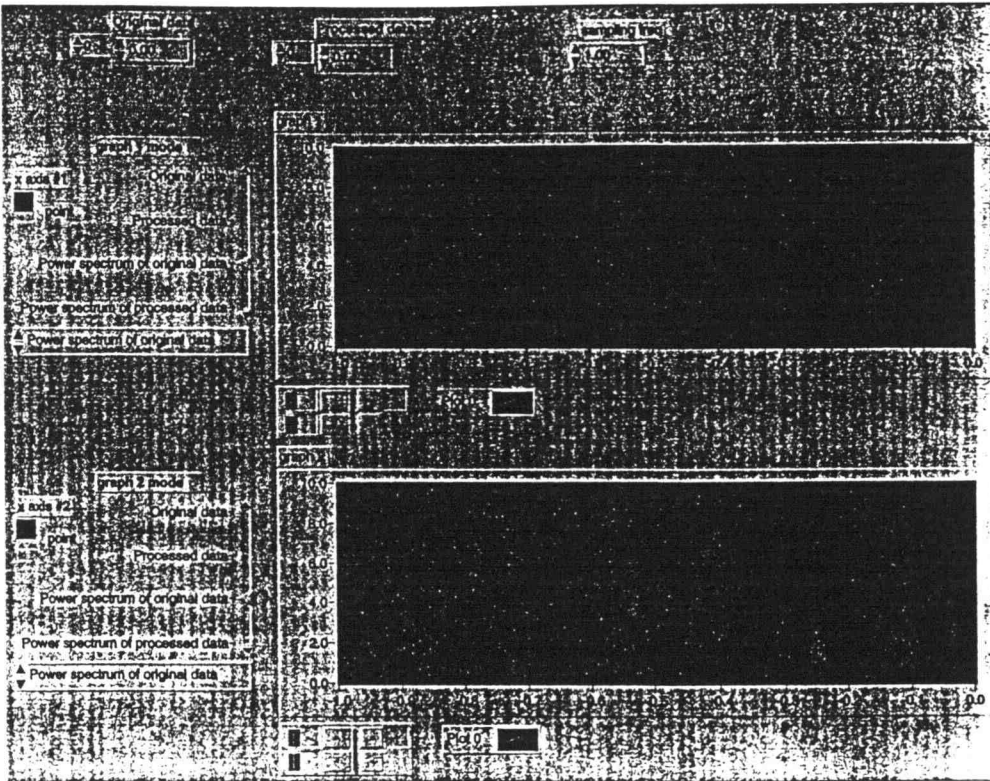


Datanote Changing 3.vi
05/30/97 01:32 AM

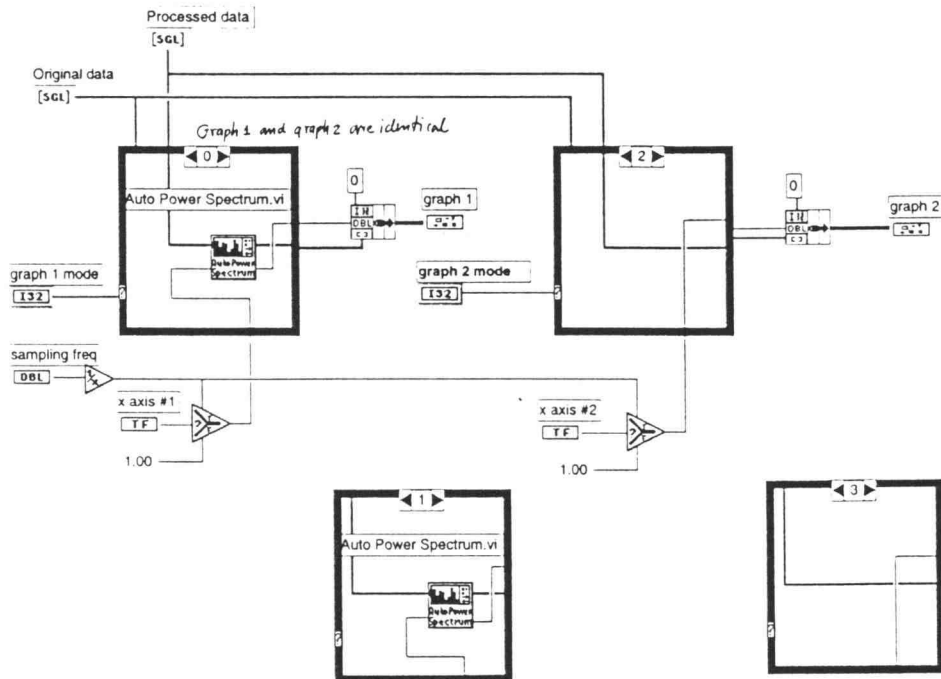
Page 4 Chng
3

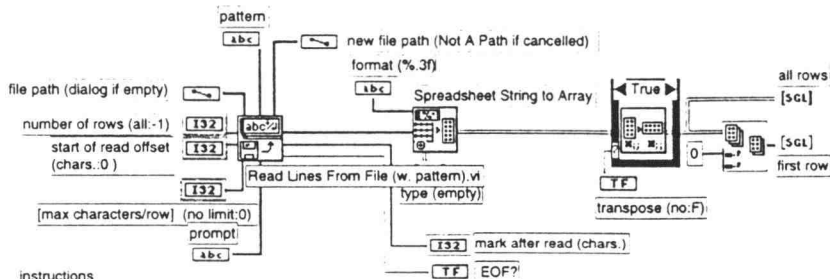
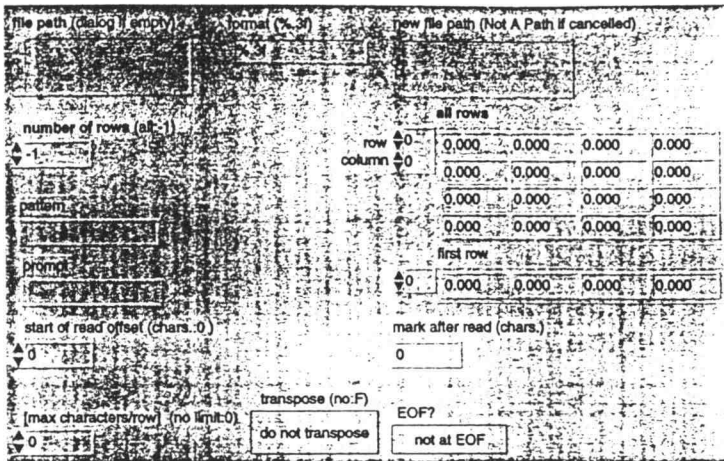


VIBRATION VISUALIZATION 23



VIBRATION VISUALIZATION 2-4





instructions

Spreadsheet String to Array requires that columns are separated with tab characters and rows are terminated with EOF characters. If your spreadsheet string uses different separators or terminators, use the Search String and Replace VI from the examples/general/strings.lib library (or something equivalent) at the output of Read Lines From File to modify the string.

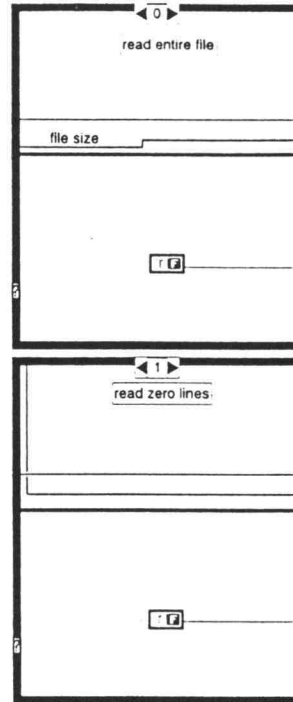
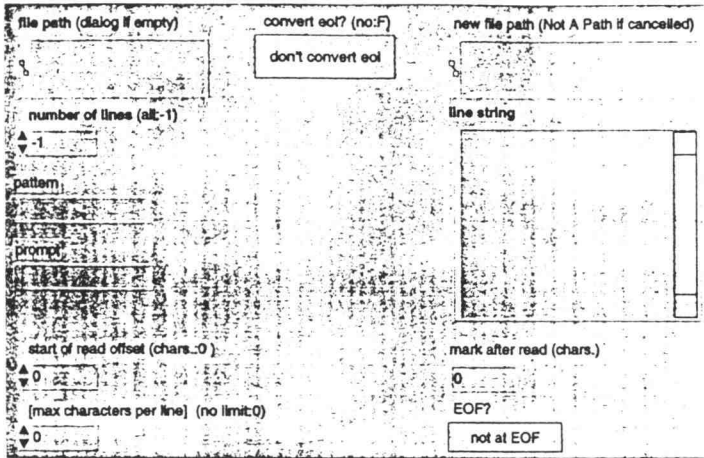
Also, Spreadsheet String to Array produces a square array whose number the columns equals the maximum number of characters of the rows that are read. Lines of text are converted to rows of zeros, although numbers within the text are converted to zeros if properly separated.

You can modify a copy of this VI to return the file contents into arrays of strings by changing all rows, first row, and type (empty) arrays to string arrays and by setting the format to %s.



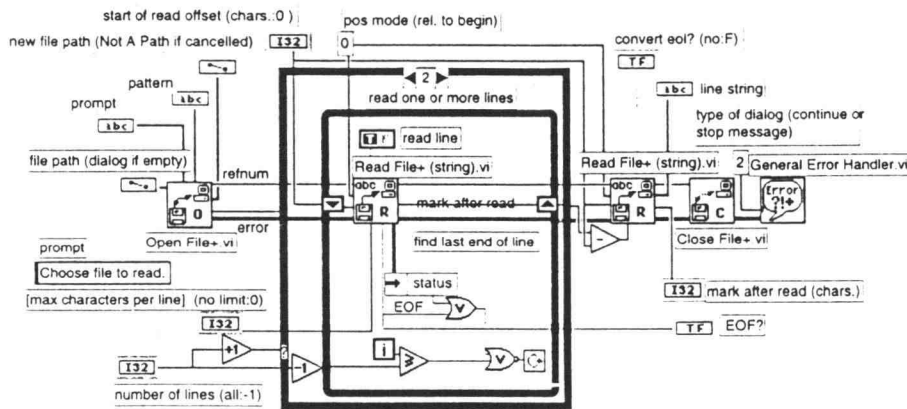
Read Lines From File (w. pattern).vi
05/30/97 01:46 AM

Page 1



Read Lines From File (w. pattern).vi
05/30/97 01:46 AM

Page 2



The read subVI inside the While Loop searches for the last end-of-line, but does not return any data. Once that location is found, the read subVI outside the loop reads the entire string. This method is faster for a large number of lines, because it does not require the use of a string concatenation function in the loop and its associated memory shuffling, but this method is slower for a few lines.

VISUALIZATION 2.6

Data filename D:\THOMAS\THANAT.TMP\DATA\CENTZA.VIZ
Systeminfo filename "D:\THOMAS\THANAT.TMP\DATA\3FC20-8C.CTR

Channel assignment

| | |
|----------------------------------|----|
| Corner0 in X direction @ channel | 2 |
| Corner0 in Y direction @ channel | 3 |
| Corner0 in Z direction @ channel | 4 |
| Corner1 in X direction @ channel | 5 |
| Corner1 in Z direction @ channel | 6 |
| Corner2 in Y direction @ channel | 7 |
| Corner2 in Z direction @ channel | 8 |
| Corner3 in X direction @ channel | 9 |
| Corner3 in Y direction @ channel | 10 |

From file YES Write to file YES

Systeminfo input
 0.00 0.00
 0.00

Systeminfo output
 1.00

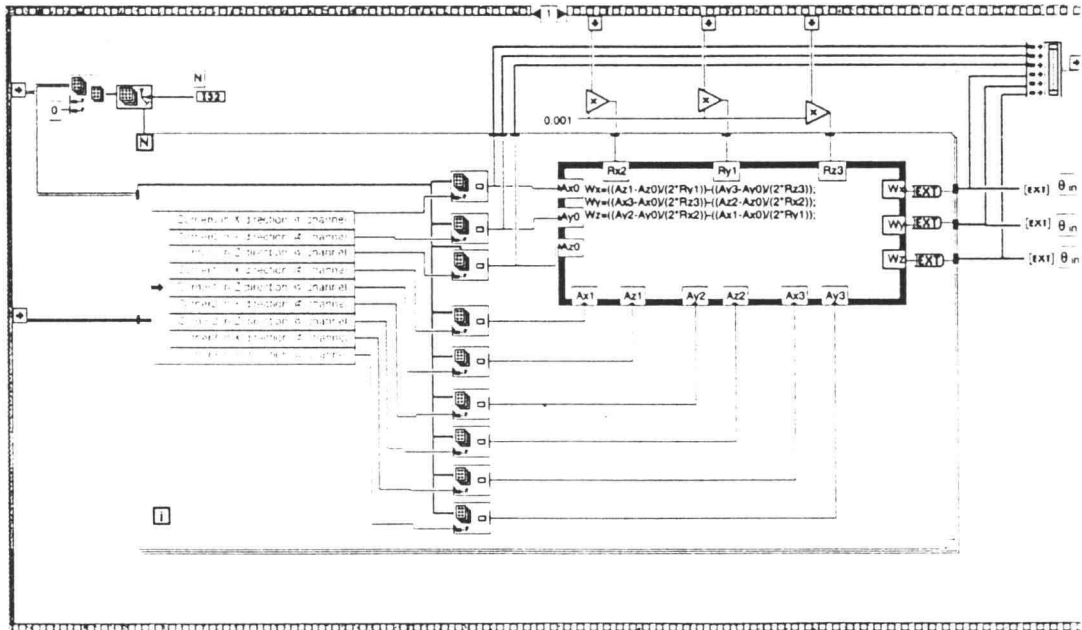
Data Input

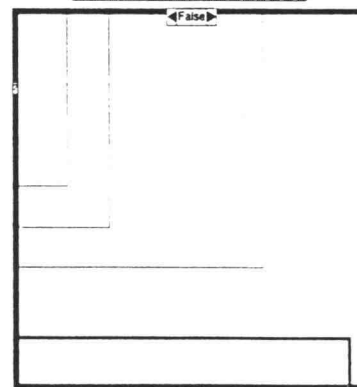
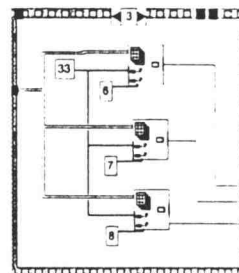
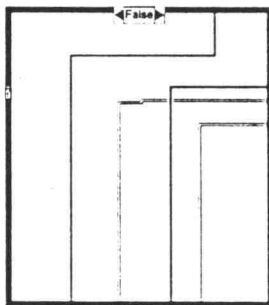
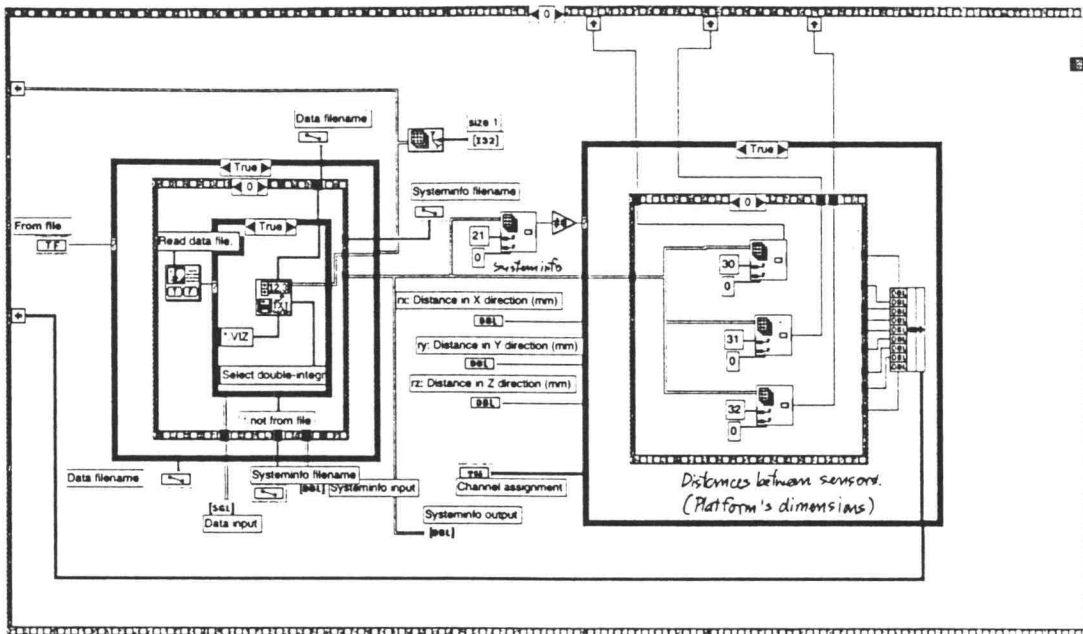
| | | | | | | | | | | | |
|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|
| <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 |
| <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 |
| <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 |
| <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 |
| <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 |
| <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 |
| <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 |
| <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 |
| <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 | <input type="checkbox"/> 0.00 |

Generalized coordinate output

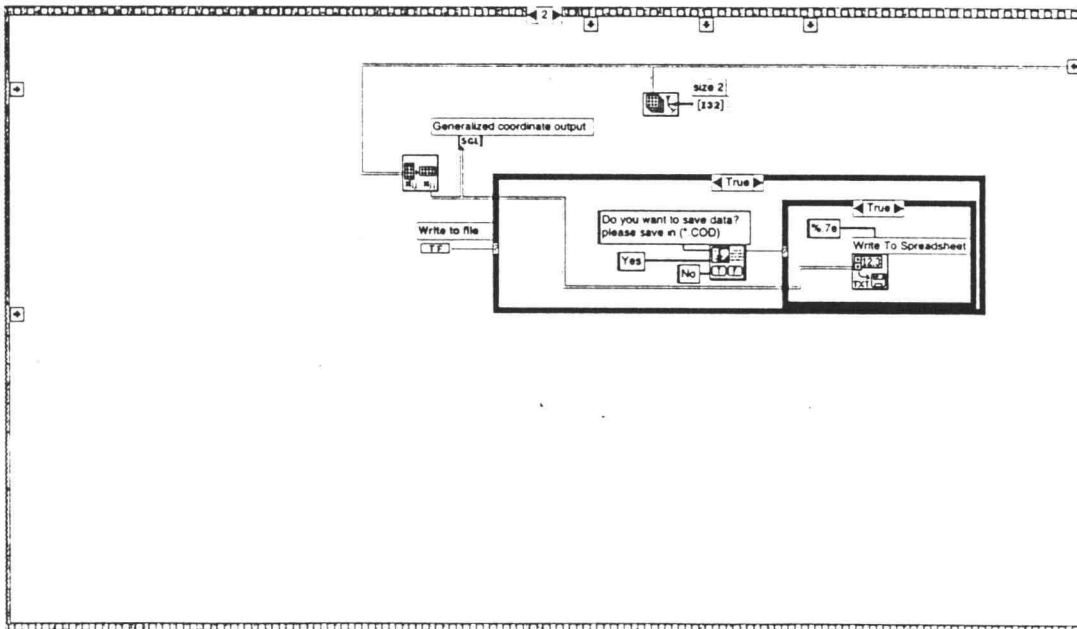
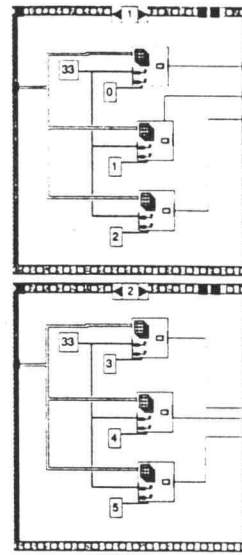
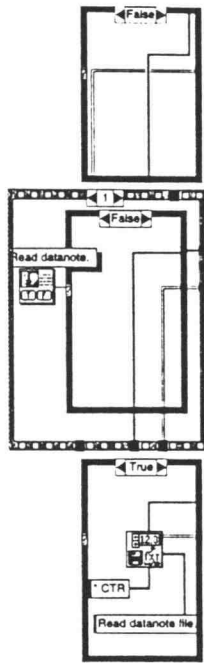
| | | | | | | |
|----------------------------|-------------|------------|------------|----------|----------|------------|
| <input type="checkbox"/> 0 | 13.27E-11 | 8.57E-12 | -147.85E-9 | 1.87E-9 | -2.21E-9 | -184.81E-9 |
| <input type="checkbox"/> 0 | 15.73E-11 | 8.59E-12 | -339.70E-9 | 3.94E-9 | -4.42E-9 | -466.32E-9 |
| | 4.45E-12 | -594.64E-9 | -533.19E-9 | 5.91E-9 | -6.08E-9 | -837.84E-9 |
| | -14.31E-11 | -10.35E-11 | -730.25E-9 | 7.97E-9 | -7.48E-9 | -1.31E-9 |
| | -41.96E-11 | -15.16E-11 | -934.95E-9 | 10.25E-9 | -8.49E-9 | -1.88E-9 |
| | -78.78E-11 | -19.18E-11 | -1.15E-9 | 12.55E-9 | -9.01E-9 | -2.51E-9 |
| | -121.67E-11 | -27.01E-11 | -1.36E-9 | 14.66E-9 | -9.05E-9 | -3.20E-9 |

VIBRATION VISUALIZATION 27



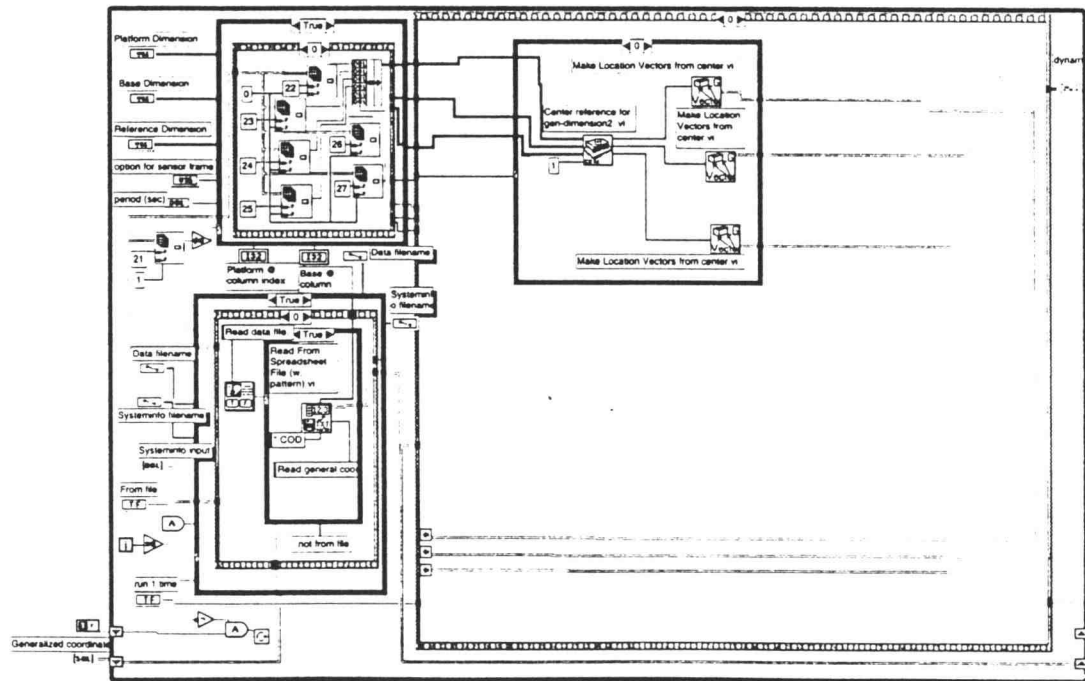
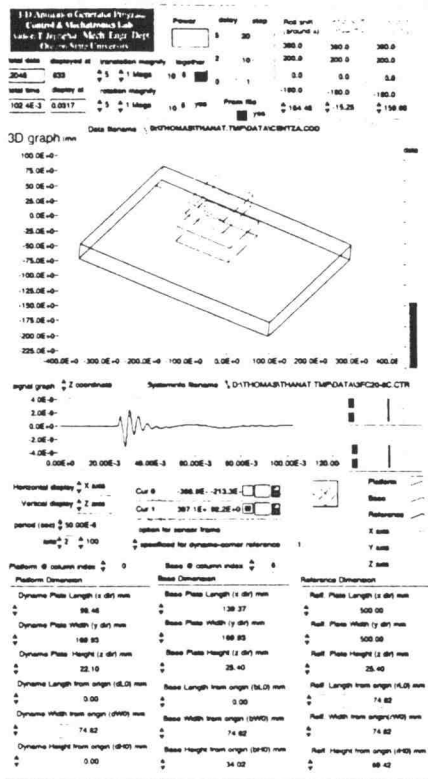


VIBRATION VISUALIZATION 2.8



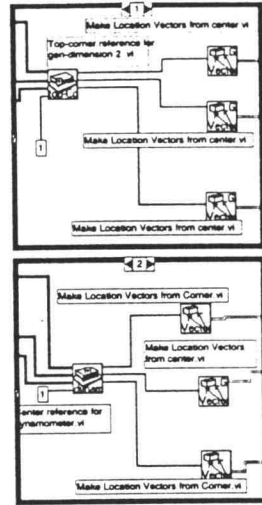
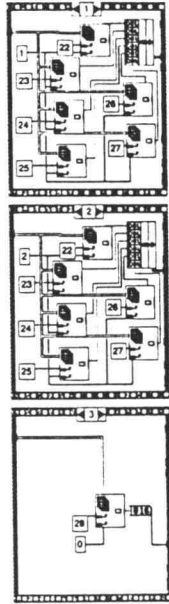
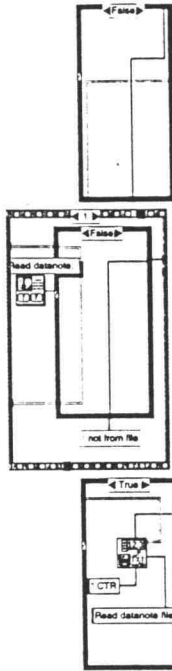
VIBRATION VISUALIZATION 29



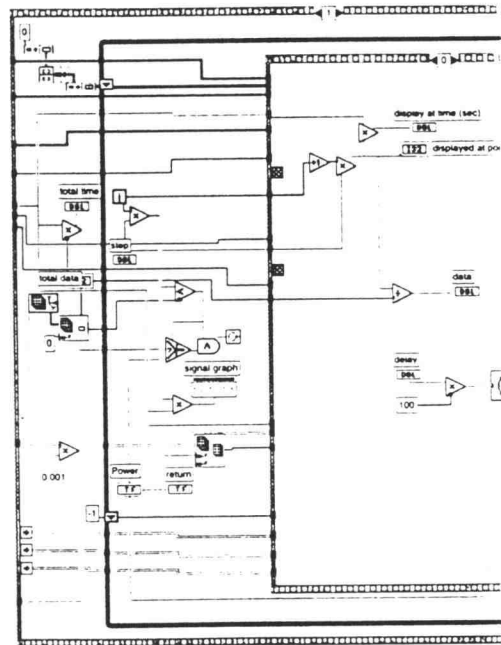
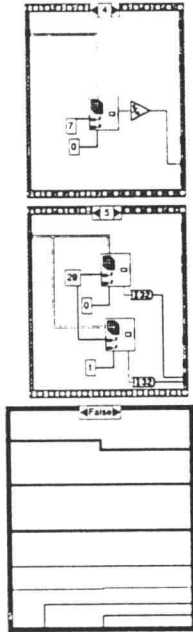
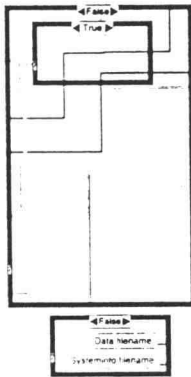


VIBRATION VISUALIZATION 3D

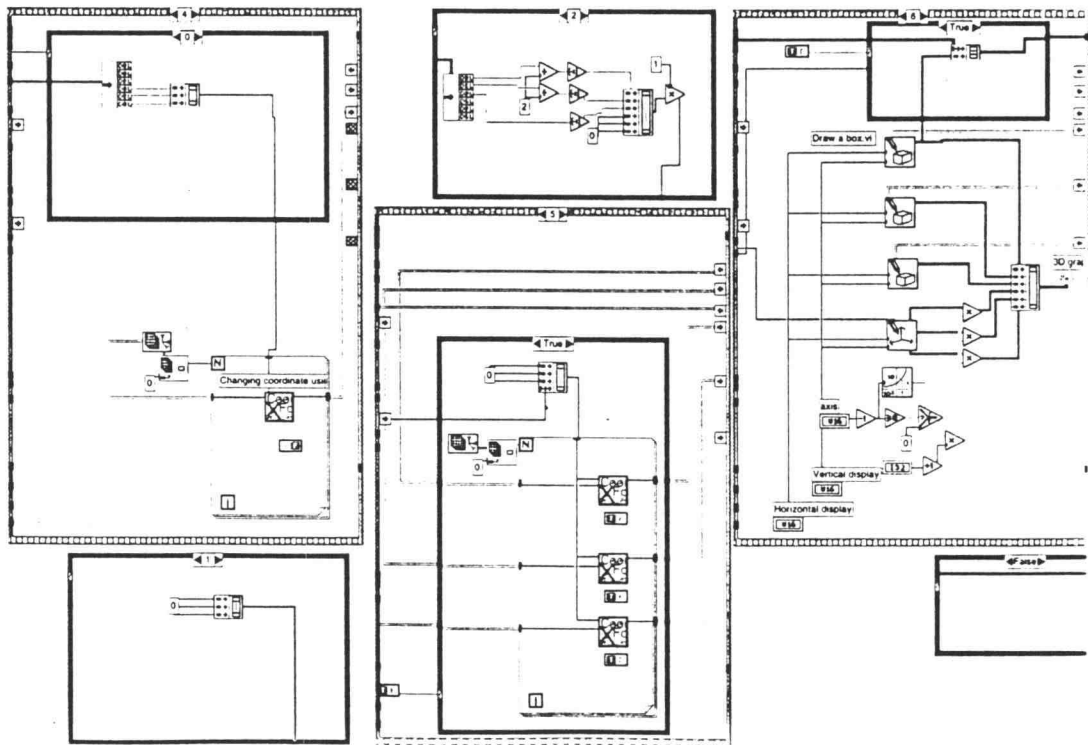
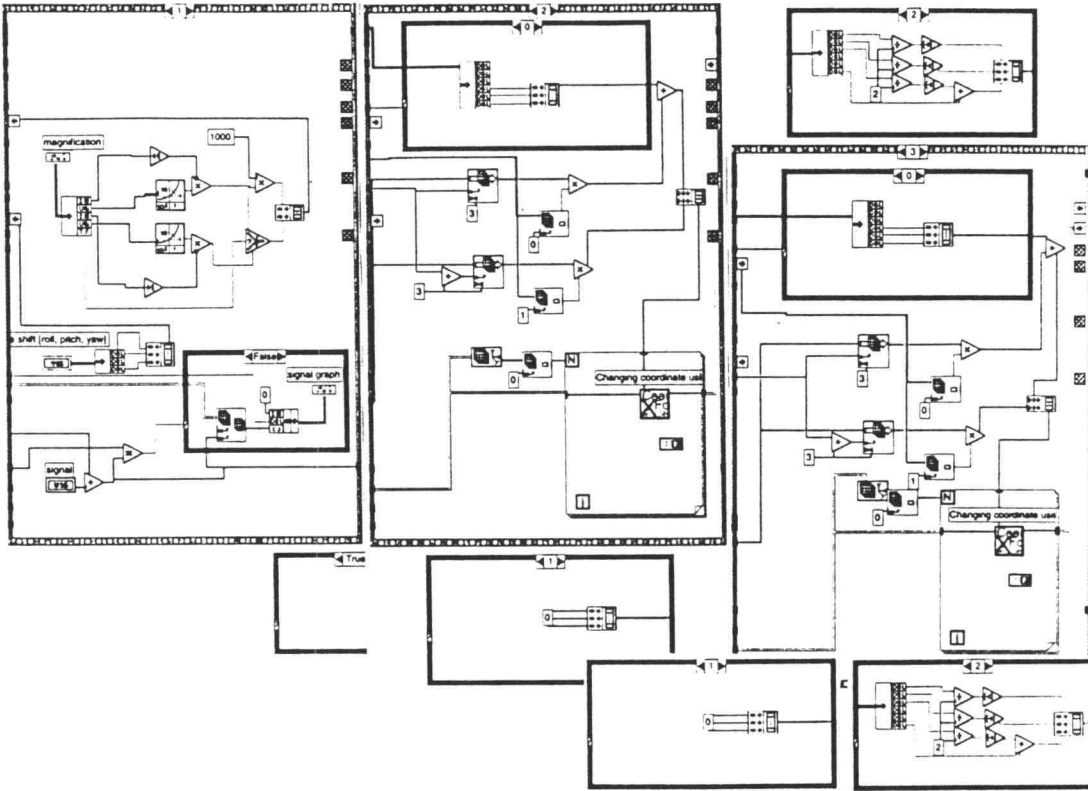
3-D Animation Generator.VI
05/30/97 01:14 AM



3-D Animation Generator.VI
05/30/97 01:14 AM



VIBRATION VISUALIZATION 31



VIBRATION VISUALIZATION 32

Changing coordinate using transformation
05/30/97 01:33 AM

4 x 4 Identity matrix

| | | | |
|------|------|------|------|
| 1.00 | 0.00 | 0.00 | 0.00 |
| 0.00 | 1.00 | 0.00 | 0.00 |
| 0.00 | 0.00 | 1.00 | 0.00 |
| 0.00 | 0.00 | 0.00 | 1.00 |

Translation & rotation

| | | | | | |
|---------|---------|---------|---------|---------|---------|
| 0.00E+0 | 0.00E+0 | 0.00E+0 | 0.00E+0 | 0.00E+0 | 0.00E+0 |
|---------|---------|---------|---------|---------|---------|

Original coordinate

| | | | | | |
|---------|---------|---------|---------|---------|---------|
| 0.00E+0 | 0.00E+0 | 0.00E+0 | 0.00E+0 | 0.00E+0 | 0.00E+0 |
|---------|---------|---------|---------|---------|---------|

New Coordinate Array @ specified location

| | | | | | |
|---------|---------|---------|---------|---------|---------|
| 1.00E+0 | 1.00E+0 | 1.00E+0 | 0.00E+0 | 0.00E+0 | 0.00E+0 |
|---------|---------|---------|---------|---------|---------|

shift angle (degree)

| | | |
|------|------|------|
| 0.00 | 0.00 | 0.00 |
|------|------|------|

shift position

| | | |
|---------|---------|---------|
| 0.00E+0 | 0.00E+0 | 0.00E+0 |
|---------|---------|---------|

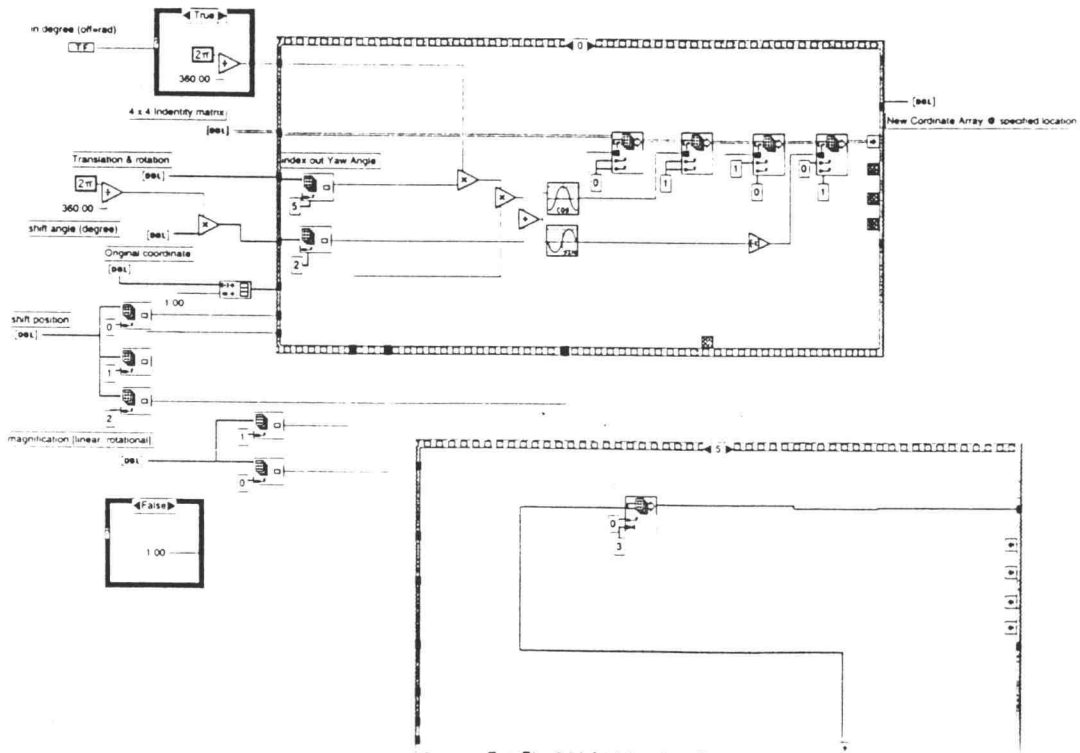
magnification [linear, rotational]

| | | |
|------|------|------|
| 1.00 | 1.00 | 1.00 |
|------|------|------|

vector size

| |
|------|
| 2048 |
|------|

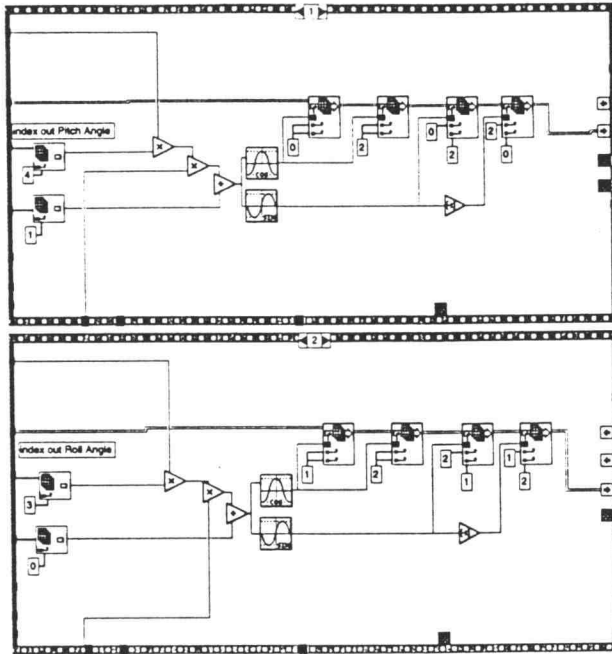
Changing coordinate using transformation
05/30/97 01:33 AM



VIBRATION VISUALIZATION 33

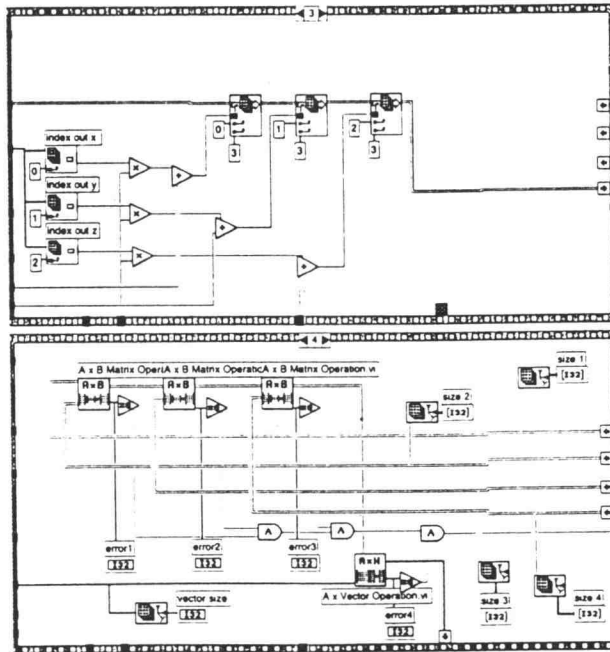
Changing coordinate using transformation
05/30/97 01:33 AM

Page 3



Page 4

Changing coordinate using transformation
05/30/97 01:33 AM



VIBRATION VISUALIZATION 34

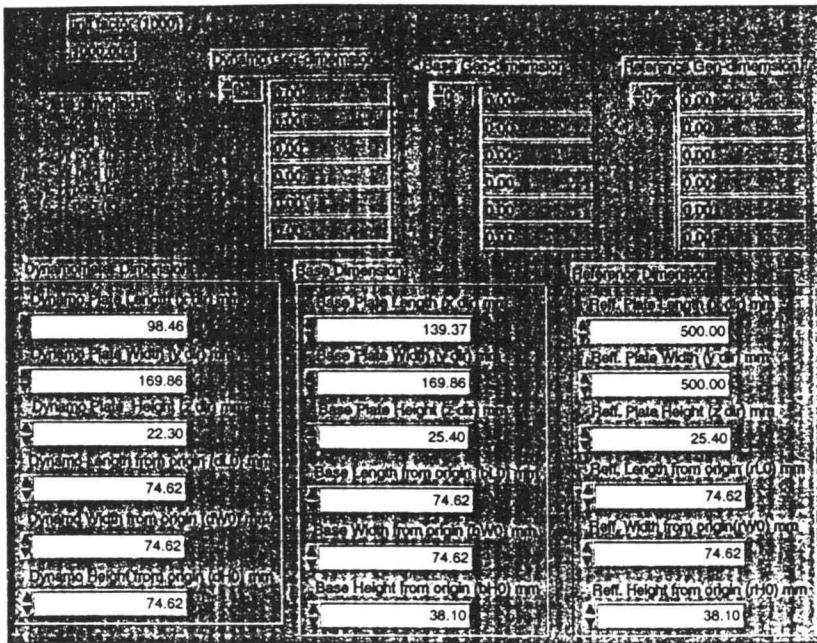
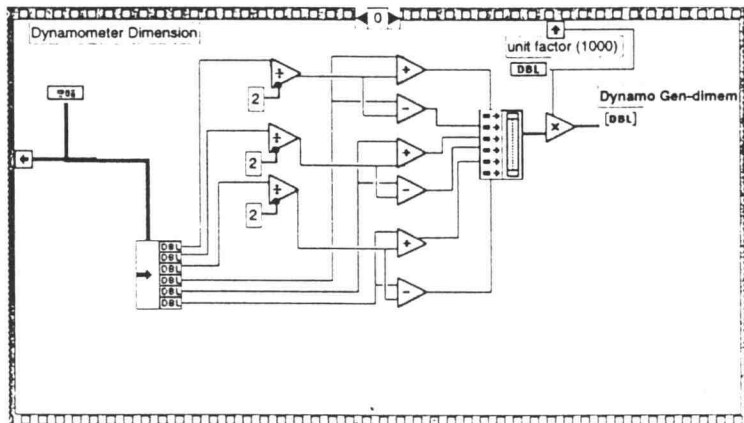
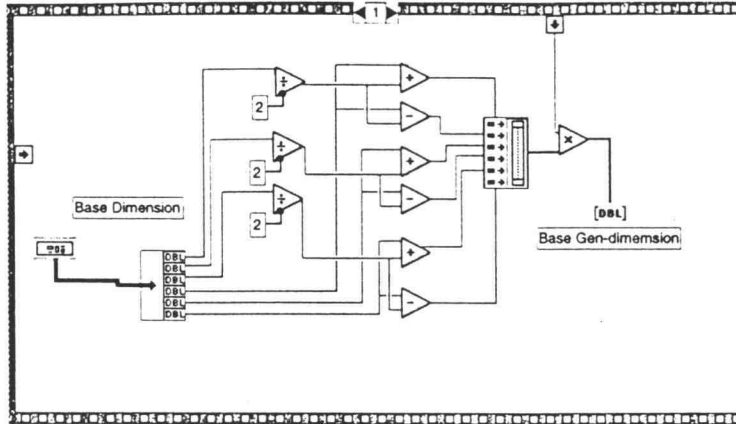


Plate dimension
 0: L pos (i)
 1: L neg (i)
 2: W pos (j)
 3: W neg (j)
 4: H pos (k)
 5: H neg (k)



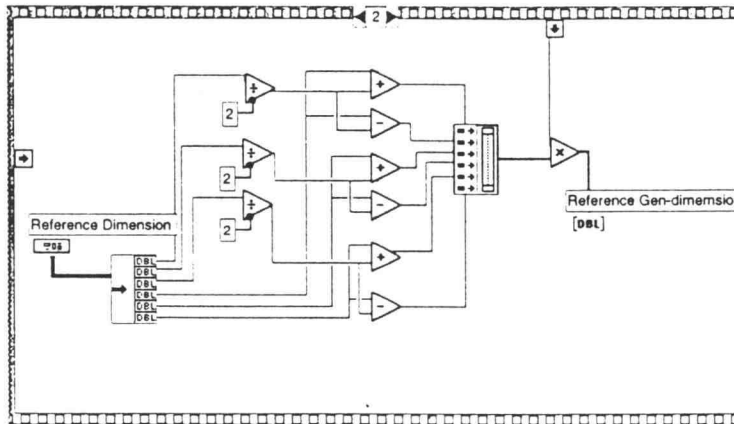
Center reference for gen-dimension2.vi
05/30/97 01:31 AM

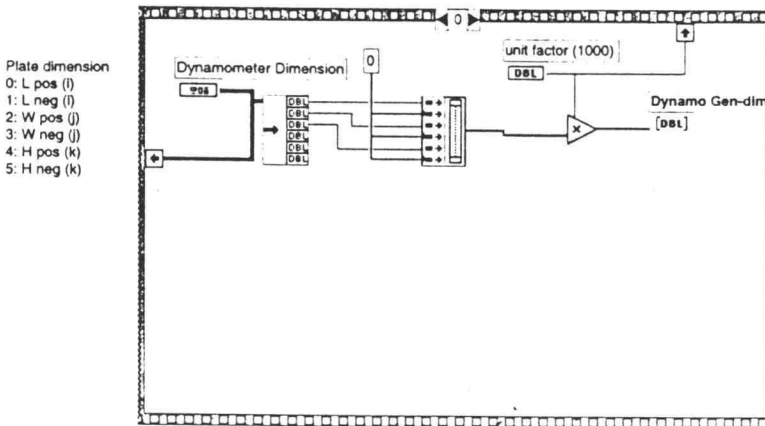
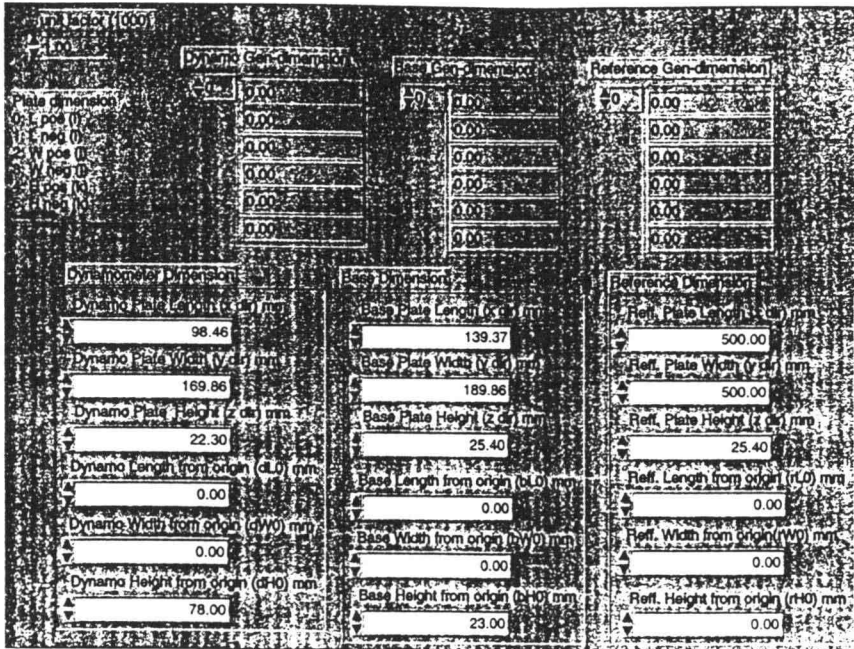
Page 3 



Center reference for gen-dimension2.vi
05/30/97 01:31 AM

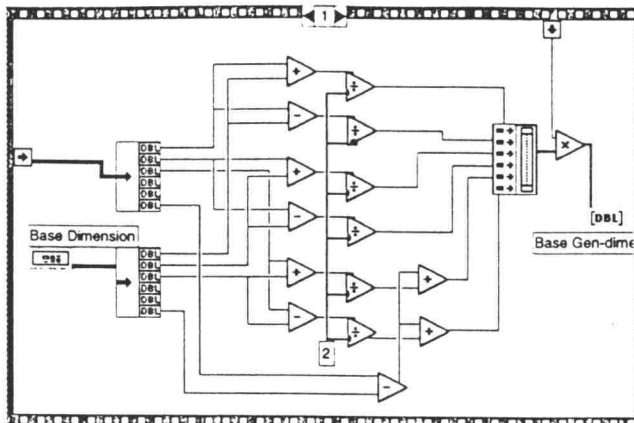
Page 4 





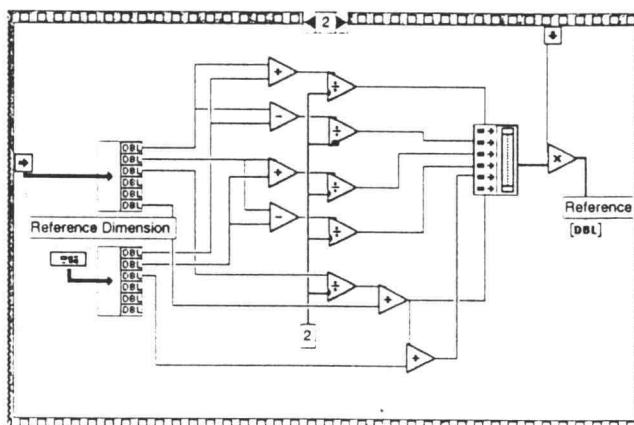
Top-corner reference for gen-dimension 2.vi
05/30/97 01:48 AM

Page 3 

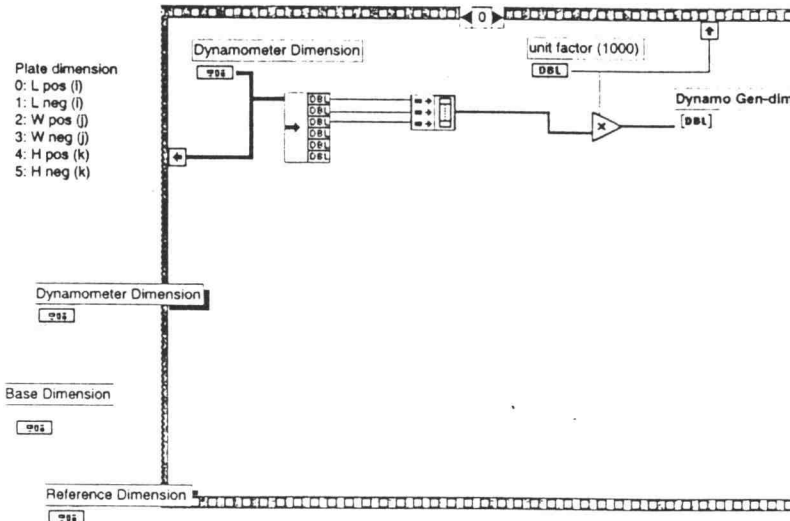
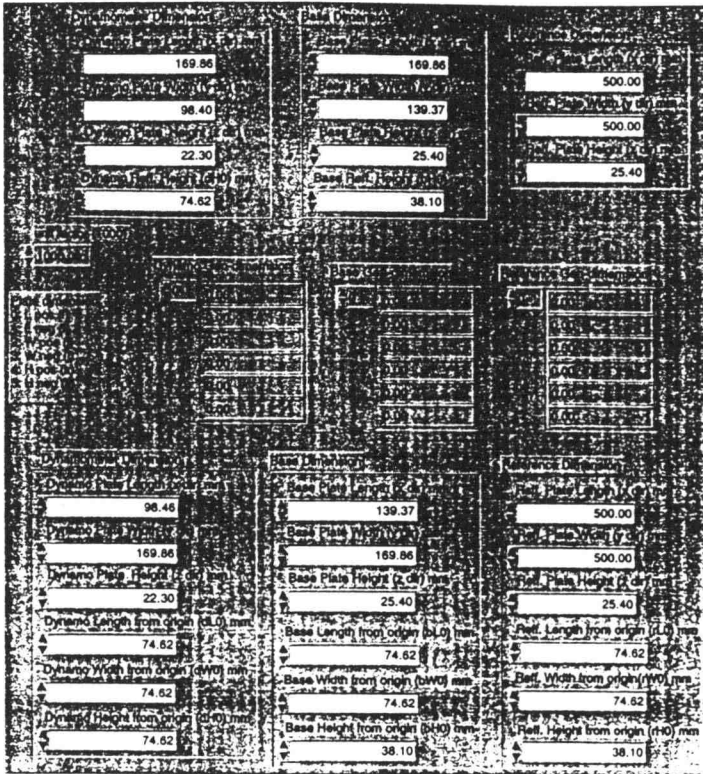


Top-corner reference for gen-dimension 2.vi
05/30/97 01:48 AM

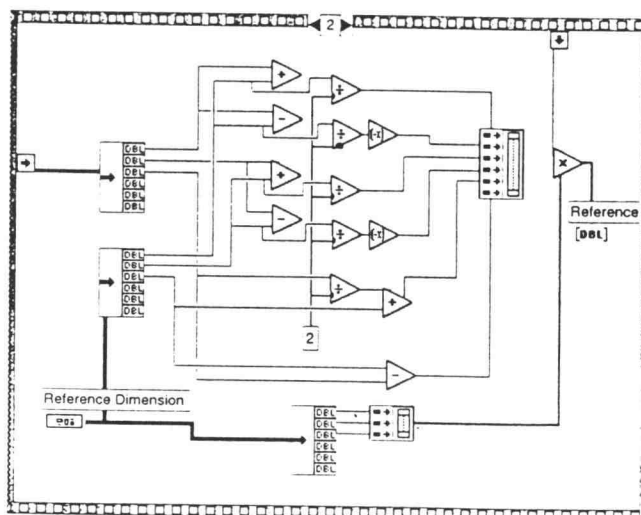
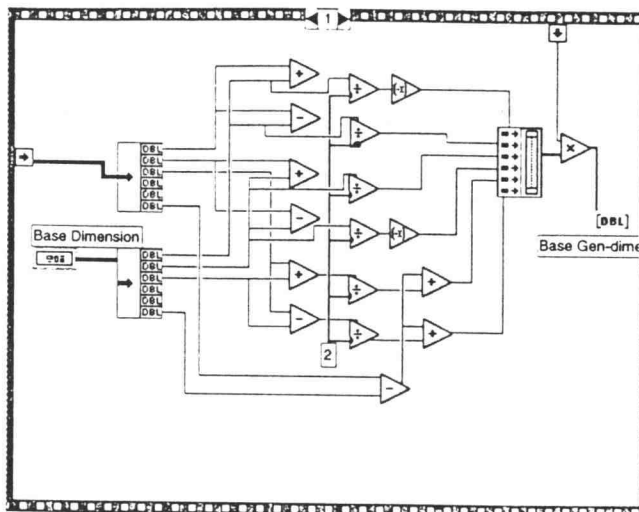
Page 4 



VIBRATION VISUALIZATION 352

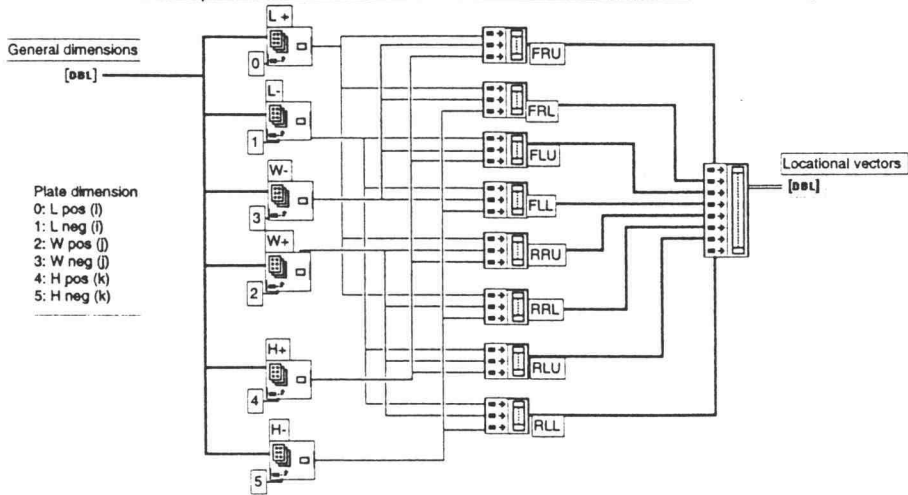
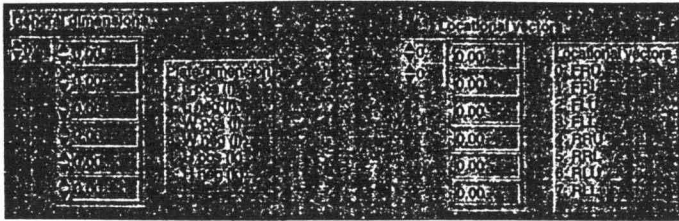


VIBRATION VISUALIZATION 39

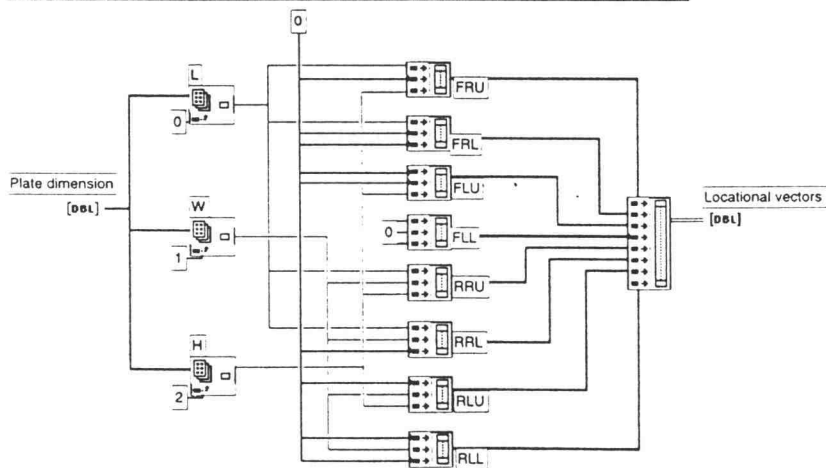
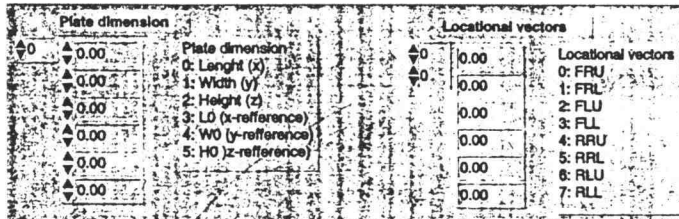


VIBRATION VISUALIZATION 40

Make Location Vectors from center.vi
05/30/97 01:41 AM

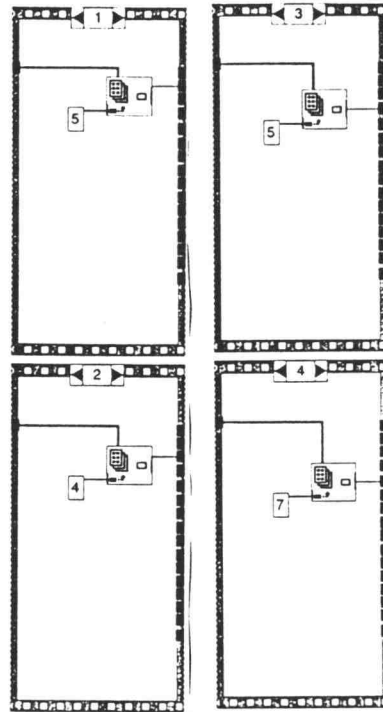
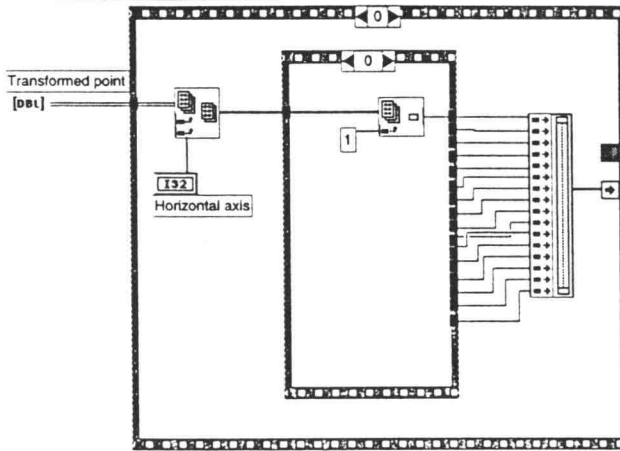
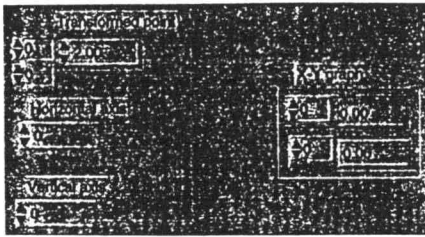


Make Location Vectors from Corner.vi
05/30/97 01:47 AM

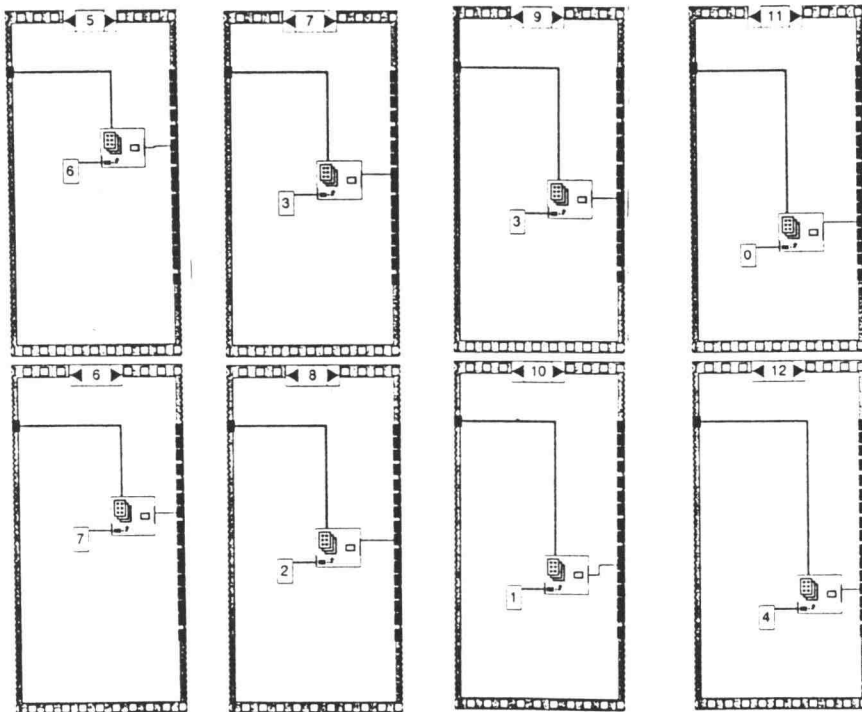


VIBRATION VISUALIZATION 41

Draw a box.vi
05/30/97 01:36 AM




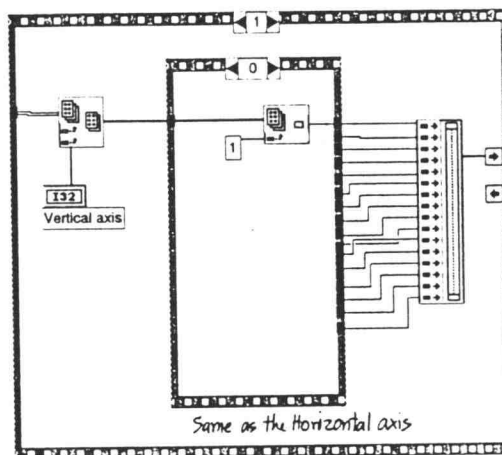
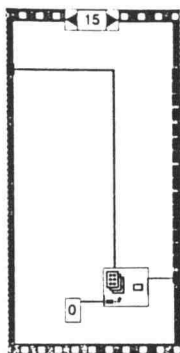
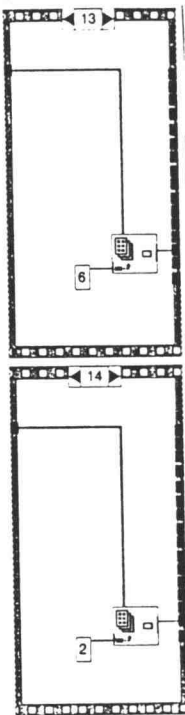
Draw a box.vi
05/30/97 01:36 AM




VIBRATION VISUALIZATION 4 2

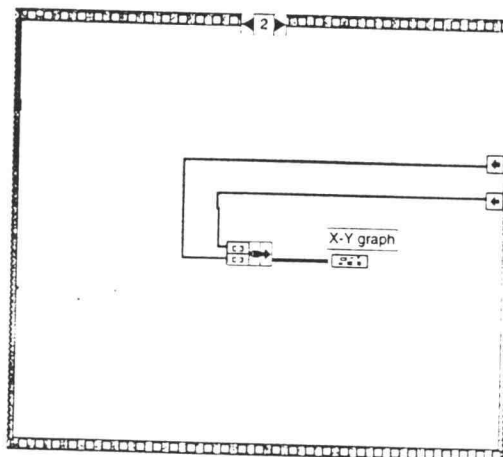
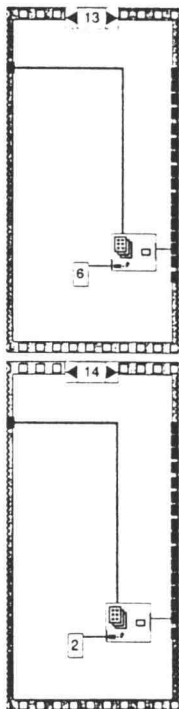
Draw a box.vi
05/30/97 01:36 AM

Page 9 



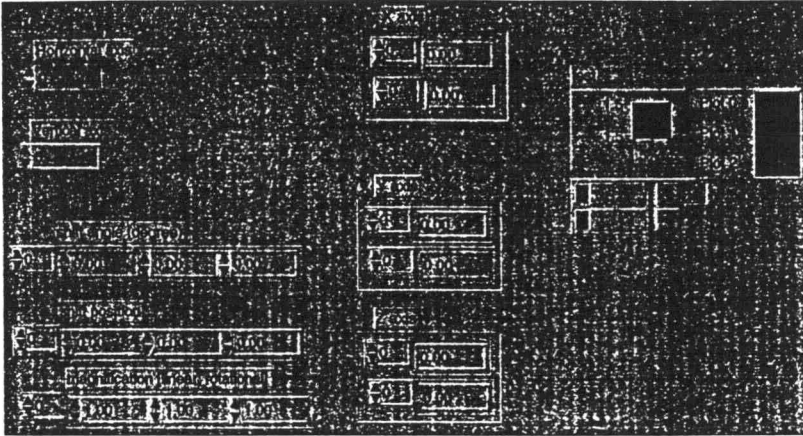
Draw a box.vi
05/30/97 01:36 AM

Page 18 



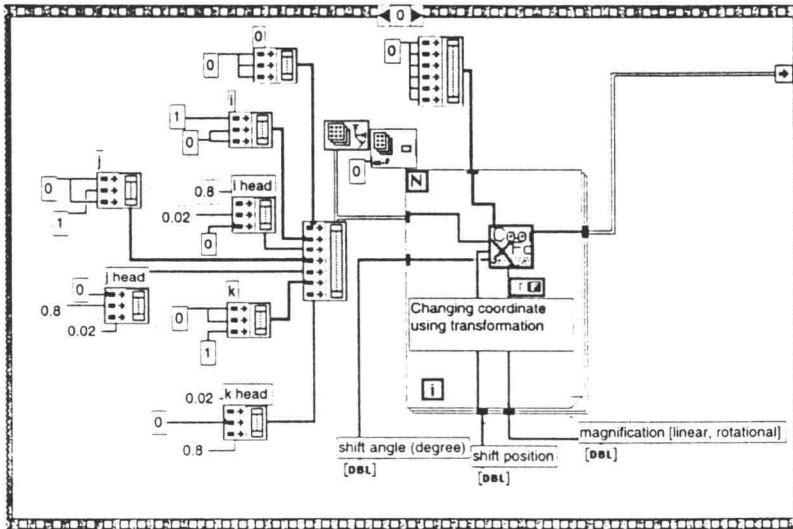
Draw axes.vi
05/30/97 01:39 AM

Page 1 




Draw axes.vi
05/30/97 01:39 AM

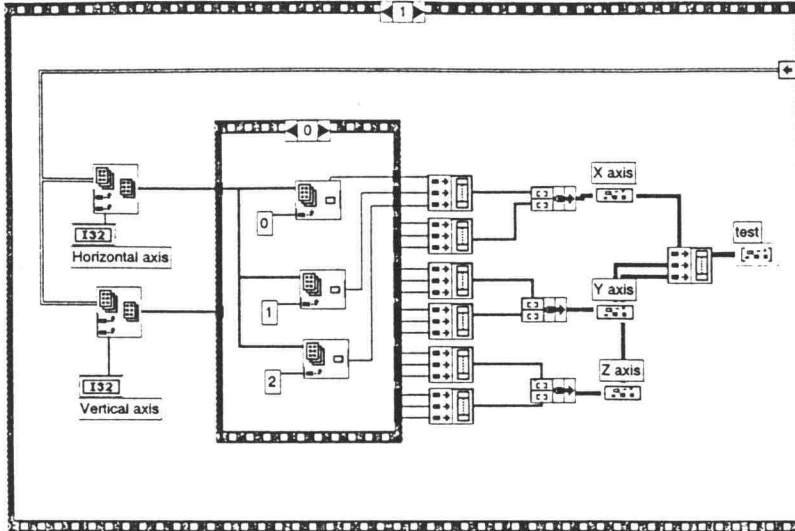
Page 2 




VIBRATION VISUALIZATION 44

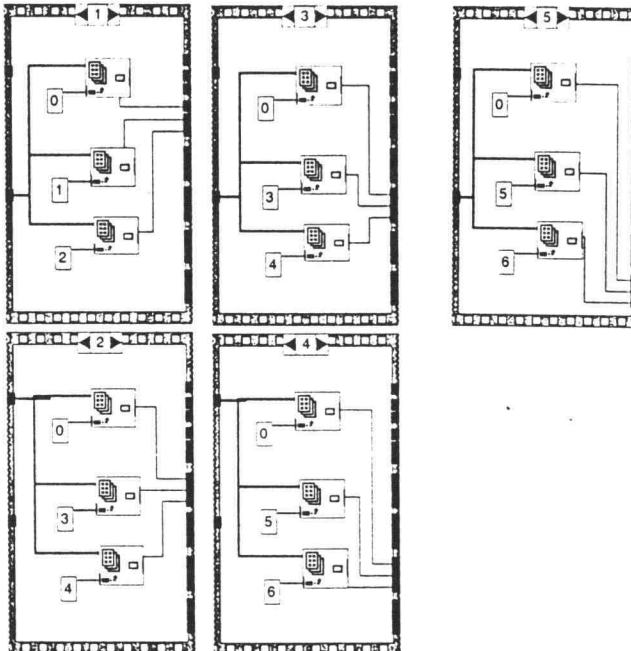
Draw axes.vi
05/30/97 01:39 AM

Page 3 



Draw axes.vi
05/30/97 01:39 AM

Page 5 



VIBRATION VISUALIZATION 45